สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การเชื่อมต่อตัวตรวจจับภาพแบบ CMOS

CMOS IMAGE SENSOR INTERFACING

โดย

นายธนศักย์ แสงจันทร์ รหัส 47010311

นางสาวธารทิพย์ เบญจมงคลชัย รหัส 47010332

อาจารย์ที่ปรึกษา

รศ.ดร. สุรพันธุ์ เอื้อไพบูลย์

ปพ.
ธ157ก
2550

เลขหมู่....................

เลขทะเบียน 82205

วัน,เดือน,ปี -9 ก.ค. 2551

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

b 11aU5825

i..............................

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านธุรกิจ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อตัวตรวจจับภาพแบบ CMOS
CMOS IMAGE SENSOR INTERFACING

โดย

นายธนศักย์ แสงจันทร์

นางสาวธารทิพย์ เบญจมงคลชัย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2550

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2550
ภาควิชาอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง การเชื่อมต่อตัวตรวจจับภาพแบบ CMOS

ผู้จัดทำ

1. นายธนศักย์ แสงจันทร์ รหัส 47010311
2. นางสาวธารทิพย์ เบญจมงคลชัย รหัส 47010332

.... อาจารย์ที่ปรึกษา

(.รศ.ดร. สุรพันธุ์.........)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อตัวตรวจจับภาพแบบ CMOS

นายธนศักย์ แสงจันทร์ รหัส 47010311
นางสาวธารทิพย์ เบญจมงคลชัย รหัส 47010332
รศ.ดร. สุรพันธุ์ เอื้อไพบูลย์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2550

บทคัดย่อ

รายงานฉบับนี้ ได้อธิบายเกี่ยวกับการประยุกต์ใช้งานกล้องแบบ CMOS เบอร์ C328R โดยการนำภาพที่ได้แสดงผลบนจอภาพสี Nokia 6100 ผ่านบอร์ดไมโครคอนโทลเลอร์ Atmel AVR ATmega128

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CMOS IMAGE SENSOR INTERFACING

Mr.Thanasak Sangjan ID.47010311

Miss.Tarntip Benjamongkolchai ID.47010332

Assist.Prof.Dr. Surapan Airphaioon (Advisor)

Education Year 2007

Abstract

This report describes about application of an interface between CMOS camera (C328R) and color LCD display (Nokia 6100). The interface was implemented using the Atmel AVR ATmega128 microcontroller.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิติกรรมประกาศ

โครงงานนี้สำเร็จได้ด้วยดีเนื่องจากการสนับสนุนจากอาจารย์ที่ปรึกษา รศ.ดร. สุรพันธุ์ เอื้อไพบูลย์ ผู้ที่คอยให้คำปรึกษาและคำแนะนำในการทำโครงงาน ขอขอบคุณรุ่นพี่ทุกคนที่ให้ความช่วยเหลือในการแก้ไขปัญหาต่างๆ และสุดท้ายนี้ขอขอบคุณคุณชุติมันต์ ยงประพัฒน์ ฝ่ายเซลล์ซัพพอร์ทของบริษัทวีนัส ซัพพลาย จำกัด ผู้ที่คอยให้คำปรึกษาเรื่องการเขียนโปรแกรมของจอสี Nokia LCD

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของโครงงาน	1
1.2 ขอบเขตของโครงงาน	1
1.3 วัตถุประสงค์	1
บทที่ 2 ทฤษฎี	2
2.1 เซ็นเซอร์รับภาพแบบ CMOS และCCD	2
2.2 ความรู้พื้นฐานของการประมวลผลภาพกราฟฟิก	6
2.2.1 ความรู้เรื่องความละเอียด	6
2.2.2 ความรู้เรื่องบิต	7
2.2.3 ระบบสี (Color Model)	7
2.2.4 รูปแบบของไฟล์	10
บทที่ 3 รายละเอียดเกี่ยวกับโมดูลกล้อง	12
3.1 โมดูลกล้อง E700	12
3.1.1 สัญญาณเอาท์พุต	13
3.1.2 วงจรที่ใช้กับโมดูลกล้อง	14
3.1.3 การอินเตอร์เฟส โมดูลกล้องกับคอมพิวเตอร์	15
3.1.4 หลักการเขียนโปรแกรมเพื่อรับข้อมูลภาพ	15
3.1.5 บทสรุปของการใช้กล้อง E700	15
3.2 กล้อง C328R	16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 4 ไมโครคอนโทรลเลอร์ AVR 19

4.1 รายละเอียดเกี่ยวกับ AVR 19

4.2 การต่อแรมให้กับ AVR 20

4.3 การสื่อสารข้อมูลอนุกรมผ่านโมดูล USART 21

บทที่ 5 จอสี LCD 24

บทที่ 6 การเขียนซอฟต์แวร์ 26

6.1 การเขียนซอฟต์แวร์ให้กับกล้อง C328R 26

6.2 การเขียนซอฟต์แวร์ให้กับจอสี LCD 29

บทที่ 7 การทดสอบ 33

บทที่ 8 บทสรุป 38

ภาคผนวก

บรรณานุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 การทำงานของกล้องแบบ CMOS และ CCD 3
รูปที่ 2.2 พิกเซลของภาพเฉดสีขาว/ดำ และของภาพสี 6
รูปที่ 2.3 การแสดงสีระบบ Addivtiv 8
รูปที่ 2.4 การแสดงสีระบบ Subtractive 9
รูปที่ 3.1 โมดูลกล้องการจัดเรียงขาของโมดูลกล้อง E700 12
รูปที่ 3.2 แสดงสัญญาณเอาท์พุตต่างๆของโมดูลกล้อง 13
รูปที่ 3.3 การป้อนไฟเลี้ยง และความถี่ให้กับโมดูลกล้อง 14
รูปที่ 3.4 การสร้างแรงดันไฟเลี้ยงของวงจร 14
รูปที่ 3.5 บล็อกไดอะแกรมของกล้อง C328R 16
รูปที่ 3.6 ขาสัญญาณต่างๆของกล้อง 16
รูปที่ 3.7 วงจรที่ใช้เชื่อมต่อ C328R กับคอมพิวเตอร์ 17
รูปที่ 3.8 หน้าตาของโปรแกรมถ่ายภาพ 18
รูปที่ 4.1 การแบ่งส่วนของหน่วยความจำภายใน AVR 20
รูปที่ 4.2 วงจรการต่อแรมเพิ่มให้กับ AVR 20
รูปที่ 4.3 บล็อกไดอะแกรมโมดูล USART 21
รูปที่ 4.4 รูปแบบเฟรมข้อมูลอนุกรม 22
รูปที่ 5.1 จอสี LCD 24
รูปที่ 5.2 วงจรสร้างแรงดันไฟเลี้ยงให้กับจอสี LCD 25
รูปที่ 6.1 การเชื่อมต่อกับกล้องด้วยคำสั่ง Sync 27
รูปที่ 6.2 การส่งคำสั่งเพื่อถ่ายภาพนิ่ง 28
รูปที่ 6.3 ไทม์มิ่งไดอะแกรม (timing diagram) การส่งข้อมูลอนุกรม 9 บิต 29
รูปที่ 6.4 ทิศทางการแสดงจุดสีของจอสี LCD 31
รูปที่ 7.1 การแสดงผลข้อมูลที่ได้จากการจับภาพสีแดงครั้งที่ 1 33
รูปที่ 7.2 การแสดงผลข้อมูลที่ได้จากการจับภาพสีแดงครั้งที่ 2 34
รูปที่ 7.3 การแสดงข้อมูลที่ได้จากการจับภาพสีน้ำเงินครั้งที่ 1 34
รูปที่ 7.4 การแสดงข้อมูลที่ได้จากการจับภาพสีน้ำเงินครั้งที่ 2 35
รูปที่ 7.5 การแสดงภาพสีแดงบนจอสี LCD 35

เอกสารนี้เป็นเอกสารที่ส่งในนามหน่วยการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 7.6 การแสดงภาพสีน้ำเงินบนจอสี LCD 36
รูปที่ 7.7 การแสดงภาพสีเขียวบนจอสี LCD 37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

		หน้า
ตารางที่ 2.1	แสดงสีที่แสดงได้ในแต่ละจำนวนบิตข้อมูลต่อพิกเซล	7
ตารางที่ 3.1	แสดงขาต่างๆของกล้อง	12
ตารางที่ 4.1	การคำนวณหาอัตราบอตเรต	23
ตารางที่ 6.1	ชุดคำสั่งของกล้อง C328R	26
ตารางที่ 6.2	แสดงการเขียนข้อมูล สำหรับ RGB 4:4:4	31
ตารางที่ 6.3	แสดงการเขียนข้อมูล สำหรับ RGB 3:3:2	31
ตารางที่ 6.4	แสดงการเขียนข้อมูล สำหรับ RGB 5:6:5	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของโครงงาน

ในปัจจุบัน จะเห็นได้ว่า กล้องได้ถูกใช้ประโยชน์อย่างแพร่หลาย เช่น การแชทที่มีการส่งให้เห็นภาพคู่สนทนา ด้วยเว็บแคมการใช้กล้องถ่ายภาพ และกล้องวีดิโอเพื่อบันทึกภาพนิ่งหรือภาพเคลื่อนไหว การใช้กล้องวงจรปิดเพื่อจับภาพคนร้าย ที่เห็นได้ชัดก็คือ โทรศัพท์มือถือ ซึ่งส่วนใหญ่ก็จะมีฟังก์ชันกล้องติดตั้งอยู่ด้วย เพราะจะนั้น จึงเกิดความสนใจที่จะทำโครงงานเกี่ยวกับการอินเตอร์เฟสกล้อง โดยโครงงานทั้งหมด จะเป็นการรับภาพจากตัวตรวจจับภาพแบบ CMOS ผ่านทางไมโครคอนโทรลเลอร์ AVR แล้วนำภาพที่ได้ไปแสดงบนจอ LCD ยี่ห้อ Nokia

1.2 ขอบเขตของโครงงาน

โครงงานนี้ จะเป็นการรับภาพนิ่งจากกล้องแบบ CMOS แล้วนำสัญญาณภาพที่ได้ ไปแสดงผลบนจอ LCD โดยใช้ไมโครคอนโทลเลอร์เป็นตัวเชื่อมต่อ

1.3 วัตถุประสงค์

1. เพื่อศึกษาการใช้งานไมโครคอนโทรลเลอร์ AVR
2. เพื่อศึกษาการทำงานของโมดูลกล้องแบบ CMOS
3. เพื่อศึกษาการประยุกต์ใช้งานจอสี LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี

2.1 เซ็นเซอร์รับภาพแบบ CMOS และ CCD

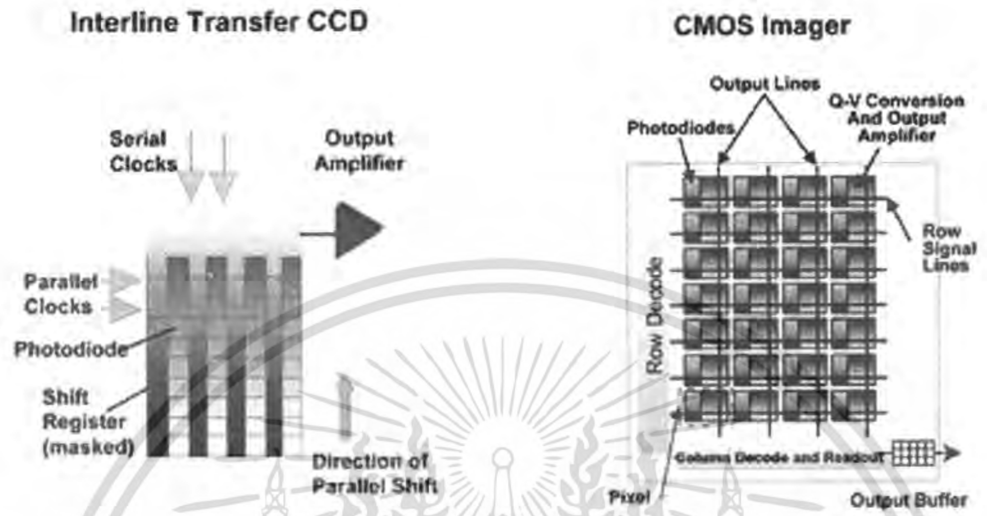
เทคโนโลยีการถ่ายภาพดิจิตอลนั้นได้มีการพัฒนาอย่างรวดเร็วในช่วง ระยะเวลา 5-6 ปีที่ผ่านมา สิ่งสำคัญที่ทำให้การถ่ายภาพก้าวกระโดดจากระบบฟิล์มมาเป็นระบบ ดิจิตอล ก็คือตัวรับรู้ หรือ เซ็นเซอร์ (Sensor) ซึ่งปัจจุบันนี้เซ็นเซอร์มีใช้กันอยู่สองชนิด คือ CCD และ CMOS

เซ็นเซอร์ทั้งสองชนิดต่างก็เป็นอุปกรณ์อิเล็คทรอนิกส์ ที่ผลิตมาจากซิลิคอน โดยมีหน้าที่เปลี่ยนแสงให้ เป็นสัญญาณไฟฟ้า ในแผง CCD และCMOS จะประกอบด้วยเซลล์รับแสงจำนวนมาก เรียกว่าจุดภาพ หรือพิกเซล (Pixel, Picture Element) ซึ่งเมื่อแต่ระจุดภาพรับแสงแล้ว จะเปลี่ยนโฟตอน (แสง) เป็นประจุไฟฟ้าก่อน จากนั้นประจุเหล่านั้นจะถูกถ่ายโอนไปยังส่วนเอาต์พุท เพื่อเปลี่ยนเป็นความต่างศักย์ ทำการบัพเฟอร์ (พักข้อมูล) ไว้ แล้วจึงส่งออกจากชิพในลักษณะที่เป็นสัญญาณอนาลอกทุกๆเซลล์สามารถใช้เป็นตัวรับแสงได้ สัญญาณเอาด์พุทที่ได้จะมีความสม่ำเสมอสูง (มีความสำคัญต่อคุณภาพของภาพ)

สำหรับเซ็นเซอร์ที่เป็น CMOSนั้น แต่ละพิกเซลจะมีส่วนที่ทำหน้าที่แปลงประจุเป็นความต่างศักย์อยู่ในตัวเอง นอกจากนี้ยังมีส่วนที่เปลี่ยนสัญญาณอนาลอกให้เป็นดิจิตอลอยู่ในเซนเซอร์ด้วย สิ่งต่างๆ ที่เพิ่มเข้าไปในเซ็นเซอร์นี้ ทำให้พื้นที่ในการรับแสงลดน้อยลง และเมื่อแต่ละพิกเซลสามารถทำการแปลงค่าได้เองนั้น ทำให้ความสม่ำเสมอของสัญญาณไม่ดีนัก แต่ก็ไม่จำเป็นต้องมีการประมวลผลอีกมากมายภายหลัง เพราะมีการทำไว้แล้วในระดับพิกเซล ดังนั้นสัญญาณที่ออกจาก CMOS จึงเป็นสัญญาณแบบดิจิตอล ขณะที่ CCD เป็นแบบอนาลอก

CCD จะให้คุณภาพที่ดีกว่า แต่กระบวนการผลิต และการนำไปใช้งานจะซับซ้อนกว่า กินพลังงานมากกว่า ทำให้ CCD มีราคาแพงกว่า CMOS ดังจะเห็นได้ว่า CMOS จะใช้ในกล้องคุณภาพต่ำและราคาถูก แต่มาจนปี 2002 CMOS คุณภาพสูงก็ได้ถูกพัฒนาขึ้นมาให้ดีกว่าเดิม ประกอบกับซอฟแวร์การจัดการภาพมีความก้าวหน้ามากยิ่งขึ้น ทำให้คุณภาพของภาพที่ได้จากกล้องที่ใช้เซ็นเซอร์ แบบ CMOS เทียบเท่ากับกล้องที่ใช้เซ็นเซอร์ แบบ CCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 การทำงานของกล้องแบบ CMOS และ CCD

เทคโนโลยีกล้องดิจิตอลที่ติดมากับโทรศัพท์มือถือ ในปัจจุบันจะนิยมใช้เซ็นเซอร์รับภาพอยู่ 2 แบบ คือ CMOS และ CCD โดยจะเป็นการนำเทคโนโลยีจากกล้องดิจิตอลทั่วไปมาประยุกต์ใช้ ซึ่งความแตกต่างของเซ็นเซอร์รับภาพ 2 แบบ จะเป็นดังนี้

CMOS (Complementary Metal Oxide Semiconductor): เป็นเซนเซอร์รับภาพชนิดที่มักจะถูกเลือกนำไปใช้กับโทรศัพท์มือถือ ที่มีราคาไม่แพง ซึ่งเป็นโทรศัพท์มือถือส่วนมากในท้องตลาด ในการคำนวณค่าของแสงที่มาตกกระทบที่แต่ละโฟโต้ไซต์(Photosite) จะมีการประจุค่านั้นโดยตรง เช่นเดียวกับ CCD แต่การส่งผ่านข้อมูลต้องอาศัยสายข้อมูลขนาดเล็ก ไปทำการประมวลผลอีกทอดหนึ่ง อาศัยเทคโนโลยีการผลิตแบบเก่า ซึ่งเป็นแบบเดียวกับการผลิตไมโครโพรเซสเซอจึงเต็มไปด้วยสัญญาณรบกวน มีคุณสมบัติความไวในการรับแสงต่ำ ดังนั้น เวลาอยู่ในสถานที่ๆ มีแสงน้อย จะให้เกิดนอยซ (Noise) หรืออาการพร่ามัวมากกว่า มีอัตราการสูญเสียเม็ดสีมากกว่า จึงมีผลทำให้คุณภาพของภาพถ่ายที่ได้ด้อยกว่า CCDแต่ข้อดีก็คือใช้พลังงานต่ำกว่า และมีขนาดของระบบการทำงานที่เล็กกว่า ส่วนการผลิตสามารถทำได้บนแผ่นซิลิคอนมาตรฐานทั่วไป จึงทำให้ต้นทุนในการผลิตต่ำกว่า CCD

CCD (Charge Coupled Device): เป็นเซ็นเซอร์รับภาพชนิดที่มีเทคโนโลยีใหม่กว่า ให้คุณภาพของภาพที่สูงกว่า ในการคำนวณค่าของแสงที่มาตกกระทบที่แต่ละโฟโต้ไซต์จะมีการประจุค่านั้น โดยตรงเช่นเดียวกับ CMOS และจะแปลงค่าแสงที่เป็นอนาล็อก ให้เป็นแบบดิจิตอล ซึ่งกระบวนการเหล่านี้เกิดขึ้นอย่างรวดเร็ว และส่งประจุได้โดยตรงไปยังชิพโดยไม่เกิดการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัดทอนสัญญาณ หรือสิ่งที่ รบกวนสัญญาณภาพ ซึ่งเกิดจากเทคโนโลยีกระบวนการผลิตขั้นสูง เพื่อให้เซนเซอร์มีคุณภาพ มีความไวในการรับภาพสูงกว่า มีอัตราการสูญเสียเม็ดสีน้อยกว่า สามารถรองรับการถ่ายภาพในที่ๆ มีแสงน้อยได้ดีกว่า ซึ่งทำให้คุณภาพของภาพถ่ายที่ได้ดีกว่า CMOS แต่ CCD ก็ยังต้องใช้พลังงานมากกว่า CMOS อยู่ และการผลิตต้องใช้แผ่นซิลิคอนแบบพิเศษที่ผลิตขึ้นมาโดยเฉพาะ จึงทำให้ต้นทุนในการผลิตนั้นสูงกว่า CMOS ไปด้วย

การเปรียบเทียบกันระหว่างเซ็นเซอร์ CMOS และ CCD

การตอบสนองการใช้งาน

จะพบว่าเซ็นเซอร์ CMOS ให้การตอบสนองการใช้งานที่ได้เปรียบกว่าและรวดเร็วกว่าเซ็นเซอร์CCD เพราะเหตุว่าการให้สัญญาณออกจากตัวรับแสงเป็นสัญญาณดิจิตอลโดยตรงไม่ต้องแปลงสัญญาณอีกครั้งหนึ่งรวมทั้งภาคขยายสัญญาณอยู่ในตัวทำให้กินพลังงานที่ต่ำกว่า ซึ่งถ้าเป็นระบบเซ็นเซอร์ CCD จำเป็นต้องมีวงจรสำหรับการขยายสัญญาณแปลงสัญญาณอีกชุดหนึ่งทำให้สิ้นเปลืองพลังงานมากกว่า อีกประการหนึ่งก็คือ การออกแบบภายในกล้องซึ่งเซ็นเซอร์ CMOS เพียงแต่มีเลนส์และระบบจัดการก็สามารถออกแบบได้กะทัดรัดแล้วในขณะที่ ระบบเซ็นเซอร์ CCD ต้องการพื้นที่ภายในกล้องสำหรับใส่แผ่นวงจรเพิ่มอีกหนึ่งชุดซึ่งทำให้การออกแบบนั้นมีความยุ่งยากมากกว่า

ความสามารถช่วงกว้างในการรับแสง (Dynamic Range)

เมื่อพิจารณาถึงเรื่องความสามารถช่วงกว้างในการรับแสงเพื่อเก็บรายละเอียดของบริเวณมืดและสว่างของภาพจะพบว่า การที่ระบบเซ็นเซอร์ CCD ทำหน้าที่เพียงการรับสัญญาณแสงแล้วส่งผ่านออกไปจากตัวรับภาพโดยตรงเพื่อไปเข้าวงจรสำหรับการแปลงสัญญาณอีกครั้งหนึ่ง ทำให้ได้สัญญาณที่มีความ สม่ำเสมอสูงกว่า ระบบเซ็นเซอร์ CMOS อีกทั้งไม่มีวงจรการแปลงสัญญาณที่ยุ่งยากภายในตัวรับแสงเองที่ก่อให้เกิดความร้อนที่เซ็นเซอร์รับภาพน้อยกว่าของเซ็นเซอร์ CMOS ดังนั้น ช่วงกว้างในการเก็บรายละเอียดของ เซ็นเซอร์ที่เป็น CCD จึงทำได้ดีกว่าเซ็นเซอร์ที่เป็น CMOS และ นอยซ หรือสัญญาณรบกวนของ CCDก็มี ต่ำกว่าของ CMOS

การตอบสนองต่อการเปิดปิดการรับแสง (Shuttering)

การตอบสนองต่อการเปิดปิดการรับแสง จะพบได้ว่าถ้าตัวรับภาพเป็นแบบ CCD ซึ่งทำหน้าที่รับแสงแปลงเป็นสัญญาณไฟฟ้าส่งผ่านออกมาแต่เพียงอย่างเดียว จะสามารถตอบสนองการเปิดปิดรับแสงแล้วให้สัญญาณที่มีความเป็นยูนิฟอมิทิ (Uniformity) ได้ดีกว่าตัวรับภาพที่เป็น CMOS ซึ่งมีระบบการแปลงสัญญาณที่ซับซ้อนกว่าเพื่อส่งสัญญาณที่เป็นดิจิตอลออกจากตัว CMOS มีโอกาสที่สัญญาณภาพจะเกิดความเป็นนอนยูนิฟืมิทิ (Nonuniformity) มากกว่าใน CCD แน่นอนว่าปัญหานี้ทำให้ได้สัญญาณที่เป็นรอง CCD ซึ่งผู้ผลิต CMOS ก็ได้พยายามพัฒนาวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อิเล็กทรอนิกส์อื่นๆ มาสนับสนุนให้ดีขึ้น ซึ่งก็สามารถที่จะแก้ปัญหาส่วนนี้ได้ระดับหนึ่งแต่ก็จะมีปัญหาในเรื่องต้นทุนการผลิตที่สูงขึ้นเช่นกัน ดังนั้นสำหรับในกล้องระดับล่างๆ ส่วนใหญ่ก็จะไม่ได้รับการแก้ปัญหาดังกล่าวยกเว้นแต่จะเป็นกล้องในระดับที่สูงขึ้นเท่านั้น

ความสามารถในการส่งผ่านข้อมูลในระบบปฏิบัติการ (Windowing)

จุดเด่นอย่างหนึ่งของเซ็นเซอร์ CMOS ก็คือความสามารถในการส่งผ่านข้อมูลที่รวดเร็วในระบบปฏิบัติการ เนื่องจากเป็นการส่งสัญญาณที่เป็นดิจิตอลโดยตรงออกจากเซ็นเซอร์รับภาพ จึงทำให้เซ็นเซอร์ CMOS มีความสามารถในการจับภาพวัตถุในพื้นที่จำกัดได้น่าสนใจกว่าทั้งยังสามารถที่จะพัฒนาต่อเนื่องได้ง่ายกว่าเซ็นเซอร์ CCD ที่มีข้อจำกัดที่มากกว่าในการพัฒนา

การลดการขยายวงของแสง (Antiblooming)

การขยายวงของแสงที่ล้นกรอบ Pixel ที่รับแสงเมื่อได้รับแสงมากเกินไป ส่วนนี้จะทำให้ภาพที่ถ่ายโดยการรับแสงมากเกินไปมีแสงฟุ้งที่ขอบวัตถุ ซึ่งโดยธรรมชาติแล้วตัวรับแสงแบบ CMOS จะปรากฏอาการมากกว่า CCD ดังนั้น จำเป็นต้องการการออกแบบทางวิศวกรรมเพิ่มเติมอีกเพื่อลดปัญหานี้ ซึ่งใน CCD จะมีต่ำกว่า

การใช้งาน

ตัวรับแสงทั้งสองชนิดนี้ต่างก็ได้รับความนิยมในการใช้งานทั้งสองอย่าง แต่ในสภาพแวดล้อมที่สมบุกสมบันแล้วจะพบว่าเซ็นเซอร์ CMOS จะมีความได้เปรียบอยู่ส่วนหนึ่งเนื่องมาจากการที่ได้รวมวงจรการแปลงสัญญาณต่างๆ เข้าไว้ด้วยกันในชิพตัวเดียวซึ่งลดการเชื่อมต่อสัญญาณต่างๆ ได้มากกว่า CCD อีกประการหนึ่งคือการที่ CMOS ได้รวมวงจรต่างๆ เข้าไว้ด้วยกันจึงเป็นการลดขนาดของกล้องไปในเวลาเดียวกัน ซึ่งจะพบได้ว่ากล้องที่ใช้ CMOS ส่วนใหญ่จะมีขนาดที่เล็กกะทัดรัดกว่ากล้องที่ใช้ CCD และใช้พลังงานที่ต่ำกว่า ที่สำคัญยิ่งก็คือเมื่อมีการผลิตเป็นจำนวนมากๆ แล้วจะมีต้นทุนที่ต่ำกว่า CCD มาก

อย่างไรก็ตาม CCD เซ็นเซอร์ ก็มีจุดเด่นที่ให้สัญญาณที่สม่ำเสมอกว่า ขนาดพิกเซลที่รับแสงที่ใหญ่กว่าสามารถออกแบบให้เต็มพิกเซลได้ มีช่วงกว้างในการรับแสงที่กว้างกว่านอยซ ที่ต่ำกว่า ซึ่งจะให้คุณภาพที่ดีกว่า แต่ต้นทุนการผลิตนั้นก็ย่อมสูงกว่าการใช้ CMOS โดยเฉพาะอย่างยิ่งที่ต้องมีแผงวงจรถึงสองแผงแทนที่จะใช้เป็นชิ้นเดียวแบบ CMOS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ความรู้พื้นฐานของการประมวลผลภาพกราฟฟิก

2.2.1 ความรู้เรื่องความละเอียด

พิกเซล (Pixel)

เป็นคำผสมของคำว่า Picture กับคำว่า Element หรือหน่วยพื้นฐานของภาพ เทียบได้กับ "จุดภาพ" 1 จุด แต่ละพิกเซลเปรียบได้กับสี่เหลี่ยมเล็กๆ ที่บรรจุค่าสี โดยถูกกำหนดตำแหน่งไว้บนเส้นกริดของแนวแกน x และแกน y หรือในตารางเมตริกซ์สี่เหลี่ยม ภาพบิตแมปจะประกอบด้วยพิกเซลหลายๆ พิกเซล

รูปที่ 2.2 พิกเซลของภาพเฉดสีขาว/ดำ และของภาพสี

จำนวนพิกเซลของภาพแต่ละภาพ จะเรียกว่า ความละเอียด (Resolution) โดยจะเทียบจำนวนพิกเซลกับความยาวต่อนิ้ว ดังนั้นจะมีหน่วยเป็น พิกเซลต่อนิ้ว (ppi: pixels per inch) หรือจุดต่อนิ้ว (dpi; dot per inch) ภาพขนาดเท่านั้นแต่มีความละเอียดต่างกัน แสดงว่าจำนวนพิกเซลต่างกัน และขนาดของจุดพิกเซลก็ต่างกันด้วย

ความละเอียดในการแสดงผล (Resolution)

คำนี้สามารถใช้ได้กับสถานการณ์ที่แตกต่างกัน เช่น ความละเอียดของการแสดงผลของเครื่องพิมพ์ หรือความละเอียดในการแสดงผลของจอภาพ ดังนั้นความละเอียดในการแสดงผลจึงหมายถึง จำนวนหน่วยต่อพื้นที่

ความละเอียดของรูปภาพ

หมายถึง จำนวนพิกเซลต่อพื้นที่การแสดงผล มีหน่วยเป็นพิกเซลต่อนิ้ว (pixels per inch - ppi) โดยพิกเซลจะมีขนาดไม่แน่นอนขึ้นกับอุปกรณ์เอาต์พุต เช่น หน้าจอคอมพิวเตอร์ หรือเครื่องพิมพ์แบบเลเซอร์ ถ้ารูปภาพมีความละเอียด 300 จุดต่อนิ้ว เครื่องพิมพ์แบบเลเซอร์สามารถพิมพ์ได้ 300 จุดต่อนิ้ว นั่นคือเครื่องพิมพ์แบบเลเซอร์จะใช้ 1 จุดสำหรับแต่ละพิกเซลของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความละเอียดของจอภาพ

หมายถึง หน่วยของจำนวนจุดที่มากที่สุดที่จอคอมพิวเตอร์สามารถผลิตได้ โดยความละเอียดในการแสดงผลของจอ จะขึ้นกับวีดีโอการ์ด ที่เรียกว่าการ์ดจอ ซึ่งจะมีความสามารถในการแสดงผลหลากหลาย เช่น แสดงผลที่ความละเอียด 800 x 600 พิกเซล หมายถึง จำนวนพิกเซลในแนวนอน เท่ากับ 800 และจำนวนพิกเซลในแนวตั้ง เท่ากับ 600

2.2.2 ความรู้เรื่องบิต

บิต หมายถึง หน่วยความจำที่เล็กที่สุดของคอมพิวเตอร์ ประกอบด้วยตัวเลข 2 จำนวน คือ 0 หมายถึงปิด และ 1 หมายถึงเปิด หรือสีขาวและสีดำ

ความลึกของบิต (Bit Depth)

ความลึกของบิต หมายถึง จำนวนบิตที่ใช้ในแต่ละพิกเซล ในกราฟิกแบบบิตแมปสีของพิกเซลถูกบันทึกโดยใช้บิต ถ้าใช้สีมากก็แสดงสีได้มากขึ้น ถ้ามีหน่วยความจำ 2 บิต ในการเก็บรวบรวมข้อมูล สามารถใช้สีได้ทั้งหมด 2 เท่ากับ 4 สี คือ สามารถกลับสีได้ 4 วิธี คือ 00, 01,10 และ 11 ถ้ามี 2 บิต สามารถสร้างสีให้กับพิกเซลทั้งหมด 4 เฉด

ตารางที่ 2.1 แสดงสีที่แสดงได้ในแต่ละจำนวนบิตข้อมูลต่อพิกเซล

จำนวนบิตเก็บข้อมูล/ พิกเซล	สีที่ได้
1 บิต	2^1 =2 สี
2 บิต	2^2 = 4 สี
4 บิต	2^4 = 16 สี
8 บิต	2^8 = 256 สี
24 บิต	2^{24} = 16.7 ล้านสี

เราเรียกสีแบบ 24 บิต ว่าเป็นสีเหมือนจริง (True Color) ในการแก้ไขภาพ ในระดับมืออาชีพอาจจะมีการใช้สีถึง 36,48,64 บิต แต่ในคอมพิวเตอร์ทั่วๆ ไปมักจะใช้สีไม่เกิน 32 บิต

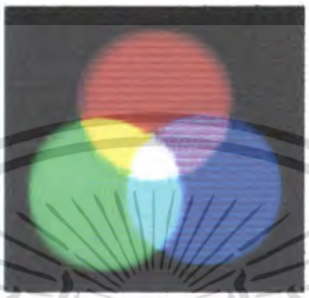
2.2.3 ระบบสี (Color Model)

ระบบสี Additive

ระบบสี Additive ปกติเมื่อพูดถึงสี มักจะนึกถึงแม่สี 3 สีแต่อย่างไรก็ตาม การใช้สีกับงานกราฟิกในคอมพิวเตอร์ มีรายละเอียดหลายประการ ซึ่งเป็นลักษณะเฉพาะ ดังนั้นจึงควรทราบระบบสีของคอมพิวเตอร์ก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบสีของคอมพิวเตอร์ จะเกี่ยวข้องกับการแสดงผลแสงที่แสดงบนจอคอมพิวเตอร์ โดยมีลักษณะการแสดงผล คือ ถ้าไม่มีแสดงผลสีใดเลย บนจอภาพจะแสดงเป็น "สีดำ" หากสีทุกสีแสดงผลพร้อมกัน จะเห็นสีบนจอภาพเป็น "สีขาว" ส่วนสีอื่นๆ เกิดจากการแสดงสีหลายๆ สี แต่มีค่าแตกต่างกัน การแสดงผลลักษณะนี้ เรียกว่า การแสดงสีระบบ Addivtive



รูปที่ 2.3 การแสดงสีระบบ Addivtive

สีในระบบ Additive ประกอบด้วยสีหลัก 3 สี (เช่นเดียวกับแม่สี) คือ สีแดง (Red) สีเขียว (Green) และ สีน้ำเงิน (Blue) เรียกรวมกันว่า RGB ซึ่งมีรูปแบบการผสมสีของ RGB ดัง

การผสมกันของแม่สีทั้งสาม

1. ถ้าแม่สีมีค่าเท่ากัน มาผสมกันเป็นคู่ จะได้ผลดังนี้

1 Red	+1 Blue	= Magenta
1 Blue	+1 Green	=Cyan
1 Green	+1 Red	=Yellow

2. ถ้าแม่สีมีค่าเท่ากัน มาผสมทั้ง 3 สี จะได้ผลดังนี้

1 Red	+1Blue	+1 Green	=White

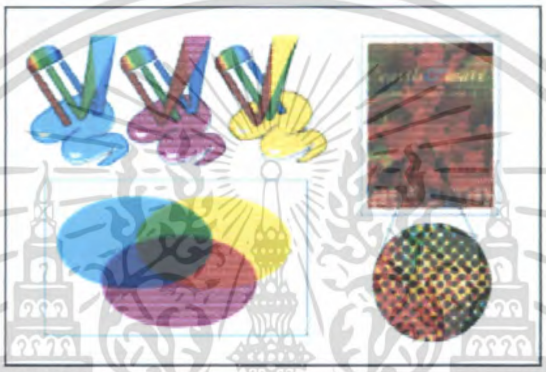
3. ถ้าแม่สีมีค่าไม่เท่ากัน ผสมกัน จะได้สีต่างๆ กันไป เช่น

2 Red	+1 Green	=Orange	
2 Green	+1 Red	=Lime	
1 Blue	+1 Green	+4 Red	=Brown

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบสี Subtractive

ระบบสี Subtractive มีลักษณะที่ตรงข้ามกับ Additive โดยสีแต่ละสีจะได้จากการลบสีต่างๆ ออกไปจากระบบ ดังนั้น หากไม่มีการแสดงสีใดๆ จะแสดงผลเป็นสีขาว ขณะที่การแสดงสีทุกสี จะปรากฏเป็นสีดำ และสีหลัก หรือแม่สีของระบบนี้ จะประกอบด้วย สีฟ้า (Cyan) สีม่วงแดง (Magenta) และสีเหลือง (Yellow) หรือระบบ CMY เป็นระบบสีที่ใช้กับงานสิ่งพิมพ์ ซึ่งมักจะรวมเอาสีดำ มาเป็นแม่สีด้วย จึงเรียกว่าระบบ CMYK (Cyan, Magenta, Yellow, Black) นั่นเอง



รูปที่ 2.4 การแสดงสีระบบ Subtractive

ตัวอย่างการผสมสี

CYAN INK	MAGENTA INK	YELLOW INK	BLACK INK	ALL INKS
100% CYAN		100% YELLOW		GREEN
100% CYAN		100% YELLOW	25% BLACK	DARK GREEN
	100% MAGENTA	100% YELLOW		RED
	100% MAGENTA	100% YELLOW	25% BLACK	DARK RED
100% CYAN	50% MAGENTA			BLUE
50% CYAN	75% MAGENTA	100% YELLOW		BROWN
45% CYAN	30% MAGENTA	30% YELLOW		GRAY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 รูปแบบของไฟล์

ไฟล์กราฟฟิก ปัจจุบันมี 3 ไฟล์หลัก ๆ คือ

1. ไฟล์สกุล GIF (Graphics Interlace File)
2. ไฟล์สกุล JPG (Joint Photographer's Experts Group)
3. ไฟล์สกุล PNG (Portable Network Graphics)

ไฟล์สกุล GIF (Graphics Interlace File)

เป็นไฟล์กราฟฟิกมาตรฐานที่ทำงานบนอินเทอร์เน็ต มักจะใช้เมื่อ

1. ต้องการไฟล์ที่มีขนาดเล็ก
2. จำนวนสีและความละเอียดของภาพไม่สูงมากนัก
3. ต้องการพื้นแบบโปร่งใส
4. ต้องการแสดงผลแบบโครงร่างก่อน แล้วค่อยแสดงผลแบบละเอียด
5. ต้องการนำเสนอภาพแบบภาพเคลื่อนไหว

จุดเด่น

1. มีขนาดไฟล์ต่ำ
2. สามารถทำพื้นของภาพให้เป็นพื้นแบบโปร่งใสได้ (Transparent)
3. มีระบบแสดงผลแบบหยาบและค่อยๆ ขยายไปสู่ละเอียดในระบบ Interlace
4. มีโปรแกรมสนับการสร้างจำนวนมาก
5. เรียกดูได้กับกราฟฟิกเบราว์เซอร์ (Graphics Browser) ทุกตัว
6. ความสามารถด้านการนำเสนอแบบภาพเคลื่อนไหว (Gif Animation)

จุดด้อย

แสดงสีได้เพียง 256 สี

ไฟล์สกุล JPG (Joint Photographer's Experts Group)

เป็นอีกไฟล์หนึ่งที่นิยมใช้บนอินเตอร์เน็ตมักใช้กรณี

1. ภาพที่ต้องการนำเสนอมีความละเอียดสูง และใช้สีจำนวนมาก (สนับสนุนถึง 24 บิตสี)
2. ต้องการบีบไฟล์ตามความต้องการของผู้ใช้
3. ไฟล์ชนิดนี้มักจะใช้กับภาพถ่ายที่นำมาสแกน และต้องการนำไปใช้บนอินเทอร์เน็ต เพราะให้ความคมชัดและความละเอียดของภาพสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดเด่น

1. สนับสนุนสีได้ถึง 24 บิต
2. สามารถกำหนดค่าการบีบไฟล์ได้ตามที่ต้องการ
3. มีระบบแสดงผลแบบหยาบและค่อยๆ ขยายไปสู่ละเอียดในระบบโพรเกรสซีฟ (Progressive)
4. มีโปรแกรมสนับสนุนการสร้างจำนวนมาก
5. เรียกดูได้กับกราฟฟิกเบราว์เซอร์ (Graphics Browser) ทุกตัว
6. ตั้งค่าการบีบไฟล์ได้ (compress files)

จุดด้อย

ทำให้พื้นของรูปโปร่งใสไม่ได้

ข้อเสียของการบีบไฟล์ (Compress File)

กำหนดค่าการบีบไฟล์ไว้สูง (1 - 10) แม้ว่าจะช่วยให้ขนาดของไฟล์มีขนาดต่ำ แต่ก็มีข้อเสีย คือ เมื่อมีการส่งภาพจากแม่ข่ายไปแสดงผลที่ลูกข่ายจะทำให้การแสดงผลช้ามาก เพราะต้องเสียเวลาในการคลายไฟล์ ดังนั้นการเลือกค่าการบีบไฟล์ ควรกำหนดให้เหมาะสมกับภาพแต่ละภาพ

ไฟล์สกุล PNG (Portable Network Graphics)

จุดเด่น

1. สนับสนุนสีได้ถึงตามค่าสีเสมือนจริง (True color) (16 บิต, 32 บิต หรือ 64 บิต)
2. สามารถกำหนดค่าการบีบไฟล์ได้ตามที่ต้องการ
3. มีระบบแสดงผลแบบหยาบและค่อยๆ ขยายไปสู่ละเอียด (Interlace)
4. สามารถทำพื้นโปร่งใสได้

จุดด้อย

1. หากกำหนดค่าการบีบไฟล์ไว้สูง จะใช้เวลาในการคลายไฟล์สูงตามไปด้วย แต่ขนาดของไฟล์จะมีขนาดต่ำ
2. ไม่สนับสนุนกับกราฟฟิกเบราว์เซอร์ (Graphic Browser) รุ่นเก่า สนับสนุนเฉพาะ IE 4 และ Netscape 4
3. ความละเอียดของภาพและจำนวนสีขึ้นอยู่กับวีดีโอการ์ด (Video Card)
4. โปรแกรมสนับสนุนในการสร้างมีน้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
รายละเอียดเกี่ยวกับโมดูลกล้อง

3.1 โมดูลกล้อง E700

เป็นโมดูลกล้องที่ใช้ในภาคเรียนแรก เป็นกล้องขนาด 640×480 พิกเซล มีการเชื่อมต่อข้อมูลดิจิตอลแบบ 8 บิต ใช้เซ็นเซอร์ตรวจจับภาพแบบ CMOS เบอร์ HV7131GP ของ Hynix เป็นโมดูลกล้องที่ใช้ในมือถือยี่ห้อ Sumsung E700 มีรูปแบบของสัญญาณภาพที่หลากหลาย เช่น YCbCr และ RGB ผู้ใช้สามารถกำหนดรีจิสเตอร์เพื่อกำหนดค่าการทำงานของกล้อง โดยผ่านการเชื่อมต่อแบบ I^2C สามารถส่งสัญญาณภาพได้ด้วยความเร็วสูงสุด เท่ากับ 30 เฟรมต่อวินาที ที่ความถี่ 25 เมกกะเฮิร์ซ

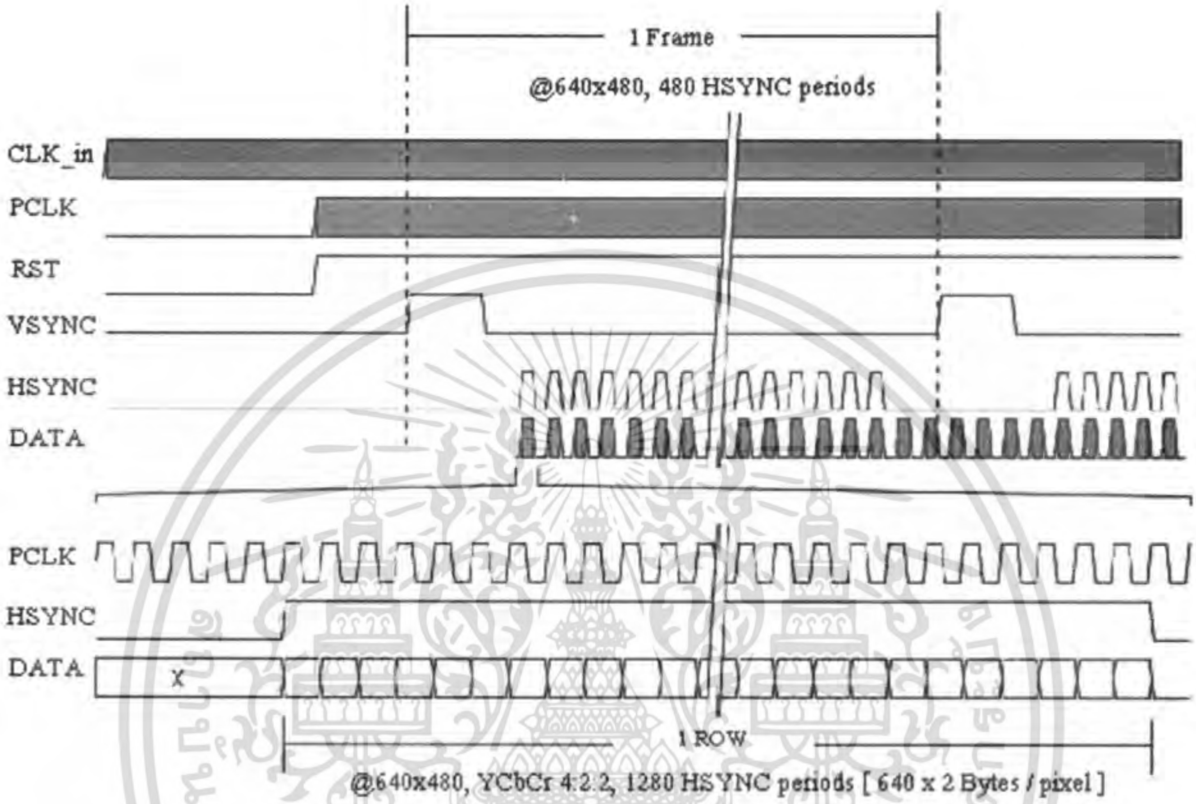
รูปที่ 3.1 โมดูลกล้องและการจัดเรียงขาของกล้อง E700

ตารางที่ 3.1 แสดงขาต่างๆของกล้อง

Pin	Description		Pin	Description
1	Y0 [out]	Data Bus	11	ENB [in] - Enable Camera
2	Y1 [out]		12	/RST [in] - Reset Camera
3	Y2 [out]		13	GND
4	Y3 [out]		14	VCC
5	Y4 [out]		15	SCL [inout] - I2C Clock
6	Y5 [out]		16	SDA [inout] - I2C Data
7	Y6 [out]		17	Vsync [out] - Vertical Sync
8	Y7 [out]		18	Hsync [out] - Horizontal Sync
9	VCC		19	PCLK [out] - Pixel Clock
10	GND		20	CLK_in [in] - Clock Input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 สัญญาณเอาท์พุต



รูปที่ 3.2 แสดงสัญญาณเอาท์พุตต่างๆของโมดูลกล้อง

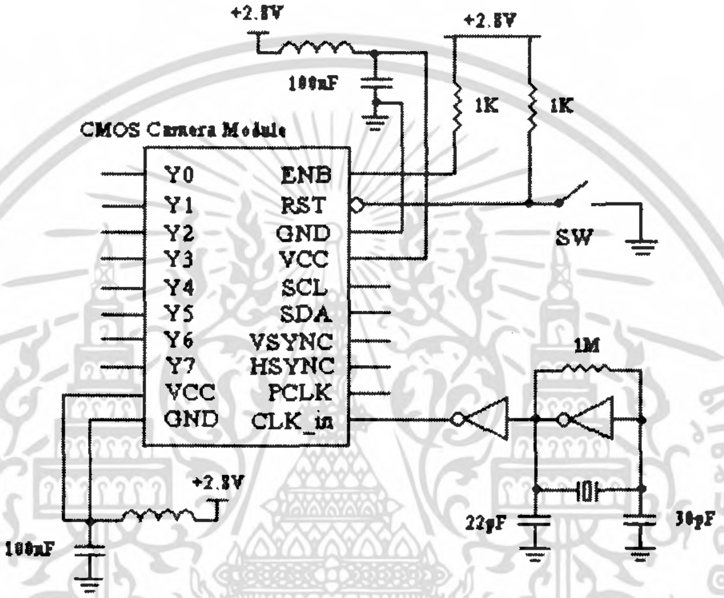
จากรูป จะเห็นว่าข้อมูลจะถูกส่งทีละไบต์ทุกพัลส์ของ PCLK สัญญาณ PCLK ไม่จำเป็นต้องมีเฟสหรือความถี่ที่เท่ากับสัญญาณ CLK_in โดยการเลื่อนของเฟสนั้นจะขึ้นอยู่กับความถี่ที่ป้อนและความถี่ของ PCLK สามารถปรับได้ผ่านการเชื่อมต่อแบบ I^2C โดยใช้ไมโครคอนโทลเลอร์ สัญญาณ HSYNC และข้อมูลจะถูกอ่านทุกๆขอบขาลงของ PLCK ซึ่งในขณะนั้นสัญญาณ HSYNC จะต้องมีสถานะเป็น high ก็จะทำให้ได้สัญญาณภาพ 1 แถวในแนวนอน ส่วนสัญญาณ VSYNC จะเป็นตัวกำหนดจำนวนแถวทั้งหมด เพื่อให้ได้ภาพ 1 เฟรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

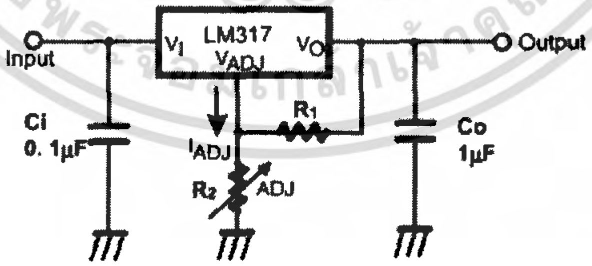
3.1.2 วงจรที่ใช้กับโมดูลกล้อง

แรงดันไฟเลี้ยงของโมดูลกล้องจะต้องมีขนาดตั้งแต่ 2.6 – 3.0 โวลต์ จึงเลือกในใช้วงจรแรงดันคงที่ โดยใช้ LM 317 เปลี่ยนไฟ 5 โวลต์ ให้มีขนาด 2.8 โวลต์ เพื่อป้อนให้โมดูลกล้อง

ส่วนวงจรกำเนิดความถี่ที่ป้อนให้กับโมดูลกล้อง จะใช้ ไอซี Hex inverter เบอร์ 74HC04 ต่อกับตัวคริสตัลดังรูป โดยจะเลือกใช้คริสตัลที่มีความถี่เท่ากับ 4 เมกกะเฮิร์ซ



รูปที่ 3.3 การป้อนไฟเลี้ยง และความถี่ให้กับโมดูลกล้อง



$$V_O = 1.25V\,(1 + R_2 / R_1) + I_{ADJ}\,R_2$$

รูปที่ 3.4 การสร้างแรงดันไฟเลี้ยงของวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 การอินเตอร์เฟสโมดูลกล้องกับคอมพิวเตอร์

การอินเตอร์เฟสจะใช้ไมโครคอนโทรลเลอร์ AVR ในการรับข้อมูลรหัสสี แล้วนำข้อมูลที่ได้ไปแสดงผลบนคอมพิวเตอร์ โดยใช้การสื่อสารแบบอนุกรมโดยใช้โมดูล UART เชื่อมต่อกับคอมพิวเตอร์ และใช้โปรแกรมไฮเปอร์เทอมินอลในการแสดงผลข้อมูล

3.1.4 หลักการเขียนโปรแกรมเพื่อรับข้อมูลภาพ

รูปแบบของข้อมูลภาพจากกล้องจะเป็นแบบ YCbCr โดยมีลำดับการส่งข้อมูล ดังนี้

Y[7:0]	Y	Cb	Y	Cr	Y	Cb	Y	Cr

ข้อมูลไบต์แรกที่ส่งออกมาจะเป็น ไบต์ Y และจะเห็นว่า ทุก 4 ค่าความเข้มข้นของแสงสว่าง จะประกอบไปด้วย Cb และCr อย่างละ 2 ไบต์

เนื่องจาก 1 พิกเซล ประกอบด้วย Y Cb และ Cr ดังนั้นจะต้องมีการใช้ Cb และCr ร่วมกัน กล่าวคือ ทุกๆ 2 พิกเซล จะต้องใช้ข้อมูลจำนวน 4 ไบต์ ได้แก่ Y Cb Y และ Cr ตามลำดับ

ในการเขียนโปรแกรมเพื่อรับข้อมูลภาพ จะรับเอาเฉพาะข้อมูลที่เป็นไบต์ Y เท่านั้น ซึ่งจะทำให้ได้ข้อมูลภาพที่เป็นโทนสีเทา

ดังนั้น ส่วนหลักๆของโปรแกรมก็จะเป็นการใช้คำสั่งเกี่ยวกับโมดูล UART และการใช้ฟังก์ชันการอินเตอร์รัปต์ในการจับสัญญาณ PCLK และ VSYNC เนื่องจาก จะต้องใช้ส่วนที่เป็นขอบขาลงและขอบขาขึ้นของสัญญาณ ในการควบคุมการรับข้อมูลภาพ

ข้อมูลภาพที่ได้จะมีขนาดเพียงส่วนหนึ่งของภาพเท่านั้น โดยมีขนาด 40x20 จุด เพื่อจะได้นำข้อมูลที่ได้ไปพล็อตเป็นภาพต่อไป

3.1.5 บทสรุปของการใช้กล้อง E700

เนื่องจากข้อมูลภาพที่ออกมาจากโมดูลกล้องเป็นข้อมูลแบบขนาน และการทำงานของไมโครคอนโทรลเลอร์ที่ใช้ ไม่ทันกับข้อมูลที่ส่งมาและข้อมูลที่ได้ก็มีความผิดพลาด ดังนั้น เราจึงทำการเปลี่ยนกล้องในการทำโครงงาน มาเป็นกล้อง C328R ซึ่งมีการส่งข้อมูลเป็นแบบอนุกรม

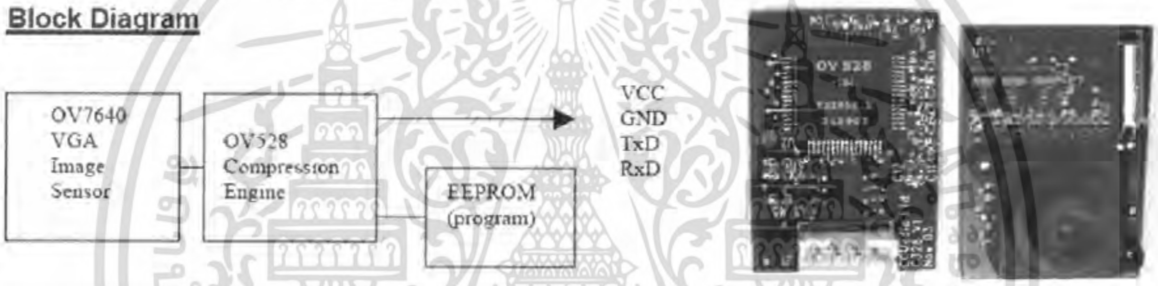
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 โมดูลกล้อง C328R

กล้อง C328R เป็นกล้อง VGA ที่ให้ข้อมูลภาพเป็น เจเป็ก (JPEG) สามารถเชื่อมต่อแบบไร้สาย (wireless) หรือ PDA กล้องรุ่นนี้ สามารถถ่ายภาพได้ทั้งแบบภาพนิ่งและภาพเคลื่อนไหว ขึ้นอยู่กับคำสั่งที่เลือกใช้

ลักษณะสำคัญ

- ขนาดเล็ก 20x28 มิลลิเมตร
- ราคาถูก ใช้กำลังไฟต่ำ
- เชื่อมต่อโดยใช้ UART มีอัตราบอดเรทสูงสุด 115.2 บิตต่อวินาที



รูปที่ 3.5 บล็อกไดอะแกรมของกล้อง C328R

ระบบของกล้องประกอบด้วย 3 ส่วน ได้แก่

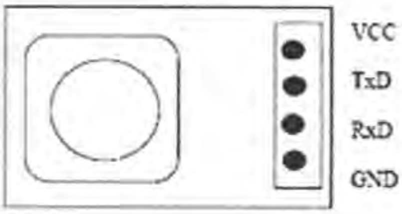
1. เซ็นเซอร์ของกล้อง (Camera Sensor) เป็นชิพ CMOS OV7640 ส่งข้อมูลเป็นแบบ YCbCr ขนาด 8 บิต ให้กับชิพควบคุม

2. OV528 Serial Bridge เป็นชิพควบคุม มีหน้าที่ส่งข้อมูลภาพจากชิพของกล้องไปสู่อุปกรณ์ภายนอก นอกจากนี้ ก็ยังทำหน้าที่เปลี่ยนแปลงระบบสีและคุณภาพของภาพ

3. โปรแกรม EEROM เป็นหน่วยความจำโปรแกรมภายใน C328R

Pin Description

Pin	Description
VCC	Power 3.3VDC
TxD	Data Transmit (3.3V)
RxD	Data Receive (3.3V)
GND	Power Ground



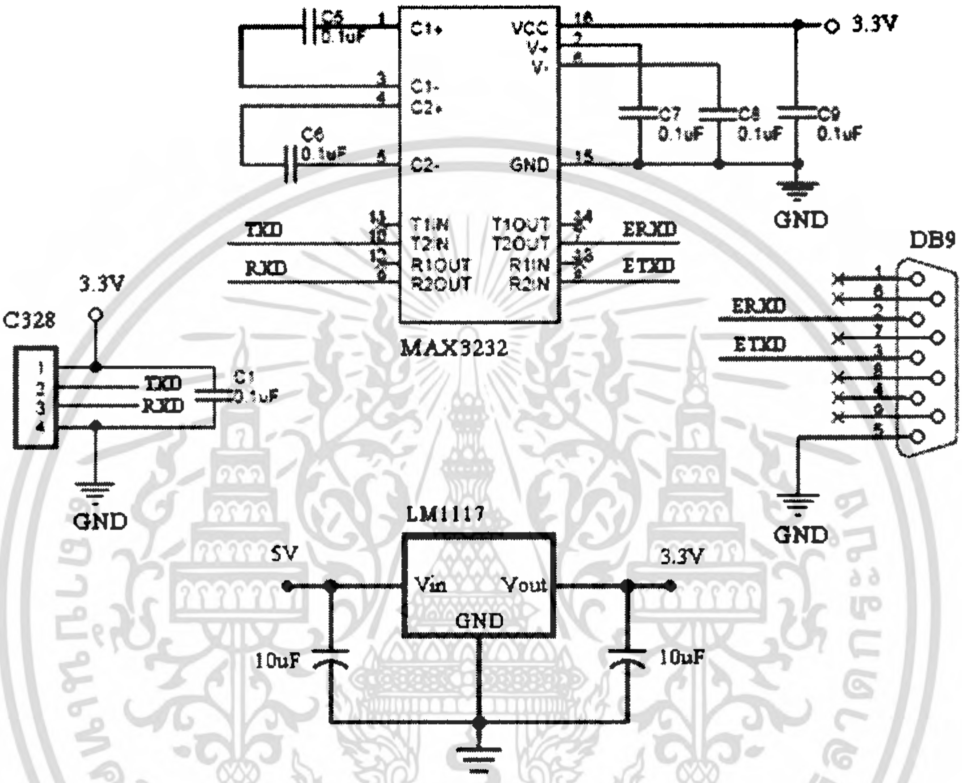
รูปที่ 3.6 ขาสัญญาณต่างๆของกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การทดสอบรูปที่ได้จากกล้อง

เราสามารถทำการทดสอบรูปที่ได้จากกล้อง โดยการนำกล้องไปเชื่อมต่อกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม ซึ่งวงจรที่ใช้ในการเชื่อมต่อเป็นดังนี้



รูปที่ 3.7 วงจรที่ใช้เชื่อมต่อ C328R กับคอมพิวเตอร์

ในวงจรจะประกอบไปด้วย ไอซี LM1117 มีหน้าที่สร้างแรงดันไฟเลี้ยง 3.3 โวลต์ ให้กับกล้อง C328R และไอซี MAX3232 ที่ทำหน้าที่ส่งผ่านข้อมูลจากกล้องไปยังพอร์ต RS-232 ของคอมพิวเตอร์

82205

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอฟแวร์ที่ใช้ เป็นซอฟแวร์ที่ถูกพัฒนาขึ้นโดยบริษัท COMedia ของฮ่องกง โปรแกรมนี้เป็นโปรแกรมที่ใช้ถ่ายภาพ โดยที่ผู้ใช้สามารถเลือกขนาดของภาพตามที่ต้องการได้ ได้แก่ ขนาด 640x480, 320x240, 160x128, 128x128, 128x96 และ 80x60 ภาพที่ได้จะเป็นไฟล์นามสกุล .jpg

รูปที่ 3.8 หน้าตาของโปรแกรมถ่ายภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ไมโครคอนโทรลเลอร์ AVR

4.1 รายละเอียดเกี่ยวกับ AVR

ไมโครคอนโทรลเลอร์ AVR เป็นไมโครคอนโทรลเลอร์ที่ได้รับความนิยมมาก ด้วยความสามารถบวกกับราคาที่ไม่สูงมากนัก ผนวกกับเครื่องมือในการพัฒนาโปรแกรมที่เรียกว่า AVR Studio ที่สามารถทำงานร่วมกับ WinAVR (C Complier) ซึ่งเป็นซอฟต์แวร์ฟรี

ไมโครคอนโทรลเลอร์ AVR เป็นไมโครคอนโทรลเลอร์ ที่ผลิตโดยบริษัท ATMEL (ผู้นำทางด้านไมโครคอนโทรลเลอร์ตระกูล MCS-51) AVR จัดเป็นไมโครคอนโทรลเลอร์ตระกูลใหม่จาก ATMEL มีสถาปัตยกรรมแบบ RISC (Advanced RISC architecture) คือหนึ่งคำสั่งทำงานใช้สัญญาณนาฬิกาเพียงหนึ่งลูก (instruction in a single clock cycle)

คุณสมบัติของบอร์ดไมโครคอนโทรลเลอร์

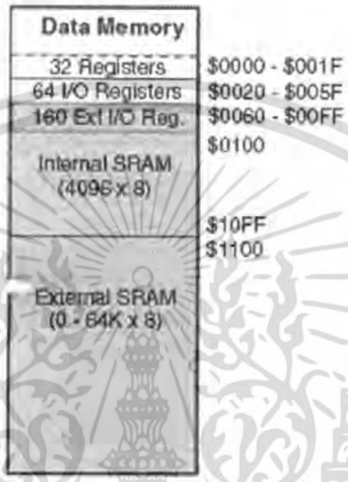
เลือกใช้ AVR เบอร์ ATmaga128 เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต ใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบ XTAL ค่า 16 เมกกะเฮิร์ซ ซึ่งคุณสมบัติเด่นของไมโครคอนโทรลเลอร์ ได้แก่

- มีหน่วยความจำ แฟลช (Flash) สำหรับเขียนโปรแกรม 128 กิโลไบต์
- มีหน่วยความจำข้อมูลถาวรแบบ EEPROM ขนาด 4 กิโลไบต์
- จำนวน I/O สูงสุดถึง 53 I/O ขา
- มีวงจรสื่อสาร SPI จำนวน 1 ช่อง, I^2C จำนวน 1 ช่อง, Programmable Serial USARTs จำนวน 2 ช่อง
- มี ADC ขนาด 10 บิต จำนวน 8 ช่อง
- มี Timer /Counters 8 บิต และ 16 บิต จำนวน 2 ช่อง , 8 บิต PWM จำนวน 2 ช่อง , Watchdog Timer , Real Time Counter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

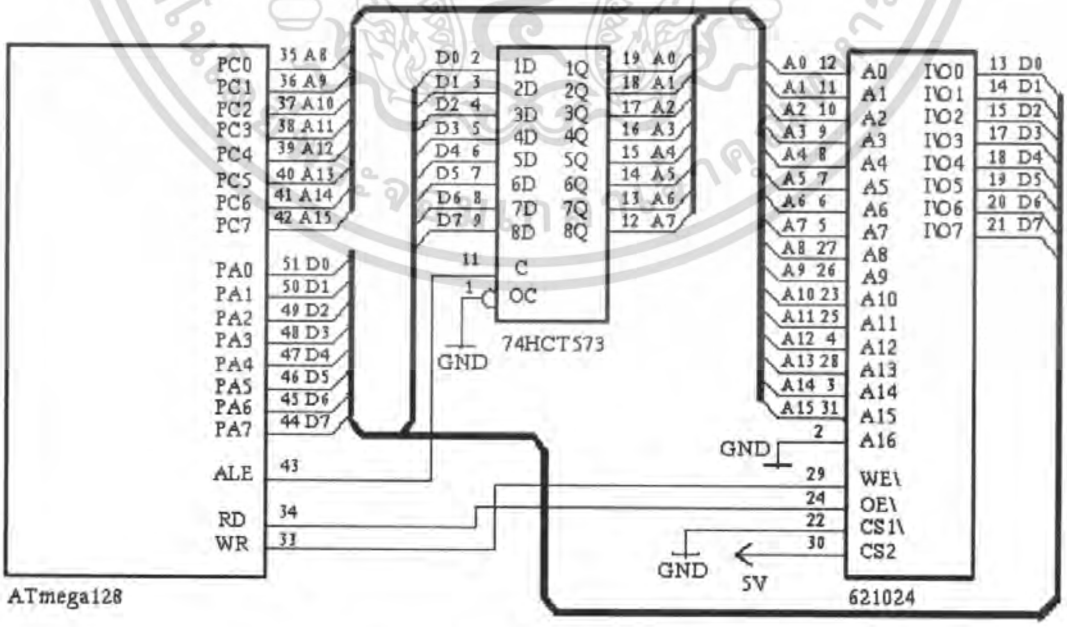
4.2 การต่อแรมให้กับ AVR

เนื่องจาก SRAM ภายในของ ATmega128 มีขนาด 4 กิโลไบต์ แต่โปรเจคนี้ต้องการใช้แรมในการแสดงผลข้อมูลภาพอย่างน้อย 17 กิโลไบต์ จึงมีความจำเป็นที่จะต้องต่อแรมภายนอก ซึ่ง ATmega128 สามารถจะเพิ่มแรมได้สูงสุด เท่ากับ 64 กิโลไบต์



รูปที่ 4.1 การแบ่งส่วนของหน่วยความจำภายใน AVR

จากรูปข้างบน แรมภายใน จะมีแอดเดรสตั้งแต่ 0x0100 ถึง 0x10FF ส่วนแอดเดรสของแรมภายนอกจะเริ่มจาก 0x1100 ไปจนถึง 0xFFFF

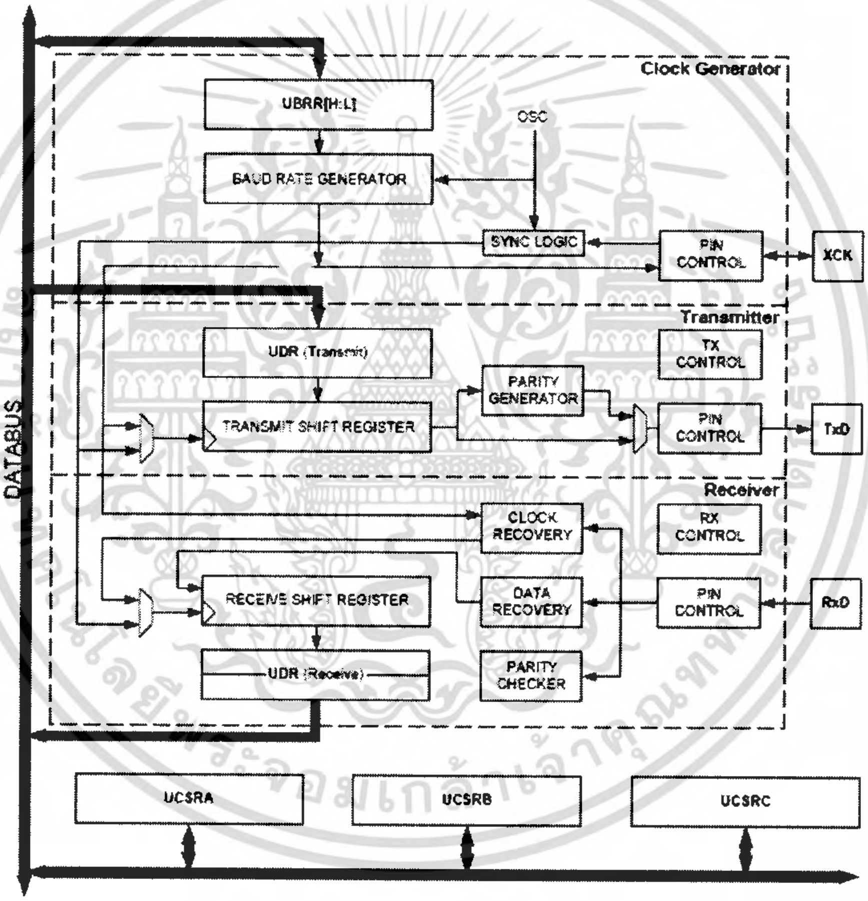


รูปที่ 4.2 วงจรการต่อแรมเพิ่มให้กับ AVR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การสื่อสารข้อมูลอนุกรมผ่านโมดูล USART

ไมโครคอนโทรลเลอร์ AVR สามารถที่จะสื่อสารข้อมูลอนุกรมได้โดยการใช้โมดูล USART (Universal Synchronous and Asynchronous serial Receiver and Transmitter) เพื่อสื่อสารข้อมูลผ่านทางพอร์ตอนุกรมได้ทั้งแบบซิงโครนัส (ข้อมูลมีความต่อเนื่อง มีการกำหนดสัญญาณมาตรฐานที่เหมือนกันทางด้านรับและด้านส่ง เพื่อให้การรับส่งมีความสัมพันธ์กัน) และอะซิงโครนัส (ข้อมูลไม่จำเป็นต้องต่อเนื่องมีบิตเริ่มต้น (Start bit) บิตข้อมูล (Data bit) และบิตหยุด (Stop bit) มีบิตพาริตี้ (Parity bit) หรือไม่มีก็ได้



รูปที่ 4.3 บล็อกไดอะแกรมโมดูล USART

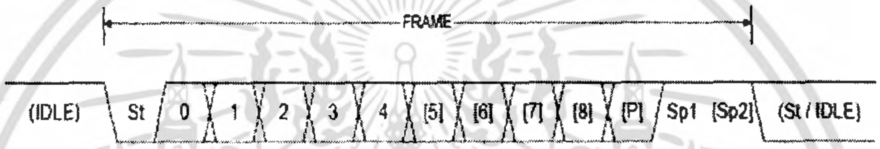
จากรูปจะเห็นว่าโมดูล USART แบ่งออกเป็น 3 ส่วน คือ

1. ส่วนสร้างสัญญาณนาฬิกา (Clock Generator) เพื่อใช้กำหนดอัตราบอดในการรับส่งข้อมูล โดยสามารถกำหนดได้ทั้งภายในและภายนอก ผ่านทางขา XCK
2. ส่วนส่งข้อมูลอนุกรม (Transmitter) โดยส่งข้อมูลออกทางขาพอร์ต TxD
3. ส่วนรับข้อมูลอนุกรม (Receiver) โดยการรับข้อมูลจากขาพอร์ต RxD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และมีรีจิสเตอร์ควบคุมการทำงาน 3 ตัว ประกอบไปด้วย UCSRA, UCSRB และUCSRC การส่งข้อมูลอนุกรมในรูปแบบอะซิงโครนัส จะเป็นการส่งข้อมูลเป็นเฟรม ลักษณะของเฟรมข้อมูลอนุกรมนี้ประกอบไปด้วย

1. บิตเริ่มต้นข้อมูล (Start bit)
2. บิตข้อมูล (Data bit)
3. พาริตี้บิต (Parity bit)
4. บิตหยุดข้อมูล (Stop bit)



St	Start bit, always low.
(n)	Data bits (0 to 8).
P	Parity bit. Can be odd or even.
Sp	Stop bit, always high.
IDLE	No transfers on the communication line (RxD or TxD). An IDLE line must be high.

รูปที่ 4.4 รูปแบบเฟรมข้อมูลอนุกรม

คุณสมบัติที่สำคัญของโมดูล USART มีดังนี้

1. การสื่อสารแบบฟูลดูเพลกซ์ (Full Duplex) ตัวรับและตัวส่งแยกอิสระต่อกัน สามารถรับส่งข้อมูลได้ในเวลาเดียวกัน

2. ทำงานได้ทั้งในโหมดซิงโครนัสและอะซิงโครนัส

3. มีคุณสมบัติของพอร์ตอนุกรมครบถ้วน เช่น การกำหนดบิตข้อมูล การกำหนดบิตหยุด และการกำหนดพาริตี้ เป็นต้น

4. มีส่วนตรวจสอบความผิดพลาดของเฟรมข้อมูลและข้อมูลโอเวอร์รัน (Framing Error and Data Overrun Detection)

5. โหมดการสื่อสารแบบมัลติโปรเซสเซอร์

6. โหมดทวีคูณความเร็วในการสื่อสารข้อมูลแบบอะซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสื่อสารข้อมูลอนุกรมโหมดอะซิงโครนัสเป็นการสื่อสารข้อมูลระหว่างไมโครคอนโทรลเลอร์กับพอร์ตอนุกรมหรือที่นิยมนำไมโครคอนโทรลเลอร์มาเชื่อมต่อกับคอมพิวเตอร์ผ่านพอร์ตอนุกรม (RS-232) นอกจากการกำหนดจำนวนบิตข้อมูล บิตหยุด และพาริตี้แล้ว จะต้องมีการกำหนดอัตราเร็วในการรับส่งข้อมูลที่เรียกว่า อัตราบอตหรือบอตเรต (baud rate) หรือการเปลี่ยนแปลงของสัญญาณใน 1 วินาที

ตารางที่ 4.1 การคำนวณหาอัตราบอตเรต

โหมดการทำงาน	การคำนวณหาอัตราบอตเรต	การคำนวณหาค่า UBRR
โหมดอะซิงโครนัสปกติ(U2X=0)	BAUD = fosc/16(UBRR+1)	UBRR=(fosc/16BAUD)-1
โหมดอะซิงโครนัสทวีคูณ (U2X=1)	BAUD = fosc/8(UBRR+1)	UBRR=(fosc/8BAUD)-1
โหมดมาสเตอร์ซิงโครนัส	BAUD = fosc/2(UBRR+1)	UBRR=(fosc/2BAUD)-1

โดย

BAUD : อัตราบอตเรตในหน่วยบิตต่อวินาที (bps)

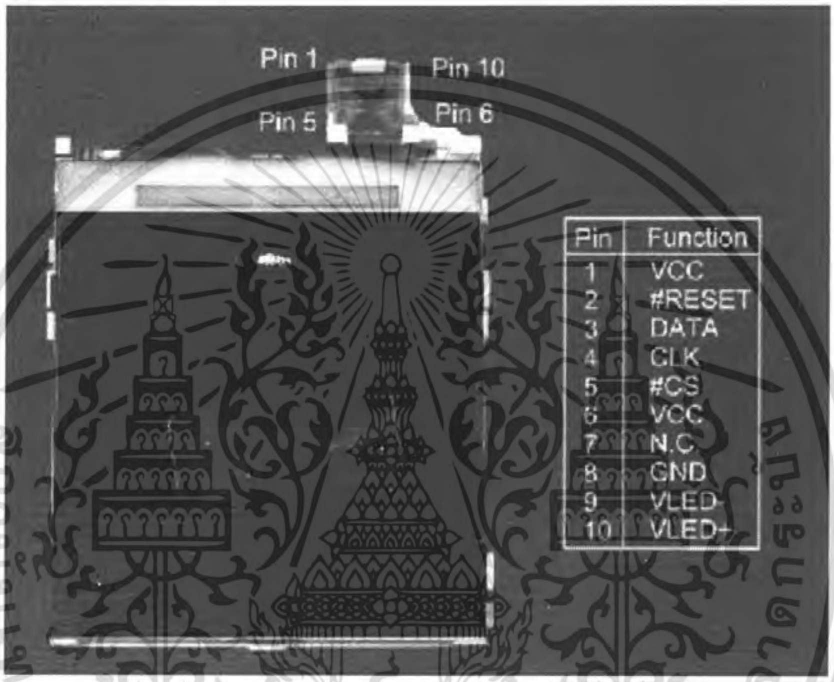
Fosc : ความถี่สัญญาณนาฬิกาหลักของระบบ

UBRR : รีจิสเตอร์ UBRRH และ UBRRL (0-4095)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
จอสี LCD

จอสี LCD ที่ใช้ เป็นจอสี ที่ใช้ในมือถือ Nokia 6100, 7210, 6610, 7250 และ 6220



รูปที่ 5.1 จอสี LCD

รายละเอียดที่สำคัญของจอ

1. มีขนาด 132x132 พิกเซล
2. สามารถเลือกการแสดงภาพ ได้ทั้งแบบ 12 บิต, 8 บิต และ 16 บิต / พิกเซล
3. มีการอินเตอร์เฟส โดยใช้การส่งข้อมูลอนุกรม SPI 9บิต

โดยปกติ จอสี LCD จะออกแบ่งได้เป็น 2 แบบ ตามชิพควบคุมกราฟฟิกที่อยู่ภาพใน ซึ่งเราสามารถสังเกตได้จากภายนอกของจอ ดังนี้

1. สายแพสีเขียว จะเป็นชิพ Epson S1D15G00
2. สายแพสีส้ม จะเป็นชิพ PCF8833

การเขียนซอฟแวร์ใช้กับจอแสดงผลทั้งสอง จะมีความแตกต่างกัน ดังนั้น จึงต้องสังเกตสีของสายแพ (ไม่ใช้สีของแผ่นลายวงจร) ให้ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาของ LCD 6100 มี 10 ขา โดยมีรายละเอียดดังนี้

ขา 1 V_{DIG} คือ แรงดันไฟเลี้ยงสำหรับวงจรดิจิตอลเช่น ไฟเลี้ยงชิพควบคุมที่ฝังอยู่

ขา 2 RST คือ ขารีเซต โดยทำงานที่ลอจิก 0

ขา 3 SDATA คือ ขาสำหรับรับส่งข้อมูลแบบอนุกรม

ขา 4 SCLK คือ ขาสำหรับสัญญาณนาฬิกาสำหรับกำหนดจังหวะการอ่านข้อมูล

ขา 5 CS คือ ขาสำหรับอีนาเบิล(enable)การรับส่งข้อมูลอนุกรมทำงานที่ลอจิก 0

ขา 6 $V_{DISPLAY}$ แรงดันที่ใช้ในการแสดงผล สามารถใช้ได้กับแรงดัน 3.3 โวลต์

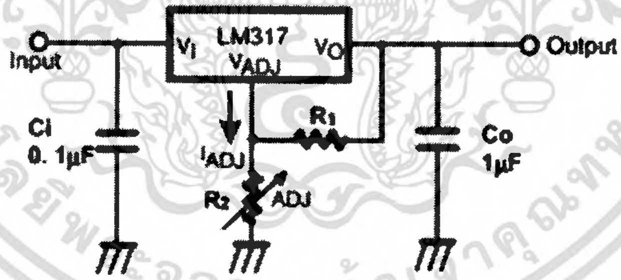
ขา 7 NC ไม่มีการใช้งาน

ขา 8 GND

ขา 9 LED- เป็นขาลบ สำหรับไฟแบคไลท์(Backlight) ถ้าไม่ต้องการควบคุมการจ่ายไฟสามารถต่อ GND ได้โดยตรง

ขา 10 LED เป็นขาบวก สำหรับไฟแบคไลท์ใช้ได้กับแรงดันประมาณ 6-7 โวลต์

จากข้อมูลข้างต้นเราจึงเลือกใช้ไอซี LM317 ในการสร้างแรงดันไฟเลี้ยง 3.3 โวลต์ และสร้างไฟเลี้ยง ขาไฟแบคไลท์ขนาด 6.3 โวลต์



รูปที่ 5.2 วงจรสร้างแรงดันไฟเลี้ยงให้กับจอสี LCD

จากสูตร $Vo = 1.25(1+R_2/R_1)$

ต้องการแรงดันไฟ 3.3 โวลต์ เลือกใช้ $R_1 = 240$ โอห์ม จะได้ค่า $R_2 = 393.6$ โอห์ม ดังนั้นจะเลือกใช้ R_2 เป็นความต้านทานปรับค่าได้ขนาด 1 กิโลโอห์ม

ต้องการแรงดันไฟ 6.3 โวลต์ เลือกใช้ $R_1 = 240$ โอห์ม จะได้ค่า $R_2 = 969.6$ โอห์ม ดังนั้นจะเลือกใช้ R_2 เป็นความต้านทานปรับค่าได้ขนาด 10 กิโลโอห์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
การเขียนซอฟต์แวร์

ในการเขียนโปรแกรมให้กับไมโครคอนโทรลเลอร์ เราจะใช้ภาษาซี ซึ่งเป็นภาษาที่เขียนได้ง่าย และใช้กันอย่างแพร่หลาย ในโครงงานนี้ เราสามารถแบ่งส่วนของการเขียนซอฟต์แวร์ได้เป็น 2 ส่วน คือ

1. ซอฟต์แวร์การรับภาพจากกล้อง C328R
2. ซอฟต์แวร์การแสดงผลภาพบนจอสี LCD

6.1 การเขียนซอฟต์แวร์ให้กับกล้อง C328R

ขาสัญญาณของกล้องที่ต้องนำไปต่อกับไมโครคอนโทรลเลอร์มีอยู่ 2 ขา คือ

1. ขาส่งสัญญาณ (TXD) ต่อกับขา PE0 (RXD0) ของ AVR
2. ขารับสัญญาณ (RXD) ต่อกับขา PE1 (TXD0) ของ AVR

เนื่องจากกล้องรุ่นนี้ ใช้การสื่อสารแบบอนุกรม เราจึงเลือกใช้โมดูล USART ของไมโครคอนโทรลเลอร์ ซึ่งวิธีการสื่อสารแบบอนุกรมชนิดหนึ่ง โดยจะใช้ความเร็วของการสื่อสาร (บอดเรต) เท่ากับ 115,200 บิตต่อวินาที ซึ่งเป็นค่าเริ่มต้นที่กล้องถูกเซตไว้

ชุดของคำสั่งของกล้อง C328R

ตารางที่ 6.1 ชุดคำสั่งของกล้อง C328R

Command	ID Number	Parameter1	Parameter2	Parameter3	Parameter4
Initial	AA01h	00h	Color Type	RAW Resolution (Still image only)	JPEG Resolution
Get Picture	AA04h	Picture Type	00h	00h	00h
Snapshot	AA05h	Snapshot Type	Skip Frame Low Byte	Skip Frame High Byte	00h
Set Package Size	AA06h	08h	Package Size Low Byte	Package Size High Byte	00h
Set Baudrate	AA07h	1st Divider	2nd Divider	00h	00h
Reset	AA08h	Reset Type	00h	00h	xxh*
Power Off	AA09h	00h	00h	00h	00h
Data	AA0Ah	Data Type	Length Byte 0	Length Byte 1	Length Byte 2
SYNC	AA0Dh	00h	00h	00h	00h
ACK	AA0Eh	Command ID	ACK counter	00h / Package ID Byte 0	00h / Package ID Byte 1
NAK	AA0Fh	00h	NAK counter	Error Number	00h

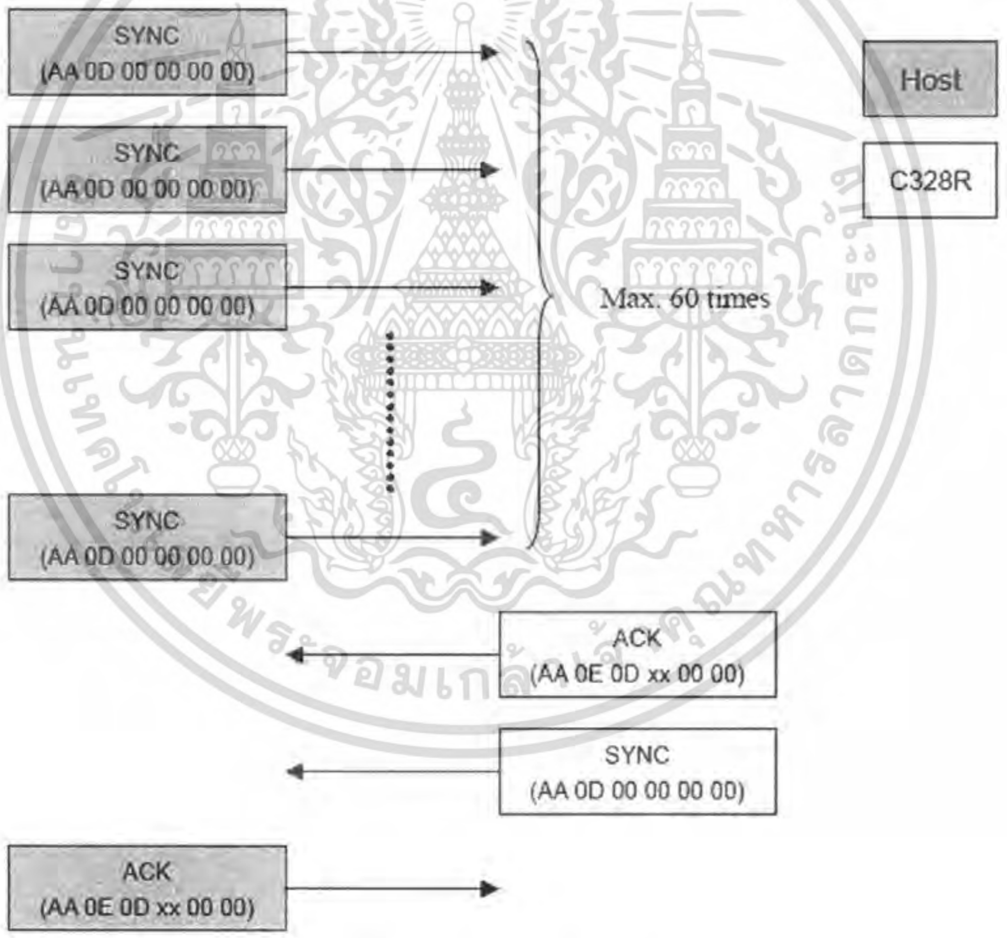
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางจะเห็นว่า รูปแบบของข้อมูล จะมีขนาด 6 ไบต์ ในสองไบต์แรกจะเป็นรหัสระบุคำสั่ง(ID Number) ของแต่ละคำสั่ง ส่วน 4 ไบต์ที่เหลือ เป็นพารามิเตอร์ที่ใช้กำหนดค่าต่างๆ

ในการเขียนซอฟต์แวร์ เราจะต้องเขียนให้ AVR ทำการรับภาพนิ่งจากกล้อง แล้วนำข้อมูลภาพที่ได้ ไปเก็บไว้ในแรมภายนอก เพื่อนำไปแสดงผลต่อไปในส่วนของจอสี LCD

ขั้นตอนการส่งคำสั่งข้อมูลไปยังกล้อง สามารถแสดงได้ ดังนี้

ขั้นแรก จะต้องเริ่มต้นด้วยการส่งคำสั่ง Sync โดยจะวนลูปจนกว่าจะได้ข้อมูลตอบกลับมาจากตัวกล้อง ปกติการวนลูปจะอยู่ในช่วง 25 ลูป แต่จะไม่เกิน 60 ลูป กล้องจะส่งค่าตอบกลับ ACK และส่งคำสั่ง Sync มา จึงต้องตอบกลับไปด้วยคำสั่ง ACK (AA 0E 0D xx 00 00), xx : Don't care

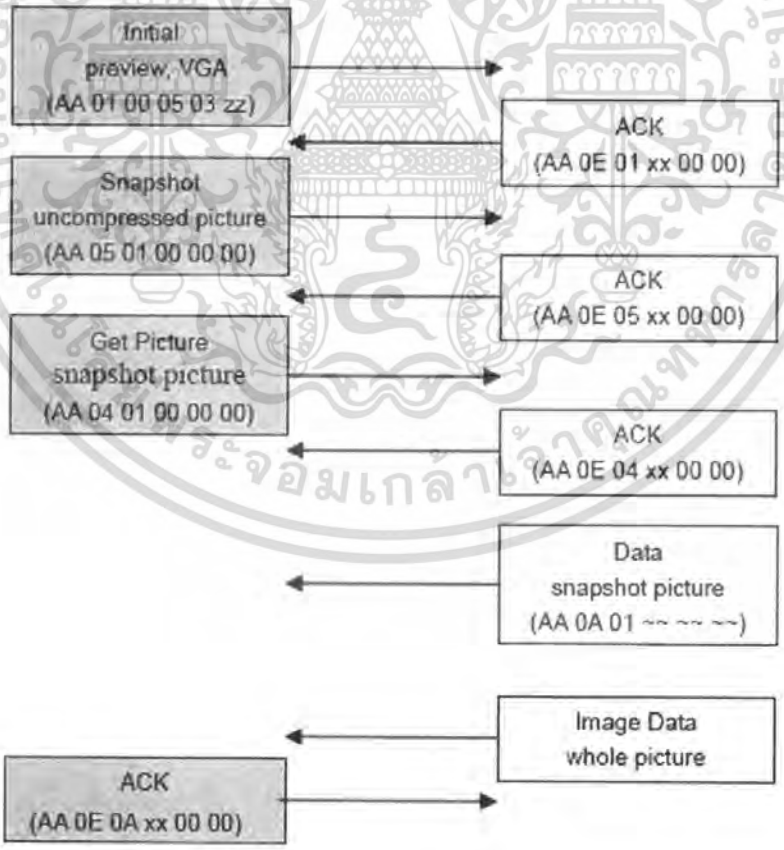


รูปที่ 6.1 การเชื่อมต่อกับกล้องด้วยคำสั่ง Sync

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การส่งคำสั่งเพื่อติดต่อกับกล้องจะต้องได้รับค่าตอบกลับ(ACK) โดยที่สองไบต์แรกจะเป็น 0xAA กับ 0x0E เสมอ ส่วนไบต์ที่สามจะมีค่าเหมือนกับไบต์ที่สองคำสั่ง

เมื่อสามารถทำการเชื่อมต่อกับกล้องได้แล้ว ขั้นต่อไปก็คือ การส่งคำสั่งเพื่อทำการถ่ายภาพนิ่ง ซึ่งจะต้องเริ่มต้นด้วยการส่งคำสั่ง Initial (AA 01 00 05 03 zz), zz : Don't care โดยจะเลือกชนิดของสีเป็นแบบ 12 บิต และเลือกขนาดของภาพเท่ากับ 160x120 พิกเซล ต่อไป คือ คำสั่ง Snapshot (AA 05 00 00 00 00) เป็นคำสั่งแสดงความต้องการที่จะถ่ายภาพนิ่ง ต่อด้วยคำสั่ง Get Picture (AA 04 01 00 00 00) เพื่อถามว่าไฟล์ภาพที่ถ่ายนั้นมีขนาดเท่าไร ข้อมูลที่ตอบกลับมาก็จะมีรูปแบบเป็น (AA 0A 01 L_1L_0 L_3L_2 L_5L_4) ซึ่งภาพที่ถ่ายนี้ก็จะมีขนาดเท่ากับ 0x $L_5L_4L_3L_2L_1L_0$ ไบต์ จากนั้นกล้องก็จะทำการส่งข้อมูลภาพมา และเมื่อสามารถรับข้อมูลได้ครบแล้ว คำสั่งสุดท้ายที่ต้องส่ง คือ คำสั่ง ACK (AA 0E 0A xx 00 00), xx : Don't care เพื่อบ่งบอกการจบขบวนการในการถ่ายภาพนิ่ง



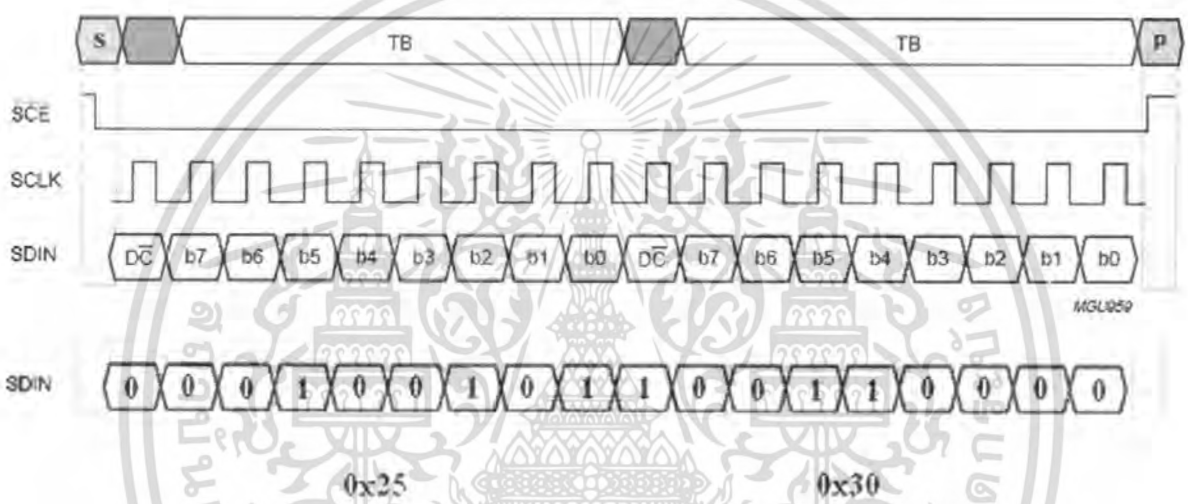
รูปที่ 6.2 การส่งคำสั่งเพื่อถ่ายภาพนิ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 การเขียนซอฟต์แวร์ให้กับจอสี LCD

ภายในจอ LCD 6100 จะมีชิพควบคุมที่ทำหน้าที่สแกนภาพออกที่จอ (เหมือนเวลาเราใช้ 7-segments) โดยจะมีหน่วยความจำ SRAM ในตัว และการติดต่อกับตัวชิพควบคุมจะเป็นการติดต่อแบบอนุกรม

สำหรับจอ LCD 6100 ที่ใช้ IC PCF8833 ของ Phillips เป็นชิพควบคุมใช้การเชื่อมต่อแบบ SPI 9 บิต มี ไทม์มิ่งไดอะแกรม (timing diagram) ดังรูป



รูปที่ 6.3 ไทม์มิ่งไดอะแกรมการส่งข้อมูลอนุกรม 9 บิต

จาก ไทม์มิ่งไดอะแกรมการส่งข้อมูลจะส่ง MSB ก่อน LSB และก่อนจะเป็นข้อมูล 8 บิต จะต้องส่ง บิต DC เพื่อบอกชิพควบคุมว่าไบต์ที่จะส่งนั้น เป็นคำสั่ง หรือเป็นค่าพารามิเตอร์ สำหรับส่งคำสั่ง บิต DC จะต้องเป็นลอจิก 0 และสำหรับพารามิเตอร์ บิต DC จะต้องเป็น 1 เช่น ถ้าต้องการกำหนด ความเข้มของภาพ (contrast) จะต้องส่งคำสั่ง 0x25 ตามด้วย ค่าพารามิเตอร์ความสว่าง ในที่นี้กำหนดเป็น 0x3F

สำหรับสัญญาณ CS (SCE ในรูป) ใช้กำหนดจังหวะการเริ่มต้นการส่งข้อมูล เพื่อให้ผู้ส่งและผู้รับ เข้าใจตรงกันว่าให้นับบิตไห••เป็นบิตแรก และแต่ละไบต์ สามารถส่งต่อเนื่องกันโดยที่ไม่ต้องหยุดการส่งข้อมูลแล้วเริ่มใหม่ โดยสามารถเริ่มต้นครั้งเดียวแล้วส่งคำสั่งตามด้วยค่าพารามิเตอร์แล้วส่งคำสั่งถัดไปได้ทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาสัญญาณของจอสี LCD ที่ใช้ติดต่อกับไมโครคอนโทรลเลอร์มีอยู่ด้วยกัน 4 ขา ได้แก่

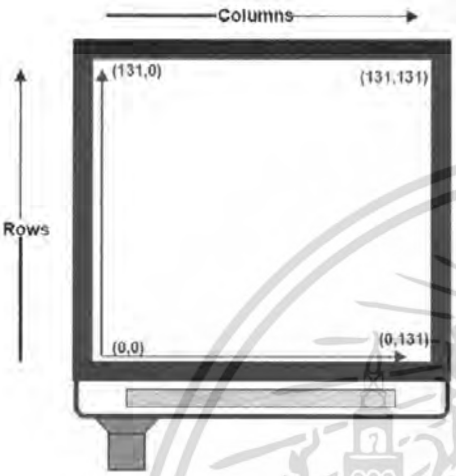
1. ขา CS ต่อกับขา PB0
2. ขา SCLK ต่อกับขา PB1
3. ขา SDATA ต่อกับขา PB2
4. ขา RST ต่อกับขา PB3

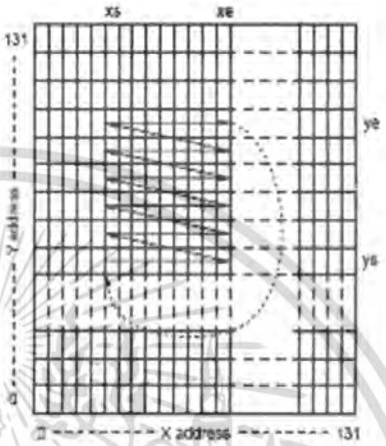
คำสั่งที่ใช้ในการเขียนโปรแกรมแสดงผลจอสี LCD

1. SLEEPOUT เป็นคำสั่งที่ส่งเพื่อให้ IC PCF8833 เกิดการ power-up
2. NORON (normal display mode on) เป็นคำสั่งที่ทำให้จออยู่ในโหมดปกติ
3. COLMOD () เป็นคำสั่งที่ใช้ในการเลือกรูปแบบของสี ซึ่งมีอยู่ด้วยกัน 3 แบบ ได้แก่ 12 บิต, 8 บิต และ 16 บิต / พิกเซล
4. MADCTL (memory data access control) เป็นคำสั่งที่ใช้ในการกำหนดค่าสถานะต่างๆที่เกี่ยวข้องกับการเขียนข้อมูลลงแรม
5. SETCON (set con trast) เป็นคำสั่งที่ใช้ในการปรับค่าไฟเลี้ยง V_{LCD}
6. DISPON (display on) เป็นคำสั่งที่ทำให้มีการแสดงผลตามข้อมูลที่อยู่ในแรม และค่าที่ได้เซ็ตไว้
7. RGBSET (colour set) เป็นคำสั่งที่ใช้ในการกำหนดสีที่ใช้ในการแสดงผล
8. CASET (column address set) เป็นคำสั่งที่ใช้ในการกำหนดจุดเริ่ม (xs) และจุดสุดท้าย (xe) ของคอลัมน์ สำหรับการเขียนข้อมูลลงในแรม
9. PASET (page address set) เป็นคำสั่งที่ใช้ในการกำหนดจุดเริ่ม (ys) และจุดสุดท้าย (ye) ของแถว สำหรับการเขียนข้อมูลลงในแรม
10. RAMWR (memory write) เป็นคำสั่งในการเขียนข้อมูลลงแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทิศทางของการแสดงจุดของสี ของจอ LCD Nokia 6100 ที่ค่าเริ่มต้นของจอ จะเป็นดังรูป แต่เราสามารถที่จะปรับแก้ทิศทางการจัดเรียงของจุดสีได้ โดยใช้คำสั่ง MADCTL





รูปที่ 6.4 ทิศทางการแสดงจุดสีของจอสี LCD

รูปแบบของข้อมูลที่จอ LCD สามารถแสดงได้มีอยู่ด้วยกัน 3 แบบ ดังที่กล่าวมาแล้วข้างต้น ซึ่งการเขียนโปรแกรมในการแสดงผลก็จะมีความแตกต่างกัน เพราะมีขนาดของข้อมูลต่อหนึ่งจุดพิกเซลไม่เท่ากัน ดังนี้

1. 12 บิต/พิกเซล ,RGB 4:4:4

จากตารางจะเห็นว่า ในการเขียนโปรแกรมแสดงภาพ จะต้องส่งข้อมูลทีละ 3 ไบต์ ซึ่งจะได้จุดสีจำนวน 2 พิกเซล

ตารางที่ 6.2 แสดงการเขียนข้อมูล สำหรับ RGB 4:4:4

BYTE	D/$\overline{C}$	7	6	5	4	3	2	1	0
1st write	1	R_3	R_2	R_1	R_0	G_3	G_2	G_1	G_0
2nd write	1	B_3	B_2	B_1	B_0	R_3	R_2	R_1	R_0
3rd write	1	G_3	G_2	G_1	G_0	B_3	B_2	B_1	B_0

2. 8 บิต/พิกเซล ,RGB 3:3:2

จากตารางจะเห็นว่า ในการเขียนโปรแกรมแสดงภาพ ก็จะทำการส่งข้อมูลที่ละไบต์ ซึ่งจะได้จุดสีจำนวน 1 พิกเซล

ตารางที่ 6.3 แสดงการเขียนข้อมูล สำหรับ RGB 3:3:2

BYTE	D/$\overline{C}$	7	6	5	4	3	2	1	0
1st write	1	R_2	R_1	R_0	G_2	G_1	G_0	B_1	B_0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. 16 บิต/พิกเซล ,RGB 5:6:5

จากตารางจะเห็นว่า ในการเขียนโปรแกรมแสดงภาพ ก็จะทำการส่งข้อมูลที่ละ 2ไบต์ เพื่อให้ได้จุดสีจำนวน 1 พิกเซล

ตารางที่ 6.4 แสดงการเขียนข้อมูล สำหรับ RGB 5:6:5

BYTE	D/C	7	6	5	4	3	2	1	0
1st write	1	R_4	R_3	R_2	R_1	R_0	G_5	G_4	G_3
2nd write	1	G_2	G_1	G_0	B_4	B_3	B_2	B_1	B_0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

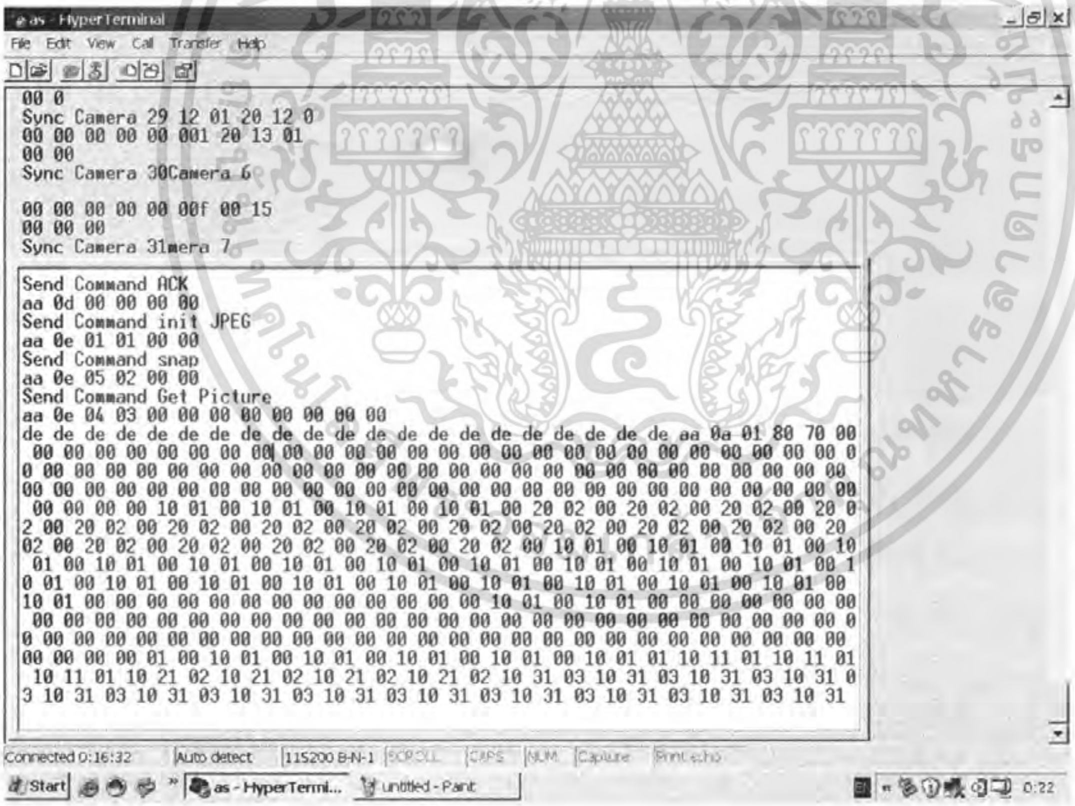
บทที่ 7

การทดสอบ

การทดสอบขั้นแรกทำได้โดยรับภาพจากจากกล้องเอาไปเก็บไว้ในแรมภายนอกแล้วดึงภาพจากแรมภายนอกไปแสดงในไฮเปอร์เทอมินอล

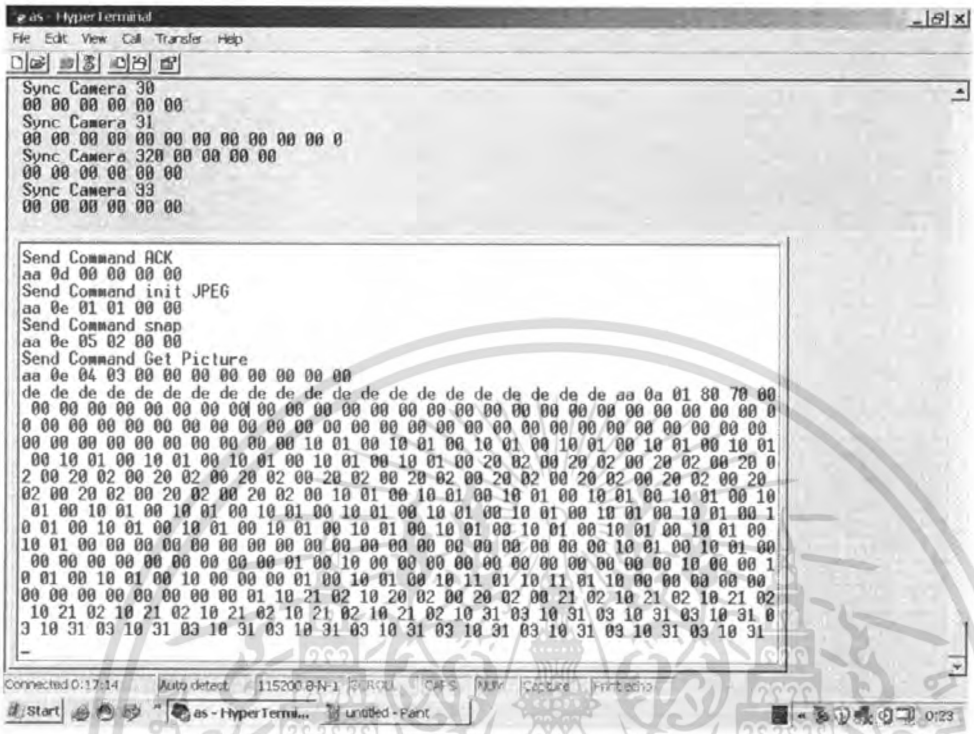
จากหน้าต่างไฮเปอร์เทอมินอลข้างล่างนี้จะเห็นว่าเมื่อเราส่งสัญญาณ Sync (aa 0d 00 00 00 00) ไปยังกล้อง กล้องจะส่งสัญญาณ ACK ตอบกลับมา โดยที่เราส่ง Sync ไปไม่เกิน 60 ลูป

เมื่อติดต่อกับกล้องได้แล้ว เราจะส่งคำสั่ง Initial ตามด้วยคำสั่ง Snapshot และ Get picture ตามลำดับ เพื่อสั่งให้กล้องส่งข้อมูลภาพกลับมา โดยกล้องจะส่งขนาดของภาพ (aa 01 0a 80 70 00) ซึ่งคิดเป็นขนาดของภาพได้เท่ากับ 0x007080 = 28.8 กิโลไบต์ แล้วตามด้วยข้อมูลภาพ

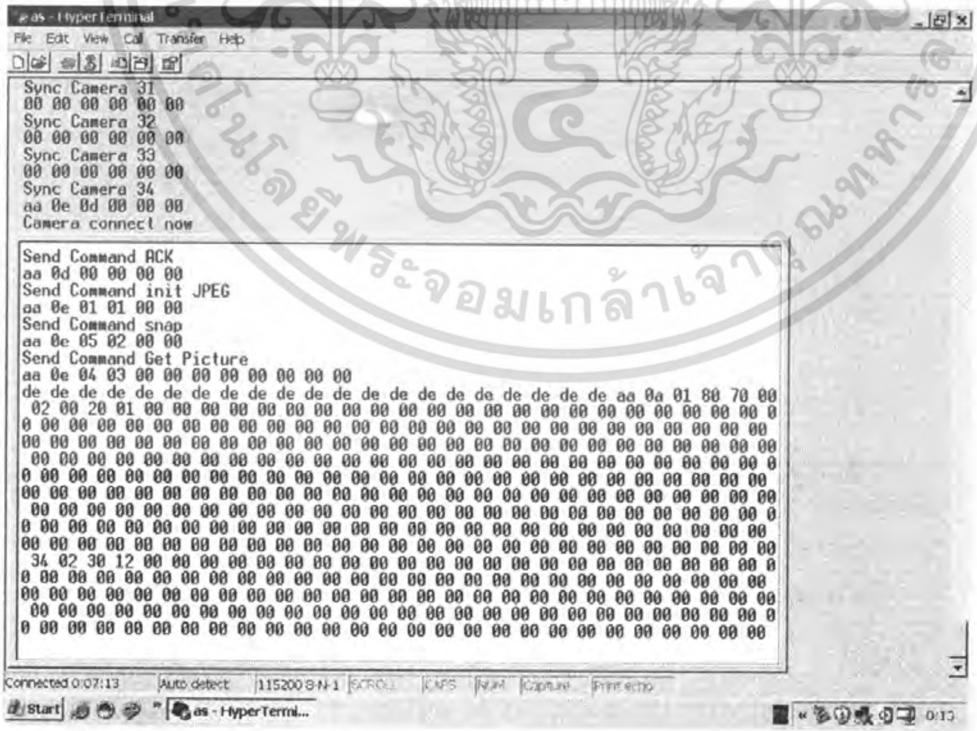


รูปที่ 7.1 การแสดงผลข้อมูลที่ได้จากการจับภาพสีแดงครั้งที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

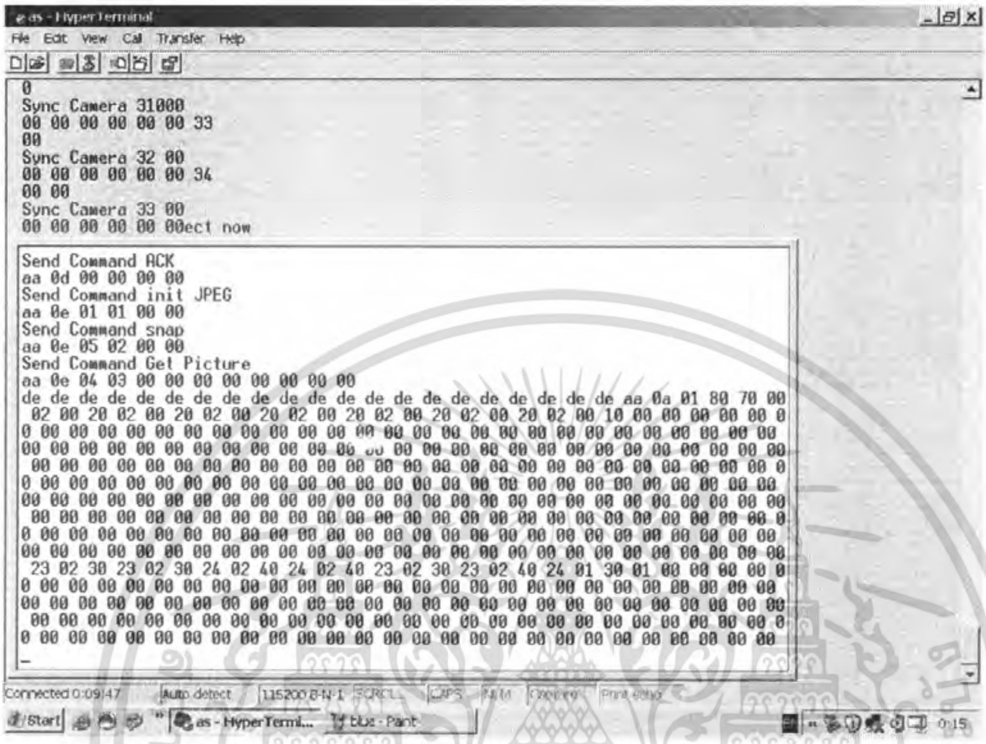


รูปที่ 7.2 การแสดงผลข้อมูลที่ได้จากการจับภาพสีแดงครั้งที่ 2



รูปที่ 7.3 การแสดงผลข้อมูลที่ได้จากการจับภาพสีน้ำเงินครั้งที่ 1

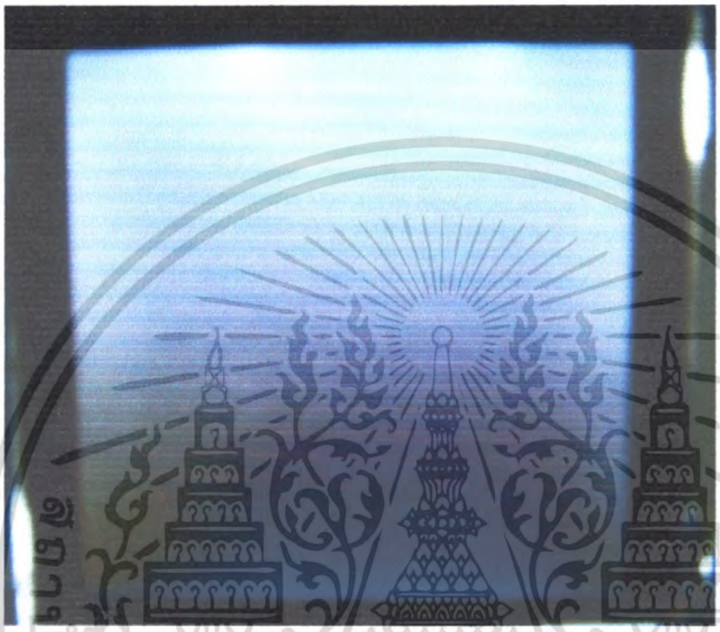
เอกสารนี้เป็นเอก ... ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



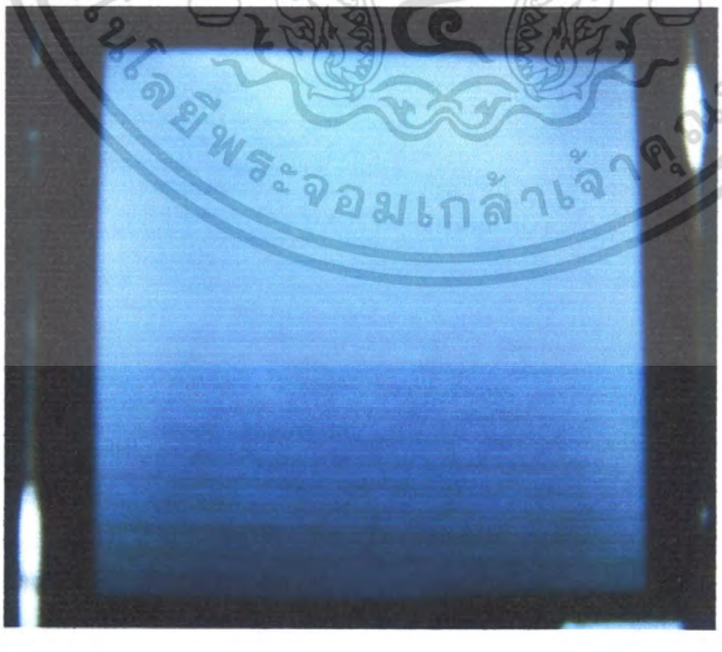
รูปที่ 7.4 การแสดงข้อมูลที่ได้จากการจับภาพสีน้ำเงินครั้งที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบขั้นต่อไป โดยรับภาพจากจากกล้องเอาไปเก็บไว้ในแรมภายนอกแล้วดึงภาพจากแรมภายนอกไปแสดงในจอสี LCD โดยในการทดสอบเราจะนำกล้องไปจับภาพฉากที่เป็นแม่สี 3 สีคือแดง, น้ำเงิน และเขียว ได้ผลดังรูป

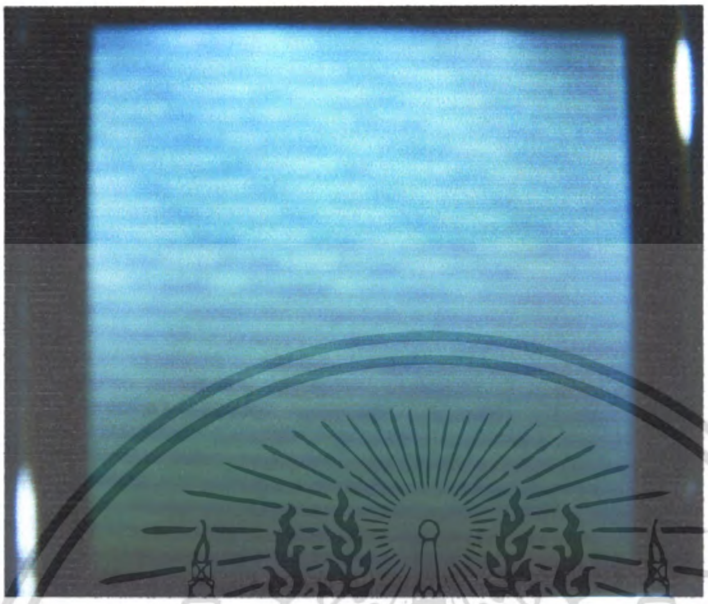


รูปที่ 7.5 การแสดงภาพสีแดงบนจอสี LCD



รูปที่ 7.6 การแสดงภาพสีน้ำเงินบนจอสี LCD

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.7 การแสดงภาพสีเขียวบนจอสี LCD

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8
บทสรุป

จากการทดสอบจะเห็นได้ว่า เมื่อเรานำกล้องไปจับฉากที่เป็นแม่สี เราสามารถแสดงออกมาเป็นเส้นสีได้อย่างถูกต้อง แต่เมื่อเรานำกล้องไปจับภาพวัตถุอื่นๆ ภาพที่ได้จะมีการผิดเพี้ยนไป สาเหตุอาจเกิดจากเมื่อเรานำกล้องไปจับฉากที่เป็นแม่สีโฟกัสของกล้องสามารถโฟกัสไปที่สีๆเดียวได้เลย ภาพที่แสดงบนจอสี LCD จึงออกมาได้ถูกต้อง แต่เมื่อเรานำกล้องไปจับภาพฉากที่เป็นวิวหรือวัตถุอื่นๆที่มีสีเป็นจำนวนมาก โฟกัสของกล้องไม่สามารถโฟกัสสีทั้งหมดได้ทันจึงทำให้ภาพที่แสดงออกมาผิดเพี้ยนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ประจิน พลังสันติกุล,“การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์ AVR ด้วยภาษา C กับ WinAVR C (Compiler)”,แอพซอฟต์เทค,กรุงเทพฯ ,365 หน้า ,2549
2. Thaieasyelec,“ควบคุมจอ Nokia GLCD 6100 (Chip Phillips) ด้วย MCU 8 bits” ,www.thaieasyelec.com
3. Thaimobilecenter Editor,“เซ็นเซอร์รับภาพแบบ CMOS และ CCD’’ ,www.thaimobilecenter.com/home/mobile_tips.asp
4. อารัมภีร์ จันทร์ใย,“สร้างกล้องดิจิตอลด้วยตัวคุณเอง”,วารสารเซมิคอนดักเตอร์ อิเล็กทรอนิกส์,ฉบับที่304,กรุงเทพฯ ,2550,หน้า 131-139.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอฟต์แวร์ที่ใช้ในโครงงาน

```
#include <avr/io.h>
#include <avr/interrupt.h>
#include <stdio.h>
#include <util/delay.h>
#include <avr/pgmspace.h>
#include "AVR-PROTO_BETA.h"
#include "LCD6100Philips.h"

#define F_CPU 8000000UL
#define Delay1ms(x)          _delay_ms(x)
#define sbi(port,bitnum)          port |= _BV(bitnum)
#define cbi(port,bitnum)          port &= ~(_BV(bitnum))

void Delay100ms(uint8_t x);

void Delay100ms(uint8_t x)
{
    uint8_t i;
    for (i=0;i<x;i++)
    {
        Delay1ms(100);
    }
}

#define    TIMECONST      5000    // Good 80 // it better at 400000

unsigned int Image=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unsigned char Recived[] = { 0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
                            0x00,0x00,0x00,0x00 };
//----------------------------------------------------------------------------//
unsigned char reset[] =  { 0xAA, 0x08, 0x01, 0x00, 0x00, 0xFF };
unsigned char sync [] =  { 0xAA, 0x0D, 0x00, 0x00, 0x00, 0x00 };
unsigned char ack  [] =  { 0xAA, 0x0E, 0x0D, 0x02, 0x00, 0x00 };
unsigned char init [] =  { 0xAA, 0x01, 0x00, 0x05, 0x03, 0x03 };
unsigned char snap [] =  { 0xAA, 0x05, 0x01, 0x00, 0x00, 0x00 };
unsigned char getp [] =  { 0xAA, 0x04, 0x01, 0x00, 0x00, 0x00 };
unsigned char ack1 [] =  { 0xAA, 0x0E, 0x0A, 0x00, 0x00, 0x00 };
/********************************Prototype function*/
static int uart_putchar(char c, FILE *stream);
static FILE uart_str =    FDEV_SETUP_STREAM(uart_putchar,NULL,
                                            _FDEV_SETUP_WRITE);
void Init_Serial(unsigned int baudrate);
/********************************USART0 putchar*/
static int uart_putchar(char c, FILE *stream)
{
        if(c =='\a') {
                fputs("*ring*\n",stderr);
                return 0;
   }
        if(c =='\n')
   uart_putchar('\r',stream);
   loop_until_bit_is_set(UCSR0A,UDRE0);
        UDR0 = c;
        return 0;
}
/********************************USART transmit*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void USART_Transmit(unsigned char data)
{
        while(!(UCSR1A & (1<<UDRE1)))
        ;
        UDR1 = data;
}
/*********************************Initiaize UART0*/
static void USART_Init0(unsigned int baud)
{
//SET baud rate
UBRR0H=(unsigned char) (baud>>8);
UBRR0L=(unsigned char) baud;
//Enable transmitter
UCSR0B= 0x08;
//Set frame format: 8data,NonParity,1stop bit
UCSR0C=0x86;
//Set address uart_str to stdout
        stdout= &uart_str;
}
/*********************************Initiaize UART1*/
static void USART_Init1(unsigned int baud)
{
//SET baud rate
UBRR1H=(unsigned char) (baud>>8);
UBRR1L=(unsigned char) baud;
//Enable receiver and transmitter
UCSR1B=0x18;
//Set frame format: 8data,NonParity,1stop bit
UCSR1C=0x86;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
//-----------------------------------------------------------------------//
unsigned char uartGetChar () // Channel 0 with ATMEGA128
{
   unsigned char x;
          unsigned long timeout;
          timeout = 0;
          do {
                      timeout++;
                      if (timeout > TIMECONST)
                            break;
          } while ((UCSR1A&(1<<RXC1)) == 0);
          if (timeout < TIMECONST)
          {
                 x = UDR1;
                 // return 1;
          }
          return x;
}
//-----------------------------------------------------------------------------//
void SendCommand (unsigned char *dat)
{
          int i;
          for (i = 0; i < 6;i++)
          {
                 USART_Transmit (dat[i]);
                 _delay_us (5);
          }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*---------------------------------------------------------------------------------------*/
int wait_for_ack(unsigned char *dat)
{
  unsigned int  i = 0;
  unsigned long timeout = 0;
        while (i < 6 && (timeout++) < 40)  // Comment 35 it ok.... sure for work
        {
                Recived[i++]=uartGetChar();
                //Recived[i]= rChar;
                timeout = 0;
        }
        if (timeout < 40)
        {
                for (i = 0; i < 6, i++)
                        printf ("%02x ",Recived[i]);
                        printf ("\n");
                return  1;
        }
        return 0;
};
/*---------------------------------------------------------------------------------------*/
int wait_for_pic(unsigned char *dat)
{
  unsigned int  i = 0;
  unsigned long timeout = 0;
  //unsigned char rChar;
        while (i < 12 && (timeout++) < 5000)  // Comment 35 it ok.... sure for work
        {
                Recived[i++]=uartGetChar();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                //Recived[i]= rChar;
                timeout = 0;
        }
        if (timeout < 40)
        {
                for (i = 0; i < 12; i++)
                        printf ("%02x ",Recived[i]);
                        printf ("\n");
                return  1;
        }
        return 0;
};
/*-----------------------------------------------------------------------------------*/
int main(void)
{
        MCUCR =0x80;
        unsigned int loop, retry;
        unsigned int rgb_size;
        unsigned int complete = 0;
        USART_Init0(8);                 //115200
        USART_Init1(8);                 //115200
        sei();          //enable interrupts
        printf ("\nDigital Camera Poject\n");
//      for (retry = 0; retry < 3; retry++)
        //{
        SendCommand (reset);
        _delay_ms (10);
        for (loop = 0; loop < 60; loop++)
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
printf ("Sync Camera %d\n",loop);
SendCommand (sync);
// Camera accept about loop = 24
//memset (Recived, 0, sizeof Recived);
if (wait_for_ack (Recived))
        {
        if ((Recived[0] == 0xAA) && (Recived[1] == 0x0E))
        break;
        }
}
// if loop > 60 then camera not connect;
if (loop > 35)
        printf ("Camera not Connect\n");
else
{
        printf ("Camera connect now\n");
        while (1)
        {
        //send ACK
                complete = 0;
                printf ("Send Command ACK\n");
                SendCommand (ack);
                //get dummy byte
                if (wait_for_ack (Recived))
                {
                        if ((Recived[0] != 0xAA) && (Recived[1] != 0x0E))
                        break;
                }
                printf ("Send Command init JPEG\n");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
		SendCommand (init);
		if (wait_for_ack (Recived))
		{
			if ((Recived[0] != 0xAA) && (Recived[2] != 0x01))
			break;
		}
		printf ("Send Command snap\n");
		SendCommand (snap);
		if (wait_for_ack (Recived))
		{
			if ((Recived[0] != 0xAA) && (Recived[2] != 0x05))
			break;
		}
		printf ("Send Command Get Picture\n");
		SendCommand (getp);
		if (wait_for_pic (Recived))
		{
			if ((Recived[0] != 0xAA) && (Recived[2] != 0x04)
			&&(Recived[6] != 0xAA) && (Recived[8] != 0x01))
			break;
		}
		rgb_size = Recived[11];
		rgb_size<<=8;
		rgb_size=rgb_size+Recived[10];
		rgb_size<<=8;
		rgb_size=rgb_size+Recived[9];
//		printf ("\nImage Size = %d",rgb_size);
		//after init camrea success just to return complete = 1;
		complete = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                break;
            }
            if(complete){           // test loop
                unsigned int  y,z=0;
                unsigned char *p_t;
                y=0x2000;
                while (z++ < 27000){
                unsigned char pic;
                pic =uartGetChar();
                p_t = (unsigned char *)y;
               *p_t = pic;
                y++;
                }
            }                           //test loop
        }       // else

//      } // retry;
  //}// while(1)
  // LCD NOKIA

        LCD6100PwrUp();
        LCD6100Init();
        LCD6100SetRGB();
        Delay100ms(10);
        LCD6100TestPic();
  return 0;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

General Description

The C328 module is a highly integrated serial camera board that can be attached to a wireless or PDA host performing as a video camera or a JPEG compressed still camera. It provides a serial interface (RS-232) and JPEG compression engine to act as a low cost and low powered camera module for high-resolution serial bus security system or PDA accessory applications.

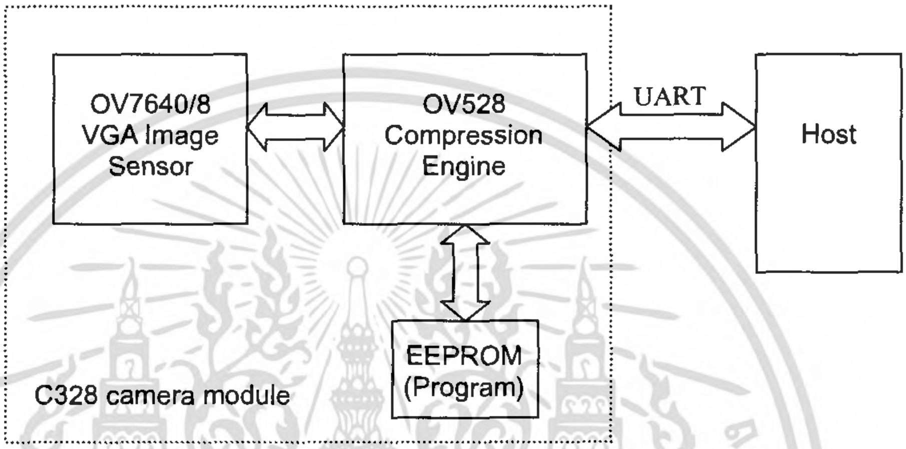


Figure 1 – System block diagram

Features

- Small in size, low cost and low powered (3.3V) camera module for high-resolution serial bus security system or PDA accessory applications.
- On-board EEPROM provides a command-based interface to external host via RS-232.
- UART: 115.2Kbps for transferring JPEG still pictures or 160x128 preview @8bpp with 0.75fps.
- On board OmniVision OV7640/8 VGA color sensor.
- Built-in JPEG CODEC for different resolutions.
- Built-in down sampling, clamping and windowing circuits for VGA, QVGA, 160x120 or 80x60 image resolutions.
- Built-in color conversion circuits for 2-bit gray, 4-bir gray, 8-bit gray, 12-bit RGB, 16-bit RGB or standard JPEG preview images.
- No external DRAM required.

System Configuration

1. Camera Sensor
   The C328-7640 module uses OmniVision OV7640/8 VGA color digital CameraChips with an 8-bit YCbCr interface.

2. OV528 Serial Bridge
   The OV528 Serial Bridge is a JPEG CODEC embedded controller chip that can compress and transfer image data from CameraChips to external device. The OV528 takes 8-bit YCbCr 422 progressive video data from an OV7640/8 CameraChip. The camera interface synchronizes with input video data and performs down sampling, clamping and windowing functions with desired resolution, as well as color conversion that is requested by the user through serial bus host commands.
   The JPEG CODEC can achieve higher compression ratio and better image quality for various image resolutions.

3. Program EEPROM
   A serial type program memory is built-in for C328-7640 to provide a set of user-friendly command interfacing to external host.

เอกสารนี้เป็น...สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด...หา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Board Layout

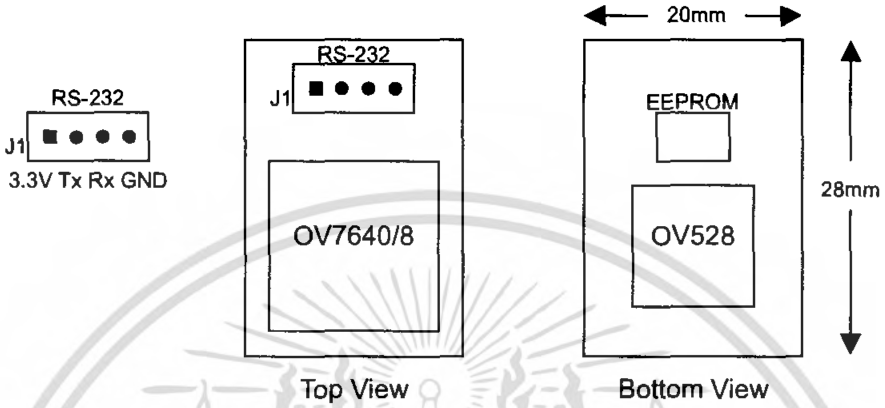


Figure 2 – C328-7640 board layout and serial interface pin

Serial Interface

1. Single Byte Timing Diagram
   A single byte RS-232 transmission consists of the start bit, 8-bit contents and the stop bit. A start bit is always 0, while a stop bit is always 1. LSB is sent out first and is right after the start bit.

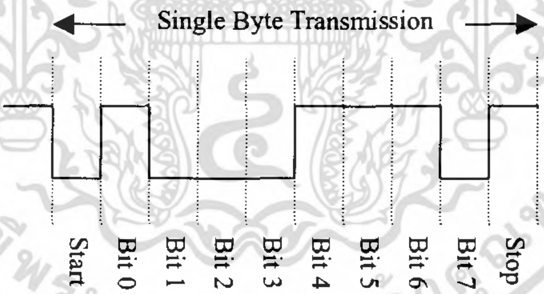


Figure 3 – RS-232 single byte timing diagram

2. Command Timing Diagram
   A single command consists of 6 continuous single byte RS-232 transmissions. The following is an example of SYNC (AA0D00000000h) command.

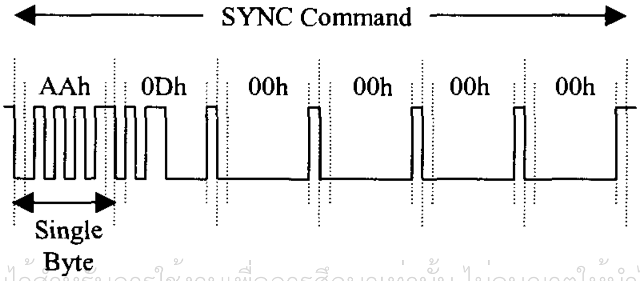


Figure 4 – RS-232 SYNC command timing diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Set

The C328-7640 module supports total 11 commands for interfacing to host as following:

Command	ID Number	Parameter1	Parameter2	Parameter3	Parameter4
Initial	AA01h	00h	Color Type	Preview Resolution	JPEG Resolution
Get Picture	AA04h	Picture Type	00h	00h	00h
Snapshot	AA05h	Snapshot Type	Skip Frame Low Byte	Skip Frame High Byte	00h
Set Package Size	AA06h	08h	Package Size Low Byte	Package Size High Byte	00h
Set Baudrate	AA07h	1st Divider	2nd Divider	00h	00h
Reset	AA08h	Reset Type	00h	00h	xxh*
Power Off	AA09h	00h	00h	00h	00h
Data	AA0Ah	Data Type	Length Byte 0	Length Byte 1	Length Byte 2
SYNC	AA0Dh	00h	00h	00h	00h
ACK	AA0Eh	Command ID	ACK counter	00h / Package ID Byte 0	00h / Package ID Byte 1
NAK	AA0Fh	00h	NAK counter	Error Number	00h

* If the parameter is 0xFF, the command is a special Reset command and the firmware responds to it immediately.

1. Initial (AA01h)

The host issues this command to configure the preview image size and color type. After receiving this command, the module will send out an ACK command to the host if the configuration success. Otherwise, an NACK command will be sent out.

1.1 Color Type

C328-7640 can support 7 different color types as follow:

2-bit Gray Scale	01h
4-bit Gray Scale	02h
8-bit Gray Scale	03h
12-bit Color	05h
16-bit Color	06h
JPEG	07h

1.2 Preview Resolution

80x60	01h
160x120	03h

1.3 JPEG Resolution

Since the Embedded JPEG Code can support only multiple of 16, the JPEG preview mode can support following image sizes. It is different from normal preview mode.

80x64	01h
160x128	03h
320x240	05h
640x480	07h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Get Picture (AA04h)

The host gets a picture from C328-7640 by sending this command.

2.1 Picture Type

Snapshot Picture	01h
Preview Picture	02h
JPEG Preview Picture	05h

3. Snapshot (AA05h)

C328-7640 keeps a single frame of JPEG still picture data in the buffer after receiving this command.

3.1 Snapshot Type

Compressed Picture	00h
Uncompressed Picture	01h

3.2 Skip Frame Counter

The number of dropped frames can be defined before compression occurs. “0” keeps the current frame, “1” captures the next frame, and so forth.

4. Set Package Size (AA06h)

The host issues this command to change the size of data package which is used to transmit JPEG image data from the C328-7640 to the host. This command should be issued before sending Snapshot command or Get Picture command to C328-7640. It is noted that the size of the last package varies for different image.

4.1 Package Size

The default size is 64 bytes and the maximum size is 512 bytes.

Byte0 ... ByteN

ID	Data Size	Image Data	Verify Code
(2 bytes)	(2 bytes)	(Package size - 6 bytes)	(2 bytes)

◄── Package Size ──►

ID -> Package ID, starts from zero for an image

Data Size -> Size of image data in the package

Verify Code -> Error detection code, equals to the lower byte of sum of the whole package data except the verify code field. The higher byte of this code is always zero. i.e. verify code = lowbyte(sum(byte[0] to byte[N-2]))

Note: As the transmission of uncompressed image is not in package mode, it is not necessary to set the package size for uncompressed image.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Set Baudrate (AA07h)

Set the C328-7640 baud rate by issuing this command. As the module can auto-detect the baud rate of the incoming command, host can make connection with one of the following baud rate in the table. The module will keep using the detected baud rate until physically power off

5.1 Baudrate Divider

Baudrate = 14.7456MHz / 2 x (2nd Divider + 1) / 2 x (1st Divider + 1)

Baudrate	1st Divider	2nd Divider	Baudrate	1st Divider	2nd Divider
7200 bps	ffh	01h	28800 bps	3fh	01h
9600 bps	bfh	01h	38400 bps	2fh	01h
14400 bps	7fh	01h	57600 bps	1fh	01h
19200 bps	5fh	01h	115200 bps	0fh	01h

6. Reset (AA08h)

The host reset C328-7640 by issuing this command.

6.1 Reset Type

"00h" resets the whole system. C328-7640 will reboot and reset all registers and state machines.
"01h" resets state machines only.

7. Power Off (AA09h)

C328-7640 will go into sleep mode after receiving this command. SYNC command (AA0Dh) must be sent to wake up C328-7640 for certain period until receiving ACK command from C328-7640.

8. Data (AA0Ah)

C328-7640 issues this command for telling the host the type and the size of the image data which is ready for transmitting out to the host.

8.1 Data Type

Snapshot Picture	01h
Preview Picture	02h
JPEG Preview Picture	05h

8.2 Length

These three bytes represent the length of data of the Snapshot Picture, Preview Picture or JPEG Preview Picture.

9. SYNC (AA0Dh)

Either the host or the C328-7640 can issue this command to make connection. An ACK command must be sent out after receiving this command.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ACK (AA0Eh)

This command indicates the success of last operation. After receiving any valid command, ACK command must be sent out except when getting preview data. The host can issue this command to request image data package with desired package ID after receiving Data command from C328-7640. The host should send this command with package ID F0F0h after receiving a package to end the package transfer. Note that the field "command ID" should be 00h when request image data package.

10.1 Command ID

The command with that ID is acknowledged by this command.

10.2 ACK Counter

No use.

10.3 Package ID

For acknowledging Data command, these two bytes represent the requested package ID. While for acknowledging other commands, these two bytes are set to 00h.

11. NAK (AA0Fh)

This command indicates corrupted transmission or unsupported features.

11.1 NAK Counter

No use.

11.2 Error Number

Picture Type Error	01h	Parameter Error	0bh
Picture Up Scale	02h	Send Register Timeout	0ch
Picture Scale Error	03h	Command ID Error	0dh
Unexpected Reply	04h	Picture Not Ready	0fh
Send Picture Timeout	05h	Transfer Package Number Error	10h
Unexpected Command	06h	Set Transfer Package Size Wrong	11h
SRAM JPEG Type Error	07h	Command Header Error	F0h
SRAM JPEG Size Error	08h	Command Length Error	F1h
Picture Format Error	09h	Send Picture Error	F5h
Picture Size Error	0ah	Send Command Error	ffh

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FAIRCHILD
SEMICONDUCTOR®

March 2000
Revised June 2005

LM317
3-Terminal Positive Adjustable Regulator

General Description

This monolithic integrated circuit is an adjustable 3-terminal positive voltage regulator designed to supply more than 1.5A of load current with an output voltage adjustable over a 1.2 to 37V. It employs internal current limiting, thermal shut-down and safe area compensation.

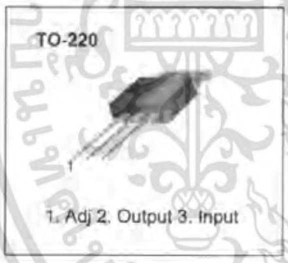
Features

- Output Current In Excess of 1.5A
- Output Adjustable Between 1.2V and 37V
- Internal Thermal Overload Protection
- Internal Short Circuit Current Limiting
- Output Transistor Safe Operating Area Compensation
- TO-220 Package
- D2 PAK Package

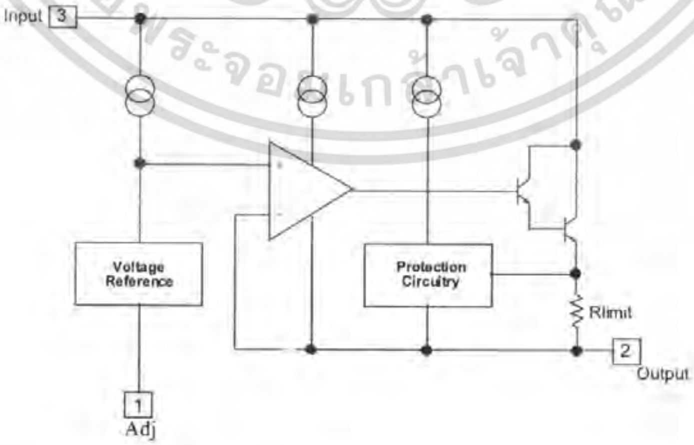
Ordering Code:

Product Number	Package	Operating Temperature
LM317T	TO-220	0°C to +125°C
LM317D2TXM	D2 PAK	0°C to +125°C

Connection Diagrams



Internal Block Diagram



© 2005 Fairchild Semiconductor Corporation DS400281

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input-Output Voltage Differential	$V_I - V_O$	40	V
Lead Temperature	T_{LEAD}	230	°C
Power Dissipation	P_D	Internally limited	W
Operating Junction Temperature Range	T_j	0 ~ +125	°C
Storage Temperature Range	T_{STG}	−65 ~ +125	°C
Temperature Coefficient of Output Voltage	$\Delta V_O/\Delta T$	±0.02	% / °C

Note 1: Absolute Maximum Ratings: are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at these limits. The parametric values defined in the Electrical Characteristics tables are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Electrical Characteristic

($V_I - V_O = 5V$, $I_O = 0.5A$, $0°C \le T_J \le +125°C$, $I_{MAX} = 1.5A$, $P_{DMAX} = 20W$, unless otherwise specified)

Parameter	Symbol	Conditions	Min	Typ	Max	Unit
Line Regulation (Note 2)	Rline	$T_A = +25°C$ $3V \le V_I - V_O \le 40V$	—	0.01	0.04	% / V
		$3V \le V_I - V_O \le 40V$	–	0.02	0.07	% / V
Load Regulation (Note 2)	Rload	$T_A = +25°C$, $10mA \le I_O \le I_{MAX}$ $V_O < 5V$ $V_O \ge 5V$	–	18.0 0.4	25.0 0.5	mV% / V_O
		$10mA \le I_O \le I_{MAX}$ $V_O < 5V$ $V_O \ge 5V$	–	40.0 0.8	70.0 1.5	mV% / V_O
Adjustable Pin Current	I_{ADJ}	–	–	46.0	100	µA
Adjustable Pin Current Change	ΔI_{ADJ}	$3V \le V_I - V_O \le 40V$ $10mA \le I_O \le I_{MAX}$ $P_D \le P_{MAX}$	–	2.0	5.0	µA
Reference Voltage	V_{REF}	$3V \le V_{IN} - V_O \le 40V$ $10mA \le I_O \le I_{MAX}$ $P_D \le P_{MAX}$	1.20	1.25	1.30	V
Temperature Stability	ST_T	–	–	0.7	–	% / V_O
Minimum Load Current to Maintain Regulation	$I_{L(MIN)}$	$V_I - V_O = 40V$	–	3.5	12.0	mA
Maximum Output Current	$I_{O(MAX)}$	$V_I - V_O \le 15V$, $P_D \le P_{MAX}$ $V_I - V_O \le 40V$, $P_D \le P_{MAX}$ $T_A = 25°C$	1.0	2.2 0.3	–	A
RMS Noise,% of V_{OUT}	eN	$T_A = +25°C$, $10Hz \le f \le 10KHz$	–	0.003	0.01	% / V_O
Ripple Rejection	RR	$V_O = 10V$, f = 120Hz without C_{ADJ} $C_{ADJ} = 10\mu F$ (Note 3)	66.0	60.0 75.0	–	dB
Long-Term Stability, $T_J = T_{HIGH}$	ST	$T_A = +25°C$ for end point measurements, 1000HR	–	0.3	1.0	%
Thermal Resistance Junction to Case	$R_{\theta JC}$	–	–	5.0	–	°C / W

Note 2: Load and line regulation are specified at constant junction temperature. Change in V_D due to heating effects must be taken into account separately. Pulse testing with low duty is used. (P_{MAX} = 20S)

Note 3: CADJ, when used, is connected between the adjustment pin and ground.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

National Semiconductor

October 2002

LM1117/LM1117I
800mA Low-Dropout Linear Regulator

General Description

The LM1117 is a series of low dropout voltage regulators with a dropout of 1.2V at 800mA of load current. It has the same pin-out as National Semiconductor's industry standard LM317.

The LM1117 is available in an adjustable version, which can set the output voltage from 1.25V to 13.8V with onlv two external resistors. In addition, it is also available in five fixed voltages, 1.8V, 2.5V, 2.85V, 3.3V, and 5V.

The LM1117 offers current limiting and thermal shutdown. Its circuit includes a zener trimmed bandgap reference to assure output voltage accuracy to within ±1%.

The LM1117 series is available in LLP, TO-263, SOT-223, TO-220, and TO-252 D-PAK packages. A minimum of 10µF tantalum capacitor is required at the output to improve the transient response and stability.

Features

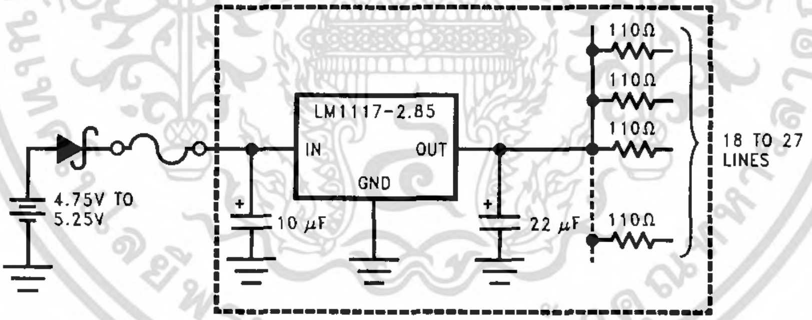
- Available in 1.8V, 2.5V, 2.85V, 3.3V, 5V, and Adjustable Versions
- Space Saving SOT-223 and LLP Packages
- Current Limiting and Thermal Protection
- Output Current 800mA
- Line Regulation 0.2% (Max)
- Load Regulation 0.4% (Max)
- Temperature Range
  - — LM1117 0°C to 125°C
  - — LM1117I −40°C to 125°C

Applications

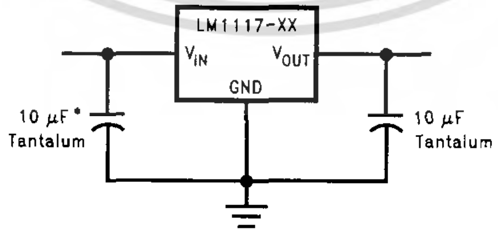
- 2.85V Model for SCSI-2 Active Termination
- Post Regulator for Switching DC/DC Converter
- High Efficiency Linear Regulators
- Battery Charger
- Battery Powered Instrumentation

Typical Application

Active Terminator for SCSI-2 Bus



Fixed Output Regulator



*Required if the regulator is located far from the power supply filter.

10091928

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

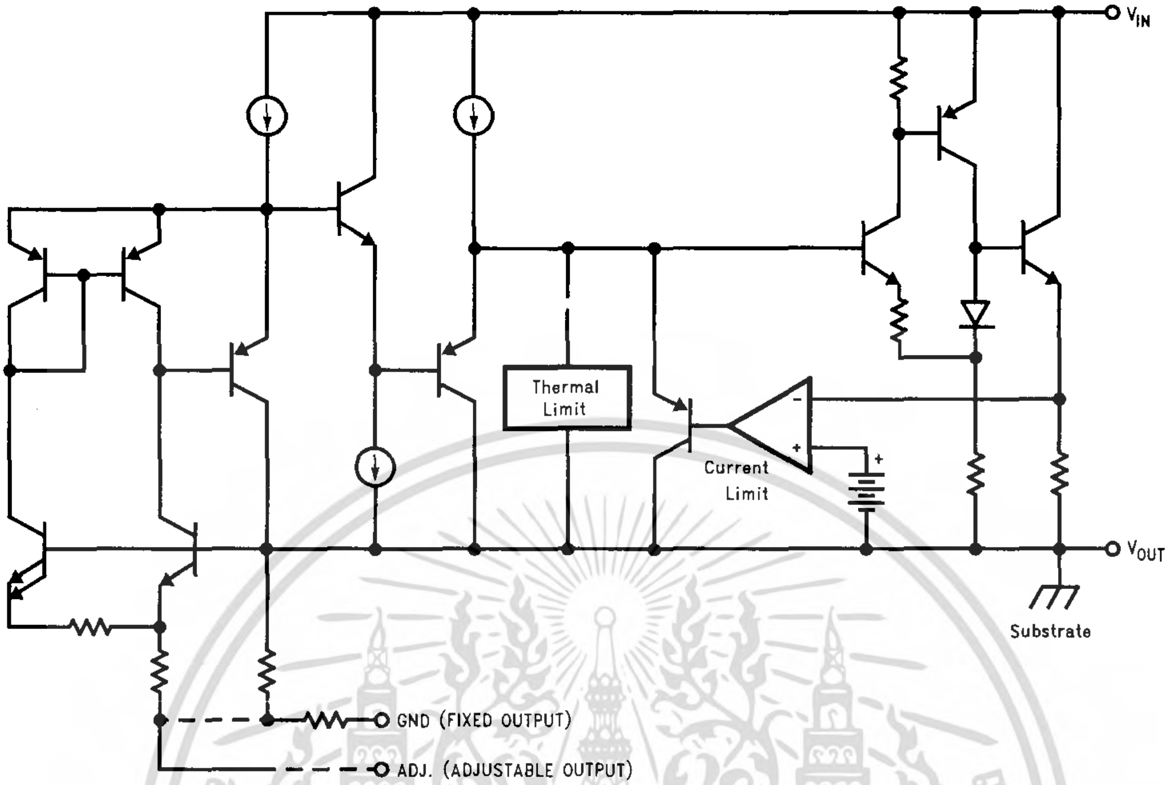
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

Package	Temperature Range	Part Number	Packaging Marking	Transport Media	NSC Drawing
3-lead SOT-223	0˚C to +125˚C	LM1117MPX-ADJ	N03A	Tape and Reel	MP04A
		LM1117MPX-1.8	N12A	Tape and Reel	
		LM1117MPX-2.5	N13A	Tape and Reel	
		LM1117MPX-2.85	N04A	Tape and Reel	
		LM1117MPX-3.3	N05A	Tape and Reel	
		LM1117MPX-5.0	N06A	Tape and Reel	
	−40˚C to +125˚C	LM1117IMPX-ADJ	N03B	Tape and Reel	
		LM1117IMPX-3.3	N05B	Tape and Reel	
		LM1117IMPX-5.0	N06B	Tape and Reel	
3-lead TO-220	0˚C to +125˚C	LM1117T-ADJ	LM1117T-ADJ	Rails	T03B
		LM1117T-1.8	LM1117T-1.8	Rails	
		LM1117T-2.5	LM1117T-2.5	Rails	
		LM1117T-2.85	LM1117T-2.85	Rails	
		LM1117T-3.3	LM1117T-3.3	Rails	
		LM1117T-5.0	LM1117T-5.0	Rails	
3-lead TO-252	0˚C to +125˚C	LM1117DTX-ADJ	LM1117DT-ADJ	Tape and Reel	TD03B
		LM1117DTX-1.8	LM1117DT-1.8	Tape and Reel	
		LM1117DTX-2.5	LM1117DT-2.5	Tape and Reel	
		LM1117DTX-2.85	LM1117DT-2.85	Tape and Reel	
		LM1117DTX-3.3	LM1117DT-3.3	Tape and Reel	
		LM1117DTX-5.0	LM1117DT-5.0	Tape and Reel	
	−40˚C to +125˚C	LM1117IDTX-ADJ	LM1117IDT-ADJ	Tape and Reel	
		LM1117IDTX-3.3	LM1117IDT-3.3	Tape and Reel	
		LM1117IDTX-5.0	LM1117IDT-5.0	Tape and Reel	
8-lead LLP	0˚C to +125˚C	LM1117LDX-ADJ	1117ADJ	Tape and Reel	LDC08A
		LM1117LDX-1.8	1117-18	Tape and Reel	
		LM1117LDX-2.5	1117-25	Tape and Reel	
		LM1117LDX-2.85	1117-28	Tape and Reel	
		LM1117LDX-3.3	1117-33	Tape and Reel	
		LM1117LDX-5.0	1117-50	Tape and Reel	
	−40˚C to 125˚C	LM1117ILDX-ADJ	1117IAD	Tape and Reel	
		LM1117ILDX-3.3	1117I33	Tape and Reel	
		LM1117ILDX-5.0	1117I50	Tape and Reel	
TO-263	0˚C to +125˚C	LM1117SX-ADJ	LM1117SADJ	Tape and Reel	TS3B
		LM1117SX-2.85	LM1117S2.85	Tape and Reel	
		LM1117SX-3.3	LM1117S3.3	Tape and Reel	
		LM1117SX-5.0	LM1117S5.0	Tape and Reel	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

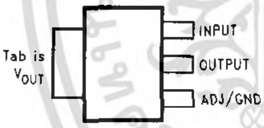
Block Diagram



10091901

Connection Diagrams

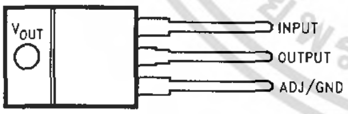
SOT-223



10091904

Top View

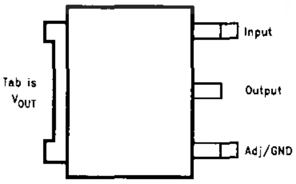
TO-220



10091902

Top View

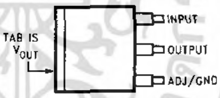
TO-252



10091938

Top View

TO-263



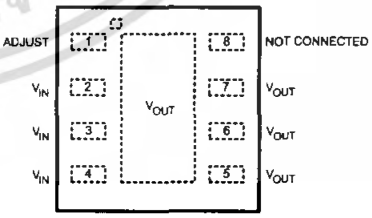
10091944

Top View

10091945

Side View

LLP



10091946

When using the LLP package
Pins 2, 3 & 4 must be connected together and
Pins 5, 6 & 7 must be connected together

Top View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UTRON

UT621024

Rev. 1.5

128K X 8 BIT LOW POWER CMOS SRAM

FEATURES

- Access time : 35/55/70ns (max.)
- Low power consumption :
  Operating : 60/50/40 mA (typical)
  Standby : 2μA (typical) L-version
  1μA (typical) LL-version
- Single 5V power supply
- All inputs and outputs TTL compatible
- Fully static operation
- Three state outputs
- Data retention voltage : 2V (min.)
- Package : 32-pin 600 mil PDIP
  32-pin 450 mil SOP
  32-pin 8mmx20mm TSOP-1
  32-pin 8mmx13.4mm STSOP

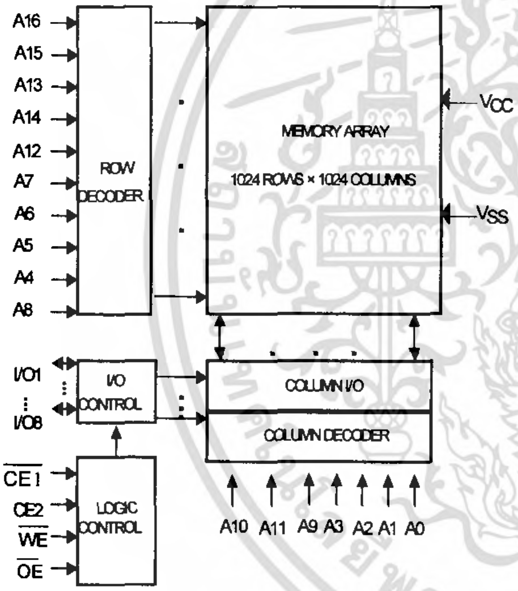
GENERAL DESCRIPTION

The UT621024 is a 1,048,576-bit low power CMOS static random access memory organized as 131,072 words by 8 bits. It is fabricated using high performance, high reliability CMOS technology.

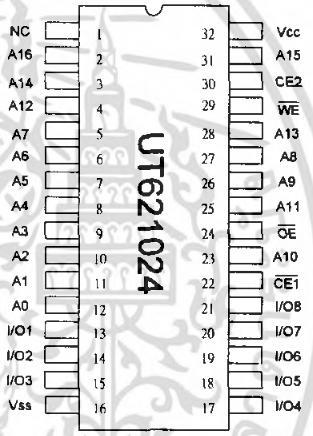
The UT621024 is designed for low power application. It is particularly well suited for battery back-up nonvolatile memory application.

The UT621024 operates from a single 5V power supply and all inputs and outputs are fully TTL compatible.

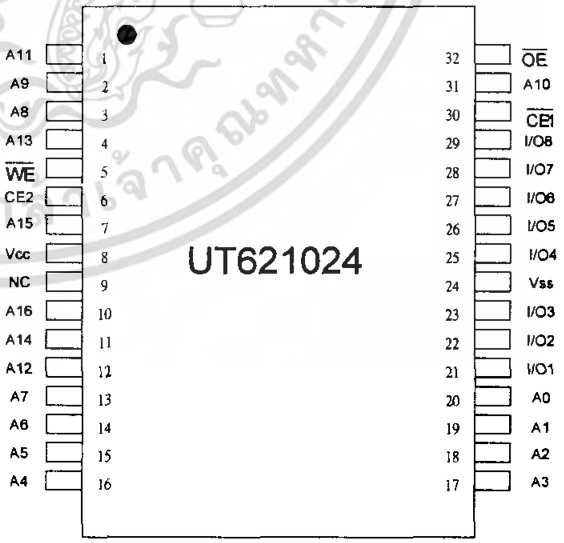
FUNCTIONAL BLOCK DIAGRAM



PIN CONFIGURATION



PDIP / SOP



TSOP-I/STSOP

PIN DESCRIPTION

SYMBOL	DESCRIPTION
A0 - A16	Address Inputs
I/O1 - I/O8	Data Inputs/Outputs
$\overline{CE1}$,CE2	Chip enable 1,2 Inputs
$\overline{WE}$	Write Enable Input
$\overline{OE}$	Output Enable Input
V_{CC}	Power Supply
V_{SS}	Ground
NC	No Connection

UTRON TECHNOLOGY INC.
1F, No. 11, R&D Rd. II, Science-Based Industrial Park, Hsinchu, Taiwan, R. O. C.
TEL: 886-3-5777882 FAX: 886-3-5777919

P80036

จัดทำไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS*

PARAMETER	SYMBOL	RATING	UNIT
Terminal Voltage with Respect to Vss	V_{TERM}	-0.5 to +7.0	V
Operating Temperature	T_A	0 to +70	℃
Storage Temperature	T_{STG}	-65 to +150	℃
Power Dissipation	P_D	1	W
DC Output Current	I_{OUT}	50	mA
Soldering Temperature (under 10 sec)	T_{solder}	260	℃

*Stresses greater than 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 or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to the absolute maximum rating conditions for extended period may affect device reliability.

TRUTH TABLE

MODE	$\overline{CE1}$	CE2	$\overline{OE}$	$\overline{WE}$	I/O OPERATION	SUPPLY CURRENT
Standby	H	X	X	X	High - Z	I_{SB}, I_{SB1}
Standby	X	L	X	X	High -Z	I_{SB}, I_{SB1}
Output Disable	L	H	H	H	High - Z	I_{CC}
Read	L	H	L	H	D_{OUT}	I_{CC}
Write	L	H	X	L	D_{IN}	I_{CC}

Note: H = V_{IH}, L=V_{IL}, X = Don't care.

DC ELECTRICAL CHARACTERISTICS (V_{CC} = 5V± 10%, TA = 0℃ to 70℃)

PARAMETER	SYMBOL	TEST CONDITION		MIN.	TYP.	MAX.	UNIT
Input High Voltage	V_{IH}			2.2	-	Vcc+0.5	V
Input Low Voltage	V_{IL}			- 0.5	-	0.8	V
Input Leakage Current	I_{IL}	$V_{SS} \leqq V_{IN} \leqq V_{CC}$		- 1	-	1	µA
Output Leakage Current	I_{OL}	$V_{SS} \leqq V_{I/O} \leqq V_{CC}$ $\overline{CE1} = V_{IH}$ or CE2 = V_{IL} or $\overline{OE} = V_{IH}$ or $\overline{WE} = V_{IL}$		- 1	-	1	µA
Output High Voltage	V_{OH}	I_{OH} = - 1mA		2.4	-	-	V
Output Low Voltage	V_{OL}	I_{OL}= 4mA		-	-	0.4	V
Average Operating Power Supply Courrent	I_{CC}	Cycle time=min, 100% duty, $\overline{CE1} = V_{IL}$, CE2 = V_{IH}, $I_{I/O}$ = 0mA	-35	-	60	100	mA
			-55	-	50	85	mA
			-70	-	40	70	mA
	I_{CC1}	Cycle time=1µs,100% duty,$I_{I/O}$=0mA $\overline{CE1} \leqq$0.2V,CE2$\geqq V_{CC}$-0.2V, other pins at 0.2V or V_{CC}-0.2V,		-	-	10	mA
Standby Power Supply Current	I_{SB}	$\overline{CE1} = V_{IH}$ or CE2 = V_{IL} other pins at 0.2V or V_{CC}-0.2V,		-	-	3	mA
	I_{SB1}	$\overline{CE1} \geqq V_{CC}$-0.2V or CE2$\leqq$0.2V other pins at 0.2V or V_{CC}-0.2V,	- L	-	2	100 40*	µA
			- LL	-	1	50 15*	µA

*Those parameters are for reference only under 50℃

สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประ ด้านการค้า
ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้