สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบกันขโมยผ่านโทรศัพท์เคลื่อนที่

A SECURITY SYSTEM THROUGH A CELLULAR

นางสาวอรุณี สงกรินทร์

นายอารยะ สูงส่งคุณธรรม

เลขหมู่.....
เลขทะเบียน.....72015
วัน,เดือน,ปี.....7 มิ.ย. 2550

.b.....11761812.....
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A SECURITY SYSTEM THROUGH A CELLULAR

ARUNEE SONGKARIN

ARAYA SOONGSONGKUNNATHUM

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2006

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์ ระบบกันขโมยผ่านโทรศัพท์เคลื่อนที่

A SECURITY SYSTEM THROUGH A CELLULAR

นักศึกษาผู้จัดทำ นางสาวอรุณี สงครินทร์ รหัสนักศึกษา 46010943

นายอารยะ สูงส่งคุณธรรม รหัสนักศึกษา 46010972

ปริญญา วิศวกรรมศาสตรบัณฑิต

สาขาวิชา วิศวกรรมการวัดคุม

ปีการศึกษา 2549

อาจารย์ผู้ควบคุมปริญญานิพนธ์	ลายมือชื่อ
ผศ. เชื้อ นกอยู่ ดร. พงษ์ชัย นิลาศ	

ภาควิชารับรองแล้ว

(รศ. ประภาษ อุดคกิมาพันธุ์)

หัวหน้าภาควิชาวิศวกรรมการวัดคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบกันขโมยผ่านโทรศัพท์เคลื่อนที่			
	A SECURITY SYSTEM THROUGH A CELLULAR			
นักศึกษาผู้จัดทำ	นางสาวอรุณี	สงครินทร์	รหัสนักศึกษา	46010943
	นายอารยะ	สูงส่งคุณธรรม	รหัสนักศึกษา	46010972
อาจารย์ที่ปรึกษา	ผศ. เชื้อ	นกอยู่		
	ดร. พงษ์ชัย	นิลาศ		
ปีการศึกษา	2549			

บทคัดย่อ

โครงงานนี้ ได้จัดนำเสนอระบบกันขโมยผ่านโทรศัพท์มือถือ ที่มีการแจ้งเตือนโดยการส่งข้อความสั้น (SMS) ไปยังโทรศัพท์มือถือของเจ้าของรถยนต์ พร้อมทั้งยังสามารถทำการตรวจสอบการทำงานของระบบโดยการส่ง SMS กลับมาที่เครื่องกันขโมย ระบบกันขโมยนี้ประกอบด้วย 2 ส่วน คือ ส่วนของตัวเซนเซอร์ที่ใช้ตรวจจับสัญญาณที่เกิดขึ้นกับรถยนต์ เช่น การขูดรถ การเปิดประตูรถ การทุบกระจก ด้วย Microphone และส่วนประมวลผลควบคุมที่เชื่อมต่อเข้ากับโทรศัพท์มือถือเพื่อทำการรับ-ส่ง SMS โดยใช้ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	A Security System Through A Cellular	
Authors	Miss Arunee	Songkarin
	Mr. Araya	Soongsongkunnathum
Thesis Advisor	Asst.Prof. Chuae	Nokyoo
	Dr. Phongchai	Nilas
Year	2006	

ABSTRACT

This project presents the security system through a cellular which alerts the car owner by sending a short message service (SMS) via cellular. The user can also check the performance of the system by sending a monitoring command through SMS back to the security device. This project consists of 2 parts; one is a sensor that uses a microphone as the sensor to detect any signals occurring at the car such as scratching, opening doors, breaking windows, etc. The other part is a microprocessor MCS-51 which is connected to the cellular for SMS transceiver.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ สำเร็จลุล่วงได้เป็นอย่างดี เพราะได้รับความเมตตา กรุณา และการสนับสนุนจาก ผศ. เชื้อ นกอยู่ และดร. พงษ์ชัย นิลาศ ที่ได้ให้คำแนะนำเกี่ยวกับการดำเนิน และการศึกษาแก่ผู้วิจัยตลอดเวลา

นอกจากนี้ต้องขอขอบคุณพี่ๆ เพื่อนๆ ทุกคนที่ได้ให้การช่วยเหลือในการทำปริญญานิพนธ์ฉบับนี้จนสำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายนี้ต้องขอขอบพระคุณ คุณพ่อ คุณแม่ และบุคคลในครอบครัวที่ได้ให้การสนับสนุนและคอยให้กำลังใจเสมอมา คุณความดีจากการทำปริญญานิพนธ์นี้ ผู้วิจัยขอมอบให้ แด่ผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญรูป.....VI

สารบัญตาราง.....VII

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและแนวความคิด.....1

1.2 วัตถุประสงค์ของปริญญานิพนธ์.....2

1.3 ขอบเขตของปริญญานิพนธ์.....2

1.4 ขั้นตอนการศึกษา.....2

บทที่ 2 ทฤษฎี.....3

2.1 พื้นฐานระบบเซลลูลาร์.....3

2.2 หลักการจัดแบ่งเซลล์ในระบบโทรศัพท์เซลลูลาร์.....4

2.3 การใช้งานระบบ GSM.....7

2.4 ลักษณะของการส่งข้อความสั้น.....8

2.5 การรับ – ส่งข้อความสั้นแบบโหมดพีดียู (PDU-Mode).....9

2.6 คำสั่ง AT Command กับมือถือ.....16

2.7 ไมโครคอนโทรลเลอร์ ตระกูล MCS-51.....24

2.7.1 คุณสมบัติของไมโครคอนโทรลเลอร์ ตระกูล MCS-51.....24

2.7.2 คุณสมบัติ AT90S2313.....25

2.7.3 การจัดขาไมโครคอนโทรลเลอร์ AT90S2313.....26

2.7.4 การรีเซตและการอินเตอร์รัพท์.....27

2.8 IC MAX 232.....28

2.8.1 การเชื่อมต่อกับพอร์ตอนุกรมของคอมพิวเตอร์.....28

2.8.2 การสื่อสารพอร์ตอนุกรม RS-232.....30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.8.3 มาตรฐาน RS-232.....30

2.9 รีเลย์ (Relay).....33

บทที่ 3 การสร้างและการออกแบบวงจร.....34

3.1 การออกแบบวงจร.....34

3.2 วงจรจ่ายแรงดันไฟฟ้ากระแสตรง.....34

3.3 ส่วนควบคุมการทำงาน.....35

3.4 วงจรในส่วนแปลงค่าแรงดันไปเป็นสัญญาณดิจิตอล.....36

3.5 ส่วนเชื่อมต่อกับโทรศัพท์เคลื่อนที่.....37

3.6 แผนผังการทำงาน.....38

บทที่ 4 โปรแกรมที่ใช้ควบคุมการทำงาน.....40

บทที่ 5 สรุปผลการทดลอง.....53

บรรณานุกรม.....55

ภาคผนวก.....56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ **หน้า**

2.1 โครงสร้างของเครือข่ายระบบโทรศัพท์เซลลูลาร์ 4

2.2 การแบ่งบริเวณพื้นที่ออกเป็นรูปสามเหลี่ยม สี่เหลี่ยม และหกเหลี่ยม.....5

2.3 การจัดเซลล์มีความถี่แตกต่างกันในแต่ละช่องสัญญาณ.....6

2.4 ลักษณะและตำแหน่งการจัดเซลล์ใช้ความถี่ซ้ำๆของขนาด K = 7.....7

2.5 แสดงลักษณะขั้นตอนของการส่งข้อความสั้น.....8

2.6 แสดงการจัดขาไมโครคอนโทรลเลอร์ AT90S2313.....26

2.7 การจัดขาของ MAX232 หรือ ICL232.....28

2.8 โครงสร้างภายในของ MAX232 หรือ ICL232.....29

2.9 แสดงวงจรเชื่อมต่อ MAX232 หรือ ICL232.....29

3.1 วงจรจ่ายแรงดันไฟฟ้ากระแสตรง.....34

3.2 วงจรที่จะนำไปใช้กับวงจรที่ออกแบบ.....35

3.3 วงจรส่วนควบคุมการทำงาน.....36

3.4 วงจรในส่วนแปลงค่าแรงดันไปเป็นสัญญาณดิจิตอล.....36

3.5 วงจรส่วนเชื่อมต่อกับโทรศัพท์เคลื่อนที่.....37

3.6 วงจรเสร็จสมบูรณ์.....37

3.7 แผนผังการทำงานของส่วนตรวจสอบสถานะมือถือและ Cell site.....38

3.8 รูปแผนผังการทำงานของส่วนควบคุมการส่งข้อความสั้น.....39

5.1 สัญญาณขณะที่ไม่มี input.....53

5.2 สัญญาณขณะเปิดประตู.....54

5.3 สัญญาณขณะมีการขูดใต้บังโคลนรถยนต์.....54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

2.1 แสดงส่วนประกอบของชุดข้อมูลในการส่งข้อความสั้นแบบโหมดพีดียู.....9

2.2 ส่วนประกอบของข้อมูลที่ส่ง.....12

2.3 ส่วนประกอบของสตริงการรับข้อมูลสั้น.....13

2.4 แสดงวิธีการแปลงตัวอักษรชนิด 7 บิต เป็นข้อมูล 8 บิต ข้อความ “hellohello”.....15

2.5 ชุดของตัวแปรของมาตรฐาน GSM 03.38.....15

2.6 ลักษณะชุดคำสั่งของ AT+CNMI.....17

2.7 ลักษณะชุดคำสั่งของ AT+CSCB.....18

2.8 ลักษณะชุดคำสั่งของ AT+CMGF.....18

2.9 ลักษณะชุดคำสั่งของ AT+CSCA.....19

2.10 ลักษณะชุดคำสั่งของ AT+CMGL.....19

2.11 ลักษณะชุดคำสั่งของ AT+CMGR.....20

2.12 ลักษณะชุดคำสั่งของAT+CMGS.....20

2.13 ลักษณะชุดคำสั่งของ AT+CMSS.....21

2.14 ลักษณะชุดคำสั่งของ AT+CMGW.....21

2.15 ลักษณะชุดคำสั่งของ AT+CMGD.....22

2.16 ลักษณะชุดคำสั่งของ AT+CSMS.....22

2.17 ลักษณะชุดคำสั่งของ AT+CPMS.....23

2.18 ลักษณะชุดคำสั่งของ AT+CMGC.....24

2.19 แสดงอินเตอร์รัพท์เวกเตอร์.....27

2.20 รายละเอียดการต่อคอนเน็กเตอร์แบบ DB9 มาตรฐาน RS-232.....30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและแนวความคิด

ทุกวันนี้มีการใช้โทรศัพท์มือถือกันอย่างแพร่หลาย และมีจำนวนการใช้โทรศัพท์มือถือเพิ่มขึ้นอย่างมากสาเหตุสำคัญที่ทำให้ได้รับความนิยมคงหนีไม่พ้นเรื่องความสะดวกสบายในการใช้งานจากการใช้เทคโนโลยีที่สูงขึ้น และราคาถูกลงกว่าแต่ก่อนมาก และผู้ให้บริการยังมีการแข่งขันกันอย่างดุเดือด ทั้งแข่งขันในเรื่องโปรโมชั่น แล้วยังแข่งขันกัน ในเรื่องพื้นที่ของการให้บริการควบคุมพื้นที่มากกว่ากัจึงไม่แปลกที่ชาวเขาที่อยู่ดอยใช้โทรศัพท์มือถือกันเป็นเรื่องปกติธรรมดามาก ดังนั้นผลดีจึงเกิดขึ้นกับผู้บริโภคอย่างเรา การใช้ประโยชน์จากเครือข่ายระบบโทรศัพท์มือถือ นอกจากจะใช้พูดคุยกันแล้วยังสามารถใช้ในการรับส่งข้อมูลได้อีกด้วย ไม่ว่าจะเป็นการใช้งานอินเทอร์เน็ตผ่านระบบ GPRS (General Packet Radio Service) การรับส่งแฟกซ์ การใช้งานแทนโมเด็ม นอกจากนี้ยังใช้ส่งข้อความสั้น (Short Message Service: SMS) ซึ่งเป็นส่วนสำคัญที่เรานำมาใช้ในโครงงานนี้

ปัจจุบัน โทรศัพท์มือถือเป็นปัจจัยหนึ่งในการดำรงชีวิตประจำวันของเราไปแล้ว ซึ่งระบบโทรศัพท์มือถือต่างๆได้มีการพัฒนารูปแบบใหม่ๆขึ้นมาอย่างรวดเร็ว ทำให้โทรศัพท์มือถือรุ่นเก่าๆถูกขายต่อในตลาดมือ 2 ซึ่งได้ราคาไม่สูงมากนักซึ่งถ้าเรานำโทรศัพท์มือถือรุ่นเก่าๆที่ไม่ได้ใช้งานนี้มาประยุกต์ใช้งานเป็น “ระบบกันขโมยผ่านโทรศัพท์มือถือ” จะทำให้เปลี่ยนจากโทรศัพท์มือถือรุ่นเก่าๆที่เราคิดว่าไม่ค่อยมีประโยชน์แล้วนำกลับมาประยุต์ใช้ในการนำมารักษาความปลอดภัยของยานพาหนะเรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อศึกษาและออกแบบวงจรของระบบกันขโมย
2. เพื่อศึกษาการทำงานของระบบโทรศัพท์มือถือ
3. เพื่อการเขียนโปรแกรมด้วย Microcontroller
4. สามารถนำระบบกันขโมยมาใช้งานได้จริง

1.3 ขอบเขตของปริญญานิพนธ์

1. ออกแบบวงจร
2. เขียนโปรแกรมด้วย Microcontroller
3. เชื่อมต่อวงจรเข้ากับโทรศัพท์มือถือ
4. สามารถส่ง SMS เมื่อมีสัญญาณเข้ามา

1.4 ขั้นตอนการศึกษา

1. ศึกษาการทำงานของวงจร
2. ศึกษาการทำงานของตัว Microcontroller
3. ศึกษาการเขียนโปรแกรม Microcontroller4.
4. ศึกษาการวงจรทำงานของโทรศัพท์มือถือ
5. ศึกษาการเชื่อมต่อวงจรให้เข้ากับโทรศัพท์มือถือ
6. ศึกษาการทำงานระบบกันขโมยแบบส่ง SMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี

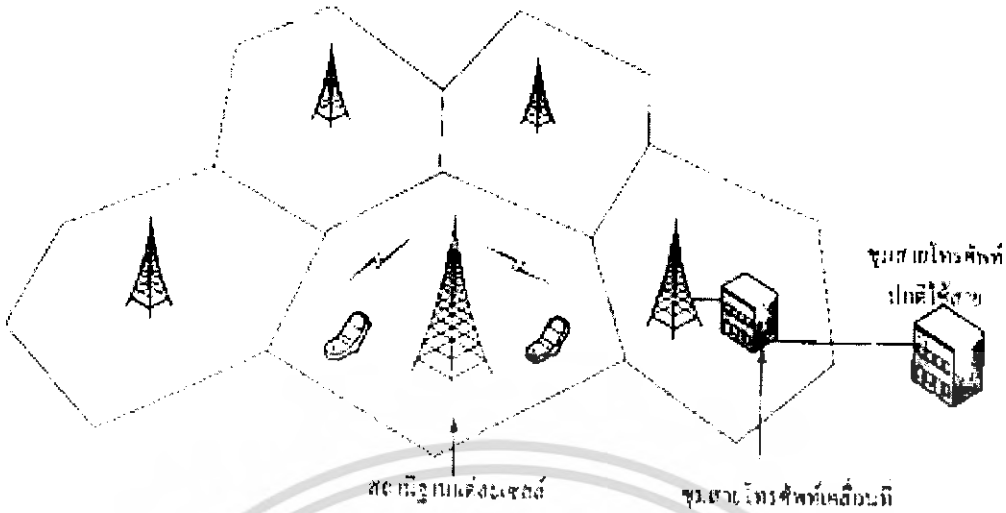
2.1 พื้นฐานระบบเซลลูลาร์

ระบบโทรศัพท์มือถือ คือ ระบบโทรศัพท์แบบหนึ่งซึ่งมีลักษณะแตกต่างออกไปจากระบบโทรศัพท์ธรรมดา คือการส่งสัญญาณเสียงระหว่างผู้รับกับผู้ส่งไม่จำเป็นต้องใช้สายโทรศัพท์แต่การส่งสัญญาณจะอยู่ในรูปของคลื่นวิทยุความถี่สูง ซึ่งจะมีสถานีเครือข่ายทำหน้าที่ รับ และส่งสัญญาณวิทยุไปยังผู้รับ และผู้ส่งโดยสถานีเครือข่าย สามารถส่งสัญญาณครอบคลุมพื้นที่บริเวณหนึ่งเท่านั้น ซึ่งพื้นที่ ที่สถานีเครือข่ายแต่ละสถานีครอบคลุม เรียกว่า เซลล์ (Cell) โดยในเซลล์ เซลล์หนึ่ง จะมีรัศมีครอบคลุม ประมาณ 2.5 - 10 กิโลเมตรในกรณีที่ต้องการให้ครอบคลุมพื้นที่กว้างกว่านี้ จะต้องตั้งสถานีรับส่งสัญญาณให้ครอบคลุมในเขตพื้นที่ที่ต้องการ โดยย่านความถี่ที่ใช้มีหลายย่านความถี่ และจะมีช่องสัญญาณต่างกัน ขึ้นอยู่กับผู้ให้บริการ โดยปัจจุบันย่านความถี่ที่ใช้งานจะถูกนำกลับมาใช้ใหม่ ซึ่งระบบนี้เรียกว่า ระบบเซลลูลาร์ (Cellular Mobile Telephone) ปัจจุบันได้ใช้งานระบบนี้กันอย่างแพร่หลาย

โทรศัพท์เซลลูลาร์ เป็นระบบโทรศัพท์เคลื่อนที่ระบบใหม่ที่ถูกพัฒนาขึ้นมาเพื่อที่จะทำให้เหมาะสมกับการใช้งานในระบบสื่อสารวิทยุโทรศัพท์เคลื่อนที่ในปัจจุบันและยังสามารถครอบคลุมพื้นที่ให้บริการกว้างมากขึ้น ขยายขอบเขตการให้บริการแก่ผู้ใช้บริการ ได้อย่างต่อเนื่องไม่มีขีดจำกัด กล่าวคือในเขตพื้นที่ที่มีประชากรหนาแน่นเช่น กรุงเทพมหานครมีความต้องการใช้โทรศัพท์เคลื่อนที่เป็นจำนวนมาก ก็จำเป็นต้องออกแบบให้มีจำนวนเซลล์เพิ่มมากขึ้น เพื่อที่จะรองรับอัตราใช้บริการหรือทราฟฟิก (Traffic) ที่เพิ่มขึ้นส่วนในเขตพื้นที่ที่มีประชากรเบาบาง เช่น ในต่างจังหวัด กรุงเทพมหานคร มีความต้องการในการใช้โทรศัพท์เคลื่อนที่ เป็นจำนวนน้อย ก็ต้องออกแบบเซลล์ให้มีขนาดใหญ่ขึ้น แต่ละเซลล์ที่ติดกันจะใช้ย่านความถี่ที่แตกต่างกัน เพื่อป้องกันการรบกวนซึ่งกันและกัน ส่วนเซลล์ที่อยู่ห่างออกไปจะนำความถี่เดิมมาใช้ซ้ำอีก เรียกว่า การนำความถี่กลับมาใช้อีก (Frequency Reuse) โดยไม่เกิดการสอดแทรก และสูญหายของสัญญาณ เมื่อต้องการใช้โทรศัพท์เคลื่อนที่เพิ่มขึ้นก็แบ่งจำนวนเซลล์ออกมาเป็นเซลล์ย่อยๆ (Cell Splitting) ให้มากขึ้นได้ตามต้องการ ทำให้การทำงานของระบบโทรศัพท์เซลลูลาร์สามารถนำความถี่มาใช้งานได้อย่างคุ้มค่า

ระบบโทรศัพท์เซลลูลาร์ ใช้หลักการติดตั้งเครื่องรับส่งวิทยุ (Transceiver) ที่กำลังส่งต่ำมากๆ จำนวนมากกระจายเป็นจุดๆ ไปทั่วพื้นที่ให้บริการ จุดที่ติดตั้งเครื่องรับส่งวิทยุเหล่านี้ เรียกว่า สถานีฐาน (Base Station) หรือ อาจเรียกว่า ที่ตั้งเซลล์ (Cell Site) ซึ่งทำหน้าที่เป็นศูนย์กลางเซลล์ ลักษณะโครงสร้างของระบบโทรศัพท์เซลลูลาร์ แสดงดังรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 โครงสร้างของเครือข่ายระบบโทรศัพท์เซลลูลาร์

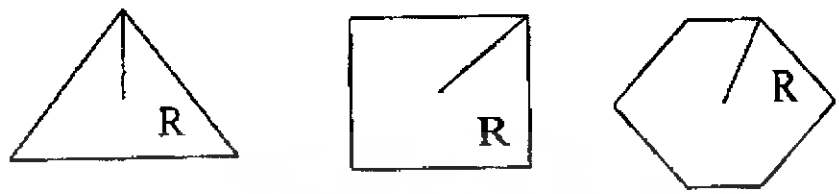
จากรูปที่ 2.1 แสดงโครงสร้างของเครือข่ายระบบโทรศัพท์เซลลูลาร์แต่ละเซลล์จะเชื่อมต่อถึงกัน ซึ่งแต่ละเซลล์มีรัศมีทำการกว้าง หรือแคบขึ้น ขึ้นอยู่กับความหนาแน่นของการใช้โทรศัพท์เซลลูลาร์ ถ้าการใช้โทรศัพท์เซลลูลาร์มีความหนาแน่นมากขึ้นจำนวนเซลล์จะเพิ่มขึ้นรัศมีของแต่ละเซลล์จะมีขนาดเล็กลง และถ้าการใช้โทรศัพท์เซลลูลาร์มีความหนาแน่นน้อย จำนวนเซลล์ลดลง รัศมีของแต่ละเซลล์ จะใหญ่ขึ้น มาตรฐานทั่วไปที่นิยมใช้ ขนาดของเซลล์มีรัศมี ตั้งแต่ 250 เมตร ถึง 30 กิโลเมตร แต่ละเซลล์ที่ถูกแบ่งออกจะต้องเพิ่มสถานีฐานเข้าไปทุกเซลล์ เครื่องรับส่งแต่ละเซลล์สามารถทำงานได้ที่กำลังส่งซึ่งครอบคลุมอยู่ในเฉพาะพื้นที่เล็กๆของเซลล์ตัวเองเท่านั้น ทำให้เครื่องรับส่งที่ใช้งานมีขนาดเล็กลง

2.2 หลักการจัดแบ่งเซลล์ในระบบโทรศัพท์เซลลูลาร์

จุดมุ่งหมายหลักของการแบ่งพื้นที่ให้บริการออกเป็นเซลล์ย่อยๆก็เพื่อสามารถใช้ประโยชน์จากสเปกตรัมที่มีอยู่อย่างจำกัดให้ได้ประสิทธิภาพมากที่สุด ลักษณะการแบ่งพื้นที่นั้นสามารถทำได้หลายรูปแบบ เช่น การแบ่งเป็นพื้นที่รูปสามเหลี่ยม สี่เหลี่ยม และหกเหลี่ยม ดังรูปที่ 2.2 พิจารณาจากรูปจะเห็นว่าการแบ่งบริเวณพื้นที่ออกเป็นรูปหกเหลี่ยมเหมือนรังผึ้ง สามารถครอบคลุมบริเวณได้กว้างกว่าการแบ่งบริเวณพื้นที่ออกเป็นรูปสามเหลี่ยมหรือสี่เหลี่ยม เมื่อกำหนด R ให้มีขนาดเท่ากันซึ่งหมายความว่าสำหรับขนาดพื้นที่ให้บริการที่เท่ากันเราจะใช้จำนวนอุปกรณ์รับส่งสัญญาณที่น้อยลงได้ นอกจากนี้ รูปหกเหลี่ยมยังมีความคล้ายคลึง กับรูปวงกลมมากที่สุด ซึ่งสอดคล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใกล้เคียงกับลักษณะการแพร่กระจายของคลื่นสัญญาณมากขึ้นและที่สำคัญที่สุด คือ รูปหกเหลี่ยม จะมีปัญญาของสัญญาณรบกวนจากเซลล์รอบข้างน้อยกว่ารูปแบบอื่น



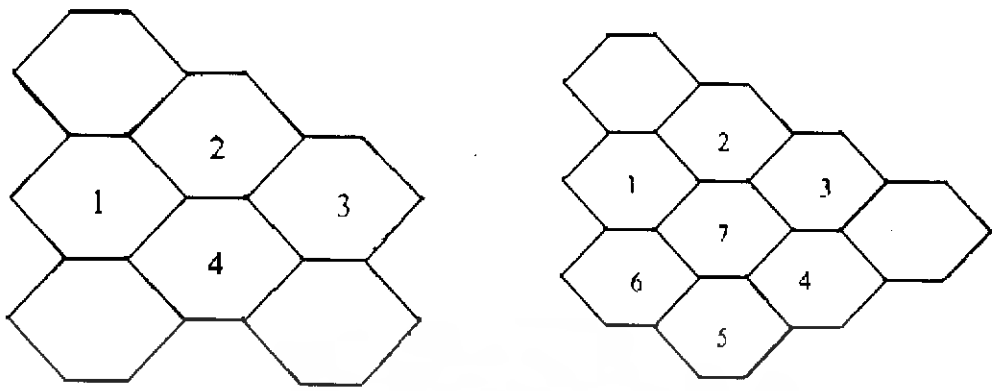
ขนาดพื้นที่ $1.3\ R^2$ $2.0\ R^2$ $2.6\ R^2$

รูปที่ 2.2 การแบ่งบริเวณพื้นที่ออกเป็นรูปสามเหลี่ยม สี่เหลี่ยม และหกเหลี่ยม

ระบบโทรศัพท์เซลลูลาร์นั้น ต้องนำความถี่ที่ใช้แล้วนำกลับมาใช้อีกในเซลล์ต่างๆที่อยู่ห่างกันออกไป จำนวนช่องสัญญาณมากน้อยขึ้นอยู่กับจำนวนแถบคลื่นความถี่ที่จัดสรรให้โดยองค์กรที่รับผิดชอบและช่วงห่างของช่องสัญญาณที่เป็นมาตรฐานใช่กันอยู่เป็นเครือข่ายช่องสัญญาณในกลุ่มเซลล์ที่ติดกัน จะต้องใช้ความถี่ที่แตกต่างกันระบบการทำงานของโทรศัพท์เซลลูลาร์ต้องระมัดระวังการสอดแทรกช่องสัญญาณที่ใช้ความถี่เดียวกันในบริเวณใกล้เคียง (Co-Channel Interference) ต้องให้อยู่ในขีดจำกัดที่ยอมรับได้ตามมาตรฐานการสื่อสารในระบบเซลลูลาร์

ในการนำความถี่เดิมมาใช้ใหม่จำเป็นต้องเลือกจำนวนช่องเซลล์สัญญาณที่แตกต่างกัน (K) ควรให้มีมากที่สุดเท่าที่จะจัดได้ เท่าที่จำเป็นจะต้องไม่ให้เกิดการทับกัน หรือเกิดช่องว่างระหว่างเซลล์ จำนวนช่องเซลล์สัญญาณที่แตกต่าง (K) ที่ยอมรับได้และนิยมใช้งานคือ K = 4, 7, 12 และ 19 เซลล์ เป็นต้น ลักษณะการจัดวางเซลล์สัญญาณที่แตกต่างกันแสดงดังรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) ขนาด K = 4 เซลล์ (ข) ขนาด K = 7 เซลล์

รูปที่ 2.3 การจัดเซลล์มีความถี่แตกต่างกันในแต่ละช่องสัญญาณ

การใช้งานของโทรศัพท์เซลลูลาร์นิยมใช้ค่า K น้อยค่า เพราะค่า K น้อยค่านั้นทำให้จำนวนช่องสัญญาณสื่อสาร ของแต่ละเซลล์มีจำนวนมากขึ้น สามารถให้บริการผู้ใช้โทรศัพท์เซลลูลาร์ภายในเซลล์มากหมายเลขแต่ถ้าค่า K เพิ่มขึ้นทำให้จำนวนช่องสัญญาณสื่อสารของแต่ละเซลล์มีจำนวนน้อยลง ให้บริการผู้ใช้โทรศัพท์เซลลูลาร์ภายในเซลล์น้อยหมายเลขลง

ระยะห่างของการใช้เซลล์มีความถี่ซ้ำ (D) หาได้จากค่ารัศมีของเซลล์ที่ใช้ และค่าจำนวนช่องเซลล์สัญญาณที่แตกต่างกัน (K) หาโดยเขียนเป็นสมการได้ ดังนี้

$$D = R \times \sqrt{3K}$$

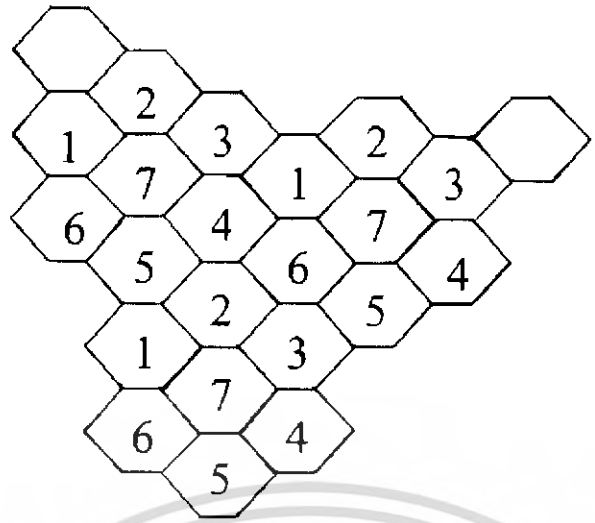
เมื่อ D = ระยะห่างของการใช้เซลล์มีความถี่ซ้ำ

R = รัศมีของเซลล์

K = จำนวนช่องสัญญาณที่แตกต่างกัน

ค่า D ของระบบจะมากขึ้นหรือน้องลงขึ้นอยู่กับค่า K ที่ใช้งานอยู่ค่า K มีค่าน้อยระยะห่างของ D น้องด้วย และถ้าค่า K มากขึ้นทำให้ค่า D มากขึ้นตามไปด้วย ลักษณะและตำแหน่งการจัดเซลล์ใช้ความถี่ซ้ำๆ แสดงดังรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 ลักษณะและตำแหน่งการจัดเซลล์ใช้ความถี่ซ้ำๆของขนาด K = 7

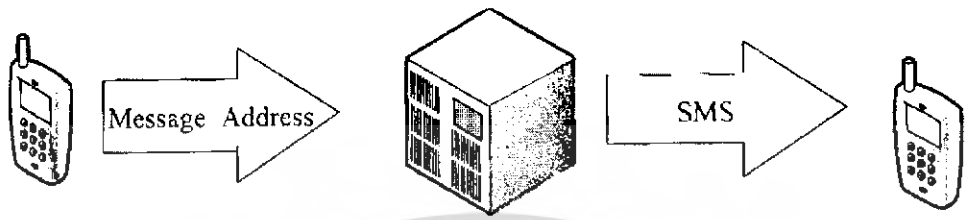
จากรูปที่ 2.4 ลักษณะและตำแหน่งการจัดเซลล์ ใช้ความถี่ซ้ำๆของขนาดเซลล์สัญญาณที่แตกต่างกัน 7 เซลล์ (K = 7) จะเห็นได้ว่ากลุ่มเซลล์ที่มีความถี่แตกต่างกัน ถูกจับกลุ่มให้มีเซลล์ติดต่อกันการใช้เซลล์ความถี่ซ้ำกลุ่มต่อไปจำเป็นต้องวางตำแหน่งเซลล์ความถี่ซ้ำ ให้มีระยะห่างจากเซลล์เดิมเป็นระยะที่เท่าๆกันอย่างสม่ำเสมอ เพื่อเป็นการป้องกันการสอดแทรกของช่องสัญญาณที่ใช้ความถี่เดียวกันการวางตำแหน่งกลุ่มเซลล์สามารถจัดได้แบ่งออกเป็นกลุ่มๆ

2.3 การใช้งานระบบ GSM

ระบบ GSM นั้นแตกต่างจากระบบอื่นที่ย่านความถี่ต่ำ 900 MHz และวิธีการแบ่งช่องสัญญาณแบบ TDMA และระบบ GSM ได้นำระบบรักษาความปลอดภัยแบบ Subscriber Identification Module (SIM) โดยระบบนี้ เป็นการเก็บความจำเกี่ยวกับหมายเลข หมายเลขเครื่อง รหัสลับ และตัวอักษรอื่นที่ต้องการเก็บลงในซิมการ์ด (SIM Card) ข้อดีของระบบ SIM คือ การใช้งานเราสามารถพกพาซิมการ์ดของเราไปใช้งานเครื่องโทรศัพท์มือถือเครื่องใดก็ได้ ที่อยู่ในระบบเดียวกัน และเมื่อเครื่องโทรศัพท์มือถือเสียหาย หรือชำรุดก็สามารถเปลี่ยนซิมการ์ดมาไว้เครื่องใหม่ โดยข้อมูลเดิมยังอยู่ครบ จากข้อดีของระบบ SIM นี้ทำให้ระบบGSM มีการใช้งานที่แตกต่างจากระบบอื่น เช่น มีการใช้งานการส่งข้อความสั้น คือข้อความถึงผู้รับแน่นอนถึงแม้ว่าผู้รับจะปิดเครื่องแต่ข้อความจะแสดงเมื่อเปิดเครื่อง เนื่องจากการทำงานของการส่งข้อความสั้น จะเป็นดังนี้ เริ่มต้นเมื่อเราเขียนข้อความและกดส่งจากโทรศัพท์มือถือ ข้อความ เบอร์โทรศัพท์มือถือ ที่เราต้องการส่งให้ เบอร์โทรศัพท์มือถือของผู้รับจะถูกส่งผ่านเครือข่ายที่ใช้เชื่อมต่อระหว่าง ผู้ส่ง กับผู้รับ หรือที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกว่า ศูนย์บริการส่งข้อความสั้น (Short Message Service Center : SMSC) หลังจากนั้นศูนย์บริการส่งข้อความสั้นจะคิดค่าใช้จ่าย และส่งข้อความไปยังเบอร์โทรศัพท์มือถือที่เราต้องการส่งโดยแสดงรูปแบบได้ตามรูปที่ 2.5



รูปที่ 2.5 แสดงลักษณะขั้นตอนของการส่งข้อความสั้น

จากความสามารถของระบบ GSM ที่สามารถส่งข้อความสั้นไปยังผู้รับได้นี้เอง เราจึงเลือกที่จะใช้ระบบ GSM เป็นเครือข่ายสำหรับส่งข้อความแจ้งภัย โดยจะส่งข้อความแจ้งภัยเป็นข้อความสั้น

2.4 ลักษณะของการส่งข้อความสั้น

การส่งข้อความสั้น (Short Message Service) คือการส่งข้อความสั้นๆหรือข้อมูลสั้นจากเครื่องโทรศัพท์มือถือผู้ส่ง ไปยังเครื่องโทรศัพท์มือถือของผู้รับ โดยส่งผ่านเครือข่ายศูนย์บริการ (Short Message Service Center : SMSC) โดยการส่งแบบข้อความสั้นนี้เราจะสามารถเลือกได้ว่าจะส่งข้อความสั้น หรือรูปภาพโลโก้ เสียงเพลงริงค์โทน ซึ่งจะมีวิธีการส่งที่แตกต่างกัน 2 แบบ คือ โหมดตัวอักษร (Text-Mode) และโหมดพีดียู (Protocol Data Unit : PDU) โดยโหมดตัวอักษร คือโหมดที่เราสามารถส่งข้อความสั้นๆประมาณ 160 ตัวอักษรไปยังเครื่องโทรศัพท์มือถือของผู้รับโดยลักษณะข้อความจะอยู่ในรูปแบบ รหัสแอสกี้ (ASCII) ส่วนโหมดพีดียู คือโหมดที่สามารถส่งได้ทั้งข้อความสั้นๆ ส่งรูปภาพ และเพลงริงค์โทนได้ ซึ่งโหมดพีดียู (PDU) นี้จะมีรูปแบบการวางข้อมูลที่จะส่งแตกต่างกับโหมดตัวอักษร คือโหมดพีดียูจะมีการเข้ารหัสที่จะแปลงข้อความเป็นรูปแบบของเลขฐานสิบหก และต้องมีการส่งหัวข้อของชุดข้อมูล (Heading) แต่ในโหมดตัวอักษรจะเป็นการส่งแบบรหัสแอสกี้ และไม่จำเป็นต้องส่งหัวข้อของชุดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การรับ – ส่งข้อความสั้นแบบโหมดพีดียู (PDU-Mode)

ในโครงงานนี้เราจะใช้การส่งข้อความสั้นแบบโหมดพีดียู ซึ่งรูปแบบการจัดรูปแบบนั้นจะซับซ้อนกว่าแบบโหมดอักษรมาก แต่การแบบโหมดพีดียูนี้ เราสามารถใช้ได้กับโทรศัพท์มือถือได้ทุกรุ่น โดยการส่งข้อความสั้นแบบโหมดพีดียูมีรายละเอียดดังนี้คือ ในนี้จะต้องมีการสร้างหัวข้อของชุดข้อมูลสำหรับส่ง ซึ่งประกอบด้วยส่วนของศูนย์บริการข้อความสั้น กับส่วนของชุดข้อความ (Transfer Protocol Unit: TPDU) โดยทั้งสองส่วนจะมีลักษณะเป็นเลขฐานสิบหกซึ่งจะวางลำดับตามนี้

ตารางที่ 2.1 แสดงส่วนประกอบของชุดข้อมูลในการส่งข้อความสั้นแบบโหมดพีดียู

หัวข้อของชุดข้อมูล (Heading : Cr)	ส่วนของศูนย์บริการข้อความสั้น	ส่วนของชุดข้อความ (TPDU)	บิดหยุด (Stop bit : Ctrl-Z)

ในส่วนของชุดข้อความก็จะประกอบด้วยส่วนย่อยๆซึ่งจะเป็นตัวกำหนดรูปแบบของการส่งข้อความสั้นที่จะส่ง โดยถ้าต้องการที่จะส่งเป็นข้อความจะต้องจัดรูปแบบเรียงตามนี้

1. โปรโตคอลพารามิเตอร์ คือ พารามิเตอร์ที่บอกว่าโปรโตคอล (Protocol) ที่สามารถใช้ส่งเป็นแบบใด กรณีส่งแบบ TPDU = 0x01

2. ตัวเลขอ้างข้อความ ในกรณี ที่มีข้อความหลายๆ ข้อความ เราสามารถ จัดลำดับข้อความโดยใช้ตัวเลขอ้างข้อความได้ (มีค่าปกติ = 0x00)

3. ความยาวของเบอร์โทรศัพท์มือถือของหมายเลขปลายทาง

4. รูปแบบของเบอร์โทรศัพท์มือถือ ของหมายเลขปลายทาง ซึ่งจะเป็นตัวบอกลักษณะของเบอร์โทรศัพท์มือถือที่เราต้องการส่งข้อความไปให้ โดยส่งแบบสากลจะใช้ค่าเป็น 0x91

5. หมายเลขโทรศัพท์มือถือของหมายเลขปลายทางที่ต้องการจะส่ง โดยหมายเลขโทรศัพท์นี้จะมีการเข้ารหัสแบบสลับ (Nibble Swapped)

6. ตัวแสดงรูปแบบชุดข้อมูล

7. ลักษณะการเข้ารหัสข้อมูล คือ พารามิเตอร์ที่บอกว่าเราจะส่งเป็นภาษาใด (มาตรฐาน คือ ระบบ GSM)

8. ความยาวของข้อความที่ต้องการส่ง (ก่อนเข้ารหัส)

9. ข้อความที่ต้องการส่ง (หลังเข้ารหัส)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่าการส่งแบบโหมดพีดียูมีการเข้ารหัสที่ซับซ้อน เช่น การเข้ารหัสสลับ และการเข้ารหัสของชุดข้อความที่จะส่ง โดยการเข้ารหัสแบบสลับมีลักษณะดังนี้ โดยจะทำการสลับเบอร์โทรศัพท์ที่ติดกันเป็นคู่ๆและถ้าเหลือเศษจะเติมค่า F เข้าไปก่อนรหัสตัวสุดท้าย เช่น เบอร์โทรศัพท์คือ 123456789 เมื่อเข้ารหัสสลับจะกลายเป็น 21436587F9 ส่วนการเข้ารหัสของชุดข้อความจะต้องทำการแปลงข้อความที่เป็นแอสกี้ มาเป็นเลขฐานสองหลักจากนั้นก็ทำการเข้ารหัส

ส่วนของศูนย์บริการข้อความสั้น จะเป็นส่วนที่กำหนดเครือข่ายการใช้บริการข้อความว่าใช้บริการผ่านศูนย์บริการส่วน ของศูนย์บริการข้อความสั้นใดๆ โดยประกอบด้วยส่วนย่อยๆ ดังนี้ คือ

1. ความยาวของเบอร์ศูนย์บริการ
2. รูปแบบของเบอร์ศูนย์บริการ (ส่งแบบสากลจะใช้ค่า = 0x91)
3. เบอร์ศูนย์บริการ โดยจะมีการเข้ารหัสแบบสลับ

เมื่อผู้รับได้รับข้อความสั้น(SMS) ที่มีการส่งแบบโหมดพีดียูรูปแบบของข้อความก็จะอยู่ในลักษณะของโหมดพีดียูเราจำเป็นต้องศึกษาถึงรูปแบบของข้อความที่ได้รับดังนี้ คือ ข้อความที่ได้รับนี้จะประกอบด้วยส่วนสำคัญสองส่วน คือ ส่วนของศูนย์บริการข้อความสั้นกับส่วนของชุดข้อความ โดยทั้งสองส่วนจะมีลักษณะเป็นเลขฐานสิบหกซึ่งจะเหมือนกับการส่ง แต่ชุดข้อมูลบางชุดเพิ่มเติมเข้ามา คือ เวลา วันเดือนปี ที่ได้รับข้อความ และเบอร์โทรศัพท์ของผู้ส่ง ดังนี้

1. ความยาวของเบอร์ศูนย์บริการ
2. รูปแบบของเบอร์ศูนย์บริการ (ส่งแบบสากลจะใช้ค่า = 0x91)
3. เบอร์ศูนย์บริการ โดยจะมีการเข้ารหัสแบบสลับ

ในส่วนของชุดข้อความสั้น (SMS) ก็จะประกอบด้วยส่วนย่อยๆ ซึ่งจะเป็นตัวกำหนดรูปแบบของข้อความที่รับมา โดยในส่วนนี้ จะมีส่วนที่แตกต่างจากการส่ง คือ เพิ่มเวลา วันเดือนปี ที่ได้รับข้อความ และเปลี่ยนเบอร์จากที่ต้องการส่ง เป็นเบอร์ที่ส่งมาจากต้นทาง โดยจัดรูปแบบเรียงตามนี้

1. โปรโตคอลพารามิเตอร์ คือ พารามิเตอร์ที่บอกว่า โปรโตคอล (Protocol) ที่ใช้ในการส่งเป็นแบบใด กรณีส่งแบบ TPDU = 0x01

2. ตัวเลขอ้างข้อความ ในกรณีมีข้อความหลายๆ ข้อความเราสามารถจัดลำดับข้อความโดยใช้ตัวเลขอ้างข้อความได้ (มีค่าปกติ = 0x00)

3. ความยาวของเบอร์โทรศัพท์มือถือของหมายเลขต้นทาง

4. รูปแบบของ เบอร์โทรศัพท์มือถือ ของหมายเลขต้นทาง ซึ่งจะเป็น ตัวบอกลักษณะของเบอร์โทรศัพท์มือถือที่เราต้องการส่งข้อความไปให้ โดยส่งแบบสากลจะใช้ค่า = 0x91

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เบอร์โทรศัพท์มือถือ ของหมายเลขต้นทางที่ต้องการจะส่ง โดยหมายเลขโทรศัพท์นี้จะมีการเข้ารหัสแบบสลับ

6. ตัวแสดงรูปแบบชุดข้อมูล

7. ลักษณะการเข้ารหัสข้อมูล คือ พารามิเตอร์ที่บอกว่าเราจะส่งเป็นภาษาใด (มาตรฐาน คือ ระบบ GSM)

8. เวลา และวันเดือนปี ที่ได้รับข้อความ เช่น 0x99 0x20 0x21 0x50 0x75 0x03 0x21 หมายถึง 12. Feb 1999 05 : 57 : 30 GMT + 3

9. ความยาวของข้อความที่ต้องการส่ง (ก่อนเข้ารหัส)

10. ข้อความที่ต้องการส่ง (หลังเข้ารหัส)

โดยปกติการส่งข้อความสั้นนี้เราสามารถกดส่งจากเครื่องโทรศัพท์มือถือของเราได้โดยเริ่มจากการเขียนข้อความ เมื่อเขียนข้อความเสร็จแล้วจะมีให้เลือกที่จะส่งไปเบอร์โทรศัพท์หมายเลขใดนอกจาก ที่เราต้องกดส่งจากโทรศัพท์มือถือ แล้วเรายังสามารถเลือกที่จะส่งข้อความสั้น ได้อีกแบบคือ ในเครื่องโทรศัพท์บางรุ่นที่มีอยู่ในปัจจุบันจะมีพอร์ตอนุกรม ซึ่งเราสามารถใช้พอร์ตอนุกรมนี้เป็นตัวเชื่อมต่อระหว่างคอมพิวเตอร์ หรือไมโครคอนโทรเลอร์ได้ พอร์ตอนุกรมที่มีนี้ทำให้เราง่ายในการส่งข้อความสั้นอย่างมาก คือ เราไม่จำเป็นต้องกดปุ่มที่เครื่องโทรศัพท์มือถือเพียงแค่เราส่งชุดคำสั่งเป็นรหัสแอสกี้ เข้าไปทางพอร์ตอนุกรมนี้เราก็สามารถสั่งงานให้เครื่องโทรศัพท์มือถือส่งข้อความสั้น

ตัวอย่างการส่งข้อความสั้นแบบโหมดพีดียู

โดยจะทำการส่งข้อความสั้นคำว่า “hellohello” โดยใช้ลักษณะการส่งแบบโหมดพีดียูไปยังหมายเลข “+66 092056208”

AT+CMGF=0 // เพื่อเลือกโหมดพีดียู

AT+CSMS=0 // เช็คว่ามือถือสนับสนุนการส่งข้อความสั้นหรือไม่

AT+CMGS=22 // ต้องการส่งทั้งหมด 22 bytes (ไม่รวมตัวเลข 00 ที่อยู่ข้างหน้าสุด) >0011000A9166295026800000AA0AE8329BFD4697D9EC37 // เมื่อพิมพ์ข้อความครบแล้วกด Ctrl+ z ส่วนประกอบของข้อมูลที่ส่งอธิบายในตารางที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 ส่วนประกอบของข้อมูลที่ส่ง

กลุ่มตัวเลข 8 บิต (Octet)	รายละเอียด
00	ความยาวของ SMSC Information 00 หมายถึงให้ใช้ SMSC Information ที่เก็บอยู่ภายในเครื่อง (ปกติเครื่องที่สามารถส่งSMSได้จะมีข้อมูล SMSC ภายในเครื่องอยู่แล้ว)
11	First octet of the SMS-SUBMIT message
00	TP-Message-Reference "00" คือให้เครื่องตั้งหมายเลขอ้างอิงข้อความขึ้นเอง
0A	Address-Length ความยาวของหมายเลขผู้รับ (0A hex = 10)
91	Type-of-Address (91 indicates international format of the phone number)
66 29 50 26 80	เลขหมายผู้รับ (แบบ Decimal Semi-Octets)เป็นเลขฐาน 10 สลับ Nibble หมายเลขที่แท้จริง คือ +66092056208
00	TP-PID (Protocol identifier) เป็น 00
00	TP-DCS (Data Coding Scheme) เป็น 00
AA	TP-Validity-Period "AA" หมายถึงช่วงเวลาหมดอายุของข้อความ 4 วัน ถ้าภายในช่วงเวลานี้ยังส่งไม่ถึงปลายทางข้อความจะถูกยกเลิกโดยอัตโนมัติ
CA	TP-User-Data-Lengthจำนวนตัวอักษรของข้อความที่ส่ง (10ตัว)
E8329BFD4697D9EC37	TP-UD ข้อความ "hellohello" ที่เข้ารหัสแล้วจากตัวอักษรแบบ 7 บิตเป็นข้อมูลไบต์ขนาด 8 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการรับข้อความสั้นแบบโหมดพีดียู

ถ้าหากเราเชื่อมต่อกับมือถือแล้วทำการอ่านข้อความสั้นที่อยู่ในถาดเข้า (Inbox) ได้โดยใช้คำสั่ง AT+CMGR ข้อมูลที่ได้รับจะอยู่ในรูปของสตริงที่ประกอบไปด้วยข้อมูลของผู้ส่ง ข้อมูล SMS Service Center (SMSC) Time Stamp และอื่นๆที่จำเป็นและตามด้วยส่วนของข้อความซึ่งจะอยู่ท้ายสุดของสตริง ตัวอย่าสตริงต่อไปนี้ข้อความที่ส่งมาคือ "hellohello" จากมือถืออีกเครื่องหนึ่ง ข้อมูลสตริงนี้จะอยู่ในรูปของตัวเลขฐาน 16 และฐาน 10 (ในบางส่วน) โดยจะเรียกตัวเลขแต่ละคู่ว่า Octetซึ่งมีรายละเอียดดังตารางที่2.3

0691668111808804 0A916629502680000040302121943482 0AE8329BFD4697D9EC37

ตารางที่ 2.3 ส่วนประกอบของสตริงการรับข้อมูลสั้น

กลุ่มตัวเลข 8 บิต (Octet)	รายละเอียด
06	ความยาวของ SMSC Information 6 Octets (ไบต์)
91	รูปแบบของหมายเลข SMSC 91 หมายถึง เลขหมายแบบสากล (international format)
00	TP-Message-Reference "00" คือให้เครื่องตั้งหมายเลขอ้างอิงข้อความขึ้นเอง
66 81 11 80 88	เลขหมาย SMSC (แบบ Decimal Semi-Octets)เป็นเลขฐาน 10 สลับ Nibble หมายเลขที่แท้จริง คือ +66092056208
04	First octet of this SMS-DELIVER message
0A	ความยาวของหมายเลขผู้รับ (0A hex = 10)
91	รูปแบบของหมายเลข SMSC 91 หมายถึง เลขหมายแบบสากล (international format)
66 29 50 26 80	เลขหมายผู้ส่ง (แบบ Decimal Semi-Octets) เป็นเลขฐาน 10 สลับ Nibble หมายเลขที่แท้จริง คือ +66092056208
00	TP-PID (Protocol identifier) เป็น 00
00	TP-DCS (Data Coding Scheme) เป็น 00 คือเข้ารหัสข้อความแบบ 7 บิต Default Alphabet
40 30 21 21 94 34 82	TP-SCTSข้อมูล Time Stamp (แบบ Decimal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	Semi-Octets) สลับ Nibble
0A	TP-UDL User data lengthจำนวนตัวอักษรของข้อความที่ส่งในที่นี้ คือ 10ตัว
E8329BFD4697D9EC37	TP-UD ข้อความ "hellohello" ที่เข้ารหัสแล้วจากตัวอักษรแบบ 7 บิตเป็นข้อมูลไบต์ขนาด 8 บิต

ข้อมูลทั้งหมดในตารางเป็นเลขฐาน 16 ขนาด 8 บิต ยกเว้นหมายเลขของศูนย์บริการข้อมูลสั้นหมายเลขผู้ส่ง Time Stamp จะเป็นเลขฐาน 10 ขนาด 8 บิตสลับหลักเป็นคู่ๆ (สลับ Nibble)ในส่วนของข้อมูลที่เป็นข้อความนั้นเป็นเลขฐาน 16 ขนาด 8 บิต เช่นกัน โดยข้อมูลนี้จะใช้แสดงข้อความที่ประกอบไปด้วยตัวอักษรขนาด 7 บิต ซึ่งผ่านการแปลง (เข้ารหัส) ข้อมูลจากตัวอักษรขนาด 7 บิต ให้เป็นเลขฐาน 16 ขนาด 8 บิต มาแล้ว ส่วนวิธีการแปลงจะกล่าวในภายหลัง

ในส่วนของข้อมูลที่เป็นเลขฐาน 10 เช่น หมายเลขผู้ส่งตัวเลขในแต่ละคู่ (1ไบต์) จะถูกสลับหลักกัน เช่น เลขหมายจริง "+66092056208" จะถูกสลับในแต่ละคู่เป็น "66 29 50 26 80" ซึ่งมีรูปแบบเป็น "YY/MM/DD HH:MM:SS:ss" หมายถึง ข้อความนี้ส่งเมื่อ "04/03/12 12:49:43:28"

การแปลงตัวอักษรชนิด 7 บิตเป็นข้อมูล 8 บิต (Octet) โดยจากตารางที่ ในส่วนของชุดข้อความ จะเป็นส่วนที่เราสามารถใส่รหัสของข้อความที่ต้องการส่ง แต่เนื่องจากเราสามรถนำรหัสของตัวอักษรแบบ 7 บิตใส่ไปได้โดยตรงจำเป็นต้องผ่านการแปลงให้เป็นรหัสข้อมูลแบบ 8 บิตก่อน โดยตัวอย่างต่อไปนี้เป็นการแปลงข้อความ "hellohello" ยาว 10 ตัวอักษรซึ่งแต่ละตัวเป็นอักษรชนิด 7 บิตให้เป็นข้อมูล 8 บิต สำหรับใช้ในการส่ง SMS การแปลงเริ่มจากการนำรหัส 7 บิตของตัวอักษรแรก (h) มาเติมข้างหน้าด้วย 1 บิตท้ายสุดของรหัส 7 บิต ของอักษรตัวที่ 2 (e) จะได้ผลลัพธ์ 8 บิต (1ไบต์) เป็น "E8" ขั้นต่อมาให้เอา 6 บิตที่เหลือของอักษรตัวที่ 2 มาเติมข้างหน้าด้วย 2 บิตท้ายของรหัส 7 บิตของตัวอักษรที่ 3 (l) จะได้ผลลัพธ์ 8บิต เป็น "32" และทำเช่นนี้เรื่อยไปโดยจำนวนบิตที่นำมากระทำจะเพิ่มขึ้นเป็น 3 บิต 4 บิต จนกระทั่งถึง 7 บิต แล้วเริ่มกระบวนการใหม่จนกระทั่งหมดชุดตัวอักษรหลังจากการแปลงข้อความ "hellohello" จะได้ข้อมูลเป็นเลขฐาน 16 จำนวน 9 ไบต์ เป็น E8 32 9B FD 46 97 D9 EC 37 โดยวิธีการแปลงแสดงดังตารางที่ 2.4 โดยที่ตัวอักษรชนิด 7 บิต ถูกกำหนดโดยมาตรฐาน GSM 03.38 ดังตารางที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 แสดงวิธีการแปลงตัวอักษรชนิด 7 บิต เป็นข้อมูล 8 บิต ข้อความ "hellohello"

h	e	l	L	o	h	e	l	l	o
104	101	108	108	111	104	101	108	108	111
11010100	1100101	1101100	1101100	1101111	1101000	1100101	1101100	1101100	1101111
1101000	1100101	1101100	1101100	1101111	1101000	1100101	1101100	1101100	1101111
11101000	00110010	10011011	11111101	01000110	10010111	11011001	1110100	110111	
E8	32	9B	FD	46	97	D9	EC	37	

ตารางที่ 2.5 ชุดของตัวแปรของมาตรฐาน GSM 03.38

Dec		0	16	32	48	64	80	96	112
	Hex	0	10	20	30	40	50	60	70
0	0	@	Δ	SP	0	¡	P		p
1	1	£		!	1	A	Q	a	q
2	2	$	Φ	"	2	B	R	b	r
3	3	¥	Γ	#	3	C	S	c	s
4	4	è	Λ	¤	4	D	T	d	t
5	5	é	Ω	%	5	E	U	e	u
6	6	ù	Π	&	6	F	V	f	v
7	7	ì	Ψ	'	7	G	W	g	w
8	8	ò	Σ	(	8	H	X	h	x
9	9	ç	Θ	)	9	I	Y	i	y
10	A	LF	Ξ	*	:	J	Z	j	z

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11	B	Ø	<ESC>	+	;	K	Ä	k	ä
12	C	ø	Æ	,	<	L	Ö	l	ö
13	D	CR	Æ	-	=	M	Ñ	m	ñ
14	E	Å		.	>	N	Ü	n	ü
15	F	å	É	/	?	O	§	o	à

2.6 คำสั่ง AT Command กับมือถือ

การสื่อสารกับอุปกรณ์สื่อสารต่างๆ เช่น โมเด็มหรืออุปกรณ์ DTE (Data Terminal Equipment) นั้นสามารถใช้ชุดคำสั่งที่เป็นมาตรฐาน เรียกว่า AT Command ในการติดต่อเพื่อโต้ตอบตั้งค่าหรือสั่งอุปกรณ์เหล่านั้น ให้ทำงานตามที่ต้องการ โดยชุดคำสั่งพื้นฐานจะถูกกำหนดไว้ใน Hayes AT Command ซึ่งบริษัท Hayes เป็นผู้คิดค้นชุดคำสั่งนี้เพื่อใช้กับโมเด็มของตนและต่อมาได้กลายเป็นมาตรฐานสำหรับผู้ผลิตโมเด็มรายอื่น ๆ ดดยอาจจะมีชุดคำสั่งขยาย (Extended AT Command) เพื่อใช้เป็นการเฉพาะสำหรับผู้ผลิตรายนั้น ๆ

การติดต่อกับมือถือ เราสามรถใช้ชุดคำสั่งที่กำหนดไว้ใน GSM AT Command ซึ่งมีคำสั่งเพิ่มเติมที่เหมาะสมสำหรับ การใช้งานและควบคุมมือถือ และเนื่องจากมีรายละเอียดค่อนข้างมาก เพราะฉะนั้นจะพูดถึงเฉพาะคำสั่งที่จำเป็นสำหรับโครงงานนี้เท่านั้นการเชื่อมต่อ คอมพิวเตอร์กับมือถือนั้น จะทำผ่านสายข้อมูล (Data Link) ซึ่งเป็นการเชื่อมต่อแบบอนุกรมโดยใช้โปรแกรมเทอร์มินอลต่างๆ เช่น ไฮเปอร์เทอร์มินอล (Hyper Terminal) ของ Windows ส่วนความเร็วที่ใช้ในการสื่อสารมักจะใช้ 19200 bps

คำสั่ง AT-Command สำหรับการส่งข้อความสั้นจาก GSM 07.05

AT+CNMI	เป็นคำสั่งเลือกสัญญาณข้อความสั้นใหม่
AT+CSCB	เป็นคำสั่งในการเลือกข้อความ Cell Broadcast
AT+CMGF	เป็นคำสั่งในการเลือกโหมดของข้อความที่จะส่ง
AT+CSCA	เป็นคำสั่งในการดูค่าของ ศูนย์บริการข้อความสั้น
AT+CMGL	เป็นคำสั่งเรียกอ่านข้อความโดยแสดงตามชนิดที่ต้องการเรียกดู
AT+CMGR	เป็นคำสั่งที่ใช้เรียกอ่านข้อความทีละอันเฉพาะที่ต้องการอ่าน
AT+CMGS	เป็นคำสั่งที่ใช้ในการส่งข้อความไปยัง Address ที่เลือกไว้
AT+CMSS	เป็นคำสั่งที่ใช้ส่งข้อความจากซิมการ์ดไปยัง Address ที่เลือกไว้
AT+CMGW	เป็นคำสั่งสำหรับเขียนข้อความเก็บไว้ในซิมการ์ต
AT+CMGD	เป็นคำสั่งสำหรับลบข้อความที่เก็บไว้ในซิมการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CSMS	เป็นคำสั่งที่ใช้เลือกบริการข้อความ
AT+CPMS	เป็นคำสั่งที่ใช้เลือกหน่วยความจำของการข้อความสั้น
AT+CMGC	เป็นคำสั่งที่ใช้ส่งคำสั่ง ข้อความสั้น

ลักษณะชุดคำสั่งของ AT+CNMI

ตารางที่ 2.6 ลักษณะชุดคำสั่งของ AT+CNMI

คำสั่ง	คำตอบสนอง
AT+ CNMI = ?	+CNMI: (list of support<mode>s), (list of support<mt>s),(list of support<bm>s), (list of support<ds>s), (list of support<bfr>s)
AT+ CNMI?	+CNMI:<mode>,<mt>,<bm>,<ds>,<bfr>
AT+ CNMI=[<mode>][,<mt>] [,<bm>][,<ds>][,<bfr>]	OK/ERROR/+CMS ERROR

<mode> 0 ถ้าบัฟเฟอร์เต็มแล้วมีสัญญาณเข้ามาใหม่จะเข้ามาแทนอันที่เก่าที่สุด
1 ไม่รับ Unsolicited result Code ใหม่เมื่อ TA-TE link ถูกจอง ไว้หรือไม่เช่นนั้นก็Forward ไปยัง TE โดยตรง
2 เก็บ Unsolicited result Codeในบัฟเฟอร์ของ TA เมื่อ TA-TE link ถูกจองไว้แล้วถูกส่งไปยัง TE เมื่อสิ้นสุดการจอง
3 ทำการ Forward ค่า Unsolicited result code ไปยัง TE โดยตรง

<mt> กฎสำหรับ การเก็บข้อความสั้น ที่รับเข้ามา ขึ้นอยู่กับวิธีการเข้า รหัสของข้อมูล การตั้งค่า Memory Format และค่านี้

<bm> กฎสำหรับการเก็บ CBMs ที่รับเข้ามานั้น ขึ้นอยู่กับวิธีการเข้า รหัสของข้อมูล การเลือกรูปแบบของ CBM และค่านี้

<ds> 0 ไม่มี SMS-STATUS-REPORT ส่งไปยัง TE
1 SMS-STATUS-REPORT ส่งไปยัง TE โดยใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
72015

Unsolicited Result Code + CDS : <length> <CR> <LF> <PDU> (PDU Mode Enable)

2 ถ้า SMS-STATUS-REPORT ส่งไปใน ME/TA สัญญาณของ Location ของหน่วยความจำ ถูกส่งไปยัง TE โดยใช้ Unsolicited Result Code

<bfr> 1 TA บัฟเฟอร์ของ Unsolicited Result Code จะถูกจำกัดความในคำสั่งนี้

ลักษณะชุดคำสั่งของ AT+CSCB

ตารางที่ 2.7 ลักษณะชุดคำสั่งของ AT+CSCB

คำสั่ง	คำตอบสนอง
AT+CSCB=?	+CSCB:(list of supported<model>s)
AT+CSCB?	+CSCB:<mode>,<mid>,<dcss>
AT+CSCB=[<model>[,<mid>[,<dcss>]]]	OK/ERROR

<mode> 0 รับข้อความ

1 ไม่รับข้อความ

<mids> ค่า CBM Message IDs : รูปแบบสตริง

<dcss> ค่า CBM Data Coding Schemes : รูปแบบสตริง

ลักษณะชุดคำสั่งของ AT+CMGF

ตารางที่ 2.8 ลักษณะชุดคำสั่งของ AT+CMGF

คำสั่ง	คำตอบสนอง
AT+CMGF=?	+CMGF:(list of support <mode>s)
AT+CMGF?	+CMGF:<mode>
AT+CMGF=[<mode>]	OK/ERROR

<mode> 0 เป็นโหมดพีดียู

1 เป็นโหมดตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะชุดคำสั่งของ AT+CSCA

ตารางที่ 2.9 ลักษณะชุดคำสั่งของ AT+CSCA

คำสั่ง	คำตอบสนอง
AS-CSCA=?	OK
AS-CSCA?	+CSCA:<sca>,<tosca>
AS-CSCA=<sca>[,<tosa>]	OK/ERROR

<sca> Service-Center Address In String Format

<tosca> Service-Center Address Format

ลักษณะชุดคำสั่งของ AT+CMGL

ตารางที่ 2.10 ลักษณะชุดคำสั่งของ AT+CMGL

คำสั่ง	คำตอบสนอง
AT+CMGL=?	
AT+CMGL=[<start>]	If PDU mode (+CMGF=0) +CMGL:<index>,<start>,[<alpha>],<length><CR> <LF><pdu><CR><LF>

<stat> ตัวบอกสถานะของข้อความที่อยู่ในซิมการ์ด

0 ข้อความที่ได้รับมาแล้วยังไม่ได้อ่าน

1 ข้อความที่ได้รับมาแล้วอ่านแล้ว

2 ข้อความที่เก็บไว้สำหรับส่งแต่ยังไม่ได้ส่ง

3 ข้อความที่ส่งไปแล้ว

4 ข้อความทุกชนิด

<index> ตัวบอกตำแหน่งที่เราต้องการเลือกว่าเป็นข้อความที่เท่าไรในซิมการ์ด

<length> ความยาวของส่วนของชุดข้อความโดยนับแบบOctet

<pdu> ข้อความ ที่เป็นส่วน ของศูนย์บริการส่งข้อความสั้น รวมกับส่วนของชุดข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะชุดคำสั่งของ AT+CMGR

ตารางที่ 2.11 ลักษณะชุดคำสั่งของ AT+CMGR

คำสั่ง	คำตอบสนอง
AT+CMGR=?	OK
AT+CMGR=<index>	If PDU mode (+CMGF=0) +CMGR;<start>,<alpha>],<lengh><CR> <LF><pdu>

<stat> ตัวบอกสถานะของข้อความที่อยู่ในซิมการ์ด

0 ข้อความที่ได้รับมาแล้วยังไม่ได้อ่าน

1 ข้อความที่ได้รับมาแล้วอ่านแล้ว

2 ข้อความที่เก็บไว้สำหรับส่งแต่ยังไม่ได้ส่ง

3 ข้อความที่ส่งไปแล้ว

4 ข้อความทุกชนิด

<index> ตัวบอกตำแหน่งที่เราต้องการเลือกว่าเป็นข้อความที่เท่าไรในซิมการ์ด

<length> ความยาวของส่วนของชุดข้อความโดยจะนับแบบOctets

<pdu> ข้อความ ที่เป็นส่วน ของศูนย์บริการส่งข้อความสั้น รวมกับส่วนของชุดข้อความ

ลักษณะชุดคำสั่งของ AT+CMGS

ตารางที่ 2.12 ลักษณะชุดคำสั่งของ AT+CMGS

คำสั่ง	คำตอบสนอง
AT+CMGS=?	OK
If PDU mode (+CMGF=0) AT=CMGS=<length><CR>PDU is given<ctrl-Z/ESC>	If sending is successful: +CMGS:<mr> If sending is not successful: +CMS ERROR :<err>

<length> ความยาวของส่วนของชุดข้อความโดยจะนับแบบ Octets

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<pdu> ข้อความ ที่เป็นส่วน ของศูนย์บริการส่งข้อความสั้น รวมกับส่วนของชุดข้อความ

<mr> จำนวนครั้งที่เราส่งข้อความสั้นหรือตัวอ้างอิงข้อความ

ลักษณะชุดคำสั่งของ AT+CMSS

ตารางที่ 2.13 ลักษณะชุดคำสั่งของ AT+CMSS

คำสั่ง	คำตอบสนอง
AT+CMSS=?	
AT+CMSS=<index>[,<da>[,<todo>]]	If sending is successful: +CMSS:<mr> If sending is not successful: +CMSS ERROR:<er>

<index> ตัวบอกตำแหน่งที่เราต้องการเลือกว่าเป็นข้อความที่เท่าไรในซิมการ์ด

<da> เป็นหมายเลขโทรศัพท์ที่เราต้องการส่งข้อความสั้นโดยจะอยู่ในรูป "หมายเลข" ซึ่งอยู่ในรหัสแอสกี้

<toda> เป็นหมายเลขโทรศัพท์ ที่เราต้องการส่งข้อความสั้น โดยจะอยู่ในรูป "+รหัสประเทศตามด้วยหมายเลข" ซึ่งอยู่ในรูปของตัวเลข

<mr> จำนวนครั้งที่เราส่งข้อความสั้นหรือตัวอ้างอิงข้อความ

ลักษณะชุดคำสั่งของ AT+CMGW

ตารางที่ 2.14 ลักษณะชุดคำสั่งของ AT+CMGW

คำสั่ง	คำตอบสนอง
AT+CMGW=?	OK
If PDU mode (+CMGF=0) AT+CMGW=<length>[,<start>]<CR> PDU is given <ctrl-Z/ESC>	 +CMGW:<index> +CMS ERROR:<err>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<stat> ตัวบอกสถานะของข้อความที่อยู่ในซิมการ์ด

0 ข้อความที่ได้รับมาแล้วยังไม่ได้อ่าน

1 ข้อความที่ได้รับมาแล้วอ่านแล้ว

2 ข้อความที่เก็บไว้สำหรับส่งแต่ยังไม่ได้ส่ง

3 ข้อความที่ส่งไปแล้ว

4 ข้อความทุกชนิด

<index> ตัวบอกตำแหน่งที่เราต้องการเลือกว่าเป็นข้อความที่เท่าไรในซิมการ์ด

<length> ความยาวของชุดข้อความ โดยจะนับแบบ Octets

<pdu> ข้อความ ที่เป็นส่วน ของศูนย์บริการส่งข้อความสั้น รวมกับส่วนของชุดข้อความ

ลักษณะชุดคำสั่งของ AT+CMGD

ตารางที่ 2.15 ลักษณะชุดคำสั่งของ AT+CMGD

คำสั่ง	คำตอบสนอง
AT+CMGD=?	OK
AT+CMGD=<index>	OK/+ERROR/+CMS ERROR

<index> ตัวบอกตำแหน่งที่เราต้องการเลือกว่าเป็นข้อความที่เท่าไรในซิมการ์ด

ลักษณะชุดคำสั่งของ AT+CSMS

ตารางที่ 2.16 ลักษณะชุดคำสั่งของ AT+CSMS

คำสั่ง	คำตอบสนอง
AT+CSMS=?	+CSMS: (list of supported<service>s)
AT+CSMS?	+CSMS:<service>,<mt>,<mo>,<bm>,
AT+CSMS=[<service>]	+CSMS:<mt>,<mo>,<bm> OK/ERROR/+CMS ERROR

<service> 0 GSM 3.40 และ 3.41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<mt> 1 รองรับรูปแบบ Mobile Terminate Message
0 ไม่รองรับรูปแบบ Mobile Terminate Message

<mo> 1 รองรับรูปแบบ Mobile Originate Message
0 ไม่รองรับรูปแบบ Mobile Originate Message

<bm> 1 รองรับรูปแบบ Broadcast Type Message
0 ไม่รองรับรูปแบบ Broadcast Type Message

ลักษณะชุดคำสั่งของ AT+CPMS

ตารางที่ 2.17 ลักษณะชุดคำสั่งของ AT+CPMS

คำสั่ง	คำตอบสนอง
AT+CPMS=?	+CPMS:(list of supported <mem1>s), (list of supported<mem2>s), (list of supported<mem3>s)
AT+CPMS?	+CPMS:<mem1>,<use1>,<total1>,<mem2>, <use2>,<total2>,<mem3>,<use3>,<total3>>
AT+CPMS=<mem1>[,<mem2>, [<mem3>]]	+CPMS:<use1>,<total1>,<use2>,<total2>,<use3>, <total3> OK/ERROR/+CMS ERROR

<mem1> ส่วนความจำสำหรับอ่านและลบข้อความ

<mem2> ส่วนความจำสำหรับเขียนและส่งข้อความ

<mem3> ส่วนความจำสำหรับข้อความที่รับมาเก็บไว้

<memx> ส่วนความจำสำหรับอ่านและลบข้อความ

<usex> จำนวนข้อความที่เก็บอยู่ใน <memx>

<totalx> จำนวนข้อความทั้งหมดที่สามารถเก็บได้ใน <memx>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะชุดคำสั่งของ AT+CMGC

ตารางที่ 2.18 ลักษณะชุดคำสั่งของ AT+CMGC

คำสั่ง	คำตอบสนอง
AT+CMGC=?	OK
If PDU mode (+CMGF=0) AT+CMGC=<length><CR>PDU is gieven<ctrl-Z/ESC	If sanding is successful: +CMGC:<mr> If sanding is not successful: +CMS ERROR:<err>

<length> ความยาวของชุดข้อความโดยจะนับแบบ Octets

<pdu> ข้อความ ที่เป็นส่วน ของศูนย์บริการส่งข้อความสั้น รวมกับส่วนของชุดข้อความ

<mr> จำนวนครั้งที่เราส่งข้อความสั้นหรือตัวอ้างอิงข้อความ

2.7 ไมโครคอนโทรลเลอร์ ตระกูล MCS-51

2.7.1 คุณสมบัติของไมโครคอนโทรลเลอร์ ตระกูล MCS-51

คุณสมบัติของไมโครคอนโทรลเลอร์ MCS-51 ที่สำคัญ ๆ มีดังนี้

- ต้องการแหล่งจ่ายไฟ 5 V เพียงชุดเดียว
- มีหน่วยความจำ สำหรับใช้เก็บโปรแกรมควบคุมการทำงาน อยู่ภายในชิพ จำนวน 4 กิโลไบต์ (เบอร์ 8031, 8032 ไม่มีหน่วยความจำส่วนนี้ ส่วนเบอร์ 8052 มีหน่วยความจำส่วนนี้ 8 กิโลไบต์ และเบอร์ 83C51FB จะมีหน่วยความจำส่วนนี้รวมทั้งสิ้น 16 กิโลไบต์)
- มีหน่วยความจำ สำหรับโปรแกรม และข้อมูล (RAM) อยู่ภายในชิพจำนวน 12 ไบต์ (ในเบอร์ 8031, 8051) หรือ 256 ไบต์ (ในเบอร์ 8031, 8052)
- สามารถใช้หน่วยความจำ สำหรับโปรแกรม และข้อมูลที่อยู่ภายนอกชิพ ได้อย่างละ 64 กิโลไบต์ แยกจากกัน
- คำสั่งส่วนใหญ่ใช้เวลาเพียง 1 ไมโครวินาที เมื่อใช้คริสตอลความถี่ 12 เมกกะเฮิรตซ์
- มีพอร์ตที่สามารถรับ หรือส่งข้อมูลได้ทั้ง 2 ทิศทาง จำนวน 4 พอร์ต พอร์ตละ 8 บิต หรือสามารถใช้งานเป็นพอร์ตขนาด 1 บิต แยกจากกัน ทำให้เหมือนมีพอร์ตขนาด 1 บิต ใช้งานรวมทั้งสิ้น 32 พอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รับ และส่ง ข้อมูลแบบอนุกรมได้ในตัว โดยสามารถกำหนดความเร็ว ในการรับและส่งข้อมูล (Baud Rate) ได้ตั้งแต่ 300 ถึง 375 กิโลบิตต่อวินาที

- จัดลำดับความสำคัญของสัญญาณอินเตอร์รัปต์ได้ 2 ระดับ

- มีรีจิสเตอร์ สำหรับใช้งานเป็น ไทม์เมอร์ หรือ เคาน์เตอร์ เพื่อนับจำนวนสัญญาณนาฬิกา ภายในชิพหรือนับเปลี่ยนสถานะของสัญญาณภายนอก 16 บิต จำนวน 2 ตัว เพื่อใช้สำหรับนับจำนวนพัลส์ วัดความกว้างของพัลส์ หรือใช้วัดช่วงเวลา (ในเบอร์ 8052 จะมี 3 ตัว)

- หน่วยความจำสำหรับเก็บข้อมูลภายในบางส่วน สามารถเข้าถึงข้อมูลได้ ทั้งระดับบิต เพื่อให้การออกแบบโปรแกรม และการควบคุมระบบทำได้ง่ายขึ้น

- มีคำสั่งคูณและหารเลขในตัวเอง

- สามารถประมวลผลแบบบูลีน เพื่อใช้งานควบคุมโดยเฉพาะ

2.7.2 คุณสมบัติ AT90S2313

ไอซีไมโครคอนโทรลเลอร์ ที่นำมาศึกษาเป็นไอซีไมโครคอนโทรลเลอร์เบอร์ AT90S2313 เป็นไอซีซีมอสขนาด 20 ขา ซึ่งมีคุณสมบัติที่สำคัญดังต่อไปนี้

1. โครงสร้างเป็นแบบ AVR R

2. ประสิทธิภาพสูง และใช้พลังงานต่ำ มี 118 คำสั่ง ซึ่งส่วนมากทำงานเสร็จใน 1 cycle รีจิสเตอร์8 บิต จำนวน 32 ตัว และความเร็ว 10 MIPS ที่ความถี่ 10 MHz

3. หน่วยความจำ มี 3 ส่วน Flash memory จำนวน 2 Kbytes เขียน/ลบ ได้ไม่ต่ำกว่า 1000 ครั้ง SRAM 128 Byte EPROM 128 Byte เขียน/ลบ ได้ไม่ต่ำกว่า 100,000 ครั้ง มีระบบรักษาความปลอดภัย สามารถล๊อคไม่ให้อ่านโปรแกรมใน flash memory และข้อมูลใน EEPROM ได้

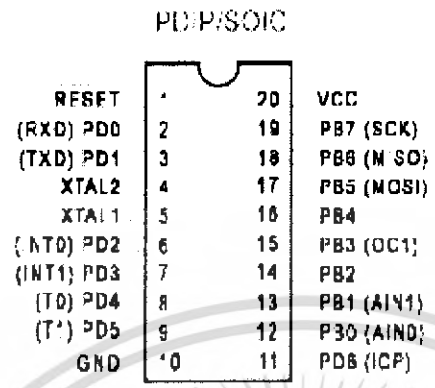
4. อุปกรณ์ภายใน มี Timer/counter 8 บิต จำนวน 1 ตัว พร้อมวงจรหารความถี่ clock แยกอิสระ มี Timer/counter 16 บิต จำนวน 1 ตัว พร้อมวงจรหารความถี่ clock แยกอิสระ วงจรเปรียบเทียบ capture mode และ PWM 8,9 หรือ 10 บิต อะนาลอกคอมพาราเตอร์ จำนวน 1 Programmable watchdog Timer พร้อม on-ship oscillator 1 ชุด SPI สำหรับใช้ทำ In-system Programming และพอร์ตอนุกรม Full duplex UART 1 ชุด

5. คุณสมบัติพิเศษ คือ Low power Idle and power down mode External and Internal Interrupt Source เป็นไอซีแบบซีมอส ความเร็วสูง แรงดันใช้งาน 4-6 V ความถี่คล็อค 0-4 MHz ที่ความถี่ 4 MHz แรงดัน 3 โวลต์ อุณหภูมิ 25 องศาเซลเซียส จะใช้กระแสไฟฟ้าในขณะ Active เท่ากับ 2.8 mA ในสภาวะ Idle เท่ากับ 0.8 mA และในสภาวะ Power Down จะใช้กระแสไฟฟ้าน้อยกว่า 1 Ua

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.3 การจัดขาไมโครคอนโทรลเลอร์ AT90S2313

ตัวถังจะเป็นแบบ PDIP/SOIC ซึ่งการจัดขาจะมีลักษณะดังรูป



รูปที่ 2.6 แสดงการจัดขาไมโครคอนโทรลเลอร์ AT90S2313

1. PORTB (PB7- PB0)

เป็นพอร์ต 8 บิต สามารถโปรแกรมให้แต่ละบิตของพอร์ตเป็นอินพุท หรือเอาต์พุตก็ได้ แยกกันเป็นอิสระในแต่ละบิต แต่ละขาของพอร์ตสามารถจ่ายกระแสไปให้อุปกรณ์ภายนอก เช่น LED ได้ถึง 20 mA นอกจากนี้ขา PB0 และ PB1 ยังใช้เป็นอินพุต AIN0 และ AIN1 สำหรับ Analog comparator อีกด้วย

2. PORTD (PD6- PD0)

เป็นพอร์ต 7 บิต คุณสมบัติคล้ายกันกับ PORTB นอกจากนี้ PORTD ยังใช้สำหรับหน้าที่อื่นอีก คือ

PD0 = RXD เป็นขาอินพุต สำหรับพอร์ตอนุกรม

PD1 = RXD เป็นขาเอาต์พุต สำหรับพอร์ตอนุกรม

PD2 = INT0 เป็นขารับอินเตอร์รัพต์จากภายนอก

PD3 = INT1 เป็นขารับอินเตอร์รัพต์จากภายนอก

PD4 = T0 เป็นขาสำหรับรับสัญญาณ clock จากภายนอกสำหรับTimer/Couter0

PD5 = T1 เป็นขาสำหรับรับสัญญาณ clock จากภายนอกสำหรับTimer/Couter1

PD6 = ICP ใช้เป็นอินพุต Timer/Couter1 เก็บค่าที่กำลังนับไว้ในรีจิสเตอร์

3. RESET

เป็นขาสำหรับรีเซ็ท ลอจิก 0 ที่มีเวลามากว่า 50 ns จะทำให้เกิดรีเซท ถึงแม้ว่าจะไม่มี สัญญาณคล็อคเข้ามาก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. XTAL1

เป็นขาอินพุตของ Inverting oscillator amplifier สำหรับรับสัญญาณจากภายนอก และเป็นอินพุตสำหรับการทำงาน ของคล็อคภายใน (กรณีที่เราต่อ x-tal ไว้)

5. XTAL2

เป็นขาเอาต์พุตของ Inverting oscillator amplifier

2.7.4 การรีเซ็ตและการอินเตอร์รัพท์

ไมโครคอนโทรลเลอร์AT90S2313 มีระบบการตอบสนองสัญญาณอินเตอร์รัพท์จาก 11 แหล่งสัญญาณ โดยแยกอินเตอร์รัพท์เวกเตอร์ของแต่ละอินเตอร์รัพท์ออกจากกันในการควบคุมการตอบสนองของอินเตอร์รัพท์แต่ละแหน่งสามารถแยกควบคุมได้จากบิทENABLE ของอินเตอร์รัพท์นั้นๆ และ บิท I ซึ่งใช้ควบคุมการอินเตอร์รัพท์ทั้งหมดโดยตำแหน่งแรกๆ ใน Program Memory จะเป็นตำแหน่งที่ถูกใช้เป็นพื้นที่ของอินเตอร์รัพท์เวกเตอร์ และรีเซ็ทการอินเตอร์รัพท์ที่มีระดับความสำคัญสูงสุดคือ RESET จนถึงอินเตอร์รัพท์ที่มีความสำคัญต่ำสุด

ตารางที่ 2.19 แสดงอินเตอร์รัพท์เวกเตอร์

Vector	Program Address	Source	Interrupt Definition
1	$000	RESET	Hardware Pin and Watchdog Reset
2	$001	INT0	External Interrut Request 0
3	$002	INT1	External Interrut Request 1
4	$003	TIMER1 CAPT1	Timer /Counter1 Capture Event
5	$004	TIMER1 COMP1	Timer /Counter1 Compare Match
6	$005	TIMER1 OVF1	Timer /Counter1 Overflow
7	$006	TIMER0 OVF0	Timer /Counter0 Overflow
8	$007	UART , RX	UART , Rx Complete
9	$008	UART , UDRE	UART Data Register Empty
10	$009	UART , TX	UART , Tx Complete
11	$00A	ANA_COMP	Analog Comparator

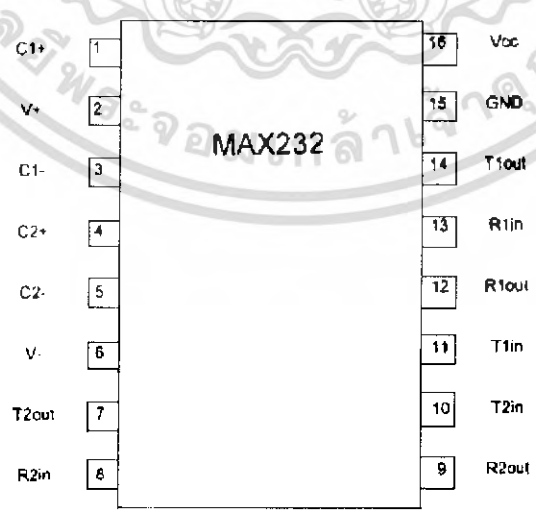
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 IC MAX 232

เป็น IC ที่ทำหน้าที่เปลี่ยนแรงดันที่เข้ามาจาก Serial Port ไปเป็นแรงดันตามมาตรฐานของ RS-232 โดยเปลี่ยนระดับแรงดันทีทีแอล (TTL) เพื่อทำให้ใช้ได้กับไมโครคอนโทรเลอร์

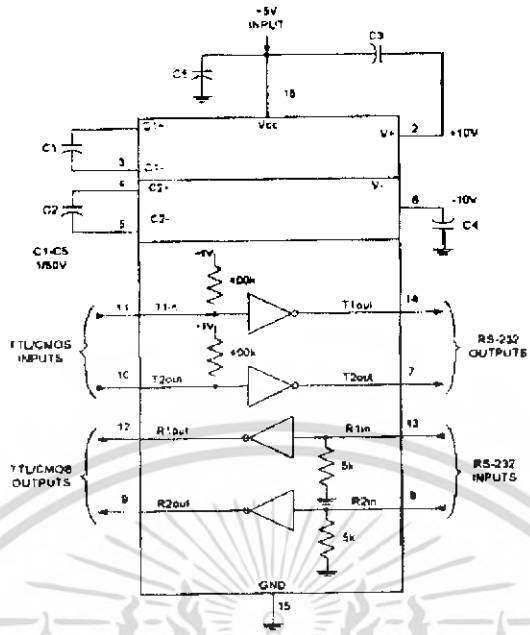
2.8.1 การเชื่อมต่อกับพอร์ตอนุกรมของคอมพิวเตอร์

การใช้งานวงจรพอร์ตอนุกรมของไมโครคอนโทรเลอร์ MCS-51 มักนิยมใช้ในการติดต่อเพื่อแลกเปลี่ยนข้อมูลกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรมในมาตรฐาน RS_232 เป็นส่วนใหญ่ แต่เนื่องจากระดับสัญญาณของพอร์ตอนุกรม RS-232 มีระดับตั้งแต่ ± 3 ถึง ±12 V ในขณะที่ระดับสัญญาณของไมโครคอนโทรเลอร์ MCS-51 อยู่ในระดับทีทีแอล ดังนั้นจึงไม่สามารถเชื่อมต่อพอร์ตอนุกรมของไมโครคอนโทรเลอร์ MCS-51 เข้ากับพอร์ดอนุกรมของคอมพิวเตอร์ได้โดยตรง จึงต้องอาศัยการเชื่อมต่อผ่านไอซีพิเศษที่ทำหน้าที่ในการแปลงระดับสัญญาณ ไอซีที่ทำหน้าที่ในการแปลงระดับสัญญาณนี้ต้องทำการแปลงข้อมูลส่งของไมโครคอนโทรเลอร์ MCS-51 จากระดับทีทีแอลไปเป็นระดับของ RS-232 และทำการแปลงข้อมูลรับจากคอมพิวเตอร์จากระดับของ RS-232 เป็นระดับทีทีแอลเพื่อให้สามารถถ่ายทอดไปยังไมโครคอนโทรเลอร์ MCS-51ได้อย่างสมบูรณ์ ไอซีดังกล่าวมีด้วยกันหลายเบอร์ จากหลายผู้ผลิต อาทิ MAX232 จาก MAXIM หรือ ICL232 จาก HARRIS เป็นต้น ในรูปที่ 2.7 แสดงการจัดขาไอซี ICL232 ซึ่งใช้ในการแปลงสัญญาณ RS-232 และรูปที่ 2.8 แสดงโครงสร้างภายในของไอซี ส่วนวงจรของการต่อกับไมโครคอนโทรเลอร์ MCS-51 แสดงในรูปที่ 2.9

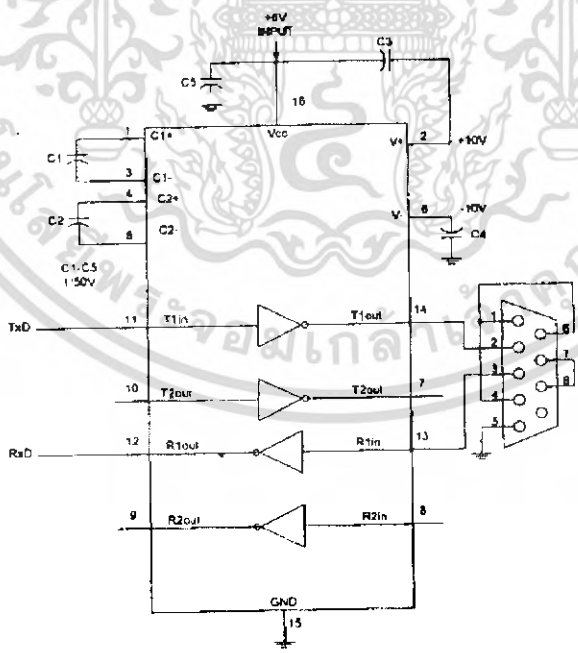


รูปที่ 2.7 การจัดขาของ MAX232 หรือ ICL232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 โครงสร้างภายในของ MAX232 หรือ ICL232



รูปที่ 2.9 แสดงวงจรเชื่อมต่อ MAX232 หรือ ICL232 เข้ากับพอร์ตอนุกรมของคอมพิวเตอร์และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรเลอร์ MCS-51

2.8.2 การสื่อสารพอร์ตอนุกรม RS-232

ลักษณะของการส่งข้อมูลแบบอนุกรมนั้น ข้อมูลจะส่งออกมาทีละบิต จากตัวส่งไปตัวรับข้อมูล ในการส่งข้อมูลอาจใช้ช่องสัญญาณเพียง 1 หรือ 2 ช่องสัญญาณเท่านั้น ทำให้ค่าใช้จ่ายในการสื่อสารจะถูกกว่าแบบขนาน แต่อัตราการรับ-ส่งข้อมูลจะช้ากว่าแบบขนาน ในการส่งข้อมูลแบบอนุกรม ข้อมูลที่ต้องการส่งจะอยู่ในลักษณะเป็นไบต์ จะทยอยส่งทีละบิต และทางตัวรับจะต้องรับข้อมูลเข้ามาทีละบิต แล้วมารวมกันเป็นไบต์ ซึ่งทางตัวรับต้องคอยตรวจสอบ ว่าบิตใดเป็นบิตเริ่มต้นหรือบิตสุดท้ายของข้อมูล การตรวจสอบนั้น จะขึ้นอยู่กับรูปแบบของรหัสของบิตข้อมูลที่ใช้ ซึ่งในการรับส่งข้อมูลแบบอนุกรม ระหว่างไมโครคอมพิวเตอร์ กับอุปกรณ์ภาย นอกนั้นจำเป็นจะต้องมีมาตรฐานในการรับส่งข้อมูล ซึ่งมาตรฐานในการรับส่งข้อมูลที่นิยมมากที่สุดคือ มาตรฐาน RS-232

2.8.3 มาตรฐาน RS-232

เพื่อจะทำอุปกรณ์จากผู้ผลิตต่างกันทำงานร่วมกันได้มาตรฐานหลากหลายชนิดจึงได้รับการออกแบบขึ้นมาตรฐานที่ใช้กันอย่างกว้างขวางที่สุดคือ RS-232C ซึ่งโดยปกติไมโครคอมพิวเตอร์จะมีพอร์ตที่เป็นแบบอนุกรมอยู่ในตัวอยู่แล้ว และทำหน้าที่รับส่งข้อมูลในแบบอนุกรม โดยการใช้สายสัญญาณเพียง 3 เส้นเท่านั้น คือ ใช้สาย TD สาย RD สายกราวด์ เท่านั้น ในการเชื่อมต่อไมโครคอนโทรเลอร์ กับพอร์ตอนุกรมของคอมพิวเตอร์นั้น จะใช้คอนเน็กเตอร์แบบ DB9 ตามมาตรฐาน RS-232

ตารางที่ 2.20 รายละเอียดการต่อคอนเน็กเตอร์แบบ DB9 มาตรฐาน RS-232

หมายเลขขาสัญญาณ	ชื่อของสายสัญญาณ
1	Data Carrier Detect
2	Receive Data
3	Transmitted Data
4	Data Terminal Ready
5	Signal Common
6	Data Set Ready
7	Request To Send
8	Clear To Send
9	Ring Indicator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ขา 1 (Data Carrier Circuit CF, DCD) ขานี้รู้จักในนามของ Receive Line Signal Detect (RLSD) หรือขา Carrier Detect (CD) สัญญาณนี้จะเกิด Active เมื่อเกิดสัญญาณพาหะที่เหมาะสมระหว่างอุปกรณ์ DCE ที่สถานีกับที่อยู่ระยะไกล เมื่อสัญญาณนี้อยู่ในระยะ "OFF" สัญญาณที่ขา RD ควรจะถูกทำให้ค้างอยู่ในสภาวะ "Mark" (สถานะ "1" ในเลขฐานสอง)

- ขา 2 (Receive Data Circuit BB, RD) สัญญาณที่ขานี้จะถูกสร้างจากอุปกรณ์ DCE กระแสบิต อนุกรมนี้จะกำเนิดขึ้นที่อุปกรณ์ DTE ปลายทางและเป็นผลผลิตของวงจรรับข้อมูลของอุปกรณ์ DCE สัญญาณนี้มักเป็นข้อมูลที่ถูกสร้างขึ้นโดยอุปกรณ์ DCE

- ขา 3 (Transmitted Data Circuit BA, TD) เป็นขาสัญญาณข้อมูลที่ออกจากอุปกรณ์ DTE กระแสบิตอนุกรมจากขานี้ คือข้อมูลที่ถูกถ่ายทอดไปโดยโมเด็ม หรือถูกถอดรหัสโดยอุปกรณ์ DCE

- ขา 4 (Data Terminal Ready Circuit CD, DTR) สัญญาณ DTR ถูกนำมาใช้ในการควบคุมสวิตช์อุปกรณ์ DCE เข้ากับตัวกลางในการสื่อสารสัญญาณ DTR ON บ่งชี้ว่า อุปกรณ์ DCE ที่กำลังเชื่อมต่อกันอยู่ ก็ยังต่อร่วมกัน และถ้ายังไม่มีการเชื่อมต่อกันก็สามารถทำให้มีการเชื่อมต่อกันครั้งใหม่ได้ ปกติแล้วสัญญาณ DTR จะอยู่ในสภาวะ "OFF" เพื่อกระตุ้นทำให้เกิดสภาวะ "ON HOOK" (วางสาย) อุปกรณ์ DCE โดยปกติแล้วจะตอบสนองต่อการกระตุ้นจากสัญญาณ DTR โดยการทำให้สัญญาณ DSR แอกทีฟ

- ขา 5 (Signal Common Circuit AB) สายนี้จะทำให้สัญญาณอ้างอิง ของกราวด์ร่วมกันสำหรับวงจรการแลกเปลี่ยนข้อมูลทั้งหมด ยกเว้นวงจร AA หรือ Protective Ground ที่ข้อกำหนด RS-232 จะอนุญาตให้วงจรนี้ถูกตัดต่อเพิ่มเติมกับ Protective Ground ภายในอุปกรณ์ DCE ได้ ถ้าจำเป็น

- ขา 6 (Data Set Ready Circuit CC, DSR) สัญญาณ DSRจะถูกนำมาต่ออุปกรณ์ DTE ว่าอุปกรณ์ DCE ได้ต่อกับตัวกลางการสื่อสารที่ถูกต้องแล้ว และในบางกรณีจะบ่งชี้ว่าสายโทรศัพท์อยู่ในสภาวะ "OFF HOOK" ซึ่งจะเป็นตัวบ่งชี้ว่าอุปกรณ์ DCE กำลังอยู่ในโหมด Dialing หรือกำลังติดต่ออยู่กับอุปกรณ์ DCE อีกตัวหนึ่งอยู่เมื่อสัญญาณ DSR นี้อยู่ในสภาวะ "OFF" อุปกรณ์ DTE ก็ควรจะถูกกำหนดให้ไม่สนใจสัญญาณอื่นๆทั้งหมดจากอุปกรณ์ DCE ถ้าสัญญาณนี้ถูกทำให้อยู่ในสภาวะ "OFF" ก่อนอุปกรณ์ DTR แล้วอุปกรณ์ DCE ก็จะสรุปว่าการสื่อสารนั้นสิ้นสุดลง

- ขา 7 (Request To Send Circuit CA, RTS) สัญญาณนี้จะใช้สำหรับ เตรียมพร้อมอุปกรณ์ DCE สำหรับการทำการส่งข้อมูลเมื่อสัญญาณ RTS นี้อยู่ในสภาวะ "ON" จะทำให้อุปกรณ์ DCE อยู่ในโหมดส่งข้อมูล (Transmit Mode) ในขณะที่สัญญาณนี้อยู่ในสภาวะ "OFF" ทำให้อุปกรณ์ DCE อยู่ในโหมดรับข้อมูล (Receive Mode) อุปกรณ์ DCE ควรจะตอบสนองต่อสัญญาณ RTS ON โดยการทำให้สัญญาณ Clear To Send (CTS) อยู่ในสภาวะ "ON" ด้วย เมื่อ RTS อยู่ในสภาวะ "OFF" สัญญาณนี้ไม่ควรจะ "ON" ขึ้นอีกจนกว่าสัญญาณ CTS อยู่ในสภาวะ "OFF" เสียก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณนี้จะถูกใช้ร่วมกับสัญญาณ DTR DSR และ DCD ขาสัญญาณ RTS จะถูกใช้อย่างมากในการควบคุมการไหลของข้อมูล

- ขา 8 (Clear To Send Circuit CB, CTS) สัญญาณนี้จะตอบรับ กลับไปยังอุปกรณ์ DTE เมื่อได้รับสัญญาณ RTS และข้อมูลสามารถส่งออกไปได้ ข้อมูลจะถูกส่งออกไปตามตัวกลางที่ใช้สื่อสารได้ ก็ต่อเมื่อสัญญาณ CTS นี้ อยู่ในสภาวะ "ON" เท่านั้น สัญญาณนี้จะใช้ร่วมกับสัญญาณ DTR DSR และ DCD ขาสัญญาณนี้จะใช้ร่วมกับขา RTS สำหรับควบคุมการไหลข้อมูล

- ขา 9 (Ring Indicator Circuit CE, RI) สภาวะ "ON" ของขานี้จะบ่งชี้ว่า ได้รับสัญญาณเรียกสายโทรศัพท์จากตัวกลางในการสื่อสารโดยปกติแล้ว ขึ้นอยู่กับโปรแกรมควบคุม ในการที่จะทำให้เกิดสัญญาณนี้ขึ้นหรือไม่

ขั้นตอนการติดต่อระหว่างอุปกรณ์ DTE และ DCE

1. เมื่อจ่ายกำลังงานให้กับ DTE และอุปกรณ์ก็จะส่งสัญญาณ DTR ออกมา
2. อุปกรณ์ DCE ถูกเปิดขึ้นและรับรู้สัญญาณ DTR ที่ส่งมาจากอุปกรณ์ DTE
3. อุปกรณ์ DCE ส่งสัญญาณ DSR ออกมา และโมเด็มก็กระทำกระบวนการ OFF HOOK
4. ถ้าสายสัญญาณอยู่ในสภาพดี และอุปกรณ์อีกข้างหนึ่ง ก็พร้อมจะรับข้อมูลแล้ว โดยจะตรวจจับพบสัญญาณพาหะแล้วอุปกรณ์ DCE แล้วส่งสัญญาณ CTS ออกมา
5. อุปกรณ์ DCE จะตอบสนองด้วยการส่งสัญญาณ CTS ออกมา
6. การติดต่อสื่อสารก็เริ่มขึ้น โปรแกรมควบคุมจะทำการส่งหรือรับข้อมูล

ส่วนลำดับขั้นตอนในการตอบรับก็เป็นดังนี้

1. อุปกรณ์ DTE จะส่งสัญญาณ DTR ออกมา
2. อุปกรณ์ DCE จะอยู่ในโหมด ตอบรับอัตโนมัติ (Auto Answer Mode) โดยจะมีสัญญาณ DSR ออกมา
3. สถานีปลายทางส่งสัญญาณเรียก อุปกรณ์ DCE และอุปกรณ์ DCE เพื่อที่จะส่งสัญญาณ RI ออกมา
4. อุปกรณ์ DTE รับรู้ถึง RI ที่ถูกส่งออกมาจากเครื่องปลายทาง และอุปกรณ์ DCE ก็เข้าสู่สภาวะ OFF HOOK
5. อุปกรณ์ DCE ทำการแลกเปลี่ยนข้อมูลกับ อุปกรณ์ DCE ที่อีกปลายทางหนึ่ง และมีการส่งสัญญาณ DCD ออกมา
6. อุปกรณ์ DTE จะส่งสัญญาณ RTS ออกมา หรือจะรอข้อมูล ขึ้นอยู่กับโปรแกรมควบคุม
7. อุปกรณ์ DCE จะตอบสนองด้วยการส่งสัญญาณ DTS ออกมา การติดต่อสื่อสาร ก็จะเริ่มขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 รีเลย์ (Relay)

เป็นอุปกรณ์ทำหน้าที่เป็นสวิตช์ มีหลักการในการทำงานคล้าย กับขดลวดแม่เหล็กไฟฟ้า หรือโซลินอยด์(Solenoid) รีเลย์ใช้ในการควบคุมวงจรไฟฟ้าได้อย่างหลากหลาย รีเลย์เป็นสวิตช์ควบคุมที่ทำงานด้วยไฟฟ้า แบ่งออกตามลักษณะการใช้งานได้เป็น 2 ประเภท คือ

1. รีเลย์กำลัง (Power Relay) หรือมักเรียกกันว่า คอนแทกเตอร์ (Contactor or Magnetic Contactor) ใช้ในการควบคุมไฟฟ้ากำลัง มีขนาดใหญ่กว่ารีเลย์ธรรมดา

2. รีเลย์ควบคุม (Control Relay) มีขนาดเล็กกำลังไฟฟ้าต่ำ ใช้ในวงจรควบคุมทั่วไปที่กำลังไฟฟ้าไม่มากนัก หรือเพื่อใช้ในการควบคุมรีเลย์ หรือคอนแทกเตอร์ขนาดใหญ่ รีเลย์ควบคุมบางทีเรียกกันง่ายๆว่า "รีเลย์"

โดยรีเลย์นั้นได้ถูกนำมาประยุกต์ใช้ในงานไฟฟ้าได้อย่างหลากหลาย รีเลย์ซึ่งมีความหมายว่า การส่งผ่านกระแสหรือแรงดันไฟฟ้าจากส่วนหนึ่งของวงจรไปสู่อีกส่วนหนึ่ง รีเลย์มีหลายขนาดและนำไปประยุกต์ใช้ได้ต่างๆกัน ถ้าไม่คำนึงถึงขนาดแล้วส่วนประกอบพื้นฐานจะเหมือนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

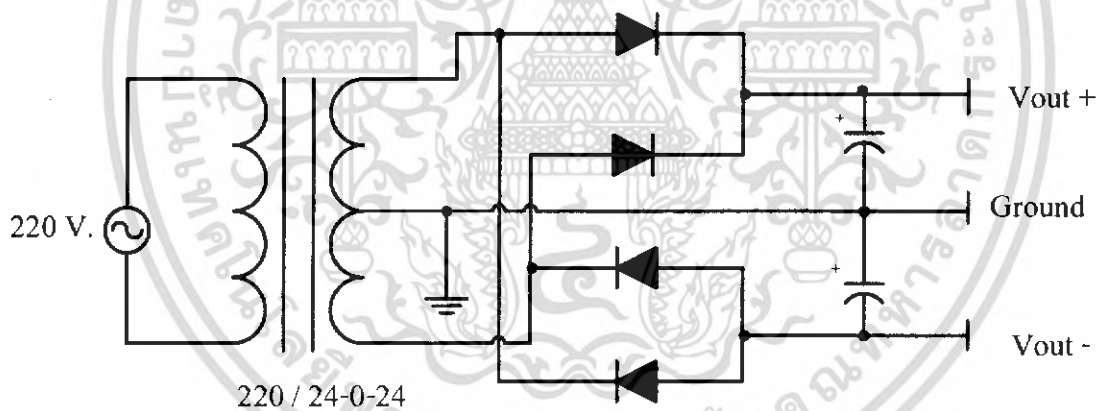
บทที่ 3
การออกแบบและการสร้างวงจร

3.1 การออกแบบวงจร

- ส่วนจ่ายไฟเลี้ยงให้กับวงจร
- ส่วนไมโครคอนโทรลเลอร์ควบคุม
- ส่วน adc รับค่าแรงดันจาก Microphone ที่จะมาวิเคราะห์
- ส่วนเชื่อมต่อกับโทรศัพท์เคลื่อนที่หรือคอมพิวเตอร์

3.2 วงจรจ่ายแรงดันไฟฟ้ากระแสตรง

จากการที่มีการต่อตัวไมโครคอนโทรลเลอร์เพื่อนำมาควบคุมการทำงานนั้น อุปกรณ์ชนิดนี้ต้องการแรงดัน ที่จะนำไปเลี้ยงวงจรที่มีขนาดเฉพาะ ในโครงงานนี้ต้องการแรงดันที่ 5 V จึงทำให้ต้องการเพิ่มในส่วนของวงจรจ่ายแรงดันดังนี้



รูปที่ 3.1 วงจรจ่ายแรงดันไฟฟ้ากระแสตรง

1. ไดโอดบริดจ์ชนิดเต็มคลื่น (Full Wave Bridge) ต้องมีความสามารถในการ ทนกระแสได้อย่างน้อยสองเท่า ของกระแสที่ไหลผ่านไดโอด และมีค่าแรงดันพังทลาย (Break Down Voltage) มากกว่า $4\sqrt{2}$ เท่าของแรงดันไฟฟ้าสูงสุด

2. ตัวเก็บประจุ ทำหน้าที่ให้ระดับแรงดันไฟฟ้ามีความเรียบมากขึ้น ซึ่งมีการคำนวณ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$C \geq I_{max} / (F_r \times \Delta V)$$

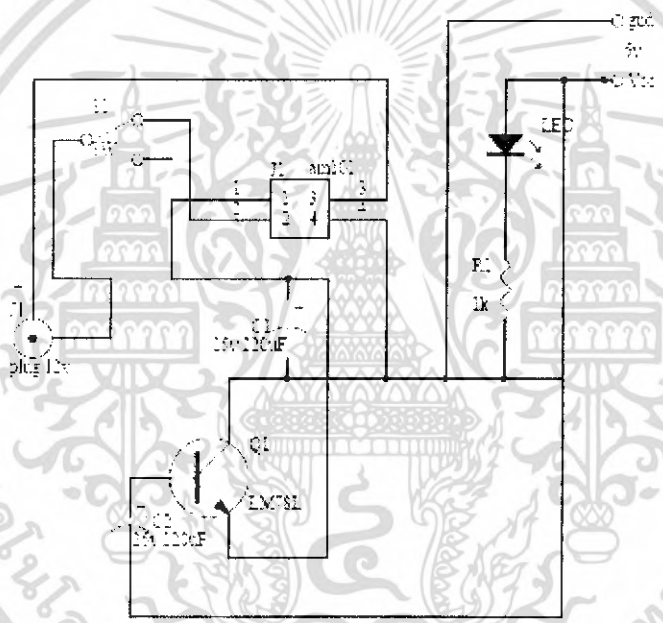
เมื่อ

I_{max} คือ กระแสสูงสุดที่ภาระไฟฟ้าต้องการ

F_r คือ ความถี่ของแรงดันที่ออกจากไดโอดบริดจ์

ΔV คือ ขนาดของแรงดันกระเพื่อมที่เราต้องการ

จากการคำนวณ จากสูตรด้านบนแล้ว จะได้วงจรที่จะนำไปใช้ กับวงจรที่ออกแบบ จะได้ค่าต่าง ๆ ดังนี้

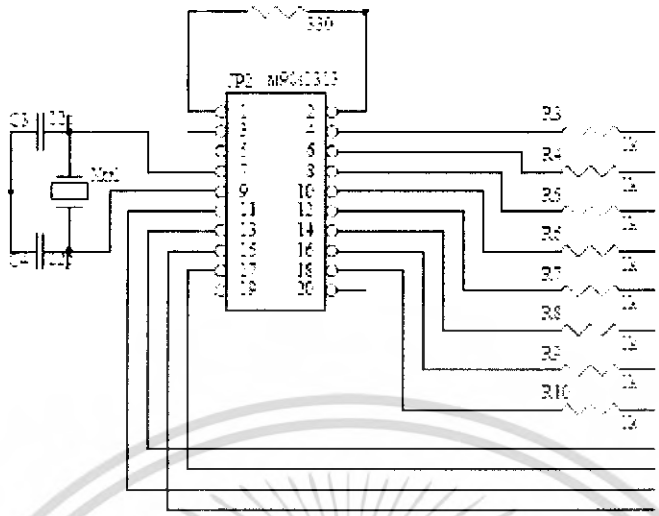


รูปที่ 3.2 วงจรที่จะนำไปใช้กับวงจรที่ออกแบบ

3.3 ส่วนควบคุมการทำงาน

ในโครงงานนี้ เราจะใช้ไอซีเบอร์ at90s2313 ในการนำมาใช้เป็นตัวควบคุม ทั้งในการสั่งให้ adc0820ccn ส่งค่าแรงดันที่สามารถตรวจจับได้มาที่ at90s2313 แล้ววิเคราะห์ค่าที่ได้มาและส่งคำสั่งไปให้กับโทรศัพท์เคลื่อนที่ในการแจ้งเตือนในกรณี ที่วิเคราะห์ได้ว่า มีคนมาขโมยรถ หรือทำให้รถเสียหาย โดยต่ออุปกรณ์ต่าง ๆ ในส่วนนี้ดังนี้

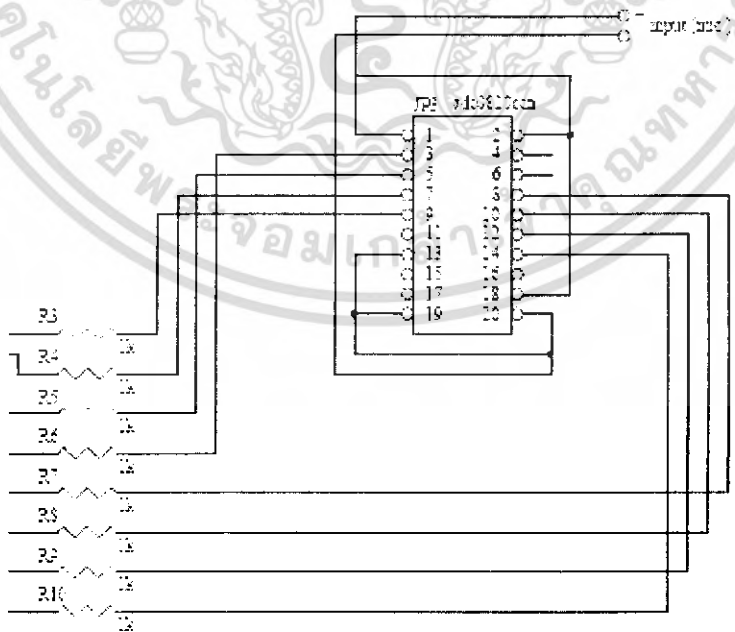
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 วงจรส่วนควบคุมการทำงาน

3.4 วงจรในส่วนแปลงค่าแรงดันไปเป็นสัญญาณดิจิตอล

จากการที่ at90s2313 ไม่มีโมดูลที่จะอ่านค่าแรงดันโดยตรง ทำให้เราจะต้องทำวงจรแปลงสัญญาณอนาลอก ไปเป็นดิจิตอลเพิ่มให้กับวงจรควบคุมก่อน จึงจะสามารถนำค่าที่ได้ออกมานั้น มาทำการวิเคราะห์ โตยวงจร a to d ที่ออกแบบเพื่อใช้กับ at90s2313 มีอุปกรณ์ต่าง ๆ ดังนี้

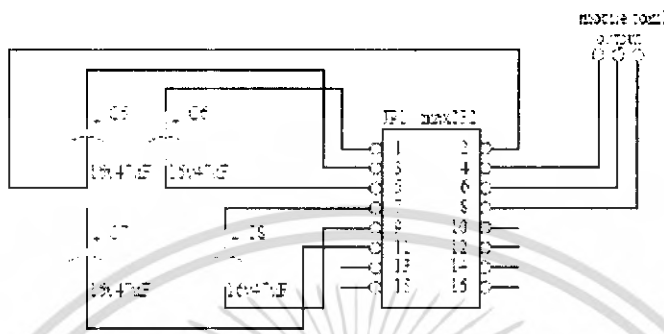


รูปที่ 3.4 วงจรในส่วนแปลงค่าแรงดันไปเป็นสัญญาณดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

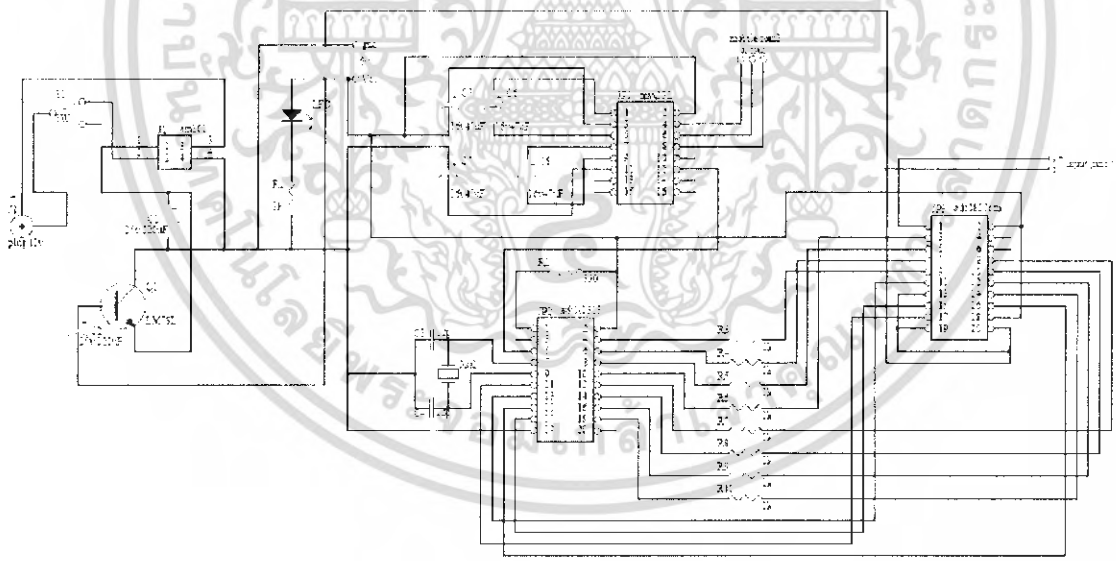
3.5 ส่วนเชื่อมต่อกับโทรศัพท์เคลื่อนที่

ในส่วนนี้จะใช้ max232 ในการทำหน้าที่เชื่อมต่อกับโทรศัพท์เคลื่อนที่ แล้วในส่วนนี้ยังใช้ในการเชื่อมต่อ กับคอมพิวเตอร์ เพื่อที่จะทดสอบว่า at90s2313 สามารถที่จะส่งข้อมูลต่างๆ ออกมาได้อย่างถูกต้องอีกด้วย การเชื่อมต่ออุปกรณ์ต่างๆ ในส่วนนี้เป็น ดังนี้



รูปที่ 3.5 วงจรส่วนเชื่อมต่อกับโทรศัพท์เคลื่อนที่

นำส่วนต่างๆ มาต่อเข้าด้วยกันจะได้จะได้ ดังนี้



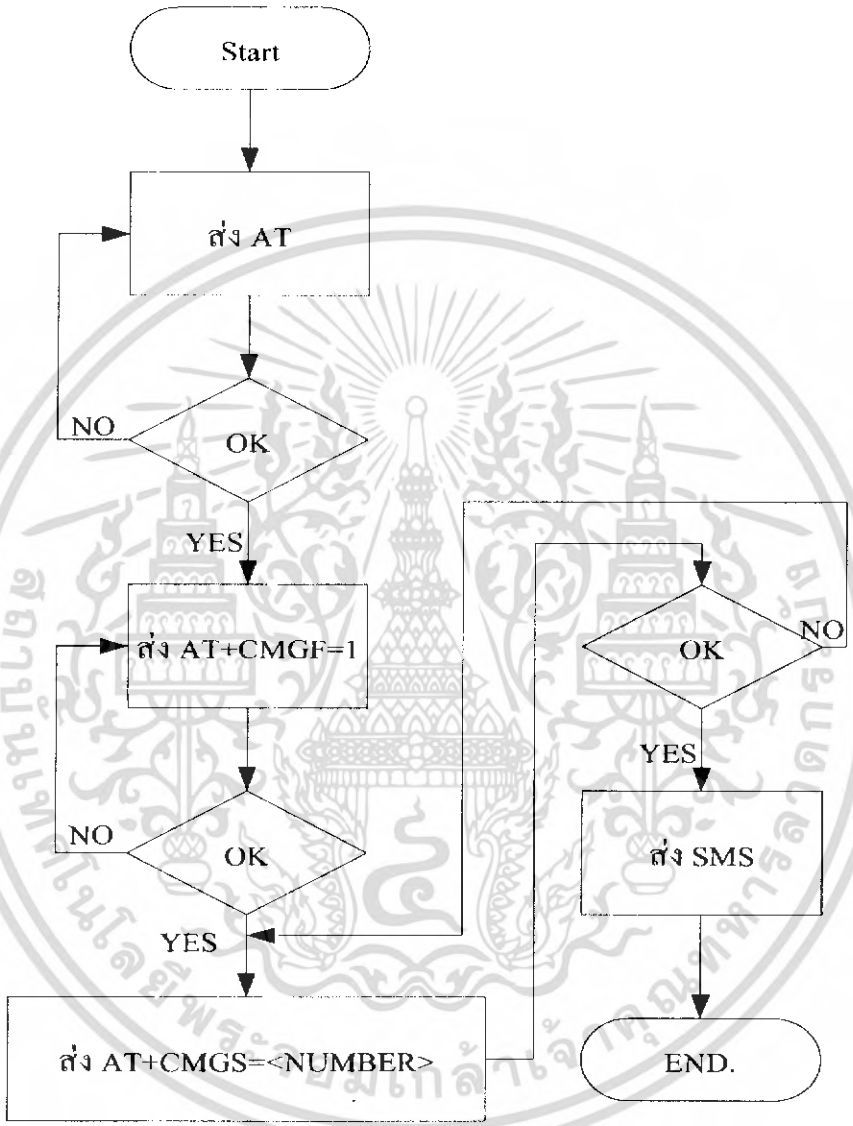
รูปที่ 3.6 วงจรเสร็จสมบูรณ์

โครงสร้างของไอซีต่าง ๆ ที่ใช้ในโครงงานสามารถดูได้จาก datasheet ในภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 แผนผังการทำงาน

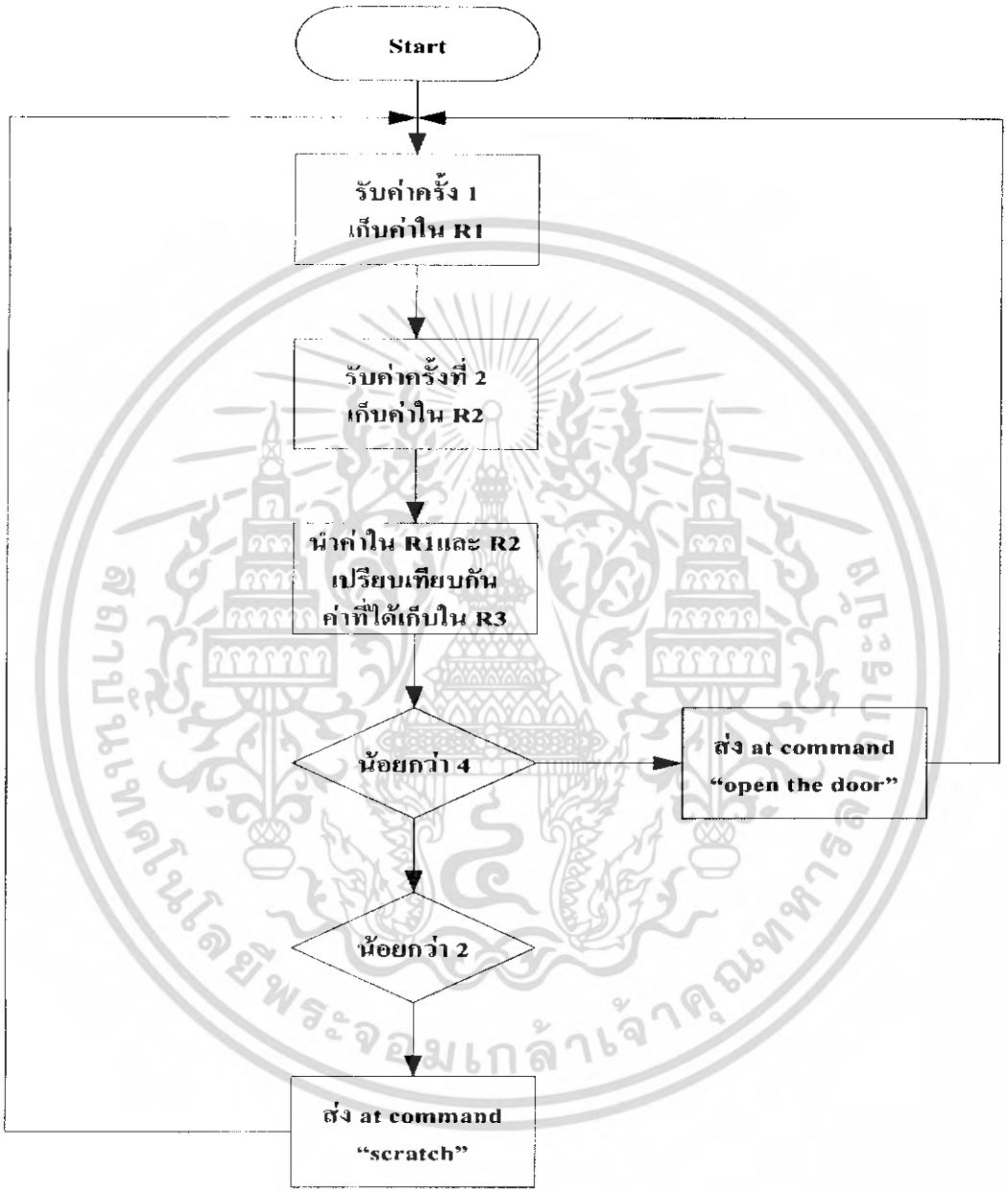
1. การตรวจสอบสถานะโทรศัพท์เคลื่อนที่ว่าพร้อมที่จะทำงานหรือไม่



รูปที่ 3.7 แผนผังการทำงานของส่วนตรวจสอบสถานะมือถือและ Cell site

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การทำงานของส่วนควบคุมส่งข้อความเตือน



รูปที่ 3.8 รูปแผนผังการทำงานของส่วนควบคุมการส่งข้อความสั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
โปรแกรมที่ใช้ควบคุมการทำงาน

```
.include "2313def.inc"

.def    baudRate        =R23

.def    bitcnt          =R16    ;bit counter

.def    Txbyte          =R18    ;Data to be transmitted
.def    RXbyte          =R19    ;Received data

;.def   sTX_byte        =R20    ;Register used for UART TX/RX
;.def   sRX_byte        =R21
.def    itemp           =R22

.def    threshold       =R21
.def    threshold2      =R20
.def    prev            =R24
.def    next            =R25

;************************************************
;register declared for Delay subroutine
.def    counter =       r17
;************************************************
.equ    RxD     =0              ;Receive pin is PD0
.equ    TxD     =1              ;Transmit pin is PD1
;************************************************
;register and const declared for lcd subroutine
.equ    rs      =       3
.equ    rw      =       4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
.equ    e       =       5


.equ    d0      =       0
.equ    d1      =       1
.equ    d2      =       2
.equ    d3      =       3
.equ    d4      =       4
.equ    d5      =       5
.equ    d6      =       6
.equ    d7      =       7
;***********************************************
.org 0
        rjmp    main

main:
        ldi     itemp,  low(ramend)             ;MUST set STACK
        out     spl,    itemp

        rcall Delay_100us                       ;Delay waiting for all system ready
        rcall Delay_100us

        rcall reset                             ;Reset and Preset some register


start_adc:
        ldi threshold,$10                       ;level threshold level
        ldi     threshold2,$20

        rcall call_adc                          ;start ADC
        mov next,Txbyte                         ;keep data in r3 for looping check
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
adc_loop:
        mov prev,next                   ;Transfer the newest data to r2
        rcall reset
        rcall call_adc                  ;start ADC and always keep new data into r3
        mov next,Txbyte

        sub prev,next                   ;r2 = r2-r3

        cp prev,threshold               ;r2 = r2-r1
        brge send_com                   ;if r2 > 0 --> go to send_com
        rcall adc_loop                  ;if r2 < 0 - NO PROBLEM - go to detect ADC
                                         loop again and again

send_com:
        rcall prog_loop                 ;There is some problem - PRESET GPRS
                                         BOARD

        cp prev,threshold2              ;r2 = r2-r1
        brge send_com_trouble           ;if r2 > 0 --> system have the voltage level too
                                         much different (more than 2*threshold) -->
                                         send text2 (SERIOUS)
        rcall send_text1                ;if r2 < 0 --> system have voltage higher than
                                         level but still under 1 threshold
        rjmp adc_loop                   ;go to check ADC again

send_com_trouble:
        rcall send_text2                ;voltage detect higher than threshold (>
2*threshold) - SERIOUS LEVEL
        rjmp adc_loop
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;#############################################
;SMS sending
;#############################################
prog_loop:

        ;test at send                    ;SEND AT<ENTER>
        ldi ZH,high(2*at_command1)       ;Load pointer that point to label
"at_command1"
        ldi ZL,low(2*at_command1)        ;Need to load using high and low command
        rcall send_at                    ;Now Jump to send data out of microcontroller
        ;end test

        rcall Delay_4s

        ldi ZH,high(2*at_command2)       ;SEND AT+CMGF=1<ENTER>
        ldi ZL,low(2*at_command2)
        rcall send_at

        rcall Delay_4s

        ldi ZH,high(2*at_command3)       ;SEND AT+CMGS=+66813761926
        ldi ZL,low(2*at_command3)
        rcall send_at


        rcall Delay_4s
        ret

send_text1:
        ldi ZH,high(2*problem1)          ;SEND Problem Occur
        ldi ZL,low(2*problem1)
        rcall send_at
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        rcall Delay_1s
        rcall Delay_1s

        ret

send_text2:
        ldi ZH,high(2*problem2)          ;SEND Serious Problem
        ldi ZL,low(2*problem2)
        rcall send_at

        rcall Delay_1s
        rcall Delay_1s

        ret

;#########################################
;ADC checking
;#########################################
call_adc:
.equ    cs      = 4
.equ    rd      = 5
.equ    rdy     = 3
.equ    int     = 2
snd_cs: cbi PORTD,cs             ;lower CS BIT - Start to wake up ADC
chkrdy: in itemp,PIND            ;WAITING for RDY SIGNAL from ADC
        andi itemp,$08
        brne chkrdy

snd_rd: cbi PORTD,rd             ;GOT RDY - NOW send RD SIGNAL to ASK for DATA
chkint: in itemp,PIND            ;WAITING for INT SIGNAL from ADC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        andi itemp,$04
        brne chkint

start_rcv:
        in Txbyte, PINB             ;GOT INT - NOW ADC send the DATA

        sbi PORTD,rd                ;SET RD BIT - Tell ADC that finish reading data
        rcall Delay_100us
        sbi PORTD,cs                ;SET CS BIT - To tell ADC to take a rest
        rcall Delay_100us

        ret

reset:
        ldi itemp, $F3
        out DDRD, itemp             ;set portD as a output port
        ldi itemp, $FF
        out PORTD, itemp

        ldi itemp, $00
        out DDRB, itemp
        ldi itemp, $FF
        out PORTB, itemp            ;set portB as a input port

        sbi     PORTD,TxD           ;Init port pins
        sbi     DDRD,TxD

        ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;=====================================================

; Module: Delay 1us, 10us, 100us

; By: Thada Soongsongkunnatum

; Create Date: 20-09-2003

; Version: 1.0.0

; Module: AT90S2313, 8MHz Crystal

; 12 Cycle/1 instruction

;=====================================================

Delay_4s:

        rcall Delay_1s

        rcall Delay_1s

        rcall Delay_1s

        rcall Delay_1s

        ret


Delay_001us:

        ldi     counter, 3

Delay_001us_loop:

        dec     counter

        brne    Delay_001us_loop

        ret


Delay_010us:

        ldi     counter, 38

Delay_010us_loop:

        dec     counter

        brne    Delay_010us_loop

        nop

        ret


Delay_100us:

        ldi     counter, 0x09
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Delay_100us_loop:
        push    counter
        rcall   Delay_010us
        pop             counter

        push counter
Delay_100us_Calib:
        ldi     counter, 0x3
Delay_100us_Calib_loop:
        dec     counter
        brne    Delay_100us_Calib_loop
        pop counter

        dec     counter
        brne    Delay_100us_loop
        ret

Delay_1ms:
        ldi     counter, 0x0A
ms1_Call_Delay_100us:
        push    counter
        rcall   Delay_100us
        pop     counter
        dec     counter
        brne    ms1_Call_Delay_100us
        ret

Delay_10ms:
        ldi     counter, 0x64
ms10_Call_Delay_100us:
        push    counter
        rcall   Delay_100us
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        pop     counter
        dec     counter
        brne    ms10_Call_Delay_100us
        ret

Delay_100ms:
        ldi     counter, 0x0A
ms100_Call_Delay_10ms:
        push    counter
        rcall   Delay_10ms
        pop     counter
        dec     counter
        brne    ms100_Call_Delay_10ms
        ret

Delay_1s:
        ldi     counter, 0x0A
s1_Call_Delay_100ms:
        push    counter
        rcall   Delay_100ms
        pop     counter
        dec     counter
        brne    s1_Call_Delay_100ms
        ret
;***************************************

;*********************************************

;
;       Put Character to RS232
;
;*********************************************

.equ            sb      =1              ;Number of stop bits (1, 2, ...)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
putchar:;cli
                ldi     bitcnt,9+sb     ;1+8+sb (sb is # of stop bits)
                com     Txbyte          ;Inverte everything
                sec                     ;Start bit

putchar0:       brcc    putchar1        ;If carry set
                cbi     PORTD,TxD       ;   send a '0'
                rjmp    putchar2        ;else

putchar1:       sbi     PORTD,TxD       ;   send a '1'
                nop

putchar2:       rcall   UART_delay      ;One bit delay
                rcall   UART_delay

                lsr     Txbyte          ;Get next bit
                dec     bitcnt          ;If not all bit sent
                brne    putchar0        ;   send next
                ;sei                    ;else
                ret                     ;   return

;***********************************************
;
;       Get Character from RS232
;
;***********************************************
getchar: ldi    bitcnt,9 ;8 data bit + 1 stop bit

getchar1:       sbic    PINB,RxD        ;Wait for start bit
                rjmp    getchar1
                rcall   UART_delay      ;0.5 bit delay
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
getchar2:       rcall   UART_delay      ;1 bit delay
                rcall   UART_delay
                clc                     ;clear carry
                sbic    PINB,RxD        ;if RX pin high
                sec                     ;
                dec     bitcnt          ;If bit is stop bit
                breq    getchar3        ;  return
                                        ;else
                ror     Rxbyte          ;  shift bit into Rxbyte
                rjmp    getchar2        ;  go get next
getchar3:       ret

;*****************************************************************************
*
UART_delay:     ldi     itemp,a
UART_delay1:    dec     itemp
                brne    UART_delay1

                ret

;*****************************************************************************
*
;a=19 is used for 8.00MHz   at baudrate 57600 bps
.equ    a       =137


conv_H2T:
        push    Txbyte
        push    itemp
        mov     Txbyte,r0
        swap    Txbyte
        andi    Txbyte,0x0F
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ldi     itemp,0x30
        add     Txbyte,itemp
        ldi     itemp,0x39
        cp      Txbyte,itemp
        brlt    next_conv_H2T_01
        ldi     itemp,0x07
        add     Txbyte,itemp
next_conv_H2T_01:
        rcall   putchar
        mov     Txbyte,r0
        andi    Txbyte,0x0F
        ldi     itemp,0x30
        add     Txbyte,itemp
        ldi     itemp,0x39
        cp      Txbyte,itemp
        brlt    next_conv_H2T_02
        ldi     itemp,0x07
        add     Txbyte,itemp
next_conv_H2T_02:
        rcall   putchar
        pop     itemp
        pop     Txbyte
        ret



send_at:
        lpm                         ;load data that ZL and ZH point to r0
        tst r0                      ;test that r0 is not zero, if r0 = zero -> end
                                    sending data
        breq end_sending            ;r0 not equal to zero -> send data
        mov Txbyte,r0               ;move data to Txbyte
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        rcall putchar                   ;send data to RS232
        adiw ZL,1                       ;increase pointer to next character
        rjmp send_at                    ;send next character -> until find zero (end of
                                        sentences)
end_sending:
        ret

at_command1:
        .DB "AT",$0D,$0A,0,0
at_command2:
        .DB "AT+CMGF=1",$0D,$0A,0
at_command3:
        .DB "AT+CMGS=+66813761926",$0D,$0A,0,0
problem1:
        .DB "Problem Occur",$1A,$0D,$0A,0,0
problem2:
        .DB "Serious Problem",$1A,$0D,$0A,0,0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลอง

1. จากการที่ลองติดตัวจับสัญญาณ (Microphone) ในจุดที่ต่างๆ กัน เช่น ซ่อนในกระโปรงหลัง โดยติด Microphone กับตัวถังรถ ที่ประตู และก้านพวงมาลัย จุดที่สามารถตรวจจับสัญญาณได้ดีที่สุด คือ ที่ตำแหน่งตรงประตู เนื่องจากเป็นเหล็กชิ้นเดียวกับที่เกิดการสั่นสะเทือน

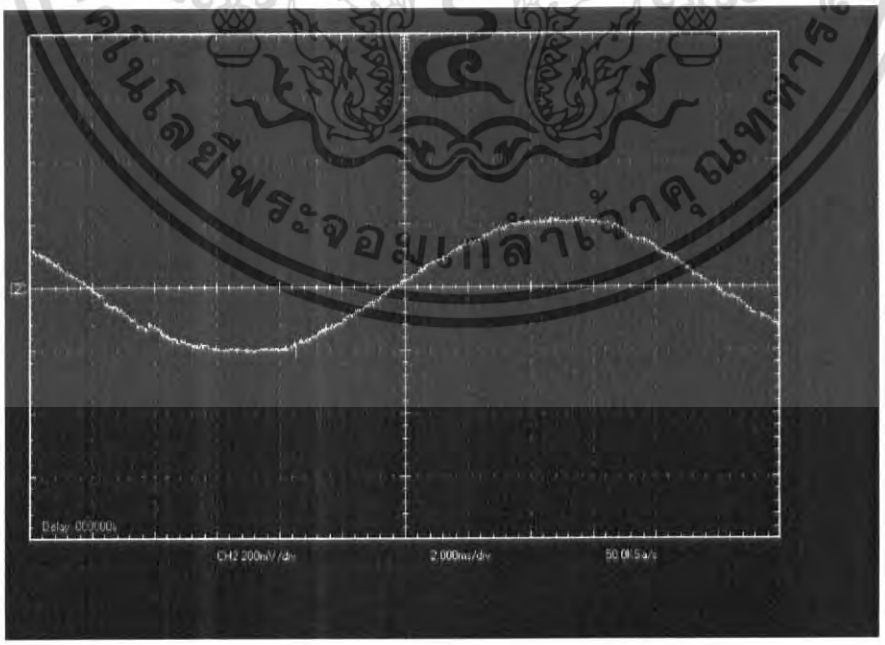
2. จากการที่ลองจับสัญญาณโดยการทดลองด้วยวิธีการต่างๆ เช่น การเปิดประตูรถ การขูดในตำแหน่งต่างๆ ของรถจะได้สัญญาณในลักษณะที่แตกต่างกัน

3. สัญญาณที่ได้สามารถที่จะนำมาวิเคราะห์และนำมาใช้ให้ ไมโครคอนโทรเลอร์ตัดสินใจว่าเป็นสถานการณ์ที่ต้องส่งสัญญาณเตือนออกไปหรือไม่

4. สัญญาณที่เป็นเหตุการณ์ปกติ จะมีค่า ประมาณ 0 – 1 V สัญญาณที่เริ่มมีการขูดรถ จะมีขนาดแรงดันเพิ่มขึ้นเป็น ประมาณ 2.5 – 3 V ส่วนการที่การเปิดประตู นั้นจะได้ค่าที่มากที่สุด คือ ประมาณ 4.5 – 5 V

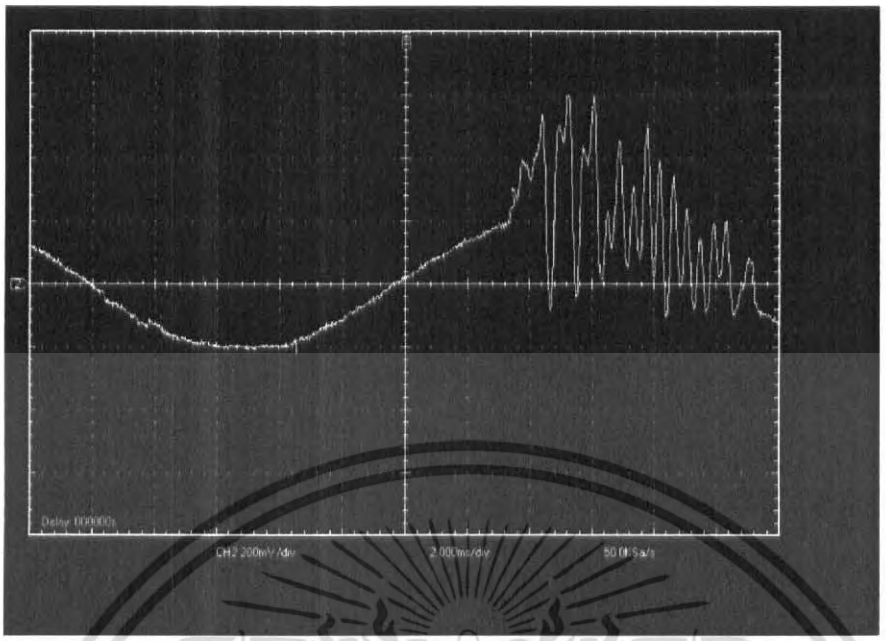
5. ในการที่จะส่งคำสั่งให้มือถือส่งข้อความไปยังหมายเลขเป้าหมายนั้น จะต้องทำการสลับคำสั่งด้วยวิธีการ Nibble ก่อนจึงจะสามารถสั่งให้มือถือส่งข้อความได้

6. การส่งคำสั่งต่างๆ ระหว่างอุปกรณ์ จะเซตค่าความเร็ว ให้การส่งให้มีความเร็วเท่ากันในทุกอุปกรณ์ก่อน จึงจะสามารถติดต่อกันได้



รูปที่ 5.1 สัญญาณขณะที่ไม่มี input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 สัญญาณขณะเปิดประตู

รูปที่ 5.3 สัญญาณขณะมีการขูดใต้บังโคลนรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ลัญฉกร วุฒิสิทธิกุลกิจ,"หลักการระบบโทรศัพท์เคลื่อนที่", สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย, 2542
2. พันธ์ศักดิ์ พุฒิมานิตพงศ์,"โทรคมนาคมเบื้องต้น", สำนักพิมพ์ศูนย์ส่งเสริมอาชีวะ, 2546
3. สุนทร วิทูสุรพจน์, "การโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล 8051" , กรุงเทพฯ, บ.ซีเอ็ดยูเคชั่น ,2537.
4. อุดม จีนประดับ,"Microcontroller MCS51",กรุงเทพฯ, สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2541.
5. ไกรวุฒิ โรจน์ประเสริฐสุด,"ไมโครโปรเซสเซอร์2", บ.อีทีที, กรุงเทพฯ,2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

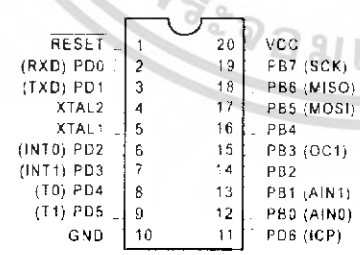
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- Utilizes the AVR® RISC Architecture
- AVR – High-performance and Low-power RISC Architecture
  - 118 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General-purpose Working Registers
  - Up to 10 MIPS Throughput at 10 MHz
- Data and Nonvolatile Program Memory
  - 2K Bytes of In-System Programmable Flash
    Endurance 1,000 Write/Erase Cycles
  - 128 Bytes of SRAM
  - 128 Bytes of In-System Programmable EEPROM
    Endurance: 100,000 Write/Erase Cycles
  - Programming Lock for Flash Program and EEPROM Data Security
- Peripheral Features
  - One 8-bit Timer/Counter with Separate Prescaler
  - One 16-bit Timer/Counter with Separate Prescaler, Compare, Capture Modes and 8-, 9- or 10-bit PWM
  - On-chip Analog Comparator
  - Programmable Watchdog Timer with On-chip Oscillator
  - SPI Serial Interface for In-System Programming
  - Full Duplex UART
- • Special Microcontroller Features
  - Low-power Idle and Power-down Modes
  - External and Internal Interrupt Sources
- • Specifications
  - Low-power, High-speed CMOS Process Technology
  - Fully Static Operation
- Power Consumption at 4 MHz, 3V, 25°C
  - Active: 2.8 mA
  - Idle Mode: 0.8 mA
  - Power-down Mode: <1 µA
- I/O and Packages
  - 15 Programmable I/O Lines
  - 20-pin PDIP and SOIC
- Operating Voltages
  - 2.7 - 6.0V (AT90S2313-4)
  - 4.0 - 6.0V (AT90S2313-10)
- Speed Grades
  - 0 - 4 MHz (AT90S2313-4)
  - 0 - 10 MHz (AT90S2313-10)

Pin Configuration

PDIP/SOIC



8-bit AVR® Microcontroller with 2K Bytes of In-System Programmable Flash

AT90S2313

Rev. 0839G–08/01

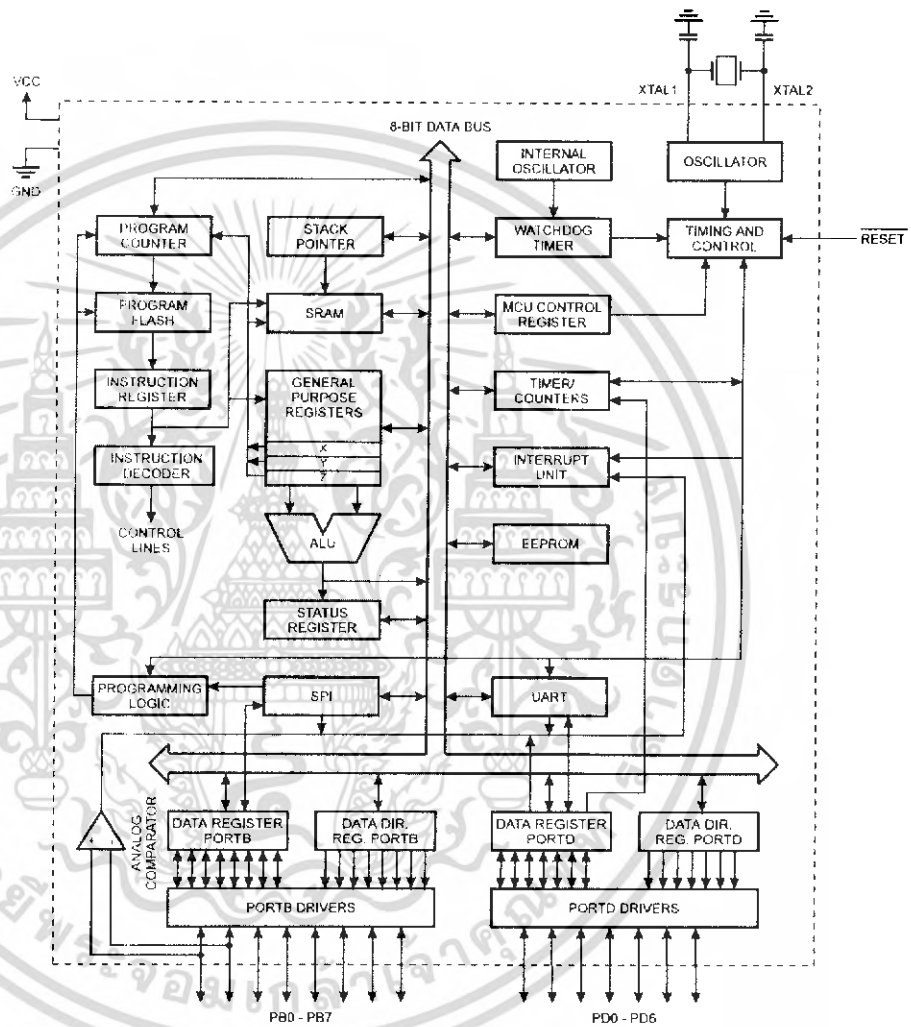
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Description

The AT90S2313 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the AT90S2313 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

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.

Figure 1. The AT90S2313 Block Diagram



The AT90S2313 provides the following features: 2K bytes of In-System Programmable Flash, 128 bytes EEPROM, 128 bytes SRAM, 15 general-purpose I/O lines, 32 general-purpose working registers, flexible timer/counters with compare modes, internal and external interrupts, a programmable serial UART, programmable Watchdog Timer with internal oscillator, an SPI serial port for Flash memory downloading and two software

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 external interrupt or hardware reset.

The device is manufactured using Atmel's high-density nonvolatile memory technology. The on-chip In-System Programmable Flash allows the program memory to be reprogrammed in-system through an SPI serial interface or by a conventional nonvolatile memory programmer. By combining an enhanced RISC 8-bit CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT90S2313 is a powerful microcontroller that provides a highly flexible and cost-effective solution to many embedded control applications.

The AT90S2313 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.

Pin Descriptions

VCC

Supply voltage pin.

GND

Ground pin.

Port B (PB7..PB0)

Port B is an 8-bit bi-directional I/O port. Port pins can provide internal pull-up resistors (selected for each bit). PB0 and PB1 also serve as the positive input (AIN0) and the negative input (AIN1), respectively, of the on-chip analog comparator. The Port B output buffers can sink 20 mA and can drive LED displays directly. When pins PB0 to PB7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not active.

Port B also serves the functions of various special features of the AT90S2313 as listed on page 53.

Port D (PD6..PD0)

Port D has seven bi-directional I/O ports with internal pull-up resistors, PD6..PD0. The Port D output buffers can sink 20 mA. 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 active.

Port D also serves the functions of various special features of the AT90S2313 as listed on page 58.

$\overline{\text{RESET}}$

Reset input. A low level on this pin for more than 50 ns will generate a reset, even if the clock is not running. 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

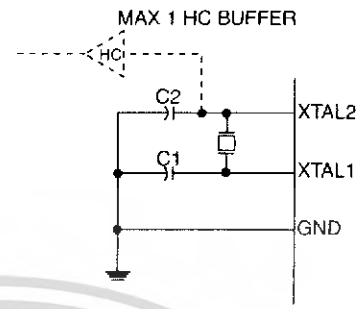
Output from the inverting oscillator amplifier.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Crystal Oscillator

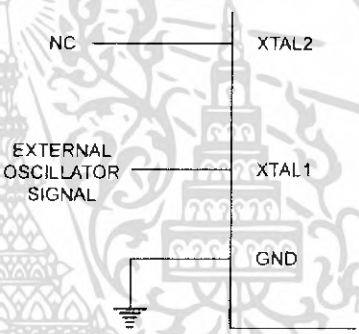
XTAL1 and XTAL2 are input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 2. Either a quartz crystal or a ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, as shown in Figure 3.

Figure 2. Oscillator Connections



Note: When using the MCU Oscillator as a clock for an external device, an HC buffer should be connected as indicated in the figure.

Figure 3. External Clock Drive Configuration

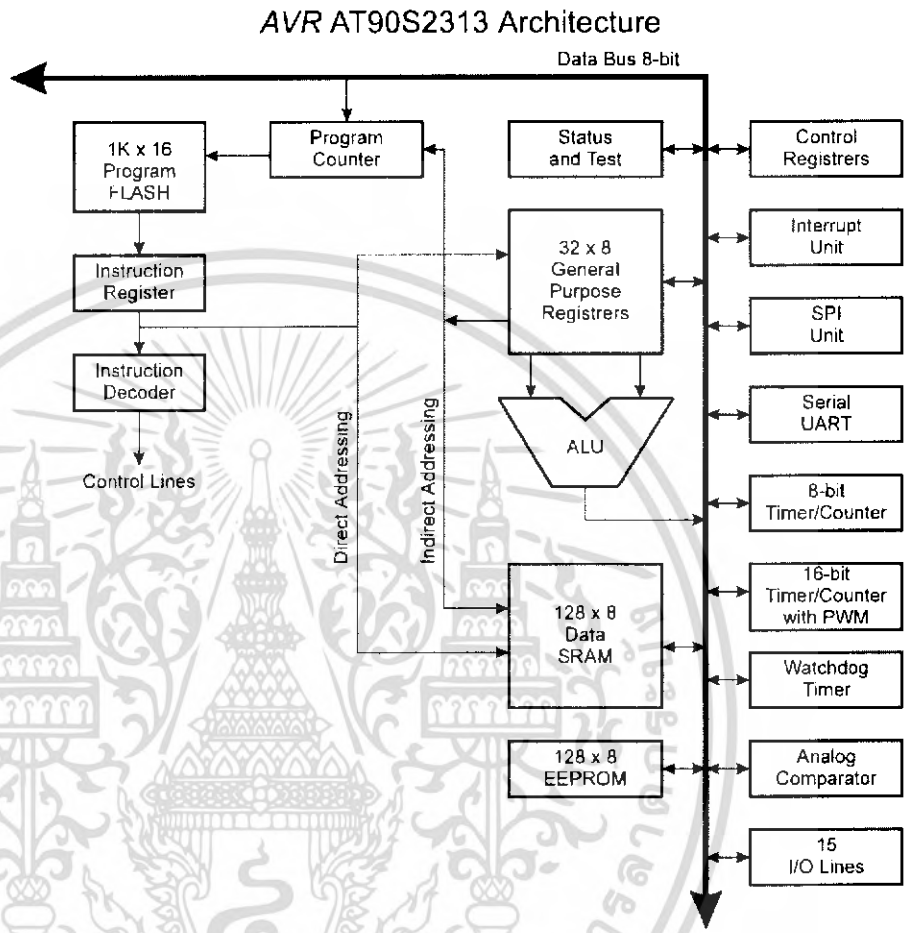


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Architectural Overview

The fast-access register file concept contains 32 x 8-bit general-purpose working registers with a single clock cycle access time. This means that during one single clock cycle, one ALU (Arithmetic Logic Unit) operation is executed. Two operands are output from the register file, the operation is executed, and the result is stored back in the register file – in one clock cycle.

Figure 4. The AT90S2313 AVR RISC Architecture



Six of the 32 registers can be used as three 16-bit indirect address register pointers for Data Space addressing – enabling efficient address calculations. One of the three address pointers is also used as the address pointer for the constant table look-up function. These added function registers are the 16-bit X-register, Y-register and Z-register.

The ALU supports arithmetic and logic functions between registers or between a constant and a register. Single register operations are also executed in the ALU. Figure 4 shows the AT90S2313 AVR RISC microcontroller architecture.

In addition to the register operation, the conventional memory addressing modes can be used on the register file as well. This is enabled by the fact that the register file is assigned the 32 lowermost Data Space addresses ($00 - $1F), allowing them to be accessed as though they were ordinary memory locations.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The I/O memory space contains 64 addresses for CPU peripheral functions such as control registers, timer/counters, A/D converters and other I/O functions. The I/O memory can be accessed directly or as the Data Space locations following those of the register file, $20 - $5F.

The AVR has Harvard architecture – with separate memories and buses for program and data. The program memory is accessed with a 2-stage pipeline. While one instruction is being executed, the next instruction is pre-fetched from the program memory. This concept enables instructions to be executed in every clock cycle. The program memory is In-System Programmable Flash memory.

With the relative jump and call instructions, the whole 1K address space is directly accessed. Most AVR instructions have a single 16-bit word format. Every program memory address contains a 16- or 32-bit instruction.

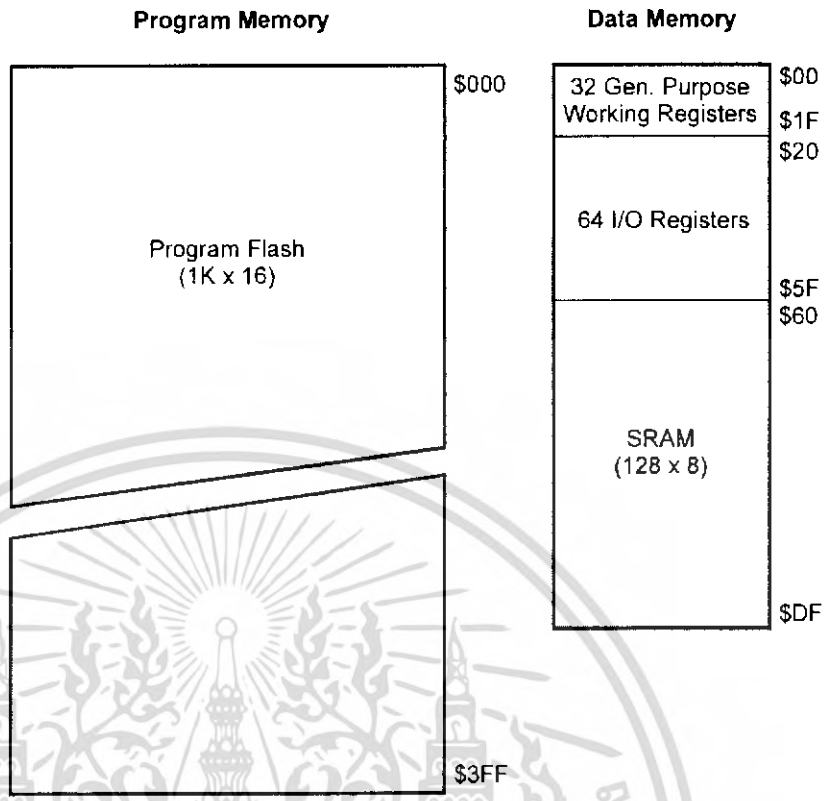
During interrupts and subroutine calls, the return address Program Counter (PC) is stored on the stack. The stack is effectively allocated in the general data SRAM, and consequently the stack size is only limited by the total SRAM size and the usage of the SRAM. All user programs must initialize the SP in the reset routine (before subroutines or interrupts are executed). The 8-bit stack pointer (SP) is read/write accessible in the I/O space.

The 128 bytes data SRAM + register file and I/O registers can be easily accessed through the five different addressing modes supported in the AVR architecture.

The memory spaces in the AVR architecture are all linear and regular memory maps.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 5. Memory Maps



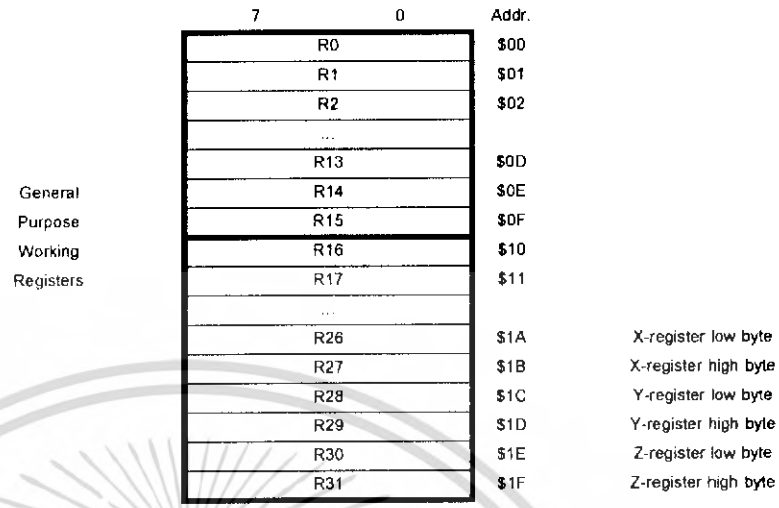
A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the Status Register. All the different interrupts have a separate interrupt vector in the interrupt vector table at the beginning of the program memory. The different interrupts have priority in accordance with their interrupt vector position. The lower the interrupt vector address, the higher the priority.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

General-purpose Register File

Figure 6 shows the structure of the 32 general-purpose registers in the CPU.

Figure 6. AVR CPU General-purpose Working Registers



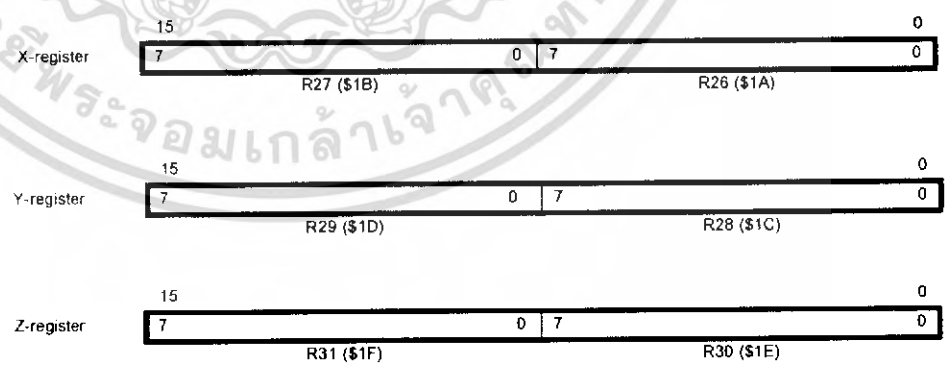
All the register operating instructions in the instruction set have direct and single-cycle access to all registers. The only exception is the five constant arithmetic and logic instructions SBCI, SUBI, CPI, ANDI, ORI between a constant and a register and the LDI instruction for load immediate constant data. These instructions apply to the second half of the registers in the register file (R16..R31). The general SBC, SUB, CP, AND, OR and all other operations between two registers or on a single register apply to the entire register file.

As shown in Figure 6, each register is also assigned a data memory address, mapping them directly into the first 32 locations of the user Data Space. Although the register file is not physically implemented as SRAM locations, this memory organization provides great flexibility in access of the registers, as the X, Y and Z registers can be set to index any register in the file.

X-register, Y-register, and Z-register

The registers R26..R31 have some added functions to their general-purpose usage. These registers are the address pointers for indirect addressing of the Data Space. The three indirect address registers X, Y and Z are defined in Figure 7.

Figure 7. X, Y and Z Registers



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

In the different addressing modes these address registers have functions as fixed displacement, automatic increment and decrement (see the descriptions for the different instructions).

ALU – Arithmetic Logic Unit

The high-performance AVR ALU operates in direct connection with all the 32 general-purpose working registers. Within a single clock cycle, ALU operations between registers in the register file are executed. The ALU operations are divided into three main categories – arithmetic, logical and bit functions.

In-System Programmable Flash Program Memory

The AT90S2313 contains 2K bytes on-chip In-System Programmable Flash memory for program storage. Since all instructions are 16- or 32-bit words, the Flash is organized as 1K x 16. The Flash memory has an endurance of at least 1,000 write/erase cycles.

The AT90S2313 Program Counter (PC) is 10 bits wide, thus addressing the 1,024 program memory addresses.

See page 62 for a detailed description on Flash data downloading. See page 11 for the different addressing modes.

EEPROM Data Memory

The AT90S2313 contains 128 bytes of EEPROM data memory. It is organized as a separate data space in which single bytes can be read and written. The EEPROM has an endurance of at least 100,000 write/erase cycles. The access between the EEPROM and the CPU is described on page 40, specifying the EEPROM address register, the EEPROM data register and the EEPROM control register.

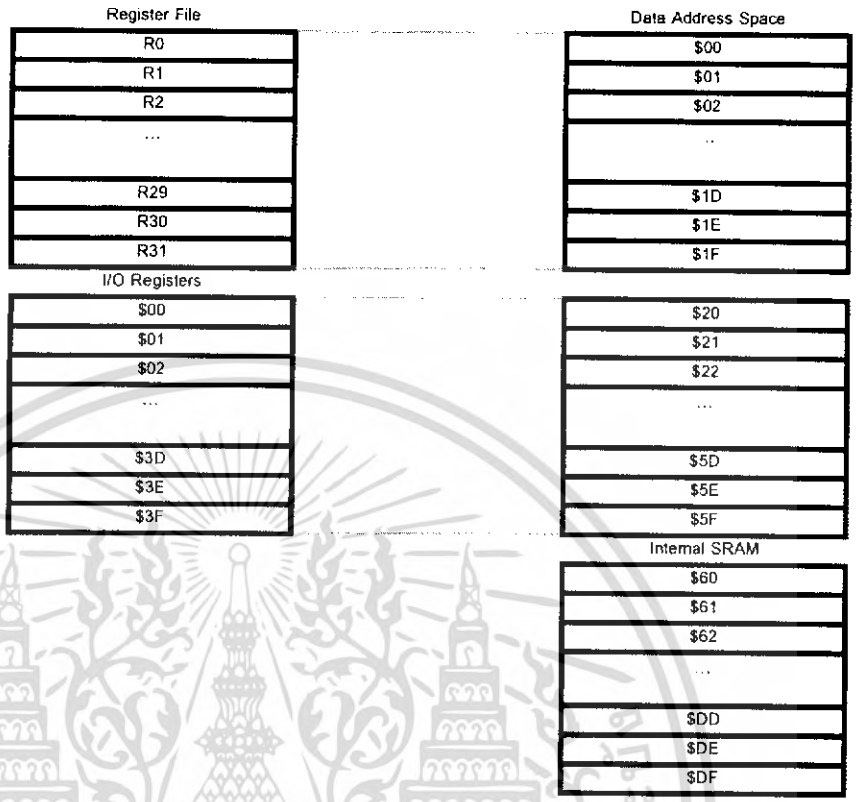
For the SPI data downloading, see page 69 for a detailed description.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SRAM Data Memory

Figure 8 shows how the AT90S2313 data memory is organized.

Figure 8. SRAM Organization



The 224 data memory locations address the Register file, I/O memory and the data SRAM. The first 96 locations address the Register File + I/O Memory, and the next 128 locations address the data SRAM.

The five different addressing modes for the data memory cover: Direct, Indirect with Displacement, Indirect, Indirect with Pre-decrement and Indirect with Post-increment. In the register file, registers R26 to R31 feature the indirect addressing pointer registers.

The Direct addressing reaches the entire data address space.

The Indirect with Displacement mode features 63 address locations reached from the base address given by the Y and Z registers.

When using register indirect addressing modes with automatic pre-decrement and post-increment, the address registers X, Y and Z are used and decremented and incremented.

The 32 general-purpose working registers, 64 I/O registers and the 128 bytes of data SRAM in the AT90S2313 are all directly accessible through all these addressing modes.

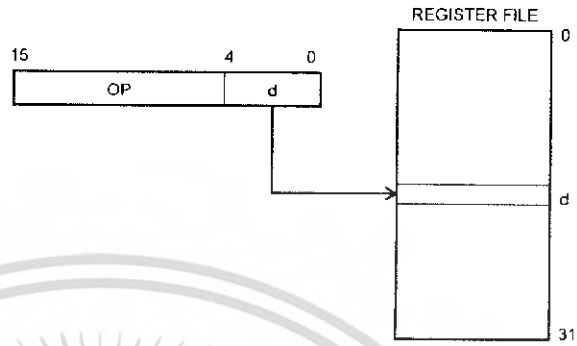
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Program and Data Addressing Modes

The AT90S2313 AVR RISC microcontroller supports powerful and efficient addressing modes for access to the program memory (Flash) and data memory. This section describes the different addressing modes supported by the AVR architecture. In the figures, OP means the operation code part of the instruction word. To simplify, not all figures show the exact location of the addressing bits.

Register Direct, Single Register Rd

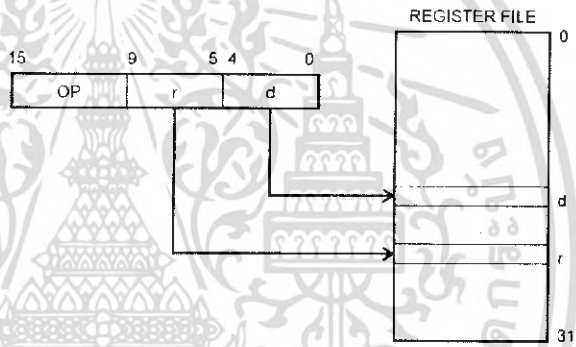
Figure 9. Direct Single Register Addressing



The operand is contained in register d (Rd).

Register Direct, Two Registers Rd and Rr

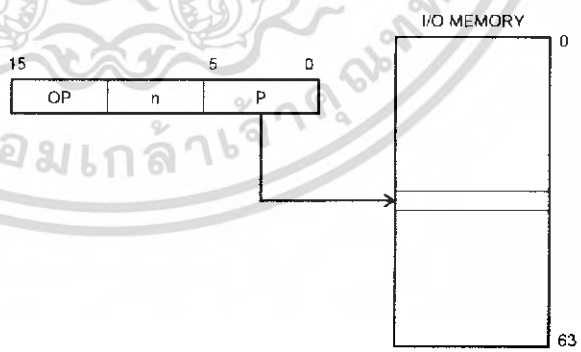
Figure 10. Direct Register Addressing, Two Registers



Operands are contained in register r (Rr) and d (Rd). The result is stored in register d (Rd).

I/O Direct

Figure 11. I/O Direct Addressing

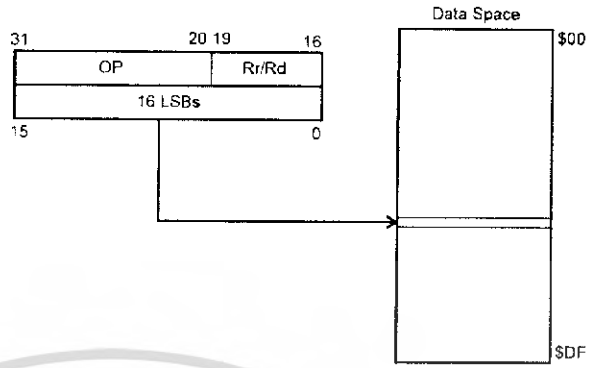


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Operand address is contained in 6 bits of the instruction word. n is the destination or source register address.

Data Direct

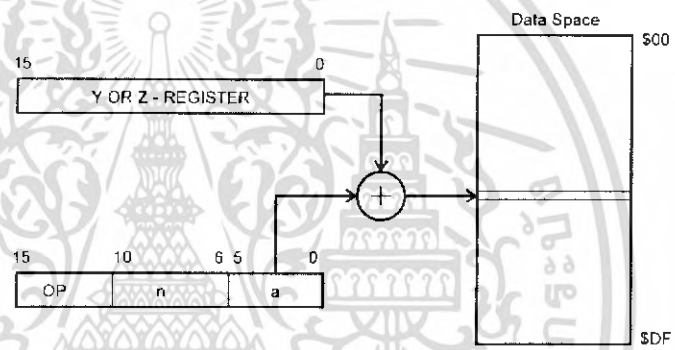
Figure 12. Direct Data Addressing



A 16-bit data address is contained in the 16 LSBs of a 2-word instruction. Rd/Rr specify the destination or source register.

Data Indirect with Displacement

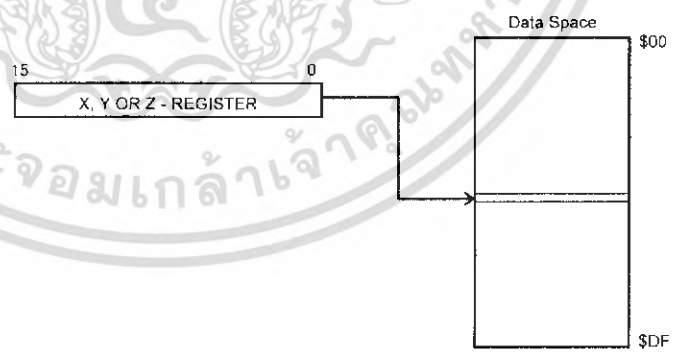
Figure 13. Data Indirect with Displacement



Operand address is the result of the Y- or Z-register contents added to the address contained in 6 bits of the instruction word.

Data Indirect

Figure 14. Data Indirect Addressing

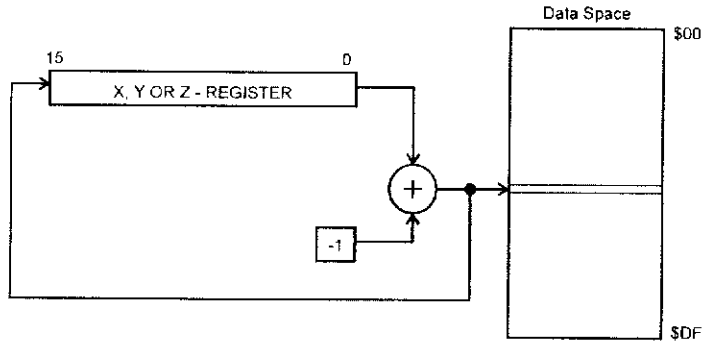


Operand address is the contents of the X-, Y- or Z-register.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data Indirect with Pre-decrement

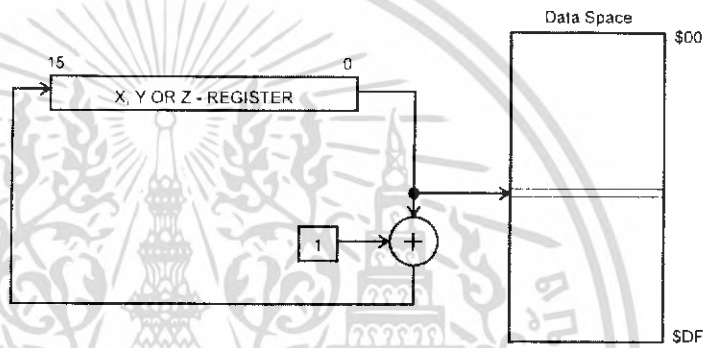
Figure 15. Data Indirect Addressing with Pre-decrement



The X-, Y- or Z-register is decremented before the operation. Operand address is the decremented contents of the X-, Y- or Z-register.

Data Indirect with Post-increment

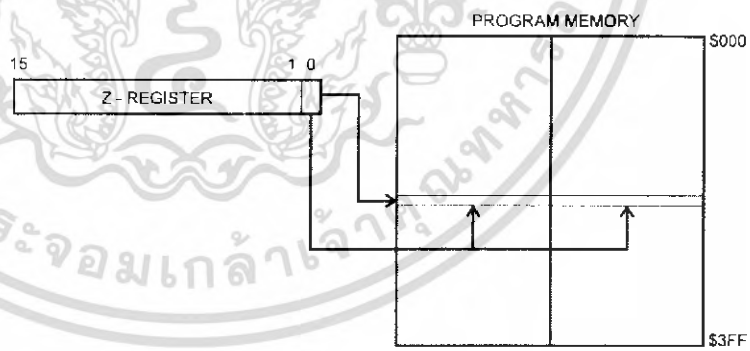
Figure 16. Data Indirect Addressing with Post-increment



The X-, Y- or Z-register is incremented after the operation. Operand address is the contents of the X-, Y- or Z-register prior to incrementing.

Constant Addressing Using the LPM Instruction

Figure 17. Code Memory Constant Addressing

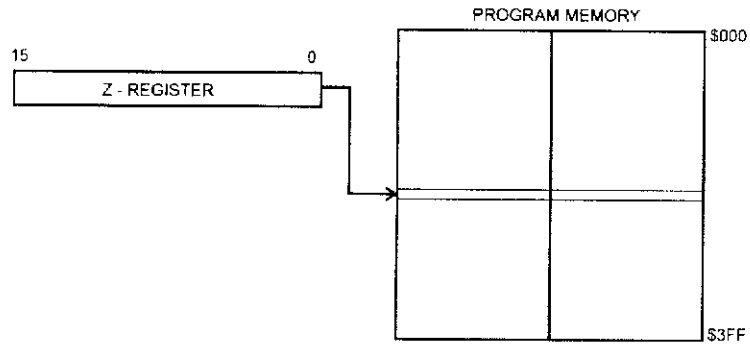


Constant byte address is specified by the Z-register contents. The 15 MSBs select word address (0 - 1K), the LSB selects low byte if cleared (LSB = 0) or high byte if set (LSB = 1).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Indirect Program Addressing, IJMP and ICALL

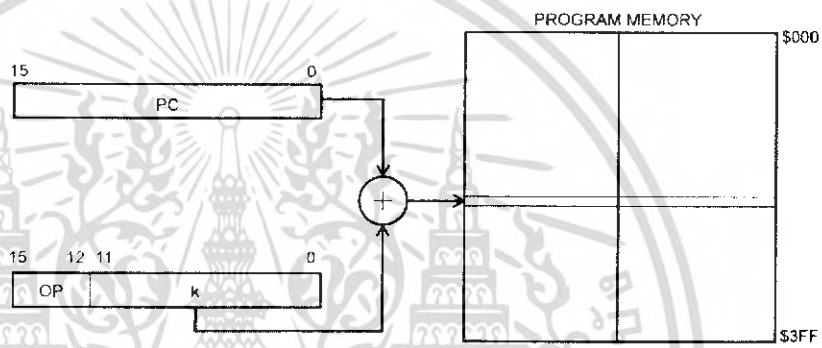
Figure 18. Indirect Program Memory Addressing



Program execution continues at address contained by the Z-register (i.e., the PC is loaded with the contents of the Z-register).

Relative Program Addressing, RJMP and RCALL

Figure 19. Relative Program Memory Addressing



Program execution continues at address PC + k + 1. The relative address k is -2048 to 2047.

Memory Access and Instruction Execution Timing

This section describes the general access timing concepts for instruction execution and internal memory access.

The AVR CPU is driven by the System Clock Ø, directly generated from the external clock crystal for the chip. No internal clock division is used.

Figure 20 shows the parallel instruction fetches and instruction executions enabled by the Harvard architecture and the fast-access register file concept. This is the basic pipelining concept to obtain up to 1 MIPS per MHz with the corresponding unique results for functions per cost, functions per clocks and functions per power-unit.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 20. The Parallel Instruction Fetches and Instruction Executions

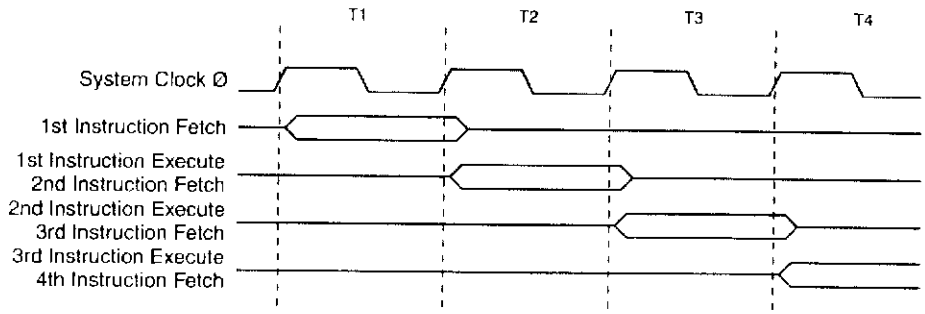
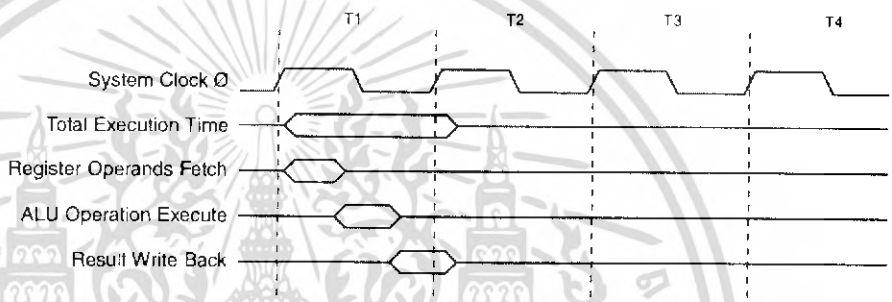


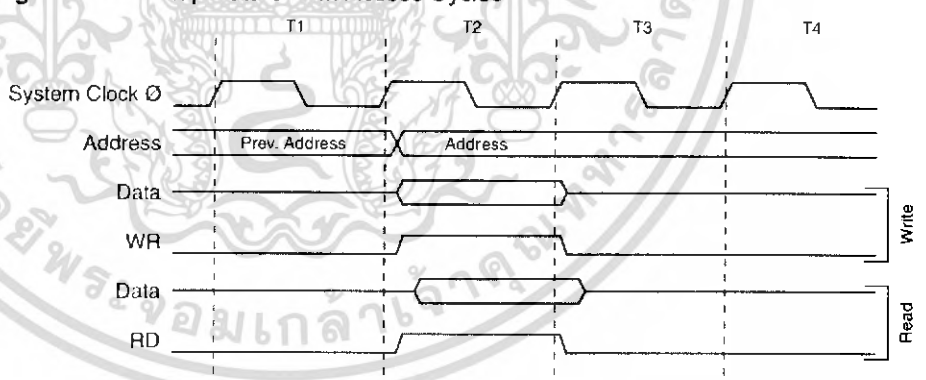
Figure 21 shows the internal timing concept for the register file. In a single clock cycle an ALU operation using two register operands is executed, and the result is stored back to the destination register.

Figure 21. Single-cycle ALU Operation



The internal data SRAM access is performed in two System Clock cycles as described in Figure 22.

Figure 22. On-chip Data SRAM Access Cycles



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I/O Memory

The I/O space definition of the AT90S2313 is shown in Table 1.

Table 1. AT90S2313 I/O Space

Address Hex	Name	Function
$3F ($5F)	SREG	Status Register
$3D ($5D)	SPL	Stack Pointer Low
$3B ($5B)	GIMSK	General Interrupt MaSK register
$3A ($5A)	GIFR	General Interrupt Flag Register
$39 ($59)	TIMSK	Timer/Counter Interrupt MaSK register
$38 ($58)	TIFR	Timer/Counter Interrupt Flag register
$35 ($55)	MCUCR	MCU general Control Register
$33 ($53)	TCCR0	Timer/Counter 0 Control Register
$32 ($52)	TCNT0	Timer/Counter 0 (8-bit)
$2F ($4F)	TCCR1A	Timer/Counter 1 Control Register A
$2E ($4E)	TCCR1B	Timer/Counter 1 Control Register B
$2D ($4D)	TCNT1H	Timer/Counter 1 High Byte
$2C ($4C)	TCNT1L	Timer/Counter 1 Low Byte
$2B ($4B)	OCR1AH	Output Compare Register 1 High Byte
$2A ($4A)	OCR1AL	Output Compare Register 1 Low Byte
$25 ($45)	ICR1H	T/C 1 Input Capture Register High Byte
$24 ($44)	ICR1L	T/C 1 Input Capture Register Low Byte
$21 ($41)	WDTCR	Watchdog Timer Control Register
$1E ($3E)	EEAR	EEPROM Address Register
$1D ($3D)	EEDR	EEPROM Data Register
$1C ($3C)	EECR	EEPROM Control Register
$18 ($38)	PORTB	Data Register, Port B
$17 ($37)	DDRB	Data Direction Register, Port B
$16 ($36)	PINB	Input Pins, Port B
$12 ($32)	PORTD	Data Register, Port D
$11 ($31)	DDRD	Data Direction Register, Port D
$10 ($30)	PIND	Input Pins, Port D
$0C ($2C)	UDR	UART I/O Data Register
$0B ($2B)	USR	UART Status Register
$0A ($2A)	UCR	UART Control Register
$09 ($29)	UBRR	UART Baud Rate Register
$08 ($28)	ACSR	Analog Comparator Control and Status Register

Note: Reserved and unused locations are not shown in the table.

All AT90S2313 I/O and peripherals are placed in the I/O space. The I/O locations are accessed by the IN and OUT instructions transferring data between the 32 general-pur-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

pose working registers and the I/O space. I/O registers within the address range $00 - $1F are directly bit-accessible using the SBI and CBI instructions. In these registers, the value of single bits can be checked by using the SBIS and SBIC instructions. Refer to the instruction set section for more details. When using the I/O specific commands IN and OUT, the I/O addresses $00 - $3F must be used. When addressing I/O registers as SRAM, $20 must be added to this address. All I/O register addresses throughout this document are shown with the SRAM address in parentheses.

For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.

Some of the status flags are cleared by writing a logical "1" to them. Note that the CBI and SBI instructions will operate on all bits in the I/O register, writing a "1" back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers $00 to $1F only.

The I/O and peripherals control registers are explained in the following sections.

Status Register – SREG

The AVR Status Register (SREG) at I/O space location $3F ($5F) is defined as:

Bit	7	6	5	4	3	2	1	0	
$3F ($5F)	I	T	H	S	V	N	Z	C	SREG
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7 – I: Global Interrupt Enable

The global interrupt enable bit must be set (one) for the interrupts to be enabled. The individual interrupt enable control is then performed in separate control registers. If the global interrupt enable bit is cleared (zero), none of the interrupts are enabled independent of the individual interrupt enable settings. The I-bit is cleared by hardware after an interrupt has occurred, and is set by the RETI instruction to enable subsequent interrupts.

• Bit 6 – T: Bit Copy Storage

The bit copy instructions BLD (Bit LoaD) and BST (Bit STore) use the T-bit as source and destination for the operated bit. A bit from a register in the register file can be copied into T by the BST instruction, and a bit in T can be copied into a bit in a register in the register file by the BLD instruction.

• Bit 5 – H: Half-carry Flag

The half-carry flag H indicates a half-carry in some arithmetic operations. See the Instruction Set description for detailed information.

• Bit 4 – S: Sign Bit, S = N ⊕ V

The S-bit is always an exclusive or between the negative flag N and the two's complement overflow flag V. See the Instruction Set description for detailed information.

• Bit 3 – V: Two's Complement Overflow Flag

The two's complement overflow flag V supports two's complement arithmetics. See the Instruction Set description for detailed information.

• Bit 2 – N: Negative Flag

The negative flag N indicates a negative result after the different arithmetic and logic operations. See the Instruction Set description for detailed information.

• Bit 1 – Z: Zero Flag

The zero flag Z indicates a zero result after the different arithmetic and logic operations. See the Instruction Set description for detailed information.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• **Bit 0 – C: Carry Flag**

The carry flag C indicates a carry in an arithmetic or logic operation. See the Instruction Set description for detailed information.

Note that the Status Register is not automatically stored when entering an interrupt routine and restored when returning from an interrupt routine. This must be handled by software.

Stack Pointer – SP

An 8-bit register at I/O address $3D ($5D) forms the stack pointer of the AT90S2313. 8 bits are used to address the 128 bytes of SRAM in locations $60 - $DF.

Bit	7	6	5	4	3	2	1	0	
$3D ($5D)	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	SPL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

The Stack Pointer points to the data SRAM stack area where the Subroutine and Interrupt stacks are located. This stack space in the data SRAM must be defined by the program before any subroutine calls are executed or interrupts are enabled. The Stack Pointer must be set to point above $60. The Stack Pointer is decremented by 1 when data is pushed onto the stack with the PUSH instruction, and it is decremented by 2 when an address is pushed onto the stack with subroutine calls and interrupts. The Stack Pointer is incremented by 1 when data is popped from the stack with the POP instruction, and it is incremented by 2 when an address is popped from the stack with return from subroutine RET or return from interrupt RETI.

Reset and Interrupt Handling

The AT90S2313 provides 10 different interrupt sources. These interrupts and the separate reset vector each have a separate program vector in the program memory space. All the interrupts are assigned individual enable bits that must be set (one) together with the I-bit in the Status Register in order to enable the interrupt.

The lowest addresses in the program memory space are automatically defined as the Reset and Interrupt vectors. The complete list of vectors is shown in Table 2. The list also determines the priority levels of the different interrupts. The lower the address, the higher the priority level. RESET has the highest priority, and next is INT0 (the External Interrupt Request 0), etc.

Table 2. Reset and Interrupt Vectors

Vector No.	Program Address	Source	Interrupt Definition
1	$000	RESET	Hardware Pin, Power-on Reset and Watchdog Reset
2	$001	INT0	External Interrupt Request 0
3	$002	INT1	External Interrupt Request 1
4	$003	TIMER1 CAPT1	Timer/Counter1 Capture Event
5	$004	TIMER1 COMP1	Timer/Counter1 Compare Match
6	$005	TIMER1 OVF1	Timer/Counter1 Overflow
7	$006	TIMER0 OVF0	Timer/Counter0 Overflow
8	$007	UART, RX	UART, RX Complete

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 2. Reset and Interrupt Vectors (Continued)

Vector No.	Program Address	Source	Interrupt Definition
9	$008	UART, UDRE	UART Data Register Empty
10	$009	UART, TX	UART, TX Complete
11	$00A	ANA_COMP	Analog Comparator

The most typical and general program setup for the Reset and Interrupt vector addresses are:

```
Address Labels  Code                 Comments

$000            rjmp RESET           ; Reset Handler
$001            rjmp EXT_INT0        ; IRQ0 Handler
$002            rjmp EXT_INT1        ; IRQ1 Handler
$003            rjmp TIM_CAPT1       ; Timer1 Capture Handler
$004            rjmp TIM_COMP1       ; Timer1 Compare Handler
$005            rjmp TIM_OVF1        ; Timer1 Overflow Handler
$006            rjmp TIM_OVF0        ; Timer0 Overflow Handler
$007            rjmp UART_RXC        ; UART RX Complete Handler
$008            rjmp UART_DRE        ; UDR Empty Handler
$009            rjmp UART_TXC        ; UART TX Complete Handler
$00a            rjmp ANA_COMP        ; Analog Comparator Handler
;
$00b    MAIN:   ldi r16,low(RAMEND); Main program start
$00c            out SPL,r16
$00d            <instr>  xxx
...     ...     ...      ...
```

Reset Sources

The AT90S2313 has three sources of reset:

- Power-on Reset. The MCU is reset when the supply voltage is below the Power-on Reset threshold (V_{POT}).
- External Reset. The MCU is reset when a low level is present on the $\overline{\text{RESET}}$ pin for more than 50 ns.
- Watchdog Reset. The MCU is reset when the Watchdog timer period expires and the Watchdog is enabled.

During reset, all I/O registers are then set to their initial values, and the program starts execution from address $000. The instruction placed in address $000 must be an RJMP (relative jump) instruction to the reset handling routine. If the program never enables an interrupt source, the interrupt vectors are not used, and regular program code can be placed at these locations. The circuit diagram in Figure 23 shows the reset logic. Table 3 defines the timing and electrical parameters of the reset circuitry.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 23. Reset Logic

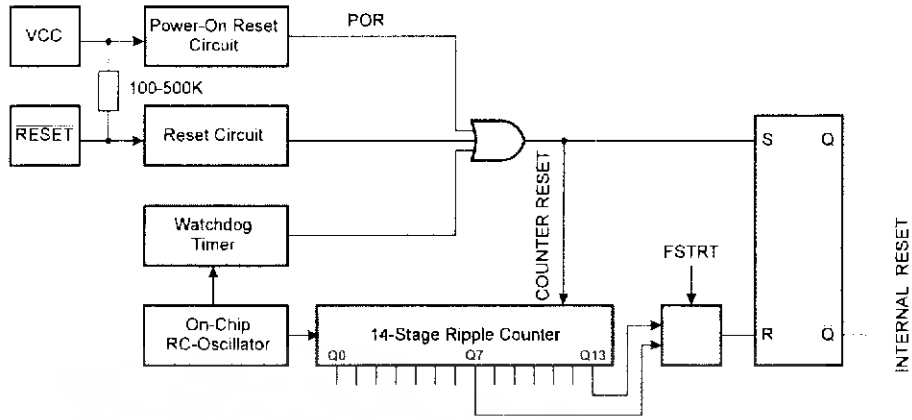


Table 3. Reset Characteristics (V_{CC} = 5.0V)

Symbol	Parameter	Min	Typ	Max	Units
V_{POT} (1)	Power-on Reset Threshold Voltage (rising)	1.0	1.4	1.8	V
	Power-on Reset Threshold Voltage (falling)	0.4	0.6	0.8	V
V_{RST}	$\overline{\text{RESET}}$ Pin Threshold Voltage		–	0.85 V_{CC}	V
t_{TOUT}	Reset Delay Time-out Period FSTRT Unprogrammed	11.0	16.0	21.0	ms
t_{TOUT}	Reset Delay Time-out Period FSTRT Programmed	0.25	0.28	0.31	ms

Note: 1. The Power-on Reset will not work unless the supply voltage has been below V_{POT} (falling).

The user can select the start-up time according to typical oscillator start-up. The number of WDT oscillator cycles used for each time-out is shown in Table 4. The frequency of the Watchdog Oscillator is voltage-dependent, as shown in "Typical Characteristics" on page 75.

Table 4. Number of Watchdog Oscillator Cycles

FSTRT	Time-out at V_{CC} = 5V	Number of WDT Cycles
Programmed	0.28 ms	256
Unprogrammed	16.0 ms	16K

Power-on Reset

A Power-on Reset (POR) circuit ensures that the device is reset from power-on. As shown in Figure 23, an internal timer is clocked from the Watchdog Timer. This timer prevents the MCU from starting until after a certain period after V_{CC} has reached the Power-on Threshold voltage (V_{POT}) (see Figure 24). The FSTRT Fuse bit in the Flash can be programmed to give a shorter start-up time if a ceramic resonator or any other fast-start oscillator is used to clock the MCU.

If the built-in start-up delay is sufficient, $\overline{\text{RESET}}$ can be connected to V_{CC} directly or via an external pull-up resistor. By holding the $\overline{\text{RESET}}$ pin low for a period after V_{CC} has been applied, the Power-on Reset period can be extended. Refer to Figure 25 for a timing example of this.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 24. MCU Start-up, $\overline{\text{RESET}}$ Tied to V_{CC}.

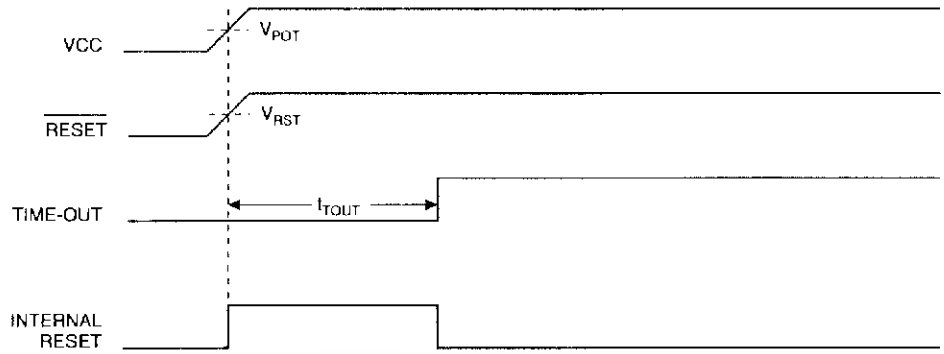
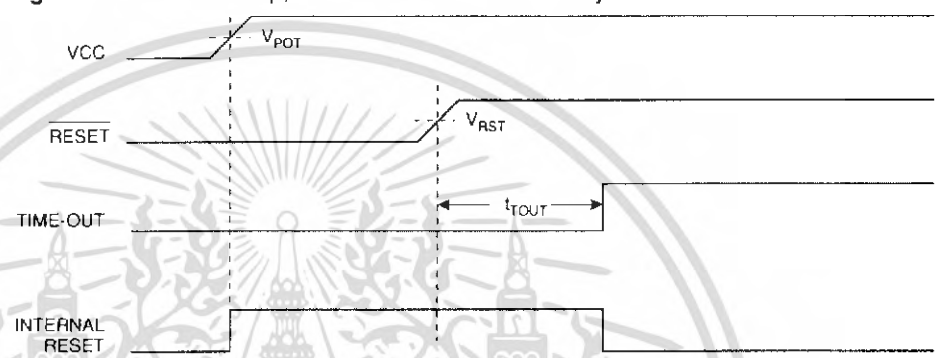


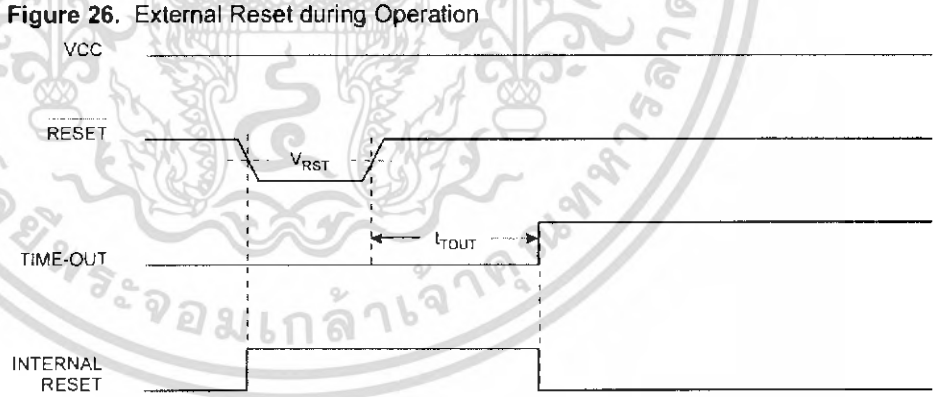
Figure 25. MCU Start-up, $\overline{\text{RESET}}$ Controlled Externally



External Reset

An external reset is generated by a low level on the $\overline{\text{RESET}}$ pin. Reset pulses longer than 50 ns will generate a reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a reset. When the applied signal reaches the Reset Threshold Voltage (V_{RST}) on its positive edge, the delay timer starts the MCU after the Time-out period t_{TOUT} has expired.

Figure 26. External Reset during Operation

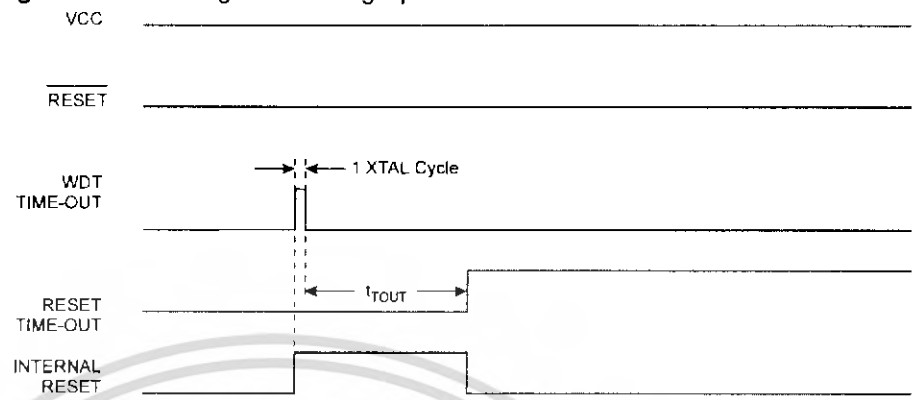


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Watchdog Reset

When the Watchdog times out, it will generate a short reset pulse of one XTAL cycle duration. On the falling edge of this pulse, the delay timer starts counting the Time-out period t_{TOUT}. Refer to page 38 for details on operation of the Watchdog.

Figure 27. Watchdog Reset during Operation



Interrupt Handling

The AT90S2313 has two 8-bit Interrupt Mask control registers: the GIMSK (General Interrupt Mask register) and the TIMSK (Timer/Counter Interrupt Mask register).

When an interrupt occurs, the Global Interrupt Enable I-bit is cleared (zero) and all interrupts are disabled. The user software can set (one) the I-bit to enable interrupts. The I-bit is set (one) when a Return from Interrupt instruction (RETI) is executed.

For interrupts triggered by events that can remain static (e.g., the Output Compare Register1 matching the value of Timer/Counter1), the interrupt flag is set when the event occurs. If the interrupt flag is cleared and the interrupt condition persists, the flag will not be set until the event occurs the next time.

When the Program Counter is vectored to the actual interrupt vector in order to execute the interrupt handling routine, hardware clears the corresponding flag that generated the interrupt. Some of the interrupt flags can also be cleared by writing a logical "1" to the flag bit position(s) to be cleared.

If an interrupt condition occurs when the corresponding interrupt enable bit is cleared (zero), the interrupt flag will be set and remembered until the interrupt is enabled, or the flag is cleared by software.

If one or more interrupt conditions occur when the global interrupt enable bit is cleared (zero), the corresponding interrupt flag(s) will be set and remembered until the global interrupt enable bit is set (one), and will be executed by order of priority.

Note that external level interrupt does not have a flag, and will only be remembered for as long as the interrupt condition is active.

Note that the Status Register is not automatically stored when entering an interrupt routine and restored when returning from an interrupt routine. This must be handled by software.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

General Interrupt Mask Register – GIMSK

Bit	7	6	5	4	3	2	1	0	
$38 ($58)	INT1	INT0	–	–	–	–	–	–	GIMSK
Read/Write	R/W	R/W	R	R	R	R	R	R	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7 – INT1: External Interrupt Request 1 Enable

When the INT1 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is enabled. The Interrupt Sense Control1 bits 1/0 (ISC11 and ISC10) in the MCU general Control Register (MCUCR) defines whether the external interrupt is activated on rising or falling edge of the INT1 pin or level sensed. Activity on the pin will cause an interrupt request even if INT1 is configured as an output. The corresponding interrupt of External Interrupt Request 1 is executed from program memory address $002. See also "External Interrupts".

• Bit 6 – INT0: External Interrupt Request 0 Enable

When the INT0 bit is set (one) and the I-bit in the Status Register (SREG) is set (one), the external pin interrupt is enabled. The Interrupt Sense Control0 bits 1/0 (ISC01 and ISC00) in the MCU general Control Register (MCUCR) defines whether the external interrupt is activated on rising or falling edge of the INT0 pin or level sensed. Activity on the pin will cause an interrupt request even if INT0 is configured as an output. The corresponding interrupt of External Interrupt Request 0 is executed from program memory address $001. See also "External Interrupts."

• Bits 5..0 – Res: Reserved Bits

These bits are reserved bits in the AT90S2313 and always read as zero.

General Interrupt FLAG Register – GIFR

Bit	7	6	5	4	3	2	1	0	
$3A ($5A)	INTF1	INTF0	–	–	–	–	–	–	GIFR
Read/Write	R/W	R/W	R	R	R	R	R	R	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7 – INTF1: External Interrupt Flag1

When an edge on the INT1 pin triggers an interrupt request, the corresponding interrupt flag, INTF1, becomes set (one). If the I-bit in SREG and the corresponding interrupt enable bit, INT1 bit in GIMSK, are set (one), the MCU will jump to the interrupt vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing a logical "1" to it. The flag is always cleared when INT1 is configured as level interrupt.

• Bit 6 – INTF0: External Interrupt Flag0

When an edge on the INT0 pin triggers an interrupt request, the corresponding interrupt flag, INTF0, becomes set (one). If the I-bit in SREG and the corresponding interrupt enable bit, INT0 bit in GIMSK, are set (one), the MCU will jump to the interrupt vector. The flag is cleared when the interrupt routine is executed. Alternatively, the flag can be cleared by writing a logical "1" to it. The flag is always cleared when INT0 is configured as level interrupt.

• Bits 5..0 – Res: Reserved Bits

These bits are reserved bits in the AT90S2313 and always read as zero.

Note that external level interrupt does not have a flag, and will only be remembered for as long as the interrupt condition is active.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timer/Counter Interrupt Mask Register – TIMSK

Bit	7	6	5	4	3	2	1	0	
$39 ($59)	TOIE1	OCIE1A	–	–	TICIE1	–	TOIE0	–	TIMSK
Read/Write	R/W	R/W	R	R	R/W	R	R/W	R	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7 – TOIE1: Timer/Counter1 Overflow Interrupt Enable

When the TOIE1 bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter1 Overflow Interrupt is enabled. The corresponding interrupt (at vector $005) is executed if an overflow in Timer/Counter1 occurs (i.e., when the TOV1 bit is set in the Timer/Counter Interrupt Flag Register [TIFR]).

• Bit 6 – OCIE1A: Timer/Counter1 Output Compare Match Interrupt Enable

When the OCIE1A bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter1 Compare Match Interrupt is enabled. The corresponding interrupt (at vector $004) is executed if a compare match in Timer/Counter1 occurs (i.e., when the OCF1A bit is set in the Timer/Counter Interrupt Flag Register [TIFR]).

• Bit 5,4 – Res: Reserved Bits

These bits are reserved bits in the AT90S2313 and always read as zero.

• Bit 3 – TICIE1: Timer/Counter1 Input Capture Interrupt Enable

When the TICIE1 bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter1 Input Capture Event Interrupt is enabled. The corresponding interrupt (at vector $003) is executed if a capture-triggering event occurs on PD6(ICP) (i.e., when the ICF1 bit is set in the Timer/Counter Interrupt Flag Register [TIFR]).

• Bit 2 – Res: Reserved Bit

This bit is a reserved bit in the AT90S2313 and always reads as zero.

• Bit 1 – TOIE0: Timer/Counter0 Overflow Interrupt Enable

When the TOIE0 bit is set (one) and the I-bit in the Status Register is set (one), the Timer/Counter0 Overflow Interrupt is enabled. The corresponding interrupt (at vector $006) is executed if an overflow in Timer/Counter0 occurs (i.e., when the TOV0 bit is set in the Timer/Counter Interrupt Flag Register [TIFR]).

• Bit 0 – Res: Reserved Bit

This bit is a reserved bit in the AT90S2313 and always read as zero.

Timer/Counter Interrupt FLAG Register – TIFR

Bit	7	6	5	4	3	2	1	0	
$38 ($58)	TOV1	OCF1A	–	–	ICF1	–	TOV0	–	TIFR
Read/Write	R/W	R/W	R	R	R/W	R	R/W	R	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7 – TOV1: Timer/Counter1 Overflow Flag

The TOV1 is set (one) when an overflow occurs in Timer/Counter1. TOV1 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, TOV1 is cleared by writing a logical "1" to the flag. When the I-bit in SREG and TOIE1 (Timer/Counter1 Overflow Interrupt Enable) and TOV1 are set (one), the Timer/Counter1 Overflow Interrupt is executed. In PWM mode, this bit is set when Timer/Counter1 changes counting direction at $0000.

• Bit 6 – OCF1A: Output Compare Flag 1A

The OCF1A bit is set (one) when a compare match occurs between the Timer/Counter1 and the data in OCR1A (Output Compare Register1 A). OCF1A is cleared by hardware

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

when executing the corresponding interrupt handling vector. Alternatively, OCF1A is cleared by writing a logical "1" to the flag. When the I-bit in SREG and OCIE1A (Timer/Counter1 Compare Match Interrupt Enable) and the OCF1A are set (one), the Timer/Counter1 Compare Match Interrupt is executed.

• Bits 5, 4 – Res: Reserved Bits

These bits are reserved bits in the AT90S2313 and always read as zero.

• Bit 3 – ICF1: Input Capture Flag 1

The ICF1 bit is set (one) to flag an input capture event, indicating that the Timer/Counter1 value has been transferred to the Input Capture Register (ICR1). ICF1 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, ICF1 is cleared by writing a logical "1" to the flag. When the SREG I-bit and TICIE1 (Timer/Counter1 Input Capture Interrupt Enable) and ICF1 are set (one), the Timer/Counter1 Capture Interrupt is executed.

• Bit 2 – Res: Reserved Bit

This bit is a reserved bit in the AT90S2313 and always reads as zero.

• Bit 1 – TOV0: Timer/Counter0 Overflow Flag

The bit TOV0 is set (one) when an overflow occurs in Timer/Counter0. TOV0 is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, TOV0 is cleared by writing a logical "1" to the flag. When the SREG I-bit and TOIE0 (Timer/Counter0 Overflow Interrupt Enable) and TOV0 are set (one), the Timer/Counter0 Overflow Interrupt is executed.

• Bit 0 – Res: Reserved Bit

This bit is a reserved bit in the AT90S2313 and always reads as zero.

External Interrupts

The External Interrupts are triggered by the INT1 and INT0 pins. Observe that, if enabled, the interrupts will trigger even if the INT0/INT1 pins are configured as outputs. This feature provides a way of generating a software interrupt. The External Interrupts can be triggered by a falling or rising edge or a low level. This is set up as indicated in the specification for the MCU Control Register (MCUCR). When the External Interrupt is enabled and is configured as level-triggered, the interrupt will trigger as long as the pin is held low.

The External Interrupts are set up as described in the specification for the MCU Control Register (MCUCR).

Interrupt Response Time

The interrupt execution response for all the enabled AVR interrupts is four clock cycles, minimum. Four clock cycles after the interrupt flag has been set, the program vector address for the actual interrupt handling routine is executed. During this 4-clock-cycle period, the Program Counter (2 bytes) is pushed onto the stack, and the Stack Pointer is decremented by 2. The vector is normally a relative jump to the interrupt routine, and this jump takes two clock cycles. If an interrupt occurs during execution of a multi-cycle instruction, this instruction is completed before the interrupt is served.

A return from an interrupt handling routine takes four clock cycles. During these four clock cycles, the Program Counter (2 bytes) is popped back from the stack, the Stack Pointer is incremented by 2, and the I-flag in SREG is set. When the AVR exits from an interrupt, it will always return to the main program and execute one more instruction before any pending interrupt is served.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MCU Control Register – MCUCR

The MCU Control Register contains control bits for general MCU functions.

Bit	7	6	5	4	3	2	1	0	
$35 ($55)	–	–	SE	SM	ISC11	ISC10	ISC01	ISC00	MCUCR
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• Bits 7, 6 – Res: Reserved Bits

These bits are reserved bits in the AT90S2313 and always read as zero.

• Bit 5 – SE: Sleep Enable

The SE bit must be set (one) to make the MCU enter the Sleep Mode when the SLEEP instruction is executed. To avoid the MCU entering the Sleep Mode unless it is the programmer's purpose, it is recommended to set the Sleep Enable (SE) bit just before the execution of the SLEEP instruction.

• Bit 4 – SM: Sleep Mode

This bit selects between the two available sleep modes. When SM is cleared (zero), Idle Mode is selected as Sleep Mode. When SM is set (one), Power-down Mode is selected as Sleep Mode. For details, refer to the paragraph "Sleep Modes".

• Bits 3, 2 – ISC11, ISC10: Interrupt Sense Control 1 Bit 1 and Bit 0

The External Interrupt 1 is activated by the external pin INT1 if the SREG I-flag and the corresponding interrupt mask in the GIMSK register is set. The level and edges on the external INT1 pin that activate the interrupt are defined in Table 5.

Table 5. Interrupt 1 Sense Control

ISC11	ISC10	Description
0	0	The low level of INT1 generates an interrupt request.
0	1	Reserved
1	0	The falling edge of INT1 generates an interrupt request.
1	1	The rising edge of INT1 generates an interrupt request.

• Bits 1, 0 – ISC01, ISC00: Interrupt Sense Control 0 Bit 1 and Bit 0

The External Interrupt 0 is activated by the external pin INT0 if the SREG I-flag and the corresponding interrupt mask is set. The level and edges on the external INT0 pin that activate the interrupt are defined in Table 6.

Table 6. Interrupt 0 Sense Control

ISC01	ISC00	Description
0	0	The low level of INT0 generates an interrupt request.
0	1	Reserved
1	0	The falling edge of INT0 generates an interrupt request.
1	1	The rising edge of INT0 generates an interrupt request.

The value on the INTn pin is sampled before detecting edges. If edge interrupt is selected, pulses with a duration longer than one CPU clock period will generate an interrupt. Shorter pulses are not guaranteed to generate an interrupt. If low-level interrupt is selected, the low level must be held until the completion of the currently executing instruction to generate an interrupt. If enabled, a level-triggered interrupt will generate an interrupt request as long as the pin is held low.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sleep Modes

To enter the sleep modes, the SE bit in MCUCR must be set (one) and a SLEEP instruction must be executed. If an enabled interrupt occurs while the MCU is in a sleep mode, the MCU awakes, executes the interrupt routine, and resumes execution from the instruction following SLEEP. The contents of the register file, SRAM and I/O memory are unaltered. If a reset occurs during Sleep Mode, the MCU wakes up and executes from the Reset vector.

Idle Mode

When the SM bit is cleared (zero), the SLEEP instruction forces the MCU into the Idle Mode, stopping the CPU but allowing Timer/Counters, Watchdog and the interrupt system to continue operating. This enables the MCU to wake up from external triggered interrupts as well as internal ones like Timer Overflow interrupt and Watchdog reset. If wake-up from the Analog Comparator Interrupt is not required, the Analog Comparator can be powered down by setting the ACD-bit in the Analog Comparator Control and Status Register (ACSR). This will reduce power consumption in Idle Mode. When the MCU wakes up from Idle Mode, the CPU starts program execution immediately.

Power-down Mode

When the SM bit is set (one), the SLEEP instruction forces the MCU into the Power-down Mode. In this mode, the external oscillator is stopped while the external interrupts and the Watchdog (if enabled) continue operating. Only an external reset, a Watchdog reset (if enabled), an external level interrupt on INT0 or INT1 can wake up the MCU.

Note that when a level-triggered interrupt is used for wake-up from power-down, the low level must be held for a time longer than the reset delay Time-out period t_{TOUT}. Otherwise, the device will not wake up.

Timer/Counters

The AT90S2313 provides two general-purpose Timer/Counters – one 8-bit T/C and one 16-bit T/C. The Timer/Counters have individual prescaling selection from the same 10-bit prescaling timer. Both Timer/Counters can either be used as a timer with an internal clock time base or as a counter with an external pin connection that triggers the counting.

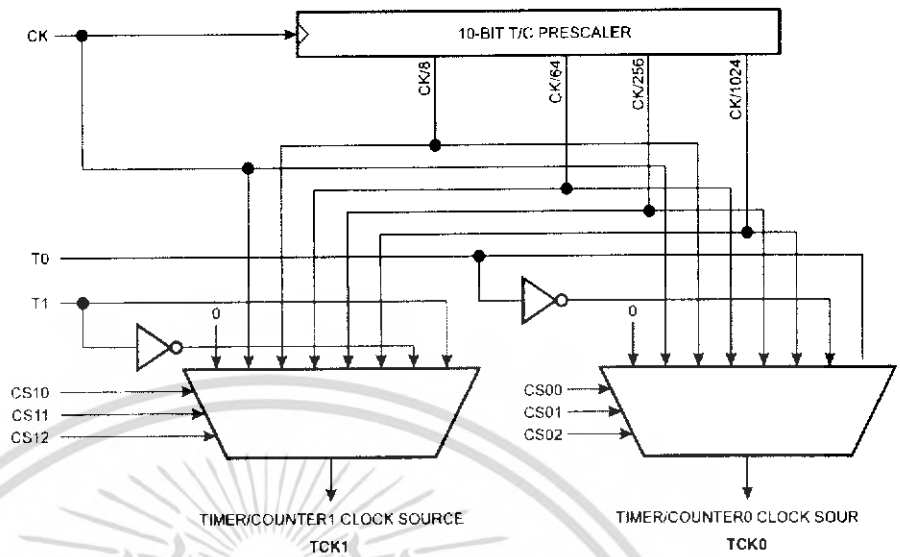
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timer/Counter0 Control Register (TCCR0). The overflow status flag is found in the

Timer/Counter Prescaler

Figure 28 shows the general Timer/Counter prescaler.

Figure 28. Timer/Counter Prescaler



The four different prescaled selections are: CK/8, CK/64, CK/256 and CK/1024, where CK is the oscillator clock. For the two Timer/Counters, added selections such as CK, external clock source and stop can be selected as clock sources.

8-bit Timer/Counter0

Figure 29 shows the block diagram for Timer/Counter0.

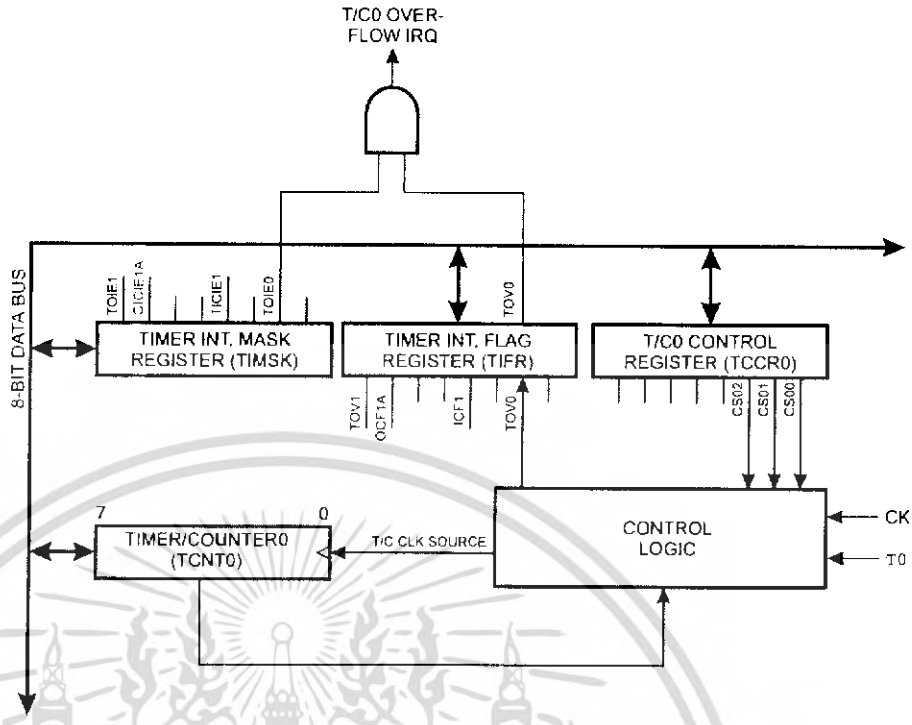
The 8-bit Timer/Counter0 can select clock source from CK, prescaled CK or an external pin. In addition, it can be stopped as described in the specification for the Timer/Counter0 Control Register (TCCR0). The overflow status flag is found in the Timer/Counter Interrupt Flag Register (TIFR). Control signals are found in the Timer/Counter0 Control Register (TCCR0). The interrupt enable/disable settings for Timer/Counter0 are found in the Timer/Counter Interrupt Mask Register (TIMSK).

When Timer/Counter0 is externally clocked, the external signal is synchronized with the oscillator frequency of the CPU. To assure proper sampling of the external clock, the minimum time between two external clock transitions must be at least one internal CPU clock period. The external clock signal is sampled on the rising edge of the internal CPU clock.

The 8-bit Timer/Counter0 features both a high-resolution and a high-accuracy usage with the lower prescaling opportunities. Similarly, the high prescaling opportunities make the Timer/Counter0 useful for lower speed functions or exact timing functions with infrequent actions.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 29. Timer/Counter0 Block Diagram



Timer/Counter0 Control Register – TCCR0

Bit	7	6	5	4	3	2	1	0	
$33 ($53)	–	–	–	–	–	CS02	CS01	CS00	TCCR0
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• Bits 7..3 – Res: Reserved Bits

These bits are reserved bits in the AT90S2313 and always read zero.

• Bits 2,1,0 – CS02, CS01, CS00: Clock Select0, Bit 2,1 and 0

The Clock Select0 bits 2, 1 and 0 define the prescaling source of Timer/Counter0.

Table 7. Clock 0 Prescale Select

CS02	CS01	CS00	Description
0	0	0	Stop, the Timer/Counter0 is stopped.
0	0	1	CK
0	1	0	CK/8
0	1	1	CK/64
1	0	0	CK/256
1	0	1	CK/1024
1	1	0	External Pin T0, falling edge
1	1	1	External Pin T0, rising edge

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The Stop condition provides a Timer Enable/Disable function. The CK down divided modes are scaled directly from the CK oscillator clock. If the external pin modes are used for Timer/Counter0, transitions on PD4/(T0) will clock the counter even if the pin is configured as an output. This feature can give the user software control of the counting.

Timer/Counter0 – TCNT0

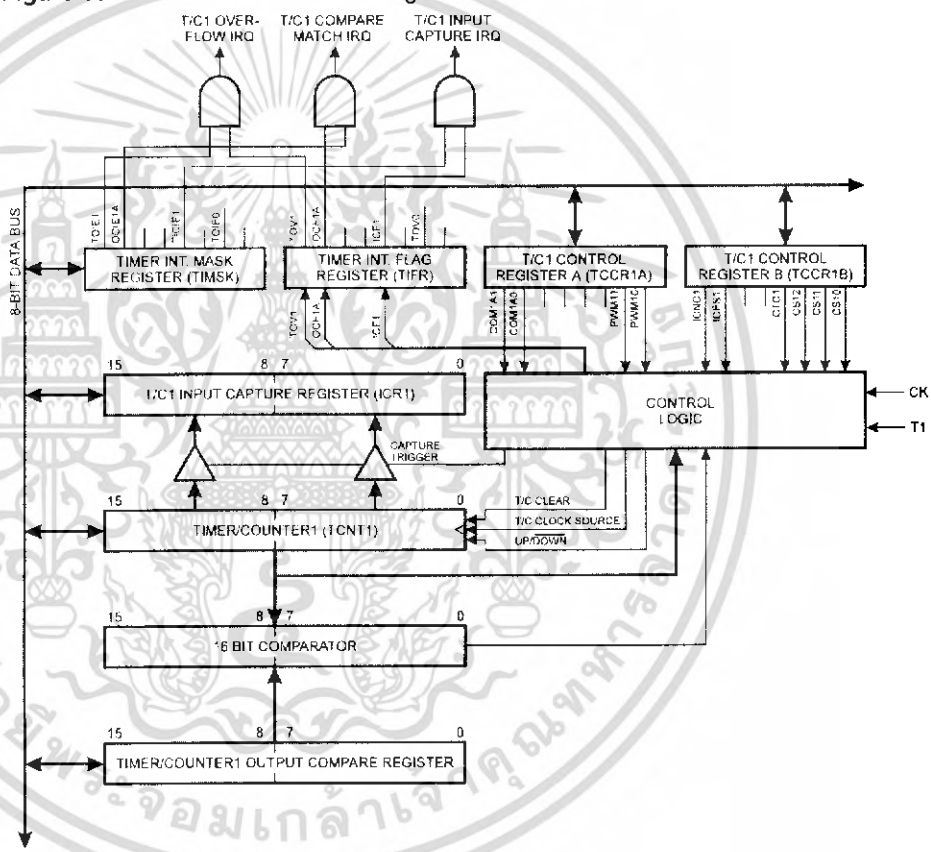
Bit	7	6	5	4	3	2	1	0	
$32 ($52)	MSB							LSB	TCNT0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

The Timer/Counter0 is realized as an up-counter with read and write access. If the Timer/Counter0 is written and a clock source is present, the Timer/Counter0 continues counting in the timer clock cycle following the write operation.

16-bit Timer/Counter1

Figure 30 shows the block diagram for Timer/Counter1.

Figure 30. Timer/Counter1 Block Diagram



The 16-bit Timer/Counter1 can select clock source from CK, prescaled CK or an external pin. In addition, it can be stopped as described in the specification for the Timer/Counter1 Control Register (TCCR1B). The different status flags (Overflow, Compare Match and Capture Event) and control signals are found in the Timer/Counter Interrupt Flag Register (TIFR). The interrupt enable/disable settings for Timer/Counter1 are found in the Timer/Counter Interrupt Mask Register (TIMSK).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

When Timer/Counter1 is externally clocked, the external signal is synchronized with the oscillator frequency of the CPU. To assure proper sampling of the external clock, the minimum time between two external clock transitions must be at least one internal CPU clock period. The external clock signal is sampled on the rising edge of the internal CPU clock.

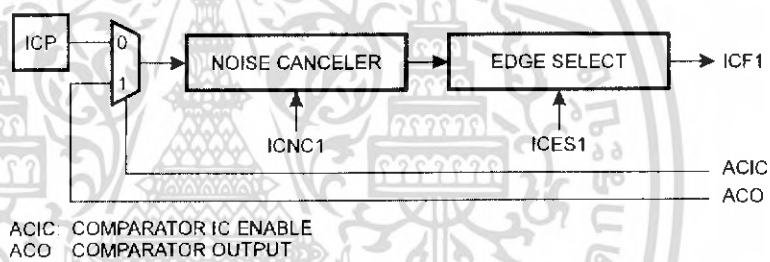
The 16-bit Timer/Counter1 features both a high-resolution and a high-accuracy usage with the lower prescaling opportunities. Similarly, the high prescaling opportunities makes the Timer/Counter1 useful for lower speed functions or exact timing functions with infrequent actions.

The Timer/Counter1 supports an Output Compare function using the Output Compare Register 1A (OCR1A) as the data source to be compared to the Timer/Counter1 contents. The Output Compare functions include optional clearing of the counter on compare matches, and actions on the Output Compare pin 1 on compare matches.

Timer/Counter1 can also be used as an 8-, 9- or 10-bit Pulse Width Modulator. In this mode the counter and the OCR1 register serve as a glitch-free standalone PWM with centered pulses. Refer to page 36 for a detailed description of this function.

The Input Capture function of Timer/Counter1 provides a capture of the Timer/Counter1 contents to the Input Capture Register (ICR1), triggered by an external event on the Input Capture Pin (ICP). The actual capture event settings are defined by the Timer/Counter1 Control Register (TCCR1B). In addition, the Analog Comparator can be set to trigger the input capture. Refer to "Analog Comparator" on page 50 for details on this. The ICP pin logic is shown in Figure 31.

Figure 31. ICP Pin Schematic Diagram



If the Noise Canceler function is enabled, the actual trigger condition for the capture event is monitored over four samples, and all four must be equal to activate the capture flag.

Timer/Counter1 Control Register A – TCCR1A

Bit	7	6	5	4	3	2	1	0	
$2F ($4F)	COM1A1	COM1A0	–	–	–	–	PWM11	PWM10	TCCR1A
Read/Write	R/W	R/W	R	R	R	R	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• Bits 7,6 – COM1A1, COM1A0: Compare Output Mode1, Bits 1 and 0

The COM1A1 and COM1A0 control bits determine any output pin action following a compare match in Timer/Counter1. Any output pin actions affect pin OC1 (Output Compare pin 1) (PB3). This is an alternative function to the I/O port, and the corresponding

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

direction control bit must be set (one) to control an output pin. The control configuration is shown in Table 8.

Table 8. Compare 1 Mode Select

COM1A1	COM1A0	Description
0	0	Timer/Counter1 disconnected from output pin OC1
0	1	Toggle the OC1 output line.
1	0	Clear the OC1 output line (to zero).
1	1	Set the OC1 output line (to one).

Note: In PWM mode, these bits have a different function. Refer to Table 12 for a detailed description.

• Bits 5..2 – Res: Reserved Bits

These bits are reserved bits in the AT90S2313 and always read zero.

• Bits 1,0 – PWM11, PWM10: Pulse Width Modulator Select Bits

These bits select PWM operation of Timer/Counter1 as specified in Table 9. This mode is described on page 35.

Table 9. PWM Mode Select

PWM11	PWM10	Description
0	0	PWM operation of Timer/Counter1 is disabled
0	1	Timer/Counter1 is an 8-bit PWM
1	0	Timer/Counter1 is a 9-bit PWM
1	1	Timer/Counter1 is a 10-bit PWM

Timer/Counter1 Control Register B – TCCR1B

Bit	7	6	5	4	3	2	1	0	
$2E ($4E)	ICNC1	ICES1	–	–	CTC1	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7 – ICNC1: Input Capture1 Noise Canceler (4 CKs)

When the ICNC1 bit is cleared (zero), the input capture trigger noise canceler function is disabled. The input capture is triggered at the first rising/falling edge sampled on the ICP (input capture pin) as specified. When the ICNC1 bit is set (one), four successive samples are measured on the ICP (input capture pin), and all samples must be high/low according to the input capture trigger specification in the ICES1 bit. The actual sampling frequency is the XTAL clock frequency.

• Bit 6 – ICES1: Input Capture1 Edge Select

While the ICES1 bit is cleared (zero), the Timer/Counter1 contents are transferred to the Input Capture Register (ICR1) on the falling edge of the input capture pin (ICP). While the ICES1 bit is set (one), the Timer/Counter1 contents are transferred to the Input Capture Register (ICR1) on the rising edge of the input capture pin (ICP).

• Bits 5, 4 – Res: Reserved Bits

These bits are reserved bits in the AT90S2313 and always read zero.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• **Bit 3 – CTC1: Clear Timer/Counter1 on Compare Match**

When the CTC1 control bit is set (one), the Timer/Counter1 is reset to $0000 in the clock cycle after a compareA match. If the CTC1 control bit is cleared, Timer/Counter1 continues counting and is unaffected by a compare match. Since the compare match is detected in the CPU clock cycle following the match, this function will behave differently when a prescaling higher than 1 is used for the timer. When a prescaling of 1 is used, and the compareA register is set to C, the timer will count as follows if CTC1 is set:

... | C-2 | C-1 | C | 0 | 1 |...

When the prescaler is set to divide by 8, the timer will count like this:

... | C-2, C-2, C-2, C-2, C-2, C-2, C-2, C-2 | C-1, C-1, C-1, C-1, C-1, C-1, C-1, C-1 | C, 0, 0, 0, 0, 0, 0, 0 |...

In PWM mode, this bit has no effect.

• **Bits 2,1,0 – CS12, CS11, CS10: Clock Select1, Bits 2, 1 and 0**

The Clock Select1 bits 2, 1 and 0 define the prescaling source of Timer/Counter1.

Table 10. Clock 1 Prescale Select

CS12	CS11	CS10	Description
0	0	0	Stop, the Timer/Counter1 is stopped.
0	0	1	CK
0	1	0	CK/8
0	1	1	CK/64
1	0	0	CK/256
1	0	1	CK/1024
1	1	0	External Pin T1, falling edge
1	1	1	External Pin T1, rising edge

The Stop condition provides a Timer Enable/Disable function. The CK down divided modes are scaled directly from the CK oscillator clock. If the external pin modes are used for Timer/Counter1, transitions on PD5/(T1) will clock the counter even if the pin is configured as an output. This feature can give the user software control of the counting.

Timer/Counter1 – TCNT1H and TCNT1L

Bit	15	14	13	12	11	10	9	8	
$2D ($4D)	MSB								TCNT1H
$2C ($4C)								LSB	TCNT1L
	7	6	5	4	3	2	1	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

This 16-bit register contains the prescaled value of the 16-bit Timer/Counter1. To ensure that both the high and low bytes are read and written simultaneously when the CPU accesses these registers, the access is performed using an 8-bit temporary register (TEMP). This temporary register is also used when accessing OCR1A and ICR1. If the main program and interrupt routines perform access to registers using TEMP, inter-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

rupts must be disabled during access from the main program or interrupts if interrupts are re-enabled.

- TCNT1 Timer/Counter1 Write:
  When the CPU writes to the high byte TCNT1H, the written data is placed in the TEMP register. Next, when the CPU writes the low byte TCNT1L, this byte of data is combined with the byte data in the TEMP register, and all 16 bits are written to the TCNT1 Timer/Counter1 register simultaneously. Consequently, the high byte TCNT1H must be accessed first for a full 16-bit register write operation.
- TCNT1 Timer/Counter1 Read:
  When the CPU reads the low byte TCNT1L, the data of the low byte TCNT1L is sent to the CPU and the data of the high byte TCNT1H is placed in the TEMP register. When the CPU reads the data in the high byte TCNT1H, the CPU receives the data in the TEMP register. Consequently, the low byte TCNT1L must be accessed first for a full 16-bit register read operation.

The Timer/Counter1 is realized as an up or up/down (in PWM mode) counter with read and write access. If Timer/Counter1 is written to and a clock source is selected, the Timer/Counter1 continues counting in the timer clock cycle after it is preset with the written value.

Timer/Counter1 Output Compare Register A – OCR1AH and OCR1AL

Bit	15	14	13	12	11	10	9	8	
$2B ($4B)	MSB								OCR1AH
$2A ($4A)								LSB	OCR1AL
	7	6	5	4	3	2	1	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

The output compare register is a 16-bit read/write register.

The Timer/Counter1 Output Compare Register contains the data to be continuously compared with Timer/Counter1. Actions on compare matches are specified in the Timer/Counter1 Control and Status registers.

Since the Output Compare Register (OCR1A) is a 16-bit register, a temporary register TEMP is used when OCR1A is written to ensure that both bytes are updated simultaneously. When the CPU writes the high byte, OCR1AH, the data is temporarily stored in the TEMP register. When the CPU writes the low byte, OCR1AL, the TEMP register is simultaneously written to OCR1AH. Consequently, the high byte OCR1AH must be written first for a full 16-bit register write operation.

The TEMP register is also used when accessing TCNT1, and ICR1. If the main program and interrupt routines perform access to registers using TEMP, interrupts must be disabled during access from the main program or interrupts if interrupts are re-enabled.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timer/Counter1 Input Capture Register – ICR1H and ICR1L

Bit	15	14	13	12	11	10	9	8	
$25 ($45)	MSB								ICR1H
$24 ($44)								LSB	ICR1L
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Initial value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

The input capture register is a 16-bit read-only register.

When the rising or falling edge (according to the input capture edge setting [ICES1]) of the signal at the input capture pin (ICP) is detected, the current value of the Timer/Counter1 is transferred to the Input Capture Register (ICR1). At the same time, the input capture flag (ICF1) is set (one).

Since the Input Capture Register (ICR1) is a 16-bit register, a temporary register TEMP is used when ICR1 is read to ensure that both bytes are read simultaneously. When the CPU reads the low byte ICR1L, the data is sent to the CPU and the data of the high byte ICR1H is placed in the TEMP register. When the CPU reads the data in the high byte ICR1H, the CPU receives the data in the TEMP register. Consequently, the low byte ICR1L must be accessed first for a full 16-bit register read operation.

The TEMP register is also used when accessing TCNT1 and OCR1A. If the main program and interrupt routines perform access to registers using TEMP, interrupts must be disabled during access from the main program or interrupts if interrupts are re-enabled.

Timer/Counter1 in PWM Mode

When the PWM mode is selected, Timer/Counter1 and the Output Compare Register1 (OCR1A) form an 8-, 9- or 10-bit, free-running, glitch-free and phase-correct PWM with output on the PB3(OC1) pin. Timer/Counter1 acts as an up/down counter, counting up from $0000 to TOP (see Table 11), where it turns and counts down again to zero before the cycle is repeated. When the counter value matches the contents of the 8, 9 or 10 least significant bits of OCR1A, the PB3(OC1) pin is set or cleared according to the settings of the COM1A1 and COM1A0 bits in the Timer/Counter1 Control Register (TCCR1). Refer to Table 12 for details.

Table 11. Timer TOP Values and PWM Frequency

PWM Resolution	Timer TOP Value	Frequency
8-bit	$00FF (255)	$f_{TC1}/510$
9-bit	$01FF (511)	$f_{TC1}/1022$
10-bit	$03FF(1023)	$f_{TC1}/2046$

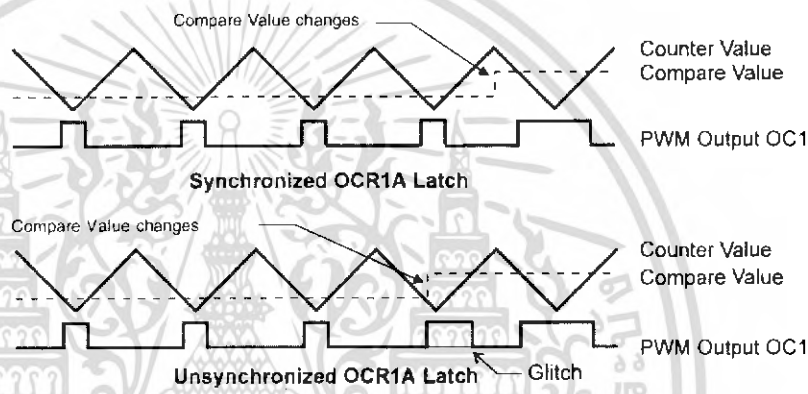
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 12. Compare1 Mode Select in PWM Mode

COM1A1	COM1A0	Effect on OC1
0	0	Not connected
0	1	Not connected
1	0	Cleared on compare match, upcounting. Set on compare match, down-counting (non-inverted PWM).
1	1	Cleared on compare match, downcounting. Set on compare match, up-counting (inverted PWM).

Note that in the PWM mode, the 10 least significant OCR1A bits, when written, are transferred to a temporary location. They are latched when Timer/Counter1 reaches TOP. This prevents the occurrence of odd-length PWM pulses (glitches) in the event of an unsynchronized OCR1A write. See Figure 32 for an example.

Figure 32. Effects on Unsynchronized OCR1 Latching



During the time between the write and the latch operations, a read from OCR1A will read the contents of the temporary location. This means that the most recently written value always will read out of OCR1A.

When the OCR1 contains $0000 or TOP, the output OC1 is updated to low or high on the next compare match according to the settings of COM1A1/COM1A0. This is shown in Table 13.

Note: If the compare register contains the TOP value and the prescaler is not in use (CS12..CS10 = 001), the PWM output will not produce any pulse at all, because the up-counting and down-counting values are reached simultaneously. When the prescaler is in use (CS12..CS10 ≠ 001 or 000), the PWM output goes active when the counter reaches the TOP value, but the down-counting compare match is not interpreted to be reached before the next time the counter reaches the TOP value, making a one-period PWM pulse.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 13. PWM Outputs OCR = $0000 or TOP

COM1A1	COM1A0	OCR1A	Output OC1
1	0	$0000	L
1	0	TOP	H
1	1	$0000	H
1	1	TOP	L

In PWM mode, the Timer Overflow Flag1 (TOV1) is set when the counter advances from $0000. Timer Overflow Interrupt1 operates exactly as in normal Timer/Counter mode (i.e., it is executed when TOV1 is set, provided that Timer Overflow Interrupt1 and global interrupts are enabled). This also applies to the Timer Output Compare1 flag and interrupt.

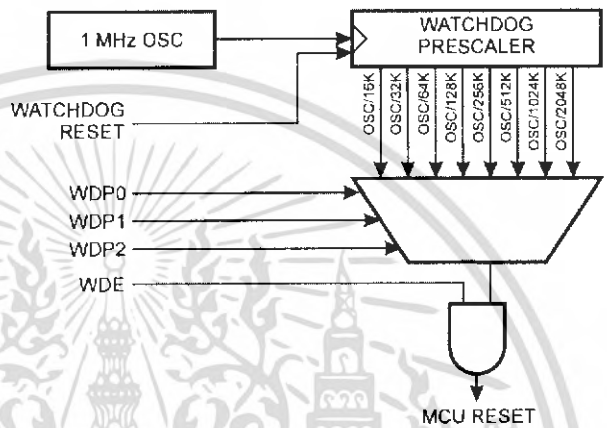
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Watchdog Timer

The Watchdog Timer is clocked from a separate on-chip oscillator that runs at 1 MHz. This is the typical value at V_{CC} = 5V. See characterization data for typical values at other V_{CC} levels. By controlling the Watchdog Timer prescaler, the Watchdog reset interval can be adjusted. See Table 14 for a detailed description. The WDR (Watchdog Reset) instruction resets the Watchdog Timer. Eight different clock cycle periods can be selected to determine the reset period. If the reset period expires without another Watchdog reset, the AT90S2313 resets and executes from the reset vector. For timing details on the Watchdog reset, refer to page 22.

To prevent unintentional disabling of the Watchdog, a special turn-off sequence must be followed when the Watchdog is disabled. Refer to the description of the Watchdog Timer Control Register for details.

Figure 33. Watchdog Timer



Watchdog Timer Control Register – WDTCR

Bit	7	6	5	4	3	2	1	0	
S21 ($41)	–	–	–	WDTOE	WDE	WDP2	WDP1	WDP0	WDTCR
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• **Bits 7..5 – Res: Reserved Bits**

These bits are reserved bits in the AT90S2313 and will always read as zero.

• **Bit 4 – WDTOE: Watchdog Turn-off Enable**

This bit must be set (one) when the WDE bit is cleared. Otherwise, the Watchdog will not be disabled. Once set, hardware will clear this bit to zero after four clock cycles. Refer to the description of the WDE bit for a Watchdog disable procedure.

• **Bit 3 – WDE: Watchdog Enable**

When the WDE is set (one) the Watchdog Timer is enabled, and if the WDE is cleared (zero), the Watchdog Timer function is disabled. WDE can only be cleared if the WDTOE bit is set (one). To disable an enabled Watchdog Timer, the following procedure must be followed:

1. In the same operation, write a logical "1" to WDTOE and WDE. A logical "1" must be written to WDE even though it is set to 1 before the disable operation starts.
2. Within the next four clock cycles, write a logical "0" to WDE. This disables the Watchdog.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• **Bits 2..0 – WDP2, WDP1, WDP0: Watchdog Timer Prescaler 2, 1 and 0**

The WDP2, WDP1 and WDP0 bits determine the Watchdog Timer prescaling when the Watchdog Timer is enabled. The different prescaling values and their corresponding time-out periods are shown in Table 14.

Table 14. Watchdog Timer Prescale Select

WDP2	WDP1	WDP0	Number of WDT Oscillator Cycles	Typical Time-out at V_{CC} = 3.0V	Typical Time-out at V_{CC} = 5.0V
0	0	0	16K cycles	47 ms	15 ms
0	0	1	32K cycles	94 ms	30 ms
0	1	0	64K cycles	0.19 s	60 ms
0	1	1	128K cycles	0.38 s	0.12 s
1	0	0	256K cycles	0.75 s	0,24 s
1	0	1	512K cycles	1.5 s	0.49 s
1	1	0	1,024K cycles	3.0 s	0.97 s
1	1	1	2,048K cycles	6.0 s	1.9 s

Note: The frequency of the Watchdog Oscillator is voltage-dependent, as shown in the Electrical Characteristics section.
The WDR (Watchdog Reset) instruction should always be executed before the Watchdog Timer is enabled. This ensures that the reset period will be in accordance with the Watchdog Timer prescale settings. If the Watchdog Timer is enabled without reset, the Watchdog Timer may not start counting from zero.
To avoid unintentional MCU reset, the Watchdog Timer should be disabled or reset before changing the Watchdog Timer Prescale Select.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EEPROM Read/Write Access

The EEPROM access registers are accessible in the I/O space.

The write access time is in the range of 2.5 - 4 ms, depending on the V_{CC} voltages. A self-timing function, however, lets the user software detect when the next byte can be written. If the user code contains code that writes the EEPROM, some precaution must be taken. In heavily filtered power supplies, V_{CC} is likely to rise or fall slowly on power-up/down. This causes the device for some period of time to run at a voltage lower than specified as minimum for the clock frequency used. CPU operation under these conditions may cause the Program Counter to perform unintentional jumps and eventually execute the EEPROM write code. To secure EEPROM integrity, the user is advised to use an external under-voltage reset circuit in this case.

In order to prevent unintentional EEPROM writes, a specific write procedure must be followed. Refer to the description of the EEPROM Control Register for details on this.

When the EEPROM is written, the CPU is halted for two clock cycles before the next instruction is executed. When the EEPROM is read, the CPU is halted for four clock cycles before the next instruction is executed.

EEPROM Address Register – EEAR

Bit	7	6	5	4	3	2	1	0	
$1E ($3E)	–	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEAR
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7 – Res: Reserved Bit

This bit is a reserved bit in the AT90S2313 and will always read as zero.

• Bit 6..0 – EEAR6..0: EEPROM Address

The EEPROM Address Register (EEAR6..0) specifies the EEPROM address in the 128 bytes EEPROM space. The EEPROM data bytes are addressed linearly between 0 and 127.

EEPROM Data Register – EEDR

Bit	7	6	5	4	3	2	1	0	
$1D ($3D)	MSB							LSB	EEDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7..0 – EEDR7..0: EEPROM Data

For the EEPROM write operation, the EEDR register contains the data to be written to the EEPROM in the address given by the EEAR register. For the EEPROM read operation, the EEDR contains the data read out from the EEPROM at the address given by EEAR.

EEPROM Control Register – EECR

Bit	7	6	5	4	3	2	1	0	
$1C ($3C)	–	–	–	–	–	EEMWE	EEWE	EERE	EECR
Read/Write	R	R	R	R	R	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

• Bit 7..3 – Res: Reserved Bits

These bits are reserved bits in the AT90S2313 and will always read as zero.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• **Bit 2 – EEMWE: EEPROM Master Write Enable**

The EEMWE bit determines whether setting EEWE to one causes the EEPROM to be written. When EEMWE is set (one), setting EEWE will write data to the EEPROM at the selected address. If EEMWE is zero, setting EEWE will have no effect. When EEMWE has been set (one) by software, hardware clears the bit to zero after four clock cycles. See the description of the EEWE bit for a EEPROM write procedure.

• **Bit 1 – EEWE: EEPROM Write Enable**

The EEPROM Write Enable Signal (EEWE) is the write strobe to the EEPROM. When address and data are correctly set up, the EEWE bit must be set to write the value into the EEPROM. The EEMWE bit must be set when the logical "1" is written to EEWE, otherwise no EEPROM write takes place. The following procedure should be followed when writing the EEPROM (the order of steps 2 and 3 is unessential):

1. Wait until EEWE becomes zero.
2. Write new EEPROM address to EEAR (optional).
3. Write new EEPROM data to EEDR (optional).
4. Write a logical "1" to the EEMWE bit in EECR (to be able to write a logical "1" to the EEMWE bit, the EEWE bit must be written to zero in the same cycle).
5. Within four clock cycles after setting EEMWE, write a logical "1" to EEWE.

When the write access time (typically 2.5 ms at V_{CC} = 5V or 4 ms at V_{CC} = 2.7V) has elapsed, the EEWE bit is cleared (zero) by hardware. The user software can poll this bit and wait for a zero before writing the next byte. When EEWE has been set, the CPU is halted for two cycles before the next instruction is executed.

Caution: An interrupt between step 4 and step 5 will make the write cycle fail, since the EEPROM Master Write Enable will time-out. If an interrupt routine accessing the EEPROM is interrupting another EEPROM access, the EEAR or EEDR register will be modified, causing the interrupted EEPROM access to fail. It is recommended to have the global interrupt flag cleared during the last four steps to avoid these problems.

• **Bit 0 – EERE: EEPROM Read Enable**

The EEPROM Read Enable Signal (EERE) is the read strobe to the EEPROM. When the correct address is set up in the EEAR register, the EERE bit must be set. When the EERE bit is cleared (zero) by hardware, requested data is found in the EEDR register. The EEPROM read access takes one instruction and there is no need to poll the EERE bit. When EERE has been set, the CPU is halted for four cycles before the next instruction is executed.

The user should poll the EEWE bit before starting the read operation. If a write operation is in progress when new data or address is written to the EEPROM I/O registers, the write operation will be interrupted and the result is undefined.

Prevent EEPROM Corruption

During periods of low V_{CC}, the EEPROM data can be corrupted because the supply voltage is too low for the CPU and the EEPROM to operate properly. These issues are the same as for board-level systems using the EEPROM, and the same design solutions should be applied.

An EEPROM data corruption can be caused by two situations when the voltage is too low. First, a regular write sequence to the EEPROM requires a minimum voltage to operate correctly. Secondly, the CPU itself can execute instructions incorrectly if the supply voltage for executing instructions is too low.

EEPROM data corruption can easily be avoided by following these design recommendations (one is sufficient):

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Keep the AVR RESET active (low) during periods of insufficient power supply voltage. This is best done by an external low V_{CC} Reset Protection circuit, often referred to as a Brown-out Detector (BOD). Please refer to the AVR 180 application note for design considerations regarding power-on reset and low-voltage detection.
2. Keep the AVR core in Power-down Sleep Mode during periods of low V_{CC}. This will prevent the CPU from attempting to decode and execute instructions, effectively protecting the EEPROM registers from unintentional writes.
3. Store constants in Flash memory if the ability to change memory contents from software is not required. Flash memory cannot be updated by the CPU and will not be subject to corruption.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UART

The AT90S2313 features a full duplex (separate receive and transmit registers) Universal Asynchronous Receiver and Transmitter (UART). The main features are:

- **Baud Rate Generator that can Generate a Large Number of Baud Rates (bps)**
- **High Baud Rates at Low XTAL Frequencies**
- **8 or 9 Bits Data**
- **Noise Filtering**
- **Overrun Detection**
- **Framing Error Detection**
- **False Start Bit Detection**
- **Three separate Interrupts on TX Complete, TX Data Register Empty and RX Complete**

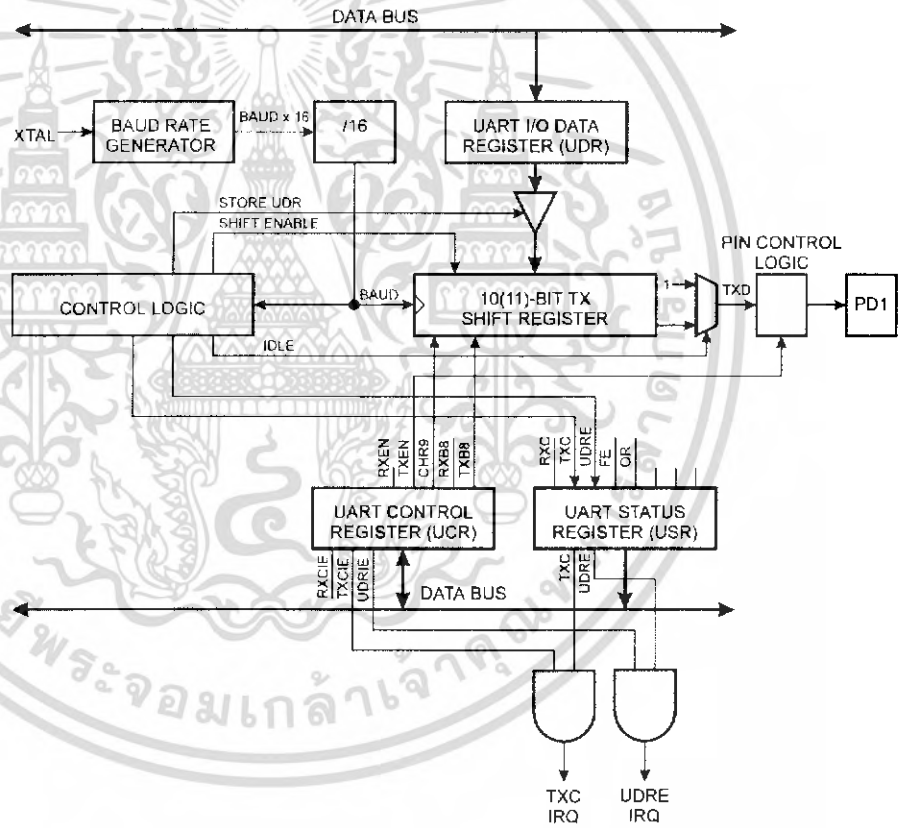
Data Transmission

A block schematic of the UART transmitter is shown in Figure 34.

Data transmission is initiated by writing the data to be transmitted to the UART I/O Data Register (UDR). Data is transferred from UDR to the Transmit shift register when:

- A new character has been written to UDR after the stop bit from the previous character has been shifted out. The shift register is loaded immediately.
- A new character has been written to UDR before the stop bit from the previous character has been shifted out. The shift register is loaded when the stop bit of the character currently being transmitted has been shifted out.

Figure 34. UART Transmitter



If the 10(11)-bit Transmitter shift register is empty, data is transferred from UDR to the shift register. At this time the UDRE (UART Data Register Empty) bit in the UART Status

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register (USR) is set. When this bit is set (one), the UART is ready to receive the next character. At the same time as the data is transferred from UDR to the 10(11)-bit shift register, bit 0 of the shift register is cleared (start bit) and bit 9 or 10 is set (stop bit). If 9-bit data word is selected (the CHR9 bit in the UART Control Register [UCR] is set), the TXB8 bit in UCR is transferred to bit 9 in the Transmit shift register.

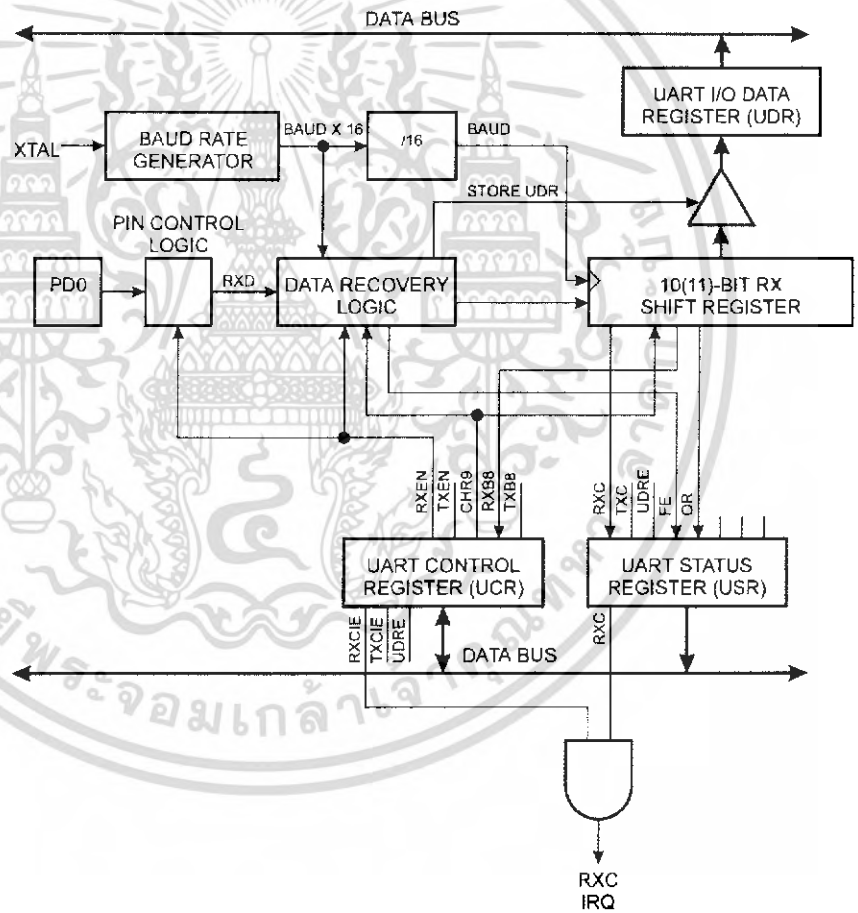
On the Baud Rate clock following the transfer operation to the shift register, the start bit is shifted out on the TXD pin. Then follows the data, LSB first. When the stop bit has been shifted out, the shift register is loaded if any new data has been written to the UDR during the transmission. During loading, UDRE is set. If there is no new data in the UDR register to send when the stop bit is shifted out, the UDRE flag will remain set until UDR is written again. When no new data has been written, and the stop bit has been present on TXD for one bit length, the TX Complete Flag (TXC) in USR is set.

The TXEN bit in UCR enables the UART transmitter when set (one). When this bit is cleared (zero), the PD1 pin can be used for general I/O. When TXEN is set, the UART Transmitter will be connected to PD1, which is forced to be an output pin regardless of the setting of the DDD1 bit in DDRD.

Data Reception

Figure 35 shows a block diagram of the UART Receiver.

Figure 35. UART Receiver

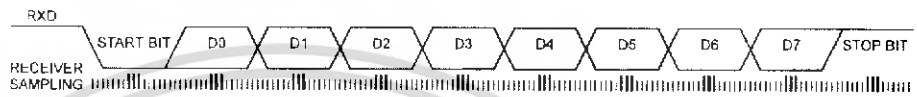


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The receiver front-end logic samples the signal on the RXD pin at a frequency of 16 times the baud rate. While the line is idle, one single sample of logical "0" will be interpreted as the falling edge of a start bit, and the start bit detection sequence is initiated. Let sample 1 denote the first zero-sample. Following the 1-to-0 transition, the receiver samples the RXD pin at samples 8, 9 and 10. If two or more of these three samples are found to be logical "1"s, the start bit is rejected as a noise spike and the receiver starts looking for the next 1-to-0 transition.

If, however, a valid start bit is detected, sampling of the data bits following the start bit is performed. These bits are also sampled at samples 8, 9 and 10. The logical value found in at least two of the three samples is taken as the bit value. All bits are shifted into the transmitter shift register as they are sampled. Sampling of an incoming character is shown in Figure 36.

Figure 36. Sampling Received Data



When the stop bit enters the receiver, the majority of the three samples must be "1" to accept the stop bit. If two or more samples are logical "0"s, the Framing Error (FE) flag in the UART Status Register (USR) is set. Before reading the UDR register, the user should always check the FE bit to detect Framing Errors.

Whether or not a valid stop bit is detected at the end of a character-reception cycle, the data is transferred to UDR and the RXC flag in USR is set. UDR is in fact two physically separate registers; one for transmitted data and one for received data. When UDR is read, the Receive Data register is accessed, and when UDR is written, the Transmit Data register is accessed. If 9-bit data word is selected (the CHR9 bit in the UART Control Register [UCR] is set), the RXB8 bit in UCR is loaded with bit 9 in the Transmit shift register when data is transferred to UDR.

If, after having received a character, the UDR register has not been read since the last receive, the OverRun (OR) flag in UCR is set. This means that the last data byte shifted into the shift register could not be transferred to UDR and has been lost. The OR bit is buffered and is updated when the valid data byte in UDR is read. Thus, the user should always check the OR bit after reading the UDR register in order to detect any overruns if the baud rate is high or CPU load is high.

When the RXEN bit in the UCR register is cleared (zero), the receiver is disabled. This means that the PD0 pin can be used as a general I/O pin. When RXEN is set, the UART Receiver will be connected to PD0, which is forced to be an input pin regardless of the setting of the DDD0 bit in DDRD. When PD0 is forced to input by the UART, the PORTD0 bit can still be used to control the pull-up resistor on the pin.

When the CHR9 bit in the UCR register is set, transmitted and received characters are nine bits long plus start and stop bits. The ninth data bit to be transmitted is the TXB8 bit in UCR register. This bit must be set to the wanted value before a transmission is initiated by writing to the UDR register. The ninth data bit received is the RXB8 bit in the UCR register.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UART Control

The UART I/O Data Register – UDR

Bit	7	6	5	4	3	2	1	0	
$0C ($2C)	MSB							LSB	UDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

The UDR register is actually two physically separate registers sharing the same I/O address. When writing to the register, the UART Transmit Data register is written. When reading from UDR, the UART Receive Data register is read.

UART Status Register – USR

Bit	7	6	5	4	3	2	1	0	
$0B ($2B)	RXC	TXC	UDRE	FE	OR	–	–	–	USR
Read/Write	R	R/W	R	R	R	R	R	R	
Initial value	0	0	1	0	0	0	0	0	

The USR register is a read-only register providing information on the UART status.

• Bit 7 – RXC: UART Receive Complete

This bit is set (one) when a received character is transferred from the Receiver Shift register to UDR. The bit is set regardless of any detected framing errors. When the RXCIE bit in UCR is set, the UART Receive Complete interrupt will be executed when RXC is set (one). RXC is cleared by reading UDR. When interrupt-driven data reception is used, the UART Receive Complete Interrupt routine must read UDR in order to clear RXC, otherwise a new interrupt will occur once the interrupt routine terminates.

• Bit 6 – TXC: UART Transmit Complete

This bit is set (one) when the entire character (including the stop bit) in the Transmit Shift register has been shifted out and no new data has been written to UDR. This flag is especially useful in half-duplex communications interfaces, where a transmitting application must enter receive mode and free the communications bus immediately after completing the transmission.

When the TXCIE bit in UCR is set, setting of TXC causes the UART Transmit Complete interrupt to be executed. TXC is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, the TXC bit is cleared (zero) by writing a logical "1" to the bit.

• Bit 5 – UDRE: UART Data Register Empty

This bit is set (one) when a character written to UDR is transferred to the Transmit shift register. Setting of this bit indicates that the transmitter is ready to receive a new character for transmission.

When the UDRIE bit in UCR is set, the UART Transmit Complete interrupt is executed as long as UDRE is set. UDRE is cleared by writing UDR. When interrupt-driven data transmittal is used, the UART Data Register Empty Interrupt routine must write UDR in order to clear UDRE, otherwise a new interrupt will occur once the interrupt routine terminates.

UDRE is set (one) during reset to indicate that the transmitter is ready.

• Bit 4 – FE: Framing Error

This bit is set if a Framing Error condition is detected (i.e., when the stop bit of an incoming character is zero).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The FE bit is cleared when the stop bit of received data is one.

• **Bit 3 – OR: Overrun**

This bit is set if an Overrun condition is detected (i.e., when a character already present in the UDR register is not read before the next character has been shifted into the Receiver Shift register). The OR bit is buffered, which means that it will be set once the valid data still in UDRE is read.

The OR bit is cleared (zero) when data is received and transferred to UDR.

• **Bits 2..0 – Res: Reserved Bits**

These bits are reserved bits in the AT90S2313 and will always read as zero.

UART Control Register – UCR

Bit	7	6	5	4	3	2	1	0	
$0A ($2A)	RXCIE	TXCIE	UDRIE	RXEN	TXEN	CHR9	RXB8	TXB8	UCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	W	
Initial value	0	0	0	0	0	0	1	0	

• **Bit 7 – RXCIE: RX Complete Interrupt Enable**

When this bit is set (one), a setting of the RXC bit in USR will cause the Receive Complete Interrupt routine to be executed provided that global interrupts are enabled.

• **Bit 6 – TXCIE: TX Complete Interrupt Enable**

When this bit is set (one), a setting of the TXC bit in USR will cause the Transmit Complete Interrupt routine to be executed provided that global interrupts are enabled.

• **Bit 5 – UDRIE: UART Data Register Empty Interrupt Enable**

When this bit is set (one), a setting of the UDRE bit in USR will cause the UART Data Register Empty Interrupt routine to be executed provided that global interrupts are enabled.

• **Bit 4 – RXEN: Receiver Enable**

This bit enables the UART receiver when set (one). When the receiver is disabled, the RXC, OR and FE status flags cannot become set. If these flags are set, turning off RXEN does not cause them to be cleared.

• **Bit 3 – TXEN: Transmitter Enable**

This bit enables the UART transmitter when set (one). When disabling the transmitter while transmitting a character, the transmitter is not disabled before the character in the shift register plus any following character in UDR has been completely transmitted.

• **Bit 2 – CHR9: 9 Bit Characters**

When this bit is set (one), transmitted and received characters are nine bits long plus start and stop bits. The ninth bit is read and written by using the RXB8 and TXB8 bits in UCR, respectively. The ninth data bit can be used as an extra stop bit or a parity bit.

• **Bit 1 – RXB8: Receive Data Bit 8**

When CHR9 is set (one), RXB8 is the ninth data bit of the received character.

• **Bit 0 – TXB8: Transmit Data Bit 8**

When CHR9 is set (one), TXB8 is the ninth data bit in the character to be transmitted.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Baud Rate Generator

The baud rate generator is a frequency divider that generates baud rates according to the following equation:

$$BAUD = \frac{f_{CK}}{16(UBRR + 1)}$$

- BAUD = Baud Rate
- f_{CK} = Crystal Clock frequency
- UBRR = Contents of the UART Baud Rate register (UBRR) (0 - 255)

For standard crystal frequencies, the most commonly used baud rates can be generated by using the UBRR settings in Table 15. UBRR values that yield an actual baud rate differing less than 2% from the target baud rate, are boldfaced in the table. However, using baud rates that have more than 1% error is not recommended. High error ratings give less noise resistance.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 15. UBRR Settings at Various Crystal Frequencies

Baud Rate	1 MHz	%Error	1.8432 MHz	%Error	2 MHz	%Error	2.4576 MHz	%Error
2400	UBRR= **25**	**0.2**	UBRR= **47**	**0.0**	UBRR= **51**	**0.2**	UBRR= **63**	**0.0**
4800	UBRR= **12**	**0.2**	UBRR= **23**	**0.0**	UBRR= **25**	**0.2**	UBRR= **31**	**0.0**
9600	UBRR= 6	7.5	UBRR= **11**	**0.0**	UBRR= **12**	**0.2**	UBRR= **15**	**0.0**
14400	UBRR= 3	7.8	UBRR= **7**	**0.0**	UBRR= 8	3.7	UBRR= 10	3.1
19200	UBRR= 2	7.8	UBRR= **5**	**0.0**	UBRR= 6	7.5	UBRR= **7**	**0.0**
28800	UBRR= 1	7.8	UBRR= **3**	**0.0**	UBRR= 3	7.8	UBRR= 4	6.3
38400	UBRR= 1	22.9	UBRR= **2**	**0.0**	UBRR= 2	7.8	UBRR= **3**	**0.0**
57600	UBRR= 0	7.8	UBRR= **1**	**0.0**	UBRR= 1	7.8	UBRR= 2	12.5
76800	UBRR= 0	22.9	UBRR= 1	33.3	UBRR= 1	22.9	UBRR= **1**	**0.0**
115200	UBRR= 0	84.3	UBRR= **0**	**0.0**	UBRR= 0	7.8	UBRR= 0	25.0

Baud Rate	3.2768 MHz	%Error	3.6864 MHz	%Error	4 MHz	%Error	4.608 MHz	%Error
2400	UBRR= **84**	**0.4**	UBRR= **95**	**0.0**	UBRR= **103**	**0.2**	UBRR= **119**	**0.0**
4800	UBRR= **42**	**0.8**	UBRR= **47**	**0.0**	UBRR= **51**	**0.2**	UBRR= **59**	**0.0**
9600	UBRR= **20**	**1.6**	UBRR= **23**	**0.0**	UBRR= **25**	**0.2**	UBRR= **29**	**0.0**
14400	UBRR= **13**	**1.6**	UBRR= **15**	**0.0**	UBRR= 16	2.1	UBRR= **19**	**0.0**
19200	UBRR= 10	3.1	UBRR= **11**	**0.0**	UBRR= **12**	**0.2**	UBRR= **14**	**0.0**
28800	UBRR= **6**	**1.6**	UBRR= **7**	**0.0**	UBRR= 8	3.7	UBRR= **9**	**0.0**
38400	UBRR= 4	6.3	UBRR= **5**	**0.0**	UBRR= 6	7.5	UBRR= 7	6.7
57600	UBRR= 3	12.5	UBRR= **3**	**0.0**	UBRR= 3	7.8	UBRR= **4**	**0.0**
76800	UBRR= 2	12.5	UBRR= **2**	**0.0**	UBRR= 2	7.8	UBRR= 3	6.7
115200	UBRR= 1	12.5	UBRR= **1**	**0.0**	UBRR= 1	7.8	UBRR= 2	20.0

Baud Rate	7.3728 MHz	%Error	8 MHz	%Error	9.216 MHz	%Error	11.059 MHz	%Error
2400	UBRR= **191**	**0.0**	UBRR= **207**	**0.2**	UBRR= **239**	**0.0**	UBRR= 287	-
4800	UBRR= **95**	**0.0**	UBRR= **103**	**0.2**	UBRR= **119**	**0.0**	UBRR= **143**	**0.0**
9600	UBRR= **47**	**0.0**	UBRR= **51**	**0.2**	UBRR= **59**	**0.0**	UBRR= **71**	**0.0**
14400	UBRR= **31**	**0.0**	UBRR= **34**	**0.8**	UBRR= **39**	**0.0**	UBRR= **47**	**0.0**
19200	UBRR= **23**	**0.0**	UBRR= **25**	**0.2**	UBRR= **29**	**0.0**	UBRR= **35**	**0.0**
28800	UBRR= **15**	**0.0**	UBRR= 16	2.1	UBRR= **19**	**0.0**	UBRR= **23**	**0.0**
38400	UBRR= **11**	**0.0**	UBRR= **12**	**0.2**	UBRR= **14**	**0.0**	UBRR= **17**	**0.0**
57600	UBRR= **7**	**0.0**	UBRR= 8	3.7	UBRR= **9**	**0.0**	UBRR= **11**	**0.0**
76800	UBRR= **5**	**0.0**	UBRR= 6	7.5	UBRR= 7	6.7	UBRR= **8**	**0.0**
115200	UBRR= **3**	**0.0**	UBRR= 3	7.8	UBRR= **4**	**0.0**	UBRR= **5**	**0.0**

UART Baud Rate Register – UBRR

Bit	7	6	5	4	3	2	1	0	
$09 ($29)	MSB							LSB	UBRR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

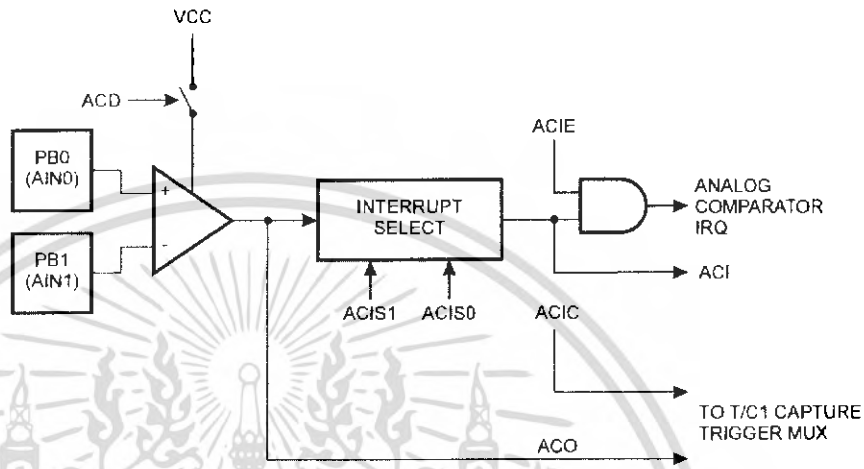
The UBRR register is an 8-bit read/write register that specifies the UART Baud Rate according to the formula on the previous page.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Analog Comparator

The Analog Comparator compares the input values on the positive input AIN0 (PB0) and the negative input PB1(AIN1). When the voltage on the positive input PB0 (AIN0) is higher than the voltage on the negative input PB1 (AIN1), the Analog Comparator Output, ACO is set (one). The comparator's output can be set to trigger the Timer/Counter1 Input Capture function. In addition, the comparator can trigger a separate interrupt exclusive to the Analog Comparator. The user can select interrupt triggering on comparator output rise, fall or toggle. A block diagram of the comparator and its surrounding logic is shown in Figure 37.

Figure 37. Analog Comparator Block Diagram



Analog Comparator Control and Status Register – ACSR

Bit	7	6	5	4	3	2	1	0	
$08 ($28)	ACD	–	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R	R	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	N/A	0	0	0	0	0	

- **Bit 7 – ACD: Analog Comparator Disable**

When this bit is set (one), the power to the Analog Comparator is switched off. This bit can be set at any time to turn off the Analog Comparator. This will reduce power consumption in active and idle modes. When changing the ACD bit, the Analog Comparator Interrupt must be disabled by clearing the ACIE bit in ACSR. Otherwise an interrupt can occur when the bit is changed.

- **Bit 6 – Res: Reserved Bit**

This bit is a reserved bit in the AT90S2313 and will always read as zero.

- **Bit 5 – ACO: Analog Comparator Output**

ACO is directly connected to the comparator output.

- **Bit 4 – ACI: Analog Comparator Interrupt Flag**

This bit is set (one) when a comparator output event triggers the interrupt mode defined by ACIS1 and ACIS0. The Analog Comparator Interrupt routine is executed if the ACIE bit is set (one) and the I-bit in SREG is set (one). ACI is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, ACI is cleared by writing a logical "1" to the flag. Observe, however, that if another bit in this register is modified using the SBI or CBI instruction, ACI will be cleared if it has become set before the operation.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• **Bit 3 – ACIE: Analog Comparator Interrupt Enable**

When the ACIE bit is set (one) and the I-bit in the Status Register is set (one), the Analog Comparator interrupt is activated. When cleared (zero), the interrupt is disabled.

• **Bit 2 – ACIC: Analog Comparator Input Capture Enable**

When set (one), this bit enables the Input Capture function in Timer/Counter1 to be triggered by the Analog Comparator. The comparator output is, in this case, directly connected to the Input Capture front-end logic, making the comparator utilize the noise canceler and edge-select features of the Timer/Counter1 Input Capture interrupt. When cleared (zero), no connection between the Analog Comparator and the Input Capture function is given. To make the comparator trigger the Timer/Counter1 Input Capture interrupt, the TICIE1 bit in the Timer Interrupt Mask Register (TIMSK) must be set (one).

• **Bits 1,0 – ACIS1, ACIS0: Analog Comparator Interrupt Mode Select**

These bits determine which comparator events trigger the Analog Comparator interrupt. The different settings are shown in Table 16.

Table 16. ACIS1/ACIS0 Settings

ACIS1	ACIS0	Interrupt Mode
0	0	Comparator Interrupt on Output Toggle
0	1	Reserved
1	0	Comparator Interrupt on Falling Output Edge
1	1	Comparator Interrupt on Rising Output Edge

Note: When changing the ACIS1/ACIS0 bits, the Analog Comparator Interrupt must be disabled by clearing its Interrupt Enable bit in the ACSR register. Otherwise an interrupt can occur when the bits are changed.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I/O Ports

All AVR ports have true read-modify-write functionality when used as general digital I/O ports. This means that the direction of one port pin can be changed without unintentionally changing the direction of any other pin with the SBI and CBI instructions. The same applies for changing drive value (if configured as output) or enabling/disabling of pull-up resistors (if configured as input).

Port B

Port B is an 8-bit bi-directional I/O port.

Three I/O memory address locations are allocated for the Port B, one each for the Data Register – PORTB, $18 ($38), Data Direction Register – DDRB, $17($37) and the Port B Input Pins – PINB, $16($36). The Port B Input Pins address is read-only, while the Data Register and the Data Direction Register are read/write.

All port pins have individually selectable pull-up resistors. The Port B output buffers can sink 20 mA and thus drive LED displays directly. When pins PB0 to PB7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated.

The Port B pins with alternate functions are shown in Table 17.

Table 17. Port B Pin Alternate Functions

Port Pin	Alternate Functions
PB0	AIN0 (Analog comparator positive input)
PB1	AIN1 (Analog comparator negative input)
PB3	OC1 (Timer/Counter1 Output compare match output)
PB5	MOSI (Data input line for memory downloading)
PB6	MISO (Data output line for memory uploading)
PB7	SCK (Serial clock input)

When the pins are used for the alternate function, the DDRB and PORTB registers have to be set according to the alternate function description.

Port B Data Register – PORTB

Bit	7	6	5	4	3	2	1	0	
$18 ($38)	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Port B Data Direction Register – DDRB

Bit	7	6	5	4	3	2	1	0	
$17 ($37)	DDB7	DOB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Port B Input Pins Address – PINB

Bit	7	6	5	4	3	2	1	0	
$16 ($36)	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Read/Write	R	R	R	R	R	R	R	R	
Initial value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The Port B Input Pins address (PINB) is not a register; this address enables access to the physical value on each Port B pin. When reading PORTB, the Port B Data Latch is read, and when reading PINB, the logical values present on the pins are read.

Port B as General Digital I/O

All eight pins in Port B have equal functionality when used as digital I/O pins.

PBn, general I/O pin: The DDBn bit in the DDRB register selects the direction of this pin. If DDBn is set (one), PBn is configured as an output pin. If DDBn is cleared (zero), PBn is configured as an input pin. If PORTBn is set (one) when the pin is configured as an input pin, the MOS pull-up resistor is activated. To switch the pull-up resistor off, the PORTBn has to be cleared (zero) or the pin has to be configured as an output pin The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not active.

Table 18. DDBn Effects on Port B Pins

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

Note: n: 7,6...0, pin number.

Alternate Functions of Port B

The alternate pin functions of Port B are:

- **SCK – Port B, Bit 7**

SCK, Clock input pin for memory up/downloading.

- **MISO – Port B, Bit 6**

MISO, Data output pin for memory uploading.

- **MOSI – Port B, Bit 5**

MOSI, Data input pin for memory downloading.

- **OC1 – Port B, Bit 3**

OC1, Output Compare Match Output. The PB3 pin can serve as an external output for timer 1 compare match. The PB3 pin has to be configured as an output (DDB3 is set [one]) to serve this function. See the timer description for further details, and how to enable the output.

- **AIN1 – Port B, Bit 1**

AIN1, Analog Comparator Negative Input. When configured as an input (DDB1 is cleared [zero]) and with the internal MOS pull-up resistor switched off (PB1 is cleared [zero]), this pin also serves as the negative input of the on-chip Analog Comparator.

- **AIN0 – Port B, Bit 0**

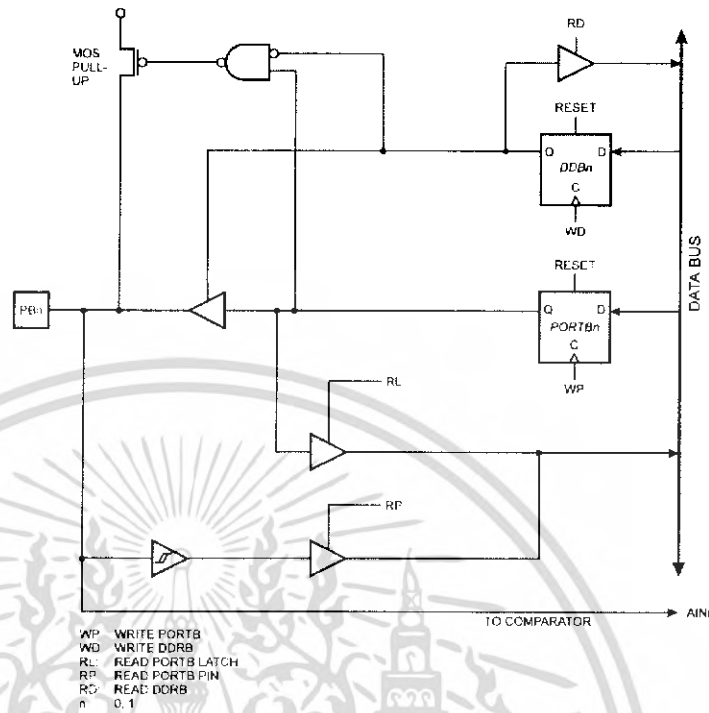
AIN0, Analog Comparator Positive Input. When configured as an input (DDB0 is cleared [zero]) and with the internal MOS pull-up resistor switched off (PB0 is cleared [zero]), this pin also serves as the positive input of the on-chip Analog Comparator.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Port B Schematics

Note that all port pins are synchronized. The synchronization latches are, however, not shown in the figures.

Figure 38. Port B Schematic Diagram (Pins PB0 and PB1)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 39. Port B Schematic Diagram (Pin PB3)

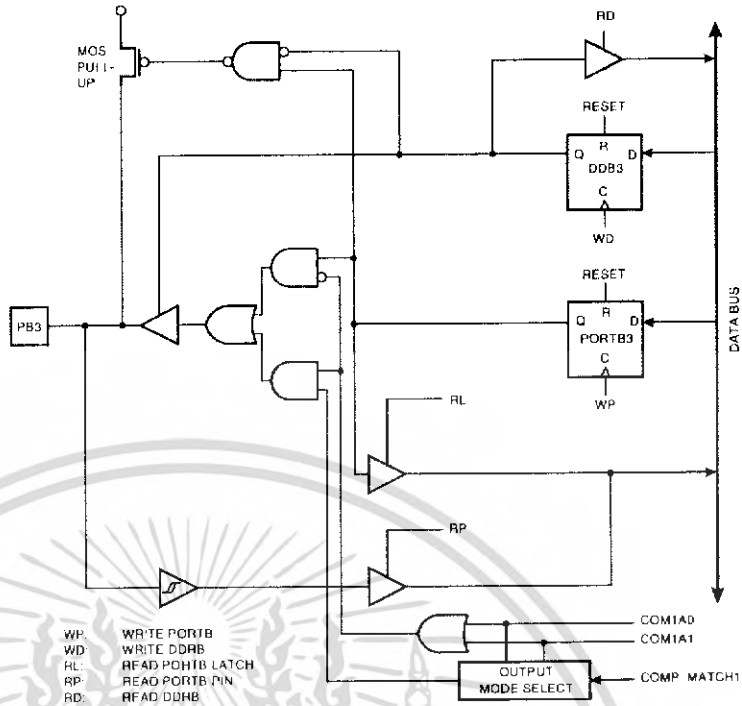
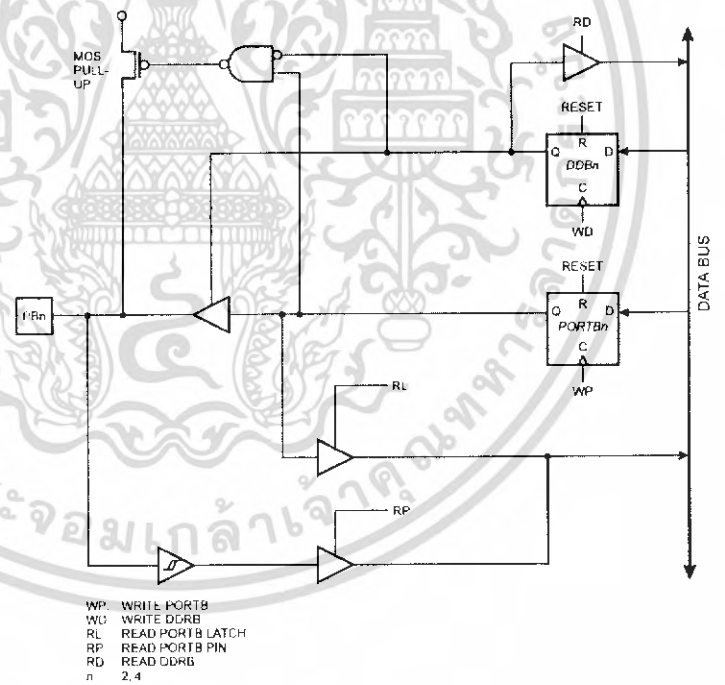


Figure 40. Port B Schematic Diagram (Pins PB2 and PB4)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 41. Port B Schematic Diagram (Pin PB5)

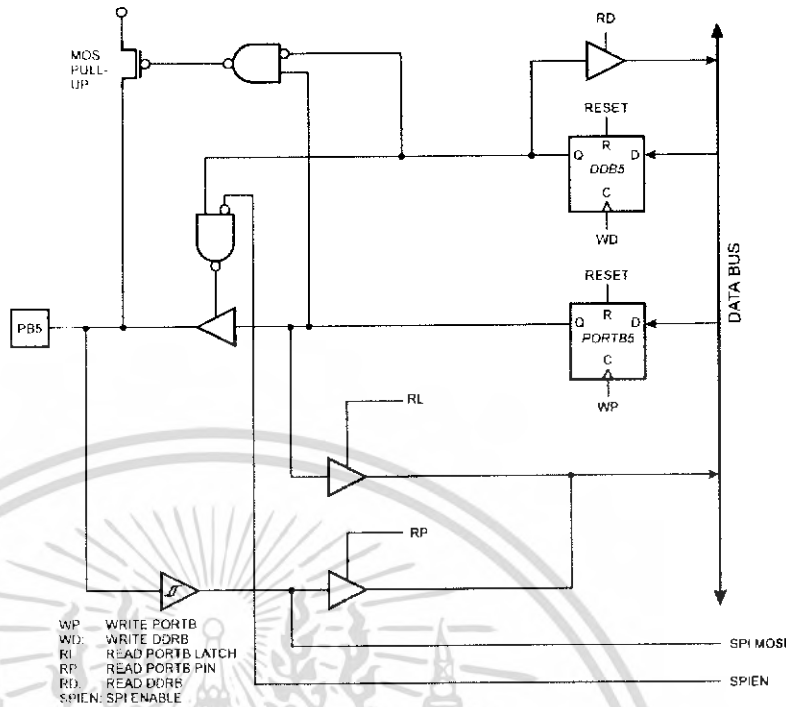
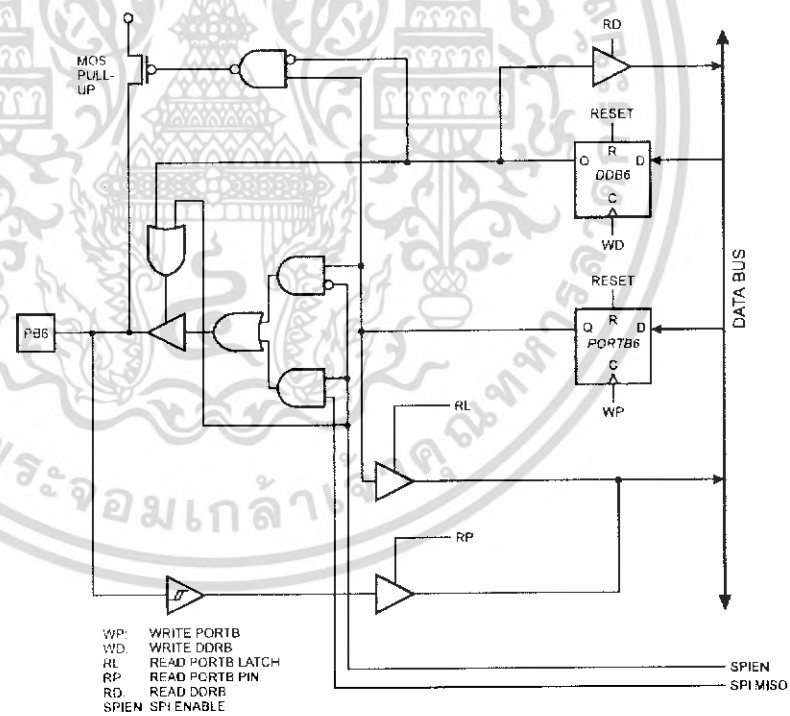
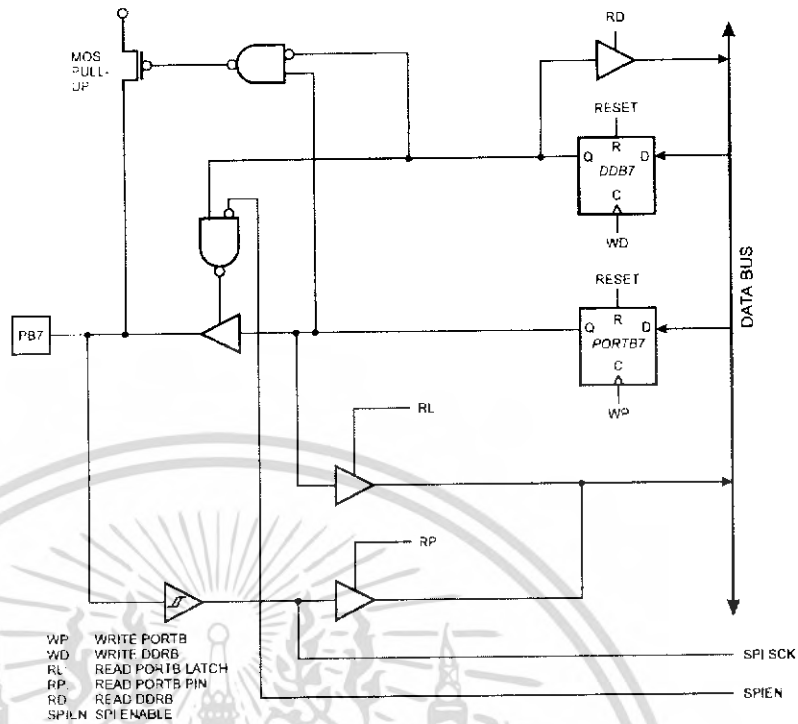


Figure 42. Port B Schematic Diagram (Pin PB6)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 43. Port B Schematic Diagram (Pin PB7)



Port D

Three I/O memory address locations are allocated for the Port D: one each for the Data Register – PORTD, $12($32), Data Direction Register – DDRD, $11($31) and the Port D Input Pins – PIND, $10($30). The Port D Input Pins address is read-only, while the Data Register and the Data Direction Register are read/write.

Port D has seven bi-directional I/O pins with internal pull-up resistors, PD6..PD0. The Port D output buffers can sink 20 mA. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated.

Some Port D pins have alternate functions as shown in Table 19:

Table 19. Port D Pin Alternate Functions

Port Pin	Alternate Function
PD0	RXD (Receive data input for the UART)
PD1	TXD (Transmit data output for the UART)
PD2	INT0 (External interrupt 0 input)
PD3	INT1 (External interrupt 1 input)
PD4	TO (Timer/Counter0 external input)
PD5	T1 (Timer/Counter1 external input)
PD6	ICP (Timer/Counter1Input Capture pin)

When the pins are used for the alternate function, the DDRD and PORTD registers have to be set according to the alternate function description.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Port D Data Register – PORTD

Bit	7	6	5	4	3	2	1	0	
$12 ($32)	–	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	PORTD
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Port D Data Direction Register – DDRD

Bit	7	6	5	4	3	2	1	0	
$11 ($31)	–	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

Port D Input Pins Address – PIND

Bit	7	6	5	4	3	2	1	0	
$10 ($30)	–	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	PIND
Read/Write	R	R	R	R	R	R	R	R	
Initial value	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

The Port D Input Pins address (PIND) is not a register; this address enables access to the physical value on each Port D pin. When reading PORTD, the Port D Data Latch is read, and when reading PIND, the logical values present on the pins are read.

Port D as General Digital I/O

PDn, general I/O pin: The DDDn bit in the DDRD register selects the direction of this pin. If DDDn is set (one), PDn is configured as an output pin. If DDDn is cleared (zero), PDn is configured as an input pin. If PORTDn is set (one) when configured as an input pin, the MOS pull-up resistor is activated. To switch the pull-up resistor off, the PORTDn has to be cleared (zero) or the pin has to be configured as an output pin. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not active.

Table 20. DDDn Bits on Port D Pins

DDDn	PORTDn	I/O	Pull-up	Comment
0	0	Input	No	Tri-state (High-Z)
0	1	Input	Yes	PDn will source current if ext. pulled low
1	0	Output	No	Push-pull Zero Output
1	1	Output	No	Push-pull One Output

Note: n: 6...0, pin number.

Alternate Functions of Port D

The alternate functions of Port D are:

- **ICP – Port D, Bit 6**

Timer/Counter1 Input Capture pin. See the Timer/Counter1 description for further details.

- **T1 – Port D, Bit 5**

T1, Timer 1 clock source. See the Timer description for further details.

- **T0 – Port D, Bit 4**

T0, Timer/Counter0 clock source. See the Timer description for further details.

- **INT1 – Port D, Bit 3**

INT1, External Interrupt Source 1. The PD3 pin can serve as an external interrupt source to the MCU. See the interrupt description for further details and how to enable the source.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• INT0 – Port D, Bit 2

INT0, External Interrupt Source 0. The PD2 pin can serve as an external interrupt source to the MCU. See the interrupt description for further details and how to enable the source.

• TXD – Port D, Bit 1

Transmit Data (Data output pin for the UART). When the UART transmitter is enabled, this pin is configured as an output regardless of the value of DDRD1.

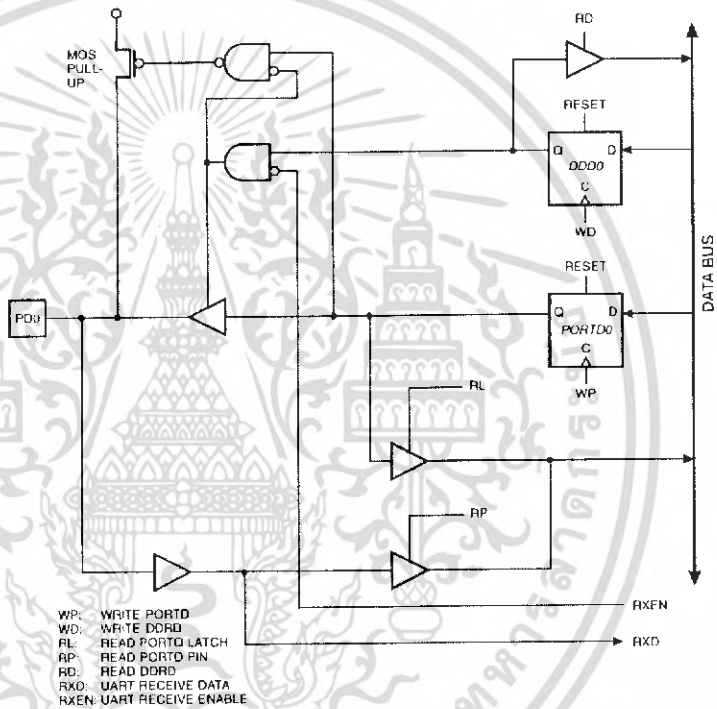
• RXD – Port D, Bit 0

Receive Data (Data input pin for the UART). When the UART receiver is enabled, this pin is configured as an input regardless of the value of DDRD0. When the UART forces this pin to be an input, a logical "1" in PORTD0 will turn on the internal pull-up.

Port D Schematics

Note that all port pins are synchronized. The synchronization latches are, however, not shown in the figures.

Figure 44. Port D Schematic Diagram (Pin PD0)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 45. Port D Schematic Diagram (Pin PD1)

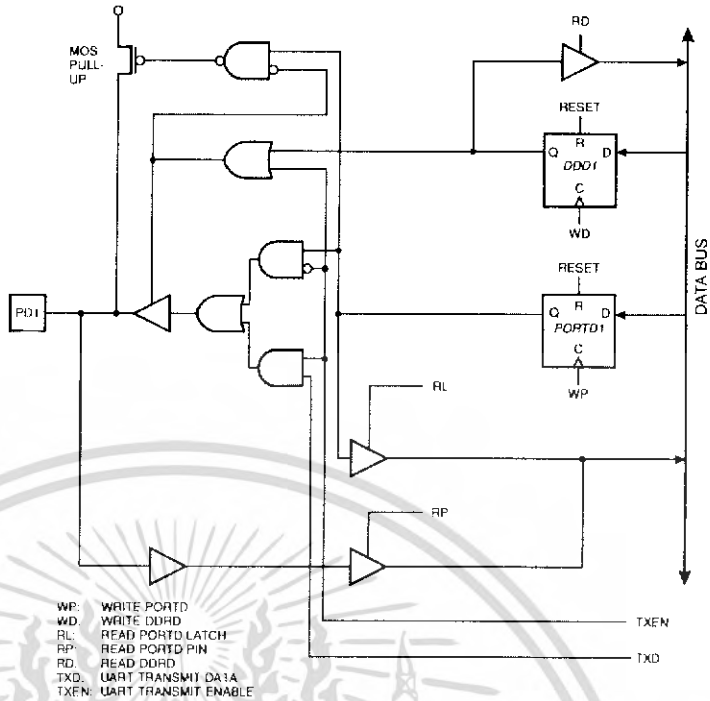
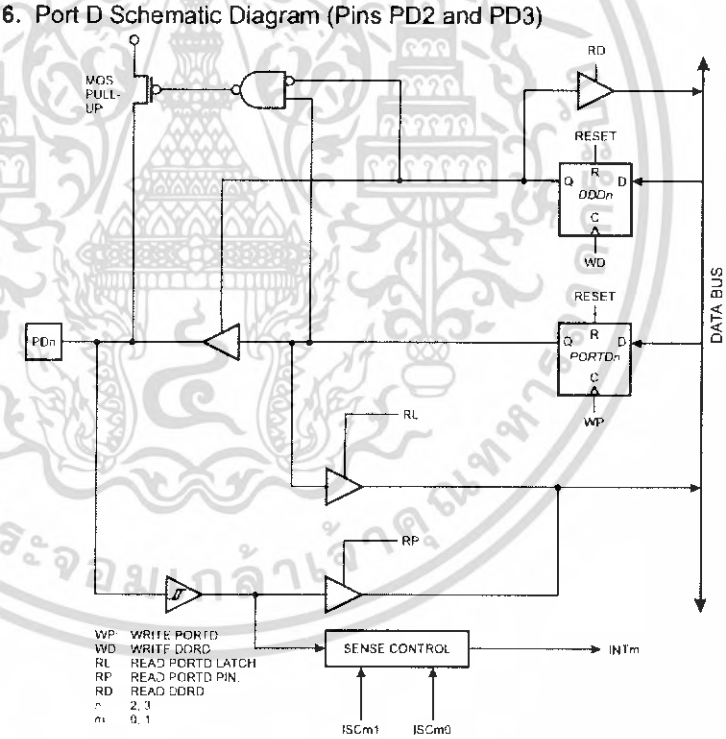


Figure 46. Port D Schematic Diagram (Pins PD2 and PD3)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 47. Port D Schematic Diagram (Pins PD4 and PD5)

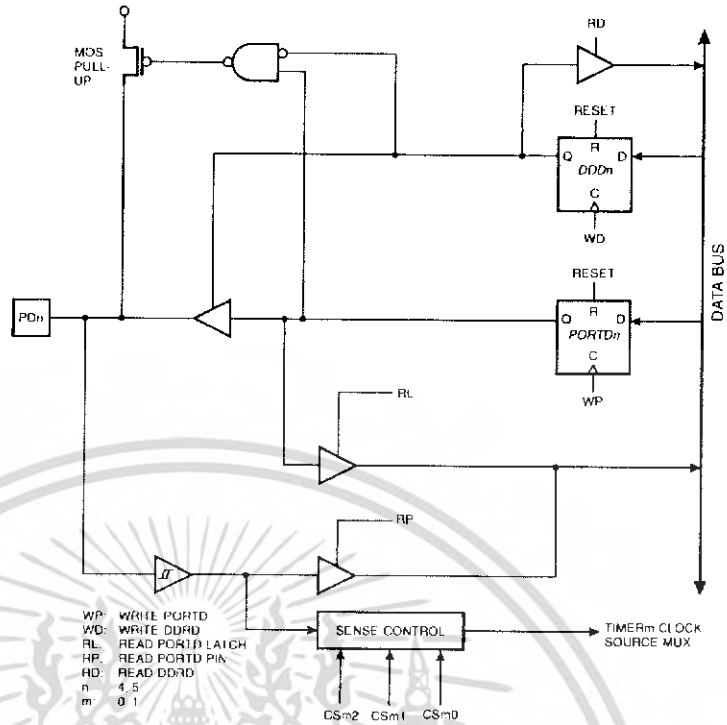
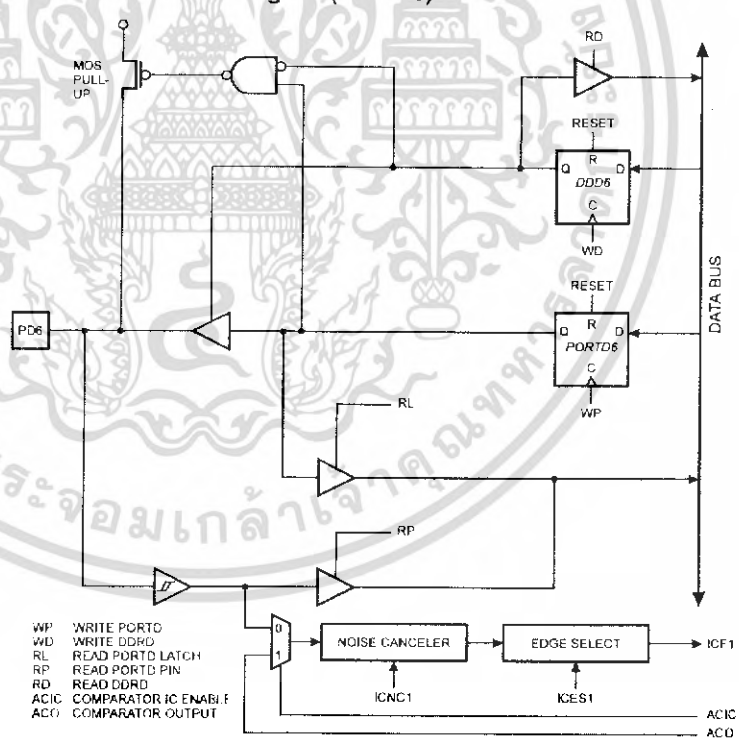


Figure 48. Port D Schematic Diagram (Pin PD6)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Memory Programming

Program and Data Memory Lock Bits

The AT90S2313 MCU provides two Lock bits that can be left unprogrammed ("1") or can be programmed ("0") to obtain the additional features listed in Table 21. The Lock bits can only be erased with the Chip Erase operation.

Table 21. Lock Bit Protection Modes

Memory Lock Bits			
Mode	LB1	LB2	Protection Type
1	1	1	No memory lock features enabled.
2	0	1	Further programming of the Flash and EEPROM is disabled.(1)
3	0	0	Same as mode 2, and verify is also disabled.

Note: 1. In the Parallel mode, further programming of the Fuse bits are also disabled. Program the Fuse bits before programming the Lock bits.

Fuse Bits

The AT90S2313 has two Fuse bits: SPIEN and FSTRT.

- When the SPIEN Fuse is programmed ("0"), Serial Program and Data Downloading is enabled. The default value is programmed ("0").
- When the FSTRT Fuse is programmed ("0"), the short start-up time is selected. The default value is unprogrammed ("1"). Parts with this bit pre-programmed ("0") can be delivered on demand.

The Fuse bits are not accessible in Serial Programming Mode. The status of the Fuses are not affected by Chip Erase.

Signature Bytes

All Atmel microcontrollers have a 3-byte signature code that identifies the device. This code can be read in both serial and parallel mode. The three bytes reside in a separate address space.

For the AT90S2313(1) they are:

1. $000: $1E (indicates manufactured by Atmel)
2. $001: $91 (indicates 2 Kb Flash memory)
3. $002: $01 (indicates AT90S2313 device when signature byte $001 is $91)

Note: 1. When both Lock bits are programmed (Lock mode 3), the signature bytes cannot be read in serial mode. Reading the signature bytes will return: $00, $01 and $02.

Programming the Flash and EEPROM

Atmel's AT90S2313 offers 2K bytes of in-system reprogrammable Flash program memory and 128 bytes of EEPROM data memory.

The AT90S2313 is shipped with the on-chip Flash program and EEPROM data memory arrays in the erased state (i.e., contents = $FF) and ready to be programmed. This device supports a high-voltage (12V) Parallel Programming Mode and a low-voltage Serial Programming Mode. The +12V is used for programming enable only, and no current of significance is drawn by this pin. The Serial Programming Mode provides a convenient way to download program and data into the AT90S2313 inside the user's system.

The program and EEPROM memory arrays in the AT90S2313 are programmed byte-by-byte in either programming mode. For the EEPROM, an auto-erase cycle is provided within the self-timed write instruction in the Serial Programming Mode. During programming, the supply voltage must be in accordance with Table 22.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 22. Supply Voltage during Programming

Part	Serial Programming	Parallel Programming
AT90S2313	2.7 - 6.0V	4.5 - 5.5V

Parallel Programming

This section describes how to parallel program and verify Flash program memory, EEPROM data memory, Lock bits and Fuse bits in the AT90S2313.

Signal Names

In this section, some pins of the AT90S2313 are referenced by signal names describing their function during parallel programming. Pins not described in the following table are referenced by pin names. See Figure 49 and Table 23. Pins not described in Table 23 are referenced by pin names.

The XA1/XA0 pins determine the action executed when the XTAL1 pin is given a positive pulse. The bit coding is shown in Table 24.

When pulsing $\overline{WR}$ or $\overline{OE}$, the command loaded determines the action executed. The command is a byte where the different bits are assigned functions as shown in Table 25.

Figure 49. Parallel Programming

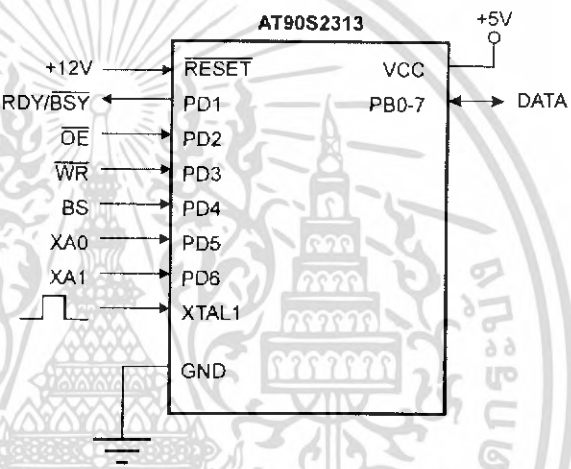


Table 23. Pin Name Mapping

Signal Name in Programming Mode	Pin Name	I/O	Function
RDY/$\overline{BSY}$	PD1	O	0: Device is busy programming, 1: Device is ready for new command
$\overline{OE}$	PD2	I	Output Enable (Active low)
$\overline{WR}$	PD3	I	Write Pulse (Active low)
BS	PD4	I	Byte Select ("0" selects low byte, "1" selects high byte)
XA0	PD5	I	XTAL Action Bit 0
XA1	PD6	I	XTAL Action Bit 1
DATA	PB7-0	I/O	Bi-directional Data Bus (Output when $\overline{OE}$ is low)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 24. XA1 and XA0 Coding

XA1	XA0	Action when XTAL1 is Pulsed
0	0	Load Flash or EEPROM Address (High or low address byte determined by BS)
0	1	Load Data (High or Low data byte for Flash determined by BS)
1	0	Load Command
1	1	No Action, Idle

Table 25. Command Byte Bit Coding

Command Byte	Command Executed
1000 0000	Chip Erase
0100 0000	Write Fuse Bits
0010 0000	Write Lock Bits
0001 0000	Write Flash
0001 0001	Write EEPROM
0000 1000	Read Signature Bytes
0000 0100	Read Fuse and Lock Bits
0000 0010	Read Flash
0000 0011	Read EEPROM

Enter Programming Mode

The following algorithm puts the device in Parallel Programming Mode:

1. Apply supply voltage according to Table 22, between V_{CC} and GND.
2. Set the $\overline{\text{RESET}}$ and BS pin to "0" and wait at least 100 ns.
3. Apply 11.5 - 12.5V to $\overline{\text{RESET}}$. Any activity on BS within 100 ns after +12V has been applied to $\overline{\text{RESET}}$, will cause the device to fail entering Programming Mode.

Chip Erase

The Chip Erase command will erase the Flash and EEPROM memories, and the Lock bits. The Lock bits are not reset until the Flash and EEPROM have been completely erased. The Fuse bits are not changed. Chip Erase must be performed before the Flash or EEPROM is reprogrammed.

Load Command "Chip Erase"

1. Set XA1, XA0 to "10". This enables command loading.
2. Set BS to "0".
3. Set DATA to "1000 0000". This is the command for Chip Erase.
4. Give XTAL1 a positive pulse. This loads the command.
5. Give $\overline{\text{WR}}$ a t_{WLWH_CE} wide negative pulse to execute Chip Erase. See Table 26 for t_{WLWH_CE} value. Chip Erase does not generate any activity on the RDY/BSY pin.

Programming the Flash

A: Load Command "Write Flash"

1. Set XA1, XA0 to "10". This enables command loading.
2. Set BS to "0".
3. Set DATA to "0001 0000". This is the command for Write Flash.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Give XTAL1 a positive pulse. This loads the command.

B: Load Address High Byte

1. Set XA1, XA0 to "00". This enables address loading.
2. Set BS to "1". This selects high byte.
3. Set DATA = Address high byte ($00 - $03).
4. Give XTAL1 a positive pulse. This loads the address high byte.

C: Load Address Low Byte

1. Set XA1, XA0 to "00". This enables address loading.
2. Set BS to "0". This selects low byte.
3. Set DATA = Address low byte ($00 - $FF).
4. Give XTAL1 a positive pulse. This loads the address low byte.

D: Load Data Low Byte

1. Set XA1, XA0 to "01". This enables data loading.
2. Set DATA = Data low byte ($00 - $FF).
3. Give XTAL1 a positive pulse. This loads the data low byte.

E: Write Data Low Byte

1. Set BS to "0". This selects low data.
2. Give $\overline{WR}$ a negative pulse. This starts programming of the data byte. RDY/$\overline{BSY}$ goes low.
3. Wait until RDY/$\overline{BSY}$ goes high to program the next byte.

(See Figure 50 for signal waveforms.)

F: Load Data High Byte

1. Set XA1, XA0 to "01". This enables data loading.
2. Set DATA = Data high byte ($00 - $FF).
3. Give XTAL1 a positive pulse. This loads the data high byte.

G: Write Data High Byte

1. Set BS to "1". This selects high data.
2. Give $\overline{WR}$ a negative pulse. This starts programming of the data byte. RDY/$\overline{BSY}$ goes low.
3. Wait until RDY/$\overline{BSY}$ goes high to program the next byte.

(See Figure 51 for signal waveforms.)

The loaded command and address are retained in the device during programming. For efficient programming, the following should be considered:

- The command needs only be loaded once when writing or reading multiple memory locations.
- Address high byte needs only be loaded before programming a new 256-word page in the Flash.
- Skip writing the data value $FF; that is, the contents of the entire Flash and EEPROM after a Chip Erase.

These considerations also apply to EEPROM programming and Flash, EEPROM and signature byte reading.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 50. Programming the Flash

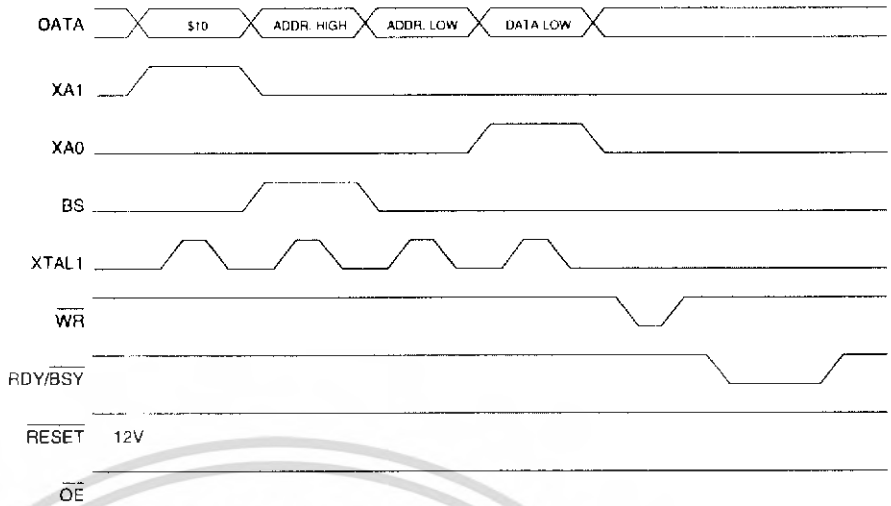


Figure 51. Programming the Flash (Continued)

Reading the Flash

The algorithm for reading the Flash memory is as follows (refer to "Programming the Flash" for details on command and address loading):

1. A: Load Command "0000 0010".
2. B: Load Address High Byte ($00 - $03).
3. C: Load Address Low Byte ($00 - $FF).
4. Set $\overline{OE}$ to "0", and BS to "0". The Flash word low byte can now be read at DATA.
5. Set BS to "1". The Flash word high byte can now be read from DATA.
6. Set $\overline{OE}$ to "1".

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programming the EEPROM

The programming algorithm for the EEPROM data memory is as follows (refer to "Programming the Flash" for details on command, address and data loading):

1. A: Load Command "0001 0001".
2. C: Load Address Low Byte ($00 - $7F).
3. D: Load Data Low Byte ($00 - $FF).
4. E: Write Data Low Byte.

Reading the EEPROM

The algorithm for reading the EEPROM memory is as follows (refer to "Programming the Flash" for details on command and address loading):

1. A: Load Command "0000 0011".
2. C: Load Address Low Byte ($00 - $7F).
3. Set $\overline{OE}$ to "0", and BS to "0". The EEPROM data byte can now be read at DATA.
4. Set $\overline{OE}$ to "1".

Programming the Fuse Bits

The algorithm for programming the Fuse bits is as follows (refer to "Programming the Flash" for details on command and data loading):

1. A: Load Command "0100 0000".
2. D: Load Data Low Byte. Bit n = "0" programs and bit n = "1" erases the Fuse bit.
   Bit 5 = SPIEN Fuse bit.
   Bit 0 = FSTRT Fuse bit.
   Bit 7-6,4-1 = "1". These bits are reserved and should be left unprogrammed ("1").
3. Give $\overline{WR}$ a t_{WLWH_PFB} wide negative pulse to execute the programming; t_{WLWH_PFB} is found in Table 26. Programming the Fuse bits does not generate any activity on the RDY/$\overline{BSY}$ pin.

Programming the Lock Bits

The algorithm for programming the Lock bits is as follows (refer to "Programming the Flash" on page 64 for details on command and data loading):

1. A: Load Command "0010 0000".
2. D: Load Data Low Byte. Bit n = "0" programs the Lock bit.
   Bit 2 = Lock Bit2
   Bit 1 = Lock Bit1
   Bit 7-3,0 = "1". These bits are reserved and should be left unprogrammed ("1").
3. E: Write Data Low Byte.

The Lock bits can only be cleared by executing Chip Erase.

Reading the Fuse and Lock Bits

The algorithm for reading the Fuse and Lock bits is as follows (refer to "Programming the Flash" on page 64 for details on command loading):

1. A: Load Command "0000 0100".
2. Set $\overline{OE}$ to "0", and BS to "1". The status of the Fuse and Lock bits can now be read at DATA ("0" means programmed).
   Bit 7 = Lock Bit1
   Bit 6 = Lock Bit2
   Bit 5 = SPIEN Fuse bit
   Bit 0 = FSTRT Fuse bit
3. Set $\overline{OE}$ to "1".

Observe that BS needs to be set to "1".

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Reading the Signature Bytes

The algorithm for reading the signature bytes is as follows (refer to "Programming the Flash" on page 64 for details on command and address loading):

1. A: Load Command "0000 1000".
2. C: Load Address Low Byte ($00 - $02).
   Set $\overline{OE}$ to "0", and BS to "0". The selected signature byte can now be read at DATA.
3. Set $\overline{OE}$ to "1".

Parallel Programming Characteristics

Figure 52. Parallel Programming Timing

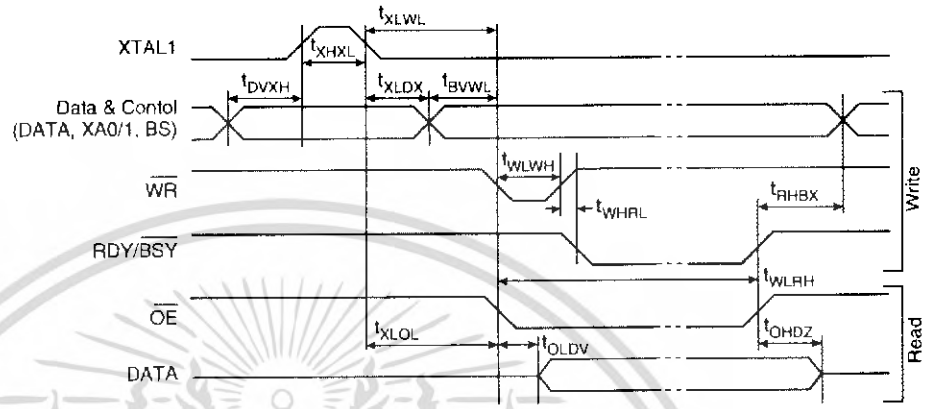


Table 26. Parallel Programming Characteristics, T_A = 25°C ± 10%, V_{CC} = 5V ± 10%

Symbol	Parameter	Min	Typ	Max	Units
V_{PP}	Programming Enable Voltage	11.5		12.5	V
I_{PP}	Programming Enable Current			250.0	µA
t_{DVXH}	Data and Control Setup before XTAL1 High	67.0			ns
t_{XHXL}	XTAL1 Pulse Width High	67.0			ns
t_{XLDX}	Data and Control Hold after XTAL1 Low	67.0			ns
t_{XLWL}	XTAL1 Low to $\overline{WR}$ Low	67.0			ns
t_{BVWL}	BS Valid to $\overline{WR}$ Low	67.0			ns
t_{RHBX}	BS Hold after RDY/$\overline{BSY}$ High	67.0			ns
t_{WLWH}	$\overline{WR}$ Pulse Width Low(1)	67.0			ns
t_{WHRL}	$\overline{WR}$ High to RDY/$\overline{BSY}$ Low(2)		20.0		ns
t_{WLRH}	$\overline{WR}$ Low to RDY/$\overline{BSY}$ High(2)	0.5	0.7	0.9	ms
t_{XLOL}	XTAL1 Low to $\overline{OE}$ Low	67.0			ns
t_{OLDV}	$\overline{OE}$ Low to DATA Valid		20.0		ns
t_{OHDZ}	$\overline{OE}$ High to DATA Tri-stated			20.0	ns
t_{WLWH_CE}	$\overline{WR}$ Pulse Width Low for Chip Erase	5.0	10.0	15.0	ms
t_{WLWH_PFB}	$\overline{WR}$ Pulse Width Low for Programming the Fuse Bits	1.0	1.5	1.8	ms

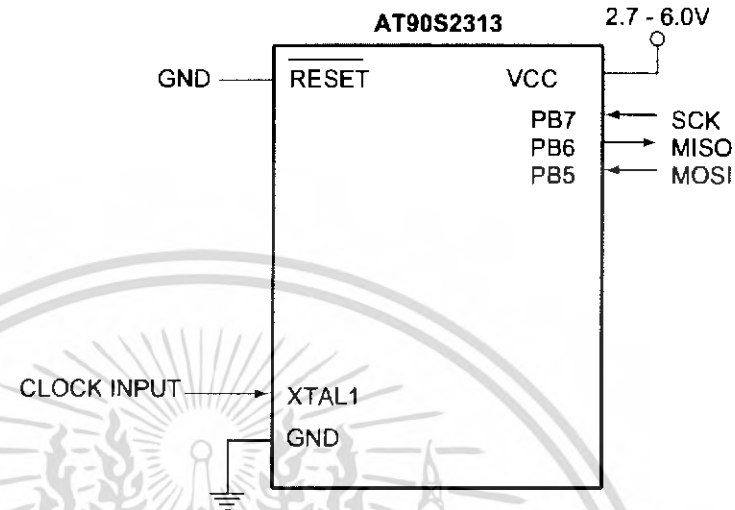
Notes: 1. Use t_{WLWH_CE} for chip erase and t_{WLWH_PFB} for programming the Fuse bits.
2. If t_{WLWH} is held longer than t_{WLRH}, no RDY/$\overline{BSY}$ pulse will be seen.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Downloading

Both the program and data memory arrays can be programmed using the serial SPI bus while RESET is pulled to GND. The serial interface consists of pins SCK, MOSI (input) and MISO (output). See Figure 53. After RESET is set low, the Programming Enable instruction needs to be executed first before program/erase instructions can be executed.

Figure 53. Serial Programming and Verify



For the EEPROM, an auto-erase cycle is provided within the self-timed write instruction and there is no need to first execute the Chip Erase instruction. The Chip Erase instruction turns the content of every memory location in both the program and EEPROM arrays into $FF.

The program and EEPROM memory arrays have separate address spaces: $0000 to $03FF for program Flash memory and $000 to $07F for EEPROM data memory.

Either an external clock is supplied at pin XTAL1 or a crystal needs to be connected across pins XTAL1 and XTAL2. The minimum low and high periods for the serial clock (SCK) input are defined as follows:

Low: > 2 XTAL1 clock cycle

High: > 2 XTAL1 clock cycles

Serial Programming Algorithm

When writing serial data to the AT90S2313, data is clocked on the rising edge of SCK.

When reading data from the AT90S2313, data is clocked on the falling edge of SCK. See Figure 54, Figure and Table 29 for timing details.

To program and verify the AT90S2313 in the Serial Programming Mode, the following sequence is recommended (See 4-byte instruction formats in Table 28):

1. Power-up sequence:
   Apply power between V_{CC} and GND while RESET and SCK are set to "0". If a crystal is not connected across pins XTAL1 and XTAL2, apply a clock signal to the XTAL1 pin. In some systems, the programmer cannot guarantee that SCK is held low during power-up. In this case, RESET must be given a positive pulse of at least two XTAL1 cycles duration after SCK has been set to "0".

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Wait for at least 20 ms and enable serial programming by sending the Programming Enable serial instruction to the MOSI (PB5) pin.
3. The serial programming instructions will not work if the communication is out of synchronization. When in sync, the second byte ($53) will echo back when issuing the third byte of the Programming Enable instruction. Whether the echo is correct or not, all four bytes of the instruction must be transmitted. If the $53 did not echo back, give SCK a positive pulse and issue a new Programming Enable instruction. If the $53 is not seen within 32 attempts, there is no functional device connected.
4. If a Chip Erase is performed (must be done to erase the Flash), wait t_{WD_ERASE} after the instruction, give $\overline{\text{RESET}}$ a positive pulse, and start over from step 2. See Table 30 for t_{WD_ERASE} value.
5. The Flash or EEPROM array is programmed one byte at a time by supplying the address and data together with the appropriate Write instruction. An EEPROM memory location is first automatically erased before new data is written. Use Data Polling to detect when the next byte in the Flash or EEPROM can be written. If polling is not used, wait t_{WD_PROG} before transmitting the next instruction. See Table 31 for t_{WD_PROG} value. In an erased device, no $FFs in the data file(s) need to be programmed.
6. Any memory location can be verified by using the Read instruction that returns the content at the selected address at the serial output MISO (PB6) pin.
7. At the end of the programming session, $\overline{\text{RESET}}$ can be set high to commence normal operation.
8. Power-off sequence (if needed):
   Set XTAL1 to "0" (if a crystal is not used).
   Set $\overline{\text{RESET}}$ to "1".
   Turn V_{CC} power off.

Data Polling EEPROM

When a byte is being programmed into the EEPROM, reading the address location being programmed will give the value P1 until the auto-erase is finished, and then the value P2. See Table 27 for P1 and P2 values.

At the time the device is ready for a new EEPROM byte, the programmed value will read correctly. This is used to determine when the next byte can be written. This will not work for the values P1 and P2, so when programming these values, the user will have to wait for at least the prescribed time t_{WD_PROG} before programming the next byte. See Table 30 for t_{WD_PROG} value. As a chip-erased device contains $FF in all locations, programming of addresses that are meant to contain $FF can be skipped. This does not apply if the EEPROM is reprogrammed without first chip-erasing the device.

Table 27. Read Back Value during EEPROM Polling

Part	P1	P2
AT90S2313	$80	$7F

Data Polling Flash

When a byte is being programmed into the Flash, reading the address location being programmed will give the value $7F. At the time the device is ready for a new byte, the programmed value will read correctly. This is used to determine when the next byte can be written. This will not work for the value $7F, so when programming this value, the user will have to wait for at least t_{WD_PROG} before programming the next byte. As a chip-erased device contains $FF in all locations, programming of addresses that are meant to contain $FF can be skipped.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 54. Serial Programming Waveforms

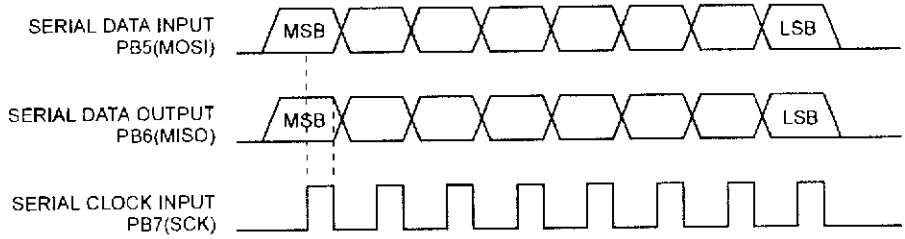


Table 28. Serial Programming Instruction Set

Instruction	Instruction Format				Operation
	Byte 1	Byte 2	Byte 3	Byte4	
Programming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx	Enable serial programming while $\overline{\text{RESET}}$ is low.
Chip Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip erase Flash and EEPROM memory arrays.
Read Program Memory	0010 **H**000	xxxx xx**aa**	**bbbb bbbb**	**oooo oooo**	Read **H** (high or low) data **o** from program memory at word address **a**:**b**.
Write Program Memory	0100 **H**000	xxxx xx**aa**	**bbbb bbbb**	**iiii iiii**	Write **H** (high or low) data **i** to program memory at word address **a**:**b**.
Read EEPROM Memory	1010 0000	xxxx xxxx	x**bbb bbbb**	**oooo oooo**	Read data **o** from EEPROM memory at address **b**.
Write EEPROM Memory	1100 0000	xxxx xxxx	x**bbb bbbb**	**iiii iiii**	Write data **i** to EEPROM memory at address **b**.
Write Lock Bits	1010 1100	111x x***21***x	xxxx xxxx	xxxx xxxx	Write Lock bits. Set bits ***1,2*** = "0" to program Lock bits.
Read Signature Bytes	0011 0000	xxxx xxxx	xxxx xx**bb**	**oooo oooo**	Read signature byte **o** at address **b**.(1)

Note: **a** = address high bits
b = address low bits
H = 0 – Low byte, 1 – High Byte
o = data out
i = data in
x = don't care
1 = Lock bit 1
2 = Lock bit 2

Note: 1. The signature bytes are not readable in lock mode 3, i.e. both Lock bits programmed.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Programming Characteristics

Figure 55. Serial Programming Timing

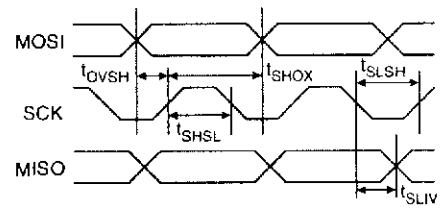


Table 29. Serial Programming Characteristics, T_A = -40°C to 85°C, V_{CC} = 2.7 - 6.0V (unless otherwise noted)

Symbol	Parameter	Min	Typ	Max	Units
$1/t_{CLCL}$	Oscillator Frequency (V_{CC} = 2.7 - 6.0V)	0		4.0	MHz
t_{CLCL}	Oscillator Period (V_{CC} = 2.7 - 6.0V)	250.0			ns
$1/t_{CLCL}$	Oscillator Frequency (V_{CC} = 4.0 - 6.0V)	0		10.0	MHz
t_{CLCL}	Oscillator Period (V_{CC} = 4.0 - 6.0V)	100.0			ns
t_{SHSL}	SCK Pulse Width High	2.0 t_{CLCL}			ns
t_{SLSH}	SCK Pulse Width Low	2.0 t_{CLCL}			ns
t_{OVSH}	MOSI Setup to SCK High	t_{CLCL}			ns
t_{SHOX}	MOSI Hold after SCK High	2.0 t_{CLCL}			ns
t_{SLIV}	SCK Low to MISO Valid	10.0	16.0	32.0	ns

Table 30. Minimum Wait Delay after the Chip Erase Instruction

Symbol	3.2V	3.6V	4.0V	5.0V
t_{WD_ERASE}	18 ms	14 ms	12 ms	8 ms

Table 31. Minimum Wait Delay after Writing a Flash or EEPROM Location

Symbol	3.2V	3.6V	4.0V	5.0V
t_{WD_PROG}	9 ms	7 ms	6 ms	4 ms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin Except $\overline{\text{RESET}}$ with Respect to Ground	-1.0V to V_{CC}+0.5V
Voltage on $\overline{\text{RESET}}$ with Respect to Ground	-1.0V to +13.0V
Maximum Operating Voltage	6.6V
DC Current per I/O Pin	40.0 mA
DC Current V_{CC} and GND Pins	200.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

DC Characteristics

T_A = -40°C to 85°C, V_{CC} = 2.7V to 6.0V (unless otherwise noted)

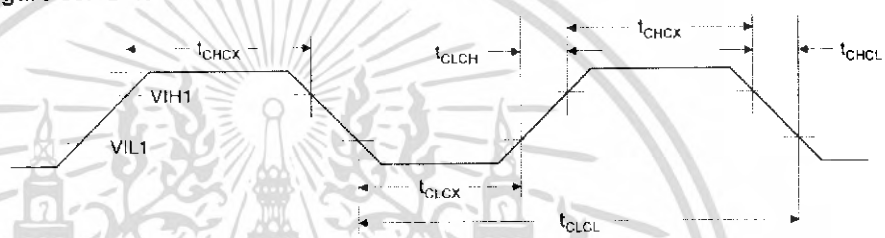
Symbol	Parameter	Condition	Min	Typ	Max	Units
V_{IL}	Input Low Voltage	(Except XTAL1)	-0.5		0.3 V_{CC}[1]	V
V_{IL1}	Input Low Voltage	(XTAL1)	-0.5		0.3 V_{CC}[1]	V
V_{IH}	Input High Voltage	(Except XTAL1, $\overline{\text{RESET}}$)	0.6 V_{CC}[2]		V_{CC} + 0.5	V
V_{IH1}	Input High Voltage	(XTAL1)	0.7 V_{CC}[2]		V_{CC} + 0.5	V
V_{IH2}	Input High Voltage	($\overline{\text{RESET}}$)	0.85 V_{CC}[2]		V_{CC} + 0.5	V
V_{OL}	Output Low Voltage[3] (Ports B, D)	I_{OL} = 20 mA, V_{CC} = 5V I_{OL} = 10 mA, V_{CC} = 3V			0.6 0.5	V V
V_{OH}	Output High Voltage[4] (Ports B, D)	I_{OH} = -3 mA, V_{CC} = 5V I_{OH} = -1.5 mA, V_{CC} = 3V	4.3 2.3			V V
I_{IL}	Input Leakage Current I/O pin	V_{CC} = 6V, pin low (absolute value)			1.5	µA
I_{IH}	Input Leakage Current I/O pin	V_{CC} = 6V, pin high (absolute value)			980.0	nA
RRST	Reset Pull-up Resistor		100.0		500.0	kΩ
$R_{I/O}$	I/O Pin Pull-up Resistor		35.0		120.0	kΩ
I_{CC}	Power Supply Current	Active Mode, V_{CC} = 3V, 4 MHz			3.0	mA
		Idle Mode V_{CC} = 3V, 4 MHz			1.0	mA
I_{CC}	Power-down Mode[5]	WDT enabled, V_{CC} = 3V		9.0	15.0	µA
		WDT disabled, V_{CC} = 3V		<1.0	2.0	µA
V_{ACIO}	Analog Comparator Input Offset Voltage	V_{CC} = 5V V_{in} = V_{CC}/2			40.0	mV
I_{ACLK}	Analog Comparator Input Leakage Current	V_{CC} = 5V V_{in} = V_{CC}/2	-50.0		50.0	nA
t_{ACPD}	Analog Comparator Propagation Delay	V_{CC} = 2.7V V_{CC} = 4.0V		750.0 500.0		ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Notes:
1. "Max" means the highest value where the pin is guaranteed to be read as low.
2. "Min" means the lowest value where the pin is guaranteed to be read as high.
3. Although each I/O port can sink more than the test conditions (20 mA at V_{CC} = 5V, 10 mA at V_{CC} = 3V) under steady state conditions (non-transient), the following must be observed:
   1] The sum of all I_{OL}, for all ports, should not exceed 200 mA
   2] The sum of all II_{OL}, for port D0 - D5 and XTAL2 should not exceed 100 mA.
   3] The sum of all I_{OL}, for ports B0 - B7 and D6 should not exceed 100 mA.
   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test condition.
4. Although each I/O port can source more than the test conditions (3 mA at V_{CC} = 5V, 1.5 mA at V_{CC} = 3V) under steady state conditions (non-transient), the following must be observed:
   1] The sum of all I_{OH}, for all ports, should not exceed 200 mA
   2] The sum of all I_{OH}, for port D0 - D5 and XTAL2 should not exceed 100 mA.
   3] The sum of all I_{OH}, for ports B0 - B7 and D6 should not exceed 100 mA.
   If I_{OH} exceeds the test condition, V_{OH} may exceed the related specification. Pins are not guaranteed to source current greater than the listed test condition.
5. Minimum V_{CC} for power-down is 2V.

External Clock Drive Waveforms

Figure 56. External Clock



External Clock Drive

Symbol	Parameter	V_{CC} = 2.7V to 6.0V		V_{CC} = 4.0V to 6.0V		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	4	0	10.0	MHz
t_{CLCL}	Clock Period	250.0		100.0		ns
t_{CHCX}	High Time	100.0		40.0		ns
t_{CLCX}	Low Time	100.0		40.0		ns
t_{CLCH}	Rise Time		1.6		0.5	µs
t_{CHCL}	Fall Time		1.6		0.5	µs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Characteristics

The following charts show typical behavior. These figures are not tested during manufacturing. All current consumption measurements are performed with all I/O pins configured as inputs and with internal pull-ups enabled. A sine wave generator with rail-to-rail output is used as clock source.

The power consumption in Power-Down Mode is independent of clock selection.

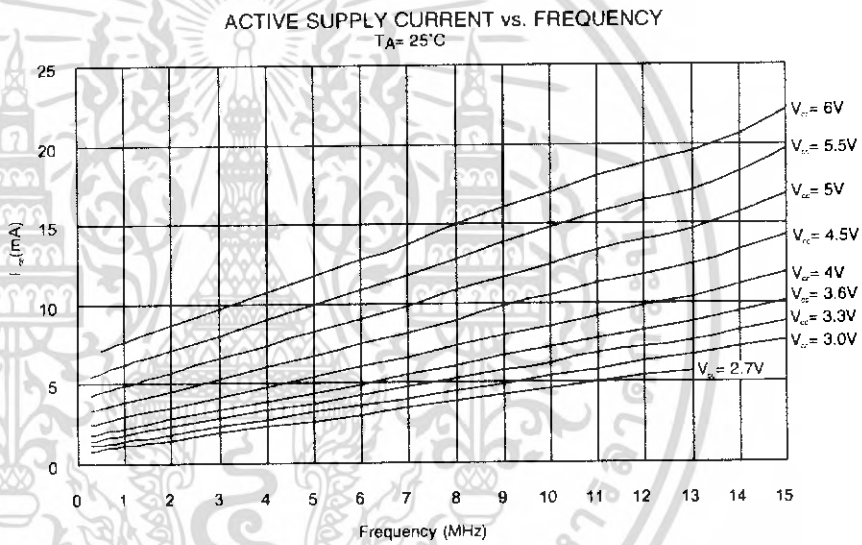
The current consumption is a function of several factors such as: operating voltage, operating frequency, loading of I/O pins, switching rate of I/O pins, code executed and ambient temperature. The dominating factors are operating voltage and frequency.

The current drawn from capacitive loaded pins may be estimated (for one pin) as $C_L \cdot V_{CC} \cdot f$ where C_L = load capacitance, V_{CC} = operating voltage and f = average switching frequency of I/O pin.

The parts are characterized at frequencies higher than test limits. Parts are not guaranteed to function properly at frequencies higher than the ordering code indicates.

The difference between current consumption in Power-down Mode with Watchdog timer enabled and Power-down Mode with Watchdog timer disabled represents the differential current drawn by the Watchdog timer.

Figure 57. Active Supply Current vs. Frequency



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 58. Active Supply Current vs. V_{CC}

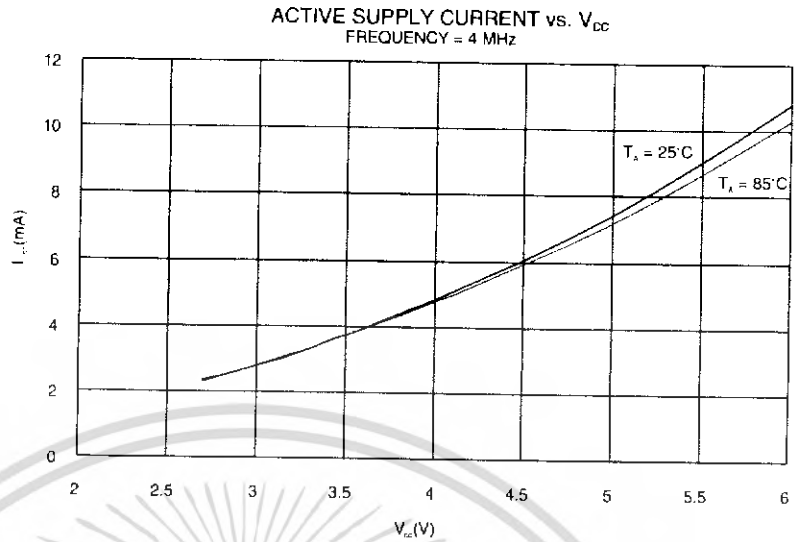
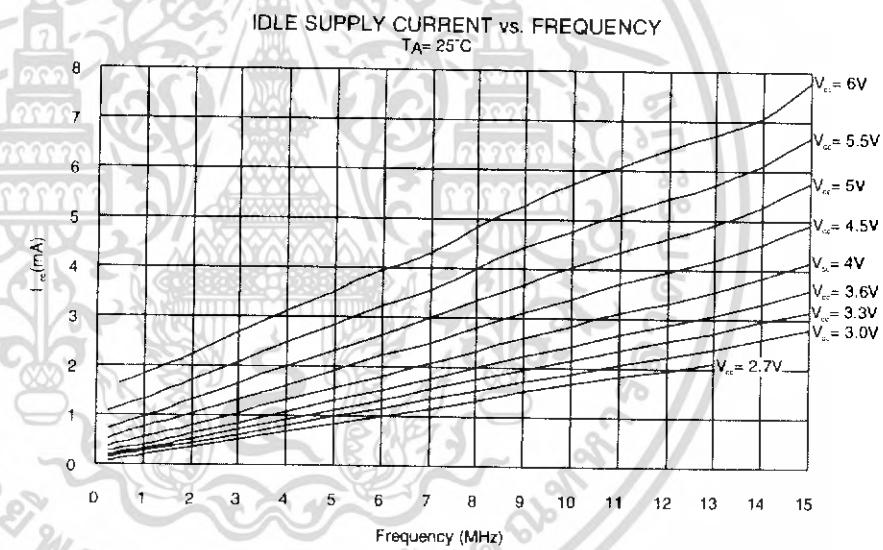


Figure 59. Idle Supply Current vs. Frequency



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 60. Idle Supply Current vs. V_{CC}

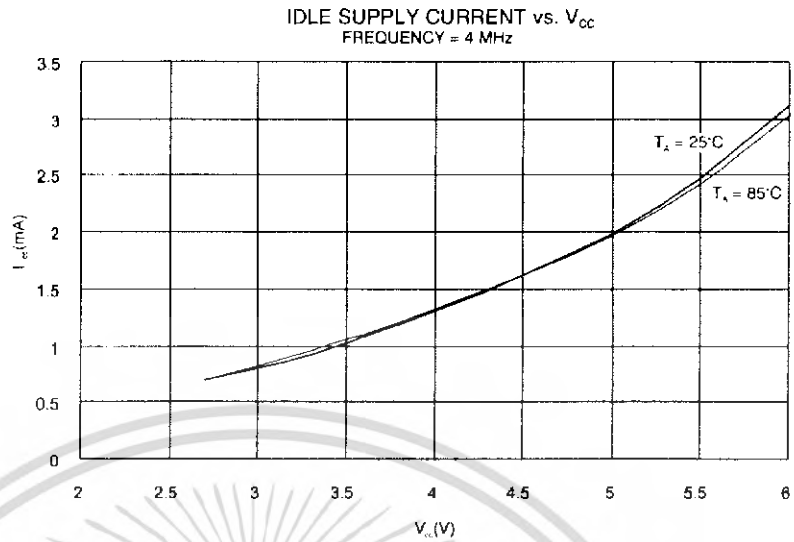
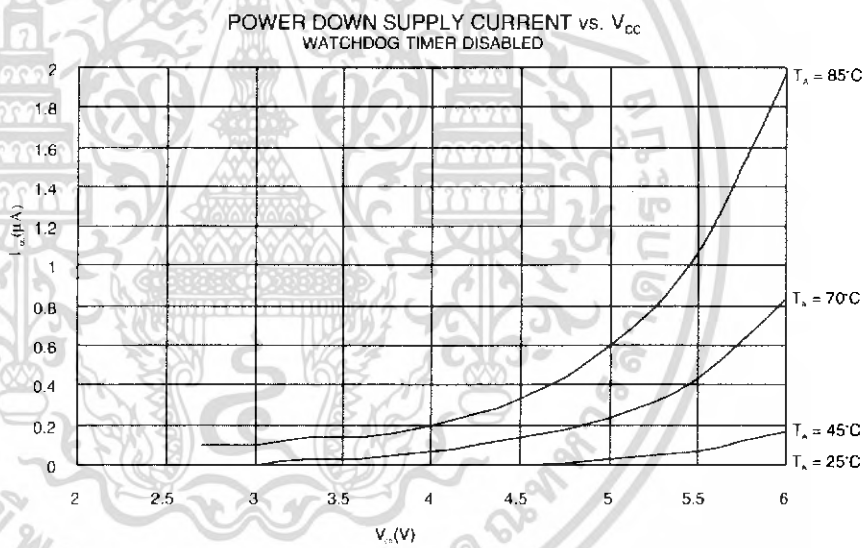


Figure 61. Power-down Supply Current vs. V_{CC}



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 62. Power-down Supply Current vs. V_{CC}

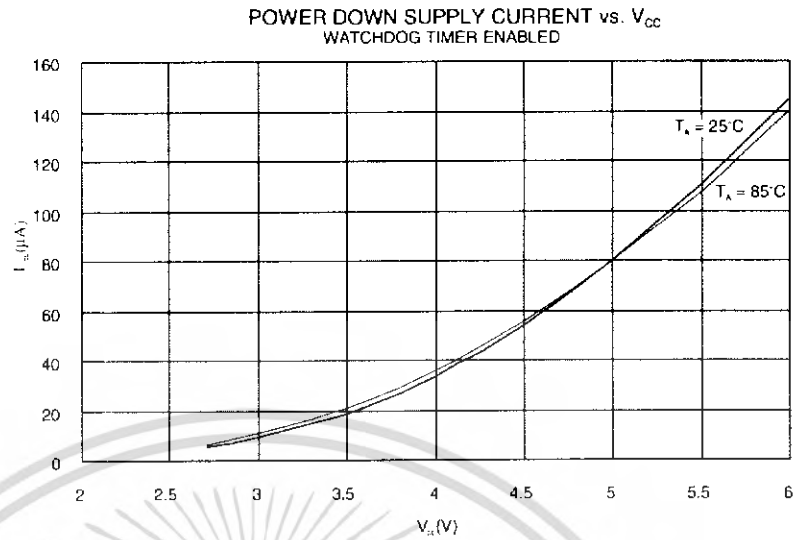
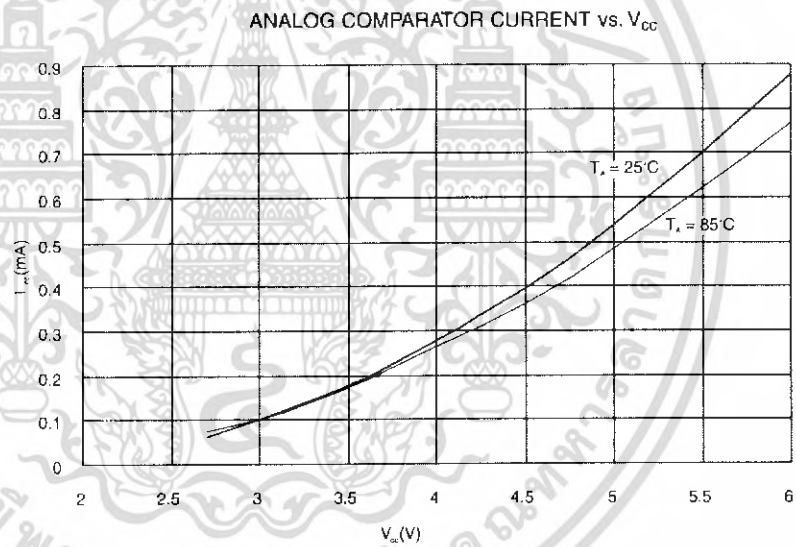


Figure 63. Analog Comparator Current vs. V_{CC}



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note: Analog Comparator offset voltage is measured as absolute offset.

Figure 64. Analog Comparator Offset Voltage vs. Common Mode Voltage

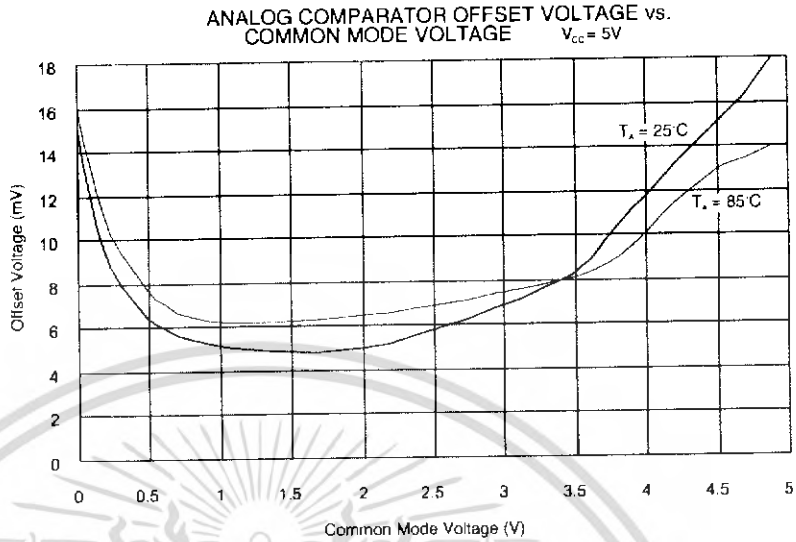
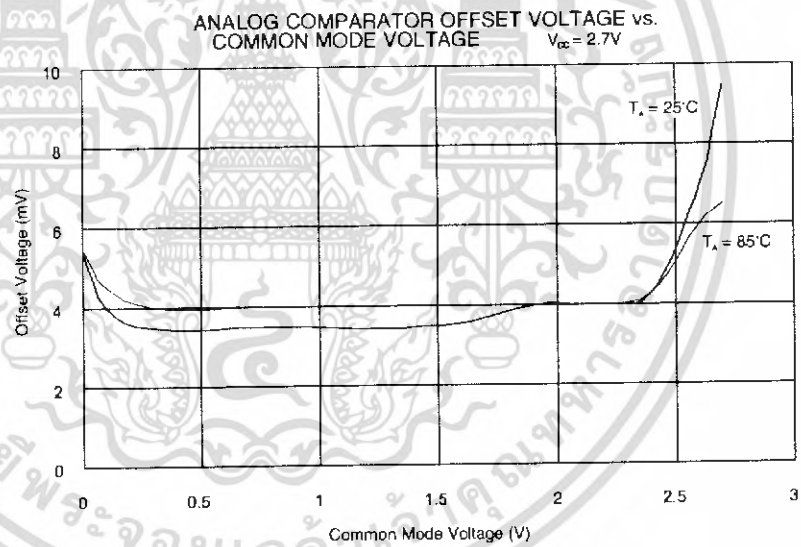


Figure 65. Analog Comparator Offset Voltage vs. Common Mode Voltage



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 66. Analog Comparator Input Leakage Current

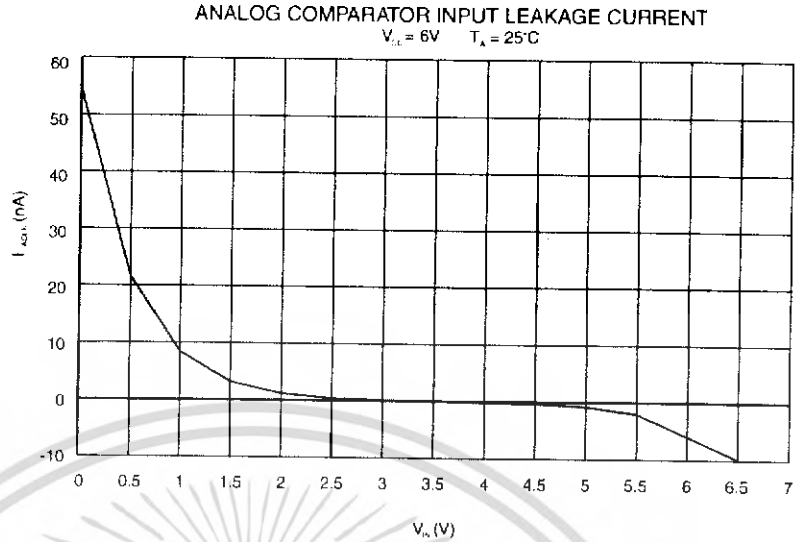
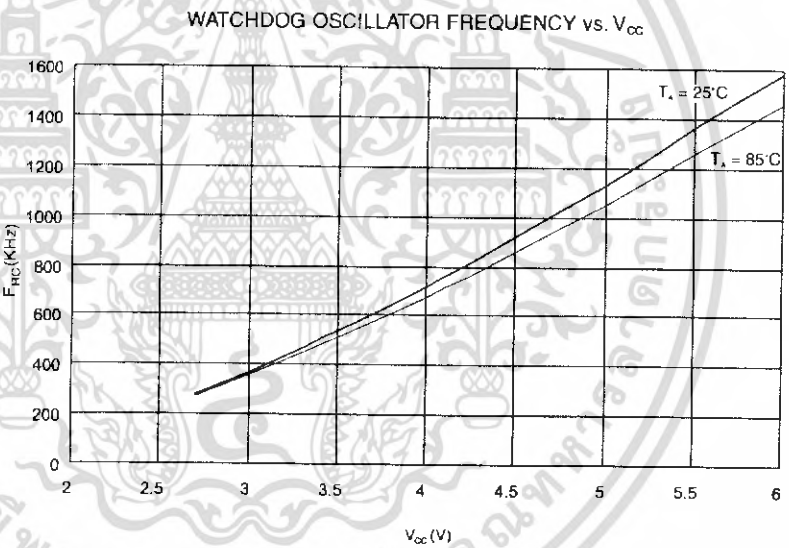


Figure 67. Watchdog Oscillator Frequency vs. V_{CC}



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note: Sink and source capabilities of I/O ports are measured on one pin at a time.

Figure 68. Pull-up Resistor Current vs. Input Voltage

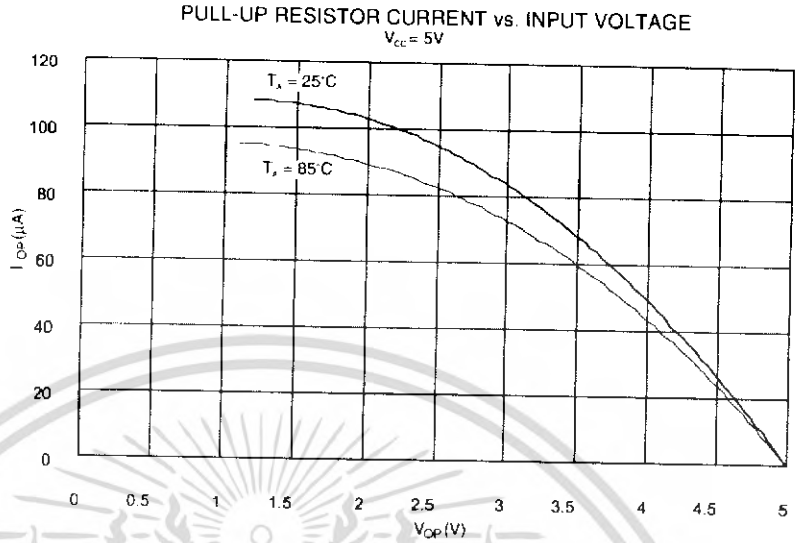
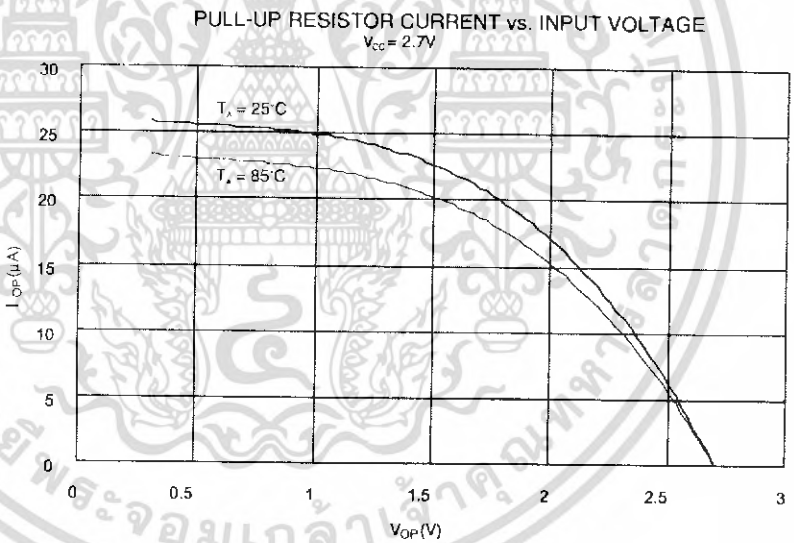


Figure 69. Pull-up Resistor Current vs. Input Voltage



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 70. I/O Pin Sink Current vs. Output Voltage

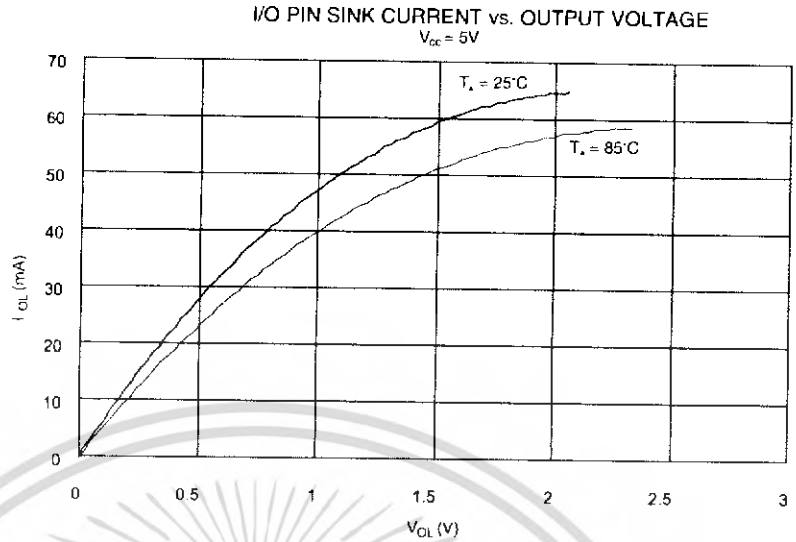
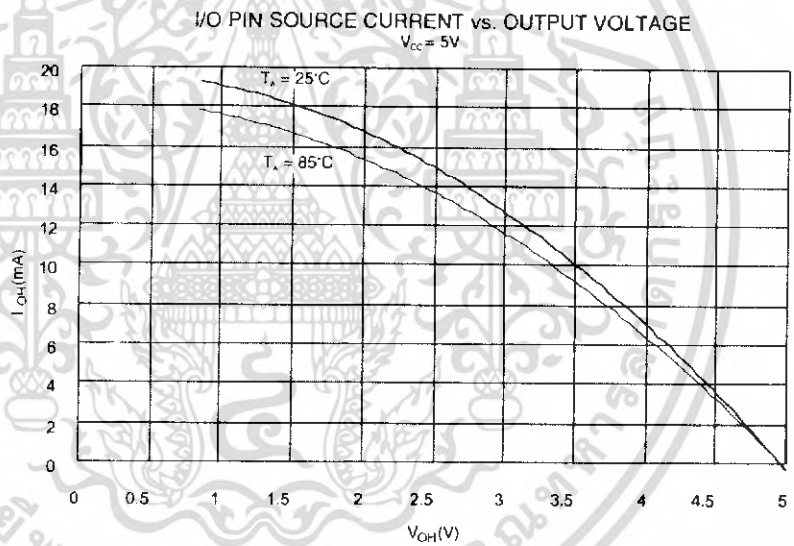


Figure 71. I/O Pin Source Current vs. Output Voltage



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 72. I/O Pin Sink Current vs. Output Voltage

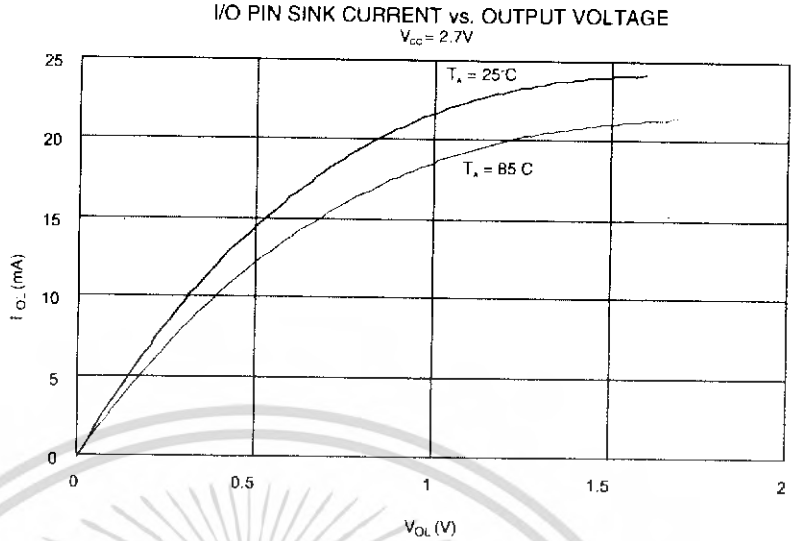
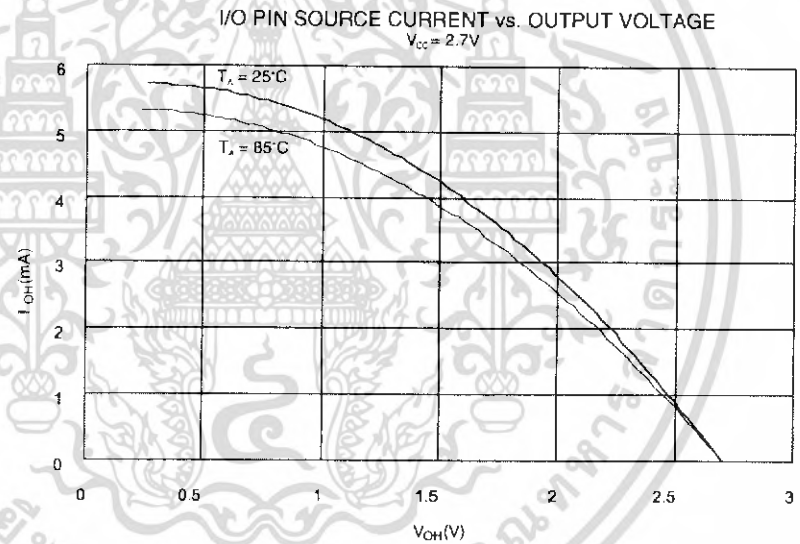


Figure 73. I/O Pin Source Current vs. Output Voltage



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 74. I/O Pin Input Threshold Voltage vs. V_{CC}

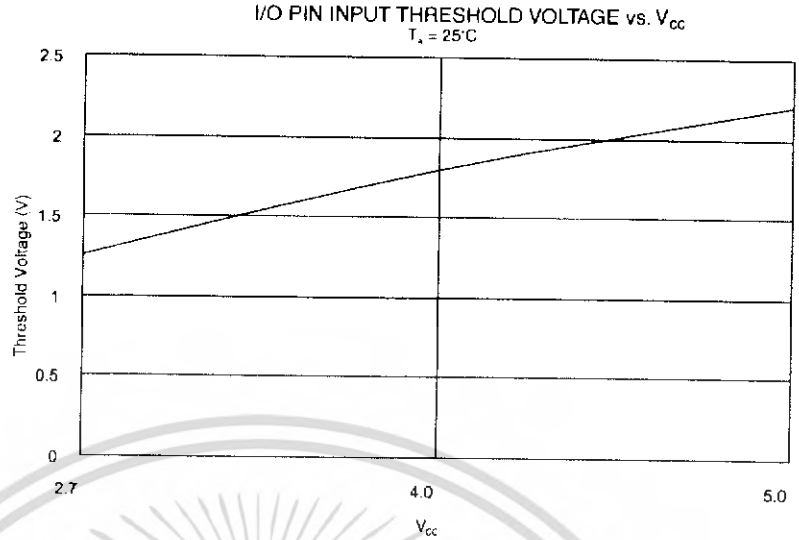
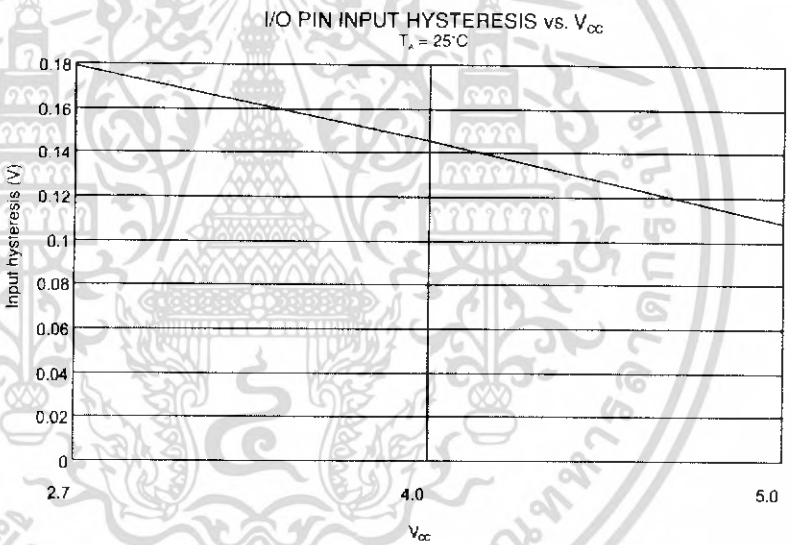


Figure 75. I/O Pin Input Hysteresis vs. V_{CC}



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register Summary

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
$3F ($5F)	SREG	I	T	H	S	V	N	Z	C	page 17
$3E ($5E)	Reserved									
$3D ($5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	page 18
$3C ($5C)	Reserved									
$3B ($5B)	GIMSK	INT1	INT0	-	-	-	-	-	-	page 23
$3A ($5A)	GIFR	INTF1	INTF0							page 23
$39 ($59)	TIMSK	TOIE1	OCIE1A	-	-	TICIE1	-	TOIE0	-	page 24
$38 ($58)	TIFR	TOV1	OCF1A	-	-	ICF1	-	TOV0	-	page 24
$37 ($57)	Reserved									
$36 ($56)	Reserved									
$35 ($55)	MCUCR	-	-	SE	SM	ISC11	ISC10	ISC01	ISC00	page 26
$34 ($54)	Reserved									
$33 ($53)	TCCR0	-	-	-	-	-	CS02	CS01	CS00	page 29
$32 ($52)	TCNT0	Timer/Counter0 (8 Bits)								page 30
$31 ($51)	Reserved									
$30 ($50)	Reserved									
$2F ($4F)	TCCR1A	COM1A1	COM1A0	-	-	-	-	PWM11	PWM10	page 31
$2E ($4E)	TCCR1B	ICNC1	ICES1	-	-	CTC1	CS12	CS11	CS10	page 32
$2D ($4D)	TCNT1H	Timer/Counter1 – Counter Register High Byte								page 33
$2C ($4C)	TCNT1L	Timer/Counter1 – Counter Register Low Byte								page 33
$2B ($4B)	OCR1AH	Timer/Counter1 – Compare Register High Byte								page 34
$2A ($4A)	OCR1AL	Timer/Counter1 – Compare Register Low Byte								page 34
$29 ($49)	Reserved									
$28 ($48)	Reserved									
$27 ($47)	Reserved									
$26 ($46)	Reserved									
$25 ($45)	ICR1H	Timer/Counter1 – Input Capture Register High Byte								page 35
$24 ($44)	ICR1L	Timer/Counter1 – Input Capture Register Low Byte								page 35
$23 ($43)	Reserved									
$22 ($42)	Reserved									
$21 ($41)	WDTCR	-	-	-	WDTOE	WDE	WDP2	WDP1	WDP0	page 38
$20 ($40)	Reserved									
$1F ($3F)	Reserved									
$1E ($3E)	EEAR	-	EEPROM Address Register							page 40
$1D ($3D)	EEDR	EEPROM Data Register								page 40
$1C ($3C)	EECR	-	-	-	-	-	EEMWE	EEWE	EERE	page 40
$1B ($3B)	Reserved									
$1A ($3A)	Reserved									
$19 ($39)	Reserved									
$18 ($38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	page 52
$17 ($37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	page 52
$16 ($36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	page 52
$15 ($35)	Reserved									
$14 ($34)	Reserved									
$13 ($33)	Reserved									
$12 ($32)	PORTD	-	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	page 58
$11 ($31)	DDRD	-	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	page 58
$10 ($30)	PIND	-	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	page 58
...	Reserved									
$0C ($2C)	UDR	UART I/O Data Register								page 46
$0B ($2B)	USR	RXC	TXC	UDRE	FE	OR	-	-	-	page 46
$0A ($2A)	UCR	RXCIE	TXCIE	UDRIE	RXEN	TXEN	CHR9	RXB8	TXB8	page 47
$09 ($29)	UBRR	UART Baud Rate Register								page 49
$08 ($28)	ACSR	ACD	-	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	page 50
...	Reserved									
$00 ($20)	Reserved									

Notes: 1. For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.

2. Some of the status flags are cleared by writing a logical "1" to them. Note that the CBI and SBI instructions will operate on all bits in the I/O register, writing a "1" back into any flag read as set, thus clearing the flag. The CBI and SBI instructions work with registers $00 to $1F only.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Instruction Set Summary

Mnemonic	Operands	Description	Operation	Flags	# Clocks
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add Two Registers	Rd ← Rd + Rr	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry Two Registers	Rd ← Rd + Rr + C	Z,C,N,V,H	1
ADIW	Rdl, K	Add Immediate to Word	Rdh:Rdl ← Rdh:Rdl + K	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract Two Registers	Rd ← Rd - Rr	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	Rd ← Rd - K	Z,C,N,V,H	1
SBIW	Rdl, K	Subtract Immediate from Word	Rdh:Rdl ← Rdh:Rdl - K	Z,C,N,V,S	2
SBC	Rd, Rr	Subtract with Carry Two Registers	Rd ← Rd - Rr - C	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Reg.	Rd ← Rd - K - C	Z,C,N,V,H	1
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 v Rr	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	Rd ← Rd v K	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	Rd ← Rd ⊕ Rr	Z,N,V	1
COM	Rd	One's Complement	Rd ← $FF - Rd	Z,C,N,V	1
NEG	Rd	Two's Complement	Rd ← $00 - Rd	Z,C,N,V,H	1
SBR	Rd, K	Set Bit(s) in Register	Rd ← Rd v K	Z,N,V	1
CBR	Rd, K	Clear Bit(s) in Register	Rd ← Rd • ($FF - 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 Minus	Rd ← Rd • Rd	Z,N,V	1
CLR	Rd	Clear Register	Rd ← Rd ⊕ Rd	Z,N,V	1
SER	Rd	Set Register	Rd ← $FF	None	1
BRANCH INSTRUCTIONS					
RJMP	k	Relative Jump	PC ← PC + k + 1	None	2
IJMP		Indirect Jump to (Z)	PC ← Z	None	2
RCALL	k	Relative Subroutine Call	PC ← PC + k + 1	None	3
ICALL		Indirect Call to (Z)	PC ← Z	None	3
RET		Subroutine Return	PC ← STACK	None	4
RETI		Interrupt Return	PC ← STACK	I	4
CPSE	Rd, Rr	Compare, Skip if Equal	if (Rd = Rr) PC ← PC + 2 or 3	None	1/2
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	if (Rr(b) = 0) PC ← PC + 2 or 3	None	1/2
SBRS	Rr, b	Skip if Bit in Register is Set	if (Rr(b) = 1) PC ← PC + 2 or 3	None	1/2
SBIC	P, b	Skip if Bit in I/O Register Cleared	if (P(b) = 0) PC ← PC + 2 or 3	None	1/2
SBIS	P, b	Skip if Bit in I/O Register is Set	if (P(b) = 1) PC ← PC + 2 or 3	None	1/2
BRBS	s, k	Branch if Status Flag Set	if (SREG(s) = 1) then PC ← PC + k + 1	None	1/2
BRBC	s, k	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
BRCS	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
BRMI	k	Branch if Minus	if (N = 1) then PC ← PC + k + 1	None	1/2
BRPL	k	Branch if Plus	if (N = 0) then PC ← PC + k + 1	None	1/2
BRGE	k	Branch if Greater or Equal, Signed	if (N ⊕ V = 0) then PC ← PC + k + 1	None	1/2
BRLT	k	Branch if Less than Zero, Signed	if (N ⊕ V = 1) then PC ← PC + k + 1	None	1/2
BRHS	k	Branch if Half-carry Flag Set	if (H = 1) then PC ← PC + k + 1	None	1/2
BRHC	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
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then PC ← PC + k + 1	None	1/2
BRID	k	Branch if Interrupt Disabled	if (I = 0) then PC ← PC + k + 1	None	1/2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Instruction Set Summary (Continued)

Mnemonic	Operands	Description	Operation	Flags	# Clocks
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move between Registers	Rd ← Rr	None	1
LDI	Rd, K	Load Immediate	Rd ← K	None	1
LD	Rd, X	Load Indirect	Rd ← (X)	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	Rd ← (X), X ← X + 1	None	2
LD	Rd, -X	Load Indirect and Pre-Dec.	X ← X - 1, Rd ← (X)	None	2
LD	Rd, Y	Load Indirect	Rd ← (Y)	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	Rd ← (Y), Y ← Y + 1	None	2
LD	Rd, -Y	Load Indirect and Pre-Dec.	Y ← Y - 1, Rd ← (Y)	None	2
LDD	Rd, Y+q	Load Indirect with Displacement	Rd ← (Y + q)	None	2
LD	Rd, Z	Load Indirect	Rd ← (Z)	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	Rd ← (Z), Z ← Z+1	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	Z ← Z - 1, Rd ← (Z)	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	Rd ← (Z + q)	None	2
LDS	Rd, k	Load Direct from SRAM	Rd ← (k)	None	2
ST	X, Rr	Store Indirect	(X) ← Rr	None	2
ST	X+, Rr	Store Indirect and Post Inc.	(X) ← Rr, X ← X + 1	None	2
ST	-X, Rr	Store Indirect and Pre-Dec.	X ← X - 1, (X) ← Rr	None	2
ST	Y, Rr	Store Indirect	(Y) ← Rr	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	(Y) ← Rr, Y ← Y + 1	None	2
ST	-Y, Rr	Store Indirect and Pre-Dec.	Y ← Y - 1, (Y) ← Rr	None	2
STD	Y+q, Rr	Store Indirect with Displacement	(Y + q) ← Rr	None	2
ST	Z, Rr	Store Indirect	(Z) ← Rr	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	(Z) ← Rr, Z ← Z + 1	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	Z ← Z - 1, (Z) ← Rr	None	2
STD	Z+q, Rr	Store Indirect with Displacement	(Z + q) ← Rr	None	2
STS	k, Rr	Store Direct to SRAM	(k) ← Rr	None	2
LPM		Load Program Memory	R0 ← (Z)	None	3
IN	Rd, P	In Port	Rd ← P	None	1
OUT	P, Rr	Out Port	P ← Rr	None	1
PUSH	Rr	Push Register on Stack	STACK ← Rr	None	2
POP	Rd	Pop Register from Stack	Rd ← STACK	None	2
BIT AND BIT-TEST INSTRUCTIONS					
SBI	P, b	Set Bit in I/O Register	I/O(P,b) ← 1	None	2
CBI	P, b	Clear Bit in I/O Register	I/O(P,b) ← 0	None	2
LSL	Rd	Logical Shift Left	Rd(n+1) ← Rd(n), Rd(0) ← 0	Z,C,N,V	1
LSR	Rd	Logical Shift Right	Rd(n) ← Rd(n+1), Rd(7) ← 0	Z,C,N,V	1
ROL	Rd	Rotate Left through Carry	Rd(0) ← C, Rd(n+1) ← Rd(n), C ← Rd(7)	Z,C,N,V	1
ROR	Rd	Rotate Right through Carry	Rd(7) ← C, Rd(n) ← Rd(n+1), C ← Rd(0)	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	Rd(n) ← Rd(n+1), n = 0..6	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	Rd(3..0) ← Rd(7..4), Rd(7..4) ← Rd(3..0)	None	1
BSET	s	Flag Set	SREG(s) ← 1	SREG(s)	1
BCLR	s	Flag Clear	SREG(s) ← 0	SREG(s)	1
BST	Rr, b	Bit Store from Register to T	T ← Rr(b)	T	1
BLD	Rd, b	Bit Load from T to Register	Rd(b) ← T	None	1
SEC		Set Carry	C ← 1	C	1
CLC		Clear Carry	C ← 0	C	1
SEN		Set Negative Flag	N ← 1	N	1
CLN		Clear Negative Flag	N ← 0	N	1
SEZ		Set Zero Flag	Z ← 1	Z	1
CLZ		Clear Zero Flag	Z ← 0	Z	1
SEI		Global Interrupt Enable	I ← 1	I	1
CLI		Global Interrupt Disable	I ← 0	I	1
SES		Set Signed Test Flag	S ← 1	S	1
CLS		Clear Signed Test Flag	S ← 0	S	1
SEV		Set Two's Complement Overflow	V ← 1	V	1
CLV		Clear Two's Complement Overflow	V ← 0	V	1
SET		Set T in SREG	T ← 1	T	1
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
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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

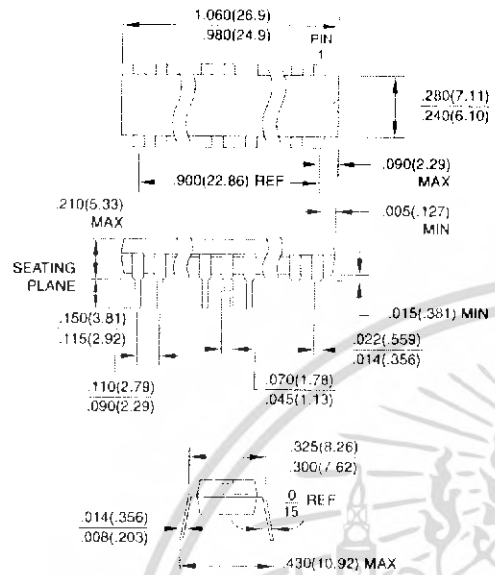
Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
4	2.7 - 6.0V	AT90S2313-4PC AT90S2313-4SC	20P3 20S	Commercial (0°C to 70°C)
		AT90S2313-4PI AT90S2313-4SI	20P3 20S	Industrial (-40°C to 85°C)
10	4.0 - 6.0V	AT90S2313-10PC AT90S2313-10SC	20P3 20S	Commercial (0°C to 70°C)
		AT90S2313-10PI AT90S2313-10SI	20P3 20S	Industrial (-40°C to 85°C)

Package Type	
20P3	20-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
20S	20-lead, 0.300" Wide, Plastic Gull Wing Small Outline (SOIC)

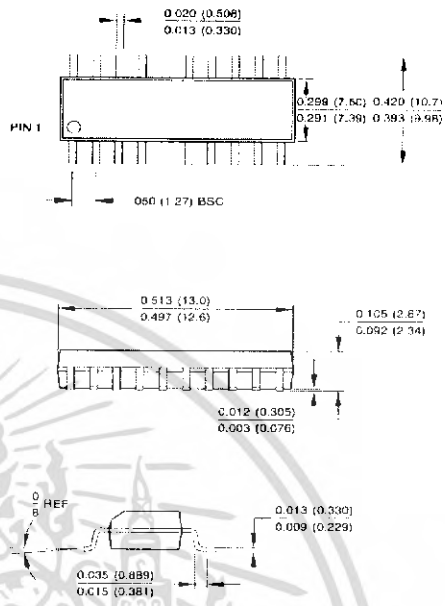
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Packaging Information

20P3, 20-lead, 0.300" Wide,
Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-001 BA



20S, 20-lead, 0.300" Wide,
Plastic Gull Wing Small Outline (SOIC)
Dimensions in Inches and (Millimeters)



0839G–08/01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Atmel Headquarters

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

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

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

Japan
Atmel Japan K.K.
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 Product Operations

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

Atmel Grenoble
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
TEL (33) 4-7658-3000
FAX (33) 4-7658-3480

Atmel Heilbronn
Theresienstrasse 2
POB 3535
D-74025 Heilbronn, Germany
TEL (49) 71 31 67 25 94
FAX (49) 71 31 67 24 23

Atmel Nantes
La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
TEL (33) 0 2 40 18 18 18
FAX (33) 0 2 40 18 19 60

Atmel Rousset
Zone Industrielle
13106 Rousset Cedex, France
TEL (33) 4-4253-6000
FAX (33) 4-4253-6001

Atmel Smart Card ICs
Scottish Enterprise Technology Park
East Kilbride, Scotland G75 0QR
TEL (44) 1355-357-000
FAX (44) 1355-242-743

Fax-on-Demand
North America:
1-(800) 292-8635
International:
1-(408) 441-0732

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

BBS
1-(408) 436-4309

© Atmel Corporation 2001.
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.

Marks bearing ® and/or ™ are registered trademarks and trademarks of Atmel Corporation.

Terms and product names in this document may be trademarks of others.

Printed on recycled paper.

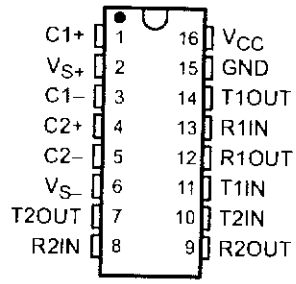
0839G–08/01/xM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

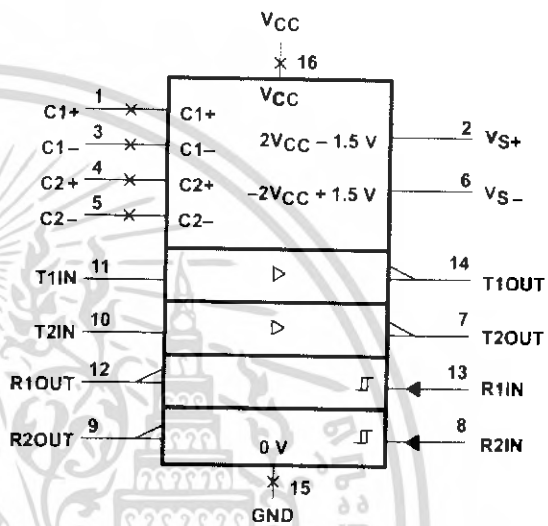
SLLS047G – FEBRUARY 1989 – REVISED AUGUST 1998

- Operates With Single 5-V Power Supply
- LinBiCMOS™ Process Technology
- Two Drivers and Two Receivers
- ±30-V Input Levels
- Low Supply Current . . . 8 mA Typical
- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28
- Designed to be Interchangeable With Maxim MAX232
- Applications
  - TIA/EIA-232-F
  - Battery-Powered Systems
  - Terminals
  - Modems
  - Computers
- ESD Protection Exceeds 2000 V Per MIL-STD-883, Method 3015
- Package Options Include Plastic Small-Outline (D, DW) Packages and Standard Plastic (N) DIPs

D, DW, OR N PACKAGE
(TOP VIEW)



logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

description

The MAX232 device is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

The MAX232 is characterized for operation from 0°C to 70°C. The MAX232I is characterized for operation from –40°C to 85°C.

AVAILABLE OPTIONS

T_A	PACKAGED DEVICES		
	SMALL OUTLINE (D)	SMALL OUTLINE (DW)	PLASTIC DIP (N)
0°C to 70°C	MAX232D‡	MAX232DW‡	MAX232N
–40°C to 85°C	MAX232ID‡	MAX232IDW‡	MAX232IN

‡ This device is available taped and reeled by adding an R to the part number (i.e., MAX232DR).

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

LinASIC and LinBiCMOS are trademarks of Texas Instruments Incorporated.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1998, Texas Instruments Incorporated

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Input supply voltage range, V_{CC} (see Note 1) . . . . . −0.3 V to 6 V
Positive output supply voltage range, V_{S+} . . . . . V_{CC} − 0.3 V to 15 V
Negative output supply voltage range, V_{S-} . . . . . −0.3 V to −15 V
Input voltage range, V_I: Driver . . . . . −0.3 V to V_{CC} + 0.3 V
Receiver . . . . . ±30 V
Output voltage range, V_O: T1OUT, T2OUT . . . . . V_{S-} −0.3 V to V_{S+} + 0.3 V
R1OUT, R2OUT . . . . . −0.3 V to V_{CC} + 0.3 V
Short-circuit duration: T1OUT, T2OUT . . . . . Unlimited
Package thermal impedance, θ_{JA} (see Note 2): D package . . . . . 113°C/W
DW package . . . . . 105°C/W
N package . . . . . 78°C/W
Storage temperature range, T_{stg} . . . . . −65°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds . . . . . 260°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: All voltage values are with respect to network ground terminal.

2. The package thermal impedance is calculated in accordance with JESD 51, except for through-hole packages, which use a trace length of zero.

recommended operating conditions

		MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}		4.5	5	5.5	V
High-level input voltage, V_{IH} (T1IN,T2IN)		2			V
Low-level input voltage, V_{IL} (T1IN, T2IN)				0.8	V
Receiver input voltage, R1IN, R2IN				±30	V
Operating free-air temperature, T_A	MAX232	0		70	°C
	MAX232I	−40		85	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (unless otherwise noted)

PARAMETER			TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage	T1OUT, T2OUT	R_L = 3 kΩ to GND		5	7		V
		R1OUT, R2OUT	I_{OH} = –1 mA		3.5			
V_{OL}	Low-level output voltage‡	T1OUT, T2OUT	R_L = 3 kΩ to GND			–7	–5	V
		R1OUT, R2OUT	I_{OL} = 3.2 mA				0.4	
V_{IT+}	Receiver positive-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V,	T_A = 25°C		1.7	2.4	V
V_{IT-}	Receiver negative-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V,	T_A = 25°C	0.8	1.2		V
V_{hys}	Input hysteresis voltage	R1IN, R2IN	V_{CC} = 5 V		0.2	0.5	1	V
r_i	Receiver input resistance	R1IN, R2IN	V_{CC} = 5,	T_A = 25°C	3	5	7	kΩ
r_o	Output resistance	T1OUT, T2OUT	V_{S+} = V_{S-} = 0,	V_O = ± 2 V	300			Ω
I_{OS}§	Short-circuit output current	T1OUT, T2OUT	V_{CC} = 5.5 V,	V_O = 0		±10		mA
I_{IS}	Short-circuit input current	T1IN, T2IN	V_I = 0				200	µA
I_{CC}	Supply current		V_{CC} = 5.5 V, T_A = 25°C	All outputs open,		8	10	mA

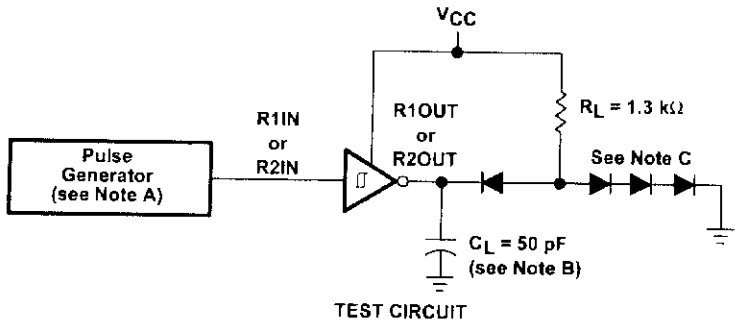
† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ The algebraic convention, in which the least positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.
§ Not more than one output should be shorted at a time.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

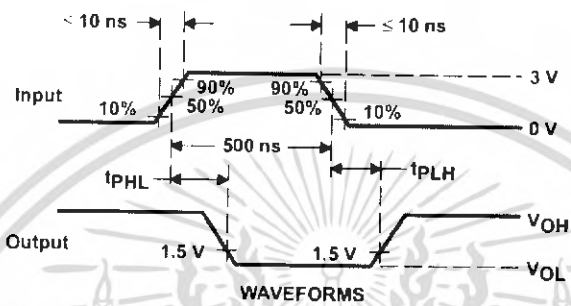
PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$t_{PLH(R)}$	Receiver propagation delay time, low- to high-level output	See Figure 1		500		ns
$t_{PHL(R)}$	Receiver propagation delay time, high- to low-level output	See Figure 1		500		ns
SR	Driver slew rate	R_L = 3 kΩ to 7 kΩ, See Figure 2			30	V/µs
SR(tr)	Driver transition region slew rate	See Figure 3		3		V/µs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PARAMETER MEASUREMENT INFORMATION



TEST CIRCUIT



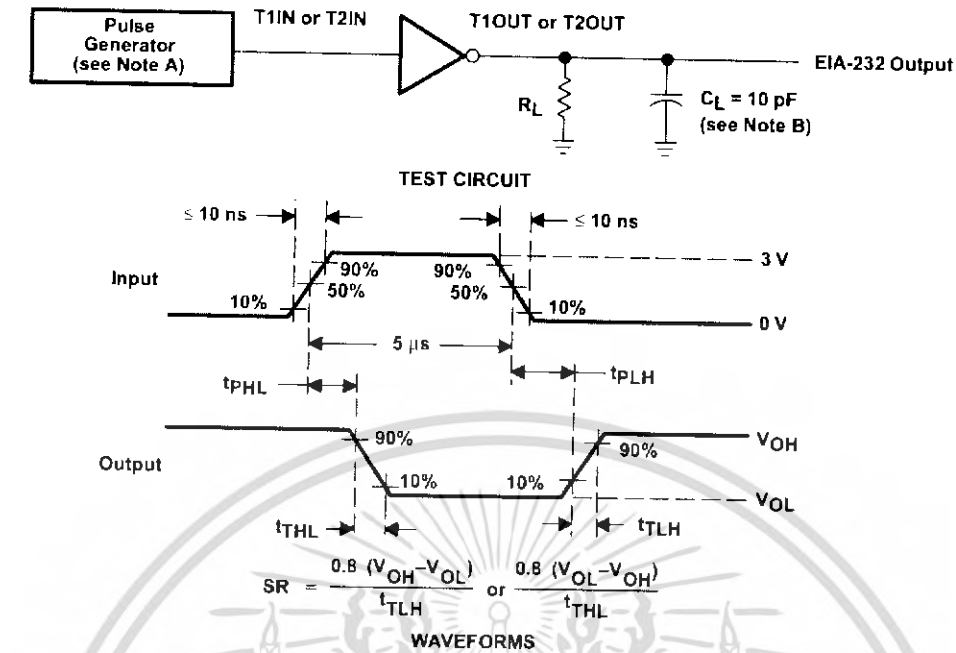
WAVEFORMS

NOTES: A. The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle ≤ 50%.
B. C_L includes probe and jig capacitance.
C. All diodes are 1N3064 or equivalent.

Figure 1. Receiver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements

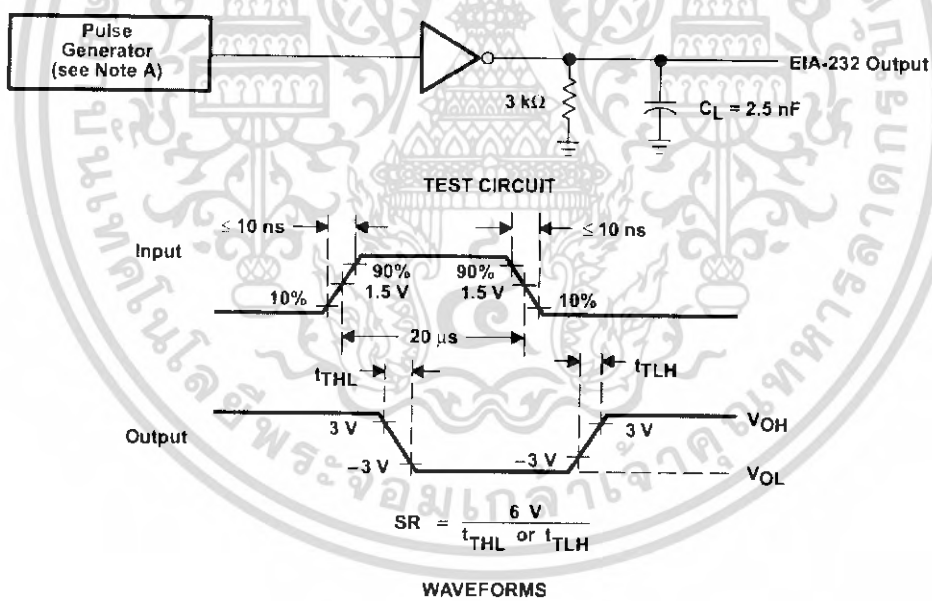
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PARAMETER MEASUREMENT INFORMATION



NOTES: A. The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle ≤ 50%.
B. C_L includes probe and jig capacitance.

Figure 2. Driver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements (5-µs input)



NOTE A: The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle ≤ 50%.

Figure 3. Test Circuit and Waveforms for t_{THL} and t_{TLH} Measurements (20-µs input)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPLICATION INFORMATION

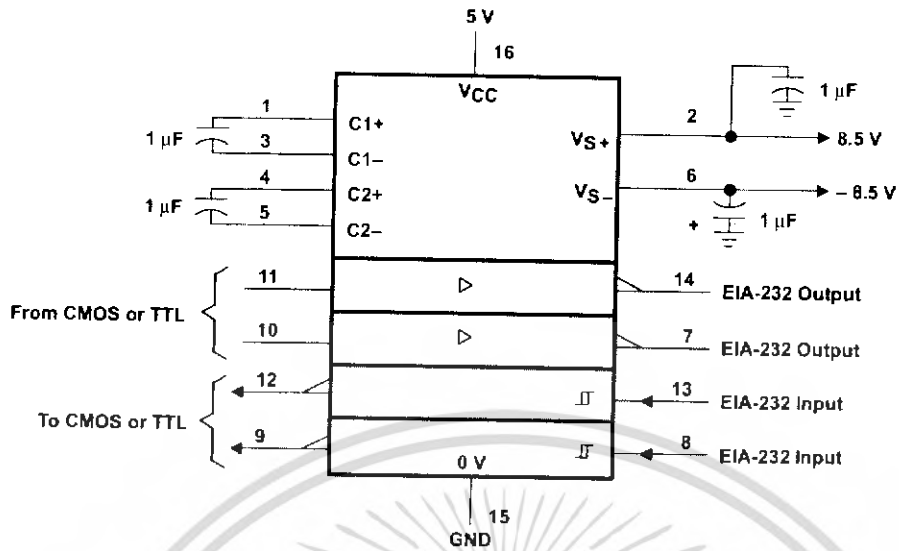


Figure 4. Typical Operating Circuit

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.

Copyright © 1998, Texas Instruments Incorporated

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

National Semiconductor

March 2004

ADC0820
8-Bit High Speed µP Compatible A/D Converter with Track/Hold Function

General Description

By using a half-flash conversion technique, the 8-bit ADC0820 CMOS A/D offers a 1.5 µs conversion time and dissipates only 75 mW of power. The half-flash technique consists of 32 comparators, a most significant 4-bit ADC and a least significant 4-bit ADC.

The input to the ADC0820 is tracked and held by the input sampling circuitry eliminating the need for an external sample-and-hold for signals moving at less than 100 mV/µs.

For ease of interface to microprocessors, the ADC0820 has been designed to appear as a memory location or I/O port without the need for external interfacing logic.

Key Specifications

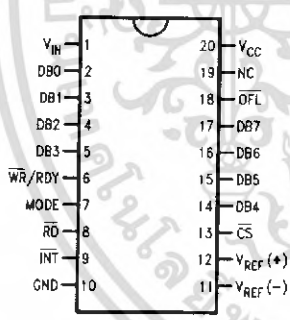
■ Resolution	8 Bits
■ Conversion Time	2.5 µs Max (RD Mode)
	1.5 µs Max (WR-RD Mode)
■ Low Power	75 mW Max
■ Total Unadjusted Error	±½ LSB and ± 1 LSB

Features

- Built-in track-and-hold function
- No missing codes
- No external clocking
- Single supply—5 V_{DC}
- Easy interface to all microprocessors, or operates stand-alone
- Latched TRI-STATE output
- Logic inputs and outputs meet both MOS and T^2L voltage level specifications
- Operates ratiometrically or with any reference value equal to or less than V_{CC}
- 0V to 5V analog input voltage range with single 5V supply
- No zero or full-scale adjust required
- Overflow output available for cascading
- 0.3" standard width 20-pin DIP
- 20-pin molded chip carrier package
- 20-pin small outline package
- 20-pin shrink small outline package (SSOP)

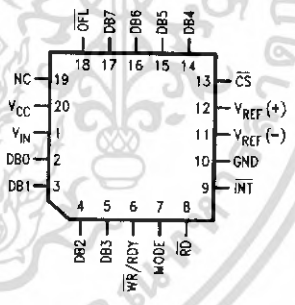
Connection and Functional Diagrams

Dual-In-Line, Small Outline and SSOP Packages



Top View

Molded Chip Carrier Package



© 2004 National Semiconductor Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Connection and Functional Diagrams (Continued)

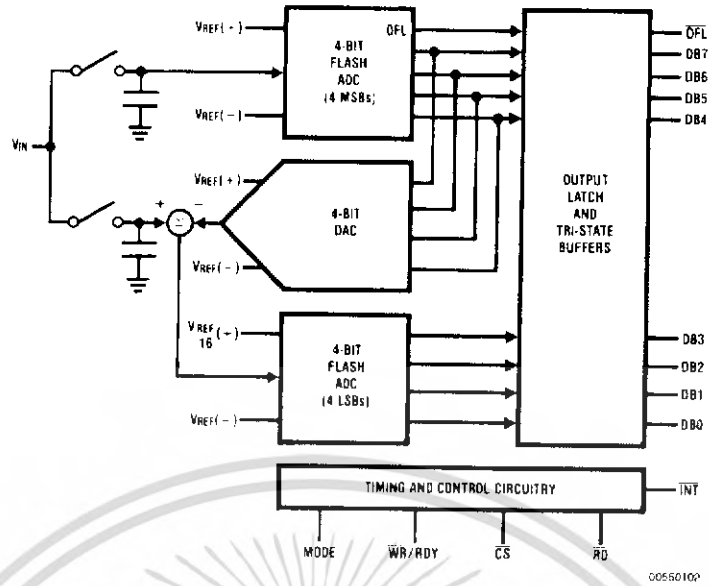


FIGURE 1.

Ordering Information

Part Number	Total Unadjusted Error	Package	Temperature Range
ADC0820BCV	±½ LSB	V20A — Molded Chip Carrier	0°C to +70°C
ADC0820BCWM		M20B — Wide Body Small Outline	0°C to +70°C
ADC0820BCN		N20A — Molded DIP	0°C to +70°C
ADC0820CCJ	±1 LSB	J20A — Cerdip	−40°C to +85°C
ADC0820CCWM		M20B — Wide Body Small Outline	0°C to +70°C
ADC0820CIWM		M20B — Wide Body Small Outline	−40°C to +85°C
ADC0820CCN		N20A — Molded DIP	0°C to +70°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings (Notes 1, 2)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Supply Voltage (V_{CC})	10V
Logic Control Inputs	−0.2V to V_{CC} +0.2V
Voltage at Other Inputs and Output	−0.2V to V_{CC} +0.2V
Storage Temperature Range	−65°C to +150°C
Package Dissipation at T_A = 25°C	875 mW
Input Current at Any Pin (Note 5)	1 mA
Package Input Current (Note 5)	4 mA
ESD Susceptability (Note 9)	900V
Lead Temp. (Soldering, 10 sec.)	
Dual-In-Line Package (plastic)	260°C
Dual-In-Line Package (ceramic)	300°C
Surface Mount Package	
Vapor Phase (60 sec.)	215°C
Infrared (15 sec.)	220°C

Operating Ratings (Notes 1, 2)

Temperature Range	$T_{MIN} \le T_A \le T_{MAX}$
ADC0820CCJ	−40°C≤T_A≤+85°C
ADC0820CIWM	−40°C≤T_A≤+85°C
ADC0820BCN, ADC0820CCN	0°C≤T_A≤70°C
ADC0820BCV	0°C≤T_A≤70°C
ADC0820BCWM, ADC0820CCWM	0°C≤T_A≤70°C
V_{CC} Range	4.5V to 8V

Converter Characteristics

The following specifications apply for RD mode (pin 7=0), V_{CC}=5V, $V_{REF}(+)$=5V,and $V_{REF}(-)$=GND unless otherwise specified. **Boldface limits apply from T_{MIN} to T_{MAX}**; all other limits $T_A=T_j$=25°C.

Parameter	Conditions	ADC0820CCJ			ADC0820BCN, ADC0820CCN ADC0820BCV, ADC0820BCWM ADC0820CCWM, ADC0820CIWM			Limit Units
		Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	
Resolution			8			8	8	Bits
Total Unadjusted Error (Note 3)	ADC0820BCN, BCWM					±½	±½	LSB
	ADC0820CCJ		±1					LSB
	ADC0820CCN, CCWM, CIWM					±1	±1	LSB
	ADC0820CCMSA					±1	±1	LSB
Minimum Reference Resistance		2.3	1.00		2.3	1.2		kΩ
Maximum Reference Resistance		2.3	6		2.3	5.3	6	kΩ
Maximum $V_{REF}(+)$ Input Voltage			V_{CC}			V_{CC}	V_{CC}	V
Minimum $V_{REF}(-)$ Input Voltage			GND			GND	GND	V
Minimum $V_{REF}(+)$ Input Voltage			$V_{REF}(-)$			$V_{REF}(-)$	$V_{REF}(-)$	V
Maximum $V_{REF}(-)$ Input Voltage			$V_{REF}(+)$			$V_{REF}(+)$	$V_{REF}(+)$	V
Maximum V_{IN} Input Voltage			V_{CC}+0.1			V_{CC}+0.1	V_{CC}+0.1	V
Minimum V_{IN} Input Voltage			GND−0.1			GND−0.1	GND−0.1	V
Maximum Analog Input Leakage Current	$\overline{CS}$ =V_{CC}							
	V_{IN}=V_{CC}		3			0.3	3	μA
	V_{IN}=GND		−3			−0.3	−3	μA
Power Supply Sensitivity	V_{CC}=5V±5%	±1/16	±¼		±1/16	±¼	±¼	LSB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC Electrical Characteristics

The following specifications apply for V_{CC}=5V, unless otherwise specified. **Boldface limits apply from T_{MIN} to T_{MAX}**; all other limits $T_A=T_J$=25°C.

Parameter	Conditions		ADC0820CCJ			ADC0820BCN, ADC0820CCN ADC0820BCV, ADC0820BCWM ADC0820CCWM, ADC0820CIWM			Limit Units
			Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	
$V_{IN(1)}$, Logical "1" Input Voltage	V_{CC}=5.25V	$\overline{CS}$, $\overline{WR}$, $\overline{RD}$		**2.0**			2.0	**2.0**	V
		Mode		**3.5**			3.5	**3.5**	V
$V_{IN(0)}$, Logical "0" Input Voltage	V_{CC}=4.75V	$\overline{CS}$, $\overline{WR}$, $\overline{RD}$		**0.8**			0.8	**0.8**	V
		Mode		**1.5**			1.5	**1.5**	V
$I_{IN(1)}$, Logical "1" Input Current	$V_{IN(1)}$=5V; $\overline{CS}$, $\overline{RD}$		0.005	**1**		0.005		**1**	µA
	$V_{IN(1)}$=5V; $\overline{WR}$		0.1	**3**		0.1	0.3	**3**	µA
	$V_{IN(1)}$=5V; Mode		50	**200**		50	170	**200**	µA
$I_{IN(0)}$, Logical "0" Input Current	$V_{IN(0)}$=0V; $\overline{CS}$, $\overline{RD}$, $\overline{WR}$, Mode		−0.005	**−1**		−0.005		**−1**	µA
$V_{OUT(1)}$, Logical "1" Output Voltage	V_{CC}=4.75V, I_{OUT}=−360 µA; DB0–DB7, $\overline{OFL}$, $\overline{INT}$			**2.4**			2.8	**2.4**	V
	V_{CC}=4.75V, I_{OUT}=−10 µA: DB0–DB7, $\overline{OFL}$, $\overline{INT}$			**4.5**			4.6	**4.5**	V
$V_{OUT(0)}$, Logical "0" Output Voltage	V_{CC}=4.75V, I_{OUT}=1.6 mA; DB0–DB7, $\overline{OFL}$, $\overline{INT}$, RDY			**0.4**			0.34	**0.4**	V
I_{OUT}, TRI-STATE Output Current	V_{OUT}=5V; DB0–DB7, RDY		0.1	**3**		0.1	0.3	**3**	µA
	V_{OUT}=0V; DB0–DB7, RDY		−0.1	**−3**		−0.1	−0.3	**−3**	µA
I_{SOURCE}, Output Source Current	V_{OUT}=0V; DB0–DB7, $\overline{OFL}$ $\overline{INT}$		−12	**−6**		−12	−7.2	**−6**	mA
			−9	**−4.0**		−9	−5.3	**−4.0**	mA
I_{SINK}, Output Sink Current	V_{OUT}=5V; DB0–DB7, $\overline{OFL}$, $\overline{INT}$, RDY		14	**7**		14	8.4	**7**	mA
I_{CC}, Supply Current	$\overline{CS}$ =$\overline{WR}$ =$\overline{RD}$ =0		7.5	**15**		7.5	13	**15**	mA

AC Electrical Characteristics

The following specifications apply for V_{CC}=5V, $t_r=t_f$=20 ns, $V_{REF}(+)$=5V, $V_{REF}(-)$=0V and T_A=25°C unless otherwise specified.

Parameter		Conditions	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Units
t_{CRD}, Conversion Time for RD Mode		Pin 7 = 0, *Figure 2*	1.6		2.5	µs
t_{ACCO}, Access Time (Delay from Falling Edge of $\overline{RD}$ to Output Valid)		Pin 7 = 0, *Figure 2*	t_{CRD}+20		t_{CRD}+50	ns
t_{CWR-RD}, Conversion Time for WR-RD Mode		Pin 7 = V_{CC}; t_{WR} = 600 ns, t_{RD}=600 ns; *Figures 3, 4*			1.52	µs
t_{WR}, Write Time	Min	Pin 7 = V_{CC}; *Figures 3, 4*		600		ns
	Max	(Note 4) See Graph	50			µs
t_{RD}, Read Time	Min	Pin 7 = V_{CC}; *Figures 3, 4* (Note 4) See Graph		600		ns
t_{ACC1}, Access Time (Delay from Falling Edge of $\overline{RD}$ to Output Valid)		Pin 7 = V_{CC}, t_{RD}<t_i; *Figure 3* C_L=15 pF	190		280	ns
		C_L=100 pF	210		320	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Electrical Characteristics (Continued)

The following specifications apply for V_{CC}=5V, t_r=t_f=20 ns, V_{REF}(+)=5V, V_{REF}(−)=0V and T_A=25°C unless otherwise specified.

Parameter	Conditions	Typ (Note 6)	Tested Limit (Note 7)	Design Limit (Note 8)	Units
t_{ACC2}, Access Time (Delay from Falling Edge of $\overline{RD}$ to Output Valid)	Pin 7 = V_{CC}, t_{RD}>t_i; *Figure 4*				
	C_L=15 pF	70		120	ns
	C_L=100 pF	90		150	ns
t_{ACC3}, Access Time (Delay from Rising Edge of RDY to Output Valid)	R_{PULLUP} = 1k and C_L = 15 pF	30			ns
t_i, Internal Comparison Time	Pin 7=V_{CC}; *Figures 4, 5* C_L=50 pF	800		1300	ns
t_{1H}, t_{0H}, TRI-STATE Control (Delay from Rising Edge of $\overline{RD}$ to Hi-Z State)	R_L=1k, C_L=10 pF	100		200	ns
$t_{\overline{INTL}}$, Delay from Rising Edge of $\overline{WR}$ to Falling Edge of $\overline{INT}$	Pin 7 = V_{CC}, C_L= 50 pF				
	t_{RD}>t_i; *Figure 4*			t_i	ns
	t_{RD}<t_i; *Figure 3*	t_{RD}+200		t_{RD}+290	ns
$t_{\overline{INTH}}$, Delay from Rising Edge of $\overline{RD}$ to Rising Edge of $\overline{INT}$	*Figures 2, 3, 4* C_L=50 pFc	125		225	ns
$t_{\overline{INTHWR}}$, Delay from Rising Edge of $\overline{WR}$ to Rising Edge of $\overline{INT}$	*Figure 5*, C_L=50 pF	175		270	ns
t_{RDY}, Delay from $\overline{CS}$ to RDY	*Figure 2*, C_L=50 pF, Pin 7 =0	50		100	ns
t_{ID}, Delay from $\overline{INT}$ to Output Valid	*Figure 5*	20		50	ns
t_{RI}, Delay from $\overline{RD}$ to $\overline{INT}$	Pin 7=V_{CC}, t_{RD}<t_i *Figure 3*	200		290	ns
t_P, Delay from End of Conversion to Next Conversion	*Figures 2, 3, 4, 5* (Note 4) See Graph			500	ns
Slew Rate, Tracking		0.1			V/μs
C_{VIN}, Analog Input Capacitance		45			pF
C_{OUT}, Logic Output Capacitance		5			pF
C_{IN}, Logic Input Capacitance		5			pF

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

Note 2: All voltages are measured with respect to the GND pin, unless otherwise specified.

Note 3: Total unadjusted error includes offset, full-scale, and linearity errors.

Note 4: Accuracy may degrade if t_{WR} or t_{RD} is shorter than the minimum value specified. See Accuracy vs. t_{WR} and Accuracy vs. t_{RD} graphs.

Note 5: When the input voltage (V_{IN}) at any pin exceeds the power supply rails (V_N < V^- or V_{IN} > V^+) the absolute value of current at that pin should be limited to 1 mA or less. The 4 mA package input current limits the number of pins that can exceed the power supply boundaries with a 1 mA current limit to four.

Note 6: Typicals are at 25°C and represent most likely parametric norm.

Note 7: Tested limits are guaranteed to National's AOQL (Average Outgoing Quality Level).

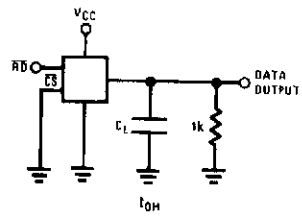
Note 8: Design limits are guaranteed but not 100% tested. These limits are not used to calculate outgoing quality levels.

Note 9: Human body model, 100 pF discharged through a 1.5 kΩ resistor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

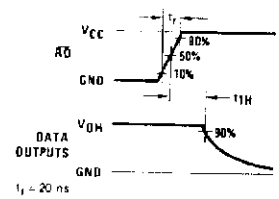
TRI-STATE Test Circuits and Waveforms

t_{1H}



00550103

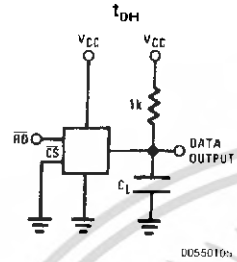
t_{1H}, C_L = 10 pF



00550104

t_r=20 ns

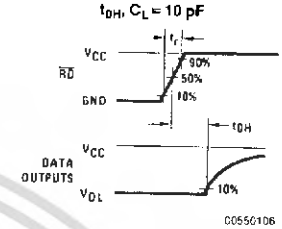
t_{0H}



00550105

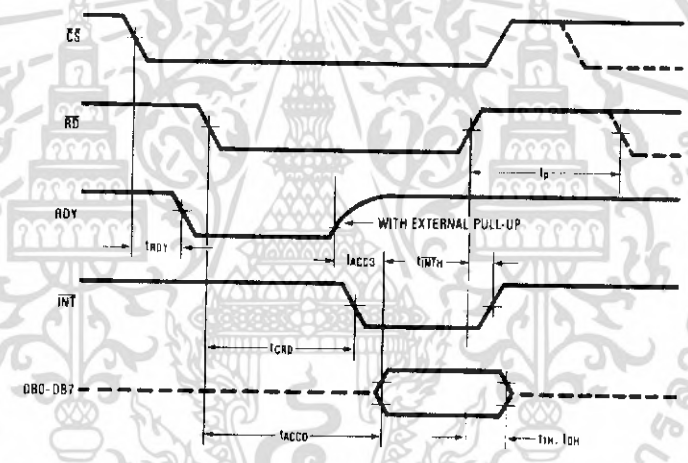
t_r=20 ns

t_{0H}, C_L = 10 pF



00550106

Timing Diagrams



00550107

Note: On power-up the state of $\overline{INT}$ can be high or low.

FIGURE 2. RD Mode (Pin 7 is Low)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timing Diagrams (Continued)

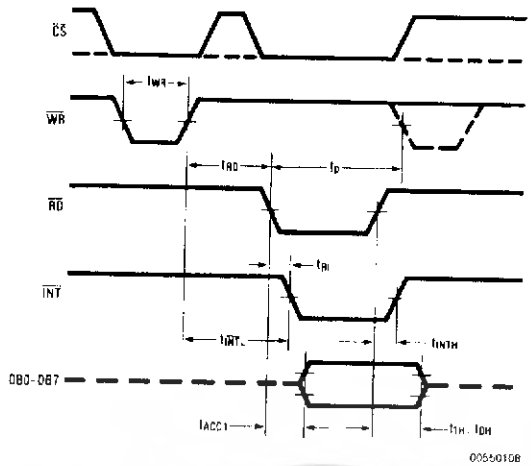


FIGURE 3. WR-RD Mode (Pin 7 is High and $t_{RD}<t_i$)

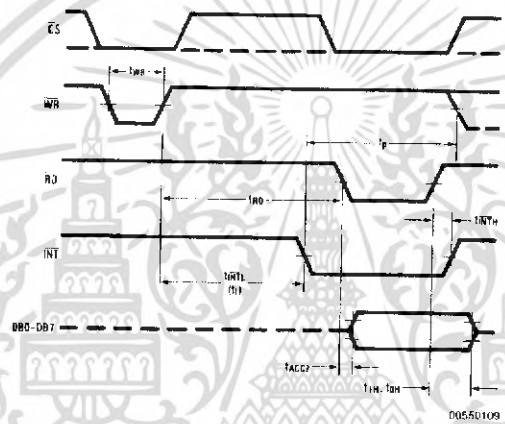


FIGURE 4. WR-RD Mode (Pin 7 is High and $t_{RD}>t_i$)

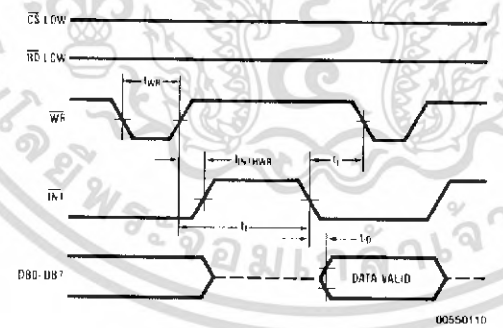
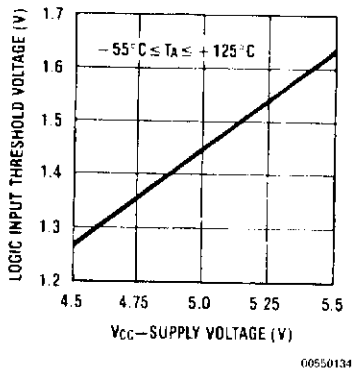


FIGURE 5. WR-RD Mode (Pin 7 is High)
Stand-Alone Operation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

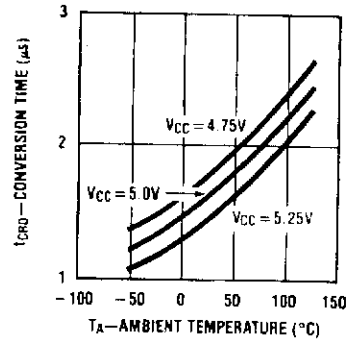
Typical Performance Characteristics

Logic Input Threshold Voltage vs. Supply Voltage



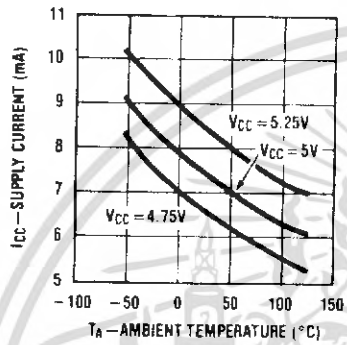
00550134

Conversion Time (RD Mode) vs. Temperature



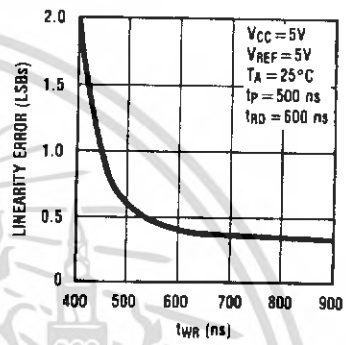
00550135

Power Supply Current vs. Temperature
(not including reference ladder)



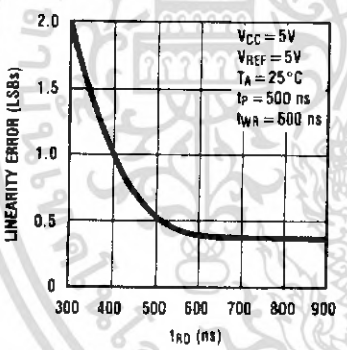
00550136

Accuracy vs. t_{WR}



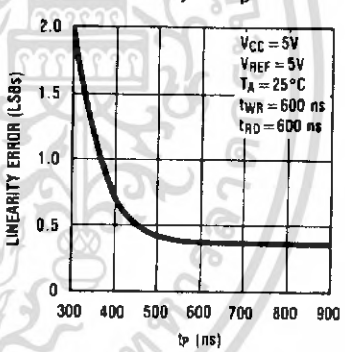
00550137

Accuracy vs. t_{RD}



00550138

Accuracy vs. t_P

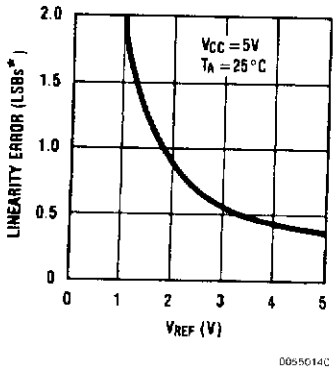


00550139

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Performance Characteristics (Continued)

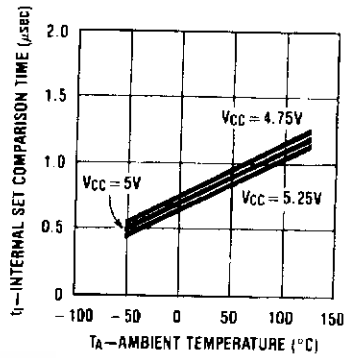
Accuracy vs. V_{REF} [$V_{REF}=V_{REF}(+)-V_{REF}(-)$]



00550140

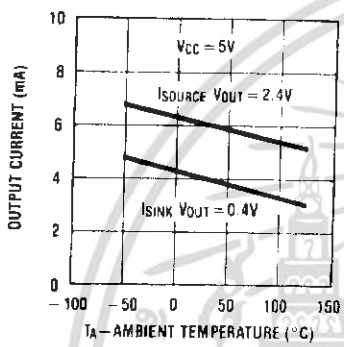
*1 LSB $= \frac{V_{REF}}{256}$

t_I, Internal Time Delay vs. Temperature



00550141

Output Current vs. Temperature



00550142

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Description of Pin Functions

Pin	Name	Function
1	V_{IN}	Analog input; range $=GND \leq V_{IN} \leq V_{CC}$
2	DB0	TRI-STATE data output—bit 0 (LSB)
3	DB1	TRI-STATE data output—bit 1
4	DB2	TRI-STATE data output—bit 2
5	DB3	TRI-STATE data output—bit 3
6	$\overline{WR}$ /RDY	**WR-RD Mode** **$\overline{WR}$:** With $\overline{CS}$ low, the conversion is started on the falling edge of $\overline{WR}$. Approximately 800 ns (the preset internal time out, t_i) after the $\overline{WR}$ rising edge, the result of the conversion will be strobed into the output latch, provided that $\overline{RD}$ does not occur prior to this time out (see *Figures 3, 4*). **RD Mode** **RDY:** This is an open drain output (no internal pull-up device). RDY will go low after the falling edge of $\overline{CS}$; RDY will go TRI-STATE when the result of the conversion is strobed into the output latch. It is used to simplify the interface to a microprocessor system (see *Figure 2*).
7	Mode	**Mode:** Mode selection input—it is internally tied to GND through a 50 µA current source. **RD Mode:** When mode is low **WR-RD Mode:** When mode is high
8	$\overline{RD}$	**WR-RD Mode** With $\overline{CS}$ low, the TRI-STATE data outputs (DB0-DB7) will be activated when $\overline{RD}$ goes low (see *Figure 5*). $\overline{RD}$ can also be used to increase the speed of the converter by reading data prior to the preset internal time out (t_i, ~800 ns). If this is done, the data result transferred to output latch is latched after the falling edge of the $\overline{RD}$ (see *Figures 3, 4*). **RD Mode** With $\overline{CS}$ low, the conversion will start with $\overline{RD}$ going low, also $\overline{RD}$ will enable the TRI-STATE data outputs at the completion of the conversion. RDY going TRI-STATE and $\overline{INT}$ going low indicates the completion of the conversion (see *Figure 2*).
9	$\overline{INT}$	**WR-RD Mode** $\overline{INT}$ going low indicates that the conversion is completed and the data result is in the output latch. $\overline{INT}$ will go low, ~800 ns (the preset internal time out, t_i) after the rising edge of $\overline{WR}$ (see *Figure 4*); or $\overline{INT}$ will go low after the falling edge of $\overline{RD}$, if $\overline{RD}$ goes low prior to the 800 ns time out (see *Figure 3*). $\overline{INT}$ is reset by the rising edge of $\overline{RD}$ or $\overline{CS}$ (see *Figures 3, 4*). **RD Mode** $\overline{INT}$ going low indicates that the conversion is completed and the data result is in the output latch. $\overline{INT}$ is reset by the rising edge of $\overline{RD}$ or $\overline{CS}$ (see *Figure 2*).
10	GND	Ground
11	$V_{REF}(-)$	The bottom of resistor ladder, voltage range: $GND \leq V_{REF}(-) \leq V_{REF}(+)$ (Note 5)
12	$V_{REF}(+)$	The top of resistor ladder, voltage range: $V_{REF}(-) \leq V_{REF}(+) \leq V_{CC}$ (Note 5)
13	$\overline{CS}$	$\overline{CS}$ must be low in order for the $\overline{RD}$ or $\overline{WR}$ to be recognized by the converter.
14	DB4	TRI-STATE data output—bit 4
15	DB5	TRI-STATE data output—bit 5
16	DB6	TRI-STATE data output—bit 6
17	DB7	TRI-STATE data output—bit 7 (MSB)
18	$\overline{OFL}$	Overflow output—If the analog input is higher than the $V_{REF}(+)$, $\overline{OFL}$ will be low at the end of conversion. It can be used to cascade 2 or more devices to have more resolution (9, 10-bit). This output is always active and does not go into TRI-STATE as DB0–DB7 do.
19	NC	No connection
20	V_{CC}	Power supply voltage

1.0 Functional Description

1.1 GENERAL OPERATION

The ADC0820 uses two 4-bit flash A/D converters to make an 8-bit measurement (*Figure 1*). Each flash ADC is made up of 15 comparators which compare the unknown input to a reference ladder to get a 4-bit result. To take a full 8-bit reading, one flash conversion is done to provide the 4 most significant data bits (via the MS flash ADC). Driven by the 4 MSBs, an internal DAC recreates an analog approximation of the input voltage. This analog signal is then subtracted from the input, and the difference voltage is converted by a second 4-bit flash ADC (the LS ADC), providing the 4 least significant bits of the output data word.

The internal DAC is actually a subsection of the MS flash converter. This is accomplished by using the same resistor ladder for the A/D as well as for generating the DAC signal. The DAC output is actually the tap on the resistor ladder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.0 Functional Description (Continued)

which most closely approximates the analog input. In addition, the "sampled-data" comparators used in the ADC0820 provide the ability to compare the magnitudes of several analog signals simultaneously, without using input summing amplifiers. This is especially useful in the LS flash ADC, where the signal to be converted is an analog difference.

1.2 THE SAMPLED-DATA COMPARATOR

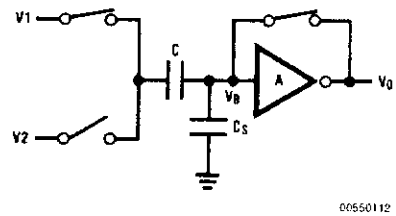
Each comparator in the ADC0820 consists of a CMOS inverter with a capacitively coupled input (*Figures 6, 7*). Analog switches connect the two comparator inputs to the input capacitor (C) and also connect the inverter's input and output. This device in effect now has one differential input pair. A comparison requires two cycles, one for zeroing the comparator, and another for making the comparison.

In the first cycle, one input switch and the inverter's feedback switch (*Figure 6*) are closed. In this interval, C is charged to the connected input (V1) less the inverter's bias voltage (V_B, approximately 1.2V). In the second cycle (*Figure 7*), these two switches are opened and the other (V2) input's switch is closed. The input capacitor now subtracts its stored voltage from the second input and the difference is amplified by the inverter's open loop gain. The inverter's input (V_B') becomes

$$V_B - (V1 - V2)\frac{C}{C + C_S}$$

and the output will go high or low depending on the sign of $V_B'-V_B$.

The actual circuitry used in the ADC0820 is a simple but important expansion of the basic comparator described above. By adding a second capacitor and another set of switches to the input (*Figure 8*), the scheme can be expanded to make dual differential comparisons. In this circuit, the feedback switch and one input switch on each capacitor (Z switches) are closed in the zeroing cycle. A comparison is then made by connecting the second input on each capacitor and opening all of the other switches (S switches). The change in voltage at the inverter's input, as a result of the change in charge on each input capacitor, will now depend on both input signal differences.

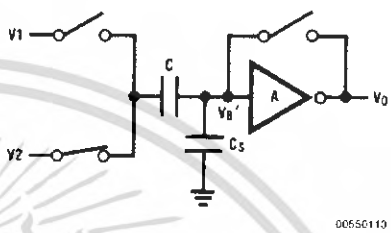


00550112

- $V_O = V_B$
- V on C = V1–V_B
- C_S = stray input node capacitor
- V_B = inverter input bias voltage

Zeroing Phase

FIGURE 6. Sampled-Data Comparator

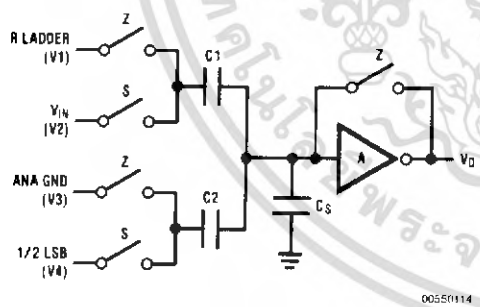


00550113

- $V_B' - V_B = (V2 - V1)\frac{C}{C + C_S}$
- $V_O' = \frac{-A}{C + C_S}[CV2 - CV1]$
- V_O' is dependent on V2 – V1

Compare Phase

FIGURE 7. Sampled-Data Comparator



00550114

$$V_O = \frac{-A}{C1 + C2 + C_S}[C1(V2 - V1) + C2(V4 - V3)]$$

$$= \frac{-A}{C1 + C2 + C_S}[\Delta Q_{C1} + \Delta Q_{C2}]$$

00550145

FIGURE 8. ADC0820 Comparator (from MS Flash ADC)

1.3 ARCHITECTURE

In the ADC0820, one bank of 15 comparators is used in each 4-bit flash A/D converter (*Figure 12*). The MS (most significant) flash ADC also has one additional comparator to detect input overrange. These two sets of comparators operate alternately, with one group in its zeroing cycle while the other is comparing.

When a typical conversion is started, the $\overline{WR}$ line is brought low. At this instant the MS comparators go from zeroing to

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.0 Functional Description (Continued)

comparison mode (*Figure 11*). When $\overline{WR}$ is returned high after at least 600 ns, the output from the first set of comparators (the first flash) is decoded and latched. At this point the two 4-bit converters change modes and the LS (least significant) flash ADC enters its compare cycle. No less than 600 ns later, the $\overline{RD}$ line may be pulled low to latch the lower 4 data bits and finish the 8-bit conversion. When $\overline{RD}$ goes low, the flash A/Ds change state once again in preparation for the next conversion.

Figure 11 also outlines how the converter's interface timing relates to its analog input (V_{IN}). In WR-RD mode, V_{IN} is measured while $\overline{WR}$ is low. In RD mode, sampling occurs during the first 800 ns of $\overline{RD}$. Because of the input connections to the ADC0820's LS and MS comparators, the converter has the ability to sample V_{IN} at one instant (Section 2.4), despite the fact that two separate 4-bit conversions are being done. More specifically, when $\overline{WR}$ is low the MS flash is in compare mode (connected to V_{IN}), and the LS flash is in zero mode (also connected to V_{IN}). Therefore both flash ADCs sample V_{IN} at the same time.

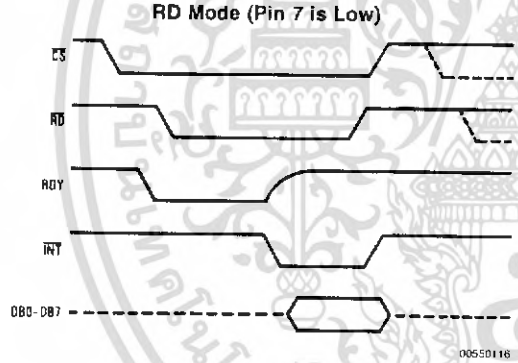
1.4 DIGITAL INTERFACE

The ADC0820 has two basic interface modes which are selected by strapping the MODE pin high or low.

RD Mode

With the MODE pin grounded, the converter is set to Read mode. In this configuration, a complete conversion is done by pulling $\overline{RD}$ low until output data appears. An $\overline{INT}$ line is provided which goes low at the end of the conversion as well as a RDY output which can be used to signal a processor that the converter is busy or can also serve as a system Transfer Acknowledge signal.

RD Mode (Pin 7 is Low)



When in RD mode, the comparator phases are internally triggered. At the falling edge of $\overline{RD}$, the MS flash converter goes from zero to compare mode and the LS ADC's comparators enter their zero cycle. After 800 ns, data from the MS flash is latched and the LS flash ADC enters compare mode. Following another 800 ns, the lower 4 bits are recovered.

WR then RD Mode

With the MODE pin tied high, the A/D will be set up for the WR-RD mode. Here, a conversion is started with the $\overline{WR}$ input; however, there are two options for reading the output data which relate to interface timing. If an interrupt driven scheme is desired, the user can wait for $\overline{INT}$ to go low before reading the conversion result (*Figure 10*). $\overline{INT}$ will typically go low 800 ns after $\overline{WR}$'s rising edge. However, if a shorter conversion time is desired, the processor need not wait for $\overline{INT}$ and can exercise a read after only 600 ns (*Figure 9*). If this is done, $\overline{INT}$ will immediately go low and data will appear at the outputs.

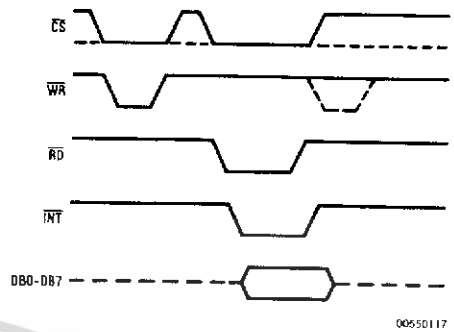


FIGURE 9. WR-RD Mode (Pin 7 is High and $t_{RD}<t_1$)

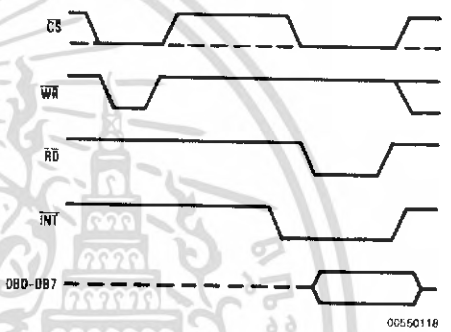
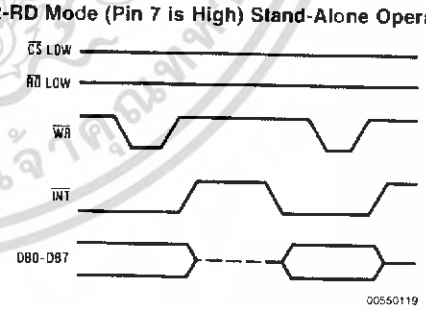


FIGURE 10. WR-RD Mode (Pin 7 is High and $t_{RD}>t_1$)

Stand-Alone

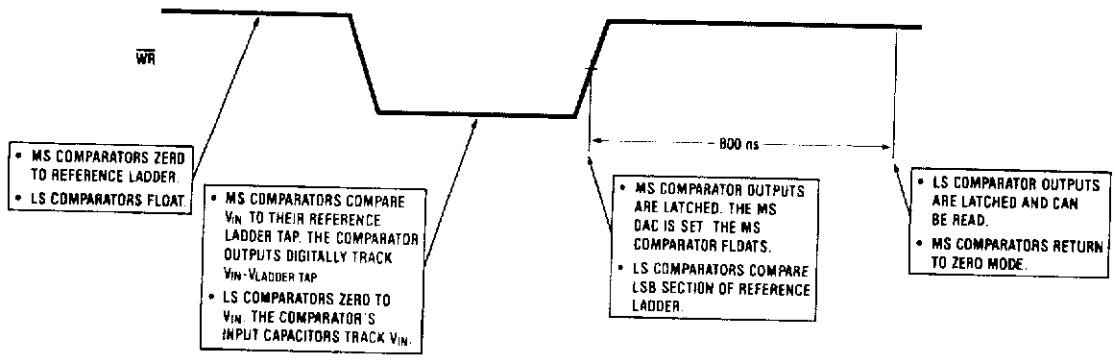
For stand-alone operation in WR-RD mode, $\overline{CS}$ and $\overline{RD}$ can be tied low and a conversion can be started with $\overline{WR}$. Data will be valid approximately 800 ns following $\overline{WR}$'s rising edge.

WR-RD Mode (Pin 7 is High) Stand-Alone Operation



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.0 Functional Description (Continued)



Note: MS means most significant
LS means least significant

FIGURE 11. Operating Sequence (WR-RD Mode)

OTHER INTERFACE CONSIDERATIONS

In order to maintain conversion accuracy, $\overline{WR}$ has a maximum width spec of 50 µs. When the MS flash ADC's sampled-data comparators (Section 1.2) are in comparison mode ($\overline{WR}$ is low), the input capacitors (C, *Figure 8*) must hold their charge. Switch leakage and inverter bias current can cause errors if the comparator is left in this phase for too long.

Since the MS flash ADC enters its zeroing phase at the end of a conversion (Section 1.3), a new conversion cannot be started until this phase is complete. The minimum spec for this time (t_P, *Figures 2, 3, 4, 5*) is 500 ns.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Detailed Block Diagram

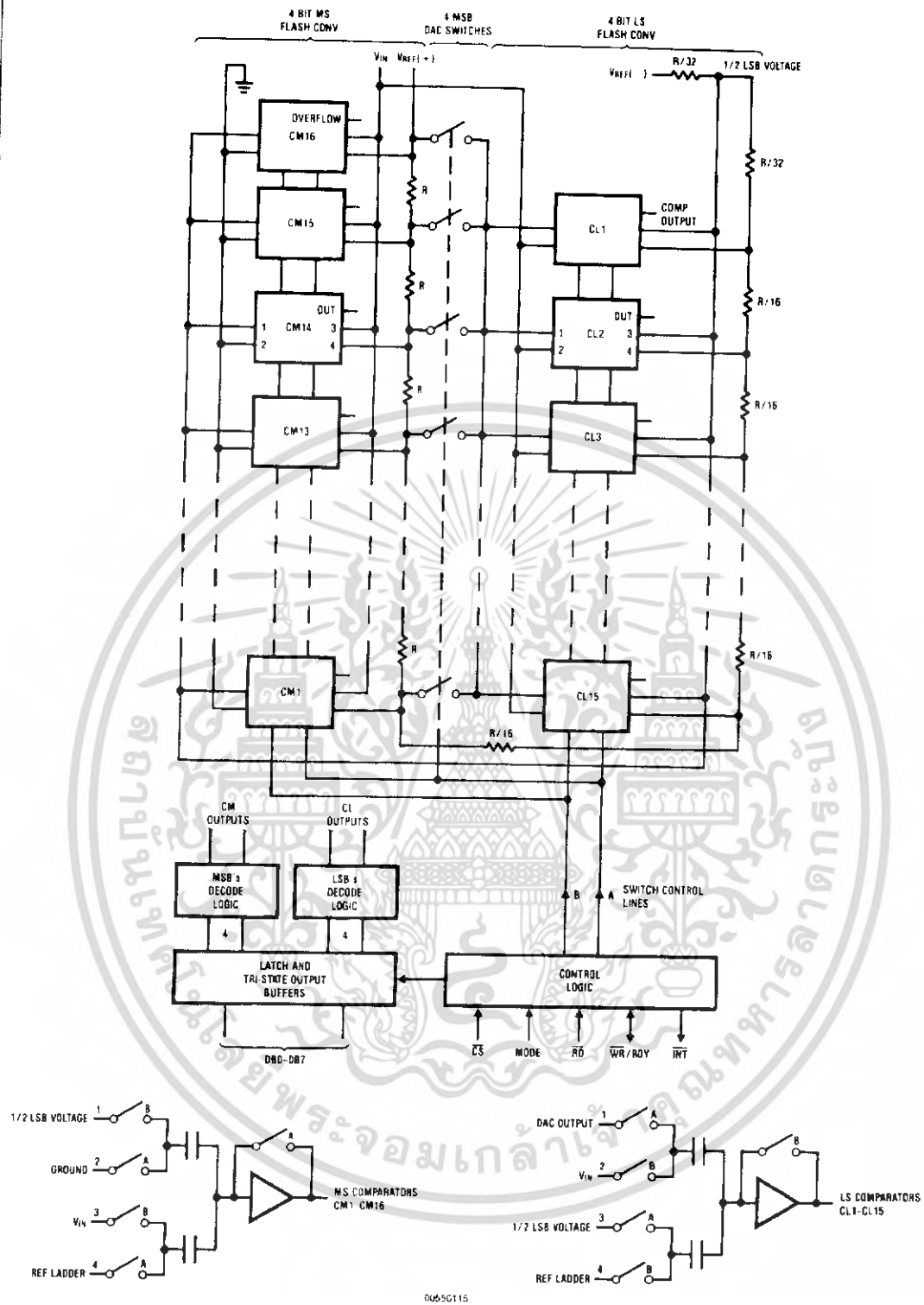


FIGURE 12.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.0 Analog Considerations

2.1 REFERENCE AND INPUT

The two V_{REF} inputs of the ADC0820 are fully differential and define the zero to full-scale input range of the A to D converter. This allows the designer to easily vary the span of the analog input since this range will be equivalent to the voltage difference between $V_{IN}(+)$ and $V_{IN}(-)$. By reducing $V_{REF}(V_{REF}=V_{REF}(+)-V_{REF}(-))$ to less than 5V, the sensitivity of the converter can be increased (i.e., if $V_{REF}=2V$ then 1 LSB=7.8 mV). The input/reference arrangement also facilitates ratiometric operation and in many cases the chip power supply can be used for transducer power as well as the V_{REF} source.

This reference flexibility lets the input span not only be varied but also offset from zero. The voltage at $V_{REF}(-)$ sets the input level which produces a digital output of all zeroes. Though V_{IN} is not itself differential, the reference design affords nearly differential-input capability for most measurement applications. *Figure 13* shows some of the configurations that are possible.

2.2 INPUT CURRENT

Due to the unique conversion techniques employed by the ADC0820, the analog input behaves somewhat differently than in conventional devices. The A/D's sampled-data comparators take varying amounts of input current depending on which cycle the conversion is in.

The equivalent input circuit of the ADC0820 is shown in *Figure 14*. When a conversion starts ($\overline{WR}$ low, WR-RD mode), all input switches close, connecting V_{IN} to thirty-one 1 pF capacitors. Although the two 4-bit flash circuits are not both in their compare cycle at the same time, V_{IN} still sees all input capacitors at once. This is because the MS flash converter is connected to the input during its compare interval and the LS flash is connected to the input during its zeroing phase (Section 1.3). In other words, the LS ADC uses V_{IN} as its zero-phase input.

The input capacitors must charge to the input voltage through the on resistance of the analog switches (about 5 kΩ to 10 kΩ). In addition, about 12 pF of input stray capacitance must also be charged. For large source resistances, the analog input can be modeled as an RC network as shown in *Figure 15*. As R_S increases, it will take longer for the input capacitance to charge.

In RD mode, the input switches are closed for approximately 800 ns at the start of the conversion. In WR-RD mode, the time that the switches are closed to allow this charging is the time that $\overline{WR}$ is low. Since other factors force this time to be at least 600 ns, input time constants of 100 ns can be accommodated without special consideration. Typical total input capacitance values of 45 pF allow R_S to be 1.5 kΩ without lengthening $\overline{WR}$ to give V_{IN} more time to settle.

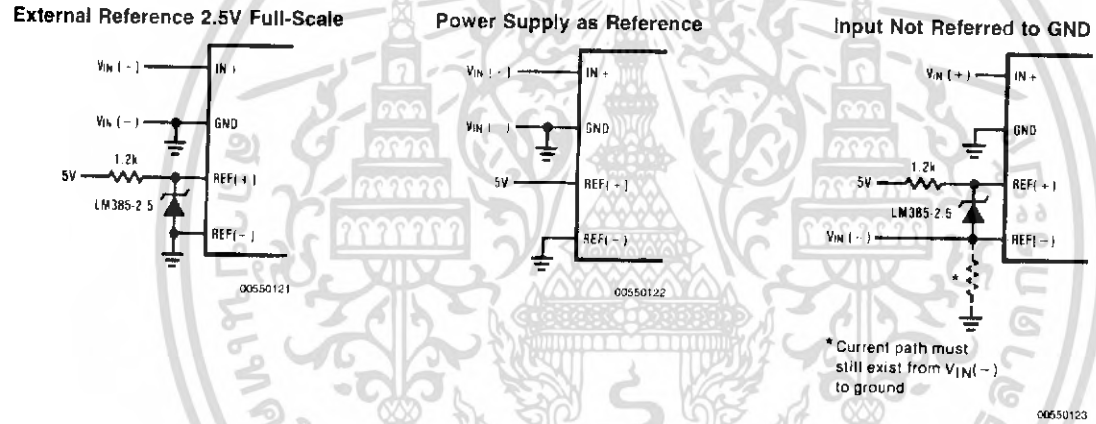


FIGURE 13. Analog Input Options

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.0 Analog Considerations (Continued)

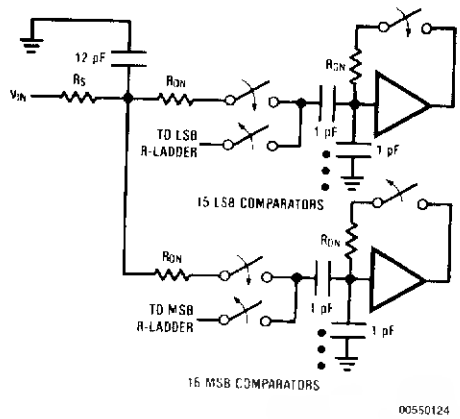


FIGURE 14.

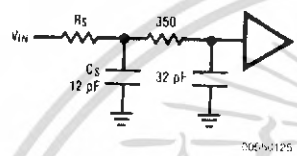


FIGURE 15.

2.3 INPUT FILTERING

It should be made clear that transients in the analog input signal, caused by charging current flowing into V_{IN}, will not degrade the A/D's performance in most cases. In effect the ADC0820 does not "look" at the input when these transients occur. The comparators' outputs are not latched while $\overline{WR}$ is low, so at least 600 ns will be provided to charge the ADC's input capacitance. It is therefore not necessary to filter out these transients by putting an external cap on the V_{IN} terminal.

2.4 INHERENT SAMPLE-HOLD

Another benefit of the ADC0820's input mechanism is its ability to measure a variety of high speed signals without the help of an external sample-and-hold. In a conventional SAR type converter, regardless of its speed, the input must remain at least ½ LSB stable throughout the conversion process if full accuracy is to be maintained. Consequently, for many high speed signals, this signal must be externally sampled, and held stationary during the conversion.

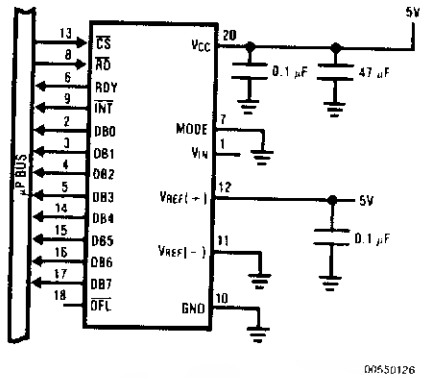
Sampled-data comparators, by nature of their input switching, already accomplish this function to a large degree (Section 1.2). Although the conversion time for the ADC0820 is 1.5 µs, the time through which V_{IN} must be ½ LSB stable is much smaller. Since the MS flash ADC uses V_{IN} as its "compare" input and the LS ADC uses V_{IN} as its "zero" input, the ADC0820 only "samples" V_{IN} when $\overline{WR}$ is low (Sections 1.3 and 2.2). Even though the two flashes are not done simultaneously, the analog signal is measured at one instant. The value of V_{IN} approximately 100 ns after the rising edge of $\overline{WR}$ (100 ns due to internal logic prop delay) will be the measured value.

Input signals with slew rates typically below 100 mV/µs can be converted without error. However, because of the input time constants, and charge injection through the opened comparator input switches, faster signals may cause errors. Still, the ADC0820's loss in accuracy for a given increase in signal slope is far less than what would be witnessed in a conventional successive approximation device. An SAR type converter with a conversion time as fast as 1 µs would still not be able to measure a 5V 1 kHz sine wave without the aid of an external sample-and-hold. The ADC0820, with no such help, can typically measure 5V, 7 kHz waveforms.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

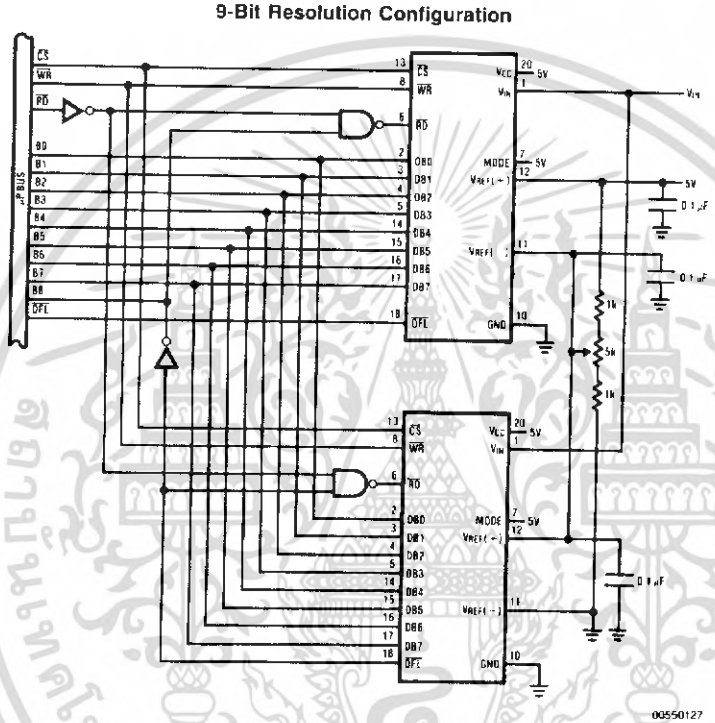
3.0 Typical Applications

8-Bit Resolution Configuration



00550126

9-Bit Resolution Configuration

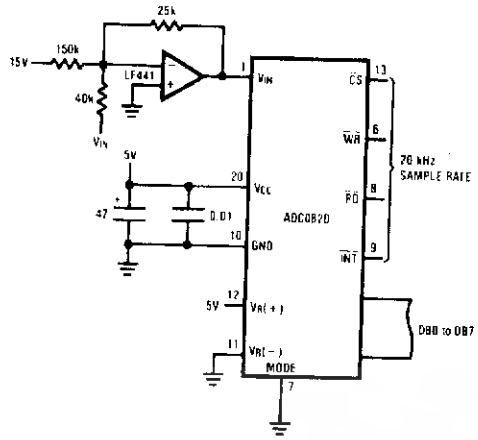


00550127

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

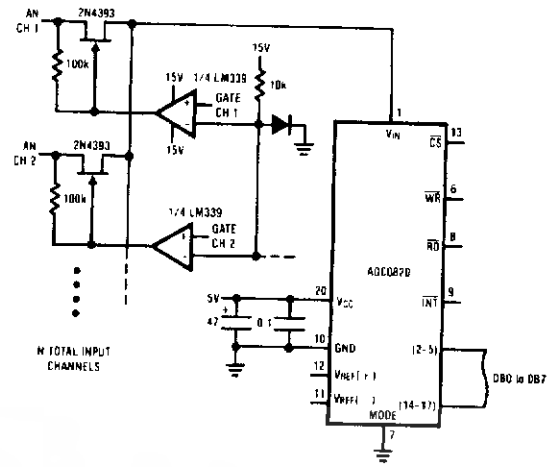
3.0 Typical Applications (Continued)

Telecom A/D Converter



00550128

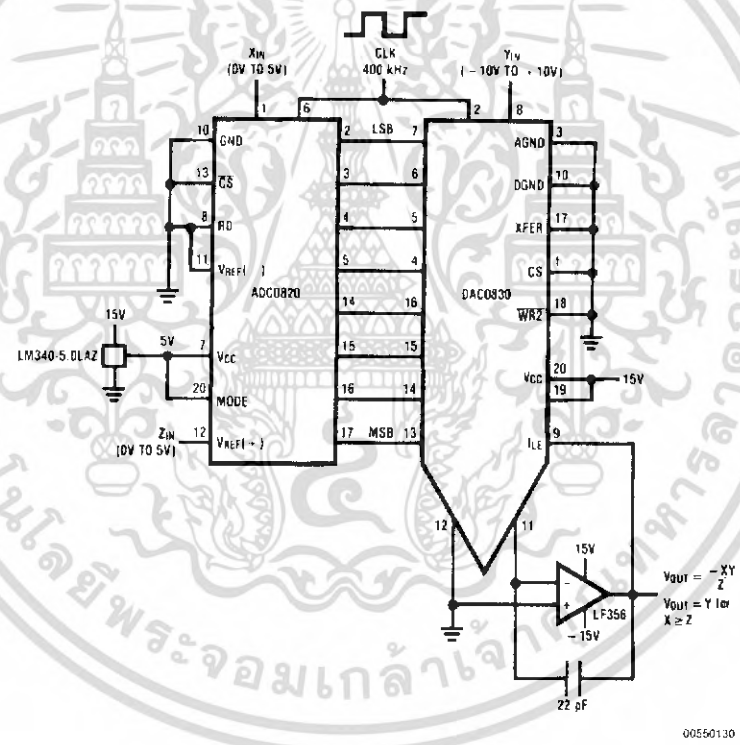
Multiple Input Channels



00550129

- V_{IN}=3 kHz max ± $4V_p$
- No track-and-hold needed
- Low power consumption

8-Bit 2-Quadrant Analog Multiplier

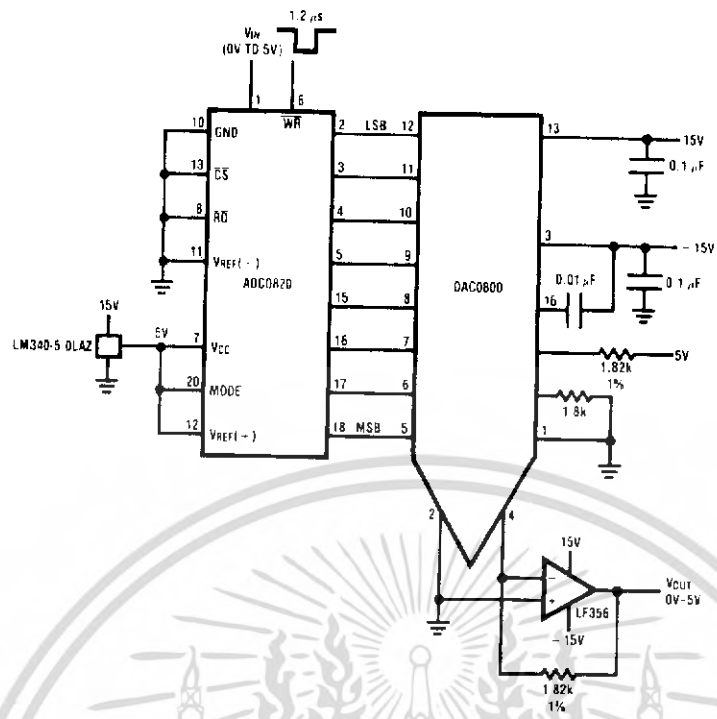


00550130

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

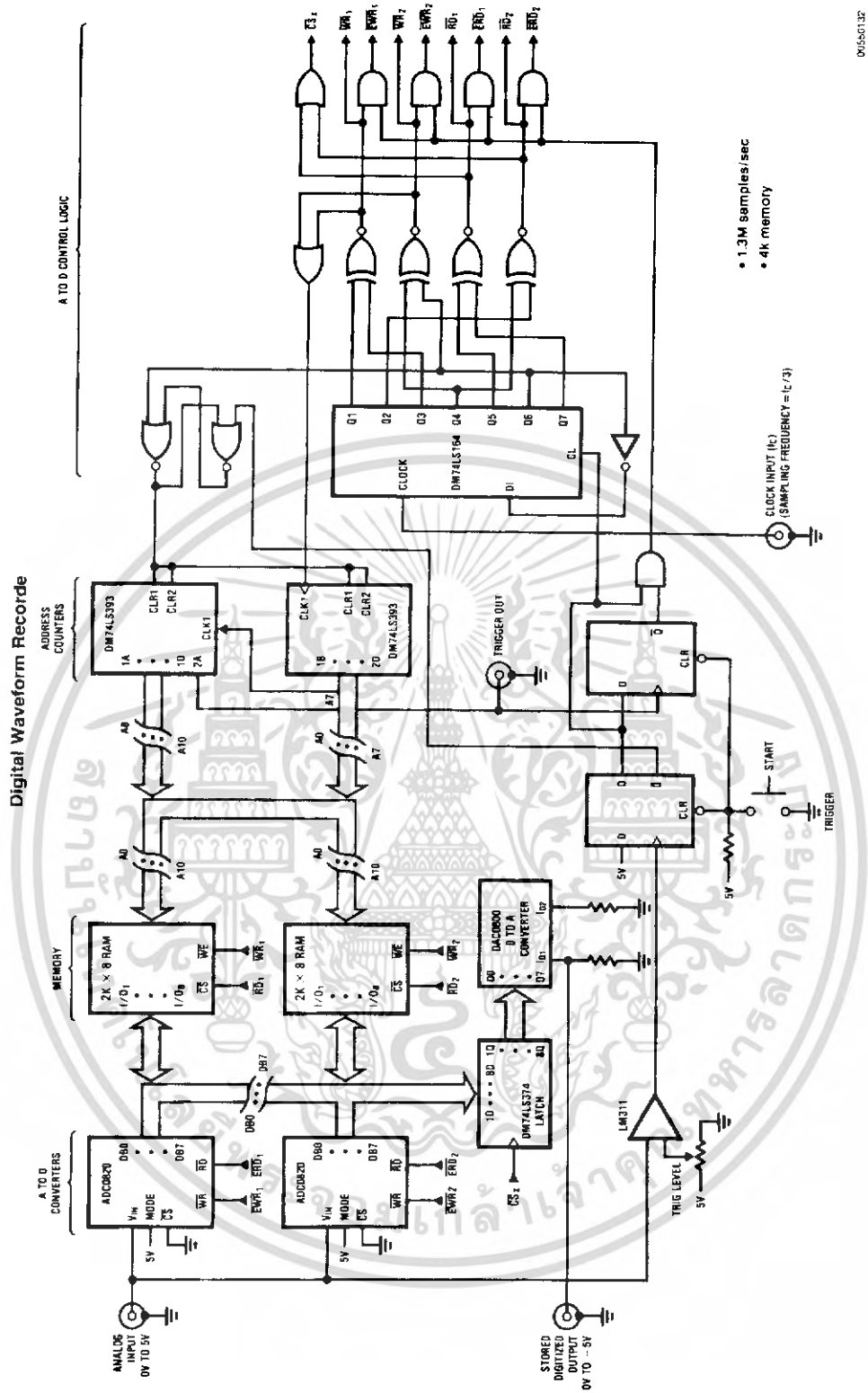
3.0 Typical Applications (Continued)

Fast Infinite Sample-and-Hold



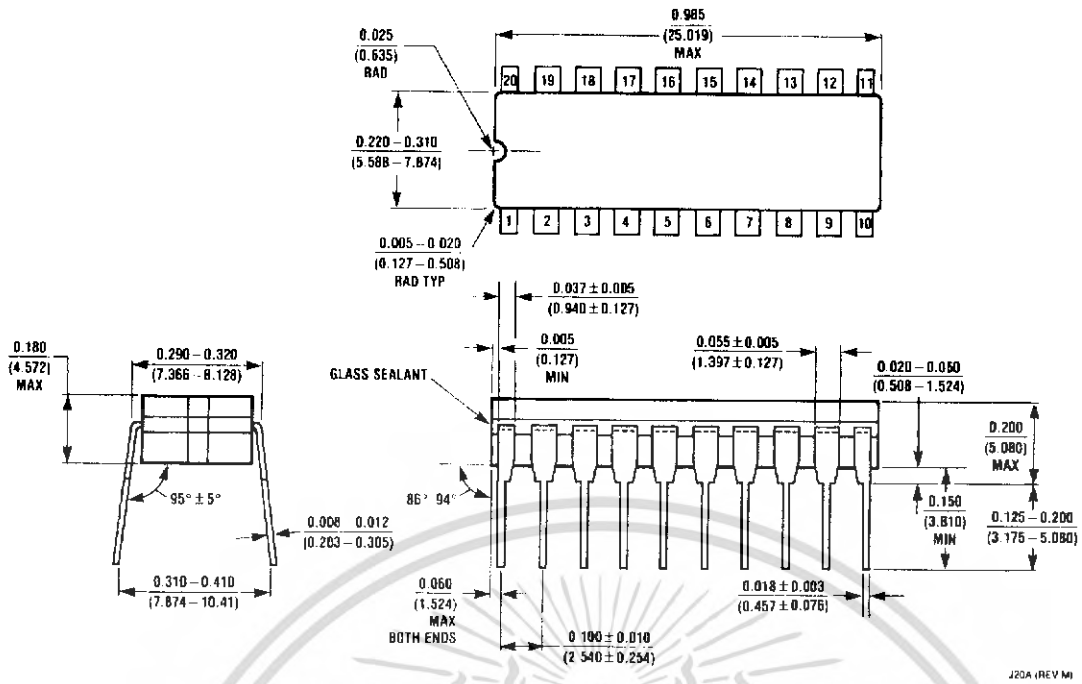
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.0 Typical Applications (Continued)

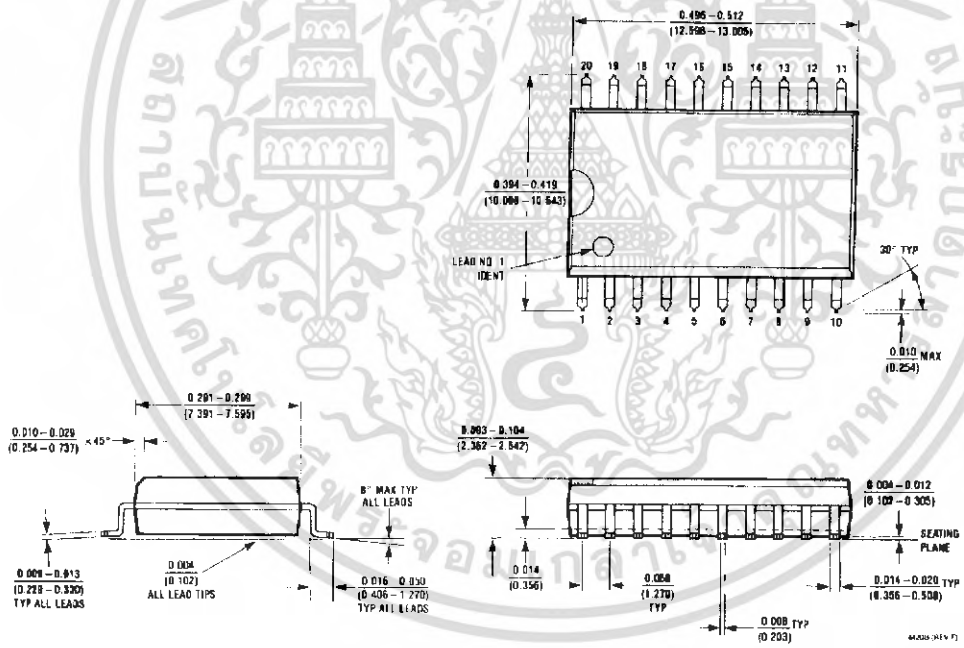


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted



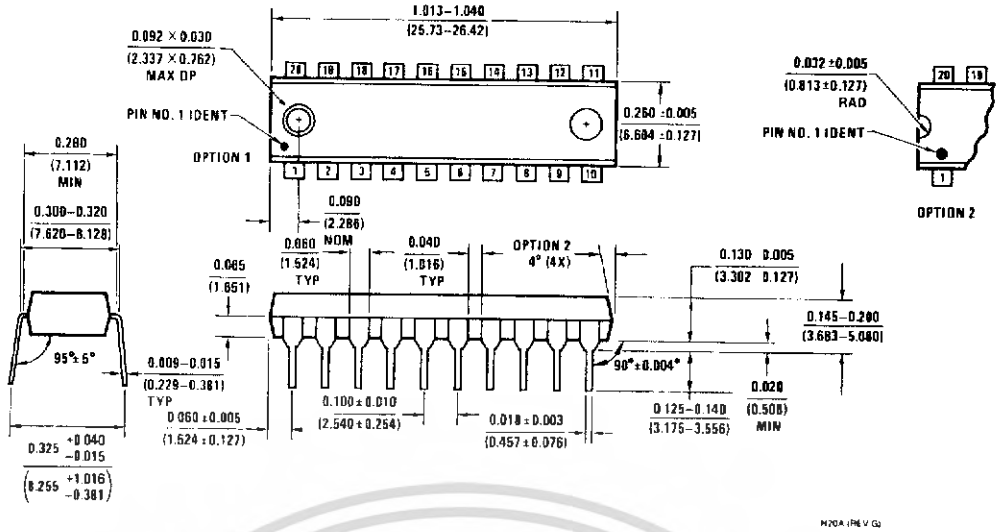
Hermetic Dual-In-Line Package (J)
Order Number ADC0820CCJ
NS Package Number J20A



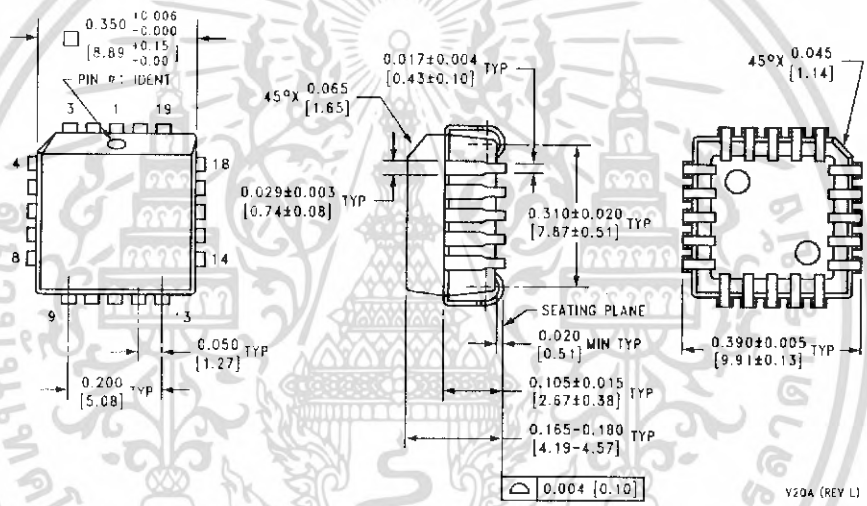
SO Package (M)
Order Number ADC0820BCWM, ADC0820CCWM or ADC0820CIWM
NS Package Number M20B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



Molded Dual-In-Line Package (N)
Order Number ADC0820BCN or ADC0820CCN
NS Package Number N20A



Molded Chip Carrier Package (V)
Order Number ADC0820BCV
NS Package Number V20A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Notes

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

BANNED SUBSTANCE COMPLIANCE

National Semiconductor certifies that the products and packing materials meet the provisions of the Customer Products Stewardship Specification (CSP-9-111C2) and the Banned Substances and Materials of Interest Specification (CSP-9-111S2) and contain no "Banned Substances" as defined in CSP-9-111S2.

National Semiconductor
Americas Customer
Support Center
Email: new.feedback@nsc.com
Tel: 1-800-272-9959

www.national.com

National Semiconductor
Europe Customer Support Center
Fax: +49 (0) 180-530 85 86
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 69 9508 6208
English Tel: +44 (0) 870 24 0 2171
Français Tel: +33 (0) 1 41 91 8790

National Semiconductor
Asia Pacific Customer
Support Center
Email: ap.support@nsc.com

National Semiconductor
Japan Customer Support Center
Fax: 81-3-5639-7507
Email: jpn.feedback@nsc.com
Tel: 81-3-5639-7560

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้