AN21.0
Rev 0.00
Jul 15, 2005

X5043, X5045 System Supervisors Manage 8051 Type Microcontrollers

Both the X5043 and X5045 feature a power on reset circuit, low voltage reset controller, programmable watchdog timer, and 4K bits of high speed, three-wire serial, nonvolatile EEPROM in a single 8-pin package.

Power-On-Reset (POR)

The X5043, X5045 Power-On-Reset circuit holds the RESET pin active for 250ms when the system power is applied. This prevents the microcontroller from operating while the power supply is stabilizing. This improves the reliability of system start up.

Low Voltage Reset (LVR)

During operation, the low voltage reset circuit monitors the supply voltage. If the voltage drops below a specied minimum, the X5043, X5045 drives the RESET pin active. This stops the operation of the microcontroller to prevent unexpected operation. If the microcontroller operates at voltages that are too low, the microcontroller or a peripheral device may fail, causing the system to "lock-up" or resulting in data corruption.

Watchdog Timer

While the POR and LVR circuits help prevent system problems, the Watchdog Timer helps the system recover when there is a problem. The Watchdog Timer works by resetting the system when there is a time-out. The micro-controller continually resets the timer, as part of the software loop, before the timer times-out. If there is ever a software problem, such as an innite loop or an operation that waits for a peripheral device, the Watchdog timer expires and resets the microcontroller.

Hardware Implementation

The circuit shown in Figure. 1 includes both a manual and X5043 controlled reset. R1 serves as a pull-up resistor for the X5043 open-drain (i.e. active LOW) reset output. The 2N7000 N-MOSFET is used to invert the active LOW reset, to directly control the 8031 RST pin. The circuit shown in Figure. 2 has both a manual and X5045 controlled reset. The circuit in Figure 2 is preferable because the X24045 has the correct reset polarity for the 8051.

Software Implementation

The following routines are included for implementing an interface to the X5043, X5045:

wren_cmd – This routine sets the write enable latch, which must be set before writing to either the EEPROM memory array or the status register. The WEL bit is automatically reset after a write operation.

wrdi_cmd – This command resets the write enable latch.

wrsr_cmd – This operation writes the watchdog timeout period bits (WD0, WD1) and the Block Protect bits (BP0, BP1) in the status register.

rdsr_cmd – This routine reads the status register.

byte_write – This command writes a single byte to the EEPROM memory array.

byte_read – This command reads a single byte from the EEPROM memory array.

page_write – This operation writes 3 consecutive bytes to the EEPROM memory array. It can easily be modied to write an entire page (maximum of 16 bytes).

sequ_read – This routine reads three consecutive bytes from the EEPROM memory array. It can be easily modied to read any number of bytes.

rst_wdog – This routine is used to reset the watchdog timer without sending a command.

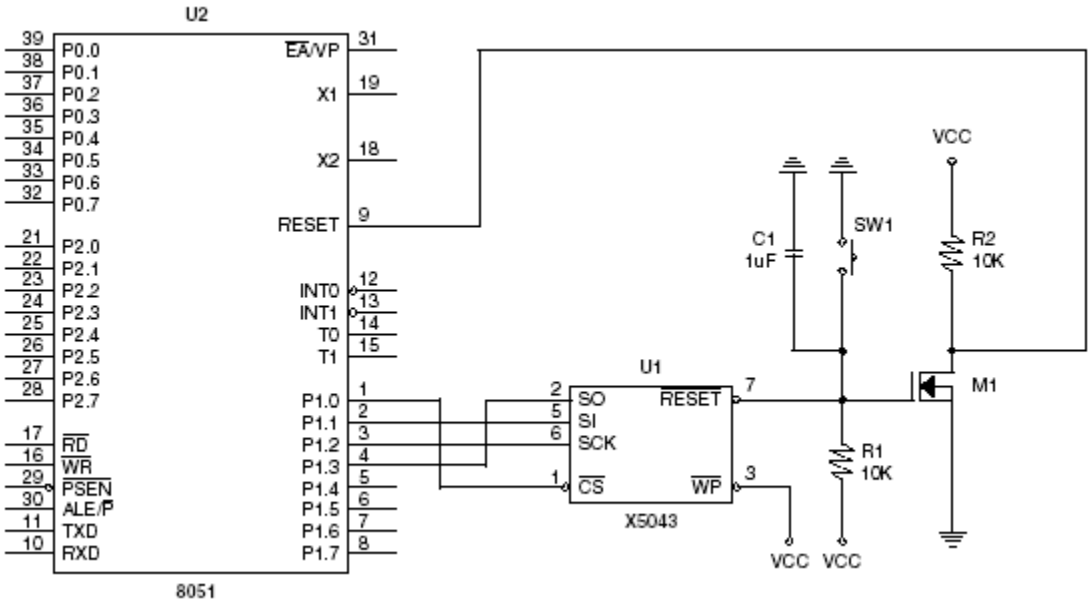

FIGURE 1. CONNECTING AN X5043 TO AN 8051 MICROCONTROLLER, WITH MANUAL RESET CONTROL

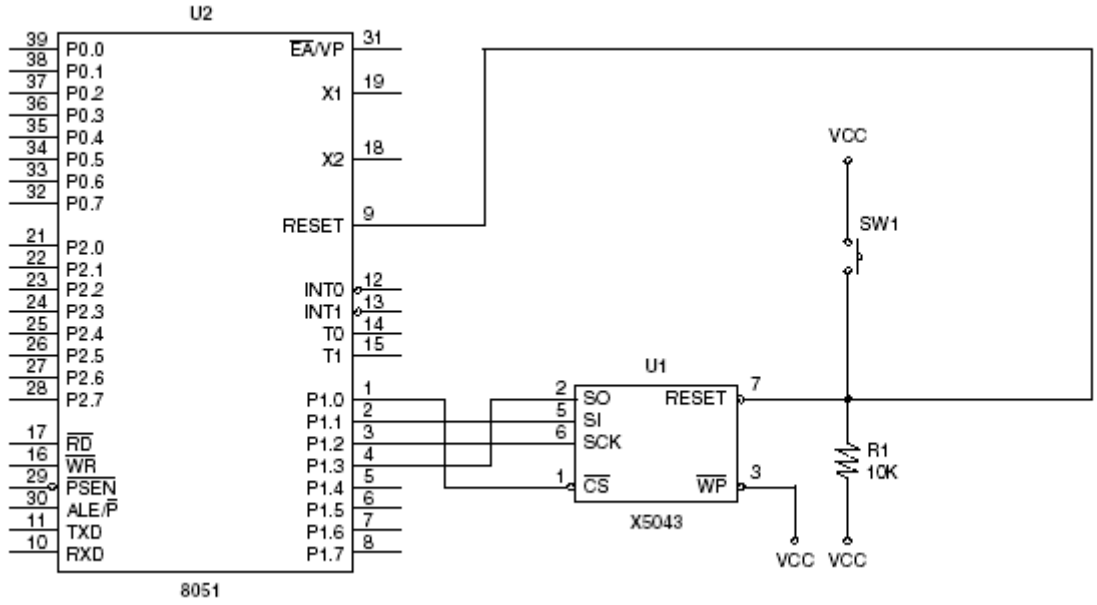

FIGURE 2. CONNECTING AN X5045 TO AN 8051 MICROCONTROLLER, WITH MANUAL RESET CONTROL

```
;*******************************************************************************************
;*Copyright (c) 1994 Xicor, Inc.
;*AUTHOR: Richard Downing
;*******************************************************************************************
;* The purpose of this code is to provide routines to interface the Xicor X5043 with the 8031
;* microcontroller. The interface uses the 8031's general purpose parallel port 1 and connects
;* P1.0 to the chip select line (/CS), P1.1 to the serial input data line (SI), P1.2 to the
;* serial clock line (SCK) and P1.3 to the serial output data line (SO).
;*
;* All X5043 commands are provided. These are :-
;*
;*     1. Set Write Enable Latch
;*     2. Reset Write Enable Latch
;*     3. Write Status Register
;*     4. Read Status Register
;*     5. Single Byte Write
;*     6. Single Byte Read
;*     7. Page Write
;*     8. Sequential Read
;*     9. Reset Watchdog Timer
;*
;* The code writes 00H to the Status Register; reads the Status Register; writes 11H to
;* address 55H in Byte Mode; performs a single Byte Read from address 55H; writes 22H,
;* 33H, 44H to addresses 1F0H, 1F1H, 1F2H in Page Mode; performs a Sequential Read
;* from addresses 1F0H, 1F1H, 1F2H; and resets watchdog timer. This code can also be used with
;* the X5045 which is identical to the X5043, except for its RESET output polarity.
;*******************************************************************************************

;* CONSTANTS
cs              bit    P1.0   ; Port 1 bit 0 used for chip select (/CS)
si              bit    P1.1   ; Port 1 bit 1 used for serial input (CI)
sck             bit    P1.2   ; Port 1 bit 2 used for serial clock (SCK)
so              bit    P1.3   ; Port 1 bit 3 used for serial output (SO)
WREN_INST       equ    06H    ; Write enable latch instruction (WREN)
WRDI_INST       equ    04H    ; Write disable latch instruction (WRDI)
WRSR_INST       equ    01H    ; Write status register instruction (WRSR)
RDSR_INST       equ    05H    ; Read status register instruction (RDSR)
WRITE_INST      equ    02H    ; Write memory instruction (WRITE)
READ_INST       equ    03H    ; Read memory instruction (READ)
BYTE_ADDR       equ    55H    ; Memory address for byte mode operations
BYTE_DATA       equ    11H    ; Data byte for byte write operation
PAGE_ADDR       equ    1F0H   ; Memory address for page mode operations
PAGE_DATA1      equ    22H    ; 1st data byte for page write operation
PAGE_DATA2      equ    33H    ; 2nd data byte for page write operation
PAGE_DATA3      equ    44H    ; 3rd data byte for page write operation
STATUS_REG      equ    00H    ; Status register
MAX_POLL        equ    99H    ; Maximum number of polls
INIT_STATE      equ    09H    ; Initialization value for control ports
USER            equ    030H   ; Address location of User Code
;*******************************************************************************************
;* INTERNAL RAM

STACK_TOP    equ    060H   ; Stack top

;*******************************************************************************************
;* CODE

             ORG    0000H  ; Reset vectors to this location
             ljmp   main

               ORG    0100H
```

```
main:
     mov     SP,#STACK_TOP      ; Initialize stack pointer
     clr     EA                 ; Disable interupts
     mov     P1, #INIT_STATE    ; Init control lines (/CS & SO =1, SCK & SI =0)
     lcall   wren_cmd           ; Set write enable latch
     lcall   wrsr_cmd           ; Write 00H to status register
     lcall   wren_cmd           ; Set write enable latch
     lcall   byte_write         ; Write 11H to address 55H (Byte Write)
     lcall   byte_read          ; Read from address location 55H (Byte Read)
     lcall   wren_cmd           ; Set write enable latch
     lcall   page_write         ; Page write 22H/33H/44H to addresses 1F0/1/2H
     lcall   sequ_read          ; Seq. Read from address locations 1F0/1/2H
     lcall   rst_wdog           ; Reset Watchdog timer
     jmp     USER

;*******************************************************************************************
;* Name: WREN_CMD
;* Description: Set write enable latch
;* Function: This routine sends the command to enable writes to the EEPROM memory array or
;* status register
;* Calls: outbyt
;* Input: None
;* Outputs: None
;* Register Usage: A
;*******************************************************************************************
wren_cmd:
     clr     sck               ; Bring SCK low
     clr     cs                ; Bring /CS low
     mov     A, #WREN_INST
     lcall   outbyt            ; Send WREN instruction
     clr     sck               ; Bring SCK low
     setb    cs                ; Bring /CS high
     ret

;*******************************************************************************************
;* Name: WRDI_CMD
;* Description: Reset write enable latch
;* Function: This routine sends the command to disable writes to the EEPROM memory array or
;* status register
;* Calls: outbyt
;* Input: None
;* Outputs: None
;* Register Usage: A
;*******************************************************************************************
wrdi_cmd:
     clr       sck               ; Bring SCK low
     clr       cs                ; Bring /CS low
     mov       A, #WRDI_INST
     lcall     outbyt            ; Send WRDI instruction
     clr       sck               ; Bring SCK low
     setb      cs                ; Bring /CS high
     ret
```

```
;*******************************************************************************************
;* Name: WRSR_CMD
;* Description: Write Status Register
;* Function: This routine sends the command to write the WD0, WD1, BP0 and BP0 EEPROM
;* bits in the status register
;* Calls: outbyt, wip_poll
;* Input: None
;* Outputs: None
;* Register Usage: A
;*******************************************************************************************
wrsr_cmd:
     clr    sck              ; Bring SCK low
     clr    cs               ; Bring /CS low
     mov    A, #WRSR_INST
     lcall  outbyt           ; Send WRSR instruction
     mov    A, #STATUS_REG
     lcall  outbyt           ; Send status register
     clr    sck              ; Bring SCK low
     setb   cs               ; Bring /CS high
     lcall  wip_poll         ; Poll for completion of write cycle
     ret

;*******************************************************************************************
;* Name: RDSR_CMD
;* Description: Read Status Register
;* Function: This routine sends the command to read the status register
;* Calls: outbyt, inbyt
;* Input: None
;* Outputs: A = status register
;* Register Usage: A
;*******************************************************************************************
rdsr_cmd:
     clr    sck              ; Bring SCK low
     clr    cs               ; Bring /CS low
     mov    A, #RDSR_INST
     lcall  outbyt           ; Send RDSR instruction
     lcall  inbyt            ; Read status register
     clr    sck              ; Bring SCK low
     setb   cs               ; Bring /CS high
     ret

;*******************************************************************************************
;* Name: BYTE_WRITE
;* Description: Single Byte Write
;* Function: This routine sends the command to write a single byte to the EEPROM memory array
;* Calls: outbyt, wip_poll
;* Input: None
;* Outputs: None
;* Register Usage: A, B
;*******************************************************************************************
byte_write:
     mov    DPTR, #BYTE_ADDR   ; Set address of byte to be written
     clr    sck                ; Bring SCK low
     clr    cs                 ; Bring /CS low
     mov    A, #WRITE_INST
     mov    B, DPH
     mov    C, B.0
     mov    ACC.3, C
     lcall  outbyt             ; Send WRITE instruction including MSB of address
     mov    A, DPL
     lcall  outbyt             ; Send 8 LSBs of address
```

```
     mov    A, #BYTE_DATA
     lcall  outbyt            ; Send data byte
     clr    sck               ; Bring SCK low
     setb   cs                ; Bring /CS high
     lcall  wip_poll          ; Poll for completion of write cycle
     ret

;*******************************************************************************************
;* Name: BYTE_READ
;* Description: Single Byte Read
;* Function: This routine sends the command to read a single byte from the EEPROM memory array
;* Calls: outbyt, inbyt
;* Input: None
;* Outputs: A = read byte
;* Register Usage: A, B
;*******************************************************************************************
byte_read:
     mov    DPTR, #BYTE_ADDR     ; Set address of byte to be read
     clr    sck                  ; Bring SCK low
     clr    cs                   ; Bring /CS low
     mov    A, #READ_INST
     mov    B, DPH
     mov    C, B.0
     mov    ACC.3, C
     lcall  outbyt              ; Send READ instruction including MSB of address
     mov    A, DPL
     lcall  outbyt              ; Send 8 LSBs of address
     lcall  inbyt               ; Read data byte
     clr    sck                 ; Bring SCK low
     setb   cs                  ; Bing /CS high
     ret

;*******************************************************************************************
;* Name: PAGE_WRITE
;* Description: Page Write
;* Function: This routine sends the command to write three consecutive bytes to the EEPROM
;* memory array using page mode
;* Calls: outbyt, wip_poll
;* Input: None
;* Outputs: None
;* Register Usage: A, B
;*******************************************************************************************
page_write:
     mov    DPTR, #PAGE_ADDR     ; Set address of 1st byte to be written
     clr    sck                  ; Bring SCK low
     clr    cs                   ; Bring /CS low
     mov    A, #WRITE_INST
     mov    B, DPH
     mov    C,  B.0
     mov    ACC.3, C
     lcall  outbyt               ; Send WRITE instruction including MSB of address
     mov    A, DPL
     lcall  outbyt               ; Send 8 LSBs of address
     mov    A, #PAGE_DATA1
     lcall  outbyt               ; Send 1st data byte
     mov    A, #PAGE_DATA2
     lcall  outbyt               ; Send 2nd data byte
     mov    A, #PAGE_DATA3
     lcall  outbyt               ; Send 3rd data byte
     clr    sck                  ; Bring SCK low
     setb   cs                   ; Bring /CS high
```

```
     lcall  wip_poll              ; Poll for completion of write cycle
     ret

;*******************************************************************************************
;* Name: SEQU_READ
;* Description: Sequential Read
;* Function: This routine sends the command to read three consecutive bytes from the EEPROM
;* memory array using sequential mode
;* Calls: outbyt, inbyt
;* Input: None
;* Outputs: A = last byte read
;* Register Usage: A, B
;*******************************************************************************************
sequ_read:
     mov    DPTR, #PAGE_ADDR   ; Set address of 1st byte to be read
     clr    sck                ; Bring SCK low
     clr    cs                 ; Bring /CS low
     mov    A, #READ_INST
     mov    B, DPH
     mov    C, B.0
     mov    ACC.3, C
     lcall  outbyt             ; Send READ instruction with MSB of address
     mov    A, DPL
     lcall  outbyt             ; Send low order address byte
     lcall  inbyt              ; Read 1st data byte
     lcall  inbyt              ; Read 2nd data byte
     lcall  inbyt              ; Read 3rd data byte
     clr    sck                ; Bring SCK low
     setb    cs                ; Bring /CS high
     ret

;*******************************************************************************************
;* Name: RST_WDOG
;* Description: Reset Watchdog Timer
;* Function: This routine resets the watchdog timer without sending a command
;* Calls: None
;* Input: None
;* Outputs: None
;* Register Usage: None
;*******************************************************************************************
rst_wdog:
     clr   cs            ; Bring /CS low to reset watchdog timer
     setb  cs            ; Bring /CS high
     ret

;*******************************************************************************************
;* Name: WIP_POLL
;* Description: Write-In-Progress Polling
;* Function: This routine polls for completion of a nonvolatile write cycle by examining the
;* WIP bit of the status register
;* Calls: rdsr_cmd
;* Input: None
;* Outputs: None
;* Register Usage: R1, A
;*******************************************************************************************
wip_poll:
       mov    R1, #MAX_POLL       ; Set maximum number of polls
wip_poll1:
       jnb   ACC.0, wip_poll2     ; If  WIP bit '0' write cycle completed
       djnz  R1, wip_poll1        ; If  WIP bit '1' continue polling
wip_poll2:
```

```
       ret

;*******************************************************************************************
;* Name: OUTBYT
;* Description: Sends byte to EEPROM
;* Function: This routine shifts out a byte, starting with the MSB, to the EEPROM
;* Calls: None
;* Input: A = byte to be sent
;* Outputs: None
;* Register Usage: R0, A
;*******************************************************************************************
outbyt:
     mov   R0, #08        ; Set bit counter to eight
outbyt1:
     clr   sck            ; Bring SCK low
     rlc   A              ; Shift byte left through carry
     mov   si, C          ; Send data bit in carry
     setb  sck            ; Bring SCK high
     djnz  R0, outbyt1    ; Finish if  last data bit
     clr   si             ; Place SI in known condition
     ret

;*******************************************************************************************
;* Name: INBYT
;* Description: Recieves byte from EEPROM
;* Function: This routine recieves a byte, MSB first, from the EEPROM
;* Calls: None
;* Input: None
;* Outputs: A = recieved byte
;* Register Usage: R0, A
;*******************************************************************************************
inbyt:
     mov    R0, #08      ; Set bit counter to eight
inbyt1:
     setb   sck          ; Bring SCK high
     clr    sck          ; Bring SCK low
     mov    C, so        ; Receive data bit and store in carry
     rlc    A            ; Shift byte left through carry
     djnz   R0, inbyt1   ; Finish if last data bit
     ret

  END
```

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
6. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.4.0-1 November 2017)

RENESAS

SALES OFFICES

Renesas Electronics Corporation

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700, Fax: +44-1628-651-804

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709 Quantum Plaza, No.27 ZhichunLu, Haidian District, Beijing, 100191 P. R. China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, 200333 P. R. China
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338

Colophon 7.0