

To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of 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. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may 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.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
 - "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
 - "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
 - "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be 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, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

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

H8/300L

Usage of Port to Implement I2C (I2Cport)

Introduction

This application note describes the necessary information and usage specification when implementing I²C with I/O ports of SLP series.

The I²C-bus supports any IC fabrication process (NMOS, CMOS, bipolar). Two wires, serial data (SDA) and serial clock (SCL), carry information between the devices connected to the bus. All I²C-bus compatible devices incorporate an on-chip interface, which allows them to communicate directly with each other via the I²C-bus.

While SLP series do not possess dedicated on-chip I²C support hardware, this serial bus can be simulated by software control of its two I/O pins.

*Please refer to Application Note on Interfacing with EEPROM with Emulating I²C for an example of implementation of SLP series I/O ports as the I²C bus ports.

Target Device

SLP – H8/38024

Contents

1. I ² C General Characteristics.....	3
2. Hardware Configuration	3
3. Timing.....	4
4. Bit Transfer.....	4
5. START and STOP conditions	5
6. Errors.....	5
7. Software Routines.....	6
Reference.....	6

1. I²C General Characteristics

*Please refer to Application Note on SPI and I²C for I²C General view.

2. Hardware Configuration

It is important to make sure that any devices on an I²C bus have the capability to pull the bus signals low. When the slave device wants to acknowledge the master polling, it must pull the SDA signal low. When a slow device cannot keep pace with a faster device, it will hold the SCL signal low to slow down the speed of communication.

When more than one device attempts to become master of the bus, the one that holds SDA low longer wins the arbitration. For this reason, it is critical that each I²C device releases the bus after it finishes its activity and allows other devices to pull the bus signal low as needed. To implement this feature, I/O ports of the SLP series are configured as open-collector drivers.

To drive the signal line low, the direction register of the I/O port is set to be output and the data register is set to zero. To drive the signal line high, the data register is set to one, which prevents other devices from pulling the signal line low. The direction register of that I/O port is set to zero so that the I/O port is configured as an input. Then, this port will show logic one because of the external pull-up resistor. This approach is implemented in routines: SciOut() and SdlOut(), shown in below:

```

/* Both SDA_DATA_REG and SCL_DATA_REG are the data register for each respective I/O
port. It may change to map to other ports. The setting example as following:
    #define SDA_DATA_REG          P_IO.PDR7.BYTE
    #define SCL_DATA_REG          P_IO.PDR8.BYTE
SCL_IO_REG and SDA_IO_REG are the control register for each respective I/O port. The
setting example as following and is changeable by user.
    #define SDA_IO_REG            P_IO.PCR7.BYTE
    #define SCL_IO_REG            P_IO.PCR8.BYTE
*/
/* Drive SCL bus */
void SclOut (unsigned char status)
{
    if (status == LOW)
    {
        SCL_DATA_REG = 0;           //Drive Port LOW
        SCL_IO_REG |= SCL_IO_SET_BIT; //Port is output
    }
    else
    {
        SCL_DATA_REG = 1;           //Port is Input & using external
        //pull-up resistor to go high
        SCL_IO_REG |= SCL_IO_SET_BIT; //Port is output
    }
}
/* Drive SDA bus */
void SdaOut (unsigned char status)
{
    if (status == LOW)
    {
        SDA_DATA_REG = 0;           //Drive Port LOW
        SDA_IO_REG |= SDA_IO_SET_BIT; //Port is output
    }
    else
    {
        SDA_DATA_REG = 1;           //Port is Input & using external pull-up
        //resistor to go high
        SDA_IO_REG |= SDA_IO_SET_BIT; //Port is output
    }
}

```

3. Timing

Another consideration is the time. The timing for the I²C clock does not have to be very accurate. When a slow device is connected to the bus, it will slow down a higher-speed device. There are many ways to set up a timer for the clock. NOP commands or a delay for loop can be used to create a delay in software. The built-in timer in the MCU can also be employed. The timer interrupt can be used to flip the clock output.

However, when setting the clock, you should pay attention to some of the important parameters, which will affect the I²C protocol. Parameters such as rise time, fall time, hold time and setup time for all the conditions such as START, STOP etc. must be appropriately set according to the specification of the device being used (min and max time). Thus, it is important for user to check the AC characteristics for the device to ensure the result of I²C implementation.

I²C does not specify any minimum bus frequency but provides two modes of operation, the STANDARD MODE up to 100 KHz and the FAST-MODE up to 400 KHz. Thus, port implementation is possible.

I²C also does not specify any timeout limit. Thus, user may need to specify the timeout limit for transmitting or receiving data to improve the efficiency of the protocol.

4. Bit Transfer

Due to the variety of different technology devices (CMOS, NMOS, bipolar) which can be connected to the I²C-bus, the levels of the logical '0' (LOW) and '1' (HIGH) are not fixed and depend on the associated level of V_{cc}. One clock pulse is generated for each data bit transferred.

One of the commonly used logic when sending the bits is '0'(LOW) and '1'(HIGH) as the following example:

```
void SendBit (unsigned char data_byte)
{
    if (data_byte != 0)
    {
        SdaOut (HIGH);
    }
    else
    {
        SdaOut (LOW);
    }
}
```

The number of data bytes transferred between the START and STOP condition from transmitter to receiver is not limited. Each byte, which must be eight bits long, is transferred serially with the most significant bit first, and is followed by an acknowledge bit.

5. START and STOP conditions

Within the procedure of the I²C-bus, unique situations arise which are defined as START (S) and STOP (P) conditions. One of the major concerns when generating these conditions is the timing. As mentioned earlier, attention should be given, particularly on the hold time and set up time.

As a transmitter, the device must provide an internal minimum delay time of 300ns to bridge the undefined region of falling edge of SCL to avoid unintended generation of START and STOP conditions.

Generally, the START and STOP conditions with delay (device and clock dependent) as below:

```

void SendStartBit(void)
{
    SclOut(HIGH);    //SCL && SDA must be HIGH to indicate bus-free.
    SdaOut(HIGH);
    SdaOut(LOW);

    Delay();
    SclOut(LOW);
    Delay();
}

void SendStopBit(void)
{
    SdaOut(LOW);
    Delay();
    SclOut(HIGH);
    Delay2x();
    SdaOut(HIGH);
}
    
```

6. Errors

Any transfer can be aborted by either the slave or the master. The master can issue a STOP condition and the slave can withhold acknowledgement after any byte to terminate the transfer. If the device detects an error, it will withhold acknowledgement. Withholding acknowledgement is required for the last byte in a read operation under the I²C specification

A device may decide to generate an error indication for one or more of the following reasons:

- Device is not ready to process the request for data (either read or write)
- Device does not recognize the command code or function requested
- Device does not permit the command code or function requested
- Overflow or underflow condition
- Incorrect size of data in a block read/write transfer
- Unrecognized or unsupported data transfer protocol used in transaction
- Any other known or unknown error condition

7. Software Routines

There are only two routines needed to implement an I2C depending on the on the functionality. There are `i2cRead()`, or `i2cWrite()` or both `i2cRead()` and `i2cWrite()`. Obviously, an LCD driver is only a receiver, while a memory or I/O chip can be both transmitter and receiver.

`i2cWrite()` is called with:-

- i. the slave address to be written,
- ii. the start address of the data buffer where the data to be written is located, and
- iii. the number of data bytes to write.

`i2cRead()` sends out :-

- i) the address of the slave, and
- ii) the number of data bytes the master is requesting to read.

If this operation is successful, it will send an acknowledgment upon receiving or writing and then proceed to the following operation.

*Please refer to Application Note on Interfacing with EEPROM with emulating I²C for example of various types of read and write operation available.

Reference

1. *The I²C-Bus Specification (Version 2.1)*, January 2000, Philips Semiconductor
2. *H8/38024 Series, H8/38024F-ZTZT Hardware Manual(version 2.0)*, 20 Feb 2002, Hitachi Ltd.
3. Leonard Haile, *Hitachi H8/3437 Series Microcontroller I2C Peripheral- A Practical SMBus/I2C Firmware Design Guide (Revision 1.2)*, 12 June 1998, Hitachi Semiconductor (America) Inc.
4. <http://www.esacademy.com/faq/i2c/>

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Sep.03	—	First edition issued

Keep safety first in your circuit designs!

1. Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage. Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
2. Renesas Technology Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.
The information described here may contain technical inaccuracies or typographical errors. Renesas Technology Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (<http://www.renesas.com>).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corporation for further details on these materials or the products contained therein.