

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หุ่นยนต์เตะฟุตบอล 3 มิติ

3D SOCCER ROBOT



รฟ.
ค 583 น
2049

เลขหมู่.....
เลขทะเบียน..... 72855
วัน,เดือน,ปี... 2.3... ส.ย... 2550

b. 11773310
i.....

ปฏิญานិพนธ์นี้เป็นส่วนหนึ่งของการศึกษาคามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์เตะฟุตบอล 3 มิติ 3D SOCCER ROBOT



**ปริญญาบัตรสำหรับปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2549

ภาควิชา วิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง หุ่นยนต์เตะฟุตบอลสามมิติ (3D SOCCER ROBOT)

ผู้จัดทำ

1.นายสันติ อ่อนพุก รหัสนักศึกษา 47015226

2.นายเอกพล วัฒนชัยสิทธิ์ รหัสนักศึกษา 47015274


.....)อาจารย์ที่ปรึกษา
(รศ.ดร.ชูชาติ ปินทวโรจน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์เตะฟุตบอล 3 มิติ

นาย สันติ อ่อนพุก รหัส47015226
นาย เอกพล วัฒนชัยสิทธิ์ รหัส47015274
รศ.ดร.ชูชาติ พิษขวิรุจน์ อาจารย์ที่ปรึกษา
ปีการศึกษา2549

บทคัดย่อ

ปฏิญานิพนธ์ฉบับนี้อธิบายถึงการออกแบบและการสร้างหุ่นยนต์เตะฟุตบอลสามมิติซึ่งส่วนประกอบของหุ่นยนต์นี้ประกอบไปด้วยส่วนของตัวหุ่นยนต์และส่วนของคอมพิวเตอร์ประมวลผล ซึ่งส่วนของคอมพิวเตอร์ประมวลผลนี้จะรับค่าจากกล้องมาทำการประมวลผลทางภาพแบบสามมิติ แล้วส่งออกที่พอร์ตนานของคอมพิวเตอร์ไปเข้าสู่วงจรเข้ารหัสส่งโดยใช้คลื่นวิทยุ ส่วนชุดรับซึ่งติดอยู่กับตัวหุ่นยนต์จะรับและถอดรหัสออกมาให้ไมโครคอนโทรลเลอร์ทำตามโปรแกรมคำสั่งเพื่อสั่งให้วงจรขับเคลื่อนบังคับให้หุ่นยนต์ เลี้ยวซ้าย เลี้ยวขวา เดินหน้าและถอยหลัง เพื่อไปเตะลูกบอลเข้าประตู โดยใช้ขดลวดโซลินอยด์ในการเตะลูกบอล

3D SOCCER ROBOT

Mr. Santi Onphuk ID.47015226

Mr. Ekapon Wattanachaisit ID.47015274

Assoc. Prof. Dr. Chuchat Pintawirut Advisor

Educational Year 2006

Abstract

This report presents a design and a construction of 3 dimension soccer robot which consists of the body and the image processing by computer. This will receive the digital data from cameras to the 3 dimension image processing and send it through parallel port to radio frequency transmitter. The radio frequency receiver and decoder will decode the data to the microcontroller which will drive robot to turn-left, turn-right, forward and backward by the drive motor circuit. This process receives image processing with a computer that moves the robot to the ball and kicks it toward the goal by using solenoid.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของหลักสูตรคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ภาควิชาวิศวกรรมอิเล็กทรอนิกส์ ขอขอบคุณสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่เปิดโอกาส ให้ข้าพเจ้าได้มีโอกาสเข้ามาศึกษาหาความรู้ในที่แห่งนี้ ซึ่งข้าพเจ้ามีความภาคภูมิใจเป็นอย่างยิ่ง ขอขอบคุณ ครอบครัวอ่อนทุก และครอบครัว วัฒนชัยสิทธิ์ ที่ได้ให้กำเนิด กำลังใจ และส่งเสริมข้าพเจ้าเข้ามาในการศึกษามาโดยตลอด ขอขอบคุณอาจารย์ ชูชาติ ปิณฑวิรุจน์ อาจารย์ที่ปรึกษา ที่ช่วยให้คำแนะนำดีๆ มาโดยตลอด ขอขอบคุณอาจารย์ภาควิชาวิศวกรรมอิเล็กทรอนิกส์ทุกๆ ท่าน ที่ให้ความรู้และคำแนะนำดีๆ มาโดยตลอด ขอขอบคุณเพื่อนๆ ร่วมชั้นเรียนห้อง R2 และห้อง R3 ทุกๆท่าน สำหรับมิตรภาพที่ดีๆ และคำปรึกษาต่างๆ ทำให้ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี และขอขอบคุณผู้มีพระคุณทุกๆ ท่านที่ไม่ได้กล่าวถึงมา ณ ที่นี้ด้วย

(สันติ อ่อนพุก)

นายสันติ อ่อนพุก

(Signature)

นายเอกพล วัฒนชัยสิทธิ์

สารบัญ

	หน้า
บทที่ 1 บทนำ	
1.1 ความเป็นมาและวัตถุประสงค์	1
1.2 ขอบเขตของงาน	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	1
บทที่ 2 ไมโครคอนโทรลเลอร์	
2.1 ความหมายของไมโครคอนโทรลเลอร์	2
2.2 ความแตกต่างของไมโครคอนโทรลเลอร์กับไมโครโปรเซสเซอร์	2
2.3 ส่วนประกอบพื้นฐานของไมโครคอนโทรลเลอร์	5
2.4 โครงสร้างและสถาปัตยกรรมของไมโครคอนโทรลเลอร์ AVR	7
2.5 คุณสมบัติของไมโครคอนโทรลเลอร์ AVR เบอร์ ATmega8-16PI	9
2.6 การจัดการและคุณสมบัติของพอร์ตต่างๆ	11
2.6.1 การจัดการของไมโครคอนโทรลเลอร์ AVR เบอร์ ATmega8-16PI	12
2.6.2 โครงสร้างและการทำงานของพอร์ตไมโครคอนโทรลเลอร์ AVR	13
2.7 หน่วยความจำและชุดคำสั่ง	16
บทที่ 3 การประมวลผลภาพ	
3.1 หลักการพื้นฐานทางอิมเมจโปรเซสซิ่ง	21
3.1.1 พิกเซล (Pixel)	21
3.1.2 ลักษณะข้อมูลภาพ	21
3.1.3 เกรย์สเกล (Gray scale)	21
3.1.4 ฮิสโตแกรม (Histogram)	23
3.1.5 ความหมายของภาพและนิยามของภาพในระบบดิจิทัล	23
3.1.6 ระบบการประมวลผลทางดิจิทัล	24
3.1.7 การสร้างภาพไบนารี (Binary)	25
3.2 เทคนิคการประมวลผลภาพ	26
3.2.1 อิมเมจเอนฮานซ์เมนต์ (Image enhancement)	26
3.2.2 การแยกวัตถุออกจากภาพ (Segmentation)	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
3.2.3 การหาจุดกึ่งกลางของวัตถุ (Center of Mass)	28
3.2.4 วิธีการปรับปรุงคุณภาพของภาพ (Enhancement)	29
3.3 การเห็นใน 3 มิติ(3D Vision)	30
3.3.1 การแปลงเรขาคณิตของวัตถุใน 3 มิติ (Geometric Transformation in 3D)	30
3.3.2 เรขาคณิตสำหรับการเห็นใน 3 มิติ (Geometry for 3D Vision)	33
3.4 การปรับเทียบกล้อง (Camera Calibration)	39
3.5 การหาพิกัด 3 มิติของจุดในฉากโดยใช้กล้อง 2 ตัว	41
3.6 การเกิดภาพสเตอริโอ (Stereo Imaging)	44
บทที่ 4 การรับส่งข้อมูล	
4.1 ทำความเข้าใจกับพอร์ต(Port) คอมพิวเตอร์ที่ใช้งาน	47
4.1.1 พอร์ตนาน (Parallel Port)	47
4.1.2 พอร์ตยูเอสบี (USB Port)	48
4.2 การรับส่งข้อมูลผ่านความถี่วิทยุ	48
4.2.1 การมอดูเลตแบบASK	49
บทที่ 5 การออกแบบและการสร้างวงจร	
5.1 บล็อกไดอะแกรมและการทำงาน	53
5.2 หลักการออกแบบและการสร้างวงจร	55
5.2.1 แหล่งจ่ายไฟ (Power Supply)	55
5.2.2 ชุดส่งข้อมูลแบบใช้ความถี่วิทยุ (Wireless Transmitter)	55
5.2.3 ชุดรับข้อมูลแบบใช้ความถี่วิทยุ (Wireless Receiver)	58
5.2.4 ไมโครคอนโทรลเลอร์ (Microcontroller)	59
5.2.5 วงจรขับเคลื่อนมอเตอร์ (Drive Motor)	60
5.2.6 ชุดต่อในการลงโปรแกรมบนตัวไมโครคอนโทรลเลอร์ (In-System Programming)	61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 6 การทดลองและสรุปผล

6.1 การควบคุมหุ่นยนต์	66
การทดลองที่ 6.1 มุมเลี้ยวของหุ่นยนต์	67
การทดลองที่ 6.2 ระยะทางการเคลื่อนที่ของหุ่นยนต์	68
การทดลองที่ 6.3 การเคลื่อนที่ไปยังพิกัดที่กำหนด	68
6.2 การปรับเทียบกล้อง(Camera Calibration)	69
การทดลองที่ 6.4 การปรับเทียบโดยใช้โปรแกรม Matlab	69
การทดลองที่ 6.5 การปรับเทียบโดยใช้โปรแกรม DELPHI	71
การทดลองที่ 6.6 การปรับเทียบกล้อง 2 ตัว(Camera Calibration with 2 Cameras)	73
การทดลองที่ 6.7 การควบคุมหุ่นยนต์ด้วยมือ (Manual Control)	74

ภาคผนวก

สารบัญรูป

	หน้า
รูป 2.1 โครงสร้างพื้นฐานของไมโครโปรเซสเซอร์	3
รูป 2.2 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์	3
รูป 2.3 ส่วนประกอบของไมโครคอนโทรลเลอร์	5
รูป 2.4 แสดงวงจรการใช้งานร่วมกันของADC และI/O	7
รูป 2.5 บล็อกไดอะแกรมของ AVR เบอร์ ATmega8-16PI	8
รูป 2.6 แสดงการจัดขาของ AVR เบอร์ ATmega8-16PI	11
รูป 2.7 โครงสร้างการทำงานภายในของพอร์ต B	14
รูป 2.8 โครงสร้างการทำงานภายในของพอร์ต C	15
รูป 2.9 โครงสร้างการทำงานภายในของพอร์ต D	16
รูป 3.1 ระบบการประมวลผลทางดิจิทัล	24
รูป 3.2 Block Diagram การประมวลผลของ Soccer Robot	26
รูป 3.3 แกนพิกัด 3 มิติและการหมุนจตุรรอบแกนพิกัด มุมของการหมุน วัดตามเข็มนาฬิกาเมื่อมองจากแกนสู่จุดกำเนิด	31
รูป 3.4 แสดงตัวอย่างการแปลงเรขาคณิตของวัตถุใน 3 มิติ	33
รูป 3.5 แบบจำลองของระบบสร้างภาพ	34
รูป 3.6 พิกัดของภาพ	36
รูป 3.7 a) แบบจำลองการเกิดภาพจากกล้อง b) เรขาคณิตของการเกิดภาพ	37
รูป 3.8 ภาพของตารางหมากรุกที่นำมาใช้ในการเปรียบเทียบกล้อง	40
รูป 3.9 การหาพิกัด 3 มิติโดยใช้กล้อง 3 ตัว	43
รูป 3.10 แบบจำลองของขบวนการสร้างภาพสเตอริโอ	45
รูป 3.11 มุมบนของขบวนการสร้างภาพสเตอริโอ โดยที่ระบบพิกัดของกล้องตัวที่ 1 ซ้อนทับกับระบบพิกัดของโลก	45
รูป 4.1 พอร์ตขนาน (Parallel Port)	47
รูป 4.2 พอร์ตยูเอสบี (USB Port)	48
รูป 4.3 รูปแสดงมอดูเลชันของสัญญาณ(Sinusoidal) กับ แครเรียร์ ที่ผลิตออกมาเรียกว่า ไซด์แบน (Sidebands)	49

สารบัญรูป(ต่อ)

	หน้า
รูป 4.4 รูปแสดงการมอดูเลต ASK ของสัญญาณแคเรียร์ที่สวิตซ์ระหว่างตำแหน่งของสัญญาณไบนารี โคลด์	50
รูป 4.5 รูปแสดงการมอดูเลตแบบASK 100% ด้วยสัญญาณแคเรียร์ที่สร้างขึ้นมา กับสัญญาณที่ Keying ด้วยสัญญาณไบนารี	51
รูป 4.6 แสดงคาบเวลา T และช่วงเวลาระหว่างบิต τ	51
รูป 5.1 บล็อกไดอะแกรมภาคส่ง(Transmitter)	53
รูป 5.2 บล็อกไดอะแกรมภาครับ (Receiver)	54
รูป 5.3 วงจรจ่ายไฟของไมโครคอนโทรลเลอร์	55
รูป 5.4 วงจรภาคส่ง(Wireless Transmitter)	56
รูป 5.5 วงจรส่งสัญญาณที่ออกแบบโดยโปรแกรม Protel 99	57
รูป 5.6 ลายทองแดงของวงจรส่งสัญญาณ	57
รูป 5.7 วงจรส่งสัญญาณที่ลงอุปกรณ์เสร็จแล้ว	58
รูป 5.8 วงจรภาครับ(Wireless Receiver)	59
รูป 5.9 วงจรไมโครคอนโทรลเลอร์(Microcontroller)	60
รูป 5.10 วงจรขับเคลื่อนมอเตอร์ (Drive Motor)	61
รูป 5.11 วงจร ISP (In-System Programming)	62
รูป 5.12 ตำแหน่งขาของหัวต่อ DB25	62
รูป 5.13 วงจรของตัวหุ่นยนต์	63
รูป 5.14 วงจรที่ออกแบบโดยโปรแกรม Protel 99	64
รูป 5.15 ลายทองแดงของวงจรที่ตัวหุ่นยนต์	64
รูป 5.16 รูปตัวหุ่นที่ประกอบวงจรแล้ว	64
รูป 6.1 แสดงลักษณะการเคลื่อนที่ของหุ่นยนต์	66
รูป 6.2 แสดงวัตถุอ้างอิง	69
รูป 6.3 แสดงหน้าต่าง ของโปรแกรมMATLAB	70
รูป 6.4 แสดงผลลัพธ์ จากโปรแกรม MATLAB	70
รูป 6.5 แสดงหน้าต่างของ โปรแกรม	71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

	หน้า
รูป 6.6 แสดงการทำงานของโปรแกรมในส่วนของการเลือกพิกัดจากภาพตัวอย่าง	72
รูป 6.7 แสดงภาพผลลัพธ์ที่ได้หลังจากผ่านการประมวลผลแล้ว	72
รูป 6.8 หน้าต่างโปรแกรมที่ทำการปรับเทียบกล้อง 2 ตัว	73
รูป 6.9 ภาพแสดงหน้าต่างการปรับเทียบกล้องที่สมบูรณ์แล้ว	74
รูป 6.10 หน้าต่างโปรแกรมที่ใช้ส่งงานหุ่นยนต์	75
รูป 6.11 หน้าต่างโปรแกรมที่ส่งงานหุ่นยนต์ด้วยมือ	75



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตาราง 2.1 เปรียบเทียบคุณสมบัติของไมโครคอนโทรลเลอร์และไมโครโปรเซสเซอร์	4
ตาราง 2.2 สถานะแสดงการทำงานของพอร์ต B	13
ตาราง 2.3 สถานะแสดงการทำงานของพอร์ต C	15
ตาราง 2.4 สถานะแสดงการทำงานของพอร์ต D	16
ตาราง 2.5 แสดงแอดเดรสรีจิสเตอร์	17
ตาราง 2.6 แสดงชุดคำสั่ง Instruction Mnemonic	18
ตาราง 4.1 Spectral lines สำหรับรูปคลื่นพัลส์ที่มอดูเลตกับความถี่แคเรียร์	52
ตาราง 6.1 แสดงคำสั่งที่ไมโครคอนโทรลเลอร์รับเข้ามา	66
ตาราง 6.2 คำสั่งของไมโครคอนโทรลเลอร์ที่ส่งออกเอาต์พุต	67
ตาราง 6.3 แสดงค่าความผิดพลาดของมุมเลี้ยวต่างๆ	68
ตาราง 6.4 แสดงค่าความผิดพลาดของระยะทางค่าต่าง	68
ตาราง 6.5 แสดงค่าความผิดพลาดของระยะที่กัคx, yต่าง	68

บทที่ 1

บทนำ

1.1 ความเป็นมาและวัตถุประสงค์

เนื่องจากในปัจจุบัน ได้มีการศึกษาเกี่ยวกับการประมวลผลทางภาพ(Digital Image Processing) เพื่อนำไปใช้งานได้หลายรูปแบบ เช่นการวิเคราะห์ภาพถ่ายทางอากาศ การส่งภาพและเก็บภาพสำหรับงานธุรกิจ การประมวลผลภาพทางการแพทย์ การประมวลผลภาพถ่ายทางเรดาร์ ไซนาร์ และสัญญาณความถี่ย่านเสียง(Acoustic) งานด้านหุ่นยนต์ และการตรวจสอบอัตโนมัติทางขึ้นส่วนทางอุตสาหกรรม จากการศึกษาที่หลักสูตรสาขาวิศวกรรมอิเล็กทรอนิกส์ได้กำหนดให้มีการทำปริญญานิพนธ์(Project) ดังนั้นผู้จัดทำจึงได้เลือกที่จะศึกษาทางด้านหุ่นยนต์ซึ่งใช้การประมวลผลทางภาพเป็นแบบสามมิติ ซึ่งมีวัตถุประสงค์เพื่อศึกษาเกี่ยวกับการประมวลผลภาพทางดิจิทัล การออกแบบและวิเคราะห์วงจร การเขียน โปรแกรมคอมพิวเตอร์ซึ่งใช้โปรแกรม Delphi ในการเขียนการใช้งานไมโครคอนโทรลเลอร์ซึ่งเลือกใช้ของบริษัท ATMEL ที่ชื่อว่า AVR การรับ-ส่งข้อมูลแบบบลูทูธซึ่งเป็นการรับ-ส่งข้อมูลด้วยความถี่วิทยุซึ่งเป็นแบบASK(Amplitude Shift Keying) และการรับ-ส่งข้อมูลผ่านพอร์ทขนาน

1.2 ขอบเขตของงาน

หุ่นยนต์สามารถวิ่งไปหาลูกบอลแล้ววิ่งไปที่ประตู แล้วยิงเข้าประตูได้โดยอัตโนมัติ ซึ่งใช้การประมวลผลทางภาพแบบสามมิติ ที่รับภาพมาจากกล้องทั้งสองตัวที่ติดตั้งอยู่ข้างสนามนำมาประมวลผล ด้วยคอมพิวเตอร์ในการประมวลผลภาพที่โปรแกรมด้วยภาษาปาสคาล(Delphi) และสั่งงานตัวหุ่นยนต์ด้วยการรับ-ส่งข้อมูลด้วยคลื่นความถี่วิทยุที่สั่งงานผ่านพอร์ทขนานของคอมพิวเตอร์

1.3 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์และผลลัพธ์ที่คาดว่าจะได้รับในการทำปริญญานิพนธ์มีดังนี้

- สามารถใช้โปรแกรมคอมพิวเตอร์ มาทำการประมวลผลภาพแบบสามมิติได้
- สามารถรับ-ส่งข้อมูลผ่านพอร์ทขนานได้
- สามารถใช้ไมโครคอนโทรลเลอร์AVR ในการควบคุมและสั่งงานหุ่นยนต์ได้
- สามารถออกแบบวงจรต่างๆที่เกี่ยวข้อง และใช้โปรแกรม Proteel ในการออกแบบลายวงจร และทำการตรวจวัดวงจรได้

บทที่ 2

ไมโครคอนโทรลเลอร์

2.1 ความหมายของไมโครคอนโทรลเลอร์

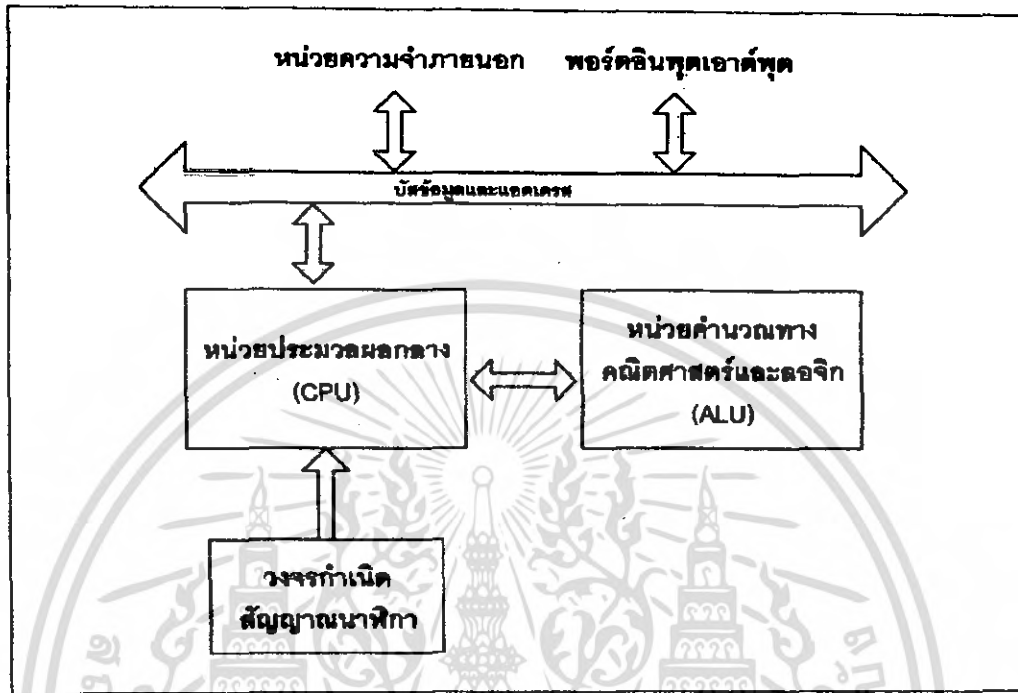
ไมโครคอนโทรลเลอร์ (Microcontroller) เป็นชื่อของอุปกรณ์อิเล็กทรอนิกส์แบบหนึ่งซึ่งรวมเอาหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์และลอจิก วงจรรับสัญญาณอินพุต วงจรขับสัญญาณเอาต์พุต หน่วยความจำ วงจรกำเนิดสัญญาณนาฬิกาไว้ด้วยกัน ทำให้สามารถนำไปใช้งานแทนวงจรอิเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี ช่วยลดจำนวนอุปกรณ์และขนาดของระบบ ในขณะที่มีขีดความสามารถสูงขึ้น ภาวได้งบประมาณที่เหมาะสม

ไมโครคอนโทรลเลอร์มาจากคำ 2 คำรวมกันคือ “ไมโคร” (micro) ซึ่งหมายถึงไมโครโปรเซสเซอร์ (microprocessor) ซึ่งเป็นอุปกรณ์ประมวลผลขนาดเล็ก ภายในประกอบด้วยหน่วยประมวลผลกลางหรือซีพียู (CPU: Central Processing Unit) หน่วยคำนวณทางคณิตศาสตร์ลอจิก (ALU: Arithmetic Logic Unit) วงจรเชื่อมต่อหน่วยความจำ และวงจรสัญญาณนาฬิกา อีกคำหนึ่งคือคำว่า “คอนโทรลเลอร์” (Controller) หมายถึง อุปกรณ์ควบคุม ดังนั้น ไมโครคอนโทรลเลอร์จึงเป็นอุปกรณ์ที่ใช้ในการควบคุมได้อย่างอิสระ

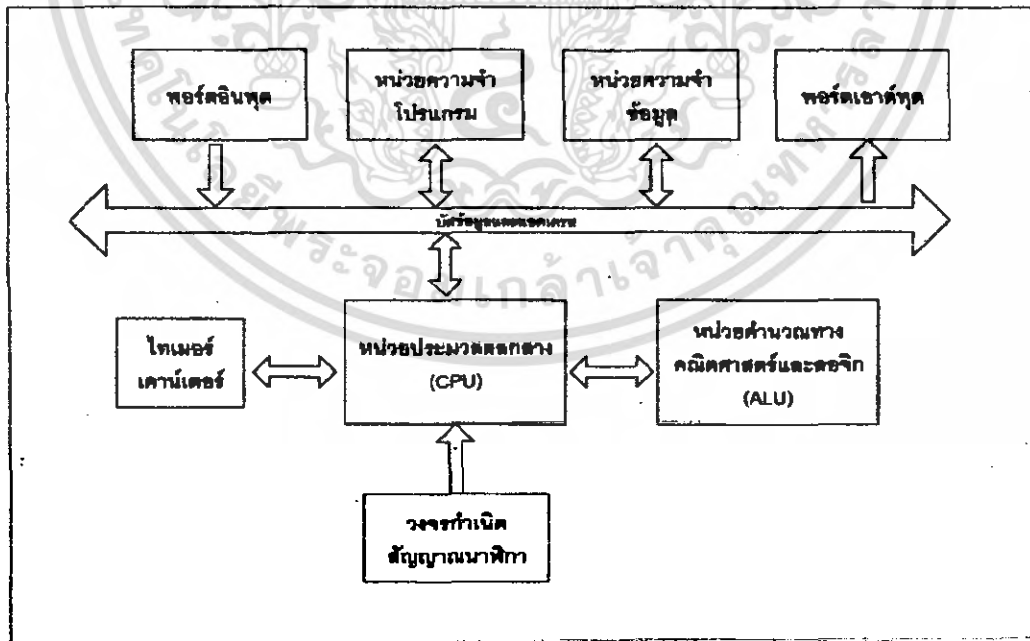
2.2 ความแตกต่างของไมโครคอนโทรลเลอร์กับไมโครโปรเซสเซอร์

ความแตกต่างของไมโครคอนโทรลเลอร์กับไมโครโปรเซสเซอร์ สามารถแบ่งได้ดังนี้ ในรูปที่ 1 แสดงโครงสร้างของไมโครโปรเซสเซอร์ ซึ่งประกอบด้วย หน่วยประมวลผลกลาง, หน่วยคำนวณทางคณิตศาสตร์และลอจิก บัสข้อมูลและแอสเซมบลีสำหรับติดต่อกับหน่วยความจำภายนอก และวงจรมีสัญญาณนาฬิกา นั้นหมายความว่า การใช้งานไมโครโปรเซสเซอร์จะต้องเชื่อมต่อกับหน่วยความจำโปรแกรมภายนอก และถ้าต้องการเชื่อมต่ออุปกรณ์อินพุตและเอาต์พุต ต้องอาศัยอุปกรณ์ที่เรียกว่า “ไอซีขยายพอร์ต” (port expander) ทำให้การสร้างระบบควบคุมจึงต้องการอุปกรณ์จำนวนมากส่งผลให้ขนาดของระบบใหญ่มากพอสมควร ในรูปที่ 2 แสดงโครงสร้างของไมโครคอนโทรลเลอร์ จะเห็นได้อย่างชัดเจนว่า ภายในไมโครคอนโทรลเลอร์มีอุปกรณ์พื้นฐานเหมือนกับไมโครโปรเซสเซอร์ หากแต่จะบรรจุหน่วยความจำโปรแกรมหน่วยความจำข้อมูลและพอร์ตอินพุตเอาต์พุตไว้ภายใต้พร้อมสรรพ ผู้ใช้งานเพียงแต่เขียนโปรแกรมควบคุมลงบนตัวไมโครคอนโทรลเลอร์ แล้วต่ออุปกรณ์ที่ใช้ในการสร้างวงจรมีสัญญาณนาฬิกา อาทิ คริสตัล ตัวเก็บประจุ เป็นต้น ทุคท้ายเชื่อมต่ออุปกรณ์อินพุตเอาต์พุตเข้ากับขาพอร์ต เพียงเท่านั้นก็สามารถใช้งานได้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.1 โครงสร้างพื้นฐานของไมโครโปรเซสเซอร์



รูป 2.2 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตาม ไมโครคอนโทรลเลอร์ก็สามารถเชื่อมต่อกับหน่วยความจำภายนอกได้เช่นกัน โดยพิจารณาให้หน่วยความจำภายนอกนั้นเป็นอุปกรณ์อินพุตเอาต์พุตตัวหนึ่ง แล้วใช้พอร์ตที่มีอยู่ทำการติดต่อ ในตารางที่ 1 เป็นตารางสรุปความแตกต่างที่สำคัญระหว่างไมโครโปรเซสเซอร์ และไมโครคอนโทรลเลอร์

คุณสมบัติ	ไมโครโปรเซสเซอร์	ไมโครคอนโทรลเลอร์
ขนาดของหน่วยประมวลผลกลาง	ไม่น้อยกว่า 8 บิต	ส่วนใหญ่จะมีขนาด 8 บิต
หน่วยคำนวณทางคณิตศาสตร์และลอจิก	มีอยู่ภายใน	มีอยู่ภายใน
วงจรถ่ายโอนข้อมูล	มีอยู่ภายใน	มีอยู่ภายใน
การเชื่อมต่อกับหน่วยความจำโปรแกรม	เชื่อมต่อภายนอกเท่านั้น	ใช้ได้ทั้งภายในและภายนอก
การเชื่อมต่อกับหน่วยความจำข้อมูล	เชื่อมต่อภายนอกเท่านั้น	ใช้ได้ทั้งภายในและภายนอก
การเชื่อมต่อกับพอร์ตอินพุตเอาต์พุต	เชื่อมต่อภายนอกเท่านั้น	มีอยู่ภายใน และสามารถขยายได้
ไทมเมอร์/เคาน์เตอร์	ไม่มีในชิปขนาดเล็ก	มีอย่างน้อย 1 ตัว ขนาด 8-16 บิต
วอตช์ดอกไทมเมอร์	ไม่มีในชิปขนาดเล็ก	มีอย่างน้อย 1 ตัว
จำนวนขาต่อใช้งาน	ไม่น้อยกว่า 40 ขา	มีตั้งแต่ 8 ขาขึ้นไป

ตาราง 2.1 เปรียบเทียบคุณสมบัติของไมโครคอนโทรลเลอร์และไมโครโปรเซสเซอร์ จะเลือกใช้ไมโครโปรเซสเซอร์หรือไมโครคอนโทรลเลอร์เมื่อใดนั้น ถ้าเป็นการสร้างระบบควบคุมขนาด 8 บิต มีความต้องการเชื่อมต่อกับอุปกรณ์ภายนอกไม่มากนัก (น้อยกว่า 10 แบบ) ควรเลือกใช้ไมโครคอนโทรลเลอร์และถ้าหากต้องมีการประมวลผลข้อมูลจำนวนมาก ต้องใช้หน่วยความจำโปรแกรมสูงถึง 8 กิโลไบต์ ทั้งยังมีความต้องการเก็บรักษาข้อมูลลงในหน่วยความจำข้อมูลมากเป็นกิโลไบต์ควรออกแบบให้ไมโครคอนโทรลเลอร์ในระบบควบคุมนี้เชื่อมต่อกับหน่วยความจำภายนอก

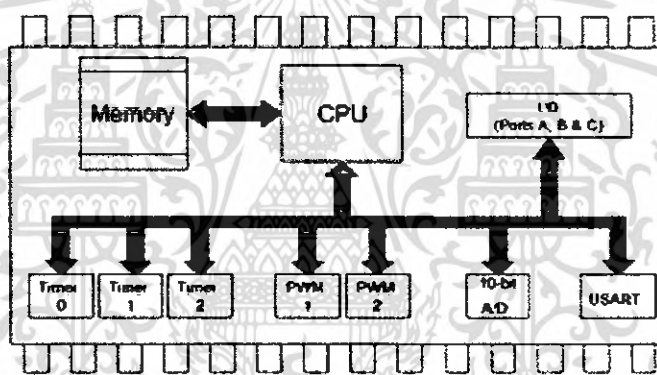
ถ้าหากต้องใช้งานกับข้อมูลมากกว่า 8 บิต ตลอดเวลา และต้องการความเร็วในการทำงานสูง ๆ สามารถติดต่อหน่วยความจำได้เป็นจำนวนมาก ๆ ติดต่อกับอุปกรณ์อินพุตเอาต์พุตได้จำนวนมากภายในเวลาเดียวกันควรเลือกใช้ไมโครโปรเซสเซอร์ ดังนั้นจะเห็นได้จากในเครื่องคอมพิวเตอร์ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ใช้ไมโครโปรเซสเซอร์เป็นหัวใจหลักในการทำงาน ทำให้สามารถเชื่อมต่อกับหน่วยความจำได้มากเป็นหน่วยกิโลไบต์ ขนาดของข้อมูลสูงถึง 64 บิต ความเร็วสูงเป็นหลายร้อยเมกะเฮิรตซ์

ดังนั้นจึงพอสรุปได้ว่า ไมโครคอนโทรลเลอร์เหมาะสำหรับการสร้างระบบควบคุม ในขณะที่ไมโครโปรเซสเซอร์เหมาะสำหรับการสร้างระบบประมวลผลข้อมูลความเร็วสูงและระบบควบคุมที่มีขนาดใหญ่ๆ

2.3 ส่วนประกอบพื้นฐานของไมโครคอนโทรลเลอร์

ซึ่งส่วนประกอบพื้นฐานของไมโครคอนโทรลเลอร์ที่ได้นำมาแสดงต่อไปนี้เป็นส่วนประกอบโดยพื้นฐานที่ไมโครคอนโทรลเลอร์ควรจะต้องมี ส่วนองค์ประกอบส่วนอื่นๆ นอกจากนี้ก็จะขึ้นอยู่กับผู้ผลิตที่จะเพิ่มเข้าไป ซึ่งในปัจจุบันมีมากมายให้เราได้เลือกใช้ ซึ่งส่วนประกอบโดยทั่วไปจะมีส่วนประกอบดังรูปที่ 3



รูป 2.3 ส่วนประกอบของไมโครคอนโทรลเลอร์

ส่วนประกอบของไมโครคอนโทรลเลอร์

1. หน่วยประมวลผลกลาง (CPU : Central Processing Unit) ซึ่งก็จะมีหน้าที่ในการประมวลผลในการทำงานซึ่งก็เสมือนมีไมโครโปรเซสเซอร์อยู่ภายในตัวไมโครคอนโทรลเลอร์นั่นเอง
2. หน่วยความจำ (Memory Unit) ซึ่งหน่วยความจำก็จะมีหน้าที่หลักอยู่สองอย่างคือ เก็บรหัสเลขฐานสองของชุดคำสั่งที่ต้องการให้ไมโครคอนโทรลเลอร์ทำงาน และเก็บข้อมูลที่ต้องการให้ไมโครคอนโทรลเลอร์ทำงานด้วย เช่น ข้อมูลของจำนวนต่างๆที่ทำการนับก่อนส่งข้อมูลต่อไปยังไมโครคอนโทรลเลอร์เพื่อใช้ในการประมวลผลต่อไป ซึ่งก็จะประกอบไปด้วย

- RAM (Random Access Memory) เป็นหน่วยความจำชั่วคราวซึ่งจะสามารถเก็บข้อมูลไว้ได้ถ้าจ่ายไฟเลี้ยงอยู่ และถ้าหยุดจ่ายไฟข้อมูลก็จะหายไป ซึ่งสามารถแบ่งได้เป็น 2 ชนิดคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Static RAM (SRAM) ซึ่งจะใช้ฟลิปฟล็อปในการเก็บข้อมูล และ Dynamic RAM (DRAM) จะใช้ประจุที่อยู่ในตัวเก็บประจุเล็กๆ แทนสถานะของข้อมูล ซึ่งจะทำให้มีขนาดเล็กกว่า และจุข้อมูลได้มากกว่า SRAM

- ROM (Read Only Memory) เป็นหน่วยความจำชนิดแบบถาวร ซึ่งใช้เก็บข้อมูลเพื่อให้ไมโครคอนโทรลเลอร์ทำงานตามคำสั่งที่โปรแกรมไว้ ซึ่งในปัจจุบันมีหลายชนิดเช่น PROM (Programmable Read Only Memory) จะใช้ในกรณีที่เราต้องการ ROM จำนวนเพียงไม่กี่ตัวเท่านั้น ROM แบบนี้จะสามารถเขียนข้อมูลเข้าไปได้ แต่หลังจากเขียนข้อมูลเข้าไปแล้วมันจะอยู่อย่างถาวร, EPROM (Erasable Programmable Read Only Memory) จะเป็น ROM แบบที่เราสามารถที่จะเขียนข้อมูลเข้าไปได้ และสามารถที่จะล้างข้อมูลเหล่านั้นได้หลังจากที่ล้างข้อมูลแล้วก็สามารถที่จะเขียนข้อมูลเข้าไปใหม่ได้ การล้างข้อมูลที่อยู่ใน EPROM ทำได้โดยการใช้แสง UV ส่องลงไปบนวงจรของ EPROM โดยผ่านทางช่อง Quartz ที่อยู่บนตัวไอซี, EEPROM (Electrically Erasable Programmable Read Only Memory) มันจะมีการใช้งานเช่นเดียวกับ EPROM แต่สิ่งที่แตกต่างกันคือการล้างข้อมูลจะทำได้โดยใช้ สัญญาณไฟฟ้าและ Flash EPROM มีลักษณะเช่นเดียวกับ EEPROM แต่ในการลบข้อมูล หรือเขียนข้อมูลเข้าไปใหม่ จะสามารถทำได้ง่ายกว่า
3. หน่วยรับและแสดงผลข้อมูล I/O Port (Input/Output Port)
- พอร์ตอนุกรม (Serial Port) ซึ่งจะทำกรรับหรือแสดงผลข้อมูลเป็นแบบอนุกรม
 - พอร์ตขนาน (Parallel Port) ซึ่งก็จะทำการรับหรือแสดงผลของข้อมูลที่เป็นลักษณะแบบขนานซึ่งจะมีความเร็วมากกว่าแบบอนุกรม
4. ไทม์เมอร์ (Timer) ซึ่งในส่วนนี้สามารถทำงานเป็นไทม์เมอร์หรือเคาน์เตอร์ก็ได้ อยู่ที่ผู้ใช้ต้องการให้เป็น
5. ตัวควบคุมการอินเตอร์รัป (Interrupt Controller) ไว้สำหรับรับค่าการอินเตอร์รัปมาจากภายนอก

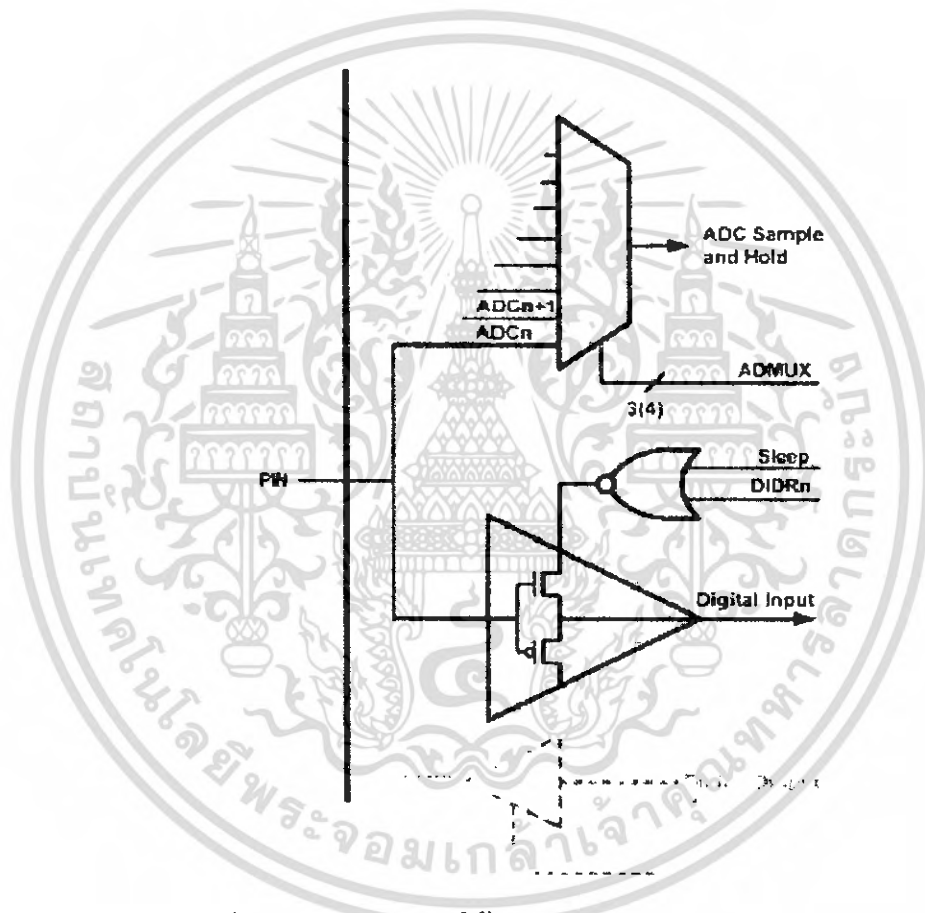
ที่กล่าวมานั้นเป็นองค์ประกอบพื้นฐานที่ไมโครคอนโทรลเลอร์แต่ละเบอร์ควรจะมียกจากนี้ ยังมีความสามารถบางอย่าง ซึ่งขึ้นอยู่กับเบอร์ของไมโครคอนโทรลเลอร์เช่น A/D (Analog to Digital Converter) PWM (Pulse Width Modulator) ฯลฯ ซึ่งปัจจุบันผู้ผลิตได้ผลิตมาหลายอย่าง หลายชนิด ซึ่งเราสามารถเลือกใช้ได้ตามความเหมาะสมของงาน ซึ่งปริณญาณิพนธ์ฉบับนี้ได้เลือกใช้ไมโครคอนโทรลเลอร์ที่สามารถโปรแกรมข้อมูลลงในหน่วยความจำโปรแกรมได้โดยที่ไม่ต้องถอดตัวไมโครคอนโทรลเลอร์ออกมาทำการโปรแกรมใหม่ หรือเรียกว่า การโปรแกรมในวงจร หรือในระบบ (In System Programming) ทำให้การพัฒนาหรือการซ่อมบำรุง ตลอดจนการปรับปรุง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือ อัปเดตข้อมูลในหน่วยความจำโปรแกรมทำได้โดยสะดวก ซึ่งไมโครคอนโทรลเลอร์ชนิดนี้ผลิตโดยบริษัท ATMEL ที่ใช้ชื่อว่า AVR และได้เลือกใช้เบอร์ ATmega8-16PI

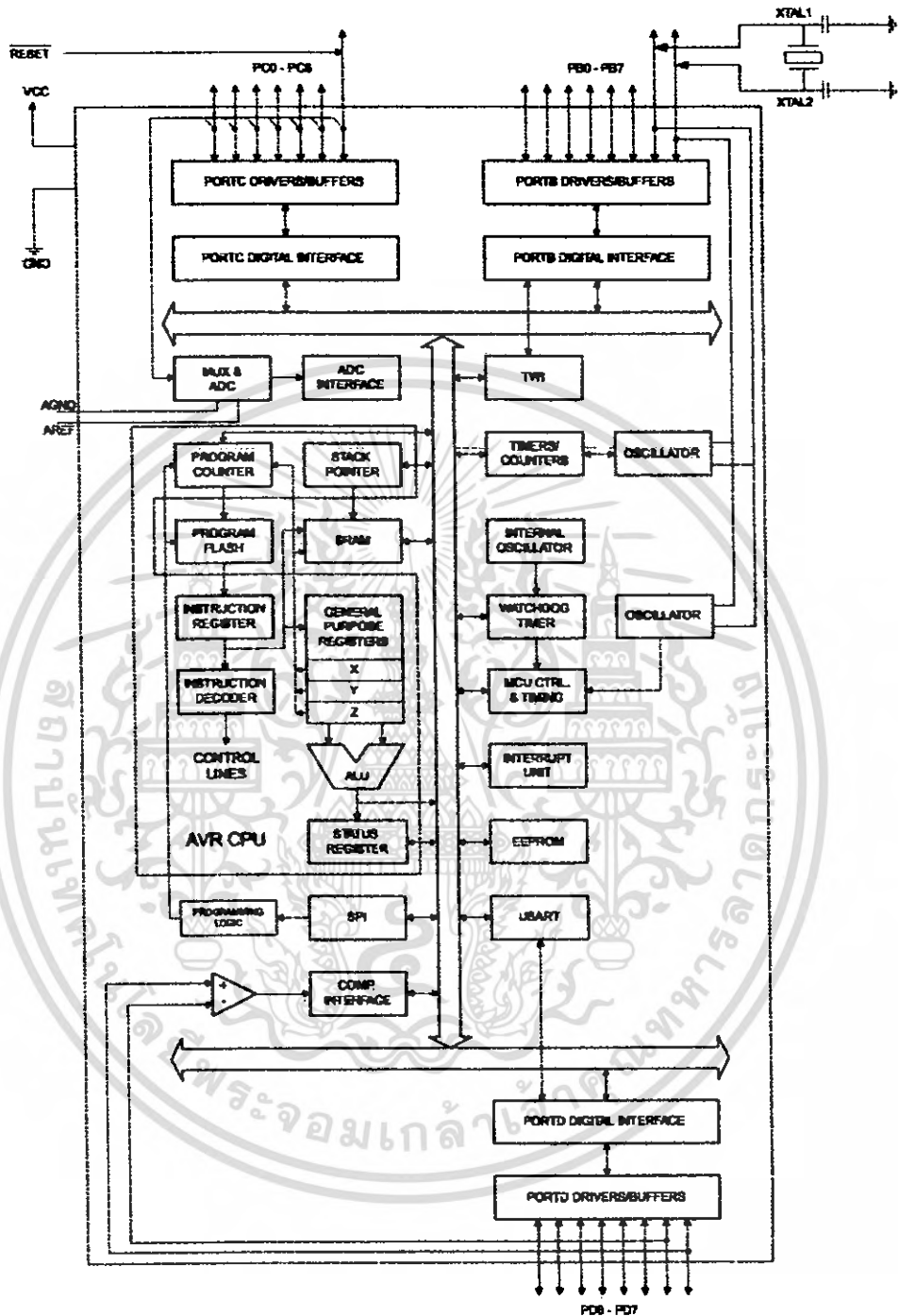
2.4 โครงสร้างและสถาปัตยกรรมของไมโครคอนโทรลเลอร์ AVR

ซึ่งไมโครคอนโทรลเลอร์ AVR เบอร์ ATmega8-16PI ที่เลือกใช้นี้ จะมีการออกแบบที่ใช้สถาปัตยกรรมแบบ RISC (Reduced Instruction Set Computer) และมีการรวมเอาสัญญาณดิจิทัล I/O และ A/D Converter ที่สามารถใช้งานได้ด้วยกัน ซึ่งมีโครงสร้างของวงจรดังแสดงในรูปที่ 2.4



รูป 2.4 แสดงวงจรการใช้งานร่วมกันของ ADC และ I/O

และมีข้อดีที่มีมากกว่าในรุ่นที่แล้วคือมีวงจรผลิตความถี่ภายใน และมีพอร์ตการใช้งานที่มากกว่า แต่สถาปัตยกรรมภายในก็จะคล้ายๆ กับรุ่นก่อนๆ ซึ่งเป็นตระกูลเดียวกันเช่นเบอร์ AT90S2313 ซึ่งปัจจุบันได้เลิกผลิตไปแล้ว และถ้าดูจากภาพของบล็อกไดอะแกรมของ AVR เบอร์ ATmega8-16PI ตามรูปที่ 2.5 ก็จะให้เห็นโครงสร้างอย่างคร่าวๆ ได้



รูป 2.5 บล็อกไดอะแกรมของ AVR เบอร์ ATmega8-16PI

AVR เป็นไมโครคอนโทรลเลอร์ตระกูล 8051 ที่ผลิตและพัฒนาโดย ATMEL ซึ่งจากการใช้การออกแบบสถาปัตยกรรมแบบ RISC (Reduced Instruction Set Computer) โดยการออกแบบสถาปัตยกรรมแบบ RISC โดยทั่วไป จะมีคุณสมบัติดังนี้

- คำสั่งต่อรอบสัญญาณนาฬิกา (One Instruction per Cycle) นับเป็นสิ่งแรกเลยที่นึกถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการออกแบบสถาปัตยกรรมแบบ RISC โดย จะพยายามทำงานให้เสร็จสิ้นภายใน 1 รอบสัญญาณนาฬิกา (1 Clock Cycle) ซึ่งก็มีการใช้ pipeline มาช่วยในการทำงาน ดังนั้น ในการทำงานจริงๆ อาจไม่ใช้ทำงาน 1 คำสั่งเสร็จสิ้นใน 1 รอบสัญญาณนาฬิกา แต่ใช้การนับเวลาในการทำงานของโปรแกรม ซึ่งแน่นอน มีการทำงานหลายๆ คำสั่ง หลายๆ ขั้นตอน แล้วคิดเป็นเวลาเฉลี่ย ซึ่งก็จะได้ประมาณ 1 คำสั่งต่อ 1 รอบสัญญาณนาฬิกา วิธีที่ใช้เพื่อทำให้ได้ 1 คำสั่งต่อรอบสัญญาณนาฬิกา จะทำโดยใช้คำสั่งง่ายๆ ธรรมดาๆ ไม่ใช่เพิ่มรอบสัญญาณนาฬิกาให้มากขึ้น

- กำหนดขนาดของ Instruction ให้มีขนาดที่แน่นอน (Fixed Instruction Length) ถ้าหากว่าจะทำให้มีการทำงานแต่ละคำสั่งให้เสร็จสิ้นภายใน 1 รอบสัญญาณนาฬิกาได้นั้น ก็แน่นอนผู้ออกแบบ RISC ก็จะต้องจำกัดขนาดของคำสั่งด้วย ไม่ให้มีขนาดที่ยาวเกินไป ซึ่งโดยปกติแล้วจะใช้ขนาด 1 Word (จะมีขนาดไม่แน่นอน แล้วแต่เครื่องแต่โดยทั่วไป CPU แบบ RISC จะมีขนาดของ Word = 32 Bit) โดยใน 1 Word นั้น ก็จะกำหนดทุกอย่างอย่างทั้งคำสั่ง, Operation, ว่าจะดึง Operand จากที่ใด, จะให้เก็บผลลัพธ์ (Result) ไว้ที่ใด และ คำสั่งถัดไปอยู่ที่ไหน

- คำสั่งในการเข้าถึงหน่วยความจำหลัก จะใช้แค่ load (ดึงข้อมูล) และ store (เก็บข้อมูล) เท่านั้น สถาปัตยกรรมแบบ RISC นั้น เมื่อจะทำการจัดการกับคำสั่งต่างๆ ก็จำเป็นจะต้องนำ Operand มาเก็บไว้ใน Register ก่อน และ ในแต่ละคำสั่งนั้น ก็ถูกจำกัดไว้ที่ 1 Word ซึ่งก็ไม่เพียงพอต่อการเก็บค่าของ Operand ต่างๆ ดังนั้นจึงใช้การอ้างตำแหน่งในหน่วยความจำแทน แต่ การเข้าถึงหน่วยความจำนั้นต้องเสียเวลาอยู่พอสมควร จึงมีการกำหนดให้ใช้เพียงแค่ 2 คำสั่งเท่านั้น คือการ load และ การ store เพื่อลด traffic ระหว่าง หน่วยประมวลผล และ หน่วยความจำ ส่วนเวลาในการ load หรือ store นั้น ก็ขึ้นอยู่กับขนาดของ Operand

- ใช้การอ้างตำแหน่งแบบตรงๆ ง่ายๆ เพราะการอ้างแบบซับซ้อน ก็แน่นอน ต้องใช้ช่วงเวลานานกว่า ซึ่งการอ้างตำแหน่งแบบ RISC นั้นจะจำกัดอยู่ 2 แบบ คือ แบบอ้างผ่าน Register (Register Indirect) และ Index โดยการอ้างผ่าน Register นั้น Register จะเก็บค่าตำแหน่งไว้ แล้วทำการอ้าง ตำแหน่งนั้นๆผ่าน Register และในแบบ Index จะเป็นการอ้างตำแหน่งจากค่าคงที่ ที่มาในคำสั่งนั้นๆเลย

- ใช้ Operation ที่เรียบง่าย ธรรมดาๆ และ มีไม่กี่ Operation เพื่อจะได้ใช้รอบการทำงานน้อยๆ และรวมไปถึงทรัพยากรของระบบที่ใช้ ก็จะใช้ให้น้อยด้วย

2.5 คุณสมบัติของไมโครคอนโทรลเลอร์ AVR เบอร์ ATmega8-16PI

- AVR เบอร์นี้มีประสิทธิภาพสูงและใช้กำลังงานต่ำ ซึ่งมีขนาด 8 บิต
- มีคำสั่งสูงถึง 130 คำสั่งเนื่องจากใช้สถาปัตยกรรมแบบ RISC คำสั่งพื้นฐานส่วนมากจะใช้เวลาของ Clock เพียง 1 Cycle ในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โดยทั่วไปมีจำนวน Register ที่ใช้ทำงาน ขนาด 32 x 8 บิตที่ทำงานได้อย่างเต็มที่
- ใช้ความเร็วได้ถึง 16 MIPS ที่ความถี่ 16 MHz
- หน่วยความจำถาวรและหน่วยความจำของข้อมูล
 - มี Flash Memory จำนวน 8K Bytes สำหรับใช้ในการเขียนโปรแกรมซึ่งมีความคงทนในเขียนและลบได้ถึง 10,000 ครั้ง
 - มี EEPROM จำนวน 512 Bytes มีความคงทนซึ่งใช้เขียนและลบได้ถึง 100,000 ครั้ง
 - มี SRAM จำนวน 1 K Bytes ภายในตัว
 - มีระบบรักษาความปลอดภัย สามารถล็อกไม่ให้อ่านโปรแกรมใน flash memory และข้อมูลใน EEPROM ได้
- อุปกรณ์ภายใน
 - มี Timer/Counter 8 บิต จำนวน 2 ตัว พร้อมวงจรรักษาความถี่ลอคแยกอิสระ และใช้การเปรียบเทียบได้ 1 โหมด
 - มี Timer/Counter 16 บิต จำนวน 1 ตัวพร้อมวงจรรักษาความถี่ลอคแยกอิสระ, วงจรเปรียบเทียบและวงจร Capture Mode
 - มีวงจรแยกส่วนของออสซิลเลเตอร์ในการนับแบบ Real Time
 - มีวงจร PWM จำนวน 3 ชุด
 - มี ADC จำนวน 6 ช่องสัญญาณภายในตัวถึงซึ่งห่อหุ้มแบบ PDIP ซึ่งมีความแม่นยำในการทำงานที่ 10 บิต จำนวน 6 ช่องทางและ 8 บิตจำนวน 2 ช่องทาง
 - มีจำนวนสายสัญญาณ 2 สายที่ใช้เชื่อมต่อภายนอกที่เป็นแบบอนุกรมซึ่งทำงานสอดคล้องกันตามจำนวน Byte
 - สามารถโปรแกรมข้อมูลแบบอนุกรมแบบ USART
 - สามารถกำหนด Master/Slave ในการเชื่อมต่อข้อมูลแบบอนุกรมแบบ SPI
 - มีความสามารถในการโปรแกรมให้ตรวจตราข้อมูลของ Timer ด้วยออสซิลเลเตอร์ที่แยกกันภายในตัวชิพ
 - มีวงจรเปรียบเทียบข้อมูลที่เป็นอนาล็อกภายในตัวชิพ
- คุณสมบัติพิเศษภายในตัวไมโครคอนโทรลเลอร์
 - สามารถรีเซ็ตตัวเองหลังจากจ่ายไฟให้ และสามารถเขียนโปรแกรมให้ตรวจจับ Brown-out ได้
 - สามารถปรับแต่งวงจร RC ออสซิลเลเตอร์ภายในได้
 - สามารถรับการ Interrupt ได้ทั้งภายนอกและภายใน

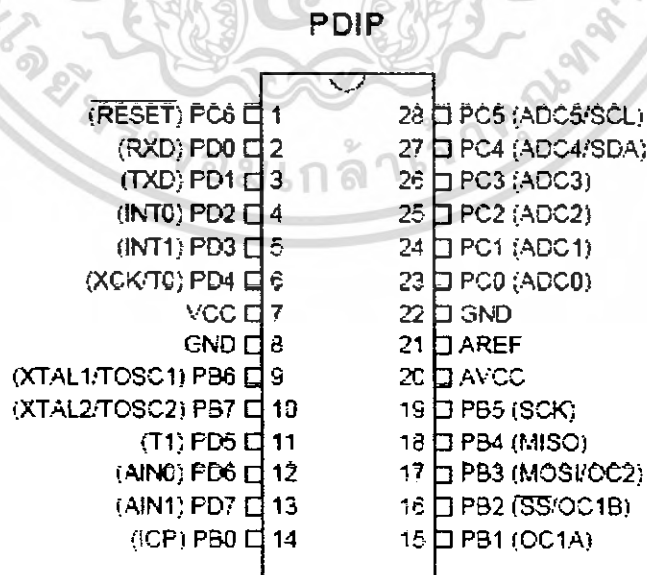
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-มีโหมด Sleep จำนวน 5 โหมด: ในอุดมคติ, ลดทอนสัญญาณรบกวนจากวงจร ADC, ประหยัดพลังงาน, ใช้พลังงานต่ำและโหมดเตรียมพร้อมที่จะทำงาน

- อินพุต/เอาต์พุตและการหีบห่อ
- สามารถโปรแกรมกำหนดให้เป็นอินพุต/เอาต์พุตได้ถึง 23 เส้นทาง
- มีจำนวนขา 28 ขาซึ่งใช้ตัวถังแบบ PDIP
- แรงดัน ไฟฟ้าที่ใช้ในการทำงาน
- 2.7-5.5V สำหรับ ATmega8L
- 4.5-5.5V สำหรับ ATmega8
- ความเร็วในการทำงาน
- 0-8 MHz สำหรับ ATmega8L
- 0-4 MHz สำหรับ ATmega8
- การใช้กำลังงานที่ 4 MHz, 3V, 25 °C
- ใช้กระแสไฟฟ้าในสถานะแอ็คทีฟที่ 3.6 mA
- ในสถานะอุดมคติใช้ที่ 1.0 mA
- ในสถานะกำลังงานต่ำใช้ที่ 0.5 μ A

2.6 การจัดขาและคุณสมบัติของพอร์ตต่างๆ

จากที่ได้กล่าวมาแล้วว่าไมโครคอนโทรลเลอร์ที่เราเลือกใช้มีขนาด 28 ขาซึ่งแต่ละขาก็จะมีคุณสมบัติต่างๆ กันไป ซึ่งเราสามารถเลือกใช้งานได้ตามความต้องการ โดยมีการจัดขา ดังรูปที่ 2.6



รูป 2.6 แสดงการจัดขาของ AVR เบอร์ ATmega8-16PI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1 การจัดขาของไมโครคอนโทรลเลอร์ AVR เบอร์ ATmega8-16PI

ขาของไมโครคอนโทรลเลอร์ AVR มีความสามารถในการต่อใช้งานที่แตกต่างกันไปซึ่งได้อธิบายมาพอสังเขปดังต่อไปนี้

- ขา VCC ใช้สำหรับต่อไฟเลี้ยงของไมโครคอนโทรลเลอร์
- ขา GND เป็นขาราวด์ สำหรับต่อกับกราวด์ของระบบ
- ขา \overline{RESET} ใช้ในการรีเซ็ตการทำงานของไมโครคอนโทรลเลอร์ โดยในการป้อนสัญญาณเพื่อรีเซ็ตนั้น จะใช้สัญญาณที่มีลอจิกศูนย์และต้องนานพอที่จะผลิตสัญญาณพัลส์ที่มีความกว้างของพัลส์เพียงพอในการสร้างสัญญาณรีเซ็ตได้ แม้ว่าจะไม่มีสัญญาณ Clock เลยก็ตาม
- ขา RXD เป็นขาอินพุตสำหรับพอร์ตอนุกรม
- ขา TXD เป็นขาเอาต์พุตสำหรับพอร์ตอนุกรม
- ขา INT0 เป็นขาจับสัญญาณอินเตอร์รัพท์จากภายนอก
- ขา INT1 เป็นขาจับสัญญาณอินเตอร์รัพท์จากภายนอก
- ขา XCK/T0 เป็นขาสำหรับรับสัญญาณ Clock จากภายนอกสำหรับ Timer/Counter 0
- ขา T1 เป็นขาสำหรับรับสัญญาณ Clock จากภายนอกสำหรับ Timer/Counter 1
- ขา XTAL1/TOSC1 เป็นขาอินพุตของ Inverting Oscillator Amplifier สำหรับรับสัญญาณ Clock จากภายนอก และเป็นอินพุตสำหรับการทำงานของ Clock ภายใน (กรณีที่เราต่อคริสตัลไว้) หรือจะใช้สัญญาณ Clock ภายในตัวไมโครคอนโทรลเลอร์ก็ได้ (กรณีที่เราไม่ได้ออกรีสตอล)
- ขา XTAL2 เป็นขาเอาต์พุตของ Inverting Oscillator Amplifier ที่ใช้ Clock จากภายนอกหรือภายในก็ได้
- ขา AIN0 ถูกใช้เป็นอินพุตของสำหรับ Analog Comparator
- ขา AIN1 ถูกใช้เป็นอินพุตของสำหรับ Analog Comparator
- ขา ICP ใช้เป็นอินพุตสำหรับตั้ง Timer/Counter1 เก็บค่าที่กำลังนับไว้ในรีจิสเตอร์
- ขา OC1A เป็นสัญญาณ Timer/Counter1 ที่ใช้เปรียบเทียบกับสัญญาณเอาต์พุต ชุด A
- ขา \overline{SS} /OC1B ขา \overline{SS} (Slave Select) เป็นขาเลือกการติดต่อในกรณีที่ไมโครคอนโทรลเลอร์เป็นอุปกรณ์ สเลฟในระบบการติดต่อแบบ SPI ส่วนขา OC1B เป็นสัญญาณ Timer/Counter1 ที่ใช้เปรียบเทียบกับสัญญาณเอาต์พุต ชุด B
- ขา MOSI/OC2 ขา MOSI (Master data output, Slave data input) ใช้ในการติดต่อกับพอร์ต SPI ส่วนขา OC2 เป็นสัญญาณ Timer/Counter2 ที่ใช้เปรียบเทียบกับสัญญาณเอาต์พุต
- ขา MISO (Master data input, Slave data output) ใช้ในการติดต่อกับพอร์ต SPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ขา SCK (Master clock output) เป็นขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต SPI
- AVCC เป็นขาที่ใช้สำหรับต่อไฟเลี้ยงสำหรับในการใช้ A/D Converter ที่ขา ADC0 ถึง ADC3 ส่วนขา ADC4 และ ADC5 จะใช้ไฟเลี้ยงจาก VCC
- AREF เป็นขาที่ไว้ต่อแรงดันอ้างอิงของสัญญาณอนาล็อก สำหรับในการใช้งาน A/D Converter
- ADC เป็นขาที่สามารถใช้งานเป็น A/D Converter ซึ่งมีตั้งแต่ ADC0 ถึง ADC5

2.6.2 โครงสร้างและการทำงานของพอร์ตไมโครคอนโทรลเลอร์ AVR

พอร์ตต่างๆ ที่ใช้ในการส่งข้อมูลหรือรับข้อมูลนั้น จะมีอยู่ 3 พอร์ตด้วยกันคือ พอร์ต B, พอร์ต C และพอร์ต D ซึ่งได้อธิบายมาโดยสังเขปดังนี้

- พอร์ต B (PB0 - PB7) เป็นพอร์ต 8 บิตสามารถโปรแกรมให้แต่ละบิตของพอร์ตเป็นอินพุตหรือเอาต์พุตก็ได้แยกอิสระในแต่ละบิต แต่ละขาของพอร์ตสามารถจ่ายกระแสออกไปให้กับอุปกรณ์ภายนอก (source) หรือรับกระแสเข้ามาภายใน (sink) ได้สูง ซึ่งในพอร์ต B นั้นจะมี I/O Memory Address อยู่ 3 ตัวที่ถูกจัดสรรไว้ให้ นั่นคือ
 - Port B Data Register – PORTB
 - Port B Data Direction Register – DDRB
 - Port B Input Pins Address – PINB

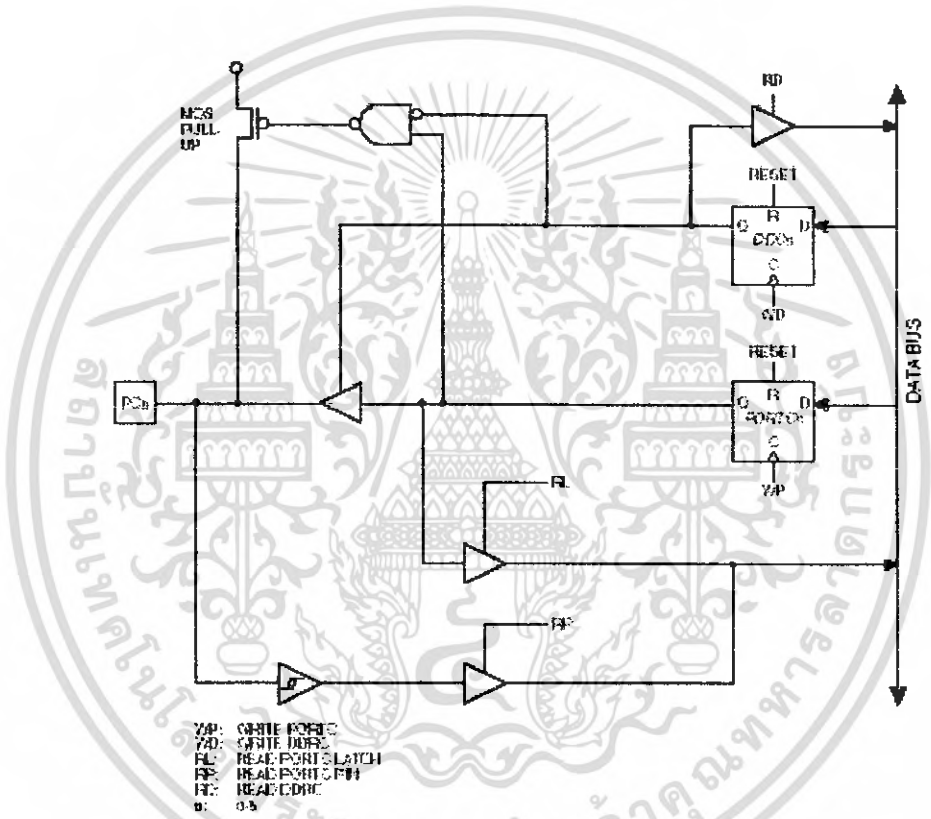
Port B Input Pins address (PINB) ไม่มี Register; Address นี้ เปิดให้มีการ Access ค่าต่าง ๆ ที่ Port B เมื่อมีการอ่าน PORTB ข้อมูลที่ Port B จะคงสถานะของการอ่าน เมื่อมีการอ่านที่ PINB มีสถานะของการทำงานต่าง ๆ ดังนี้

DDBn	PORTBn	I/O	Pull-up	Comment
0	0	Input	No	Tri-state (high-Z)
0	1	Input	Yes	PBn will source current if ext. pulled low
1	0	Output	No	Push-pull Zero Output
1	1	Output	No	Push-pull One Output

ตาราง 2.2 สถานะแสดงการทำงานของพอร์ต B

DDCn	PORTCn	I/O	Pull-up	Comment
0	0	Input	No	Tri-state (high-Z)
0	1	Input	Yes	PCn will source current if ext. pulled low
1	0	Output	No	Push-pull Zero Output
1	1	Output	No	Push-pull One Output

ตาราง 2.3 สถานะแสดงการทำงานของพอร์ต C



รูป 2.8 โครงสร้างการทำงานภายในของพอร์ต C

• พอร์ต D (PD0 - PD7) เป็นพอร์ต 8 บิตสามารถโปรแกรมให้แต่ละบิตของพอร์ตเป็นอินพุตหรือเอาต์พุตก็ได้แยกอิสระในแต่ละบิต แต่ละขาของพอร์ตสามารถจ่ายกระแสออกไปให้กับอุปกรณ์ภายนอก (source) หรือรับกระแสเข้ามาภายใน (sink) ได้สูง ซึ่งในพอร์ต B นั้นจะมี I/O Memory Address อยู่ 3 ตัวที่ถูกจัดสรรไว้ให้ นั่นคือ

- Port D Data Register – PORTD
- Port D Data Direction Register → DDRD
- Port D Input Pins Address – PIND

Port D Input Pins address (PIND) ไม่มี Register; Address นี้ เปิดให้มีการ Access ค่าต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ไมโครคอนโทรลเลอร์ AVR จะยอมรับคำสั่งตามตารางที่ 2.6 ซึ่งเป็นตารางแสดงชุดคำสั่ง

Instruction Mnemonic

Address	Mnemonic	BR 7	BR 6	BR 5	BR 4	BR 3	BR 2	BR 1	BR 0	Page
0x0F (0x0F)	SREG	1	7	H	S	V	M	Z	C	9
0x0E (0x0E)	SPH	-	-	-	-	-	SP10	SP9	SP8	11
0x0D (0x0D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	11
0x0C (0x0C)	Reserved									
0x0B (0x0B)	GICR	INT1	INT0	-	-	-	-	IVSEL	NCE	47, 85
0x0A (0x0A)	GIFR	INTF1	INTF0	-	-	-	-	-	-	86
0x09 (0x09)	TIMSK	OCIE2	TOIE2	TCIE1	OCIE1A	OCIE1B	TOIE1	-	TOIE0	70, 100, 129
0x08 (0x08)	TIER	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	-	TOV0	71, 101, 130
0x07 (0x07)	SPMCR	SPMIE	RWWSB	-	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	208
0x06 (0x06)	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWRE	167
0x05 (0x05)	MCUCR	SE	SM2	SM1	SM0	ISC11	ISC10	ISC01	ISC00	31, 84
0x04 (0x04)	MCLUCSR	-	-	-	-	WDRF	BOFIF	EXTRIF	POFIF	39
0x03 (0x03)	TCCR0	-	-	-	-	-	CS02	CS01	CS00	70
0x02 (0x02)	TCNT0	Timer/Counter0 (8 Bits)								70
0x01 (0x01)	OSCAL	Oscillator Calibration Register								29
0x00 (0x00)	SIFR	-	-	-	ADIFSM	ACIFE	FUD	PIF0	PIF10	36, 73, 121, 188
0x0F (0x0F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	85
0x0E (0x0E)	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	86
0x0D (0x0D)	TCNT1H	Timer/Counter1 - Counter Register High Byte								89
0x0C (0x0C)	TCNT1L	Timer/Counter1 - Counter Register Low Byte								89
0x0B (0x0B)	OCR1AH	Timer/Counter1 - Output Compare Register A High Byte								89
0x0A (0x0A)	OCR1AL	Timer/Counter1 - Output Compare Register A Low Byte								89
0x09 (0x09)	OCR1BH	Timer/Counter1 - Output Compare Register B High Byte								89
0x08 (0x08)	OCR1BL	Timer/Counter1 - Output Compare Register B Low Byte								89
0x07 (0x07)	ICR1H	Timer/Counter1 - Input Capture Register High Byte								100
0x06 (0x06)	ICR1L	Timer/Counter1 - Input Capture Register Low Byte								100
0x05 (0x05)	TCCR2	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	115
0x04 (0x04)	TCNT2	Timer/Counter2 (8 Bits)								117
0x03 (0x03)	OCR2	Timer/Counter2 Output Compare Register								117
0x02 (0x02)	ASSR	-	-	-	-	AS2	TCN2UB	OCR2UB	TCR2UB	117
0x01 (0x01)	WDTCSR	-	-	-	WDFE	WDE	WDF2	WDF1	WDF0	41
0x00 ¹⁾ (0x00 ¹⁾)	UBRRH	URSEL	-	-	-	-	UBRRH(11 bits)			154
	UCSRH	URSEL	URSEL	UPM1	UPM0	USBS	UCS21	UCS20	UCPOL	152
0x1F (0x1F)	EEARH	-	-	-	-	-	-	-	EEAR0	18
0x1E (0x1E)	EEARL	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	18
0x1D (0x1D)	EEDR	EEPROM Data Register								18
0x1C (0x1C)	EERCR	-	-	-	-	EEPE	EEMWE	EEWE	EERE	18
0x1B (0x1B)	Reserved									
0x1A (0x1A)	Reserved									
0x19 (0x19)	Reserved									
0x18 (0x18)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	63
0x17 (0x17)	DDRB	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0	63
0x16 (0x16)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	63
0x15 (0x15)	PORTC	-	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	63
0x14 (0x14)	DDRC	-	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	63
0x13 (0x13)	PINC	-	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	63
0x12 (0x12)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	63
0x11 (0x11)	DDR0	DD07	DD06	DD05	DD04	DD03	DD02	DD01	DD00	63
0x10 (0x10)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	63
0x0F (0x0F)	SPDR	SPI Data Register								128
0x0E (0x0E)	SPSR	SPIF	WCOL	-	-	-	-	-	SP2X	128
0x0D (0x0D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	128
0x0C (0x0C)	UDR	USART I/O Data Register								148
0x0B (0x0B)	UCSRA	RXC	TXC	UDRIE	FE	DCR	FE	UDZL	MPCM	150
0x0A (0x0A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCS22	UCS20	TXB8	151
0x09 (0x09)	UBRRFL	USART Baud Rate Register Low Byte								154
0x08 (0x08)	ACSR	AC0	ACBG	AC0	AC1	AC1E	AC1C	ACS1	ACSD	189
0x07 (0x07)	ADMUX	REFS1	REFS0	ADLAR	-	MUX2	MUX1	MUX0	ADSC	200
0x06 (0x06)	ADCSRA	ADEN	ADSC	ADFR	ADIF	ADIFE	ADPS2	ADPS1	ADPS0	202
0x05 (0x05)	ADCH	ADC Data Register High Byte								203
0x04 (0x04)	ADCL	ADC Data Register Low Byte								203
0x03 (0x03)	TWDR	Two-wire Serial Interface Data Register								189
0x02 (0x02)	TWAR	TWAS	TWAS	TWAS	TWAS	TWAS	TWAS	TWAS	TWAS	189

ตาราง 2.5 แสดงแอดเดรสรีจิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 72855
 ไม่วการณใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
0x01 (0x21)	TW2R	TW27	TW26	TW25	TW24	TW23	-	TW21	TW20	188
0x00 (0x20)	TW2R									187

Two-wire Serial Interface Bit Rate Register

ตาราง 2.5 แสดงแอดแครสรีจิสเตอร์(ต่อ)

Mnemonic	Operands	Description	Operation	Flags	#Cycles
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add two Registers	Rd ← Rd + Rr	Z, C, MV, H	1
ADC	Rd, Rr	Add with Carry two Registers	Rd ← Rd + Rr + C	Z, C, MV, H	1
ADW	Rd, K	Add Immediate to Word	Rd ← Rd + Rd ← K	Z, C, MV, H	2
SUB	Rd, Rr	Subtract two Registers	Rd ← Rd - Rr	Z, C, MV, H	1
SUBB	Rd, K	Subtract Constant from Register	Rd ← Rd - K	Z, C, MV, H	1
SBC	Rd, Rr	Subtract with Carry into Register	Rd ← Rd - Rr - C	Z, C, MV, H	1
SBCD	Rd, K	Subtract with Carry Constant from Reg.	Rd ← Rd - K - C	Z, C, MV, H	1
SBW	Rd, K	Subtract Immediate from Word	Rd ← Rd - Rd ← K	Z, C, MV, H	2
AND	Rd, Rr	Logical AND Registers	Rd ← Rd & Rr	Z, N, V	1
ANDI	Rd, K	Logical AND Register and Constant	Rd ← Rd & K	Z, N, V	1
OR	Rd, Rr	Logical OR Registers	Rd ← Rd Rr	Z, N, V	1
ORI	Rd, K	Logical OR Register and Constant	Rd ← Rd K	Z, N, V	1
EOR	Rd, Rr	Exclusive OR Registers	Rd ← Rd ⊕ Rr	Z, N, V	1
COM	Rd	One's Complement	Rd ← ~Rd	Z, C, MV	1
NEG	Rd	Two's Complement	Rd ← -Rd	Z, C, MV, H	1
BSR	Rd, K	Set Bit(s) in Register	Rd ← Rd K	Z, N, V	1
BSR	Rd, K	Clear Bit(s) in Register	Rd ← Rd & (~K)	Z, N, V	1
INC	Rd	Increment	Rd ← Rd + 1	Z, N, V	1
DEC	Rd	Decrement	Rd ← Rd - 1	Z, N, V	1
TST	Rd	Test for Zero or Other	Rd ← Rd & Rd	Z, N, V	1
CLR	Rd	Clear Register	Rd ← Rd & 0	Z, N, V	1
SEI	Rd	Set Register	Rd ← Rd 0xFF	None	1
MUL	Rd, Rr	Multiply Unsigned	R1:R0 ← Rd * Rr	Z, C	2
MULS	Rd, Rr	Multiply Signed	R1:R0 ← Rd * Rr	Z, C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	R1:R0 ← Rd * Rr	Z, C	2
FMUL	Rd, Rr	Fractional Multiply Unsigned	R1:R0 ← (Rd * Rr) << 1	Z, C	2
FMULS	Rd, Rr	Fractional Multiply Signed	R1:R0 ← (Rd * Rr) << 1	Z, C	2
FMULSU	Rd, Rr	Fractional Multiply Signed with Unsigned	R1:R0 ← (Rd * Rr) << 1	Z, C	2
BRANCH INSTRUCTIONS					
JMP	k	Relative Jump	PC ← PC + k + 1	None	2
LMP		Relative Jump to (Z)	PC ← Z	None	2
JMP	k	Direct Jump	PC ← k	None	3
NCALL	k	Relative Subroutine Call	PC ← PC + k + 1	None	3
ICALL		Indirect Call to (Z)	PC ← Z	None	3
CALL	k	Direct Subroutine Call	PC ← k	None	4
RET		Return from Subroutine	PC ← STACK	None	4
RETI		Interrupt Return	PC ← STACK	!	4
CPSIE	Rd, Rr	Compare, Stop if Equal	if (Rd == Rr) PC ← PC + 2 or 3	None	1/2/3
CP	Rd, Rr	Compare	Rd - Rr	Z, N, V, C, H	1
CPC	Rd, Rr	Compare with Carry	Rd - Rr - C	Z, N, V, C, H	1
CPI	Rd, K	Compare Register with Immediate	Rd - K	Z, N, V, C, H	1
SBSC	Rd, b	Stop if Bit in Register Cleared	if (Rd[b] == 0) PC ← PC + 2 or 3	None	1/2/3
SBRS	Rd, b	Stop if Bit in Register is Set	if (Rd[b] == 1) PC ← PC + 2 or 3	None	1/2/3
SBC	F, b	Stop if Bit in MO Register Cleared	if (F[b] == 0) PC ← PC + 2 or 3	None	1/2/3
SBS	F, b	Stop if Bit in MO Register is Set	if (F[b] == 1) PC ← PC + 2 or 3	None	1/2/3
BRSC	s, b	Branch if Status Flag Cleared	if (SREG[s] == 0) then PC ← PC + k + 1	None	1/2
BRSC	s, b	Branch if Status Flag Cleared	if (SREG[s] == 0) then PC ← PC + k + 1	None	1/2
BREQ	k	Branch if Equal	if (Z == 1) then PC ← PC + k + 1	None	1/2
BRNE	k	Branch if Not Equal	if (Z == 0) then PC ← PC + k + 1	None	1/2
BRCC	k	Branch if Carry Set	if (C == 1) then PC ← PC + k + 1	None	1/2
BRCC	k	Branch if Carry Cleared	if (C == 0) then PC ← PC + k + 1	None	1/2
BRSH	k	Branch if Same or Higher	if (C == 0) then PC ← PC + k + 1	None	1/2
BRLO	k	Branch if Lower	if (C == 1) then PC ← PC + k + 1	None	1/2
BRD	k	Branch if Above	if (N == 1) then PC ← PC + k + 1	None	1/2
BRD	k	Branch if Above	if (N == 1) then PC ← PC + k + 1	None	1/2
BRGE	k	Branch if Greater or Equal, Signed	if (N == 0) then PC ← PC + k + 1	None	1/2
BRGT	k	Branch if Greater Than, Signed	if (N == 0) then PC ← PC + k + 1	None	1/2
BRCS	k	Branch if Half Carry Flag Set	if (H == 1) then PC ← PC + k + 1	None	1/2
BRCS	k	Branch if Half Carry Flag Cleared	if (H == 0) then PC ← PC + k + 1	None	1/2
BRTS	k	Branch if T Flag Set	if (T == 1) then PC ← PC + k + 1	None	1/2
BRTC	k	Branch if T Flag Cleared	if (T == 0) then PC ← PC + k + 1	None	1/2
BRVS	k	Branch if Overflow Flag is Set	if (V == 1) then PC ← PC + k + 1	None	1/2
BRVC	k	Branch if Overflow Flag is Cleared	if (V == 0) then PC ← PC + k + 1	None	1/2

ตาราง 2.6 แสดงชุดคำสั่ง Instruction Mnemonic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mnemonic	Operands	Description	Operation	Flags	Clocks
BRE	k	Branch if Interrupt Enabled	$if (I=1) \text{Set } PC \leftarrow PC+k+1$	None	1/2
BRD	k	Branch if Interrupt Disabled	$if (I=0) \text{Set } PC \leftarrow PC+k+1$	None	1/2
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move Between Registers	$Rd \leftarrow Rr$	None	1
MOVW	Rd, Rr	Copy Register Word	$Rd+1:Rd \leftarrow Rr+1:Rr$	None	1
LDI	Rd, K	Load Immediate	$Rd \leftarrow K$	None	1
LD	Rd, X	Load Indirect	$Rd \leftarrow (X)$	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	$Rd \leftarrow (X); X \leftarrow X+1$	None	2
LD	Rd, -X	Load Indirect and Pre-Dec.	$X \leftarrow X-1; Rd \leftarrow (X)$	None	2
LD	Rd, Y	Load Indirect	$Rd \leftarrow (Y)$	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	$Rd \leftarrow (Y); Y \leftarrow Y+1$	None	2
LD	Rd, -Y	Load Indirect and Pre-Dec.	$Y \leftarrow Y-1; Rd \leftarrow (Y)$	None	2
LDD	Rd, Y+q	Load Indirect with Displacement	$Rd \leftarrow (Y+q)$	None	2
LD	Rd, Z	Load Indirect	$Rd \leftarrow (Z)$	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	$Rd \leftarrow (Z); Z \leftarrow Z+1$	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	$Z \leftarrow Z-1; Rd \leftarrow (Z)$	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	$Rd \leftarrow (Z+q)$	None	2
LDS	Rd, t	Load Direct from SRAM	$Rd \leftarrow (t)$	None	2
ST	X, Rr	Store Indirect	$(X) \leftarrow Rr$	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	$(X) \leftarrow Rr; X \leftarrow X+1$	None	2
ST	-X, Rr	Store Indirect and Pre-Dec.	$X \leftarrow X-1; (X) \leftarrow Rr$	None	2
ST	Y, Rr	Store Indirect	$(Y) \leftarrow Rr$	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	$(Y) \leftarrow Rr; Y \leftarrow Y+1$	None	2
ST	-Y, Rr	Store Indirect and Pre-Dec.	$Y \leftarrow Y-1; (Y) \leftarrow Rr$	None	2
STD	Y+q, Rr	Store Indirect with Displacement	$(Y+q) \leftarrow Rr$	None	2
ST	Z, Rr	Store Indirect	$(Z) \leftarrow Rr$	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	$(Z) \leftarrow Rr; Z \leftarrow Z+1$	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	$Z \leftarrow Z-1; (Z) \leftarrow Rr$	None	2
STD	Z+q, Rr	Store Indirect with Displacement	$(Z+q) \leftarrow Rr$	None	2
STS	t, Rr	Store Direct to SRAM	$(t) \leftarrow Rr$	None	2
LPM		Load Program Memory	$RD \leftarrow (Z)$	None	3
LPM	Rd, Z	Load Program Memory	$Rd \leftarrow (Z)$	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc.	$Rd \leftarrow (Z); Z \leftarrow Z+1$	None	3
SPM		Store Program Memory	$(Z) \leftarrow Rr: RD$	None	-
IN	Rd, P	In Port	$Rd \leftarrow P$	None	1
OUT	P, Rr	Out Port	$P \leftarrow Rr$	None	1
PUSH	Rr	Push Register on Stack	$STACK \leftarrow Rr$	None	2
POP	Rd	Pop Register from Stack	$Rd \leftarrow STACK$	None	2
BIT AND BIT-TEST INSTRUCTIONS					
SB	P,b	Set Bit in I/O Register	$IOP(b) \leftarrow 1$	None	2
CB	P,b	Clear Bit in I/O Register	$IOP(b) \leftarrow 0$	None	2
LSL	Rd	Logical Shift Left	$Rd(b+1) \leftarrow Rd(b); Rd(b) \leftarrow 0$	Z,C,M,V	1
LSR	Rd	Logical Shift Right	$Rd(b) \leftarrow Rd(b+1); Rd(b+1) \leftarrow 0$	Z,C,M,V	1
ROL	Rd	Rotate Left Through Carry	$Rd(b); C, Rd(b+1) \leftarrow Rd(b); C; Rd(b)$	Z,C,M,V	1
ROR	Rd	Rotate Right Through Carry	$Rd(b+1); C, Rd(b) \leftarrow Rd(b+1); C; Rd(b)$	Z,C,M,V	1
ASR	Rd	Arithmetic Shift Right	$Rd(b) \leftarrow Rd(b+1); n=0,0$	Z,C,M,V	1
SWAP	Rd	Swap Nibbles	$Rd(3,0) \leftarrow Rd(7,4); Rd(7,4) \leftarrow Rd(3,0)$	None	1
SBSET	s	Flag Set	$SREG(s) \leftarrow 1$	SREG(s)	1
BCLR	s	Flag Clear	$SREG(s) \leftarrow 0$	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	$T \leftarrow Rd(b)$	T	1
BLD	Rd, b	Bit Load from T to Register	$Rd(b) \leftarrow T$	None	1
SEC		Set Carry	$C \leftarrow 1$	C	1
CLC		Clear Carry	$C \leftarrow 0$	C	1
SEN		Set Negative Flag	$N \leftarrow 1$	N	1
CLN		Clear Negative Flag	$N \leftarrow 0$	N	1
SEZ		Set Zero Flag	$Z \leftarrow 1$	Z	1
CLZ		Clear Zero Flag	$Z \leftarrow 0$	Z	1
SEI		Global Interrupt Enable	$I \leftarrow 1$	I	1
CLI		Global Interrupt Disable	$I \leftarrow 0$	I	1
SEI		Set Signum Test Flag	$S \leftarrow 1$	S	1
CL S		Clear Signum Test Flag	$S \leftarrow 0$	S	1
SEV		Set Two's Complement Overflow	$V \leftarrow 1$	V	1
CLV		Clear Two's Complement Overflow	$V \leftarrow 0$	V	1
SET		Set T to SPFC	$T \leftarrow 1$	T	1

ตาราง 2.6 แสดงชุดคำสั่ง Instruction Mnemonic (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mnemonic	Operands	Description	Operation	Flags	#Cycles
CLT		Clear T in SREG	T ← 0	T	1
SBH		Set Half Carry Flag in SREG	H ← 1	H	1
CLH		Clear Half Carry Flag in SREG	H ← 0	H	1
MCU CONTROL INSTRUCTIONS					
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep function)	None	1
WDR		Watchdog Reset	(see specific descr. for WDRtimer)	None	1

ตาราง 2.6 แสดงชุดคำสั่ง Instruction Mnemonic (ต่อ)

จากทั้งหมดที่กล่าวมาในบทนี้ จะทำให้ทราบรายละเอียดของไมโครคอนโทรลเลอร์ AVR เบอร์ ATmega8-16PI ได้อย่างคร่าวๆแล้ว ซึ่งในบทต่อไปจะได้อธิบายเกี่ยวกับการต่อวงจรใช้งาน และการออกแบบรวมถึงการเขียนโปรแกรมลงบนตัวไมโครคอนโทรลเลอร์ ซึ่งใช้การเชื่อมต่อในการโปรแกรมโดยผ่านพอร์ตขนานต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การประมวลผลภาพ

3.1 หลักการพื้นฐานทางอิมเมจโพรเซสซิง

อิมเมจโพรเซสซิง (Image processing) เป็นกระบวนการที่ใช้ในการจัดเก็บข้อมูลที่เป็นรูปภาพต่าง ๆ ให้อยู่ในลักษณะของสัญญาณไฟฟ้าเพื่อที่จะได้นำข้อมูลที่เป็นสัญญาณไฟฟ้านั้นไปใช้ประโยชน์ทางอื่น เป็นต้นว่า การตกแต่ง, การส่งรูปภาพไปตามสายนำสัญญาณจากที่แหล่งหนึ่งไปยังอีกแห่งหนึ่ง (หลักการของโทรสาร), การเก็บข้อมูลไว้ในหน่วยความจำเพื่อทำอัลบั้มภาพอิเล็กทรอนิกส์เพื่อใช้ประโยชน์เป็นแฟ้มข้อมูลพนักงาน, ตรวจสอบลายนิ้วมือ เป็นต้น

3.1.1 พิกเซล (Pixel)

ภาพที่จะถูกแปลงสัญญาณไฟฟ้า จะได้รับการแบ่งรายละเอียดของภาพเป็นตารางเล็ก ๆ โดยตารางเล็ก ๆ นี้เรียกว่า พิกเซล (Pixel) มีการจัดเรียงเป็นแถว (row) และ คอลัมน์ (column) ซึ่งตำแหน่ง แต่ละพิกเซลจะแทนด้วย $P(i,j)$ โดยที่ i และ j จะเป็นเลขจำนวนเต็มและเรียกว่าการจัดเรียง พิกเซลเมตริกซ์ (pixel matrix)

เมื่อทราบตำแหน่งของพิกเซลแล้วก็จำเป็นต้องทราบว่าตำแหน่งนั้น ๆ พิกเซลมีค่าเท่าไร ซึ่งค่าที่ได้นี้คือค่าเฉลี่ยความเข้มของแสงที่ตกกระทบบนตำแหน่งของแต่ละพิกเซล

3.1.2 ลักษณะข้อมูลภาพ

ภาพ 2 ระดับ คือ มีแค่จุดขาวกับดำเท่านั้น โดยแต่ละจุดมีข้อมูล 1 บิต

ภาพ 16 ระดับ ซึ่งในแต่ละจุดภาพจะเป็นข้อมูล 4 บิต ซึ่งสามารถทำให้แสดงภาพได้ 16 ระดับสี หรือ 16 ระดับสีเทา ขึ้นอยู่กับภาพนั้นเป็นภาพสีหรือภาพขาว-ดำ

ภาพ 256 ระดับ ซึ่งในแต่ละจุดจะเป็นข้อมูล 8 บิต ซึ่งทำให้สามารถแสดงภาพได้ 256 ระดับ หรือระดับสีเทา

ภาพ true color ซึ่งในแต่ละจุดจะเป็นข้อมูลขนาด 24 บิต ทำให้สามารถแสดงผลภาพได้เหมือนภาพจริงที่สุด เพราะสามารถแสดงสีได้ถึง 16,777,216 สี แสดงได้เฉพาะภาพสีเท่านั้น

3.1.3 เกรย์สเกล (Gray scale)

เกรย์สเกล หมายถึง ความแตกต่างของระดับความเข้มของแสง โดยเกรย์สเกลหนึ่ง ๆ อาจแบ่งเป็น 13,20 หรือ 9 ระดับ โดยระดับที่วานี้ก็คือ ระดับสีเทา ในภาพหนึ่ง ๆ ถ้าต้องการแบ่งระดับความเข้มแสงหรือระดับสีเทาให้มีหลาย ๆ ค่า นั้น จำเป็นอย่างยิ่งที่จะต้องเพิ่มจำนวนบิต ที่แสดงค่ากับพิกเซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างเช่นถ้าต้องการภาพที่มีระดับสีเทา 4 ระดับ ต้องแทนด้วยเลขฐานสองจำนวน 2 บิต ถ้าต้องการภาพที่มีระดับสีเทา 16 ระดับต้องแทนด้วยเลขฐานสองจำนวน 4 บิต และถ้าต้องการภาพที่มีระดับสีเทา 256 ระดับ ต้องแทนด้วยเลขฐานสองจำนวน 8 บิต เป็นต้น

จำนวนระดับสีเทาที่ต้องการนี้คือค่ายกกำลังขอบ 2 นั้นเอง ซึ่งค่าต่ำสุดหรือ 0 จะแทนสีดำ หรือไม่มีความสว่างเลย และค่าที่มากที่สุดคือค่าที่น้อยกว่าจำนวนระดับสีเทาอยู่ 1 เช่น ค่า 15 ในระดับที่มี ระดับสีเทา 16 ระดับ ก็จะเป็นสีขาวหรือสว่างมากที่สุด เป็นต้น

ในยุคแรก ๆ ระดับการมองเห็น (Vision system) จะใช้ระบบเลขฐานสองเพราะสะดวกต่อการนำเซนเซอร์มาใช้ นอกจากนี้การรวบรวมข้อมูล การเก็บรักษาข้อมูลยังสามารถทำได้ง่ายอีกด้วย

ในปัจจุบัน ไมโครโปรเซสเซอร์ที่ใช้กันโดยทั่วไป ขนาดที่เล็กที่สุดคือ 8 บิต ดังนั้นเกรย์สเกลขนาด 8 ค่า, 16 ค่า และ 256 ค่า จึงไม่เป็นปัญหาในการประมวลผล

ความสามารถในการแบ่งแยกระดับความแตกต่างของสายตามนุษย์ โดยทั่ว ๆ ไปจะอยู่ระหว่าง 10 ถึง 15 ระดับ ดังนั้น เกรย์สเกลขนาด 16 ระดับ จึงถือได้ว่าใกล้เคียงกับสายตามนุษย์หรืออาจจะละเอียดน้อยกว่าสายตามนุษย์เล็กน้อย (ในบางคน) ในขณะที่เกรย์สเกลขนาด 64 หรือ 256 นั้นละเอียดเกินไปสำหรับมนุษย์

การแสดงผลนี้ใช้วิธีตั้งค่าของแม่สีในตารางสี โดยอาจเลือกเป็นแบบ 16 สี จาก 64 สี หรือ 16 สีจาก 262,144 สีหรือ 256 สีจาก 262,144 สี ขึ้นอยู่กับโหมดการแสดงผลสำหรับทรูคัลเลอร์ จะไม่มีการเลือกสี แสดงผลโดยการส่งค่าสี RGB ผ่าน D/A สีละ 8 บิต ออกไปความแตกต่างของการแสดงผลสีและขาวดำคือ ภาพขาวดำจะต้องตั้งให้แม่สีทั้งสามสีมีค่าเท่ากัน เนื่องจาก VGA กำหนดให้แม่สีแต่ละสีใช้ได้เพียง 64 ระดับสีเท่านั้นหากต้องการให้เห็นทั้ง 256 ระดับ ต้องแสดงในโหมดทรูคัลเลอร์ แล้วให้ RGB มีค่าเท่ากัน ซึ่งในโหมดนี้ จะสามารถใช้รีจิสเตอร์ 8 บิต สำหรับแต่ละแม่สี

โดยทั่วไปวิธีการประมวลผลภาพเชิงเลขที่ทำให้คอมพิวเตอร์สามารถรู้จักวัตถุภาพในภาพได้นั้นแบ่งได้สองระดับด้วยกันคือ การประมวลผลภาพในระดับต่ำ (Low level image processing) และการประมวลผลภาพในระดับสูง (High level image processing) การประมวลผลภาพระดับต่ำจะเป็นการประมวลผลเชิงตัวเลขเกือบทั้งหมด เพื่อหาค่าตัวแปรต่าง ๆ มาอธิบายข้อมูลภาพ โดยมีจุดประสงค์เพื่อนำตัวแปรเหล่านี้ไปใช้ในการประมวลผลระดับสูงต่อไป

การประมวลผลภาพในระดับสูง เป็นหารนำผลลัพธ์ที่ได้จากการประมวลผลระดับต่ำมาตีความหรือเพื่อให้คอมพิวเตอร์ สามารถรู้จักและเข้าใจภาพได้สำหรับความแตกต่างของการประมวลผลภาพระดับต่ำและสูง คือ ข้อมูลที่นำมาใช้ในการประมวลผลโดยการประมวลผลภาพระดับต่ำจะใช้ค่าความสว่างของจุดโดยตรงส่วนการประมวลผลภาพระดับสูงนั้นข้อมูลภาพที่นำมา

ประมวลผลผลจะถูกแสดงในรูปลักษณะ ซึ่งจะแสดงถึงสิ่งต่าง ๆ ที่อยู่ในภาพ เช่น ขนาด หรือ รูปร่างของวัตถุในภาพ

3.1.4 ฮิสโตแกรม (Histogram)

ฮิสโตแกรมคือกราฟที่บอกให้ทราบถึงจำนวนของระดับสีเทาในภาพหนึ่ง ๆ โดยที่แกน X จะเป็นค่าของระดับสีเทาและแกน y เป็นจำนวนของพิกเซล ในฮิสโตแกรมจะประกอบด้วย

- จำนวนพิกเซลทั้งหมดของภาพ
- จำนวนพิกเซลในแต่ละค่าของระดับสีเทา
- กราฟที่แสดงจำนวนในแต่ละค่าของระดับสีเทา
- กราฟที่ใช้ในฮิสโตแกรมจะเป็นกราฟแท่ง ซึ่งสามารถแสดงจำนวนพิกเซลในแต่ละค่าระดับสีเทาได้เป็นอย่างดี

-รูปร่างหรือขนาดของฮิสโตแกรมจะเป็นข้อมูลที่แสดงถึงคุณสมบัติของภาพว่า มีความคมชัดมากน้อยเพียงใด ซึ่งข้อมูลนี้ก็คือประโยชน์ของฮิสโตแกรมที่จะใช้ในการกำหนดค่าเทรชโฮลโดยค่าจะใช้ในการแปลงรูปภาพให้กลายเป็นภาพที่มีระดับความเข้ม 2 ระดับ คือ ขาวกับดำ การสร้างฮิสโตแกรม

ต้องกำหนดก่อนว่าภาพที่จะนำมาสร้างฮิสโตแกรมนั้น จะแบ่งเป็นกี่พิกเซลสร้างเมตริกซ์จากพิกเซลเล็ก ๆ นำค่าพิกเซลในเมตริกซ์ที่ได้จากข้อ 2 มาสร้างตารางความสัมพันธ์ระหว่างค่าระดับสีเทากับจำนวนของพิกเซล ในแต่ละค่าระดับสีเทาว่ามีกี่พิกเซล นำค่ามาพล็อตกราฟแท่ง โดยทางแนวนอนเป็นค่าระดับสีเทา และแกนทางแนวตั้งเป็นแกนของจำนวนพิกเซล และกราฟนี้คือฮิสโตแกรม

3.1.5 ความหมายของภาพและนิยามของภาพในระบบดิจิทัล

ภาพ (image) ในเชิงคณิตศาสตร์หมายถึง ฟังก์ชัน 2 มิติ $f(x,y)$ โดย x และ y เป็นแกนพิกัดในระนาบ 2 มิติ ค่าฟังก์ชัน $f(x,y)$ จะเป็นสัดส่วนกับความสว่างหรือความเข้มของภาพที่ตำแหน่ง x,y ซึ่งเราเรียกว่า ระดับสีเทา (gray scale) ซึ่งปกติเราจะใช้จุดกำเนิดของแกนพิกัด (coordinate) อยู่ทางมุมซ้ายของภาพ ภาพ 2 มิติ ที่แทนด้วยฟังก์ชัน $f(x,y)$ โดย x และ y แกนในระนาบของภาพค่าของฟังก์ชันที่จุด (x,y) คือความเข้มของแสงเป็นพลังงานรูปหนึ่ง ดังนั้น $f(x,y)$ ต้องไม่เป็นศูนย์และมีค่าจำกัดนั้นคือ

$$0 < f(x, y) < a \quad (3.1)$$

โดยธรรมชาติของแสง ซึ่งจะต้องมีแหล่งกำเนิดแสงและส่วนที่สะท้อนของแสงดังนั้นเราสามารถแยกฟังก์ชัน $f(x,y)$ ออกเป็นสองส่วนคือ อิลลูมินันซ์คอมโพเนนต์ (illumination component) และรีเฟล็กแทนท์คอมโพเนนต์ (reflect ant component) จะได้ว่า

$$F(x,y) = i(x,y) * r(x,y) \quad (3.2)$$

เมื่อ

$$0 < i(x,y) < a \quad (3.3)$$

และ

$$0 < r(x,y) < a \quad (3.4)$$

โดยที่ a คือ ค่าคงที่

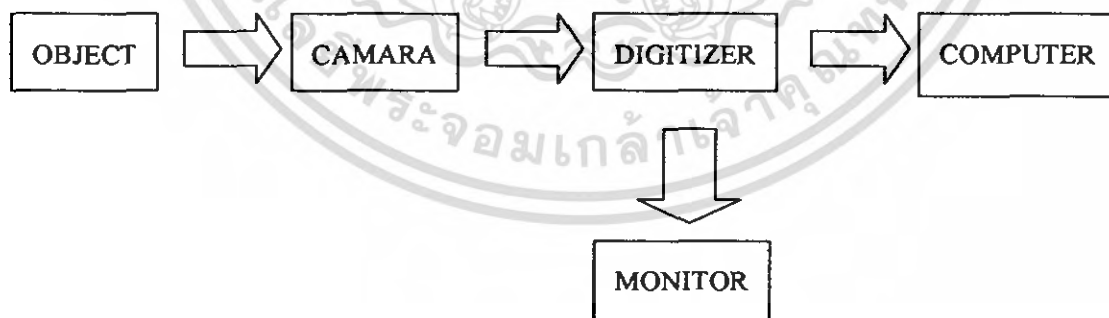
$i(x,y)$ คือ ฟังก์ชันอิลลูมินันซ์คอมโพเนนต์ (illumination component)

$r(x,y)$ คือ ฟังก์ชันรีเฟล็กแทนท์คอมโพเนนต์ (reflect ant component)

สมการที่ 3.4 แสดงให้เห็นว่า ฟังก์ชันการสะท้อนถูกจำกัดขอบเขตระหว่าง 0 (ซึ่งหมายถึงการดูดซึมสมบูรณ์) และ 1 (ซึ่งหมายถึงการสะท้อนโดยสมบูรณ์) ธรรมชาติของ $i(x,y)$ ขึ้นอยู่กับแหล่งกำเนิดแสงในขณะที่ $r(x,y)$ ขึ้นอยู่กับวัตถุที่สะท้อนแสงมาเข้าตา

3.1.6 ระบบการประมวลผลทางดิจิทัล

ระบบการประมวลผลภาพทางดิจิทัลประกอบด้วย 3 ส่วนใหญ่ ๆ คือ ส่วนเปลี่ยนสัญญาณอนาลอกให้เป็นสัญญาณดิจิทัล ซึ่งเรียกว่า ดิจิไตเซอร์ (Digitizer) ส่วนประมวลผล (processing) และส่วนแสดงผล (Display)



รูป 3.1 ระบบการประมวลผลทางดิจิทัล

จากภาพ ส่วนแรกคือ ส่วนที่เปลี่ยนสัญญาณอนาลอกให้เป็นสัญญาณดิจิทัล กล้อง (Camera) เปรียบเสมือนวงตาของมนุษย์ ทำหน้าที่เปลี่ยนภาพวัตถุมาเป็นสัญญาณทางไฟฟ้าและส่งให้ดิจิไตเซอร์ (digitizer) ซึ่งทำหน้าที่เปลี่ยนสัญญาณไฟฟ้าเป็นสัญญาณดิจิทัล อุปกรณ์ส่วนนี้ได้แก่

กล้องโทรทัศน์ดิจิทัลเซอร์ ซึ่งในภาพประกอบด้วยหลอดที่ทำหน้าที่เป็นสื่อไฟฟ้ากับความสว่างของภาพในตำแหน่งนั้น ๆ ค่อยไปทำการควอนไทซ์ (quantizing) ข้อมูลภาพที่ได้เป็นดิจิทัล

ส่วนประมวลผล คือ คอมพิวเตอร์ ซึ่งเปรียบเสมือนสมองทำหน้าที่ประมวลผลและวิเคราะห์ข้อมูลภาพ

ส่วนแสดงผลทำหน้าที่เปลี่ยนข้อมูลเป็นตัวเลข (ซึ่งมีระดับเป็นสีเทา) ที่เก็บเป็นอารีย์ในคอมพิวเตอร์ ให้อยู่ในรูปที่เหมาะสมและสื่อความหมายกับมนุษย์ได้ คือเป็นภาพที่ปกติทั่ว ๆ ไป อุปกรณ์ในส่วนนี้ได้แก่จอ (Monitor)

ภาพ 1 ภาพ ที่เปลี่ยนจากสัญญาณดิจิทัล สำหรับคอมพิวเตอร์ที่มีขนาดใหญ่ขึ้นอยู่กับความละเอียดของภาพที่ต้องการ และจะมีผลทำให้เนื้อที่ในหน่วยความจำมากในการเก็บข้อมูล ภาพ 1 ภาพ เช่น การเก็บภาพ ขนาด $256 * 256$ จุด ที่มีความแตกต่างของระดับความเข้มของแต่ละจุดเท่ากับ 256 ระดับ จะต้องใช้เนื้อที่ในหน่วยความจำให้น้อยที่สุด และยังคงรักษาความละเอียดของภาพตามการใช้งานได้

3.1.7 การสร้างภาพไบนารี (Binary)

การสร้างภาพไบนารี หมายถึง การแปลงข้อมูลที่มีความเข้มหลายระดับ ให้เป็นภาพที่มีระดับความเข้ม 2 ระดับ คือ จุด 1 ภาพ มีค่าได้แค่ 2 ค่า เท่านั้น โดยเป็น 0 กับ 1 ซึ่ง 1 จะหมายถึงภาพที่มีสีดำ และ 0 จะหมายถึงภาพสีขาว การแปลงข้อมูลภาพหลายระดับเป็นภาพไบนารีจึงมีความจำเป็นและประโยชน์มากในการแสดงผลและประโยชน์อีกอย่างหนึ่งคือการลดเนื้อที่การเก็บข้อมูลภาพเหลือเพียง 8 บิต นั่นคือ 1 จุด ภาพจะใช้เนื้อที่การเก็บ 1 บิต ดังนั้นจึงสามารถนำไปวิเคราะห์ในขั้นต้น การประมวลผลขั้นต้น (Processing) ในการสร้างภาพไบนารี สามารถทำได้โดยใช้เทคนิคการทำเทรชโฮล (Thresholding Technics) โดยพิจารณาว่าจุดใดควรเป็นจุดขาวหรือจุดดำ จะกระทำโดยการเปรียบเทียบระหว่างจุดเริ่มต้นกับค่าคงที่ที่เรียกว่า ค่าเทรชโฮล เมื่อข้อมูลภาพที่ลักษณะแตกต่างกันระหว่างวัตถุและพื้นหลัง โดยค่าที่มีน้อยกว่าค่าเทรชโฮลจะถูกกำหนดให้เป็นค่า 1 (จุดสีดำ) และค่าที่มากกว่าค่าเทรชโฮลจะถูกกำหนดให้เป็นค่า 0 (จุดสีขาว)

$$b(x,y) = \begin{cases} 1: g(x,y) < \text{Thr} \\ 0: g(x,y) > \text{Thr} \end{cases} \quad (3.5)$$

โดยที่ $b(x,y)$ คือ ข้อมูลภาพผลลัพธ์เป็นภาพไบนารี

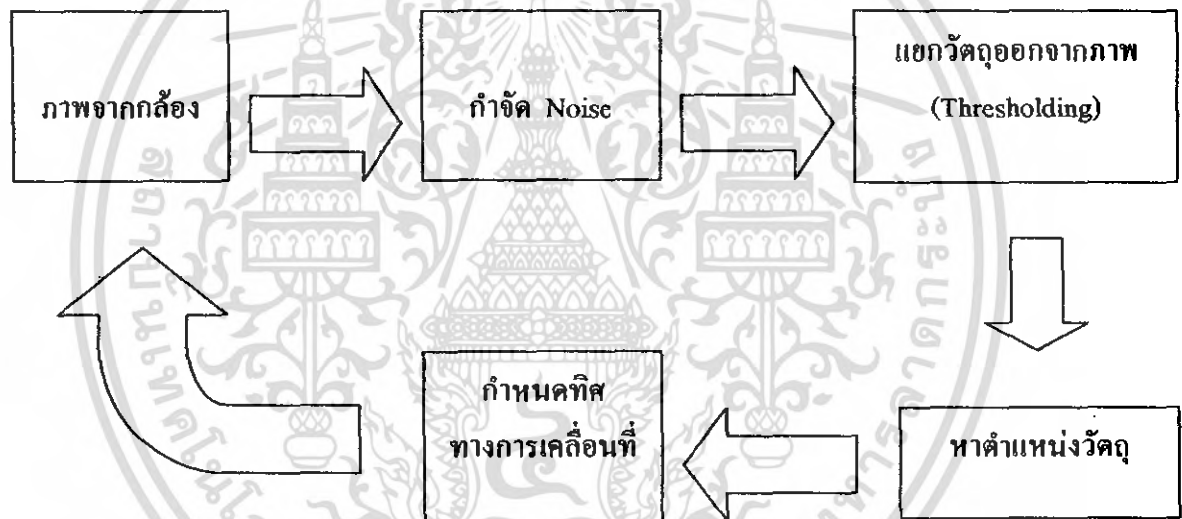
$g(x,y)$ คือ ข้อมูลภาพอินพุตที่มีระดับความเข้ม 0 ถึง 1 ระดับ

Thr คือ ค่าเทรชโฮล เป็นค่าคงที่

ในการสร้างภาพไบนารีโดยใช้เทคนิคเทรชโฮลเพื่อให้ได้ผลลัพธ์ที่เหมาะสมและคมชัด สิ่งสำคัญที่สุด คือ ค่าเทรชโฮล (Thr) เนื่องจากถ้าเลือกค่าเทรชโฮลที่ไม่เหมาะสมภาพที่ได้จะไม่เหมาะสมขาดความคมชัดและรายละเอียดของภาพจะขาดหายไป

3.2 เทคนิคการประมวลผลภาพ

จากภาพที่ได้จากกล้องวีดีโอนั้นจะถูกส่งมาประมวลผลที่คอมพิวเตอร์ด้วยขบวนการอิมเมจโปรเซสซิ่ง โดยมีเทคนิคของการประมวลผลภาพ สามารถแสดงเป็นบล็อกไดอะแกรมได้ดังนี้



รูป 3.2 Block Diagram การประมวลผลของ Soccer Robot

3.2.1 อิมเมจเอนฮานเม้นท์ (Image enhancement)

เป็นการจัดการภาพให้อยู่ในรูปที่เหมาะสมขึ้น หรือ ให้มีความคมชัดมากขึ้น เพราะว่าภาพที่ได้มาจากกล้องวีดีโอนั้นมีความผิดพลาดเกิดขึ้นระหว่างการส่งข้อมูลภาพเรียกว่า Noise ดังนั้นทำให้คุณภาพของภาพที่ได้ไม่ดีเท่าที่ควรจึงต้องกำจัด noise ซึ่งวิธีการกำจัดคือ พิจารณาข้อมูลในแต่ละ pixel จากนั้นทำการหาค่าเฉลี่ยของความแตกต่างของข้อมูลระหว่างจุดรอบ ๆ ของจุดที่พิจารณา กับจุดที่ต้องการพิจารณา เพื่อเป็นการแยกความแตกต่างของจุดภาพที่ใกล้เคียงกันเพื่อหาขอบเขตภาพที่เหมาะสมที่สุดดังนี้

$$x(i,j) = \frac{\sum_{i=r-1}^{r+1} \sum_{j=c-1}^{c+1} f(i,j) - f(r,c)}{2} \quad (3.6)'$$

โดยที่ $x(i,j)$ คือ ค่าของจุดภาพที่พิจารณาที่ถูกรักษาการกำจัด noise

$f(i,j)$ คือ ค่าของจุดภาพรอบ ๆ ข้างของจุดที่พิจารณา N จุด

$f(r,c)$ คือ ค่าของจุดภาพที่พิจารณา

r,i คือ ค่าแสดงตำแหน่งของแถวของจุดภาพ

c,j คือ ค่าแสดงตำแหน่งของคอลัมน์ของจุดภาพ

ในการพิจารณานั้นจะเห็นได้ว่าการหาค่าเฉลี่ยของความแตกต่างของข้อมูลจุดภาพที่พิจารณากับจุดรอบข้างใกล้เคียงจะมีข้อมูลใหม่ที่เหมาะสมกว่ารูปภาพที่ได้ตอนแรกที่ได้จากกล้องวิดีโอเพื่อนำไปวิเคราะห์ในกระบวนการต่าง ๆ ต่อไป

3.2.2 การแยกวัตถุออกจากภาพ (Segmentation)

เป็นเทคนิคที่สำคัญอีกอย่างหนึ่งในการประมวลผลภาพโดยจะแยกเอาวัตถุออกจากพื้นหลังเพื่อนำข้อมูลของวัตถุที่แยกออกมานำไปใช้ในงานที่ต้องการได้ การแยกวัตถุออกจากภาพนี้ทำได้หลายวิธีซึ่งจะเหมาะกับลักษณะการทำงานที่แตกต่างไป ซึ่งวิธีหนึ่งก็คือวิธีการของการเทรชโฮล (Thresholding) โดยการเทรชโฮลนี้จะเป็นการหาค่าเทรชโฮลที่เหมาะสมเพื่อกำหนดการสร้างภาพแบบไบนารีที่ได้กล่าวไปข้างต้น ซึ่งสามารถแยกวัตถุออกจากภาพได้โดยข้อมูล 1 บิต คือข้อมูล 1 และ 0 ดังนั้นค่าของเทรชโฮลนั้นจะมีวิธีการหาได้หลายวิธีคือ

3.2.2.1 การหาค่าเทรชโฮลโดยการกำหนดค่าล่วงหน้า (Preassigned Threshold Value)

การหาค่าเทรชโฮลโดยการกำหนดค่าล่วงหน้านี้เป็นวิธีที่ง่ายที่สุด เป็นการกำหนดค่าเทรชโฮลเองจากผู้ใช้งาน ซึ่งจะขึ้นอยู่กับประสบการณ์ของผู้ใช้นั้น ๆ โดยการเลือกค่าคงที่ค่าหนึ่งจะเป็นค่าที่อยู่ระหว่างค่าต่ำสุดกับค่าสูงสุด ของระดับความเข้มของข้อมูลอินพุต เช่นภาพอินพุต มีเกรย์สเกล 256 ระดับ จะมีค่าเทรชโฮลได้ถึง 0 - 255

3.2.2.2 การหาค่าเทรชโฮลจากค่ากลาง (Mid Range Threshold Value)

การหาค่าเทรชโฮลโดยพิจารณาจากค่ากลาง เป็นการหาค่าที่แตกต่างจากวิธีการแรก สำหรับวิธีการนี้จะคำนวณหาค่าเทรชโฮลโดยอัตโนมัติโดยไม่ต้องให้ผู้ใช้งานเป็นผู้กำหนด โดยวิธีนี้อาศัยการคำนวณพื้นฐานทางสถิติคือการหาค่าเฉลี่ยมาประยุกต์ใช้ค่าเทรชโฮลที่คำนวณได้จากค่ากึ่งกลางที่อยู่ระหว่างค่าระดับความเข้มสูงสุดและระดับความเข้มต่ำสุดของภาพอินพุต เมื่อทำการคำนวณค่าเทรชโฮลได้แล้ว ก็สามารถสร้างภาพไบนารีได้โดยนำค่าเทรชโฮลที่ได้มาใช้

เมื่อทำการเทรซโฮลแล้วจะได้ภาพที่เป็นไบนารี โดยที่ค่าเทรซโฮลเป็นตัวแบ่งระดับความเข้มของภาพเพื่อกำหนดข้อมูลเป็นไบนารี 0 กับ 1 ซึ่งค่าระดับความเข้มที่มีค่ามากกว่าค่าเทรซโฮลนั้นจะกำหนดให้เป็น 0 เป็นสีขาว ส่วนค่าระดับความเข้มที่ต่ำกว่าค่าเทรซโฮลจะกำหนดให้เป็น 1 เป็นสีดำ ดังนั้น จะเห็นการแบ่งแยกวัตถุออกจากภาพซึ่งเป็นพื้นหลังได้นำข้อมูลที่ได้ไปพิจารณาค่าตำแหน่งของวัตถุของภาพต่อไป

3.2.3 การหาจุดกึ่งกลางของวัตถุ (Center of Mass)

การหาค่าตำแหน่งกึ่งกลางของวัตถุใช้ข้อมูลที่ได้จากการแยกวัตถุออกจากพื้นที่หลังข้างต้นมา โดยการนำข้อมูลของตำแหน่ง (ij) ของวัตถุที่แยกออกมาได้ มาหาค่าเฉลี่ยของข้อมูลภายในวัตถุทั้งหมด จะแยกออกเป็น ค่าตำแหน่งจุดกึ่งกลางของแถว C_r และค่าตำแหน่งจุดกึ่งกลางของคอลัมน์ C_c หาได้จาก

$$C_r = \frac{\sum_{i=1}^n X_i}{N} \quad (3.7)$$

และ

$$C_c = \frac{\sum_{j=1}^n Y_j}{N} \quad (3.8)$$

โดย C_r คือ ค่าเฉลี่ยตำแหน่งของแถว

C_c คือ ค่าเฉลี่ยตำแหน่งของคอลัมน์

X_i คือ ค่าตำแหน่งของแถว

Y_j คือ ค่าตำแหน่งของคอลัมน์

ij คือ จำนวนเต็มเท่ากับ $1 \dots n$

N คือ จำนวนตำแหน่งทั้งหมด

เมื่อหาจุดกึ่งกลางของวัตถุได้แล้วคือ (C_r, C_c) โดยเป็นเทคนิคพื้นฐานทางสถิติที่สามารถหาค่าได้เช่นกันแต่ในความเป็นจริงของจุดกึ่งกลางของวัตถุจริง ๆ นั้นที่ได้มาจากกล้องตั้งแต่ค่าเริ่มต้นนั้นจะมีค่าอีกค่าหนึ่งดังนั้นความผิดพลาดที่เกิดขึ้นนั้นจะเกิดขึ้นระหว่างการใช้เทคนิคต่าง ๆ ตั้งแต่แรกเรื่อยมานั้นก็จะขึ้นอยู่กับความจำเป็นของสิ่งที่จะต้องพิจารณาว่าต้องการความละเอียดของชิ้นงานมากน้อยเพียงใดในโครงการนี้ต้องการหาค่าตำแหน่งของวัตถุบนพื้นหลังของภาพเท่านั้นว่ามีค่าเพียงใดเพื่อนำค่าที่ได้นั้นไปกำหนดทิศทางการเคลื่อนที่ของหุ่นยนต์ซึ่งเป็นวัตถุหนึ่งในภาพที่ให้เคลื่อนที่ไปหาลูกบอลแล้วเล็งลูกบอลและเข้าเป้าหมายได้อย่างอัตโนมัติ

นอกจากเทคนิคข้างต้นนี้แล้วยังสามารถใช้เทคนิคดังต่อไปนี้ได้เพื่อทำการประมวลผลภาพที่ต้องการ ดังนี้

- อิมเมจดิจิทัลไลเซชัน (Image digitization)

ดังที่ได้กล่าวมาแล้วนั้นถึงความหมายของการ Digitize ภาพ ซึ่งความละเอียดของภาพที่ได้ก็ขึ้นอยู่กับการจัดระดับภาพ ในปัจจุบันเครื่องมือที่ใช้ทำกระบวนการนี้เรียกว่า ดิจิไลเซอร์ ซึ่งสามารถเปลี่ยนสัญญาณอนาลอกเป็นสัญญาณดิจิทัลได้ ดังนั้นข้อมูลที่ได้อาจเก็บเป็นตัวเลขไบนารีโดยใช้ดิจิทเป็นตัวจัดการ

- อิมเมจเอนฮานซ์เมนต์และรีสตอเรชัน (Image enhancement and restoration)

อิมเมจเอนฮานซ์เมนต์ เป็นการทำให้ภาพให้อยู่ในรูปที่เหมาะสมขึ้น มีการจัดเจนมากขึ้นสำหรับการนำไปใช้งานเฉพาะอย่าง

3.2.4 วิธีการปรับปรุงคุณภาพของภาพ (Enhancement)

- คอนทราสต์เอนฮานซ์เมนต์ (Contrast enhancement) เป็นวิธีที่ทำให้ภาพคมชัดขึ้น โดยอาศัยวิธีฮิสโตแกรม อาจใช้แบบทีเนียร์สเตรท (linear stretch) , พีซทีเนียร์ - สเตรท (piecewise linear stretch) หรือ อีควอลไลเซชัน (Equalization)

- เอนฮานซ์เมนต์ (Edge enhancement) เป็นการแยกความแตกต่างของจุดภาพที่ใกล้เคียงกันเพื่อหาขอบเขตของภาพ

- การประมวลผลภาพสีเทียม (Pseudo - color image processing) เป็นการใช้เทคนิคของการทำ density slicing และใส่สีเทียมให้กับภาพขาว - ดำ ที่มีระดับ Grey - Level ต่าง ๆ กัน

- อิมเมจรีสตอเรชัน (Image restoration) เป็นกระบวนการสร้างภาพกลับคืน โดยการหาค่าชดเชย และแก้ความคลาดเคลื่อนเนื่องจากข้อมูลในภาพผิดพลาดไป หรือเป็นขบวนการสร้างภาพกลับคืน จากภาพที่ถูกทำให้เสียไปเนื่องจากปรากฏต่างโดยใช้หลักการของพีชคณิตเชิงเส้น

- อิมเมจเอ็นโค้ดดิ้ง (Image encoding)

เป็นการใช้เทคนิคต่าง ๆ เพื่อเข้ารหัสข้อมูลเนื่องจากข้อมูลภาพที่ได้จะถูกเก็บในลักษณะจำนวนไบนารีซึ่งถ้าภาพมีขนาดใหญ่มาก ก็ต้องใช้พื้นที่ในการเก็บมาก การเข้ารหัสข้อมูลจึงมีประโยชน์ในการลดข้อมูลเรียกว่าการลดข้อมูล (Data reduction หรือ Data compressing)

- อิมเมจคอนสตรัคชัน (Image reconstruction)

เป็นวิธีการสร้างภาพตัดขวางของวัตถุ โดยไม่ต้องผ่า หรือทำลาย เราเรียกการสร้างภาพตัดขวางว่า คอมพิวเตอร์โทโมกราฟี (Computer tomography)

3.3 การเห็นใน 3 มิติ(3D Vision)

ในบทนี้เราจะกล่าวถึงการเห็นใน 3 มิติซึ่งเป็นการดึงข้อมูลสามมิติออกจากภาพ 2 มิติ มีเหตุผลหลายประการที่ทำให้การเห็นใน 3 มิติ โดยใช้ข้อมูลความเข้มแสง 2 มิติมีความยาก เหตุผลดังกล่าวได้แก่

- ระบบภาพของกล้องและตาของมนุษย์มีการ โปรเจ็คชันแบบเพอสเปกทีฟ(Perspective Projection)ซึ่งทำให้มีการสูญเสียข้อมูลอย่างมาก ทุกจุดที่อยู่ในแนวเส้นที่ลากจากจุดกึ่งกลางของการโปรเจ็คชันถึงจุดในฉากถูกโปรเจ็คลงบนจุดเดียวกัน การเห็นใน 3 มิติเป็นขบวนการย้อนกลับซึ่งเป็นการหาพิกัด 3 มิติจากข้อมูลที่วัดได้จากภาพ ในการแก้ปัญหาที่เราจำเป็นต้องให้ข้อมูลเพิ่มเติม
- ความสัมพันธ์ระหว่างภาพใน 2 มิติและเรขาคณิตใน 3 มิติของจุดที่สอดคล้องกันในฉากมีความซับซ้อนมาก ทั้งนี้เนื่องจากค่าความเข้มของพิกเซลขึ้นกับพารามิเตอร์การสะท้อนแสงของผิว การวางตัวของผิวชนิดและตำแหน่งของแสงที่ตกกระทบ ตำแหน่งของผู้สังเกต
- การบังกันของวัตถุในฉากเป็นการเพิ่มความยุ่งยากของการมองเห็นในภาพใน 3 มิติ
- การปรากฏอยู่ของนอยส์ในภาพ

3.3.1 การแปลงเรขาคณิตของวัตถุใน 3 มิติ (Geometric Transformation in 3D)

การแปลงเรขาคณิตสำหรับวัตถุใน 2 มิติจะมีพิกัดอยู่ในรูป(x,y) ซึ่งต่างกับพิกัด 3 มิติซึ่งมีพิกัดอยู่ในรูป(x,y,z) ซึ่งเราจะกล่าวถึงเฉพาะสามมิติดังนี้

3.3.1.1 การเคลื่อนย้าย(Translation)

สมมุติว่าเราต้องการย้ายจุดที่มีพิกัด(x,y,z) ไปยังจุดใหม่ที่มีระยะเคลื่อนที่ (t_x, t_y, t_z)

$$u=x+t_x$$

$$v=y+t_y$$

$$w=z+t_z$$

(3.9)

โดยที่(u,v,w)คือพิกัดใหม่ของจุด สมการที่(3.9) สามารถเขียนในพิกัดโฮโมจีเนียสได้เป็น

$$\begin{bmatrix} u \\ v \\ w \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3.10)$$

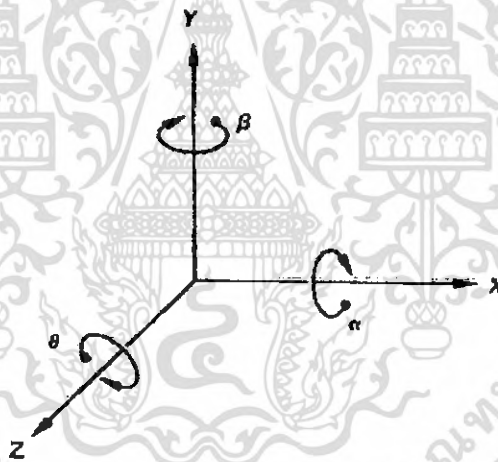
3.3.1.2 การย่อขยายใน 3 มิติ (Scaling)

การย่อขยายใน 3 มิติที่มีพารามิเตอร์ S_x, S_y และ S_z ตามแนวแกน x, y และ z สามารถเขียนในรูปเมทริกซ์ได้เป็น

$$\begin{bmatrix} u \\ v \\ w \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3.11)$$

3.3.1.3 การหมุนใน 3 มิติ (Rotation)

การหมุนใน 3 มิติสามารถแบ่งได้เป็น 3 ชนิดนั่นคือหมุนรอบแกน x แกน y และแกน z เช่นเดียวกับกรณี 2 มิติ ในการหมุนวัตถุรอบจุดใดๆ เราต้องทำการย้ายจุดนั้นมาที่จุดกำเนิดเสียก่อน จากนั้นทำการหมุน ก่อนที่จะย้ายวัตถุกลับที่เดิม



รูป 3.3 แกนพิกัด 3 มิติและการหมุนจุดรอบแกนพิกัด มุมของการหมุนวัดตามเข็มนาฬิกาเมื่อมองจากแกนสู่จุดกำเนิด

พิจารณาแกนพิกัด 3 มิติที่แสดงในรูปที่ 1 เมทริกซ์สำหรับการหมุนรอบแกน z แทนด้วย

$$R_\theta = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.12)$$

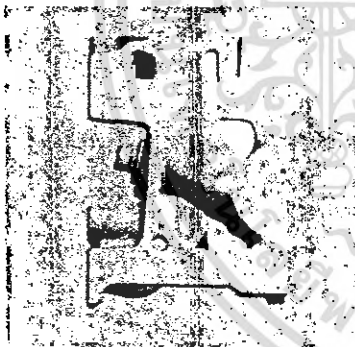
มุมของการหมุนวัดตามเข็มนาฬิกา เมื่อมองไปที่จุดกำเนิดจากแกน +z การหมุนรอบแกน x ด้วยมุม α สามารถแทนด้วยเมทริกซ์

$$R_\alpha = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.13)$$

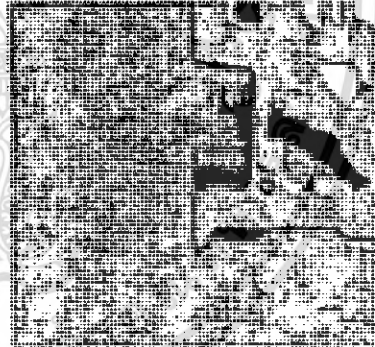
สุดท้ายการหมุนรอบแกน y ด้วยมุม β สามารถแทนด้วยเมทริกซ์

$$R_\beta = \begin{bmatrix} \cos \beta & 0 & -\sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.14)$$

เช่นเดียวกับกรณี 2 มิติ เมทริกซ์การแปลงเรขาคณิตใน 3 มิติ สามารถคูณกันเพื่อรวมการแปลง และการแปลงย้อนกลับก็ถูกนิยามในลักษณะเดียวกับกรณี 2 มิติ



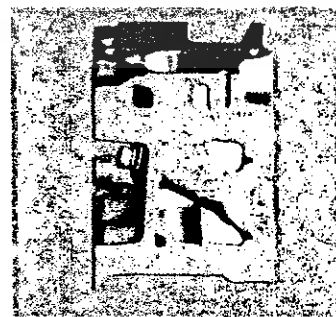
(a)



(b)

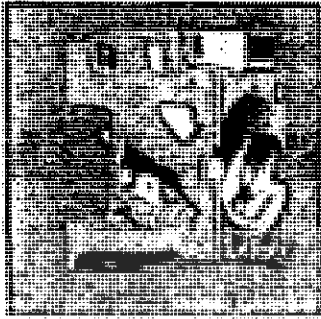


(c)



(d)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(e)



(f)

รูป 3.4 แสดงตัวอย่างการแปลงเรขาคณิตของวัตถุใน 3 มิติ รูป a) วัตถุเดิม; b) การย้ายวัตถุ; c) การย่อขยายวัตถุ; d) การหมุนรอบแกน x; e) การหมุนรอบแกน y; f) การหมุนรอบแกน z

3.3.2 เรขาคณิตสำหรับการเห็นใน 3 มิติ (Geometry for 3D Vision)

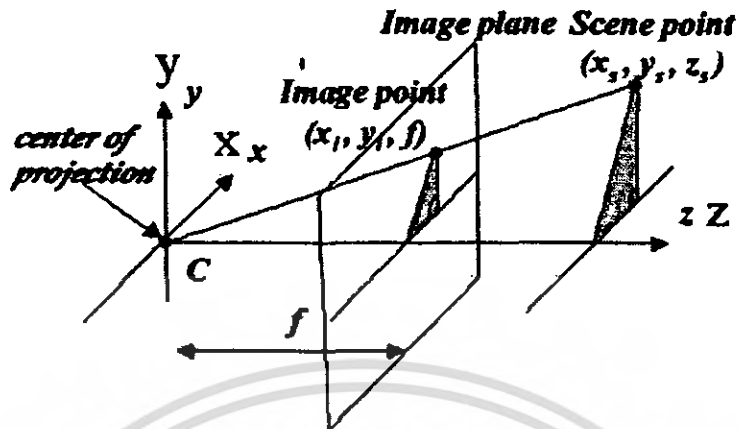
3.3.2.1 พื้นฐานของการโปรเจกชันแบบเพอสเปกทีฟของกล้อง (Basis of Perspective Camera)

ภาพ 2 มิติส่วนใหญ่ได้มาจากการมองวัตถุ 3 มิติผ่านเลนส์ของตาหรือเลนส์ของกล้อง ขบวนการเกิดภาพสามารถถูกจำลองได้โดยการใช้ Perspective Transformation รูปที่ 3.5 แสดงแบบจำลองง่ายๆ ของระบบสร้างภาพ เราแทนแกนระบบพิกัดโลก (World Coordinate System) ด้วยอักษรตัวใหญ่ (X, Y, Z) และแทนระบบพิกัดของกล้องด้วยอักษรตัวเล็ก (x, y, z) เราสมมุติว่าพิกัดของโลกซ้อนทับกับระบบพิกัดของกล้อง แสงจากวัตถุใน 3 มิติส่องผ่านศูนย์กลางของการโปรเจกชันแล้วโปรเจกต์ลงบนภาพให้จุด (x_i, y_i, z_i) แทนจุดพิกัดของวัตถุ 3 มิติ ให้ (x_i, y_i) แทนจุดพิกัดของวัตถุในระนาบของภาพ เราสมมุติว่าระนาบของภาพอยู่ f ห่างไปทางขวาของจุดกำเนิดของพิกัดกล้อง โดยที่ f แทนความยาวโฟกัสของเลนส์ โดยการใช้สามเหลี่ยมคล้ายเราจะได้ว่า

$$x_i = \frac{fx_s}{z_s} \quad (3.15)$$

$$y_i = \frac{fy_s}{z_s} \quad (3.16)$$

ดังนั้นจุดที่โปรเจกต์ (x_i, y_i) มีความสัมพันธ์แบบไม่เป็นเชิงเส้นกับจุดของวัตถุ (x_s, y_s, z_s) ความสัมพันธ์นี้สามารถมองเห็นได้ง่ายถ้าเราใช้ระบบพิกัดโฮโมจีเนียส



รูป 3.5 แบบจำลองของระบบสร้างภาพ

ให้

$$v = \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix} \quad (3.17)$$

เป็นเวกเตอร์ที่ประกอบด้วยจุดพิกัดของวัตถุ เวกเตอร์ในระบบโฮโมจีเนียส \tilde{v} ของ v คือ

$$\tilde{v} = \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \quad (3.18)$$

เวกเตอร์ในระบบพิกัด Cartesian v สามารถได้จากเวกเตอร์ในระบบโฮโมจีเนียสได้โดยการหารสามสมาชิกแรกด้วยสมาชิกที่สี่

พิจารณาเมทริกซ์ของ Perspective Transformation

$$P = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (3.19)$$

ทำการคูณเวกเตอร์

$$\tilde{w} = p\tilde{v} \quad (3.20)$$

ได้เป็น

$$\tilde{w} = \begin{bmatrix} fx_s \\ fy_s \\ z_s \end{bmatrix} \quad (3.21)$$

ทำการนอร์มอลไลซ์สมาชิกที่สามให้เป็น 1 เพื่อแปลงเป็นเวกเตอร์ในระบบพิกัด Cartesian คำนึง

$$w = \begin{bmatrix} \frac{fx_s}{z_s} \\ \frac{fy_s}{z_s} \\ 1 \end{bmatrix} \quad (3.22)$$

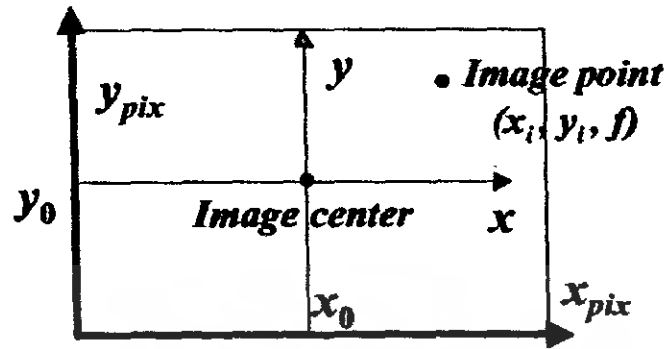
สังเกตว่าสองสมาชิกแรกของ w สอดคล้องกับพิกัดของภาพที่ได้ในสมการ(3.15) และ(3.16)

3.3.2.2 การแปลงจากหน่วยความยาวเป็นพิกเซล

ขั้นตอนต่อไปคือการหาพิกัดของจุดที่ถูกโปรเจกชัน (x_p, y_p, f) ในหน่วยของพิกเซล กำหนดให้พิกัดของภาพมีจุดกำเนิดอยู่ที่มุมล่างซ้ายของภาพดังแสดงในรูปที่ 3.6 การแปลงจากความยาวเป็นพิกเซลเราจำเป็นต้องรู้ค่าการปรับเสถียร (Aspect Ratio) ของแต่ละแกนของระบบพิกัดภาพค่าการปรับเสถียรเป็นค่าที่ระบุจำนวนพิกเซลต่อหนึ่งหน่วยความยาว ในระบบภาพบางระบบพิกเซลในแนวตั้งไปอาจมีการเอียงกัน จำเป็นต้องรู้ค่าที่เรียกว่าพารามิเตอร์ของการเอียง (Skew parameter) นอกจากนี้ยังจำเป็นต้องรู้จุดหลัก (Principal Point) $U_0=(x_0, y_0)$ หรือบางครั้งถูกเรียกว่าจุดกึ่งกลางของภาพของขบวนการปรับ โดยที่ k_x คือการปรับเสถียรในทิศทาง y และ s แทนพารามิเตอร์ของการเอียง เราเขียนในรูปเทียบกล้อง (Camera Calibration Procedure) เป็นจุดตัดระหว่างแกนออบติคกับระนาบภาพ

พิกัดของจุด (x_p, y_p, f) หาได้จากความสัมพันธ์

$$\begin{aligned} x_{px} &= k_x x_1 + s y_1 + x_0 \\ &= k_x f \frac{x_s}{z_s} + s f \frac{y_s}{z_s} + x_0 \end{aligned} \quad (3.33)$$



รูป 3.6 พิกัดของภาพ

$$\begin{aligned} y_{pix} &= k_y y_i + y_0 \\ &= k_y f \frac{y_s}{z_s} + y_0 \end{aligned} \quad (3.34)$$

เราสามารถอยู่ในรูปของเมทริกซ์ได้เป็น

$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = \begin{bmatrix} \alpha_x & s & u_0 & 0 \\ 0 & \alpha_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \quad (3.35)$$

โดยที่ $\alpha_x = k_x x_s$ ซึ่งเป็นค่าความยาวโฟกัสในทิศ x ในหน่วยของพิกเซลและ $\alpha_y = k_y y_s$ ซึ่งเป็นค่าความยาวโฟกัสในทิศ y ในหน่วยของพิกเซล เราสามารถเขียน

$$\begin{bmatrix} \alpha_x & s & u_0 & 0 \\ 0 & \alpha_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} \alpha_x & s & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = K [I_3 | O_3] \quad (3.36)$$

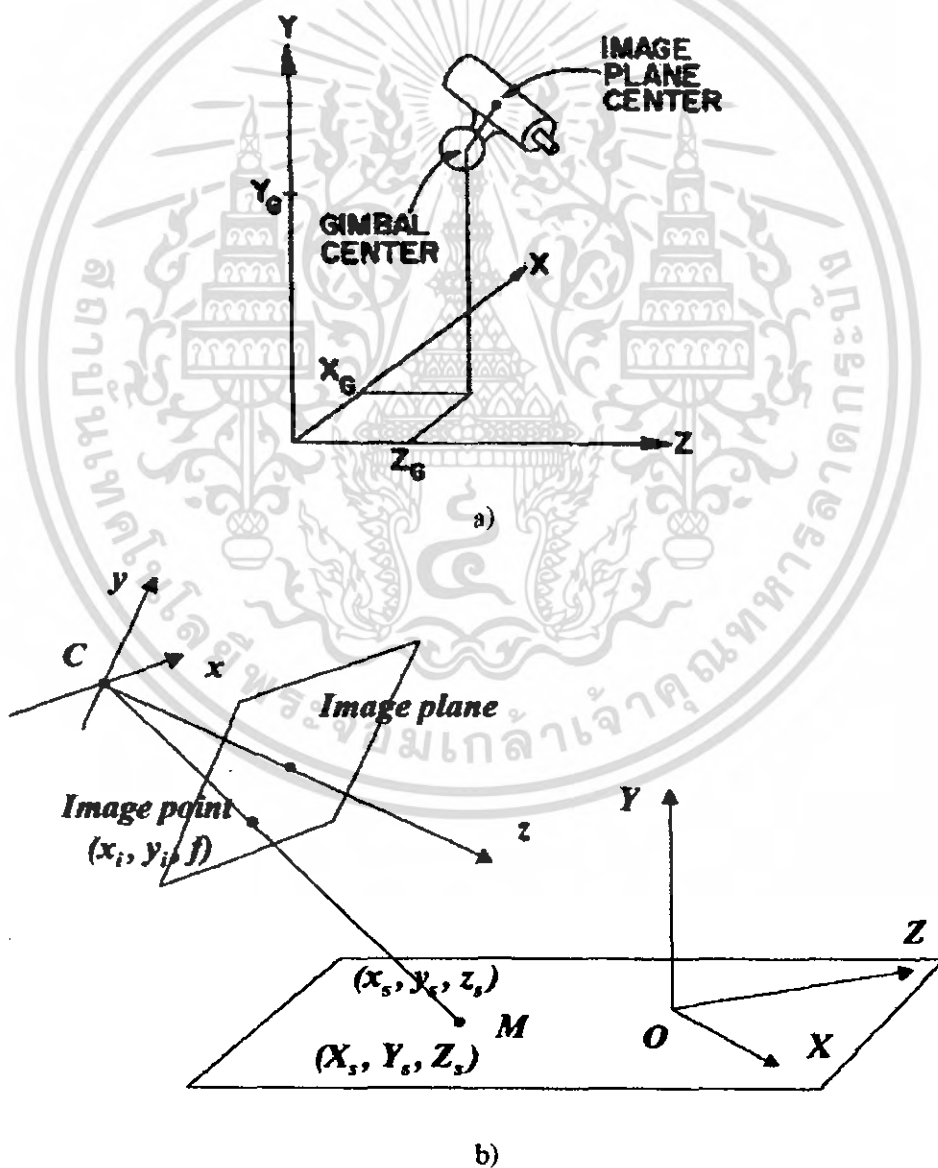
เมทริกซ์ K มีลักษณะเป็นสามเหลี่ยมข้างบน (Upper Triangle) ขนาด 3×3 มีชื่อว่า Calibration Matrix เป็นเมทริกซ์ที่ประกอบพารามิเตอร์ภายใน (Intrinsic Parameter) ที่สำคัญของกล้อง

3.3.2.3 กรณีที่พิกัดของโลกไม่ซ้อนทับกับพิกัดของกล้อง

แบบจำลองการเกิดภาพที่ได้อธิบายในหัวข้อก่อนหน้าที่ตั้งสมมติฐานที่ว่า จุดศูนย์กลางของระนาบภาพเป็นจุดศูนย์กลางเดียวกับของระบบพิกัดอ้างอิงของโลก (World Reference Coordinate System) ทั้งนี้เพื่อให้เกิดความสะดวกในการวิเคราะห์ ในหัวข้อนี้เราจะกล่าวถึงแบบจำลองการเกิดภาพจากกล้องที่ไม่มีสมมติฐานดังกล่าว

รูป 3.7 a) แสดงกล้องอเล็กทรอนิกส์ในช่องว่างพิกัดของโลก (World Coordinate Space) กล้องนี้ถูกยึดโดยหัวยึดกล้อง (Gimbal) ที่สามารถหมุนซ้าย (Pan) ทำมุม β กับแนวนอน (Horizontal) และเอียง (Tilt) ทำมุม α กับแนวตั้ง จุดกึ่งกลางหัวยึดกล้องอยู่ที่จุดพิกัด (X_G, Y_G, Z_G) เทียบกับระบบพิกัดอ้างอิงของโลก จุดกึ่งกลางหัวยึดกล้องและจุดกึ่งกลางของการโปรเจกชันของระนาบภาพอยู่ห่างกันด้วยระยะ (X_0, Y_0, Z_0) รูป 3.7 b) แสดงระดับพิกัดโลกและระบบพิกัดกล้องที่ไม่ซ้อนทับกัน

กรณีจุดกึ่งกลางของการโปรเจกชันของกล้องถูกวางไว้ที่จุดกึ่งกลางของระบบพิกัดอ้างอิงของโลก ไม่มีการส่ายหรือเอียงทำมุมกับแกนอ้างอิง จุดกึ่งกลางของหัวยึดกล้องและของ



รูป 3.7 a) แบบจำลองการเกิดภาพจากกล้อง b) เรขาคณิตของการเกิดภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระนาบภาพอยู่ที่เดียวกัน แบบจำลองการเกิดภาพ (Imaging Model) ในระบบพิกัดโฮโมจีเนียสนั้นคือ

$$\tilde{w} = M\tilde{v} \quad (3.37)$$

โดยที่ \tilde{v} คือ เวกเตอร์ของจุดบนวัตถุในระบบพิกัดโฮโมจีเนียส \tilde{w} คือเวกเตอร์ของจุดบนระนาบภาพในระบบพิกัดโฮโมจีเนียส และ M คือเมทริกซ์ของ Perspective Transformation ในกรณีนี้แบบจำลองการเกิดภาพจากกล้องสามารถหาได้ง่ายโดยการเปลี่ยนแปลง (Modify) สมการที่ (3.37)

การที่ (หัวขีดกล้อง) กล้องอยู่ที่พิกัด (X_G, Y_G, Z_G) เทียบกับระบบพิกัดอ้างอิงของโลกดังนั้นแบบจำลองใหม่คือ

$$\tilde{w} = MT_G\tilde{v} \quad (3.38)$$

โดยที่

$$T_G = \begin{bmatrix} 1 & 0 & 0 & -X_G \\ 0 & 1 & 0 & -Y_G \\ 0 & 0 & 1 & -Z_G \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.39)$$

การส่ายหรือเอียง (Pan and Tilt) สามารถถูกจำลองโดยใช้คูณด้วยเมทริกซ์การหมุน

$$\tilde{w} = MRT_G\tilde{v} \quad (3.40)$$

โดยที่ $R = R_\beta R_\alpha$

$$R_\beta = \begin{bmatrix} \cos \beta & 0 & -\sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.41)$$

และ

$$R_\alpha = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.42)$$

ท้ายสุดจุดกึ่งกลางของหัวขีดกล้องและจุดกึ่งกลางของกลางของการโปรเจ็คชันของระนาบภาพอยู่ห่างกันด้วยระยะ (X_0, Y_0, Z_0) ดังนั้นแบบจำลองของการเกิดภาพจากกล้องคือ

$$\tilde{w} = MT_cRT_G\tilde{v} \quad (3.43)$$

โดยที่

$$T_c = \begin{bmatrix} 1 & 0 & 0 & -X_0 \\ 0 & 1 & 0 & -Y_0 \\ 0 & 0 & 1 & -Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.44)$$

สมการ(3.43)สามารถเขียนใหม่ได้เป็น

$$\begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = K [I_3 | O_3] \begin{bmatrix} R & -T \\ O_3^T & 1 \end{bmatrix} \begin{bmatrix} x_s \\ y_s \\ z_s \\ 1 \end{bmatrix} \quad (3.45)$$

โดยที่ R แทนเมทริกซ์ที่เป็นผลรวมของการย้ายระบบพิกัด (ขนาด 3×3 ของระบบพิกัดธรรมดา) และ T แทนเวกเตอร์ที่เป็นผลรวมของการย้ายระบบพิกัด (ขนาด 3×1) เรากล่าวว่าเมทริกซ์

$$\begin{bmatrix} R & -T \\ O_3^T & 1 \end{bmatrix} \quad (3.46)$$

เป็นเมทริกซ์ที่รวมเอาพารามิเตอร์ภายนอก (Extrinsic Parameter) ของขบวนการปรับเทียบกล้อง เราสามารถเขียนสมการ (3.45) ในรูปที่ง่ายได้เป็น

$$x = MX \quad (3.47)$$

$$\text{โดยที่ } M = [KR | -KRT] \quad (3.48)$$

และ M มีขนาด 3×4

3.4 การปรับเทียบกล้อง (Camera Calibration)

การปรับเทียบกล้องเป็นการหาประมาณเมทริกซ์ M จากจุดในฉาก 3 มิติที่รู้ตำแหน่งและภาพที่เกิดขึ้นของจุด จากนั้นทำการหาพารามิเตอร์ภายในและพารามิเตอร์ภายนอกพิจารณาการปรับเทียบโดยใช้กล้องตัวเดียว

ในการหาเมทริกซ์ M เราทำการหาจุดในฉากที่เราทราบพิกัด $X = [x, y, z]^T$ และจุดในภาพ 2 มิติที่สอดคล้องกัน $[u, v]^T$ ซึ่งจะได้สมการที่อยู่ในรูปสมการ (3.47) ดังนี้

$$\begin{bmatrix} \alpha u \\ \alpha v \\ \alpha \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3.49)$$

$$\begin{bmatrix} cu \\ \alpha v \\ \alpha \end{bmatrix} = \begin{bmatrix} m_{11}x + m_{12}y + m_{13}z + m_{14} \\ m_{21}x + m_{22}y + m_{23}z + m_{24} \\ m_{31}x + m_{32}y + m_{33}z + m_{34} \end{bmatrix} \quad (3.50)$$

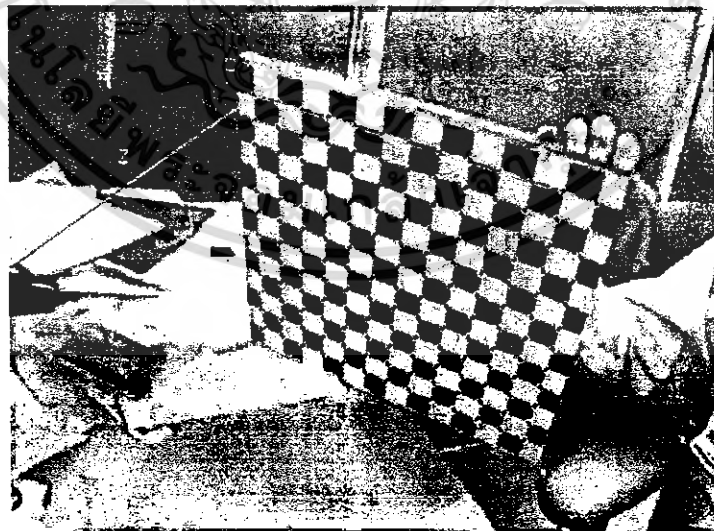
ทำการนอร์มอลไลซ์ให้สมาชิกที่ 3 เป็น 1 เราจะได้สมการ

$$\begin{aligned} u(m_{31}x + m_{32}y + m_{33}z + m_{34}) &= m_{11}x + m_{12}y + m_{13}z + m_{14} \\ v(m_{31}x + m_{32}y + m_{33}z + m_{34}) &= m_{21}x + m_{22}y + m_{23}z + m_{24} \end{aligned} \quad (3.51)$$

ดังนั้นเราจะจะได้สมการเชิงเส้น 2 สมการสำหรับจุดในฉาก 3 มิติหนึ่งจุดและจุดในภาพ 2 มิติที่สอดคล้องกัน ทำการเขียนสมการ(3.51)ใหม่ในรูปของเมทริกซ์ได้เป็น

$$\begin{bmatrix} x & y & z & 1 & 0 & 0 & 0 & 0 & -ux & -uy & -uz & -u \\ 0 & 0 & 0 & 0 & x & y & z & 1 & -vx & -vy & -vz & -v \\ & & & & & & & & & & & \vdots \\ & & & & & & & & & & & m_{34} \end{bmatrix} \begin{bmatrix} m_{11} \\ m_{12} \\ \vdots \\ m_{34} \end{bmatrix} = 0 \quad (3.52)$$

AM=0



รูป 3.8 ภาพของตารางหมากรุกที่นำมาใช้ในการปรับเทียบกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่าเรามีตัวแปรที่ไม่ทราบค่า 11 ตัวแปร แทนที่จะเป็น 12 ตัวแปรเนื่องจากแพ็คเกจการย่อ/ขยายที่ไม่สามารถทราบค่าได้ ในการแก้สมการโฮโมจีเนียสเราต้องใช้อย่างน้อย 6 จุด ถ้ามีมากกว่า 6 จุดเราจะได้สมการ Over-determined ซึ่งสามารถแก้ได้โดยวิธี Least Square หรือใช้การหา Singular Value Decomposition (SVD) ของ A ของเมทริกซ์ M ได้จากแถวสุดท้ายของ V เมื่อได้เมทริกซ์ M แล้วขั้นต่อไปทำการแยกพารามิเตอร์ภายใน (Intrinsic Parameter) เนื่องจาก

$$M = [KR | -KRT] = [A|b] \quad (3.53)$$

ดังนั้นเมทริกซ์ย่อย 3×3 แทนด้วย A เวกเตอร์คอลัมน์ขวามือสุดแทนด้วย b เวกเตอร์ที่มีพารามิเตอร์ระยะเคลื่อนที่สามารถแยกออกมาได้โดยง่ายโดย $t = -A^{-1} b$ เวกเตอร์ t ให้ข้อมูลที่บอกถึงตำแหน่งกึ่งกลางของ Image Plane

จากนั้นพิจารณา $A=KR$ โดยที่ K เป็นเมทริกซ์สามเหลี่ยมบน (Upper Triangle) และเมทริกซ์ R เป็นเมทริกซ์ออร์โธกอนอล การแยกเมทริกซ์ K และ R สามารถทำได้โดยใช้เทคนิค QR Decomposition สำหรับเมทริกซ์

3.5 การหาพิคต 3 มิติของจุดในฉากโดยใช้กล้อง 2 ตัว

ในหัวข้อนี้เราพิจารณาการหาพิคต 3 มิติของจุดในฉากโดยใช้กล้อง 2 ตัว โดยที่กล้องทั้งสองตัวผ่านขบวนการปรับเทียบกล้อง ในขบวนการปรับเทียบเราให้กล้องทั้งสองถ่ายภาพวัตถุในฉาก 3 มิติที่เราทราบพิคต ซึ่งอาจเป็นตารางหมากรุกที่เราทราบขนาดของแต่ละตาราง แสดงดังรูปที่ 3.8 จากนั้นทำการหาพิคตของจุดตัดของตารางในกล้องทั้งสอง ใช้สมการที่ 3.52 ทำการคำนวณหาเมทริกซ์ของ Perspective Transformation M ของกล้องที่ 1 และ 2 เป็น M และ M' ตามลำดับ เราแทนแต่ละแถวของเมทริกซ์ M ด้วย m_1^T, m_2^T, m_3^T ทำนองกันสำหรับกล้องที่สอง เราแทนแต่ละแถวของเมทริกซ์ M' ด้วย $m_1'^T, m_2'^T, m_3'^T$ สำหรับกล้องที่ 1 เราได้

$$u = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = MX = \begin{bmatrix} m_1^T \\ m_2^T \\ m_3^T \end{bmatrix} x \quad (3.54)$$

ทำนองเดียวกันสำหรับกล้องที่ 2 เราได้

$$u' = \begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} = M'X = \begin{bmatrix} m_1'^T \\ m_2'^T \\ m_3'^T \end{bmatrix} X \quad (3.55)$$

โดยที่ u และ u' แทนพิกัดในภาพของวัตถุในกล้อง 1 และ 2 ตามลำดับในการกำจัดค่าสเกลที่ไม่ทราบค่า เราทำการหาสัดส่วนระหว่างสามแถวในเมทริกซ์ Perspective Transformation M

$$u:v:w = m_1^T X : m_2^T X : m_3^T X \\ u' : v' : w' = m_1'^T X : m_2'^T X : m_3'^T X \quad (3.56)$$

ดังนั้นเราได้สมการสำหรับกล้อง 1 และ 2

$$\begin{aligned} um_2^T X = vm_1^T X & \quad u'm_2'^T X = v'm_1'^T X \\ um_3^T X = vm_2^T X & \quad u'm_3'^T X = v'm_2'^T X \\ um_3^T X = vm_3^T X & \quad u'm_3'^T X = v'm_3'^T X \end{aligned} \quad (3.57)$$

สมการ (3.56) สามารถเขียนได้เป็น

$$\begin{bmatrix} um_2^T & -vm_1^T \\ um_3^T & -wm_1^T \\ vm_3^T & -wm_2^T \end{bmatrix} X = 0 \quad (3.58)$$

และ

$$\begin{bmatrix} u'm_2'^T & -v'm_1'^T \\ u'm_3'^T & -w'm_1'^T \\ v'm_3'^T & -w'm_2'^T \end{bmatrix} X = 0 \quad (3.59)$$

ทำการคูณแถวแรกของสมการ (3.58) ด้วย w แถวที่สองด้วย $-v$ แล้วทำการบวก เราได้

$$(uwm_2^T - vwm_1^T - uvm_3^T + vwm_3^T)X = (uwm_2^T - uvm_3^T)X = 0 \quad (3.60)$$

จะเห็นว่าสมการที่ (3.60) มีคุณสมบัติ Linearly Dependent กับแถวที่สามของสมการที่ (3.58) ดังนั้นแต่ละกล้องเราใช้ได้เพียง 2 สมการดังนี้

$$\begin{bmatrix} um_3^T & -wm_1^T \\ vm_3^T & -wm_2^T \end{bmatrix} X = 0 \quad (3.61)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{และ } \begin{bmatrix} u'_3 m_1^T & -w'_1 m_1^T \\ v'_3 m_2^T & -w'_2 m_2^T \end{bmatrix} X = 0 \quad (3.62)$$

เขียนสมการ (3.61) และ (3.62) ใหม่ได้ดังนี้

$$\begin{bmatrix} um_3^T & -wm_1^T \\ vm_3^T & -wm_2^T \\ u'_3 m_1^T & -w'_1 m_1^T \\ v'_3 m_2^T & -w'_2 m_2^T \end{bmatrix} X = AX = 0 \quad (3.63)$$

ทำการแก้สมการโฮโมจีเนียสสมการ (3.63) สำหรับพิกัด 3 มิติ X โดยเราสนใจเฉพาะคำตอบประเภท Non-Trivial หรือกรณี $\det(A)=0$

สำหรับกรณีของการใช้กล้อง 3 กล้องหรือมากกว่าเราได้สมการในลักษณะเดียวกัน เช่นถ้าเราใช้กล้อง 3 ตัวดังแสดงในรูปที่ 3.9 เราได้สมการ



รูป 3.9 การหาพิกัด 3 มิติโดยใช้กล้อง 3 ตัว

$$\begin{bmatrix} u'_3 m_1^T & -w'_1 m_1^T \\ v'_3 m_2^T & -w'_2 m_2^T \\ u''_3 m_1^T & -w''_1 m_1^T \\ v''_3 m_2^T & -w''_2 m_2^T \\ u'''_3 m_1^T & -w'''_1 m_1^T \\ v'''_3 m_2^T & -w'''_2 m_2^T \end{bmatrix} X = AX = 0 \quad (3.64)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าพิกัด 3D ในเมทริกซ์ X หาได้โดยการทำ Singular Value Decomposition (SVD) ของ A เมทริกซ์ X คือแถวสุดท้ายของ V

3.6 การเกิดภาพสเตอริโอ (Stereo Imaging)

การเกิดภาพสเตอริโอเป็นการสร้างภาพของวัตถุเดียวกันด้วยกล้อง 2 ตัววางอยู่ข้างกัน ภาพที่ได้จากกล้องทั้งสองสามารถนำมาใช้ในการหาพิกัด 3 มิติของวัตถุได้ พิจารณาแบบจำลองการเกิดภาพสเตอริโอในรูปที่ 3.10 ระยะระหว่างจุดกึ่งกลางของเลนส์ของกล้องทั้งสองเราเรียกว่า Baseline B เราต้องการหาพิกัด (X, Y, Z) ของจุด w ซึ่งมีจุดภาพ (x_1, y_1) และ (x_2, y_2) เราสมมติว่ากล้องทั้งสองนั้นมีลักษณะที่เหมือนกัน และระบบพิกัดของกล้องทั้งสองนั้นวางอยู่ในแนวที่ตรงกัน (Perfectly Aligned) ถ้าระบบพิกัดของกล้องและของโลกซ้อนทับกันพอดี ระนาบ xy ของภาพวางอยู่ในแนวเดียวกับระนาบ XY ของโลก ดังนั้นค่าพิกัด z ของ w สำหรับระบบพิกัดกล้องทั้งสองมีค่าเดียวกัน ถ้าระบบพิกัดของกล้องที่หนึ่งและของโลกซ้อนทับกันพอดี ดังแสดงในรูปที่ 3.10 เราได้ว่า

$$X_1 = \frac{x_1}{f} (f - Z_1) \quad (3.65)$$

โดยที่ตัวห้อยบน X และ Z เป็นการบอกว่าพิกัดกล้องที่หนึ่งซ้อนทับกับพิกัดของโลก ทำนองเดียวกัน ถ้าระบบพิกัดของกล้องที่สองและของโลกซ้อนทับกันพอดี เราได้ว่า

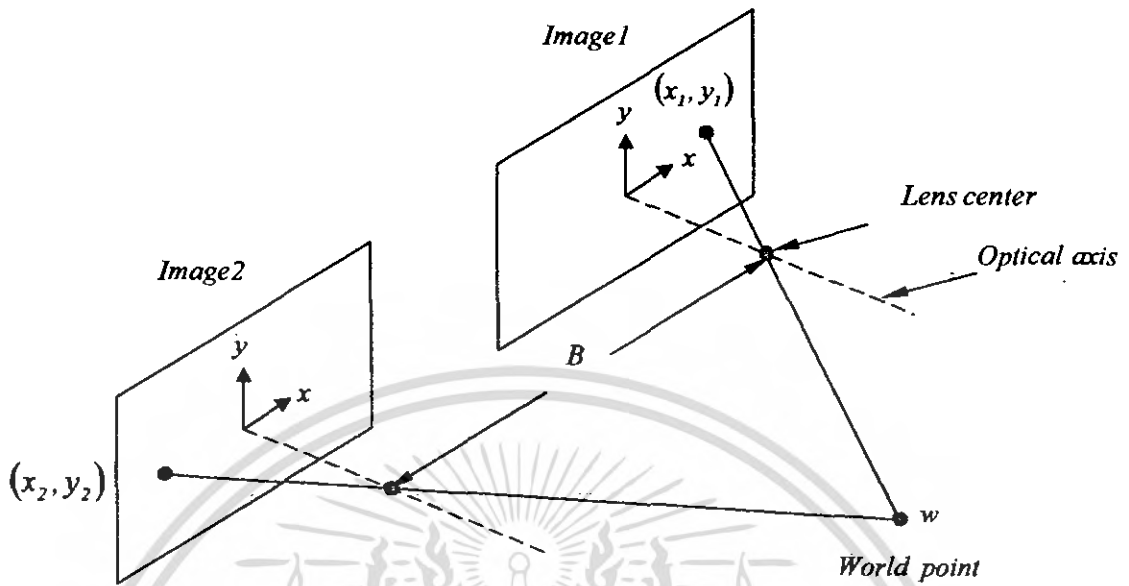
$$X_2 = \frac{x_2}{f} (f - Z_2) \quad (3.66)$$

เนื่องจากการแยกกันของกล้องทั้งสองและเนื่องจาก w เป็นค่าเดียวสำหรับระบบพิกัดของทั้งสองกล้อง เราได้ว่า

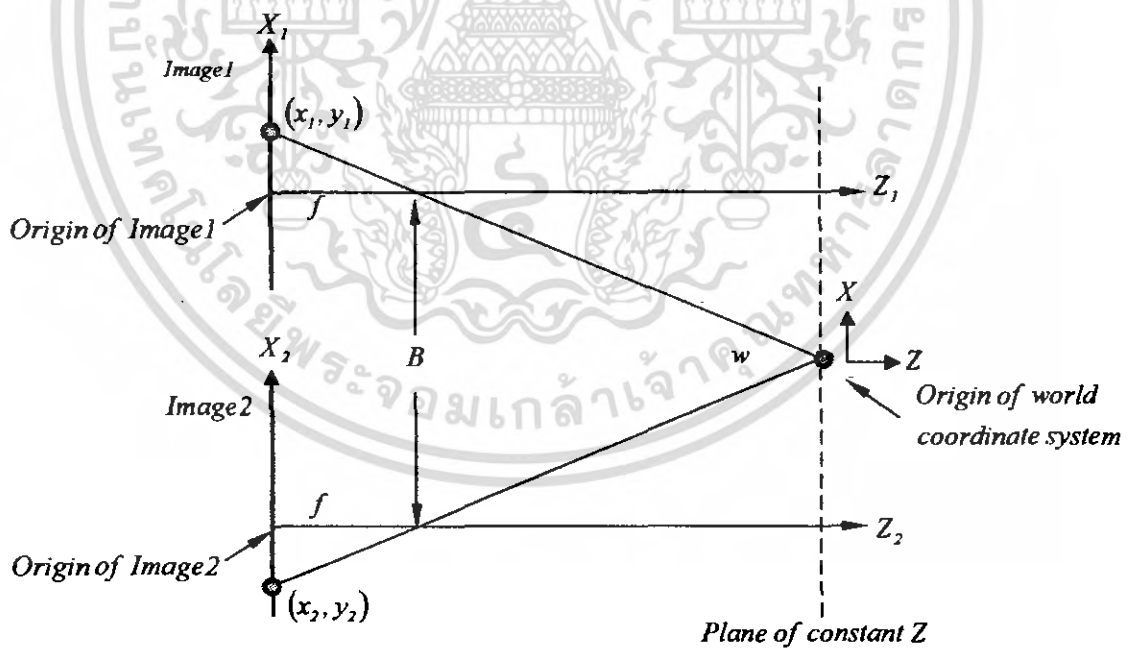
$$X_2 = X_1 + B \quad (3.67)$$

และ $Z_2 = Z_1 = Z \quad (3.68)$

โดยที่ B เป็นระยะ Baseline



รูป 3.10 แบบจำลองของขบวนการสร้างภาพสเตอริโอ



รูป 3.11 มุมบนของขบวนการสร้างภาพสเตอริโอ โดยที่ระบบพิกัดของกล้องตัวที่ 1 ซ้อนทับกับระบบพิกัดของโลก

แทนค่า (3.67) และ (3.68) ลงใน (3.65) และ (3.66) เราได้ว่า

$$X_1 = \frac{x_1}{f}(f - Z) \tag{3.69}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{และ } X_1 + B = \frac{x_2}{f}(f - Z) \quad (3.70)$$

ลบ (3.70) จาก (3.69) และแก้สมการหา Z เราได้ว่า

$$Z = f - \frac{fB}{x_2 - x_1} \quad (3.71)$$

นั่นหมายความว่าถ้าเราทราบผลต่างของพิกัดภาพ x_2 และ x_1 ระยะ Baseline และความยาวโฟกัสของกล้องเราสามารถคำนวณหาพิกัด Z ของ w ได้ สำหรับพิกัด X และ Y สามารถคำนวณโดยใช้พิกัด (x_1, y_1) หรือ (x_2, y_2)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การรับส่งข้อมูล

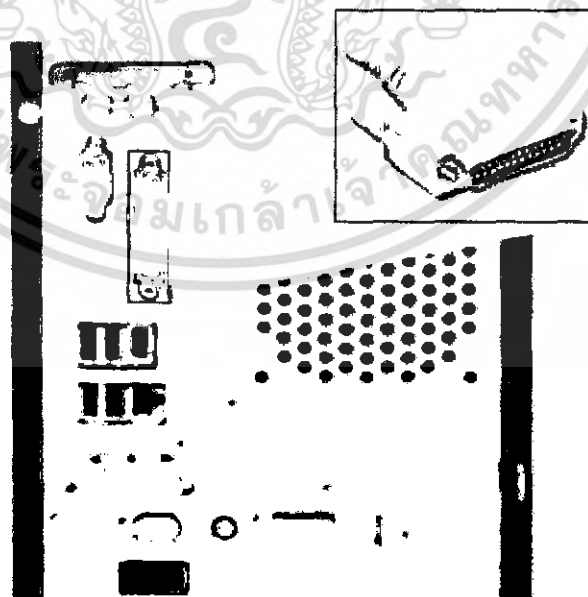
4.1 ทำความเข้าใจกับพอร์ต (Port) คอมพิวเตอร์ที่ใช้งาน

ซึ่งการรับ-ส่งในปฏิญานีฉบับนี้ได้เลือกใช้พอร์ตนานในการรับส่งข้อมูลและพอร์ทยูเอสบีซึ่งใช้ในการติดต่อกับกล้องทั้งสามตัว ซึ่งได้อธิบายถึงพอร์ทที่ต่อใช้งานมาพอสังเขป ดังนี้

4.1.1 พอร์ตขนาน (Parallel Port)

หน้าที่ของพอร์ตตัวนี้ก็คือใช้สำหรับติดต่อกับเครื่องพิมพ์เป็นหลัก ปัจจุบันมีการพัฒนาให้สามารถใช้งานร่วมกับสแกนเนอร์ หรือว่าใครที่ซื้อเครื่องคิดเลขได้ด้วย พอร์ตแบบนี้มีขนาดยาวกว่าพอร์ตอนุกรมทั่ว ๆ ไป โดยมีจำนวนพินเท่ากับ 25 พิน ซึ่งหุ่นยนต์แต่ละฟุตบอลสามมิตินี้ก็ได้อาศัยการส่งผ่านข้อมูลผ่านพอร์ทนี้ด้วยซึ่งสังเกตได้ง่าย

- พอร์ตขนานจะมีรู 25 รู (พอร์ตนี้จะเป็นตัวเมีย หมายถึงมีรูที่ตัวพอร์ต)
- พอร์ตนี้จะต่อกับอุปกรณ์ต่าง ๆ เช่น เครื่องพิมพ์ เทปไดรฟ์ สแกนเนอร์ เป็นต้น
- สามารถต่อความยาวไม่มากนัก แกรมมีราคาแพงกว่าสายของพอร์ตอนุกรมด้วย
- การส่งสัญญาณจะส่งได้เร็วกว่าพอร์ตอนุกรม



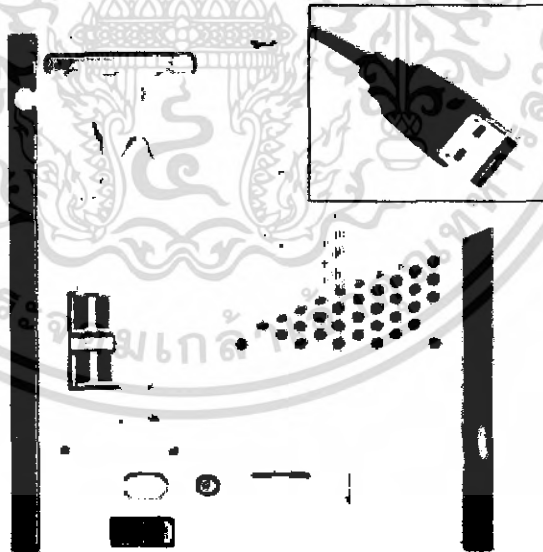
รูป 4.1 พอร์ตขนาน (Parallel Port)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 พอร์ตยูเอสบี (USB Port)

พอร์ตยูเอสบี เป็นพอร์ตแบบใหม่ล่าสุด ที่ถูกออกแบบมาให้ใช้กับพีซีคอมพิวเตอร์ ให้สามารถรับส่งข้อมูลให้รวดเร็วขึ้น สามารถต่ออุปกรณ์ได้มากถึง 127 ชิ้น เพราะมีแบนด์วิดธ์ในการรับส่งข้อมูลสูงกว่า พอร์ตแบบที่ถูกออกแบบมาให้ใช้กับระบบปลั๊กแอนด์เพลย์บนวินโดวส์ 98 และมาจนถึงปัจจุบัน มีฮาร์ดแวร์จำนวนมากที่สนับสนุนการเชื่อมต่อแบบนี้ เช่น กล้องดิจิทัล เม้าส์ คีย์บอร์ด จออสติก สแกนเนอร์ ซีดีอาร์ดับบลิว แฟรทไดรว์ ฮาร์ดดิสก์ไดรว์ เป็นต้น สำหรับคอมพิวเตอร์รุ่นใหม่ ๆ จะมีพอร์ตแบบนี้จะมีพอร์ตแบบนี้อยู่ในเครื่องเรียบร้อยแล้ว

- คอมพิวเตอร์พกพาจะมี 2 USB Port ถ้าเป็นเครื่องรุ่นเก่าที่ไม่มี USB สามารถหาซื้อการ์ด USB มาติดตั้งได้
- เป็นช่องสี่เหลี่ยมเล็ก ๆ สูงประมาณ 3-5 มิลลิเมตร ยาวประมาณ 1-2 เซนติเมตร
- พอร์ตชนิดใหม่นี้รับส่งความเร็ว ได้สูงกว่า port ทั่ว ๆ ไป
- สามารถเชื่อมต่ออุปกรณ์ต่อเนื่องได้ 127 ตัว
- เป็นมาตรฐานใหม่ที่มีมากับเครื่องคอมพิวเตอร์
- การติดตั้ง เพียงต่ออุปกรณ์เข้ากับ USB port ก็สามารถใช้งานอุปกรณ์นั้นๆ ได้ โดยไม่จำเป็นต้อง boot เครื่องใหม่



รูป 4.2 พอร์ตยูเอสบี (USB Port)

4.2 การรับส่งข้อมูลผ่านความถี่วิทยุ

กำลังงานที่แผ่กระจายออกจากสายอากาศไปรอบๆพื้นที่ ซึ่งอยู่ในรูปคลื่นแม่เหล็กไฟฟ้า (Electromagnetic wave) ส่วนประกอบของสัญญาณที่มีผลต่อคลื่นแม่เหล็กไฟฟ้ามี 3 ชนิดคือ พลังงาน (Power) ความถี่ (Frequency) และตำแหน่งของเฟส (Phase position) ข้อมูลข่าวสาร เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

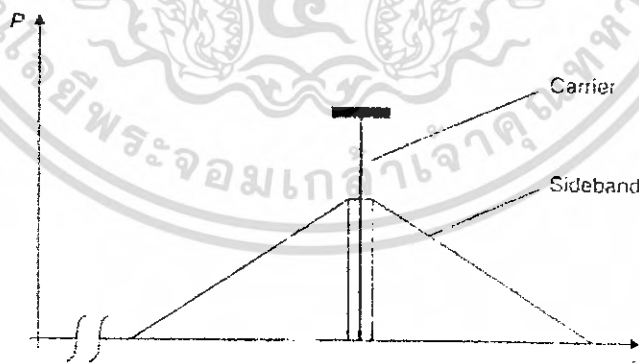
สามารถเข้ารหัสและส่งออกไปยังที่ต่างๆได้ กระบวนการสำคัญในการส่งข้อมูล เรียกว่า การมอดูเลชัน (modulation) และอันมอดูเลชัน (unmodulation) ด้วยคลื่นแม่เหล็กไฟฟ้าที่เรียกว่าคลื่นแควเรียร์ (carrier)

เมื่อเราพิจารณาคคุณสมบัติของคลื่นแม่เหล็กไฟฟ้าที่จุดใดๆ เราจะพบว่า เราจะสามารถเปลี่ยนแปลงคลื่นแม่เหล็กไฟฟ้ากลับมาเป็นข้อมูลข่าวสารได้ ซึ่งอาจจะมาในรูป พลังงาน ความถี่ หรือตำแหน่งของเฟส กระบวนการนี้เรียกว่า ดีมอดูเลชัน(Demodulation)

ในเทคโนโลยีของวิทยุสมัยเก่า ส่วนใหญ่จะเป็นกระบวนการทางอนาล็อก ซึ่งสามารถพบข้อแตกต่างระหว่าง การมอดูเลชันแบบแอมปริจูด (Amplitude modulation) การมอดูเลชันทางความถี่ (Frequency modulation) และการมอดูเลชันทางเฟส (Phase modulation) ทั้งหมดนี้เป็นตัวแปรที่สำคัญในการส่งคลื่นแม่เหล็กไฟฟ้า และในกระบวนการมอดูเลชันแบบดิจิตอลจะแบ่งเป็นกระบวนการมอดูเลชันได้ดังนี้คือ ASK (Amplitude Shift Keying), FSK (Frequency Shift Keying) และ PSK (Phase Shift Keying) ซึ่งในปฏิญานพนธ์นี้ได้เลือกใช้แบบ ASK

4.2.1 การมอดูเลชันแบบASK

ซึ่งเป็นวิธีการที่นำมาใช้ในปฏิญานพนธ์ฉบับนี้ ในการมอดูเลชันแบบASKนี้ ค่าแอมปริจูดของความถี่แควเรียร์จะทำการสวิตช์ระหว่างค่าของ U_0 และ U_1 (Keying) ที่มาจากสัญญาณไบนารี คือ 0 และ 1 ซึ่งค่าของ U_1 สามารถปรับเปลี่ยนค่าได้ระหว่างค่าของ U_0 และ 0 อัตราส่วนของ U_0 และ U_1 เรา จะเรียกว่าค่าคิวตี้แฟคเตอร์ (duty factor) m



รูป 4.3 รูปแสดงมอดูเลชันของสัญญาณ(Sinusoidal) กับ แควเรียร์ ที่ผลิตออกมาเรียกว่า ไซด์แบน (sidebands)

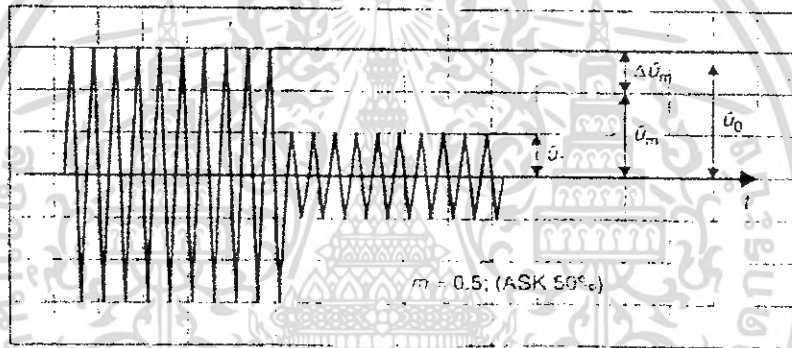
เราสามารถหาค่าคิวตี้แฟคเตอร์ m ได้จากแอมปริจูดที่ได้จากการ Keying กับ Unkeying ของสัญญาณพาห้

$$\hat{U}_m = \frac{\hat{U}_0 + \hat{U}_1}{2} \quad (4.1)$$

ค่าคิวดี้แฟคเตอร์เราจะหาได้จากอัตราส่วนของแอมปริจูดที่เปลี่ยนแปลงของ $\hat{U}_0 - \hat{U}_M$ ค่าของ \hat{U}_M เขียนเป็นสมการได้คือ

$$m = \frac{\Delta \hat{U}_M}{\hat{U}_M} = \frac{\hat{U}_0 - \hat{U}_M}{\hat{U}_M} = \frac{\hat{U}_0 - \hat{U}_1}{\hat{U}_0 + \hat{U}_1} \quad (4.2)$$

ในการมอดูเลชันแบบASK 100% แอมปริจูดของความถี่แคเรียร์ จะทำการสวิตช์ระหว่างขนาดของแอมปริจูดเท่ากับ $2\hat{U}_m$ และ 0 (เหมือนกับการเปิด-ปิดคิงรูป) ในการมอดูเลตทางแอมปริจูดที่ใช้สัญญาณอนาลอก (sinusoidal oscillation) จะมีค่าของการมอดูเลชันแฟคเตอร์ $m=1$ (หรือ 100%)



รูป 4.4 รูปแสดงการมอดูเลต ASK ของสัญญาณแคเรียร์ที่สวิตช์ระหว่างตำแหน่งของสัญญาณในนาบิต

จากการหาค่าคิวดี้แฟคเตอร์ที่ผ่านมานั้น คล้ายกับการหาค่ามอดูเลชันแฟคเตอร์ของการมอดูเลตทางแอมปริจูดที่ใช้สัญญาณอนาลอก อย่างไรก็ตาม ก็ยังมีส่วนหนึ่งที่แตกต่างกันระหว่างแบบสัญญาณดิจิทัลกับสัญญาณอนาลอก ในการมอดูเลชันแบบ ASK สัญญาณแคเรียร์จะอยู่บนแอมปริจูด \hat{U}_0 เป็นส่วนที่เรียกว่าส่วนอันมอดูเลต (Unmodulated) ส่วนในการมอดูเลชันแบบอนาลอก (AM) สัญญาณแคเรียร์ จะอยู่บนแอมปริจูด \hat{U}_m ที่เป็นส่วนของอันมอดูเลต

บางทีค่าคิวดี้แฟคเตอร์จะพูดถึงเปอร์เซ็นต์การลดทอนของสัญญาณแคเรียร์ m'

$$m' = 1 - \frac{\hat{U}_1}{\hat{U}_0} \quad (4.3)$$

จากตัวอย่างในรูป 4.12 ให้ค่าคิวดี้แฟคเตอร์ $m' = 0.66$ (66%) และในกรณี คิวดี้แฟคเตอร์น้อยกว่า 15% และคิวดี้แฟคเตอร์มากกว่า 85% ผลต่างระหว่างค่าทั้งสองเราสามารถหาค่าได้

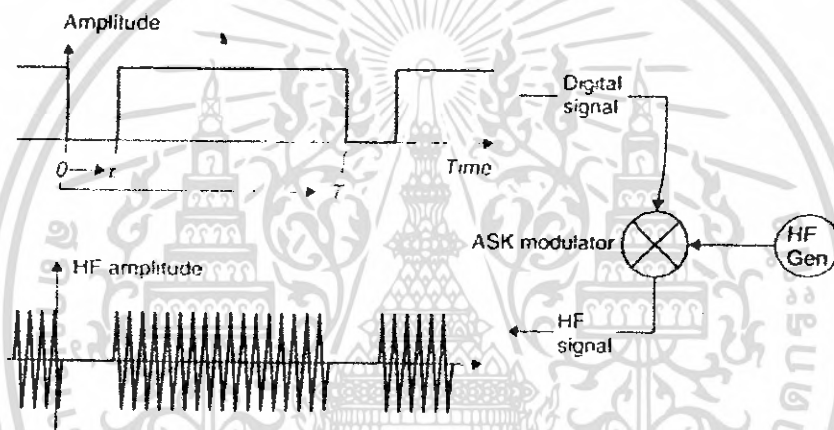
สัญญาณที่เป็นไบนารี จะประกอบด้วยส่วนที่เป็นลอจิก 1 และ 0 ด้วยคาบเวลา T และช่วงเวลาระหว่างบิต τ จากกระบวนการทางคณิตศาสตร์ในการพิจารณาการมอดูเลชันแบบASK ที่สมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะต้องคูณด้วยสัญญาณรหัส $U_{code}(t)$ จากการผลิตความถี่แคเรียร์ $U_{cr}(t)$ สำหรับค่าคงที่แฟลคเตอร์ $m < 1$ เราจะบวกค่าคงที่ $(1-m)$ ฉะนั้นสำหรับกรณีนี้ เราสามารถคูณ $U_{HF}(t)$ ด้วย 1 ในส่วน Unkeyed

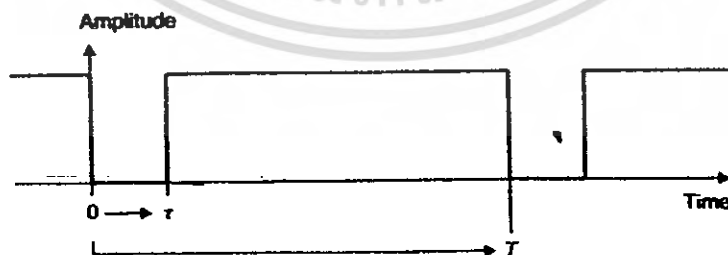
$$U_{ASK}(t) = (m \cdot U_{code}(t) + 1 - m) U_{HF}(t) \quad (4.4)$$

สเปกตรัมของสัญญาณASKจะหาได้จากการคอนโวลูชันของสัญญาณรหัสสเปกตรัมด้วยความถี่แคเรียร์ f_{cr} หรือด้วยผลคูณของการกระจายฟูรีเยร์ ของสัญญาณรหัสด้วยความถี่แคเรียร์ ส่วนประกอบของสเปกตรัมของรหัส code signal ใน upper และ lower sideband ที่สมมาตรกันของแคเรียร์



รูป 4.5 รูปแสดงการมอดูเลตแบบASK 100% ด้วยสัญญาณแคเรียร์ที่สร้างขึ้นมากับสัญญาณที่ keying ด้วยสัญญาณไบนารี

สัญญาณพัลส์สี่เหลี่ยมที่มีคาบเวลา T และช่วงเวลาระหว่างบิต τ จะได้สเปกตรัม ดังตาราง 4.1



รูป 4.6 แสดงคาบเวลา T และช่วงเวลาระหว่างบิต τ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Designation	Frequency	Amplitude
Carrier oscillation	f_{CR}	$u_{HF} \cdot (1-m) \cdot (T-\tau)/T$
1 st spectral line	$f_{CR} \pm 1/T$	$u_{HF} \cdot m \cdot \sin(\pi \cdot \tau / T)$
2 nd spectral line	$f_{CR} \pm 2/T$	$u_{HF} \cdot m \cdot \sin(2\pi \cdot \tau / T)$
3 rd spectral line	$f_{CR} \pm 3/T$	$u_{HF} \cdot m \cdot \sin(3\pi \cdot \tau / T)$
N th spectral line	$f_{CR} \pm n/T$	$u_{HF} \cdot m \cdot \sin(n\pi \cdot \tau / T)$

ตาราง 4.1 Spectral lines สำหรับรูปคลื่นพัลส์ที่มอดูเลตกับความถี่แคเรียร์



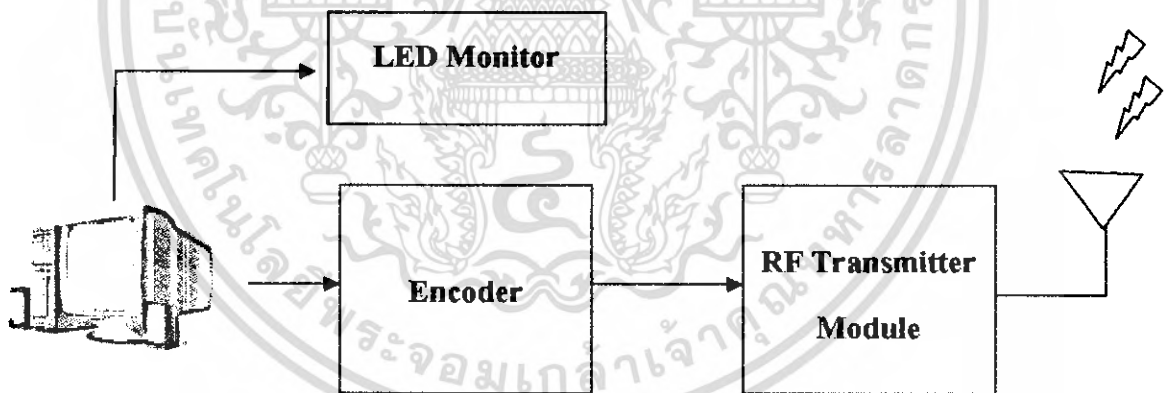
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การออกแบบและการสร้างวงจร

5.1 บล็อกไดอะแกรมและการทำงาน

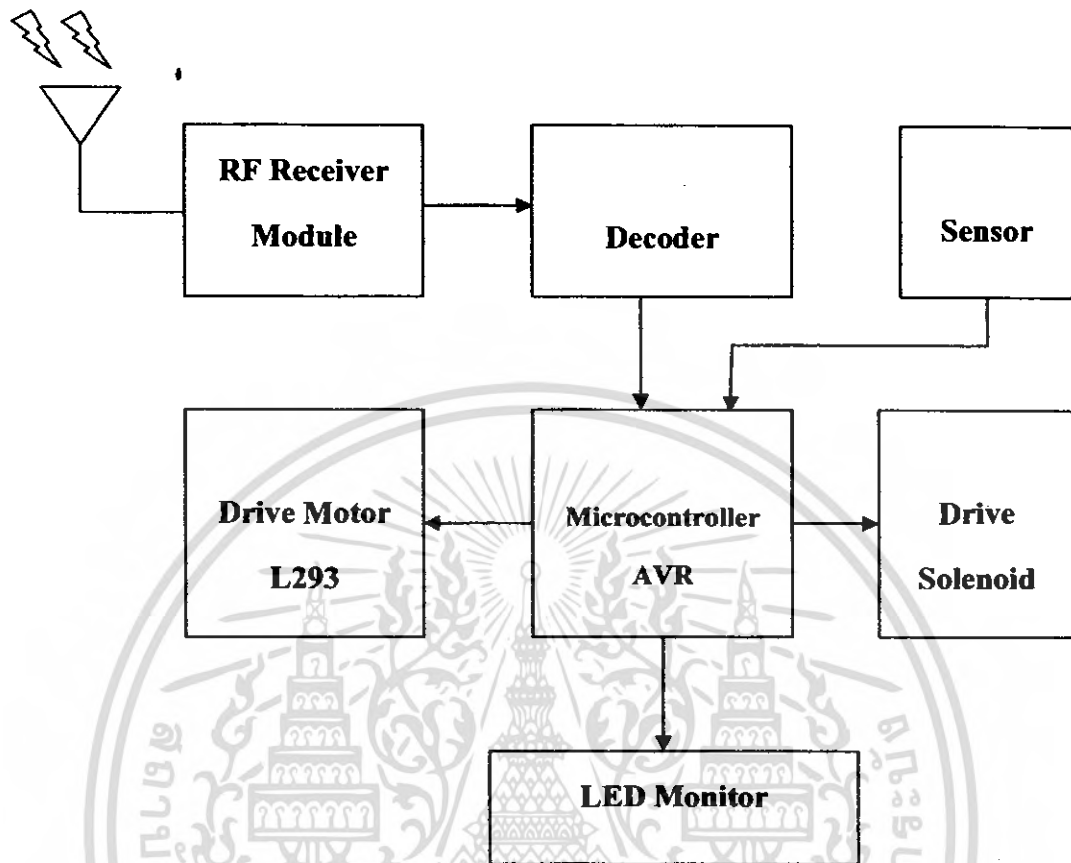
หลังจากที่เราได้ศึกษาเกี่ยวกับทฤษฎีที่เกี่ยวข้องต่างๆ มาพอเข้าใจแล้ว ในส่วนต่อไปเป็นส่วนที่สำคัญที่สุดอย่างหนึ่งคือการออกแบบวงจร ซึ่งในการออกแบบวงจรมันก่อนอื่นเราต้องทราบวัตถุประสงค์ที่เราต้องการเสียก่อน หลังจากนั้นจึงค่อยเริ่มในการสร้างวงจร ซึ่งขั้นตอนในการสร้างวงจรมันเราควรที่จะต้องกำหนดการทำงานต่างๆ อาจเป็นลำดับขั้นตอนในการทำว่าจะเริ่มทำอะไรก่อนส่วนไหนหลัง ซึ่งอาจเขียนเป็นลำดับขั้นตอนการทำ(โฟลว์ชาท)หรืออาจกำหนดเป็นบล็อกไดอะแกรมอย่างคร่าวๆก่อน ซึ่งในการออกแบบวงจรของปริญญาโทฉบับนี้ จะแยกบล็อกไดอะแกรมออกเป็น 2 ส่วนหลักๆ คือ บล็อกไดอะแกรมของภาคส่งดังแสดงในรูปที่ 5.1 และ บล็อกไดอะแกรมของภาครับดังแสดงในรูปที่ 5.2



รูป 5.1 บล็อกไดอะแกรมภาคส่ง(Transmitter)

จากบล็อกไดอะแกรมข้อมูลที่ต้องการจะส่งจะถูกส่งผ่านพอร์ตขนานมาจากคอมพิวเตอร์ซึ่งในส่วนนี้คอมพิวเตอร์จะทำการประมวลผลโดยวิธีอิมเมจโปรเซสซึ่งจากกล้อง และจะทำการกำหนดพิกัดที่ต้องการจะให้หุ่นยนต์ไป เมื่อบอกพิกัดแล้วคอมพิวเตอร์จะส่งข้อมูลออกที่พอร์ตขนาน ซึ่งข้อมูลที่ส่งออกมาจะมีขนาด 4 บิต แล้วนำข้อมูลที่ได้นี้มาเข้าที่วงจรเข้ารหัส (Encoder) ซึ่งเป็นไอซีสำเร็จรูปเบอร์ HT12E ซึ่งมี 18ขา และมีพินสวิทช์ 8 บิต ในการเซตแอดเดรสของไอซีเข้ารหัส ซึ่งต้องเซตให้ตรงกับวงจรถอดรหัสที่ภาครับด้วย เมื่อเข้ารหัสเสร็จแล้ว ข้อมูลจะส่งออกไปเข้าโมดูลของชุดส่งแบบใช้ความถี่วิทยุซึ่งเลือกใช้เบอร์ TLP434 ที่ผลิตโดยบริษัท Laipac Technology ซึ่งใช้การส่งสัญญาณแบบASK ความถี่ 433.92Mhz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป.2 บล็อกโคอะแกรมภาครับ (Receiver)

จากบล็อกโคอะแกรมภาครับเมื่อส่งข้อมูลมาจากชุดส่งทางสายอากาศแล้ว ส่วนของชุดรับก็จะทำการรับข้อมูลเข้ามาทางสายอากาศเพื่อมาเข้าโมดูลของชุดรับสัญญาณแบบASK ซึ่งใช้เบอร์ RLP434A แล้วส่งข้อมูลที่รับได้มาเข้าสู่วงจรถอดรหัสซึ่งเป็น ไอซีสำเร็จรูปเบอร์ HT12D ซึ่งมี18ขา และมีคิฟสวิทซ์ขนาด8บิตในการเซตแอดเดรสซึ่งจะต้องเซตให้ตรงกับคิฟสวิทซ์ของวงจรถอดรหัสที่ภาคส่ง แล้วส่งสัญญาณซึ่งมีขนาด4 บิต มาให้ไมโครคอนโทรลเลอร์เพื่อทำการประมวลผลตามโปรแกรมที่ได้กำหนดไว้ ต่อจากนั้นเมื่อไมโครคอนโทรลเลอร์ประมวลผลตามโปรแกรมที่กำหนดไว้แล้ว ก็จะสั่งให้หุ่นยนต์เคลื่อนที่ไปตามการประมวลผลซึ่งมีการตรวจสอบกับเซ็นเซอร์แสงที่เปลี่ยนแปลงเมื่อหุ่นยนต์เคลื่อนที่ซึ่งจะได้ออกมาเป็นสัญญาณพัลส์สี่เหลี่ยมมาทำการประมวลผลเพื่อวิเคราะห์ทางที่หุ่นยนต์เคลื่อนที่ซึ่งโปรแกรมให้นับจำนวนพัลส์ที่เข้ามาแล้วคำนวณเป็นระยะทางเพื่อให้ความสัมพันธ์กันในการเคลื่อนที่ หรือสั่งให้เตะฟุตบอลโดยใช้คลวดโซลินอยด์ในการเตะ และมีการแสดงผลของการทำงานด้วย โดยแสดงผลผ่านหลอดแอลอีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

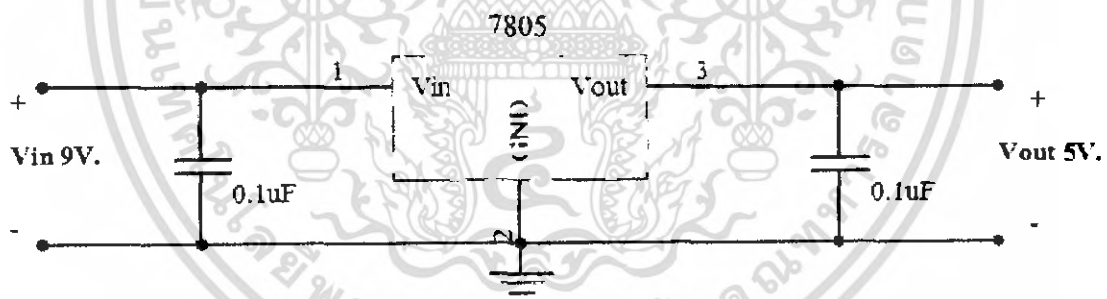
5.2 หลักการออกแบบและการสร้างวงจร

การออกแบบแบ่งเป็นส่วนใหญ่ว่าได้ดังนี้

- แหล่งจ่ายไฟ (Power Supply)
- ชุดส่งข้อมูลแบบไร้ความถี่วิทยุ (Wireless Transmitter)
- ชุดรับข้อมูลแบบไร้ความถี่วิทยุ (Wireless Receiver)
- ไมโครคอนโทรลเลอร์ (Microcontroller)
- วงจรขับมอเตอร์ (Drive Motor)
- ชุดต่อในการลงโปรแกรมบนตัวไมโครคอนโทรลเลอร์ (In-System Programming)

5.2.1 แหล่งจ่ายไฟ (Power Supply)

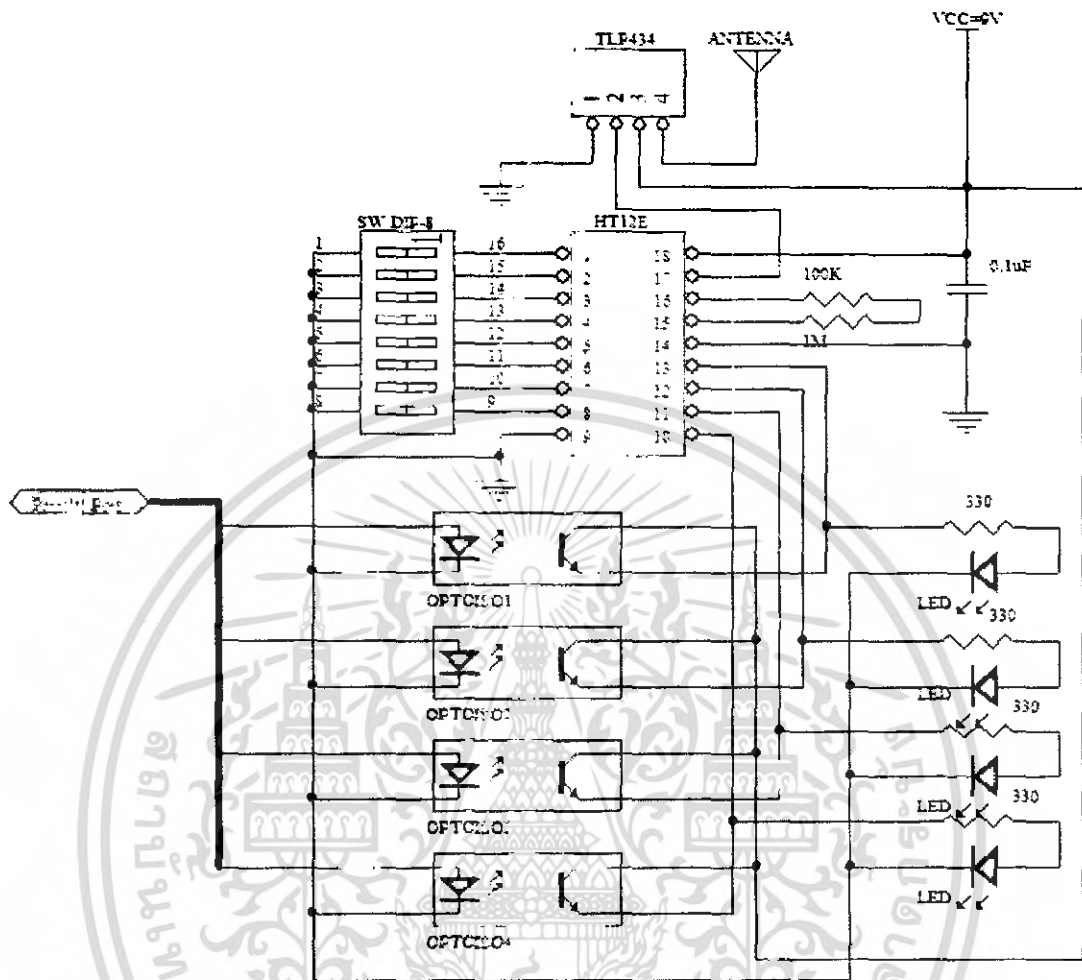
แหล่งจ่ายไฟที่จ่ายให้กับชุดส่งสัญญาณจะใช้เป็นแบตเตอรี่ขนาด 1.5 โวลต์จำนวน 6 ก้อน บรรจุอยู่ในรางถ่าน ก็จะได้แรงดันไฟฟ้ามีค่าเท่ากับ 9 โวลต์จ่ายให้กับวงจรชุดส่ง ส่วนทางด้าน วงจรชุดรับจะใช้แบตเตอรี่ที่จ่ายให้กับวงจรจำนวน 2 ชุด ชุดแรกใช้จ่ายให้กับวงจรภาครับสัญญาณ ความถี่วิทยุซึ่งใช้แรงดันไฟฟ้าขนาด 9 โวลต์ ส่วนอีกชุดหนึ่งจ่ายให้กับวงจรขับมอเตอร์กับขดลวด โซลินอยด์และผ่านวงจรแปลงระดับแรงดันไฟฟ้าให้เหลือแรงดันไฟฟ้าเท่ากับ 5 โวลต์เพื่อใช้จ่าย ให้กับตัวไมโครคอนโทรลเลอร์ ซึ่งวงจรจ่ายไฟที่จ่ายให้กับไมโครคอนโทรลเลอร์จะเป็นดังรูปที่ 5.3



รูป 5.3 วงจรจ่ายไฟของไมโครคอนโทรลเลอร์

5.2.2 ชุดส่งข้อมูลแบบไร้ความถี่วิทยุ (Wireless Transmitter)

วงจรที่ใช้ส่งสัญญาณนี้มีความสามารถในการทำงานโดยใช้แหล่งจ่ายไฟฟ้ากระแสตรงที่มีค่า ตั้งแต่ 2 โวลต์จนถึง 12 โวลต์ สามารถทนกระแสได้สูงสุดที่ 19.4 มิลลิแอมป์ อิมพีแดนซ์ที่เอาต์พุต มีค่าเท่ากับ 50 โอห์ม และในปริยายานิพนธ์นี้ได้เลือกใช้แหล่งจ่ายไฟของวงจรเท่ากับ 9 โวลต์ ซึ่ง วงจรจะแสดงดังรูปที่ 5.4



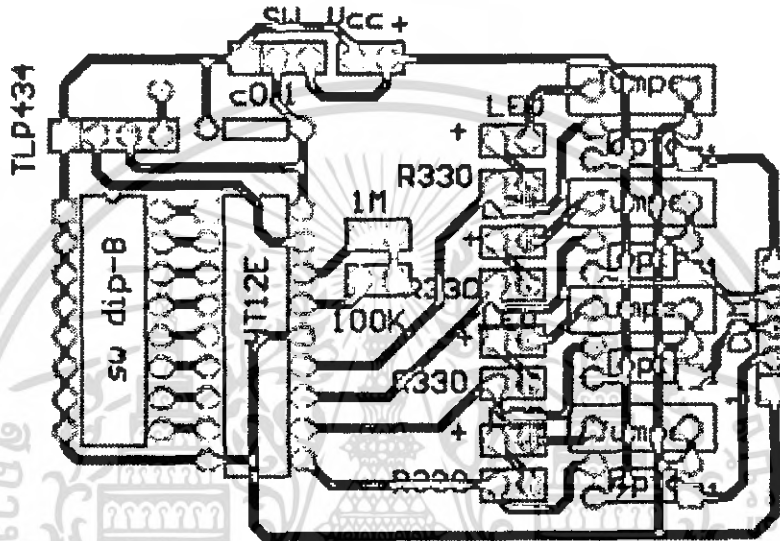
รูป 5.4 วงจรภาคส่ง(Wireless Transmitter)

จากวงจรจะเห็นได้ว่าจะมีการต่อ Opto Isolator ระหว่างพอร์ตขนาบกับ HT12E เนื่องจากกระแสไฟฟ้าที่ออกจากพอร์ตขนาบมีปริมาณน้อยและต้องมีการแสดงผลที่หลอดแอลอีดีอีก ทำให้ต้องใช้กระแสไฟฟ้ามากขึ้นอีกทั้งยังเป็นการป้องกันพอร์ตขนาบไม่ให้เสียหายไปในตัวด้วย ซึ่งการต่อจะทำการต่อเป็นแบบคอมมอนคอลเล็กเตอร์

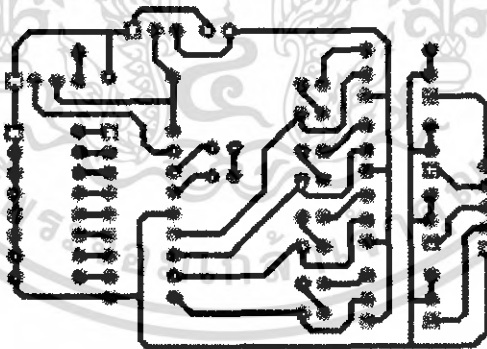
ในการออกแบบวงจรจะใช้โปรแกรม Proteel 99 ในการออกแบบ โดยวงจรที่ออกแบบเสร็จแล้วจะแสดงดังรูปที่ 5.5 ส่วนในรูปที่ 5.6 จะเป็นลายแม่แบบของ ลายทองแดงที่จะนำไปสร้างเป็นแผ่นวงจรซึ่งในการสร้างเราจะสั่งพิมพ์ออกมาที่เครื่องพิมพ์เอกสาร แล้วนำเอกสารที่ได้นี้ไปถ่ายเอกสาร โดยใช้แผ่นใส จากนั้นนำแผ่นใสด้านที่มีหมึกติดอยู่มาวางแผ่นทองแดงที่เราตัดให้ได้ขนาดและทำการล้างด้วยทินเนอร์ให้สะอาดและแห้งแล้ว จากนั้นใช้เตารีดที่ตั้งอุณหภูมิให้ร้อนระดับปานกลางมารีดบนแผ่นใส โดยห้ามให้แผ่นใสเลื้อนหรือขยับจะทำให้ลายวงจรเสียหายได้ เมื่อร้อนดีแล้ว นำน้ำแข็งมาลูบบนแผ่นใสที่รีดแล้วให้ทั่วบริเวณลายวงจร แล้วรีบรอกแผ่นใสออกมาก็จะได้แผ่นทองแดงที่มีหมึกติดอยู่เป็นลวดลายวงจร อาจจะตกแต่งด้วยปากกาเขียนแผ่นปริ้นเพื่อให้ลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรมีความคมมากขึ้น จากนั้นนำแผ่นทองแดงไปทำการกัดลายโดยใช้ยากัดลาย เมื่อกัดเสร็จแล้วก็จะได้ลายทองแดงตามที่เราต้องการ นำไปล้างทำความสะอาดด้วยน้ำแล้วเช็ดด้วยทินเนอร์ จากนั้นก็ทำการเจาะรู ซึ่งวิธีการข้างต้นในการสร้างลายวงจรอาจใช้วิธีอื่นในการทำก็ได้แล้วแต่จะสะดวก เมื่อเจาะรูเสร็จแล้วก็ทำการลงอุปกรณ์ก็จะได้วงจรที่พร้อมใช้งานดังรูปที่ 5.7

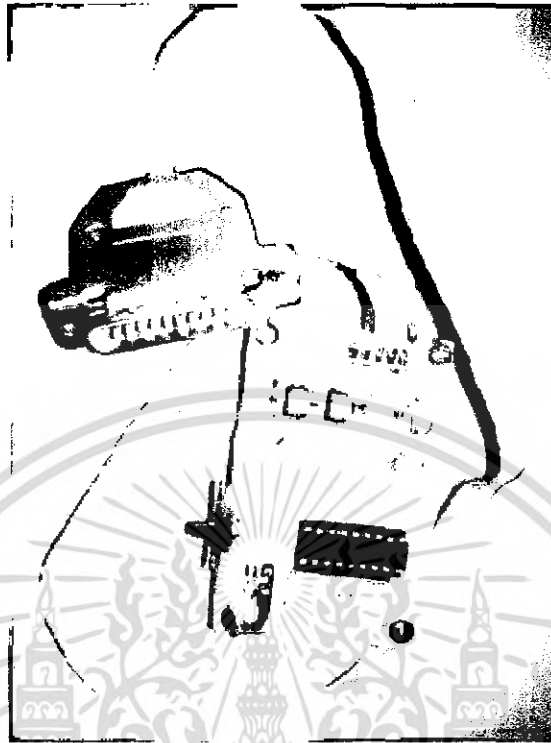


รูป 5.5 วงจรส่งสัญญาณที่ออกแบบโดยโปรแกรม Protel 99



รูป 5.6 ลายทองแดงของวงจรส่งสัญญาณ

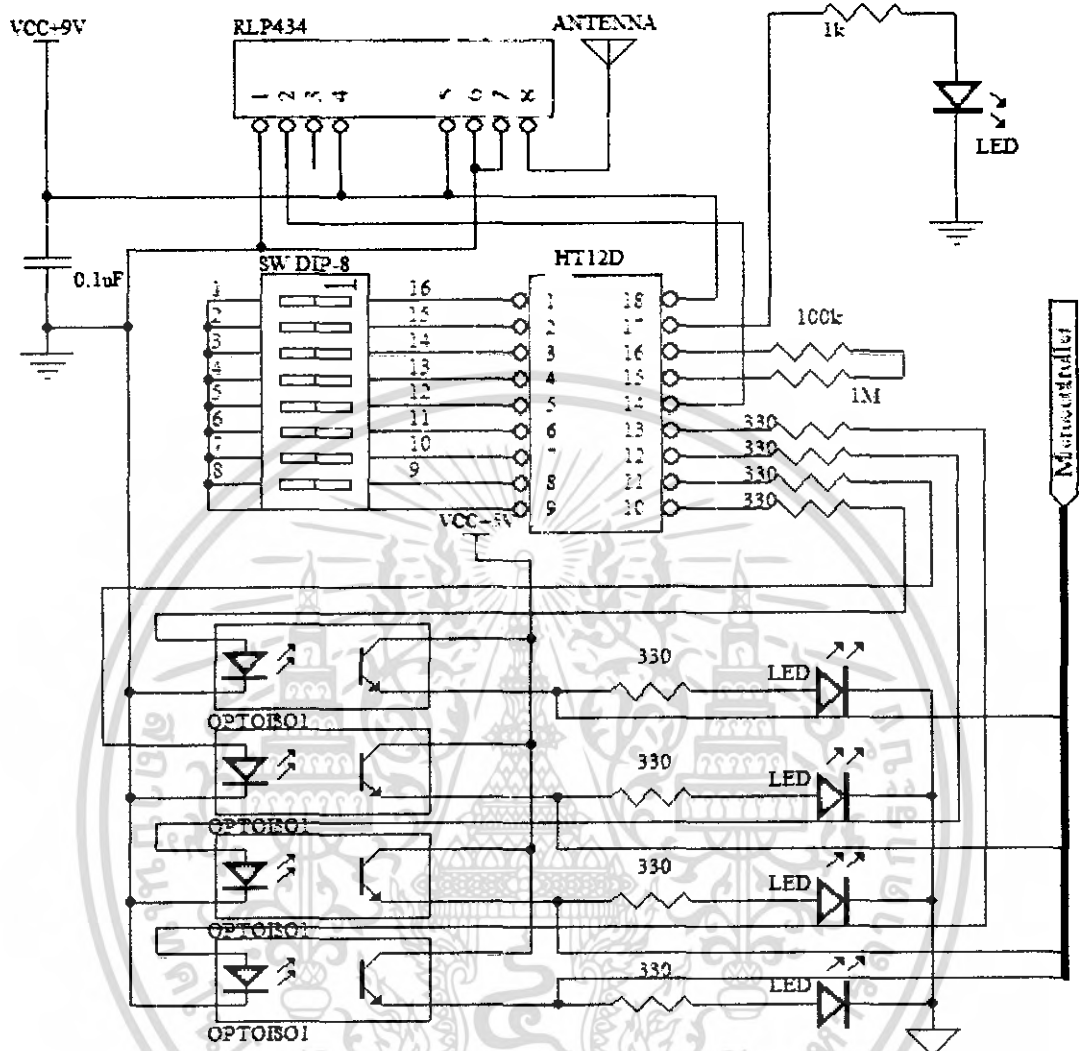
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 5.7 วงจรส่งสัญญาณที่ลงอุปกรณ์เสร็จแล้ว

5.2.3 ชุดรับข้อมูลแบบใช้ความถี่วิทยุ (Wireless Receiver)

เมื่อชุดส่งทำการส่งสัญญาณออกอากาศแล้ว ส่วนของชุดรับก็จะทำการรับข้อมูลเข้ามาที่สายอากาศมาเข้าโมดูลของชุดรับแบบ ASK เบอร์ RLP434A แล้วส่งข้อมูลที่รับได้มาเข้าสู่วงจรถอดรหัส ซึ่งเป็นไอซีสำเร็จรูปเบอร์ HT12D ซึ่งมี 18 ขา และมีคิฟสวิทช์ขนาด 8 บิตในการเซทแอดเดรสซึ่งจะต้องเซทให้ตรงกับคิฟสวิทช์ของวงจรเข้ารหัสที่ภาคส่ง ซึ่งค่าความต้านทานที่ขา 15 และขา 16 ค่า 100k และ 1M จะเป็นตัวกำหนดความถี่ของไอซีเข้ารหัสและที่ไอซีถอดรหัสก็จะมีควมต้านทานค่าเดียวกันนี้คืออยู่ที่ขา 15 และ 16 เหมือนกัน ที่ขา 17 ของไอซีจะต่อค่าความต้านทานเพื่อจำกัดกระแสที่จะเข้าไปที่หลอด LDE เพื่อแสดงผลในการรับ-ส่งข้อมูลว่ารับสัญญาณข้อมูลได้หรือไม่ ซึ่งถ้ารับสัญญาณได้หลอดแอลอีดีจะสว่างแต่ถ้ารับสัญญาณไม่ได้หลอดแอลอีดีก็จะดับ และเมื่อรับสัญญาณและทำการถอดรหัสแล้วข้อมูลที่ได้อีกก็จะถูกส่งออกที่ขา 10, 11, 12 และ 13 ซึ่งมีขนาด 4 บิตไปเข้าที่ Opto Isolator เพื่อแยกกราวด์ก่อนที่จะส่งข้อมูลนี้ไปเข้าที่ขาอินพุทของไมโครคอนโทรลเลอร์ และได้มีการแสดงผลในการส่งข้อมูลโดยใช้หลอดแอลอีดีจำนวน 4 หลอด ซึ่งวงจรจะเป็นดังรูปที่ 5.8



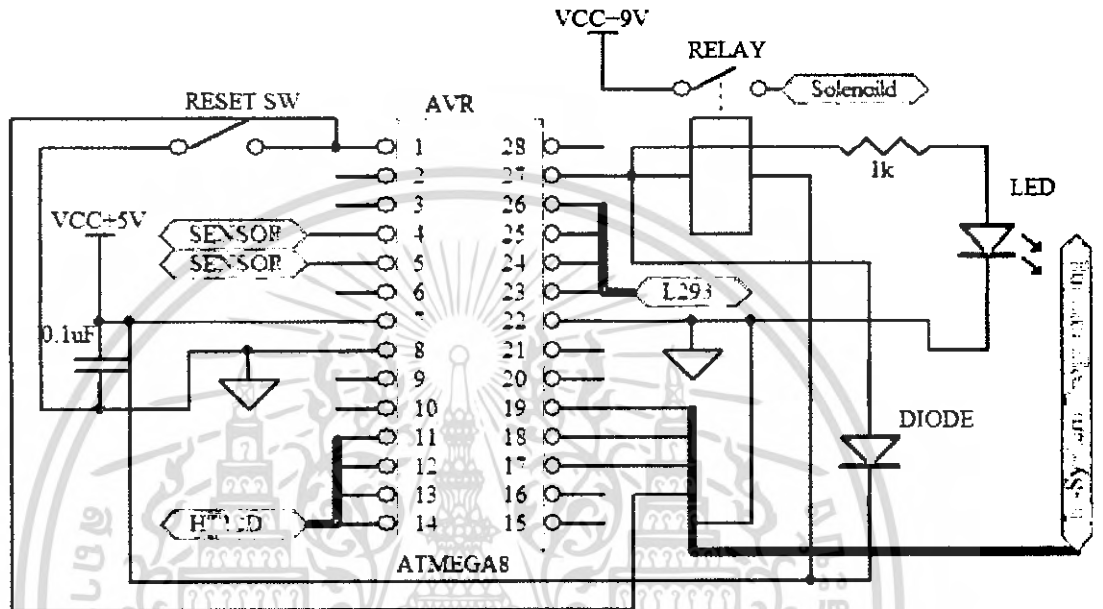
รูป 5.8 วงจรภาครับ(Wireless Receiver)

5.2.4 ไมโครคอนโทรลเลอร์ (Microcontroller)

ไมโครคอนโทรลเลอร์ที่เลือกใช้มีชื่อว่า AVR เมอร์ ATmega8-16PI เป็นของบริษัท ATMAL ซึ่งเป็นไอซีขนาด 28 ขาซึ่งจะเป็นตัวรับข้อมูลจากวงจรถอดรหัสที่มีขนาด 4 บิตมาทำการประมวลผล ตามโปรแกรมที่กำหนดไว้ และที่พอร์ทอื่นๆของไมโครคอนโทรลเลอร์ก็จะแบ่งเป็นส่วนที่รับข้อมูลจากเซ็นเซอร์ที่ถือของหุ่นยนต์ซึ่งใช้อินฟาเรดในการเซ็นเซอร์แสงที่เปลี่ยนแปลงเมื่อหุ่นยนต์เคลื่อนที่ซึ่งจะได้ออกมาเป็นสัญญาณพัลส์ที่ถี่หรือช้าซึ่งเข้าที่พอร์ท PD2(ขา4) และพอร์ท PD3(ขา5) ซึ่งคือของถ้อยด้านซ้ายและด้านขวา มาทำการประมวลผลเพื่อวัดระยะทางที่หุ่นยนต์เคลื่อนที่ซึ่งโปรแกรมให้นับจำนวนพัลส์ที่เข้ามาแล้วคำนวณเป็นระยะทาง พอร์ทPC4(ขา27) จะไว้ใช้สำหรับส่งรีเลย์ ให้ควบคุมขดลวดโซลินอยส์ในการชิงลูกบอล พอร์ทPC6(ขา11) เป็นขาเรเซทการทำงานของตัวไมโครคอนโทรลเลอร์โดยจะต่อกับสวิทช์ลงกราวด์ ส่วนพอร์ทPC0(ขา23), พอร์ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PC1(ขา24), พอร์ตPC2(ขา25) และพอร์ตPC3(ขา26) จะเป็นพอร์ตที่จะส่งข้อมูลออกไปให้วงจรขับมอเตอร์ และใช้ พอร์ตPB5(ขา19),พอร์ตPB4(ขา18),พอร์ตPB3(ขา17),พอร์ตPC61(ขา1)และกราวด์เพื่อไว้สำหรับลงโปรแกรมให้กับตัวไมโครคอนโทรลเลอร์ ซึ่งจะต่อกับพอร์ตขานานของคอมพิวเตอร์ ซึ่งวงจรจะเป็นดังรูปที่ 5.9

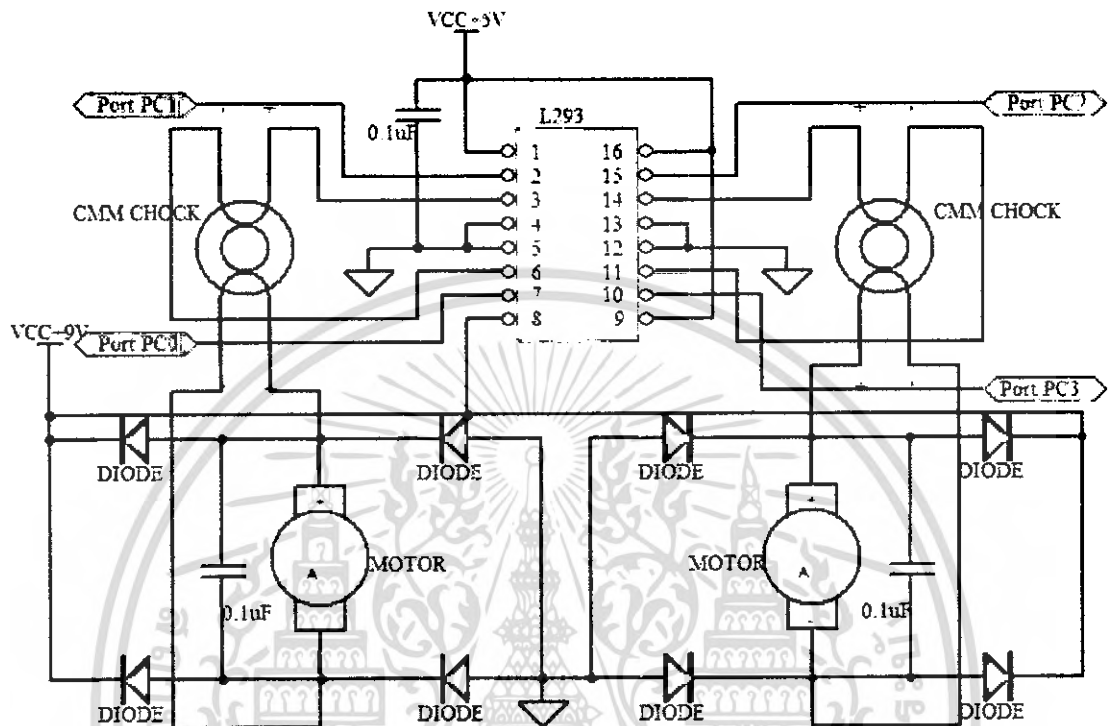


รูป 5.9 วงจร ไมโครคอนโทรลเลอร์(Microcontroller)

5.2.5 วงจรขับมอเตอร์ (Drive Motor)

หลังจากที่ไมโครคอนโทรลเลอร์ทำการประมวลผลแล้วก็จะส่งให้วงจรขับมอเตอร์ทำการขับมอเตอร์ให้หุ่นยนต์เคลื่อนที่เดินทาง กอขหลัง เลี้ยวซ้าย เลี้ยวขวา ตามโปรแกรมที่ต้องการซึ่งได้ใช้ไอซีสำเร็จรูปเป็นตัวทำหน้าที่ในการขับมอเตอร์ซึ่งใช้เบอร์ L293D ซึ่งเป็นวงจรขับมอเตอร์แบบ Push-Pull Four Channel Driver ที่สามารถขับกระแสได้ในช่วง 600 มิลลิแอมแปร์ถึง 1.2 แอมแปร์ต่อแชนแนล จ่ายแรงดันไฟฟ้าที่ใช้เลี้ยงวงจร(VCC) ได้ถึง36โวลต์ ซึ่งส่วนนี้จะเป็นตัวที่เราจะกำหนดว่าจะให้จ่ายแรงดันให้กับมอเตอร์เท่าใดและสามารถจ่ายแรงดันไฟฟ้าของวงจรลจิกได้ในช่วง4.5-36โวลต์ และยังสามารถกำหนดให้วงจรทำงานที่แรงดันต่ำ(Logic Low)หรือแรงดันสูง(Logic High)ได้ ซึ่งสามารถต่อมอเตอร์กระแสตรงที่สามารถกลับทางหมุนได้สูงสุด2ตัวใช้การบังคับที่เป็นอิสระต่อกัน ซึ่งจะมีไดโอดต่ออยู่ที่มอเตอร์ข้างละ4ตัว เพื่อป้องกันค่ากระแสที่เกิดจากสนามแม่เหล็กไหลกลับซึ่งเกิดจากขดลวดในตัวมอเตอร์จ่ายออกมาคอนทูดมอเตอร์หยุดหมุน ซึ่งอาจจะไปทำให้วงจรที่อยู่ภายในไอซีเสียหายได้ ดังรูปที่5.10 และจากรูปจะเห็นว่ามิชขดลวดที่พันอยู่บนแกนเฟอร์ไรต์ แบบโค่นัท จำนวน 2 ตัวต่อก่อนเข้ามอเตอร์ ซึ่งจะเรียกว่าคอมมอนโหมดไอซี(Common mode

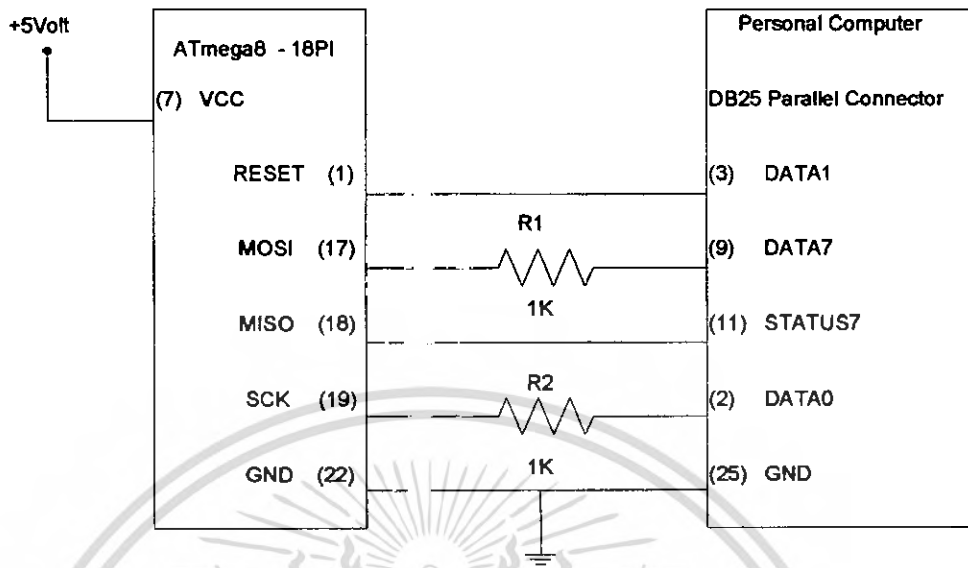
Chock) และมีตัวเก็บประจุต่อค่อมที่ขั้วของมอเตอร์ทั้งสองเพื่อป้องกันสัญญาณรบกวนที่เกิดจากมอเตอร์ มารบกวนสัญญาณในการรับ-ส่งข้อมูล หรือสัญญาณต่างๆของวงจร



รูป 5.10 วงจรขับมอเตอร์(Drive Motor)

5.2.6 จุดต่อในการลงโปรแกรมบนตัวไมโครคอนโทรลเลอร์ (In-System Programming)

ไมโครคอนโทรลเลอร์เบอร์ ATmega8-16PI นี้มีโครงสร้างของ ISP (In-System Programming) ที่ทำให้เราทำการเขียนโปรแกรมลงบนตัวไอซีได้โดยผ่านทางสายต่อระหว่างไมโครคอนโทรลเลอร์กับเครื่องคอมพิวเตอร์ที่เราใช้เขียนโปรแกรม โดยที่ไอซีไมโครคอนโทรลเลอร์ของเรายังคงต่ออยู่กับวงจรที่อาจจะมีแผงแสดงผล เช่น แอลอีดี สวิตช์ ฯลฯ อยู่ โดยที่ไม่ต้องถอดตัวไมโครคอนโทรลเลอร์ ออกจากวงจรเหล่านั้นเสียก่อน ทำให้ลดเวลาในการทดลองลงได้มาก เพราะไม่ต้องถอดไมโครคอนโทรลเลอร์ เข้าออกจาก Zip-socket เหมือนสมัยก่อน ซึ่งสิ่งที่จะต้องใช้ในการนี้ก็มี หัวต่อพอร์ตขนาน, สายไฟ, โปรแกรม SPI2 และงานบัดกรีเล็กน้อย เท่านั้นก็สามารถทำการโปรแกรมไมโครคอนโทรลเลอร์ได้แล้ว ซึ่งวงจรจะเป็นดังรูปที่ 5.11 และหัวต่อดังรูปที่ 5.12

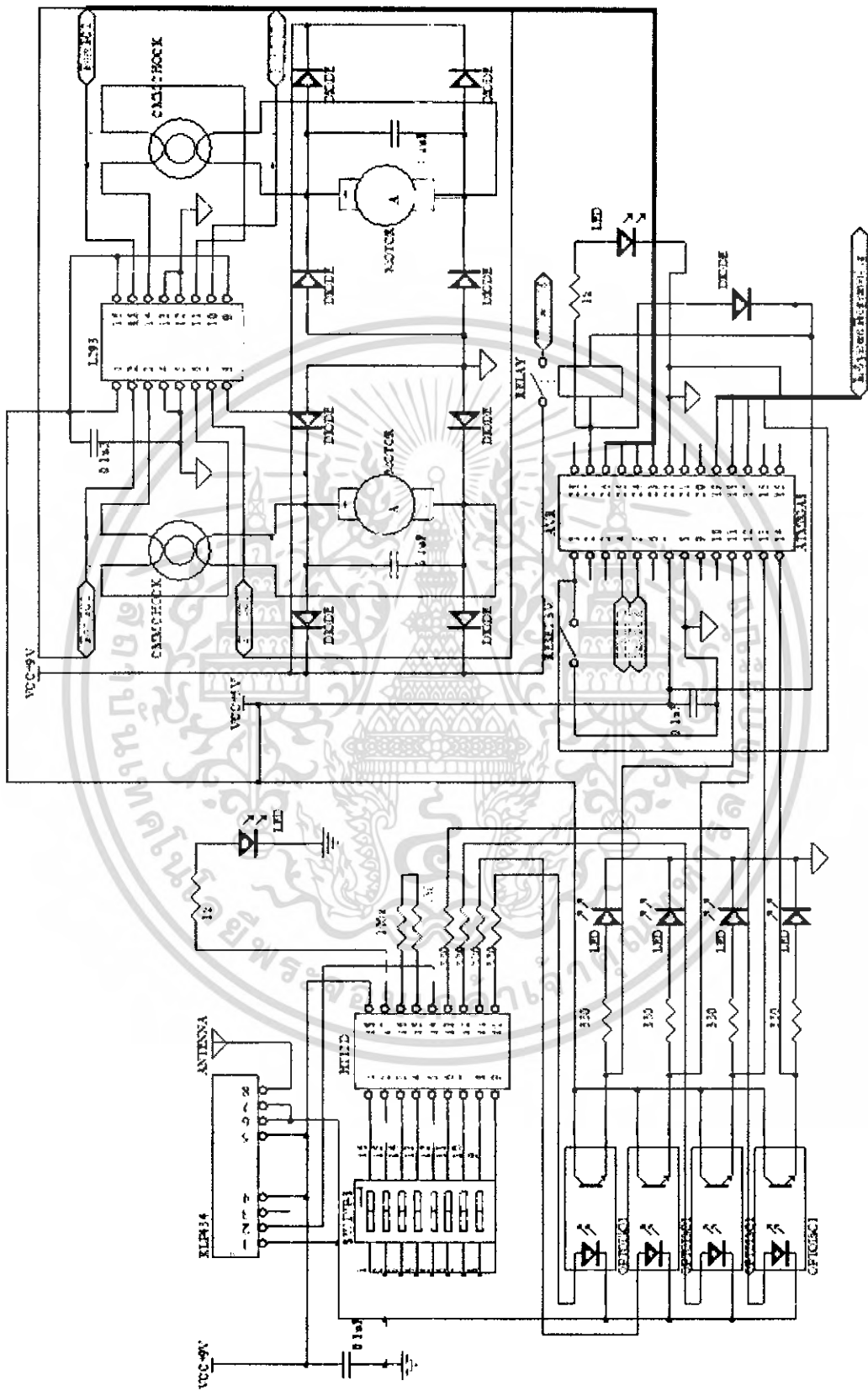


รูป 5.11 วงจร ISP (In-System Programming)



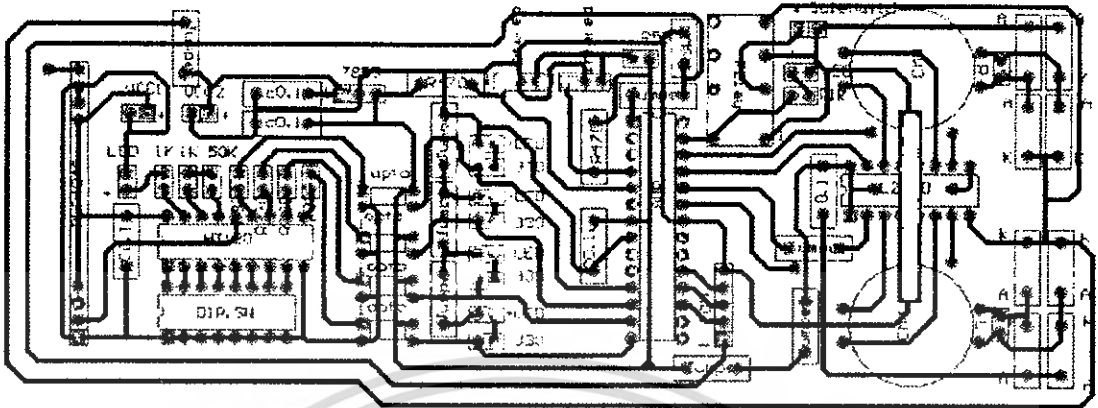
รูป 5.12 ตำแหน่งขาของหัวต่อ DB25

จากที่กล่าวมาทั้งหมดจะเห็นว่าในส่วนของวงจรภาครับ จะแบ่งเป็นส่วนย่อยๆของวงจรทั้งหมด ซึ่งจะต้องนำส่วนต่างๆมาประกอบเป็นวงจรเดียวกันแล้ว เมื่อรวมกันเป็นวงจรเดียวกันแล้ว จึงค่อยมาถึงขั้นตอนการออกแบบลายวงจรที่จะนำไปเป็นแบบในการสร้างลายทองแดงของวงจร ซึ่งจะได้แสดงดังรูปที่ 5.13 และรูปลายทองแดงของวงจรดังรูปที่ 5.14 เมื่อได้ลายทองแดงแล้วก็จะนำไปเจาะรู แล้วลงอุปกรณ์เมื่อลงอุปกรณ์เสร็จแล้วจึงนำไปประกอบกับตัวหุ่นยนต์ ดังรูปที่ 5.16 จะเป็นรูปของตัวหุ่นที่ทำการประกอบวงจรแล้ว จากนั้นก็ทำการทดลองทางด้านโปรแกรมเพื่อให้หุ่นยนต์เคลื่อนที่ไปในที่ต่างๆ ที่เราต้องการได้ ซึ่งขั้นตอนการเขียนโปรแกรมจะได้กล่าวในส่วนต่อไป

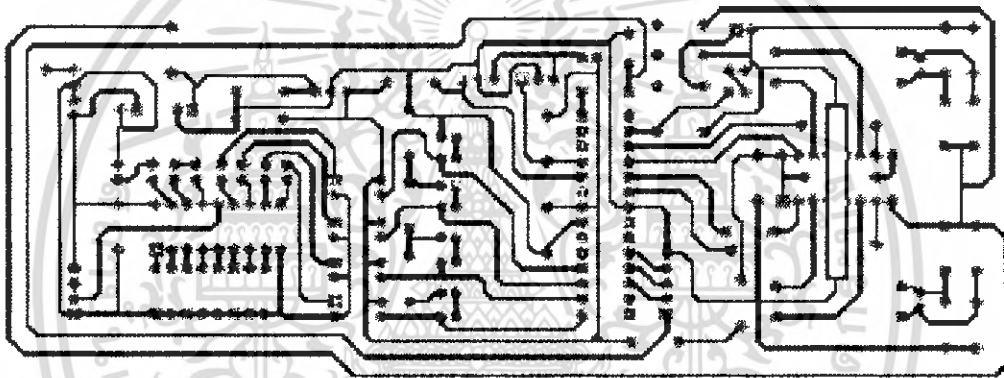


รูป 5.13 วงจรของตัวนับหน่วย

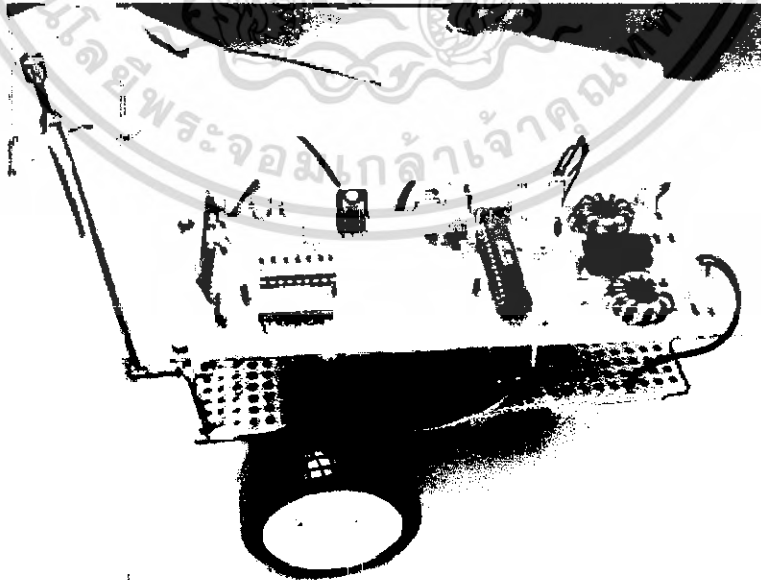
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 5.14 วงจรที่ออกแบบโดยโปรแกรม Protel 99



รูป 5.15 ลายทองแดงของวงจรที่ตัวหุ่นยนต์



รูป 5.16 รูปตัวหุ่นที่ประกอบวงจรแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากที่ได้กล่าวมาทั้งหมดในบทนี้ก็จะกล่าวถึงเกี่ยวกับการออกแบบและการสร้างวงจร ที่ใช้ในการประดิษฐ์หุ่นยนต์เตะฟุตบอล 3 มิติ ซึ่งในขั้นต่อไปก็คือการเขียนโปรแกรมในส่วนของไมโครคอนโทรลเลอร์ กับในส่วนของ การเขียนโปรแกรมเพื่อประมวลผลภาพแบบ 3 มิติ และจะต้องตกแต่งหุ่นยนต์ ดิจิตอลหรือสีเพื่อใช้ในการจับภาพของหุ่นยนต์ ซึ่งจะได้กล่าว ในส่วนต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การทดลองและสรุปผล

6.1 การควบคุมหุ่นยนต์

เป็นการทดลองการบังคับให้หุ่นยนต์เคลื่อนที่ไปในตำแหน่งและทิศทางที่เราต้องการ ซึ่งที่ตัวหุ่นเราได้ทำการโปรแกรมลงบนตัวไมโครคอนโทรลเลอร์ โดยเราได้กำหนดค่าในการเคลื่อนที่ดังตารางที่ 6.1 จะแสดงข้อมูลที่ส่งงานออกมาผ่านพอร์ตขนาน เพื่อมาเข้าที่ขาอินพุตของไมโครคอนโทรลเลอร์และในตารางที่ 6.2 จะเป็นข้อมูลที่เราเป็นคนกำหนดให้ไมโครคอนโทรลเลอร์ ส่งออกที่พอร์ตเอาต์พุตเพื่อสั่งให้ L293 ทำการขับเคลื่อนให้หุ่นยนต์เคลื่อนที่ และเตะลูกบอลโดยใช้ขดลวดโซลินอยด์ในการเตะ โดย "0" และ "1" แทนลอจิกในการทำงาน

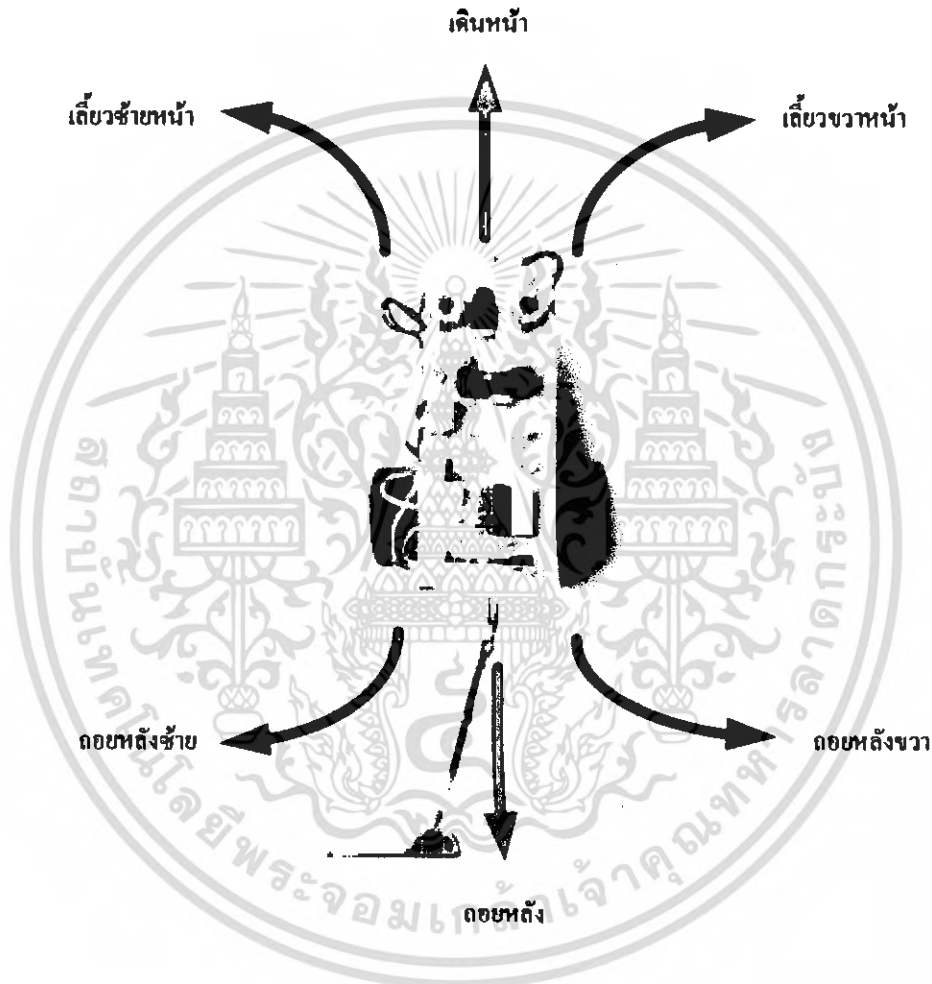
PORTD5	PORTD6	PORTD7	PORTB0	คำสั่ง
0	0	0	0	หยุด
0	0	0	1	เดินหน้า
0	0	1	0	ถอยหลัง
0	0	1	1	เลี้ยวขวาหน้า
0	1	0	0	เลี้ยวซ้ายหน้า
0	1	0	1	ถอยหลังซ้าย
0	1	1	0	ถอยหลังขวา
1	0	0	0	เตะฟุตบอล

ตาราง 6.1 แสดงคำสั่งที่ไมโครคอนโทรลเลอร์รับเข้ามา

PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	คำสั่ง
0	0	0	0	0	หยุด
0	1	0	0	1	เดินหน้า
0	0	1	1	0	ถอยหลัง
0	1	0	0	0	เลี้ยวซ้ายหน้า
0	0	0	0	1	เลี้ยวขวาหน้า

0	0	1	0	0	ถอยหลังซ้าย
0	0	0	1	0	ถอยหลังขวา
1	0	0	0	0	เตะลูกบอล

ตาราง 6.2 คำสั่งของไมโครคอนโทรลเลอร์ที่สั่งออกเอาต์พุต



รูป 6.1 แสดงลักษณะการเคลื่อนที่ของหุ่นยนต์

การทดลองที่ 6.1 มุมเลี้ยวของหุ่นยนต์

-การทดลองนี้จะเป็นการเขียนโปรแกรม เพื่อสั่งงานให้หุ่นยนต์ เลี้ยวซ้าย-เลี้ยวขวาด้วยมุมต่างๆ โดยให้ค่าผิดพลาดน้อยที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมอ้างอิง	มุมเฉลี่ยของหุ่นยนต์	%ความผิดพลาด
45°	43°	4.4%
90°	88°	2.2%
180°	185°	2.7%
360°	366°	1.6%

ตาราง 6.3 แสดงค่าความผิดพลาดของมุมเฉลี่ยต่างๆ

การทดลองที่ 6.2 ระยะทางการเคลื่อนที่ของหุ่นยนต์

-การทดลองนี้จะเป็นการเขียน โปรแกรม เพื่อสั่งงานให้หุ่นยนต์ เคลื่อนที่เป็นเส้นตรงด้วย ระยะทางต่างดังนี้

ระยะอ้างอิง(Cm.)	ระยะทางที่หุ่นยนต์เคลื่อนที่ได้(Cm.)	%ความผิดพลาด
10	10.4	4%
20	20.7	3.5%
30	31.3	4.3%
50	51.5	3%

ตาราง 6.4 แสดงค่าความผิดพลาดของระยะทางค่าต่าง

การทดลองที่ 6.3 การเคลื่อนที่ไปยังพิกัดที่กำหนด

-การทดลองนี้จะเป็นการสั่งงานหุ่นยนต์ให้เคลื่อนที่ไปยังพิกัด x,y ที่กำหนดให้ดังนี้

พิกัดอ้างอิง		พิกัดที่หุ่นยนต์เคลื่อนที่ไป		%ความผิดพลาด	
พิกัด x(Cm.)	พิกัด y(Cm.)	พิกัด x(Cm.)	พิกัด y(Cm.)	พิกัด x	พิกัด y
50	50	56	52	12%	4%
50	30	55	31.5	10%	5%
50	20	54	22	8%	10%
50	10	53	11	6%	10%
30	50	32	51.5	6.67%	3%
20	50	21	52	5%	4%
10	50	11	52	10%	4%

ตาราง 6.5 แสดงค่าความผิดพลาดของระยะพิกัด x, y ต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

จากการทดลอง ค่าผิดพลาดเกิดจากเครื่องมือวัดที่ยังไม่ค่อยมีความแม่นยำเท่าที่ควรเพราะใช้เครื่องมือเท่าที่หาได้มาประยุกต์ใช้งาน และวิธีการวัดอาจจะยังไม่ได้มาตรฐาน ค่าระยะความผิดพลาดที่หาคิดต่างๆ มีค่าค่อนข้างสูงเพราะเกิดจากผลของความผิดพลาดที่มุลเกี่ยวกับระยะทางการเคลื่อนที่ รวมกัน ซึ่งแนวทางแก้ไขต่อไปคือ จะเพิ่มส่วนป้อนกลับของระบบทั้งหมดโดยจะใช้กล้องสเตอริโอเป็นส่วนป้อนกลับเพื่อความถูกต้องแม่นยำเพิ่มขึ้น

6.2 การปรับเทียบกล้อง(Camera Calibration)

เนื่องจากระบวนการในการประมวลผลภาพจะต้องทำการปรับเทียบกล้องซึ่งการทดลองต่อไปนี้เป็น การทดลองเทียบกับทฤษฎีที่เป็นโปรแกรม MATLAB ส่วนการทดลองจะใช้โปรแกรม DELPHI

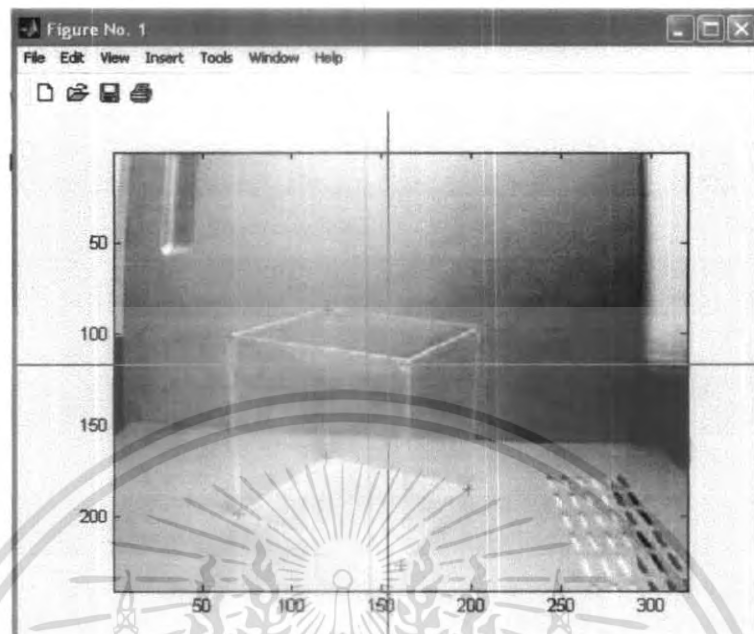
การทดลองที่ 6.4 การปรับเทียบโดยใช้โปรแกรม Matlab

-การทดลองนี้แสดงการปรับเทียบกล้อง โดยจะใช้กล่องพลาสติกใส ขนาด กว้าง×ยาว×สูง = 10 cm. × 10 cm. × 10 cm. เป็นวัตถุอ้างอิง



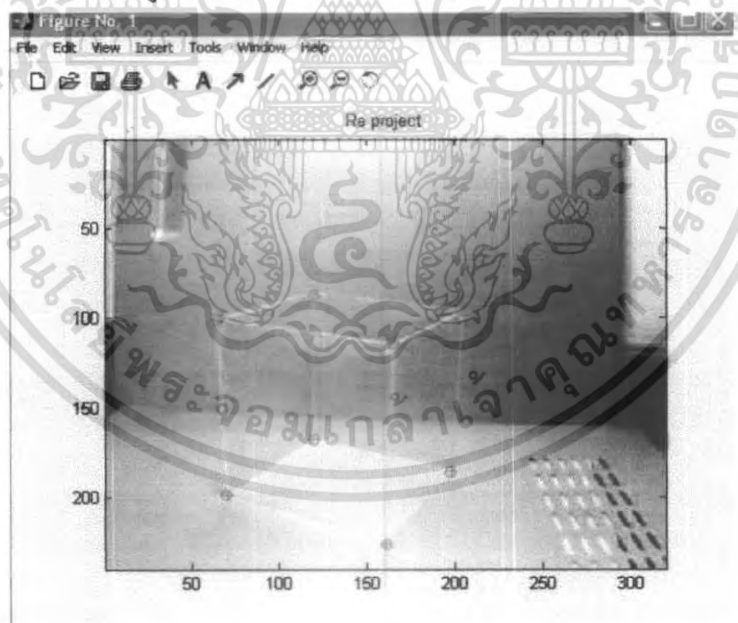
รูป 6.2 แสดงวัตถุอ้างอิง

ทำการรัน โปรแกรม MATLAB ด้วย Source Code ที่เขียนไว้แล้ว หลังจากนั้นโปรแกรมจะปรากฏหน้าต่าง Figure ออกมาดังรูป



รูป 6.3 แสดงหน้าต่าง ของโปรแกรมMATLAB

-ทำการคลิกที่มุมแต่ละมุมของกล่องในภาพ ให้ครบทั้ง 8จุด เมื่อครบแล้วโปรแกรมจะทำการประมวลผล และแสดงคังรูป



รูป 6.4 แสดงผลลัพธ์ จากโปรแกรม MATLAB

โดยสัญลักษณ์ “+” คือจุดที่เราทำการเลือกจากภาพตัวอย่าง “o” คือจุดที่โปรแกรมคำนวณได้และนำมาพล็อตลงบนภาพตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

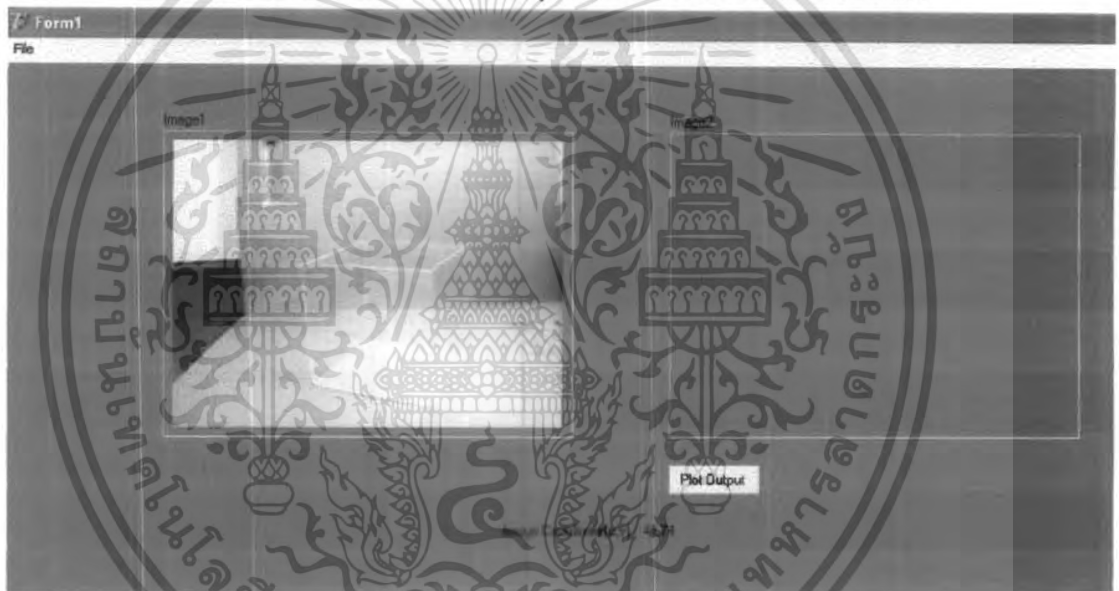
ผลการทดลอง ค่าของเมตริกซ์ T, R และ K ที่ได้จากโปรแกรม MATLAB

T = -26.6078	R = -0.6667 0.7454 -0.0008
16.8013	-0.7454 -0.6667 -0.0008
21.3005	-0.0011 0.0000 1.0000

K = 0.0541	0	0
0	0.0465	0
0	0	0.0001

การทดลองที่ 6.5 การปรับเทียบโดยใช้โปรแกรม DELPHI

-การทดลองนี้จะทำการปรับเทียบกล้องโดยใช้วัตถุอ้างอิงเดียวกันกับ การทดลองที่แล้ว



รูป 6.5 แสดงหน้าต่างของโปรแกรม

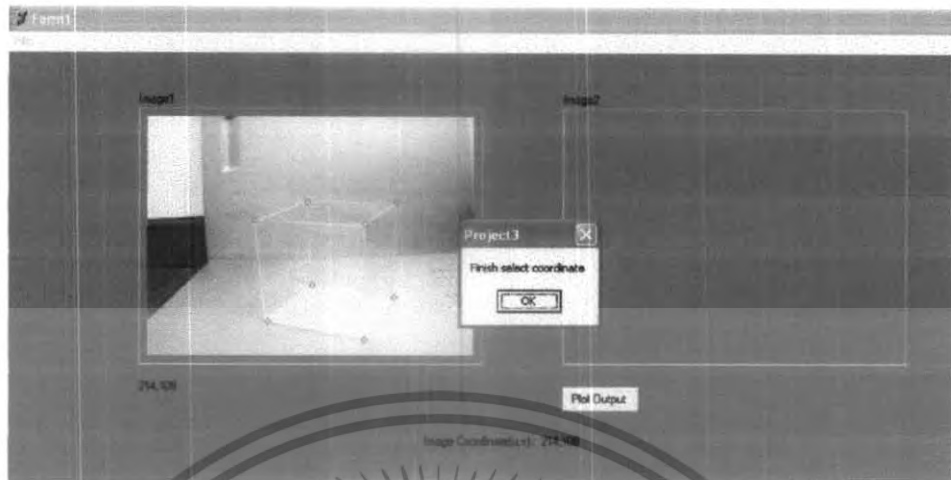
โดยส่วนต่างๆของโปรแกรมมีการทำงานดังนี้

- File : สำหรับทำการเปิดไฟล์ภาพที่จะใช้ในการปรับเทียบกล้อง
- Image1 : เป็นหน้าต่างสำหรับแสดงภาพที่จะนำมาทำการปรับเทียบกล้อง
- Image1 : เป็นหน้าต่างสำหรับแสดงภาพผลลัพธ์ที่ได้หลังจากทำการปรับเทียบกล้องแล้ว
- Plot Output : เป็นปุ่มสำหรับสั่งงานให้โปรแกรมทำการประมวลผล และแสดงภาพเอ้าท์พุท

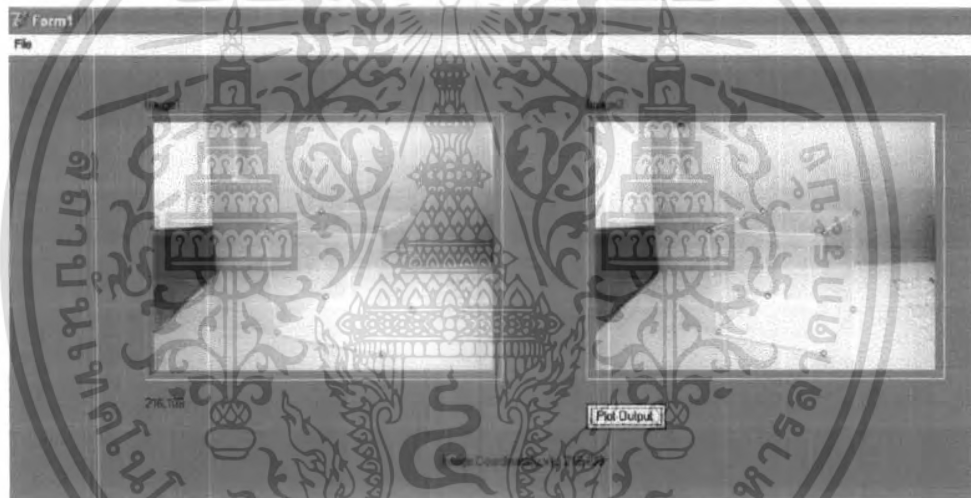
Image Coordinate (u,v):แสดงพิกัดของภาพ(u,v)ที่เมาส์ชี้อยู่

-การทดลองเริ่ม โดยการเปิด ไฟล์ภาพขึ้นมา จากนั้นก็คลิกทำมุมของกล้องให้ครบทั้ง 8 จุด เมื่อเลือกได้ครบทั้ง 8 จุดแล้วจะมีข้อความแสดงว่า "Finish select coordinate" จากนั้นก็คลิก "OK"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 6.6 แสดงการทำงานของโปรแกรมในส่วนของการเลือกพิกัดจากภาพตัวอย่าง
-หลังจากนั้นก็ทำการคลิกที่ "Plot Output" ผลลัพธ์ที่ได้จะเป็นดังรูป



รูป 6.7 แสดงภาพผลลัพธ์ที่ได้หลังจากผ่านการประมวลผลแล้ว

ผลการทดลอง ค่าของเมตริกซ์ T, R และ K ที่ได้จาก โปรแกรมDELPHI

$$T = -25.1248$$

$$R = \begin{bmatrix} -0.7886 & 0.6149 & -0.0011 \\ 0.6149 & -0.7886 & -0.0009 \\ -0.0014 & -0.0000 & 1.0000 \end{bmatrix}$$

$$15.9095$$

$$-0.6149 \quad -0.7886 \quad -0.0009$$

$$18.0847$$

$$-0.0014 \quad -0.0000 \quad 1.0000$$

$$K = \begin{bmatrix} 0.0519 & 0 & 0 \\ 0 & 0.0438 & 0 \\ 0 & 0 & 0.0001 \end{bmatrix}$$

$$0 \quad 0.0438 \quad 0$$

$$0 \quad 0 \quad 0.0001$$

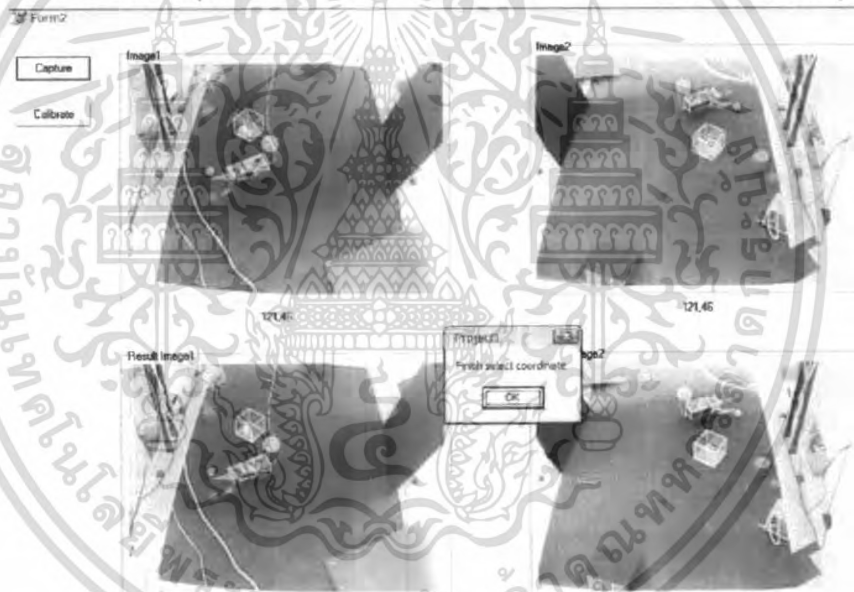
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลอง

จากการทดลองพบว่าค่าพารามิเตอร์ต่างของกล้อง คือ เมตริกซ์ T, R และ K จากทั้งสองโปรแกรมเปรียบเทียบกันถือว่ามีความใกล้เคียงกัน แต่ก็ยังมีความผิดพลาดอยู่บ้าง ซึ่งค่าที่ผิดพลาดอาจเกิดจากผู้ที่ทำทดลองเพราะในการคลิกเลือกจุดที่มุมของกล่อง จากการที่ใช้งานถึง 2 โปรแกรม อาจจะทำให้จุดที่เลือกทั้ง 2 โปรแกรมอาจจะไม่ใช่จุดๆเดียวกันจึงทำให้ค่าที่ได้ผิดพลาดไป

การทดลองที่ 6.6 การปรับเทียบกล้อง 2 ตัว(Camera Calibration with 2 Cameras)

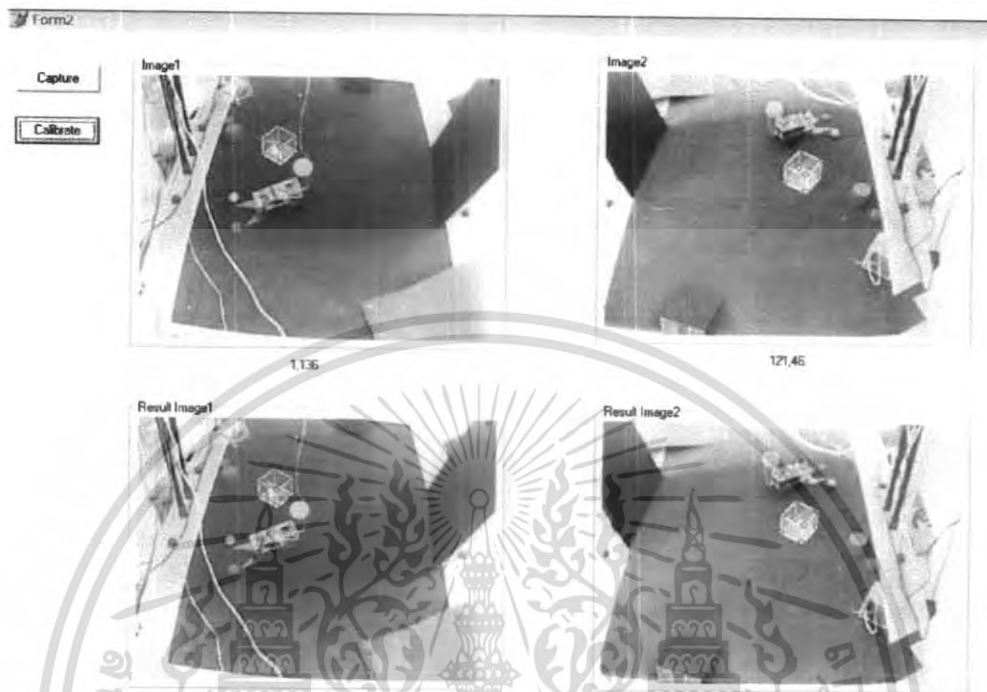
- ซึ่งเราจะทำการปรับเทียบกล้องทั้ง 2 ตัวโดยการเปิด โปรแกรมขึ้นมาแล้วทำการคลิกเลือกที่มุมทั้ง 8 มุมของกล่องสี่เหลี่ยม โดยจะต้องเริ่มคลิกตำแหน่งเดียวกัน โดยเรียงลำดับเหมือนกันด้วย
- เมื่อคลิกครบทั้ง 8 จุดแล้วก็จะขึ้นคำว่า Finish select coordinate ซึ่งมีหน้าต่างดังรูปที่ 6.8



รูป 6.8 หน้าต่างโปรแกรมที่ทำการปรับเทียบกล้อง 2 ตัว

- หลังจากนั้นก็คลิกที่ปุ่ม OK เพื่อให้โปรแกรมได้ทำการปรับเทียบกล้อง ซึ่งจะต้องเกิดจุดที่มุมของรูปกล่องที่ภาพทั้ง 2 ด้านล่างดังรูปที่ 6.9

ผลการทดลอง เกิดจุดที่มัมทั้ง 8 ของกล่อง



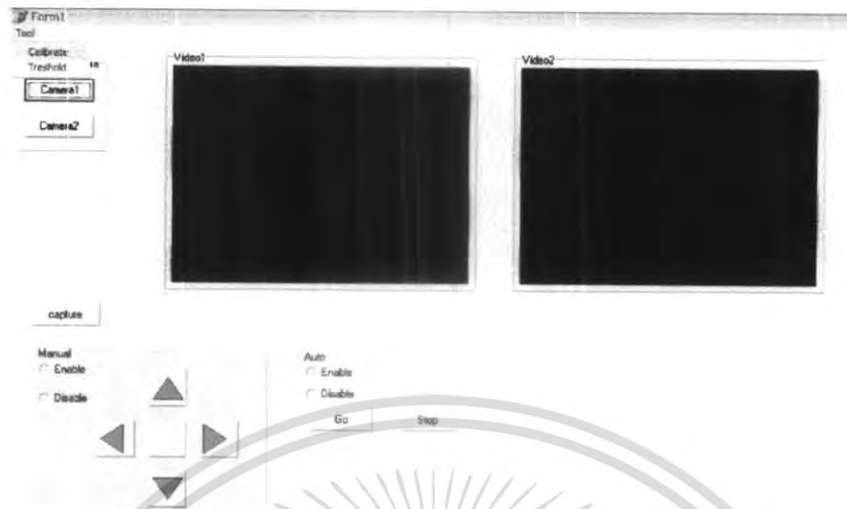
รูป 6.9 ภาพแสดงหน้าต่างการปรับเทียบกล้องที่สมบูรณ์แล้ว

สรุปผลการทดลอง

จากการทดลองจะเห็นว่ารูปผลลัพธ์ที่ได้ยังมีความคลาดเคลื่อนจากความเป็นจริงเล็กน้อยโดยค่าความผิดพลาดเกิดขึ้นจากความโค้งของเลนส์ และระยะวัตถุที่ใช้อ้างอิงห่างจากกล้องมากเกินไป นอกจากนี้ การคำนวณของคอมพิวเตอร์ค่าที่ได้จะเป็นค่าที่เกิดจากการประมาณ ไม่ใช่ค่าที่แท้จริง ซึ่งค่าที่แท้จริงนั้นจะเป็นค่าทศนิยม ส่วนค่าที่คอมพิวเตอร์นำมาคำนวณจะเป็นค่าที่เป็นเลขจำนวนเต็ม จึงทำให้เกิดความคลาดเคลื่อน

การทดลองที่ 6.7 การควบคุมหุ่นยนต์ด้วยมือ (Manual Control)

- เป็นการสั่งการ โดยตรงเพื่อให้หุ่นยนต์เคลื่อนที่ตามที่เราต้องการซึ่งมีหน้าต่างโปรแกรม ดังรูปที่ 6.10
- ซึ่งเมื่อนำหน้าต่างแสดงขึ้นมาแล้ว ให้ทำการคลิกเลือกที่ปุ่ม Enable เพื่อใช้เลือกโหมดให้สามารถควบคุมด้วยมือได้



รูป 6.10 หน้าต่างโปรแกรมที่ใช้ส่งงานหุ่นยนต์
ผลการทดลอง สามารถบังคับหุ่นยนต์เพื่อทำคำสั่งต่างๆ ตามที่เราต้องการได้



รูป 6.11 หน้าต่างโปรแกรมที่ส่งงานหุ่นยนต์ด้วยมือ

สรุปผลการทดลอง

จากการทดลองปรากฏว่าหุ่นยนต์สามารถเคลื่อนที่ไปยังตำแหน่งต่างๆที่เราต้องการได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

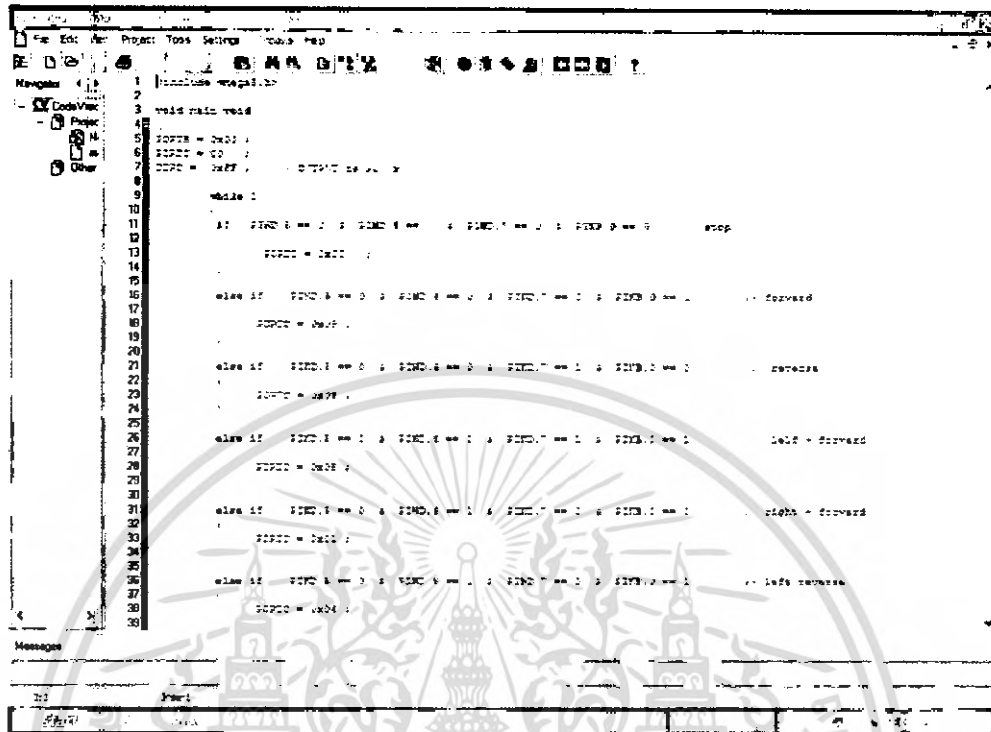
1. กำธร เรือนผ่ายกาส “my AVR”, Digital Signal Processing Laboratory, มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี, 207 หน้า, 2544
2. อ.ชินภัทร นันทจิวงกรชัย “Microprocessor and Programming”, คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2549
3. รศ.ดร. ชูชาติ ปิณฑวิรุจน์ “การประมวลผลภาพเชิงเลข”, คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 248 หน้า, 2549
4. ทีมงาน อีทีที “คู่มือ AVR Enhanced RISC Microcontrollers”, บริษัท อีทีที จำกัด, 96 หน้า
5. บัณฑิต จามรภูติ “คู่มือการใช้งาน Potel 99”, บัณฑิตเพรส, 279 หน้า, 2544
6. วรพงษ์ ภัทรแก้ววัฒนกุล, ชัชวัฒน์ ถิ่นพรจิตรวิไล “เรียนรู้และปฏิบัติการ ไมโครคอนโทรลเลอร์ MCS-51 แบบเฟิร์ม”, บริษัท อินโนเวทีฟ เอ็กเพอริเมนต์ จำกัด, ฉบับ AT89C5x ของ Atmel, 399 หน้า, 2541
7. Klaus Finkenzeller “RFID Handbook Fundamentals and Application in Contactless Smart Cards and Identification”, Translated by Rachel Waddington, John Wiley & Sons

ภาคผนวก

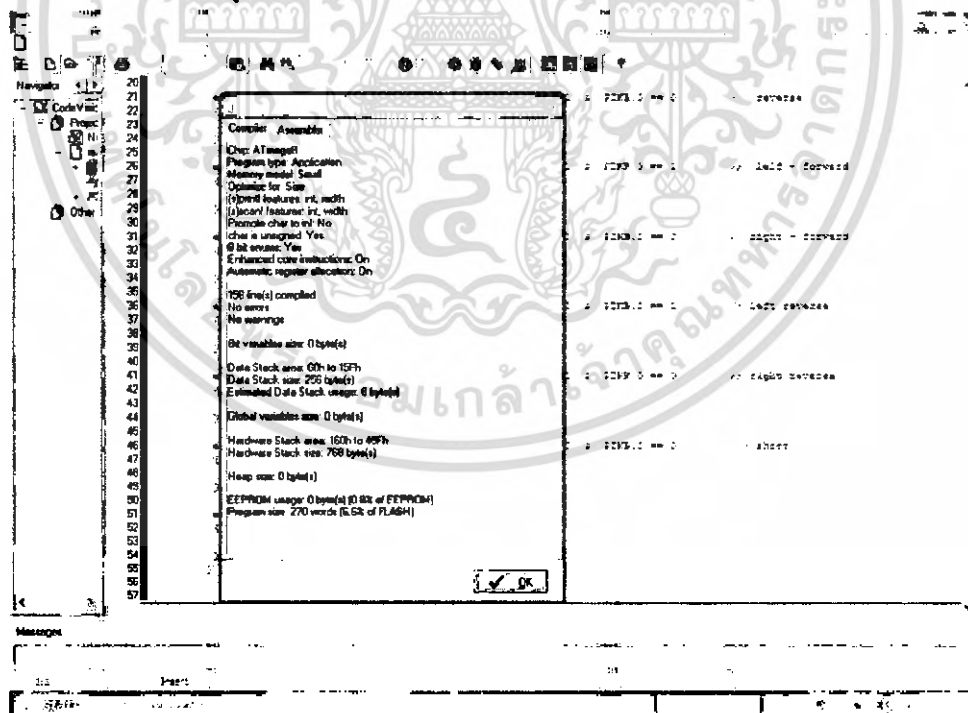


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแสดงหน้าต่างโปรแกรมที่ใช้ในการทดลองหุ่นยนต์

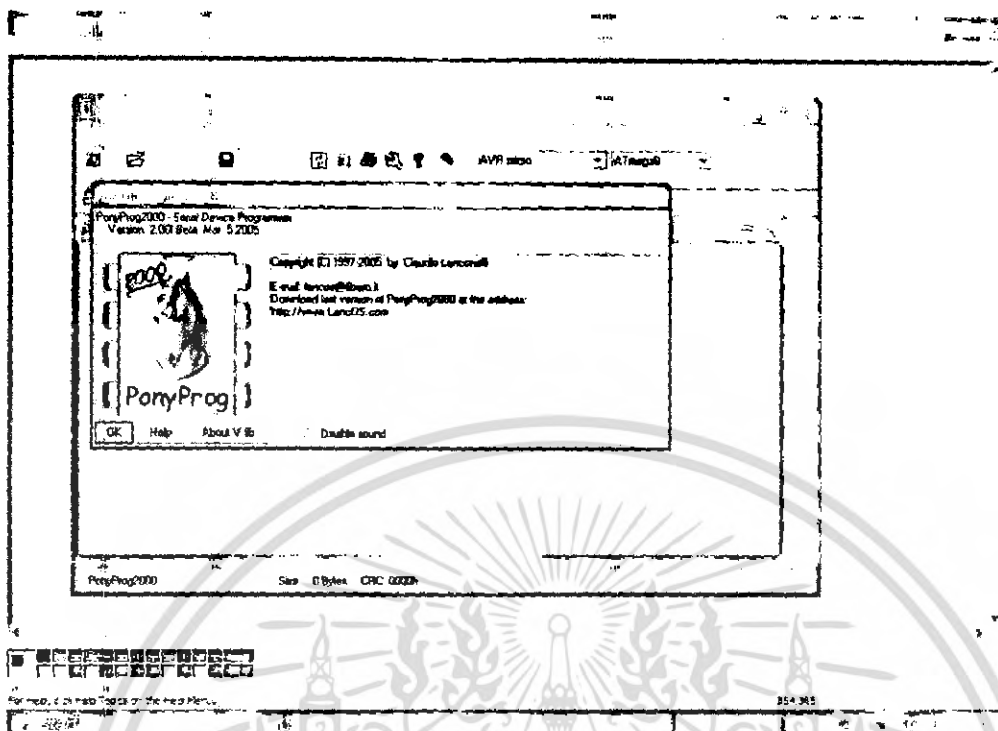


รูปที่ 1 แสดงหน้าต่างโปรแกรม CodeVisionAVR

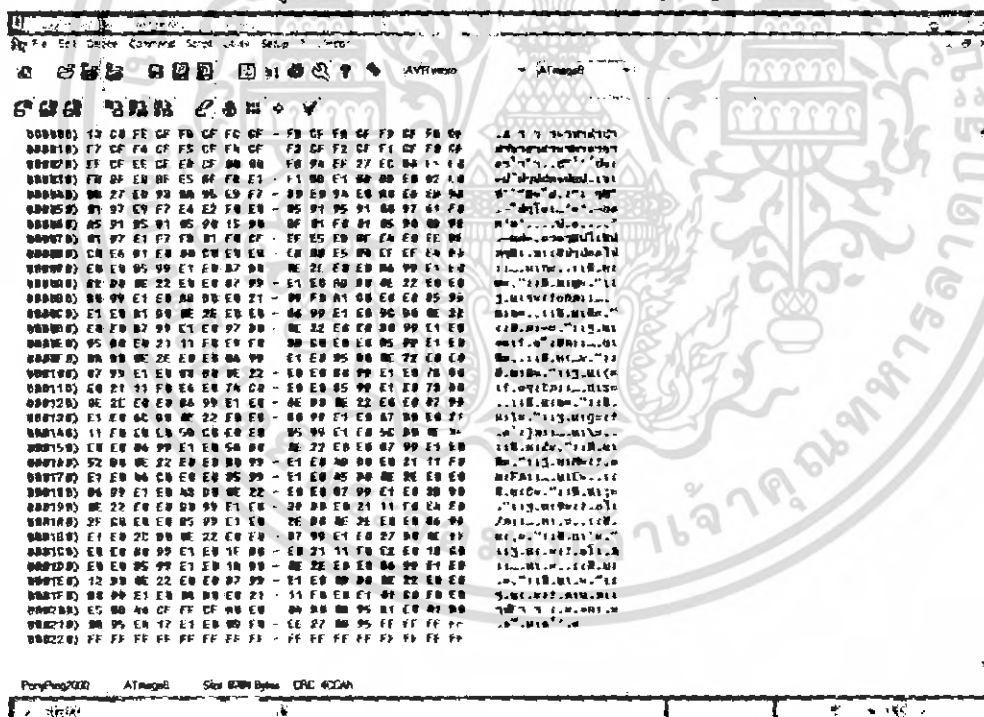


รูปที่ 2 แสดงการ Compile โปรแกรม AVR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3 แสดงหน้าต่างโปรแกรม PonyProg



รูปที่ 4 แสดงโปรแกรม .hex ที่จะทำการเบิร์น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- High-performance, Low-power AVR[®] 8-bit Microcontroller
- Advanced RISC Architecture
 - 130 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
 - 8K Bytes of In-System Self-Programmable Flash
Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
In-System Programming by On-chip Boot Program
True Read-While-Write Operation
 - 512 Bytes EEPROM
Endurance: 100,000 Write/Erase Cycles
 - 1K Byte Internal SRAM
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Three PWM Channels
 - 8-channel ADC in TQFP and MLF package
 - Six Channels 10-bit Accuracy
 - Two Channels 8-bit Accuracy
 - 6-channel ADC in PDIP package
 - Four Channels 10-bit Accuracy
 - Two Channels 8-bit Accuracy
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-lead PDIP, 32-lead TQFP, and 32-pad MLF
- Operating Voltages
 - 2.7 - 5.5V (ATmega8L)
 - 4.5 - 5.5V (ATmega8)
- Speed Grades
 - 0 - 8 MHz (ATmega8L)
 - 0 - 16 MHz (ATmega8)
- Power Consumption at 4 Mhz, 3V, 25°C
 - Active: 3.6 mA
 - Idle Mode: 1.0 mA
 - Power-down Mode: 0.5 μ A



8-bit AVR[®]
with 8K Bytes
In-System
Programmable
Flash

ATmega8
ATmega8L

Preliminary
Summary

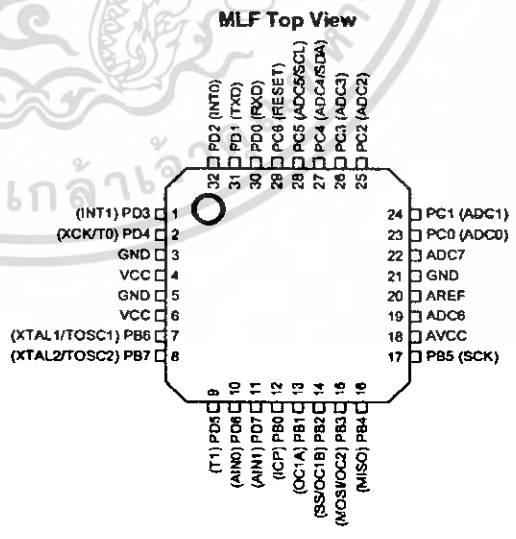
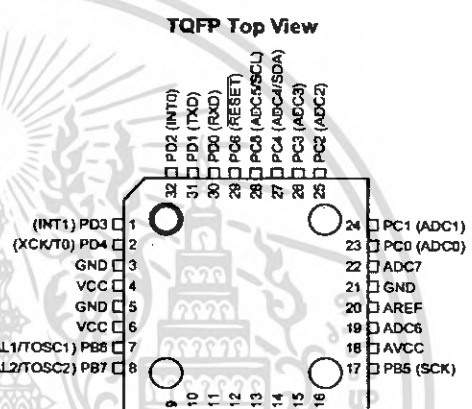
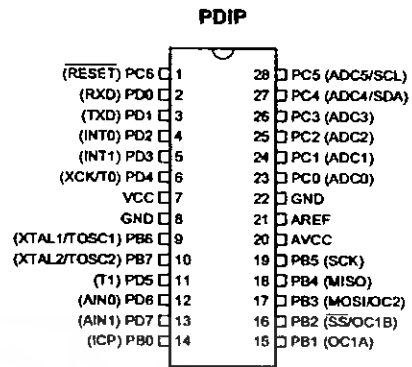
Rev. 2486IS-AVR-12/02



Note: This is a summary document. A complete document is available on our website at www.atmel.com.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Configurations



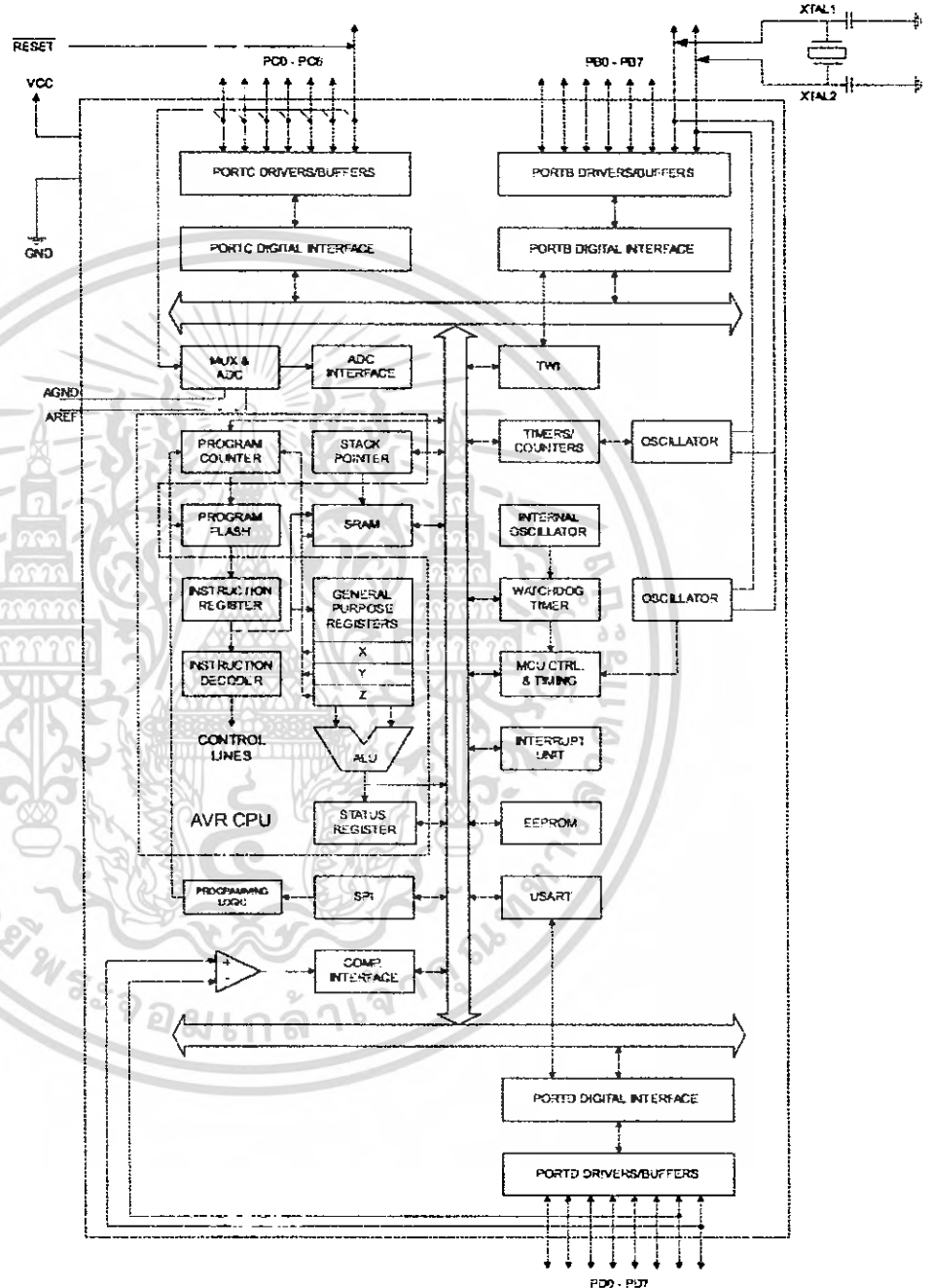
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Overview

The ATmega8 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8 achieves throughputs approaching 1 MIPS per MHz, allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 1. Block Diagram





The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega8 provides the following features: 8K bytes of In-System Programmable Flash with Read-While-Write capabilities, 512 bytes of EEPROM, 1K byte of SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, a 6-channel ADC (eight channels in TQFP and MLF packages) where four (six) channels have 10-bit accuracy and two channels have 8-bit accuracy, a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next Interrupt or Hardware Reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption.

The device is manufactured using Atmel's high density non-volatile memory technology. The Flash Program memory can be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash Memory. Software in the Boot Flash Section will continue to run while the Application Flash Section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega8 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The ATmega8 AVR is supported with a full suite of program and system development tools, including C compilers, macro assemblers, program debugger/simulators, In-Circuit Emulators, and evaluation kits.

Disclaimer

Typical values contained in this data sheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

4 ATmega8(L)

2486IS-AVR-12/02

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Descriptions

VCC	Digital supply voltage.
GND	Ground.
Port B (PB7..PB0)/XTAL1/ XTAL2/TOSC1/TOSC2	<p>Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.</p> <p>Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.</p> <p>If the Internal Calibrated RC Oscillator is used as chip clock source, PB7..6 is used as TOSC2..1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.</p> <p>The various special features of Port B are elaborated on page 56.</p>
Port C (PC5..PC0)	<p>Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p>
PC6/RESET	<p>If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.</p> <p>If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in Table 15 on page 36. Shorter pulses are not guaranteed to generate a Reset.</p> <p>The various special features of Port C are elaborated on page 59.</p>
Port D (PD7..PD0)	<p>Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.</p> <p>Port D also serves the functions of various special features of the ATmega8 as listed on page 61.</p>
RESET	Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in Table 15 on page 36. Shorter pulses are not guaranteed to generate a reset.
XTAL1	Input to the inverting Oscillator amplifier and input to the internal clock operating circuit.
XTAL2	Output from the inverting Oscillator amplifier.

AVCC

AVCC is the supply voltage pin for the A/D Converter, Port C (3..0), and ADC (7..6). It should be externally connected to V_{CC} , even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter. Note that Port C (5..4) use digital supply voltage, V_{CC} .

AREF

AREF is the analog reference pin for the A/D Converter.

ADC7..6 (TQFP and MLF Package Only)

In the TQFP and MLF package, ADC7..6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

About Code Examples

This datasheet contains simple code examples that briefly show how to use various parts of the device. These code examples assume that the part specific header file is included before compilation. Be aware that not all C compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C compiler documentation for more details.



Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
0x3F (0x5F)	SREG	I	T	H	S	V	N	Z	C	9
0x3E (0x5E)	SPH	-	-	-	-	-	SP10	SP9	SP8	11
0x3D (0x5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	11
0x3C (0x5C)	Reserved									
0x3B (0x5B)	GICR	INT1	INT0	-	-	-	-	IVSEL	IVCE	47, 65
0x3A (0x5A)	GIFR	INTF1	INTF0	-	-	-	-	-	-	68
0x39 (0x59)	TIMSK	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	-	TOIE0	70, 100, 120
0x38 (0x58)	TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	-	TOV0	71, 101, 120
0x37 (0x57)	SPMCR	SPMIE	RWWSB	-	RWWSRE	BLBSET	PGWRT	PGERS	SPMEN	208
0x36 (0x56)	TWCR	TWINT	TWEA	TWSTA	TWST0	TWWC	TWEN	-	TWIE	167
0x35 (0x55)	MCUCR	SE	SM2	SM1	SM0	ISC11	ISC10	ISC01	ISC00	31, 64
0x34 (0x54)	MCUCSR	-	-	-	-	WDRF	BORF	EXTRF	PORF	39
0x33 (0x53)	TCCR0	-	-	-	-	-	CS02	CS01	CS00	70
0x32 (0x52)	TCNT0	Timer/Counter0 (8 Bits)								70
0x31 (0x51)	OSCCAL	Oscillator Calibration Register								29
0x30 (0x50)	SFCSR	-	-	-	ADHSM	ACME	PUD	PSR2	PSR10	58, 73, 121, 188
0x2F (0x4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	95
0x2E (0x4E)	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10	98
0x2D (0x4D)	TCNT1H	Timer/Counter1 – Counter Register High Byte								99
0x2C (0x4C)	TCNT1L	Timer/Counter1 – Counter Register Low Byte								99
0x2B (0x4B)	OCR1AH	Timer/Counter1 – Output Compare Register A High Byte								99
0x2A (0x4A)	OCR1AL	Timer/Counter1 – Output Compare Register A Low Byte								99
0x29 (0x49)	OCR1BH	Timer/Counter1 – Output Compare Register B High Byte								99
0x28 (0x48)	OCR1BL	Timer/Counter1 – Output Compare Register B Low Byte								99
0x27 (0x47)	ICR1H	Timer/Counter1 – Input Capture Register High Byte								100
0x26 (0x46)	ICR1L	Timer/Counter1 – Input Capture Register Low Byte								100
0x25 (0x45)	TCCR2	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	115
0x24 (0x44)	TCNT2	Timer/Counter2 (8 Bits)								117
0x23 (0x43)	OCR2	Timer/Counter2 Output Compare Register								117
0x22 (0x42)	ASSR	-	-	-	-	AS2	TCNZUB	OCR2UB	TCR2UB	117
0x21 (0x41)	WDTGR	-	-	-	WOCE	WDE	WDP2	WDP1	WDP0	41
0x20 ⁽¹⁾ (0x40 ⁽¹⁾)	UBRRH	URSEL	-	-	-	-	UBRR[11:8]			154
	UCSRC	URSEL	UMSEL	UPM1	UPM0	USBS	UCS21	UCS20	UCPOL	152
0x1F (0x3F)	EEARH	-	-	-	-	-	-	-	EEARB	18
0x1E (0x3E)	EEARL	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	18
0x1D (0x3D)	EEDR	EEPROM Data Register								18
0x1C (0x3C)	EEDR	-	-	-	-	EERIE	EEMWE	EEWE	EERE	18
0x1B (0x3B)	Reserved									
0x1A (0x3A)	Reserved									
0x19 (0x39)	Reserved									
0x18 (0x38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	63
0x17 (0x37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	63
0x16 (0x36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	63
0x15 (0x35)	PORTC	-	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	63
0x14 (0x34)	DDRC	-	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0	63
0x13 (0x33)	PINC	-	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0	63
0x12 (0x32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	63
0x11 (0x31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	63
0x10 (0x30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	63
0x0F (0x2F)	SPDR	SPI Data Register								128
0x0E (0x2E)	SPSR	SPIF	WCOL	-	-	-	-	-	SPI2X	128
0x0D (0x2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	128
0x0C (0x2C)	UDR	USART I/O Data Register								149
0x0B (0x2B)	UCSRA	RXC	TXC	UDRE	FE	DOR	PE	U2X	MPCM	150
0x0A (0x2A)	UCSRB	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCS22	RXB8	TXB8	151
0x09 (0x29)	UBRRL	USART Baud Rate Register Low Byte								154
0x08 (0x28)	ACSR	ACD	ACBG	ACD	ACI	ACIE	ACIC	ACIS1	ACIS0	189
0x07 (0x27)	ADMUX	REFS1	REFS0	ADLAR	-	MUX3	MUX2	MUX1	MUX0	200
0x06 (0x26)	ADCSRA	ADEN	ADSC	ADFR	ADIF	ADIE	ADPS2	ADPS1	ADPS0	202
0x05 (0x25)	ADCH	ADC Data Register High Byte								203
0x04 (0x24)	ADCL	ADC Data Register Low Byte								203
0x03 (0x23)	TWDR	Two-wire Serial Interface Data Register								169
0x02 (0x22)	TWAR	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWCE	169



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register Summary (Continued)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
0x01 (0x21)	TWSR	TWS7	TWS6	TWS5	TWS4	TWS3	—	TWPS1	TWPS0	168
0x00 (0x20)	TWBR	Two-wire Serial Interface Bit Rate Register								167

- Notes:
1. Refer to the USART description for details on how to access UBRRH and UCSRC.
 2. For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.
 3. Some of the Status Flags are cleared by writing a logical one to them. Note that the CBI and SBI instructions will operate on all bits in the I/O Register, writing a one back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers 0x00 to 0x1F only.



Instruction Set Summary

Mnemonics	Operands	Description	Operation	Flags	#Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add two Registers	$Rd \leftarrow Rd + Rr$	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry two Registers	$Rd \leftarrow Rd + Rr + C$	Z,C,N,V,H	1
ADIW	RdI,K	Add Immediate to Word	$RdH:RdL \leftarrow RdH:RdL + K$	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract two Registers	$Rd \leftarrow Rd - Rr$	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	$Rd \leftarrow Rd - K$	Z,C,N,V,H	1
SBC	Rd, Rr	Subtract with Carry two Registers	$Rd \leftarrow Rd - Rr - C$	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	$Rd \leftarrow Rd - K - C$	Z,C,N,V,H	1
SBW	RdI,K	Subtract Immediate from Word	$RdH:RdL \leftarrow RdH:RdL - K$	Z,C,N,V,S	2
AND	Rd, Rr	Logical AND Registers	$Rd \leftarrow Rd \& Rr$	Z,N,V	1
ANDI	Rd, K	Logical AND Register and Constant	$Rd \leftarrow Rd \& K$	Z,N,V	1
OR	Rd, Rr	Logical OR Registers	$Rd \leftarrow Rd \vee Rr$	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	$Rd \leftarrow Rd \vee K$	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	$Rd \leftarrow Rd \oplus Rr$	Z,N,V	1
COM	Rd	One's Complement	$Rd \leftarrow 0xFF - Rd$	Z,C,N,V	1
NEG	Rd	Two's Complement	$Rd \leftarrow 0x00 - Rd$	Z,C,N,V,H	1
SBR	Rd,K	Set Bit(s) in Register	$Rd \leftarrow Rd \vee K$	Z,N,V	1
CBR	Rd,K	Clear Bit(s) in Register	$Rd \leftarrow Rd \& (0xFF - K)$	Z,N,V	1
INC	Rd	Increment	$Rd \leftarrow Rd + 1$	Z,N,V	1
DEC	Rd	Decrement	$Rd \leftarrow Rd - 1$	Z,N,V	1
TST	Rd	Test for Zero or Minus	$Rd \leftarrow Rd \& Rd$	Z,N,V	1
CLR	Rd	Clear Register	$Rd \leftarrow Rd \& Rd$	Z,N,V	1
SER	Rd	Set Register	$Rd \leftarrow 0xFF$	None	1
MUL	Rd, Rr	Multiply Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
MULS	Rd, Rr	Multiply Signed	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
FMUL	Rd, Rr	Fractional Multiply Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \lll 1$	Z,C	2
FMULS	Rd, Rr	Fractional Multiply Signed	$R1:R0 \leftarrow (Rd \times Rr) \lll 1$	Z,C	2
FMULSU	Rd, Rr	Fractional Multiply Signed with Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \lll 1$	Z,C	2
BRANCH INSTRUCTIONS					
RJMP	k	Relative Jump	$PC \leftarrow PC + k + 1$	None	2
IJMP		Indirect Jump to (Z)	$PC \leftarrow Z$	None	2
JMP	k	Direct Jump	$PC \leftarrow k$	None	3
RCALL	k	Relative Subroutine Call	$PC \leftarrow PC + k + 1$	None	3
ICALL		Indirect Call to (Z)	$PC \leftarrow Z$	None	3
CALL	k	Direct Subroutine Call	$PC \leftarrow k$	None	4
RET		Subroutine Return	$PC \leftarrow STACK$	None	4
RETI		Interrupt Return	$PC \leftarrow STACK$	I	4
CPSE	Rd,Rr	Compare, Skip if Equal	$\text{if } (Rd = Rr) PC \leftarrow PC + 2 \text{ or } 3$	None	1/2/3
CP	Rd,Rr	Compare	$Rd - Rr$	Z, N, V, C, H	1
CPC	Rd,Rr	Compare with Carry	$Rd - Rr - C$	Z, N, V, C, H	1
CPI	Rd,K	Compare Register with Immediate	$Rd - K$	Z, N, V, C, H	1
SBRC	Rr, b	Skip if Bit in Register Cleared	$\text{if } (Rr(b)=0) PC \leftarrow PC + 2 \text{ or } 3$	None	1/2/3
SBR S	Rr, b	Skip if Bit in Register is Set	$\text{if } (Rr(b)=1) PC \leftarrow PC + 2 \text{ or } 3$	None	1/2/3
SBIC	P, b	Skip if Bit in I/O Register Cleared	$\text{if } (P(b)=0) PC \leftarrow PC + 2 \text{ or } 3$	None	1/2/3
SBIS	P, b	Skip if Bit in I/O Register is Set	$\text{if } (P(b)=1) PC \leftarrow PC + 2 \text{ or } 3$	None	1/2/3
BRBS	s, k	Branch if Status Flag Set	$\text{if } (SREG(s) = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRBC	s, k	Branch if Status Flag Cleared	$\text{if } (SREG(s) = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BR EQ	k	Branch if Equal	$\text{if } (Z = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRNE	k	Branch if Not Equal	$\text{if } (Z = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRCS	k	Branch if Carry Set	$\text{if } (C = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRCC	k	Branch if Carry Cleared	$\text{if } (C = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRSH	k	Branch if Same or Higher	$\text{if } (C = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRLO	k	Branch if Lower	$\text{if } (C = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRMI	k	Branch if Minus	$\text{if } (N = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRPL	k	Branch if Plus	$\text{if } (N = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRGE	k	Branch if Greater or Equal, Signed	$\text{if } (N \oplus V = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRLT	k	Branch if Less Than Zero, Signed	$\text{if } (N \oplus V = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRHS	k	Branch if Half Carry Flag Set	$\text{if } (H = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRHC	k	Branch if Half Carry Flag Cleared	$\text{if } (H = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRTS	k	Branch if T Flag Set	$\text{if } (T = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRTC	k	Branch if T Flag Cleared	$\text{if } (T = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRYS	k	Branch if Overflow Flag is Set	$\text{if } (V = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRVC	k	Branch if Overflow Flag is Cleared	$\text{if } (V = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Instruction Set Summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clocks
BRIE	k	Branch if Interrupt Enabled	$\text{if } (I = 1) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
BRID	k	Branch if Interrupt Disabled	$\text{if } (I = 0) \text{ then } PC \leftarrow PC + k + 1$	None	1/2
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move Between Registers	$Rd \leftarrow Rr$	None	1
MOVW	Rd, Rr	Copy Register Word	$Rd+1:Rd \leftarrow Rr+1:Rr$	None	1
LDI	Rd, K	Load Immediate	$Rd \leftarrow K$	None	1
LD	Rd, X	Load Indirect	$Rd \leftarrow (X)$	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	$Rd \leftarrow (X), X \leftarrow X + 1$	None	2
LD	Rd, -X	Load Indirect and Pre-Dec.	$X \leftarrow X - 1, Rd \leftarrow (X)$	None	2
LD	Rd, Y	Load Indirect	$Rd \leftarrow (Y)$	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	None	2
LD	Rd, -Y	Load Indirect and Pre-Dec.	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	None	2
LDD	Rd, Y+q	Load Indirect with Displacement	$Rd \leftarrow (Y + q)$	None	2
LD	Rd, Z	Load Indirect	$Rd \leftarrow (Z)$	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	$Rd \leftarrow (Z + q)$	None	2
LDS	Rd, k	Load Direct from SRAM	$Rd \leftarrow (k)$	None	2
ST	X, Rr	Store Indirect	$(X) \leftarrow Rr$	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	$(X) \leftarrow Rr, X \leftarrow X + 1$	None	2
ST	-X, Rr	Store Indirect and Pre-Dec.	$X \leftarrow X - 1, (X) \leftarrow Rr$	None	2
ST	Y, Rr	Store Indirect	$(Y) \leftarrow Rr$	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	None	2
ST	-Y, Rr	Store Indirect and Pre-Dec.	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	None	2
STD	Y+q, Rr	Store Indirect with Displacement	$(Y + q) \leftarrow Rr$	None	2
ST	Z, Rr	Store Indirect	$(Z) \leftarrow Rr$	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	None	2
STD	Z+q, Rr	Store Indirect with Displacement	$(Z + q) \leftarrow Rr$	None	2
STS	k, Rr	Store Direct to SRAM	$(k) \leftarrow Rr$	None	2
LPM		Load Program Memory	$R0 \leftarrow (Z)$	None	3
LPM	Rd, Z	Load Program Memory	$Rd \leftarrow (Z)$	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	None	3
SPM		Store Program Memory	$(Z) \leftarrow R1:R0$	None	-
IN	Rd, P	In Port	$Rd \leftarrow P$	None	1
OUT	P, Rr	Out Port	$P \leftarrow Rr$	None	1
PUSH	Rr	Push Register on Stack	$STACK \leftarrow Rr$	None	2
POP	Rd	Pop Register from Stack	$Rd \leftarrow STACK$	None	2
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P, b	Set Bit in I/O Register	$\text{I/O}(P, b) \leftarrow 1$	None	2
CBI	P, b	Clear Bit in I/O Register	$\text{I/O}(P, b) \leftarrow 0$	None	2
LSL	Rd	Logical Shift Left	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	Z, C, N, V	1
LSR	Rd	Logical Shift Right	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	Z, C, N, V	1
ROL	Rd	Rotate Left Through Carry	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	Z, C, N, V	1
ROR	Rd	Rotate Right Through Carry	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	Z, C, N, V	1
ASR	Rd	Arithmetic Shift Right	$Rd(n) \leftarrow Rd(n+1), n=0..6$	Z, C, N, V	1
SWAP	Rd	Swap Nibbles	$Rd(3..0) \leftarrow Rd(7..4), Rd(7..4) \leftarrow Rd(3..0)$	None	1
BSET	s	Flag Set	$SREG(s) \leftarrow 1$	SREG(s)	1
BCLR	s	Flag Clear	$SREG(s) \leftarrow 0$	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	$T \leftarrow Rr(b)$	T	1
BLD	rd, b	Bit Load from T to Register	$Rd(b) \leftarrow T$	None	1
SEC		Set Carry	$C \leftarrow 1$	C	1
CLC		Clear Carry	$C \leftarrow 0$	C	1
SEN		Set Negative Flag	$N \leftarrow 1$	N	1
CLN		Clear Negative Flag	$N \leftarrow 0$	N	1
SEZ		Set Zero Flag	$Z \leftarrow 1$	Z	1
CLZ		Clear Zero Flag	$Z \leftarrow 0$	Z	1
SEI		Global Interrupt Enable	$I \leftarrow 1$	I	1
CLI		Global Interrupt Disable	$I \leftarrow 0$	I	1
SES		Set Signed Test Flag	$S \leftarrow 1$	S	1
CLS		Clear Signed Test Flag	$S \leftarrow 0$	S	1
SEV		Set Twos Complement Overflow	$V \leftarrow 1$	V	1
CLV		Clear Twos Complement Overflow	$V \leftarrow 0$	V	1
SET		Set T in SREG	$T \leftarrow 1$	T	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Instruction Set Summary (Continued)

Mnemonics	Operands	Description	Operation	Flags	#Clocks
CLT		Clear T in SREG	T ← 0	T	1
SEH		Set Half Carry Flag in SREG	H ← 1	H	1
CLH		Clear Half Carry Flag in SREG	H ← 0	H	1
MCU CONTROL INSTRUCTIONS					
NOP		No Operation		None	1
SLEEP		Sleep	(see specific descr. for Sleep function)	None	1
WDR		Watchdog Reset	(see specific descr. for WDR/timer)	None	1





Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
8	2.7 - 5.5	ATmega8L-8AC	32A	Commercial (0°C to 70°C)
		ATmega8L-8PC	28P3	
		ATmega8L-8MC	32M1-A	
16	4.5 - 5.5	ATmega8L-8AI	32A	Industrial (-40°C to 85°C)
		ATmega8L-8PI	28P3	
		ATmega8L-8MI	32M1-A	
16	4.5 - 5.5	ATmega8-16AC	32A	Commercial (0°C to 70°C)
		ATmega8-16PC	28P3	
		ATmega8-16MC	32M1-A	
16	4.5 - 5.5	ATmega8-16AI	32A	Industrial (-40°C to 85°C)
		ATmega8-16PI	28P3	
		ATmega8-16MI	32M1-A	

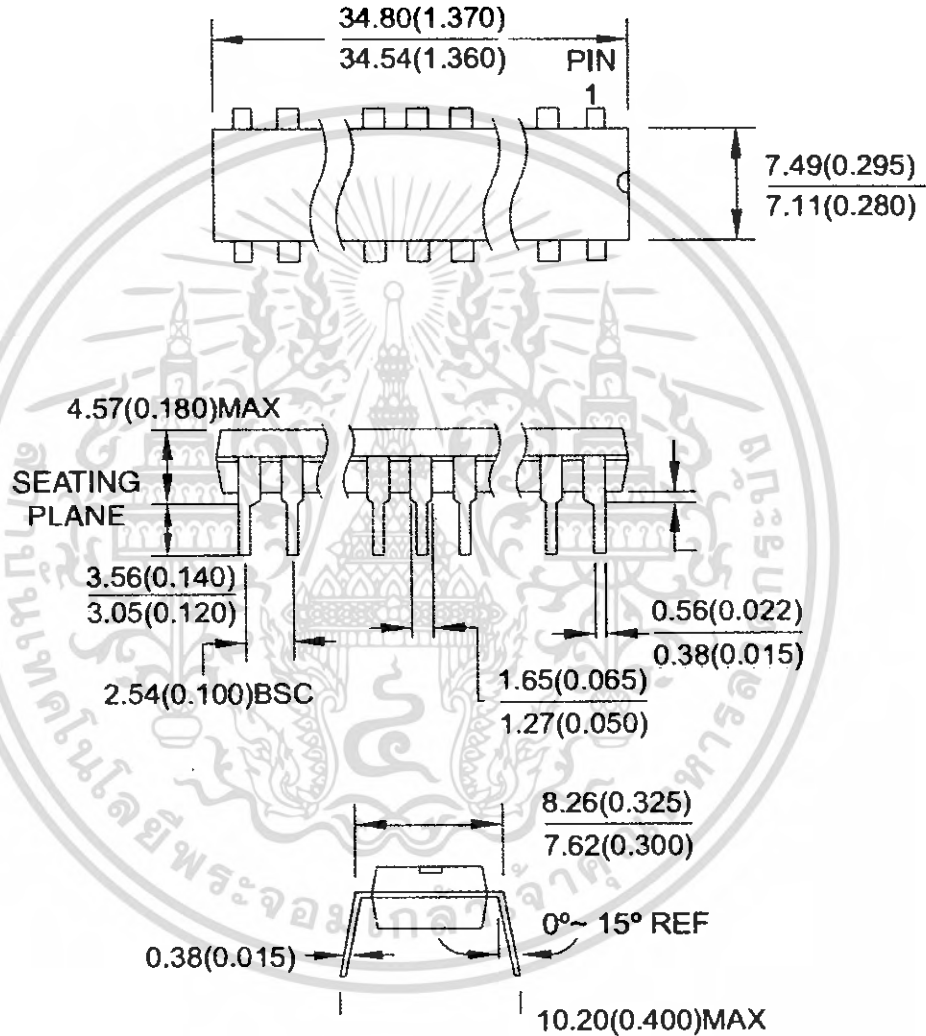
Note: This device can also be supplied in wafer form. Please contact your local Atmel sales office for detailed ordering information and minimum quantities.



Package Type	
32A	32-lead, Thin (1.0 mm) Plastic Quad Flat Package (TQFP)
28P3	28-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
32M1-A	32-pad, 5 x 5 x 1.0 body, Lead Pitch 0.50 mm Micro Lead Frame Package (MLF)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

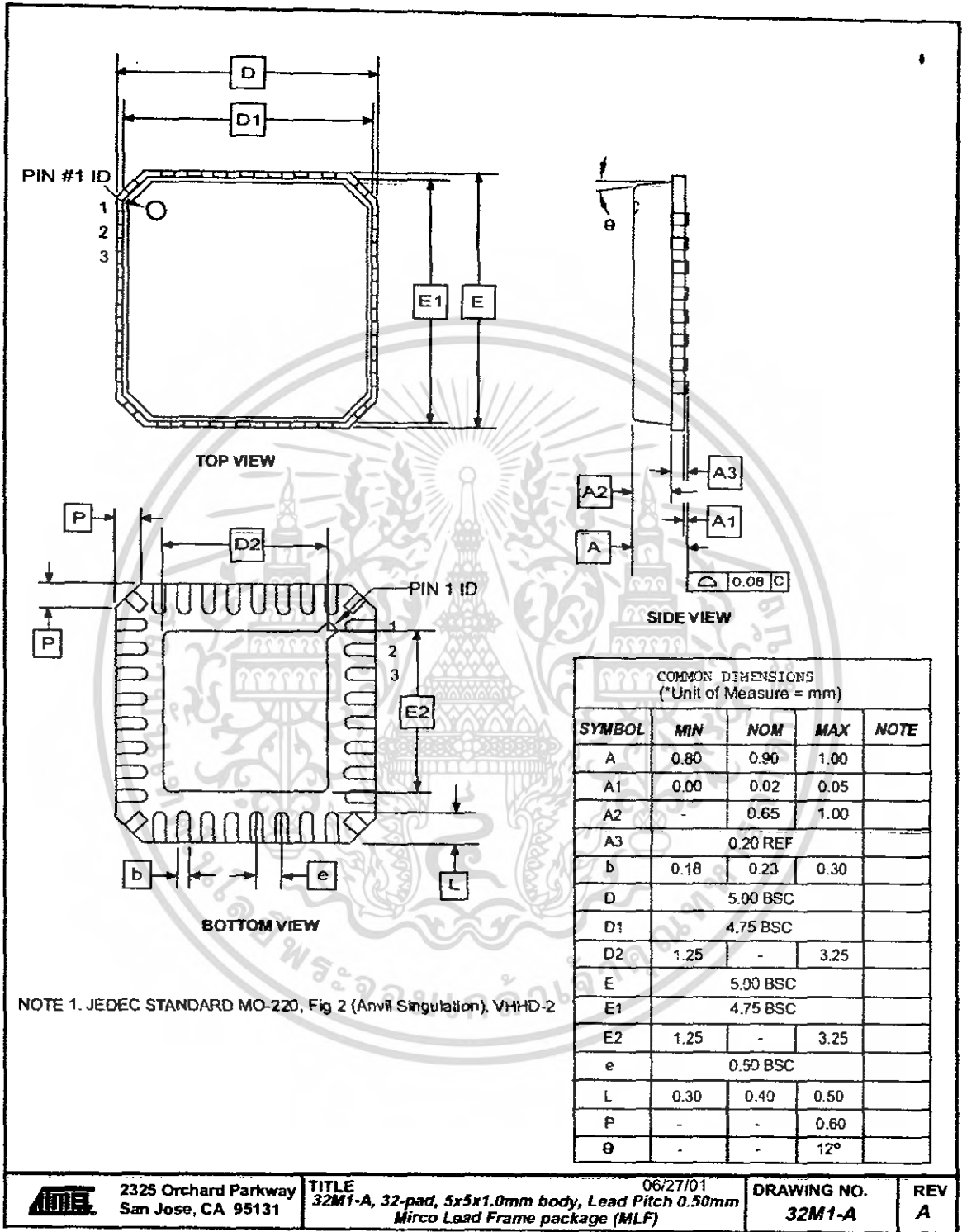
28-lead, Plastic Dual Inline
 Package (PDIP), 0.300" Wide, (0.288" body width)
 Dimensions in Millimeters and (Inches)*



*Controlling dimension: Inches

REV. A 04/11/2001

32M1-A



2325 Orchard Parkway
San Jose, CA 95131

TITLE
32M1-A, 32-pad, 5x5x1.0mm body, Lead Pitch 0.50mm
Micro Lead Frame package (MLF)

06/27/01

DRAWING NO.
32M1-A

REV
A



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data Sheet Change Log for ATmega8

This document contains a log on the changes made to the data sheet for ATmega8.

Changes from Rev. 2486B-12/01 to Rev. 2486C-03/02

All page numbers refers to this document.

1 Updated TWI Chapter.

More details regarding use of the TWI Power-down operation and using the TWI as Master with low TWBRR values are added into the data sheet.

Added the note at the end of the "Bit Rate Generator Unit" on page 165.

Added the description at the end of "Address Match Unit" on page 166.

2 Updated Description of OSCCAL Calibration Byte.

In the data sheet, it was not explained how to take advantage of the calibration bytes for 2, 4, and 8 MHz Oscillator selections. This is now added in the following sections:

Improved description of "Oscillator Calibration Register – OSCCAL" on page 29 and "Calibration Byte" on page 218.

3 Added Some Preliminary Test Limits and Characterization Data.

Removed some of the TBD's in the following tables and pages:

Table 3 on page 24, Table 15 on page 36, Table 16 on page 40, Table 17 on page 42, Table 99 on page 233, "DC Characteristics $T_A = -40^\circ\text{C}$ to 85°C , $V_{CC} = 2.7\text{V}$ to 5.5V (unless otherwise noted)" on page 234, Table 100 on page 236, and Table 103 on page 238.

4 Updated Programming Figures.

Figure 104 on page 219 and Figure 112 on page 229 are updated to also reflect that AVCC must be connected during Programming mode.

5 Added a Description on how to Enter Parallel Programming Mode if RESET Pin is Disabled or if External Oscillators are Selected.

Added a note in section "Enter Programming Mode" on page 221

Changes from Rev. 2486C-03/02 to Rev. 2486D-03/02

All page numbers refers to this document.

1 Updated Typical Start-up Times.

The following tables has been updated:

Table 5, "Start-up Times for the Crystal Oscillator Clock Selection," on page 26, Table 6, "Start-up Times for the Low-frequency Crystal Oscillator Clock Selection," on page 26, Table 8, "Start-up Times for the External RC Oscillator Clock Selection," on page 27, and Table 12, "Start-up Times for the External Clock Selection," on page 30.

2 Added "ATmega8 Typical Characteristics – Preliminary Data" on page 241.

Changes from Rev. 2486D-03/02 to Rev. 2486E-06/02

All page numbers refers to this document.

1 Updated Some Preliminary Test Limits and Characterization Data

The following tables have been updated:

Table 15, "Reset Characteristics," on page 36, Table 16, "Internal Voltage Reference Characteristics," on page 40, DC Characteristics on page 234, Table , "ADC Characteristics – Preliminary Data," on page 240.

2 Changes in External Clock Frequency

Added the description at the end of "External Clock" on page 30.

Added period changing data in Table 100, "External Clock Drive," on page 236.

3 Updated TWI Chapter

More details regarding use of the TWI bit rate prescaler and a Table 65, "TWI Bit Rate Prescaler," on page 169.

Changes from Rev. 2486E-06/02 to Rev. 2486F-07/02

All page numbers refers to this document.

1 Changes in "Digital Input Enable and Sleep Modes" on page 53.

2 Addition of OCS2 in "MOSI/OC2 – Port B, Bit 3" on page 57.

3 The following tables has been updated:

Table 51, "CPOL and CPHA Functionality," on page 129, Table 59, "UCPOL Bit Settings," on page 154, Table 72, "Analog Comparator Multiplexed Input⁽¹⁾," on page 190, Table 73, "ADC Conversion Time," on page 195, Table 75, "Input Channel Selections," on page 201, and Table 84, "Explanation of Different Variables used in Figure 103 and the Mapping to the Z-pointer," on page 215.

5 Changes in "Reading the Calibration Byte" on page 227.

6 Corrected Errors in Cross References.

Changes from Rev. 2486F-07/02 to Rev. 2486G-09/02

All page numbers refers to this document.

1 Updated Table 104, "ADC Characteristics," on page 240.

Changes from Rev. 2486G-09/02 to Rev. 2486H-09/02

1. Changed the Endurance on the Flash to 10,000 Write/Erase Cycles.

Changes from Rev. 2486H-09/02 to Rev. 2486I-12/02

1. Added errat for Rev D, E, and F on page 18.

Erratas

The revision letter in this section refers to the revision of the ATmega8 device.

ATmega8 Rev. D, E, and F

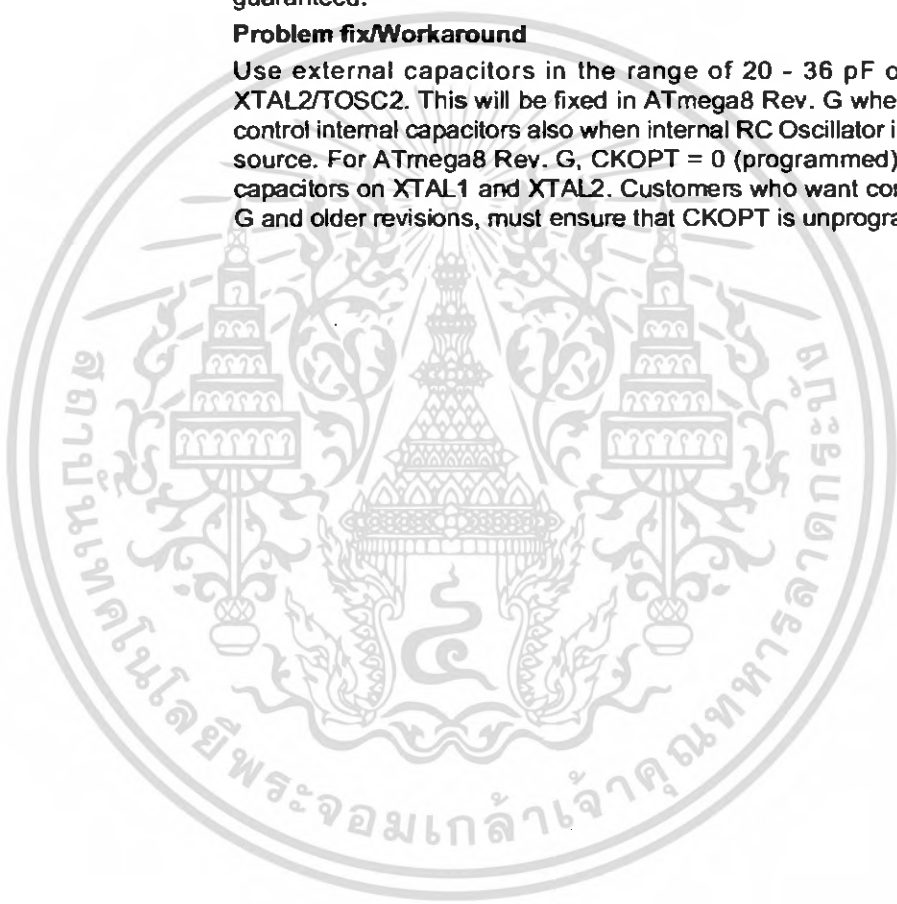
• **CKOPT Does not Enable Internal Capacitors on XTALn/TOSCn Pins when 32 KHz Oscillator is Used to Clock the Asynchronous Timer/Counter2**

1. CKOPT Does not Enable Internal Capacitors on XTALn/TOSCn Pins when 32 KHz Oscillator is Used to Clock the Asynchronous Timer/Counter2

When the internal RC Oscillator is used as the main clock source, it is possible to run the Timer/Counter2 asynchronously by connecting a 32 KHz Oscillator between XTAL1/TOSC1 and XTAL2/TOSC2. But when the internal RC Oscillator is selected as the main clock source, the CKOPT Fuse does not control the internal capacitors on XTAL1/TOSC1 and XTAL2/TOSC2. As long as there are no capacitors connected to XTAL1/TOSC1 and XTAL2/TOSC2, safe operation of the Oscillator is not guaranteed.

Problem fix/Workaround

Use external capacitors in the range of 20 - 36 pF on XTAL1/TOSC1 and XTAL2/TOSC2. This will be fixed in ATmega8 Rev. G where the CKOPT Fuse will control internal capacitors also when internal RC Oscillator is selected as main clock source. For ATmega8 Rev. G, CKOPT = 0 (programmed) will enable the internal capacitors on XTAL1 and XTAL2. Customers who want compatibility between Rev. G and older revisions, must ensure that CKOPT is unprogrammed (CKOPT = 1).







Atmel Headquarters

Corporate Headquarters

2325 Orchard Parkway
San Jose, CA 95131
TEL 1(408) 441-0311
FAX 1(408) 487-2600

Europe

Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
TEL (41) 26-426-5555
FAX (41) 26-426-5500

Asia

Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimhatsui
East Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Japan

9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
TEL (81) 3-3523-3551
FAX (81) 3-3523-7581

Atmel Operations

Memory

2325 Orchard Parkway
San Jose, CA 95131
TEL 1(408) 441-0311
FAX 1(408) 436-4314

Microcontrollers

2325 Orchard Parkway
San Jose, CA 95131
TEL 1(408) 441-0311
FAX 1(408) 436-4314

La Chanterrie

BP 70602
44306 Nantes Cedex 3, France
TEL (33) 2-40-18-18-18
FAX (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards

Zone Industrielle
13106 Rousset Cedex, France
TEL (33) 4-42-53-60-00
FAX (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.

Colorado Springs, CO 80906
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Scottish Enterprise Technology Park

Maxwell Building
East Kilbride G75 0QR, Scotland
TEL (44) 1355-803-000
FAX (44) 1355-242-743

RF/Automotive

Theresienstrasse 2
Postfach 3535
74025 Heilbronn, Germany
TEL (49) 71-31-67-0
FAX (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Biometrics/Imaging/Hi-Rel MPU/ High Speed Converters/RF Datacom

Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
TEL (33) 4-76-58-30-00
FAX (33) 4-76-58-34-80

e-mail

literature@atmel.com

Web Site

<http://www.atmel.com>

© Atmel Corporation 2002.

Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

ATMEL® and AVR® are the registered trademarks of Atmel.

Other terms and product names may be the trademarks of others.



Printed on recycled paper.

2486IS-AVR-12/02

DM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- Operating voltage
 - 2.4V~5V for the HT12A
 - 2.4V~12V for the HT12E
- Low power and high noise immunity CMOS technology
- Low standby current: 0.1μA (typ.) at V_{DD}=5V
- HT12A with a 38kHz carrier for infrared transmission medium
- Minimum transmission word
 - Four words for the HT12E
 - One word for the HT12A
- Built-in oscillator needs only 5% resistor
- Data code has positive polarity
- Minimal external components
- HT12A/E: 18-pin DIP/20-pin SOP package

Applications

- Burglar alarm system
- Smoke and fire alarm system
- Garage door controllers
- Car door controllers
- Car alarm system
- Security system
- Cordless telephones
- Other remote control systems

General Description

The 2¹² encoders are a series of CMOS LSIs for remote control system applications. They are capable of encoding information which consists of N address bits and 12-N data bits. Each address/data input can be set to one of the two logic states. The programmed addresses/data are transmitted together with the header bits

via an RF or an infrared transmission medium upon receipt of a trigger signal. The capability to select a \overline{TE} trigger on the HT12E or a DATA trigger on the HT12A further enhances the application flexibility of the 2¹² series of encoders. The HT12A additionally provides a 38kHz carrier for infrared systems.

Selection Table

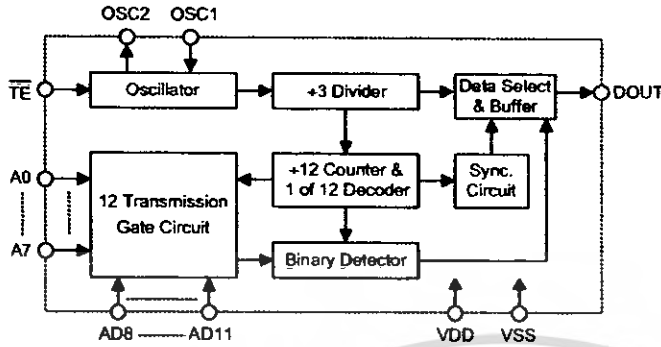
Function Part No.	Address No.	Address/ Data No.	Data No.	Oscillator	Trigger	Package	Carrier Output	Negative Polarity
HT12A	8	0	4	455kHz resonator	D8~D11	18 DIP 20 SOP	38kHz	No
HT12E	8	4	0	RC oscillator	\overline{TE}	18 DIP 20 SOP	No	No

Note: Address/Data represents pins that can be address or data according to the decoder requirement.

Block Diagram

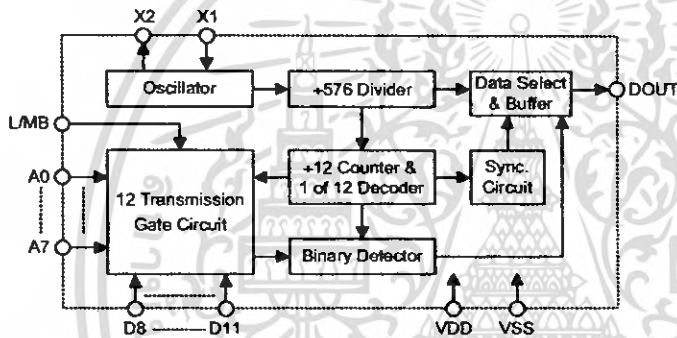
TE trigger

HT12E

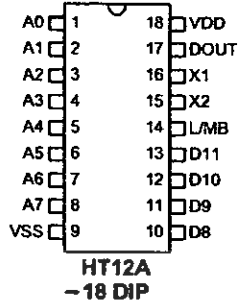
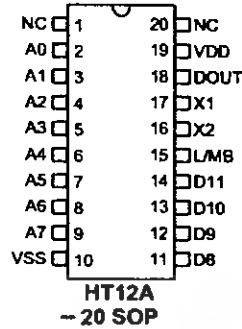
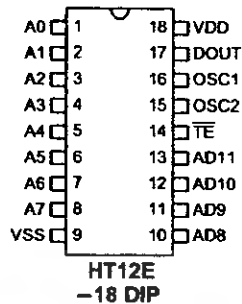
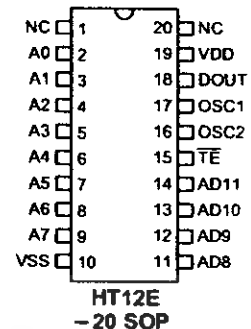


DATA trigger

HT12A



Note: The address data pins are available in various combinations (refer to the address/data table).

Pin Assignment
**8-Address
4-Data**

**8-Address
4-Data**

**8-Address
4-Address/Data**

**8-Address
4-Address/Data**

Pin Description

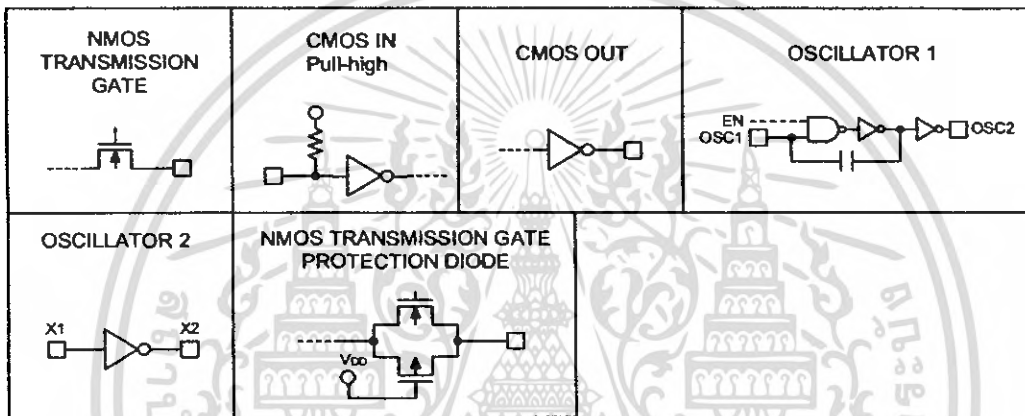
Pin Name	I/O	Internal Connection	Description
A0-A7	I	CMOS IN Pull-high (HT12A) NMOS TRANSMISSION GATE PROTECTION DIODE (HT12E)	Input pins for address A0-A7 setting These pins can be externally set to VSS or left open
AD8-AD11	I	NMOS TRANSMISSION GATE PROTECTION DIODE (HT12E)	Input pins for address/data AD8-AD11 setting These pins can be externally set to VSS or left open
D8-D11	I	CMOS IN Pull-high	Input pins for data D8-D11 setting and transmission enable, active low These pins should be externally set to VSS or left open (see Note)
DOUT	O	CMOS OUT	Encoder data serial transmission output
LMB	I	CMOS IN Pull-high	Latch/Momentary transmission format selection pin: Latch: Floating or VDD Momentary: VSS

Pin Name	I/O	Internal Connection	Description
\overline{TE}	I	CMOS IN Pull-high	Transmission enable, active low (see Note)
OSC1	I	OSCILLATOR 1	Oscillator input pin
OSC2	O	OSCILLATOR 1	Oscillator output pin
X1	I	OSCILLATOR 2	455kHz resonator oscillator input
X2	O	OSCILLATOR 2	455kHz resonator oscillator output
VSS	I	—	Negative power supply, grounds
VDD	I	—	Positive power supply

Note: D8-D11 are all data input and transmission enable pins of the HT12A.

\overline{TE} is a transmission enable pin of the HT12E.

Approximate internal connections



Absolute Maximum Ratings

Supply Voltage (HT12A)	-0.3V to 5.5V	Supply Voltage (HT12E)	-0.3V to 13V
Input Voltage.....	$V_{SS}-0.3$ to $V_{DD}+0.3V$	Storage Temperature.....	-50°C to 125°C
Operating Temperature.....	-20°C to 75°C		

Note: These are stress ratings only. Stresses exceeding the range specified under "Absolute Maximum Ratings" may cause substantial damage to the device. Functional operation of this device at other conditions beyond those listed in the specification is not implied and prolonged exposure to extreme conditions may affect device reliability.

Electrical Characteristics
HT12A

Ta=25°C

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V _{DD}	Conditions				
V _{DD}	Operating Voltage	—	—	2.4	3	5	V
I _{STB}	Standby Current	3V	Oscillator stops	—	0.1	1	μA
		5V		—	0.1	1	μA
I _{DD}	Operating Current	3V	No load f _{OSC} =455kHz	—	200	400	μA
		5V		—	400	800	μA
I _{DOUT}	Output Drive Current	5V	V _{OH} =0.9V _{DD} (Source)	-1	-1.6	—	mA
			V _{OL} =0.1V _{DD} (Sink)	2	3.2	—	mA
V _{IH}	"H" Input Voltage	—	—	0.8V _{DD}	—	V _{DD}	V
V _{IL}	"L" Input Voltage	—	—	0	—	0.2V _{DD}	V
R _{DATA}	D8-D11 Pull-high Resistance	5V	V _{DATA} =0V	—	150	300	kΩ

HT12E

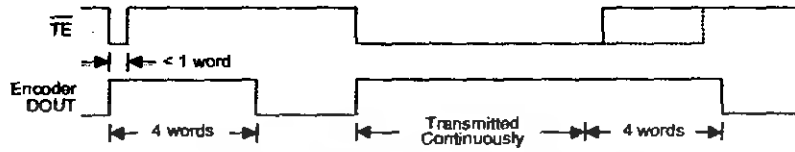
Ta=25°C

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V _{DD}	Conditions				
V _{DD}	Operating Voltage	—	—	2.4	5	12	V
I _{STB}	Standby Current	3V	Oscillator stops	—	0.1	1	μA
		12V		—	2	4	μA
I _{DD}	Operating Current	3V	No load f _{OSC} =3kHz	—	40	80	μA
		12V		—	150	300	μA
I _{DOUT}	Output Drive Current	5V	V _{OH} =0.9V _{DD} (Source)	-1	-1.6	—	mA
			V _{OL} =0.1V _{DD} (Sink)	1	1.6	—	mA
V _{IH}	"H" Input Voltage	—	—	0.8V _{DD}	—	V _{DD}	V
V _{IL}	"L" Input Voltage	—	—	0	—	0.2V _{DD}	V
f _{OSC}	Oscillator Frequency	5V	R _{OSC} =1.1MΩ	—	3	—	kHz
R _{TE}	TE Pull-high Resistance	5V	V _{TE} =0V	—	1.5	3	MΩ

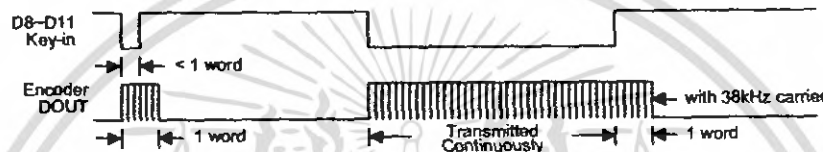
Functional Description

Operation

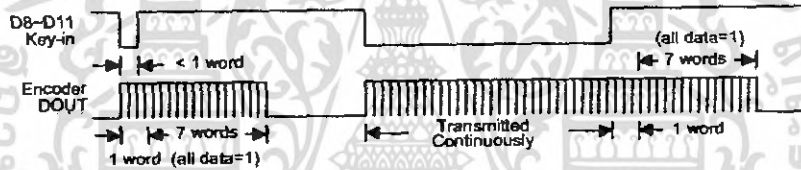
The 2¹² series of encoders begin a 4-word transmission cycle upon receipt of a transmission enable (\overline{TE} for the HT12E or D8-D11 for the HT12A, active low). This cycle will repeat itself as long as the transmission enable (\overline{TE} or D8-D11) is held low. Once the transmission enable returns high the encoder output completes its final cycle and then stops as shown below.



Transmission timing for the HT12E



Transmission timing for the HT12A (L/MB=Floating or VDD)

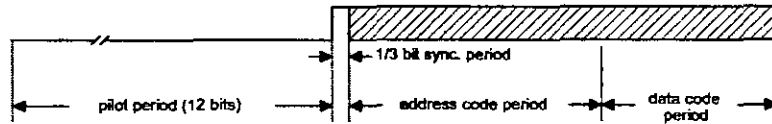


Transmission timing for the HT12A (L/MB=VSS)

Information word

If L/MB=1 the device is in the latch mode (for use with the latch type of data decoders). When the transmission enable is removed during a transmission, the DOUT pin outputs a complete word and then stops. On the other hand, if L/MB=0 the device is in the momentary mode (for use with the momentary type of data decoders). When the transmission enable is removed during a transmission, the DOUT outputs a complete word and then adds 7 words all with the "1" data code.

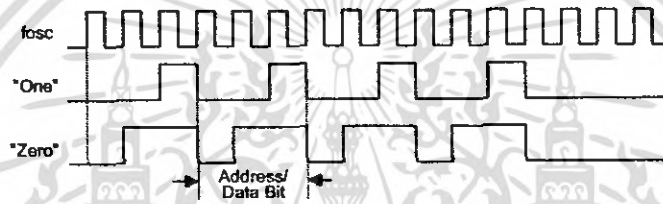
An information word consists of 4 periods as illustrated below.



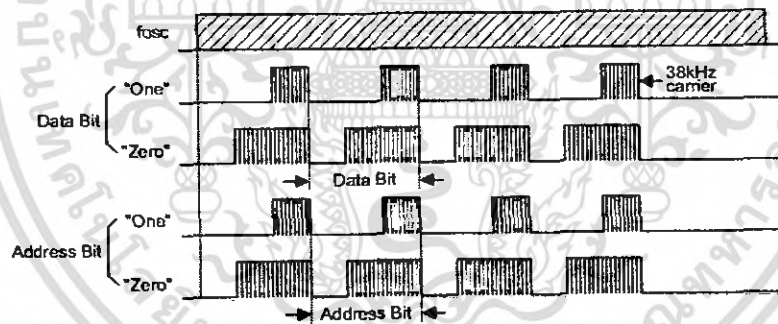
Composition of information

Address/data waveform

Each programmable address/data pin can be externally set to one of the following two logic states as shown below.



Address/Data bit waveform for the HT12E



Address/Data bit waveform for the HT12A

The address/data bits of the HT12A are transmitted with a 38kHz carrier for infrared remote controller flexibility.

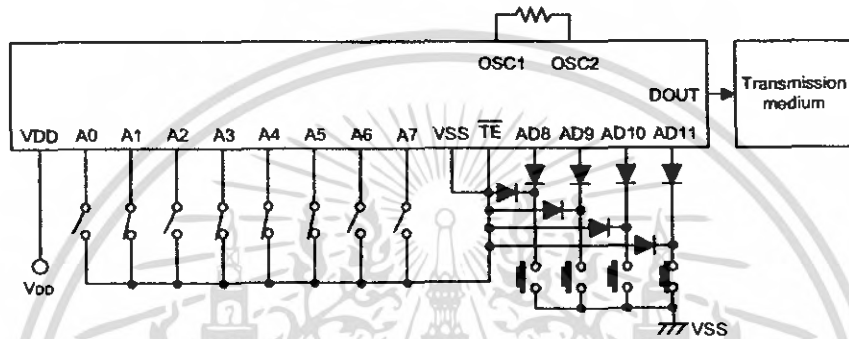
Address/data programming (preset)

The status of each address/data pin can be individually pre-set to logic "high" or "low". If a transmission-enable signal is applied, the encoder scans and transmits the status of the 12 bits of address/data serially in the order A0 to AD11 for the HT12E encoder and A0 to D11 for the HT12A encoder.

During information transmission these bits are transmitted with a preceding synchronization bit. If the trigger signal is not applied, the chip enters the standby mode and consumes a reduced current of less than 1µA for a supply voltage of 5V.

Usual applications preset the address pins with individual security codes using DIP switches or PCB wiring, while the data is selected by push buttons or electronic switches.

The following figure shows an application using the HT12E:



The transmitted information is as shown:

Pilot & Sync.	A0	A1	A2	A3	A4	A5	A6	A7	AD8	AD9	AD10	AD11
1	0	1	0	0	0	1	1	1	1	1	1	0

Address/Data sequence

The following provides the address/data sequence table for various models of the 2¹² series of encoders. The correct device should be selected according to the individual address and data requirements.

Part No.	Address/Data Bits											
	0	1	2	3	4	5	6	7	8	9	10	11
HT12A	A0	A1	A2	A3	A4	A5	A6	A7	D8	D9	D10	D11
HT12E	A0	A1	A2	A3	A4	A5	A6	A7	AD8	AD9	AD10	AD11

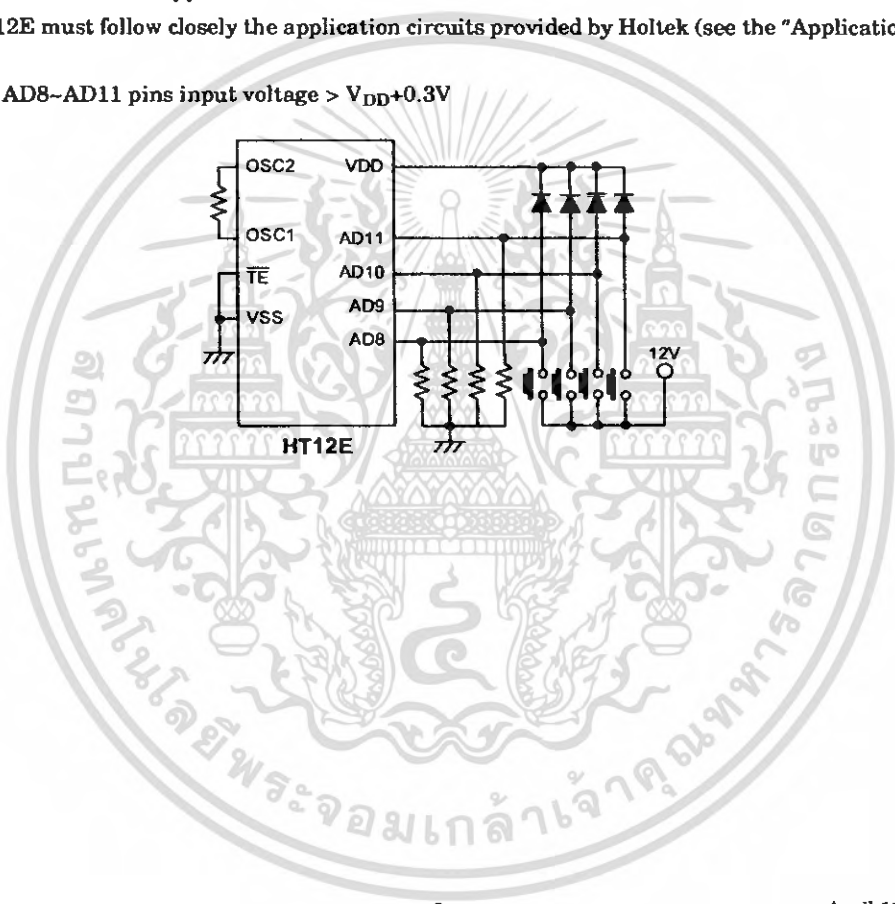
Transmission enable

For the HT12E encoders, transmission is enabled by applying a low signal to the \overline{TE} pin. For the HT12A encoders, transmission is enabled by applying a low signal to one of the data pins D8-D11.

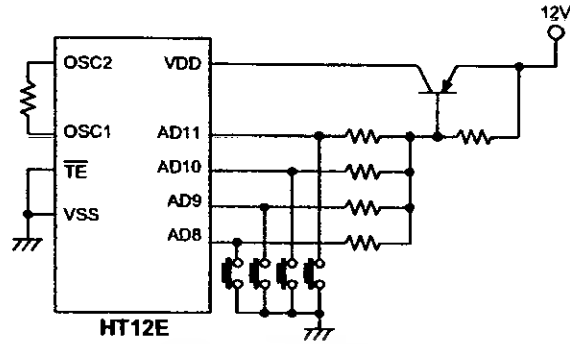
Two erroneous HT12E application circuits

The HT12E must follow closely the application circuits provided by Holtek (see the "Application circuits").

- Error: AD8-AD11 pins input voltage > V_{DD}+0.3V

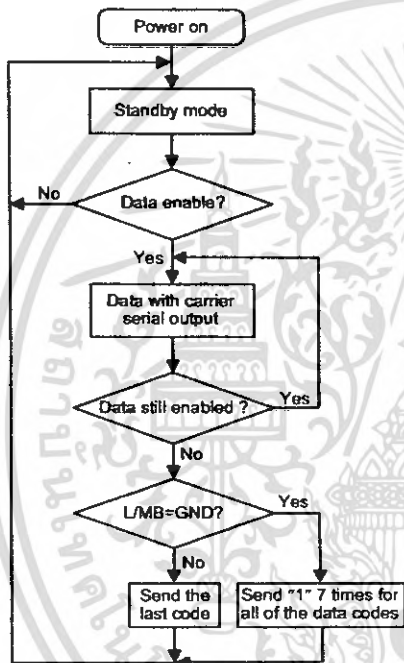


- Error: The IC's power source is activated by pins AD8~AD11

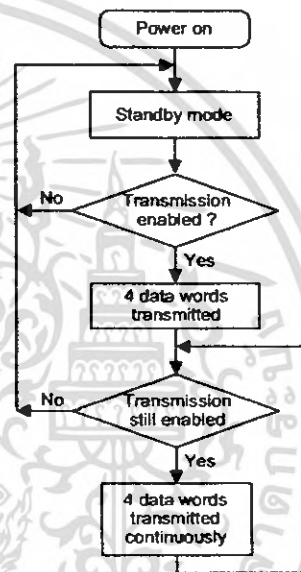


Flowchart

- HT12A



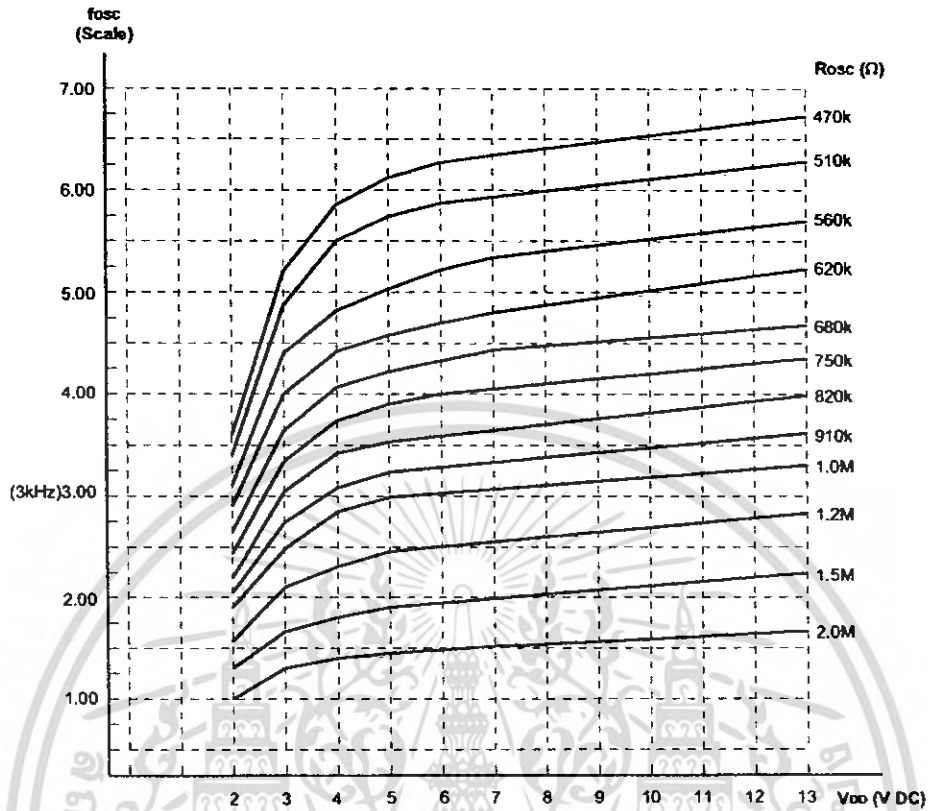
- HT12E



Note: D8~D11 are transmission enables of the HT12A.

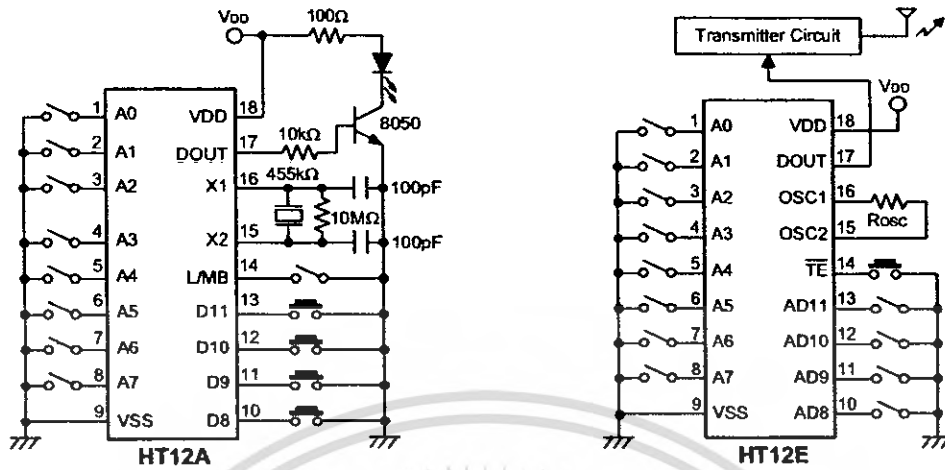
\overline{TE} is the transmission enable of the HT12E.

Oscillator frequency vs supply voltage

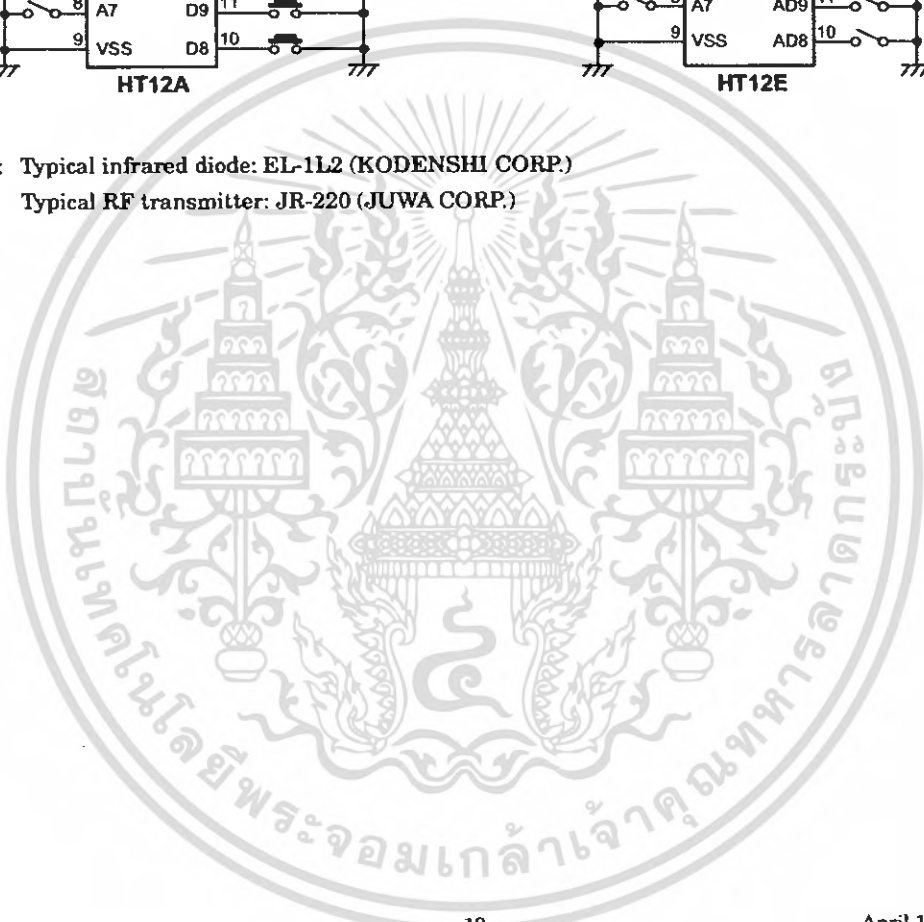


The recommended oscillator frequency is $f_{OSCD} \text{ (decoder)} \cong 50 f_{OSCE} \text{ (HT12E encoder)}$
 $\cong \frac{1}{3} f_{OSCE} \text{ (HT12A encoder)}$

Application Circuits



Note: Typical infrared diode: EL-1L2 (KODENSHI CORP.)
 Typical RF transmitter: JR-220 (JUWA CORP.)



Holtek Semiconductor Inc. (Headquarters)

No.3, Creation Rd. II, Science-based Industrial Park, Hsinchu, Taiwan, R.O.C.
Tel: 886-3-563-1999
Fax: 886-3-563-1189

Holtek Semiconductor Inc. (Taipei Office)

11F, No.576, Sec.7 Chung Hsiao E. Rd., Taipei, Taiwan, R.O.C.
Tel: 886-2-2782-9635
Fax: 886-2-2782-9636
Fax: 886-2-2782-7128 (International sales hotline)

Holtek Semiconductor (Hong Kong) Ltd.

RM. 711, Tower 2, Cheung Sha Wan Plaza, 833 Cheung Sha Wan Rd., Kowloon, Hong Kong
Tel: 852-2-745-8288
Fax: 852-2-742-8657

Holtek Semiconductor (Shanghai) Ltd.

7th Floor, Building 2, No.889, Yi Shan Rd., Shanghai, China
Tel: 021-6485-5560
Fax: 021-6485-0313

Holmate Technology Corp.

48531 Warm Springs Boulevard, Suite 413, Fremont, CA 94539
Tel: 510-252-9880
Fax: 510-252-9885

Copyright © 2000 by HOLTEK SEMICONDUCTOR INC.

The information appearing in this Data Sheet is believed to be accurate at the time of publication. However, Holtek assumes no responsibility arising from the use of the specifications described. The applications mentioned herein are used solely for the purpose of illustration and Holtek makes no warranty or representation that such applications will be suitable without further modification, nor recommends the use of its products for application that may present a risk to human life due to malfunction or otherwise. Holtek reserves the right to alter its products without prior notification. For the most up-to-date information, please visit our web site at <http://www.holtek.com.tw>.

Features

- Operating voltage: 2.4V~12V
- Low power and high noise immunity CMOS technology
- Low standby current
- Capable of decoding 12 bits of information
- Pair with Holtek's 2¹² series of encoders
- Binary address setting
- Received codes are checked 3 times
- Address/Data number combination
 - HT12D: 8 address bits and 4 data bits
 - HT12F: 12 address bits only
- Built-in oscillator needs only 5% resistor
- Valid transmission indicator
- Easy interface with an RF or an infrared transmission medium
- Minimal external components

Applications

- Burglar alarm system
- Smoke and fire alarm system
- Garage door controllers
- Car door controllers
- Car alarm system
- Security system
- Cordless telephones
- Other remote control systems

General Description

The 2¹² decoders are a series of CMOS LSIs for remote control system applications. They are paired with Holtek's 2¹² series of encoders (refer to the encoder/decoder cross reference table). For proper operation, a pair of encoder/decoder with the same number of addresses and data format should be chosen.

The decoders receive serial addresses and data from a programmed 2¹² series of encoders that are transmitted by a carrier using an RF or an IR transmission medium. They compare the serial input data three times continuously with

their local addresses. If no error or unmatched codes are found, the input data codes are decoded and then transferred to the output pins. The VT pin also goes high to indicate a valid transmission.

The 2¹² series of decoders are capable of decoding informations that consist of N bits of address and 12-N bits of data. Of this series, the HT12D is arranged to provide 8 address bits and 4 data bits, and HT12F is used to decode 12 bits of address information.

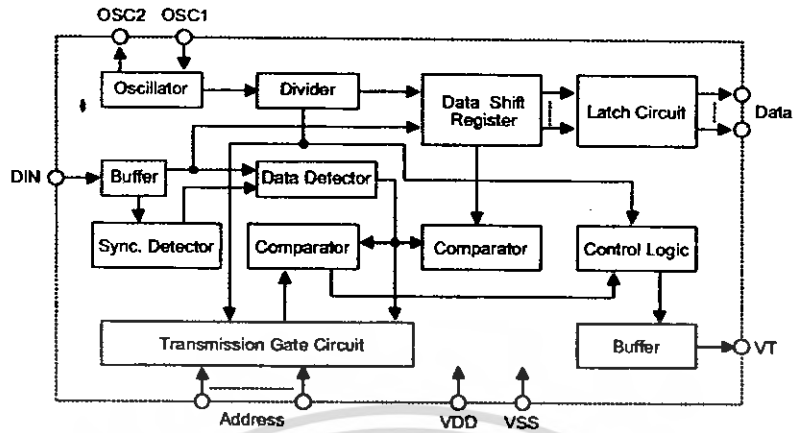
Selection Table

Function Part No.	Address No.	Data		VT	Oscillator	Trigger	Package
		No.	Type				
HT12D	8	4	L	√	RC oscillator	DIN active "Hi"	18 DIP/20 SOP
HT12F	12	0	—	√	RC oscillator	DIN active "Hi"	18 DIP/20 SOP

Notes: Data type: L stands for latch type data output.

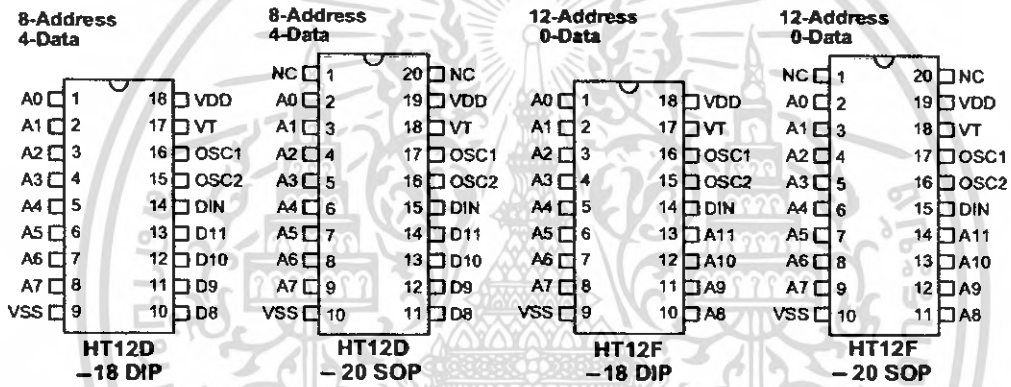
VT can be used as a momentary data output.

Block Diagram



Note: The address/data pins are available in various combinations (see the address/data table).

Pin Assignment

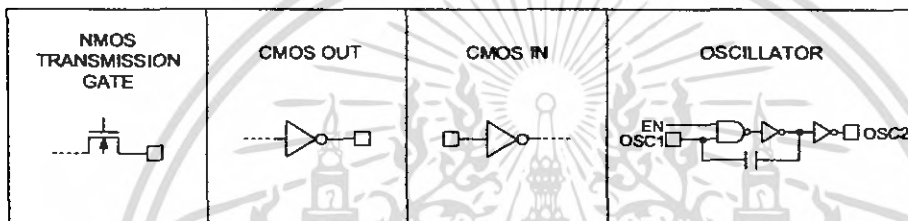


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description

Pin Name	I/O	Internal Connection	Description
A0~A11	I	NMOS TRANSMISSION GATE	Input pins for address A0~A11 setting. They can be externally set to VDD or VSS.
D8~D11	O	CMOS OUT	Output data pins
DIN	I	CMOS IN	Serial data input pin
VT	O	CMOS OUT	Valid transmission, active high
OSC1	I	OSCILLATOR	Oscillator input pin
OSC2	O	OSCILLATOR	Oscillator output pin
VSS	I	—	Negative power supply (GND)
VDD	I	—	Positive power supply

Approximate internal connection circuits



Absolute Maximum Ratings

Supply Voltage.....-0.3V to 13V Storage Temperature.....-50°C to 125°C
 Input Voltage.....V_{SS}-0.3 to V_{DD}+0.3V Operating Temperature-20°C to 75°C

Note: These are stress ratings only. Stresses exceeding the range specified under "Absolute Maximum Ratings" may cause substantial damage to the device. Functional operation of this device at other conditions beyond those listed in the specification is not implied and prolonged exposure to extreme conditions may affect device reliability.

Electrical Characteristics

Ta=25°C

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V _{DD}	Conditions				
V _{DD}	Operating Voltage	—	—	2.4	5	12	V
I _{STB}	Standby Current	5V	Oscillator stops	—	0.1	1	μA
		12V		—	2	4	μA
I _{DD}	Operating Current	5V	No load f _{OSC} =150kHz	—	200	400	μA
I _O	Data Output Source Current (D8~D11)	5V	V _{OH} =4.5V	-1	-1.6	—	mA
	Data Output Sink Current (D8~D11)	5V	V _{OL} =0.5V	1	1.6	—	mA
I _{VT}	VT Output Source Current	5V	V _{OH} =4.5V	-1	-1.6	—	mA
	VT Output Sink Current		V _{OL} =0.5V	1	1.6	—	mA
V _{IH}	"H" Input Voltage	5V	—	3.5	—	5	V
V _{IL}	"L" Input Voltage	5V	—	0	—	1	V
f _{OSC}	Oscillator Frequency	5V	R _{OSC} =51kΩ	—	150	—	kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description

Operation

The 2¹² series of decoders provides various combinations of addresses and data pins in different packages so as to pair with the 2¹² series of encoders.

The decoders receive data that are transmitted by an encoder and interpret the first N bits of code period as addresses and the last 12-N bits as data, where N is the address code number. A signal on the DIN pin activates the oscillator which in turn decodes the incoming address and data. The decoders will then check the received address three times continuously. If the received address codes all match the contents of the decoder's local address, the 12-N bits of data are decoded to activate the output pins and the VT pin is set high to indicate a valid transmission. This will last unless the address code is incorrect or no signal is received.

The output of the VT pin is high only when the transmission is valid. Otherwise it is always low.

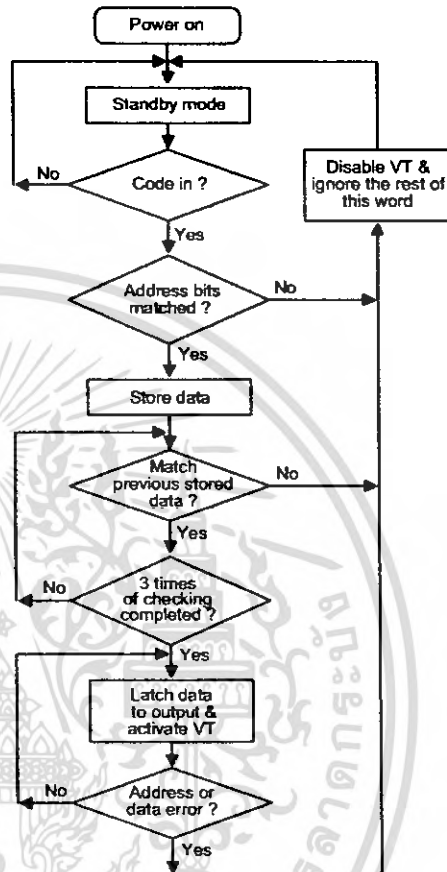
Output type

Of the 2¹² series of decoders, the HT12F has no data output pin but its VT pin can be used as a momentary data output. The HT12D, on the other hand, provides 4 latch type data pins whose data remain unchanged until new data are received.

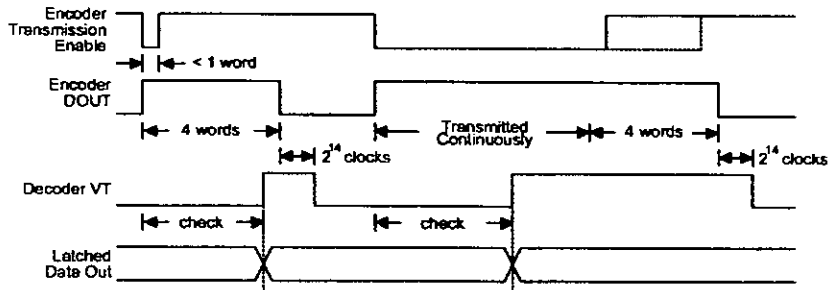
Part No.	Data Pins	Address Pins	Output Type	Operating Voltage
HT12D	4	8	Latch	2.4V~12V
HT12F	0	12	—	2.4V~12V

Flowchart

The oscillator is disabled in the standby state and activated when a logic "high" signal applies to the DIN pin. That is to say, the DIN should be kept low if there is no signal input.



Decoder timing



Encoder/Decoder cross reference table

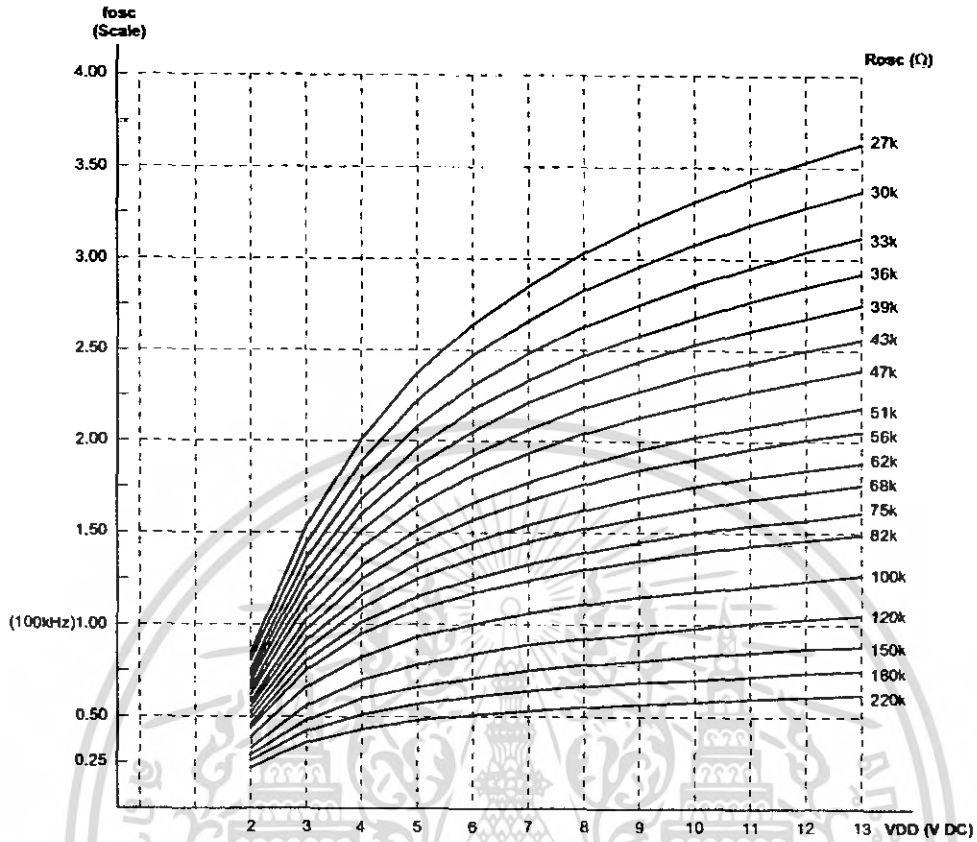
Decoders Part No.	Data Pins	Address Pins	VT	Pair Encoder	Package			
					Encoder		Decoder	
					DIP	SOP	DIP	SOP
HT12D	4	8	√	HT12A	18	20	18	20
				HT12E	18	20		
HT12F	0	12	√	HT12A	18	20	18	20
				HT12E	18	20		

Address/Data sequence

The following table provides address/data sequence for various models of the 2¹² series of decoders. A correct device should be chosen according to the requirements of the individual addresses and data.

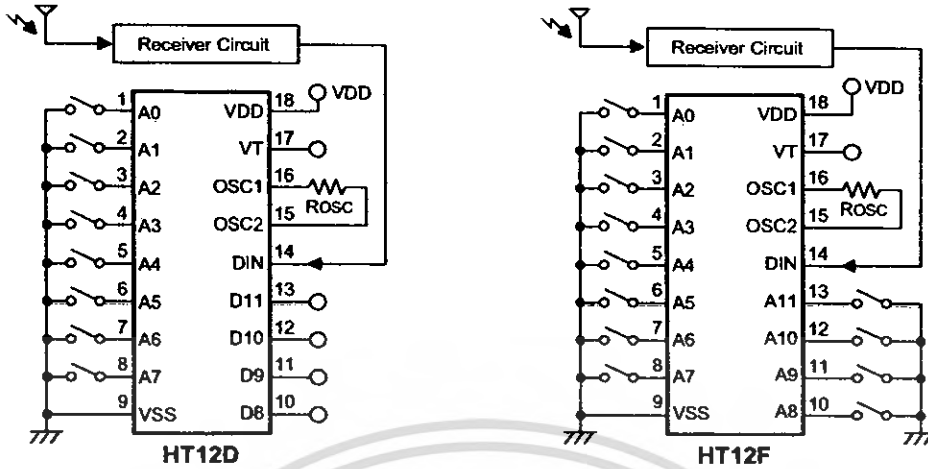
Part No.	Address/Data Bits											
	0	1	2	3	4	5	6	7	8	9	10	11
HT12D	A0	A1	A2	A3	A4	A5	A6	A7	D8	D9	D10	D11
HT12F	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11

Oscillator frequency vs supply voltage



The recommended oscillator frequency is $f_{OSC(D)} \approx 50 f_{OSCE(HT12E)}$
 $\approx \frac{1}{3} f_{OSCE(HT12A)}$

Application Circuits



Notes: Typical infrared receiver: PIC-12043T/PIC-12043S (KODESHI CORP.)
or LTM9052 (LITEON CORP.)

Typical RF receiver: JR-200 (JUWA CORP.)
RE-99 (MING MICROSYSTEM, U.S.A.)



Holtek Semiconductor Inc. (Headquarters)
No.3 Creation Rd. II, Science-based Industrial Park, Hsinchu, Taiwan, R.O.C.
Tel: 886-3-563-1999
Fax: 886-3-563-1189

Holtek Semiconductor Inc. (Taipei Office)
11F, No.576, Sec.7 Chung Hsiao E. Rd., Taipei, Taiwan, R.O.C.
Tel: 886-2-2782-9635
Fax: 886-2-2782-9636
Fax: 886-2-2782-7128 (International sales hotline)

Holtek Semiconductor (Hong Kong) Ltd.
RM.711, Tower 2, Cheung Sha Wan Plaza, 833 Cheung Sha Wan Rd., Kowloon, Hong Kong
Tel: 852-2-745-8288
Fax: 852-2-742-8657

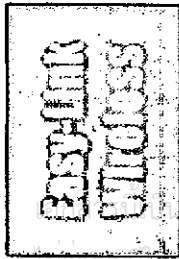
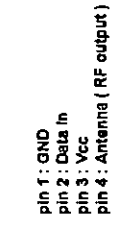
Holtek Semiconductor (Shanghai) Ltd.
7th Floor, Building 2, No.889, Yi Shan Road, Shanghai, China
Tel:021-6485-5560
Fax:021-6485-0313

Holmate Technology Corp.
48631 Warm Springs Boulevard, Suite 413, Fremont, CA 94539
Tel: 510-252-9860
Fax: 510-252-9885

Copyright © 1999 by HOLTEK SEMICONDUCTOR INC.

The information appearing in this Data Sheet is believed to be accurate at the time of publication. However, Holtek assumes no responsibility arising from the use of the specifications described. The applications mentioned herein are used solely for the purpose of illustration and Holtek makes no warranty or representation that such applications will be suitable without further modification, nor recommends the use of its products for application that may present a risk to human life due to malfunction or otherwise. Holtek reserves the right to alter its products without prior notification. For the most up-to-date information, please visit our web site at <http://www.holtek.com.tw>.

TLP434A Ultra Small Transmitter



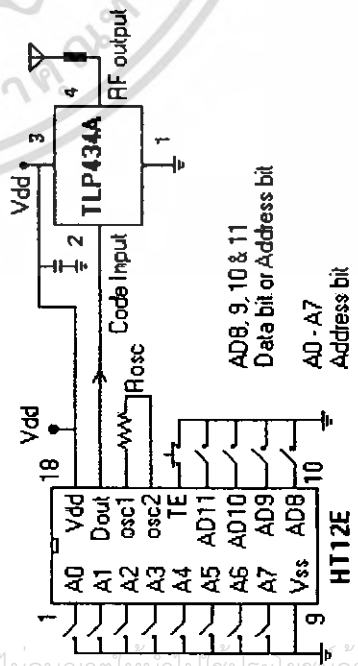
Modulation : ASK
Operation Voltage : 2 - 12 VDC

Frequency 315, 418 and 433.92 Mhz

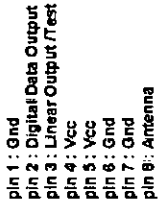
Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Vcc	Operating supply voltage		2.0	-	12.0	V
Icc1	Peak Current (2V)		-	-	1.64	mA
Icc2	Peak Current (12V)		-	-	19.4	mA
Vh	Input High Voltage	Idata = 100uA (High)	Vcc-0.5	Vcc	Vcc+0.5	V
Vi	Input Low Voltage	Idata = 0 uA (Low)	-	-	0.3	V
FO	Absolute Frequency	315Mhz module	314.8	315	315.2	MHz
PO	RF Output Power- 50ohm	Vcc = 9V-12V	-	16	-	dBm
		Vcc = 5V-6V	-	14	-	dBm
DR	Data Rate	External Encoding	512	4.8K	200K	bps

Notes : (Case Temperature = 25°C +/- 2°C , Test Load Impedance = 50 ohm)

Application Circuit :
Typical Key-chain Transmitter using HT112E-18DIP, a Binary 12 bit Encoder from Holtek Semiconductor Inc.



RLP434A SAW Based Receiver



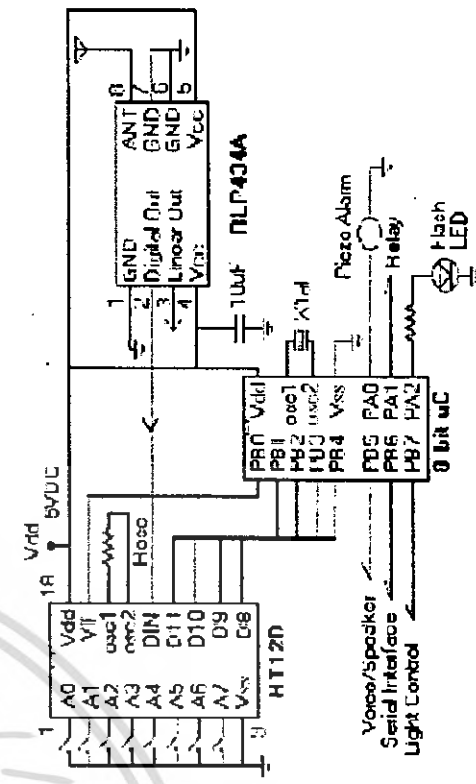
Modulation : ASK
Supply Voltage : 3.3 - 6.0 VDC
Output : Digital & Linear

Frequency 315, 418 and 433.92 Mhz

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Vcc	Operating supply voltage		3.3	5.0V	6.0	V
Icc	Operating Current		-	4.5	-	mA
Vdata	Data Out	Idata = +200 uA (High)	Vcc-0.5	-	Vcc	V
		Idata = -10 uA (Low)	-	-	0.3	V

Electrical Characteristics					
Characteristics	SYM	Min	Typ	Max	Unit
Operation Radio Frequency	FC	315, 418 and 433.92			MHz
Sensitivity	Pref	-110			dBm
Channel Width		+500			Khz
Noise Equivalent BW		4			Khz
Receiver Turn On Time		5			ms
Operation Temperature	Top	-20		80	C
Baseboard Data Rate		4.8			KHz

Application Circuit :
Typical RF Receiver using HT12D-18DIP, a Binary 12 bit Decoder with 8 bit uC HT48RXX from Holtek Semiconductor Inc.



Push-Pull Four Channel Driver

FEATURES

- Output Current 1A Per Channel (600mA for L293D)
- Peak Output Current 2A Per Channel (1.2A for L293D)
- Inhibit Facility
- High Noise Immunity
- Separate Logic Supply
- Over-Temperature Protection

DESCRIPTION

The L293 and L293D are quad push-pull drivers capable of delivering output currents to 1A or 600mA per channel respectively. Each channel is controlled by a TTL-compatible logic input and each pair of drivers (a full bridge) is equipped with an inhibit input which turns off all four transistors. A separate supply input is provided for the logic so that it may be run off a lower voltage to reduce dissipation.

Additionally the L293D includes the output clamping diodes within the IC for complete interfacing with inductive loads.

Both devices are available in 16-pin Batwing DIP packages. They are also available in Power S01C and Hermetic DIL packages.

TRUTH TABLE

V_i (each channel)	V_{INH}^*	V_o
H	H	H
L	H	L
H	L	X**
L	L	X**

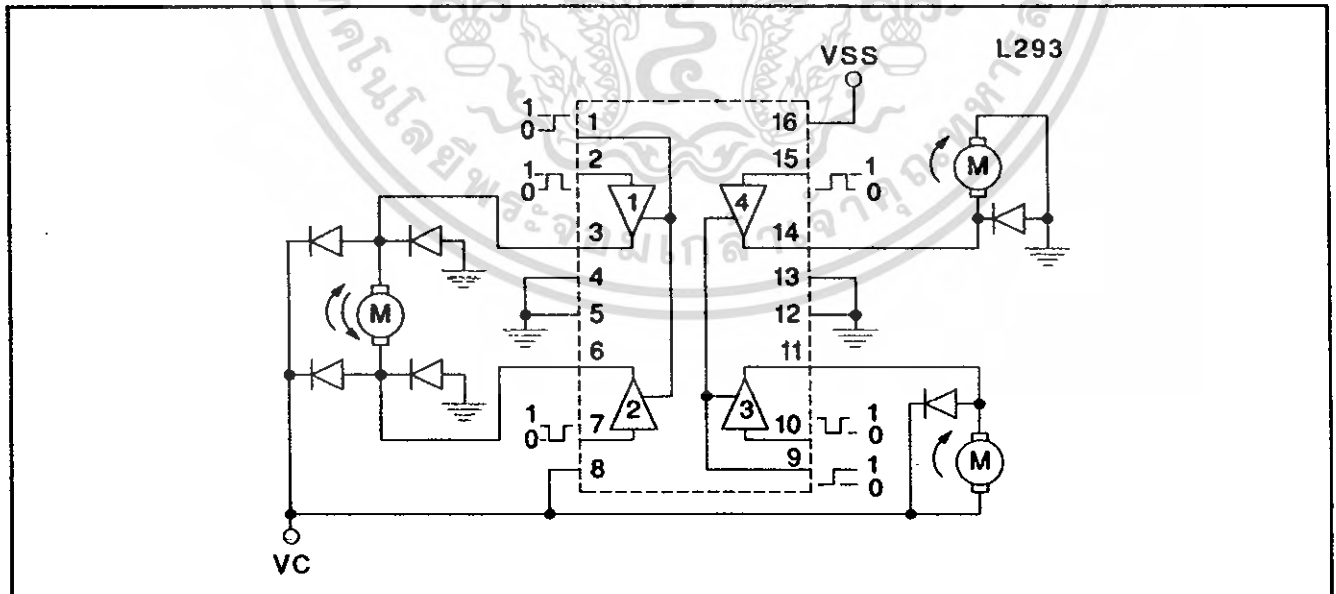
*Relative to the considered channel
**High output impedance

ABSOLUTE MAXIMUM RATINGS

Collector Supply Voltage, V_c	36V
Logic Supply Voltage, V_{SS}	36V
Input Voltage, V_i	7V
Inhibit Voltage, V_{INH}	7V
Peak Output Current (Non-Repetitive), I_{out} (L293).....	2A
I_{out} (L293D).....	1.2A
Total Power Dissipation at $T_{ground-pins} = 80^\circ\text{C}$, N Batwing pkg, (Note).....	5W
Storage and Junction Temperature, T_{stg}, T_j	-40 to +150°C

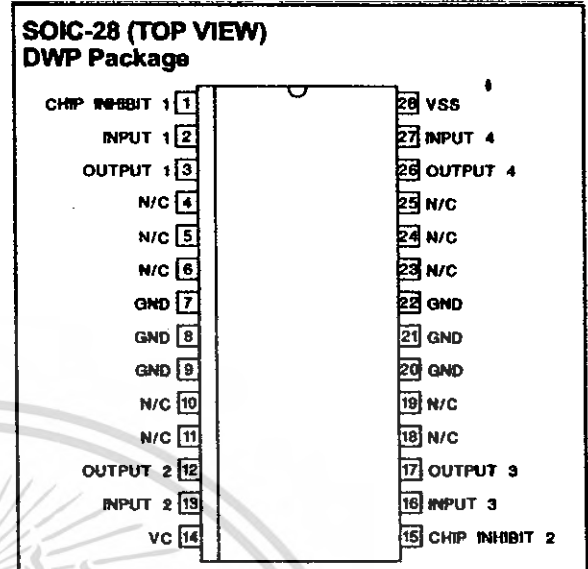
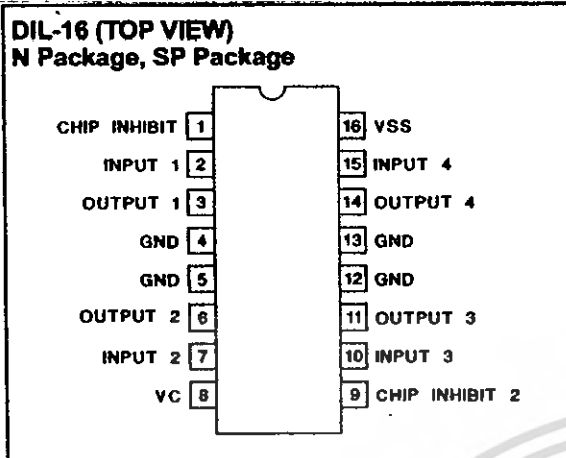
Note: Consult packaging section of Databook for thermal limitations and considerations of packages.

BLOCK DIAGRAM



Note: Output diodes are internal in L293D.

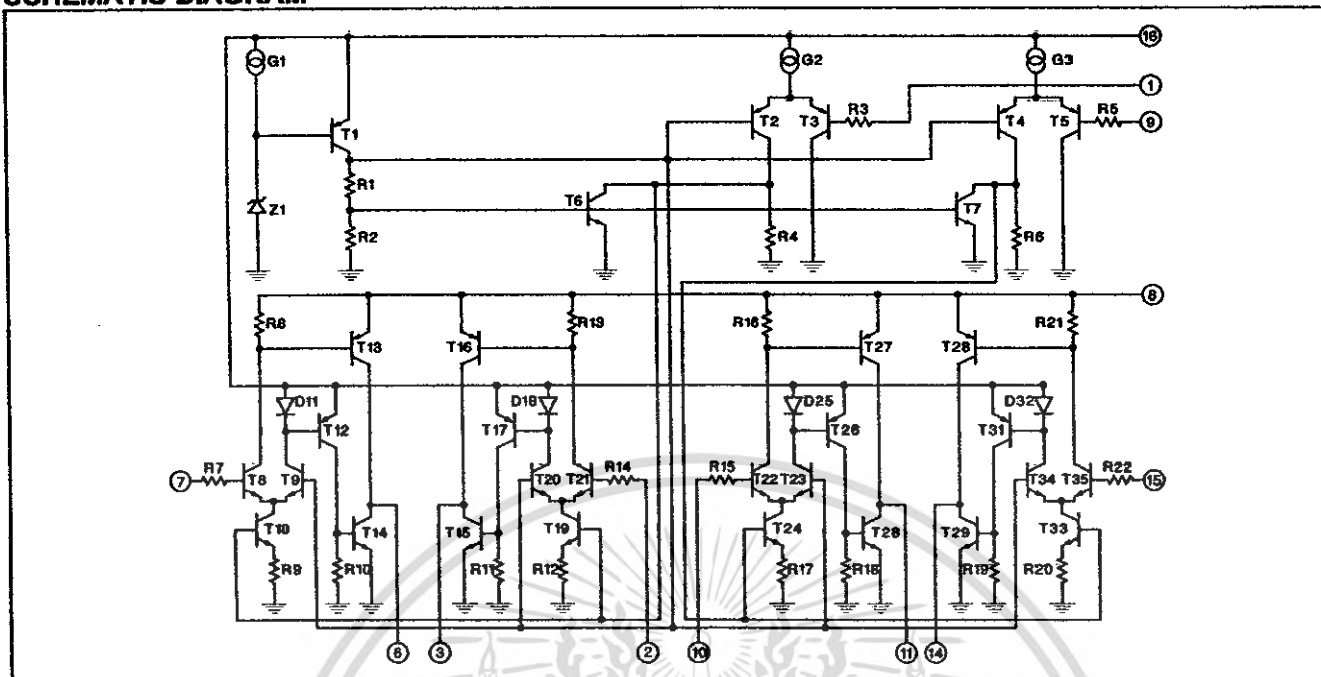
CONNECTION DIAGRAMS



ELECTRICAL CHARACTERISTICS: (For each channel, $V_C = 24V$, $V_{SS} = 5V$, $T_{AMB} = 25^\circ C$, unless otherwise specified; $T_A = T_J$)

PARAMETER	SYMBOL	TEST CONDITION	MIN.	TYP.	MAX.	UNITS
Collector Supply Voltage	V_C				36	V
Logic Supply Voltage	V_{SS}		4.5		36	V
Collector Supply Current	I_C	$V_I = L, I_O = 0, V_{INH} = H$		2	6	mA
		$V_I = H, I_O = 0, V_{INH} = H$		16	24	mA
		$V_{INH} = L$			4	mA
Total Quiescent Logic Supply Current	I_{SS}	$V_I = L, I_O = 0, V_{INH} = H$		44	60	mA
		$V_I = H, I_O = 0, V_{INH} = H$		16	22	mA
		$V_{INH} = L$		16	24	mA
Input Low Voltage	V_{IL}		-0.3		1.5	V
Input High Voltage	V_{IH}	$V_{SS} \leq 7V$	2.3		V_{SS}	V
		$V_{SS} \geq 7V$	2.3		7	V
Low Voltage Input Current	I_{IL}	$V_I = 0V$			-10	μA
High Voltage Input Current	I_{IH}	$V_I = 4.5V$		30	100	μA
Inhibit Low Voltage	$V_{INH, L}$		-0.3		1.5	V
Inhibit High Voltage	$V_{INH, H}$	$V_{SS} \leq 7V$	2.3		V_{SS}	V
		$V_{SS} > 7V$	2.3		7	V
Low Voltage Inhibit Current	$V_{INH, L}$			-30	-100	μA
High Voltage Inhibit Current	$V_{INH, H}$				10	μA
Source Output Saturation Voltage	V_{CEsatH}	$I_O = -1A$ (-0.6A for L293D)		1.4	1.8	V
Sink Output Saturation Voltage	V_{CEsatL}	$I_O = 1A$ (0.6A for L293D)		1.2	1.8	V
Clamp Diode Forward Voltage (L293D only)	V_F	$I_F = 0.6A$		1.3		V
Rise Time	T_R	0.1 to 0.9 V_O (See Figure 1)		100		ns
Fall Time	T_F	0.9 to 0.1 V_O (See Figure 1)		350		ns
Turn-on Delay	T_{ON}	0.5 V_I to 0.5 V_O (See Figure 1)		750		ns
Turn-off Delay	T_{OFF}	0.5 V_I to 0.5 V_O (See Figure 1)		200		ns

SCHEMATIC DIAGRAM



APPLICATION INFORMATION

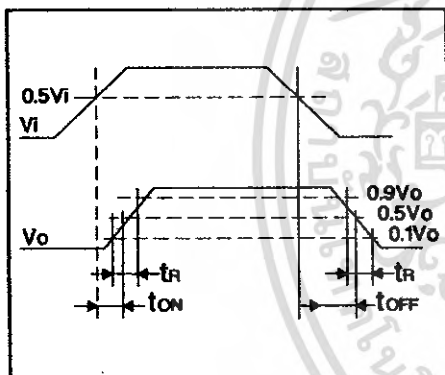


Figure 1: Switching Times

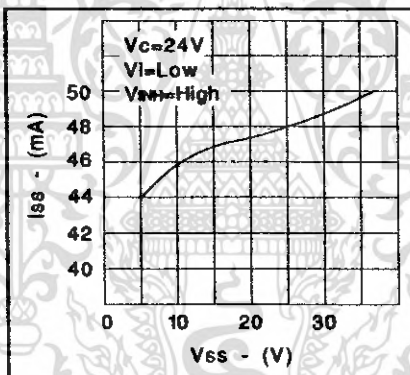


Figure 2: Quiescent Logic Supply Current vs Logic Supply Voltage

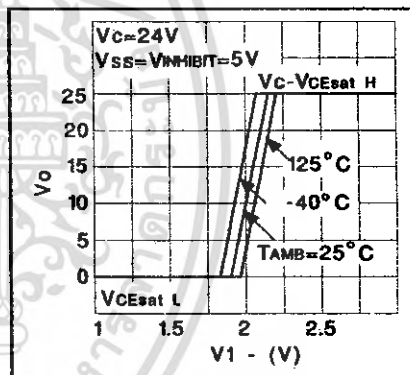


Figure 3: Output Voltage vs Input Voltage

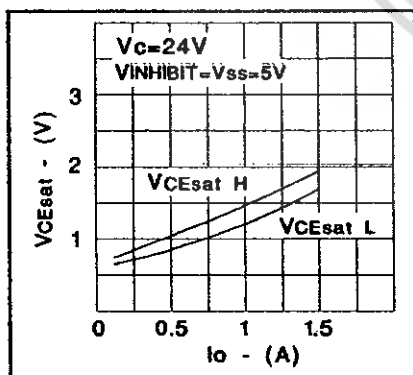


Figure 4: L293 Saturation vs Output Current

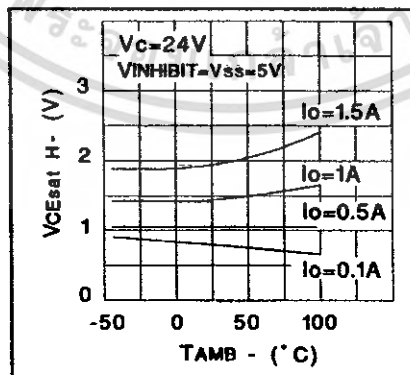


Figure 5: L293 Source Saturation vs Ambient Temperature

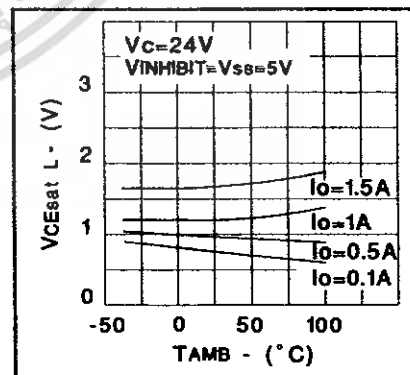


Figure 6: L293 Sink Saturation Voltage vs Ambient Temperature

NOTE: For L293D curves, multiply output current by 0.6.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPLICATION INFORMATION (Cont.)

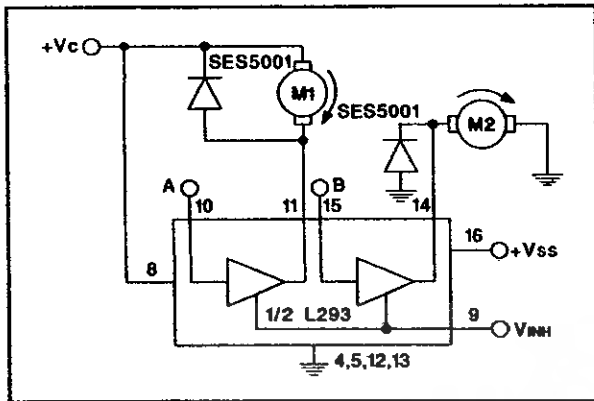


Figure 7: DC Motor Controls (with Connection to Ground and to Supply Voltage)

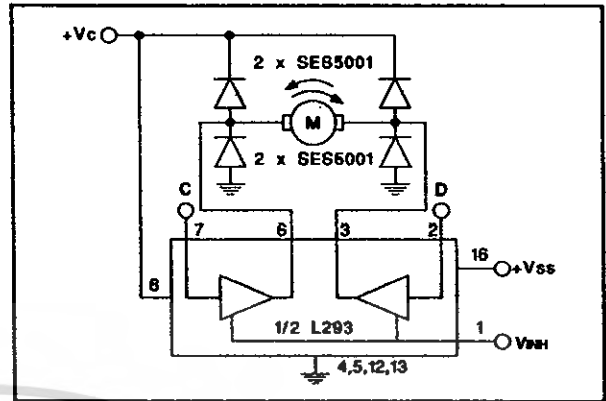


Figure 8: Bidirectional DC Motor Control

V _{INH}	A	M1	B	M2
H	H	Fast Motor Stop	H	Run
H	L	Run	L	Fast Motor Stop
L	X	Free Running Motor Stop	X	Free Running Motor Stop

L = Low H = High X = Don't Care

INPUTS		FUNCTION
V _{INH} = H	C = H; D = L	Turn Right
	C = L; D = H	Turn Left
	C = D	Fast Motor Stop
V _{INH} = L	C = X; D = X	Free Running Motor Stop

L = Low H = High X = Don't Care

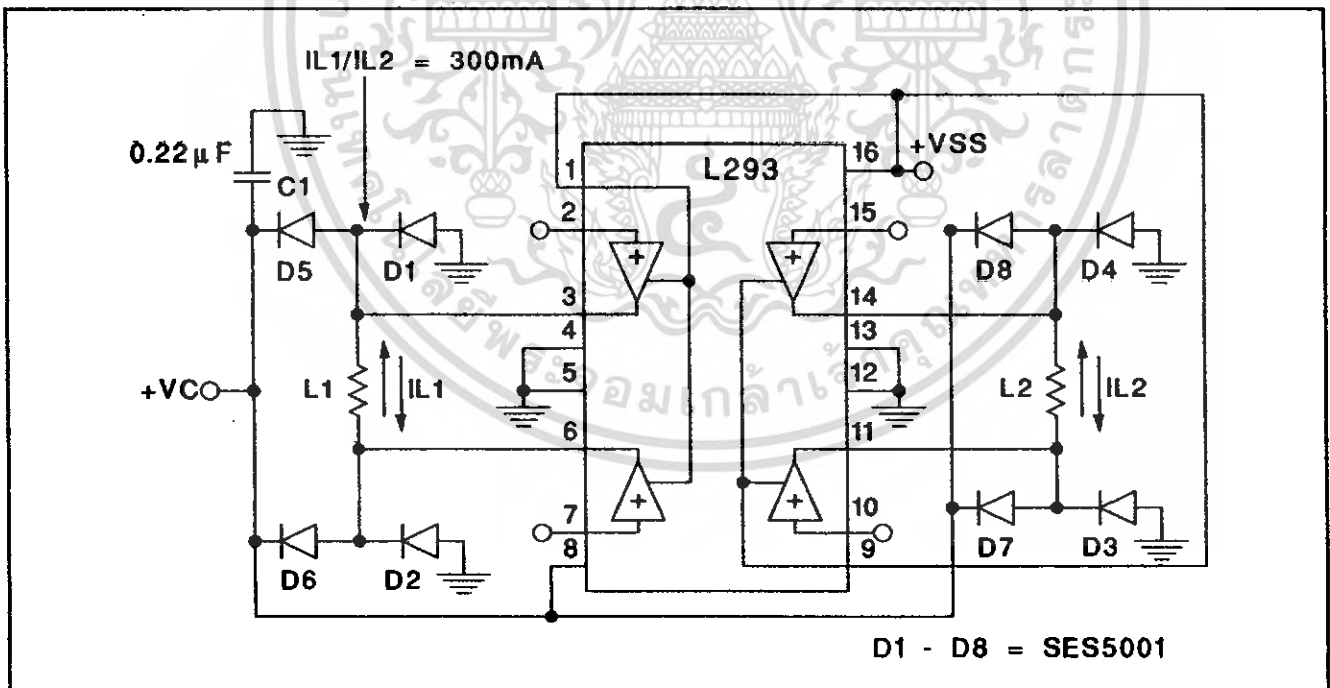


Figure 9: Bipolar Stepping Motor Control