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 SLP Series

Auto Baud Rate Detection (AutoBaud)

Introduction

This application note describes the software implementation of baud rate detection of incoming data. The SLP MCU is used to demonstrate the detection of 1200, 4800, 9600, and 19200 bps.

This automatic detection is useful for establishing communication link between two devices. The slave device will be able to detect the baud rate of master controller and adjust accordingly.

This protocol can be implemented on any MCU that carries an asynchronous serial port with a baud rate generator.

In this application note, the protocol is demonstrated using the ALE300L emulator (the SLP MCU H8/38024) connected to the general application board. The SLP MCU is emulating as a slave device, whereas the PC (using the built-in HyperTerminal) acts as the master controller.

Target Device

H8/38024

Contents

1. Theory

1.1 Detection Algorithm

The detection algorithm is to read the sequence of bit received, based on a preset baud rate, and determine the incoming data rate.

In this example, the incoming data from the master is predefined as the “RETURN” character (0x0D), and the initial baud rate is preset to 9600 bps.

The detection algorithm can be classified into three main methods.

1. Baud < 1200
2. Baud > 1200 and baud <= 9600
3. Baud > 9600

The following are the communication mode settings:

Start Bit: 1
Data Bit: 8
Stop Bit: 1
Parity Bit: None
Flow Control: None

For the <RETURN> character (0x0D)

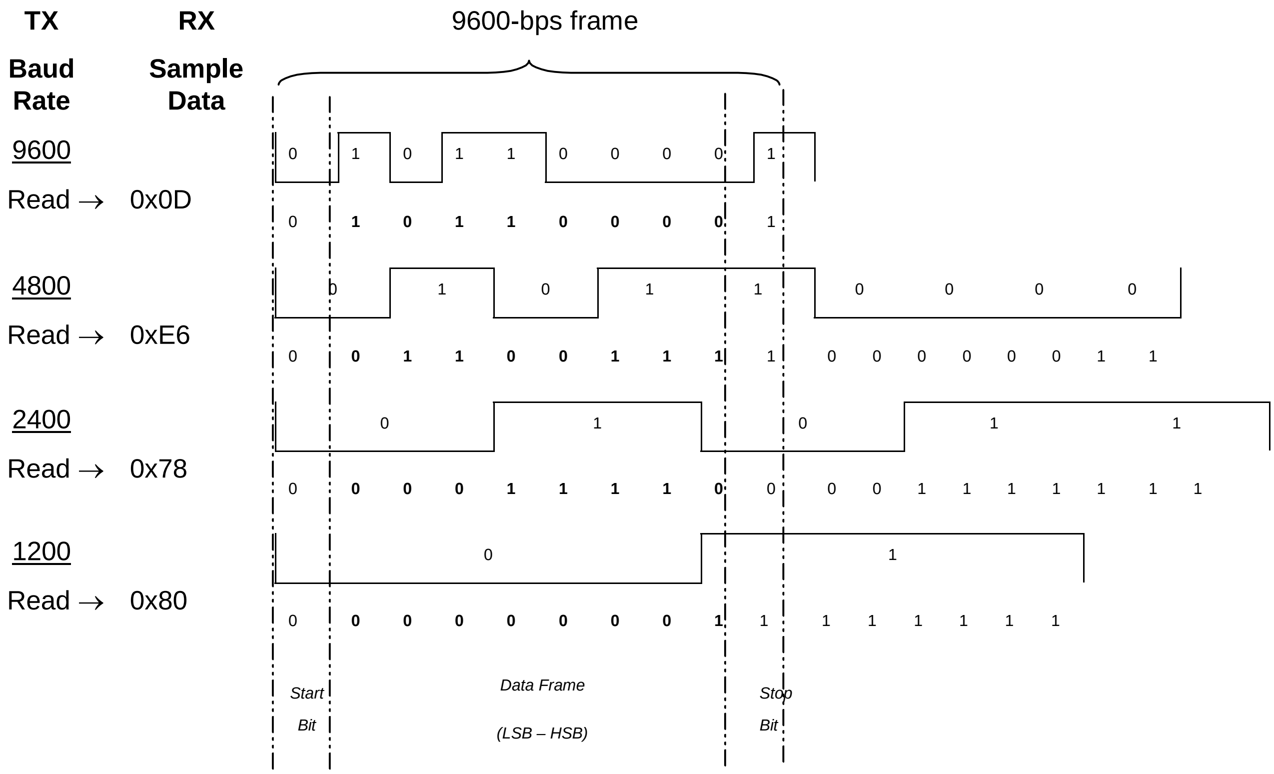
0	1	0	1	1	0	0	0	0	0
Start Bit	Data – 0x0D								Stop Bit

If data is sent based on 9600 baud, the receiver will receive the same data as the transmitted data if it is set at the same baud rate. This is because the incoming data is sampled correctly, upon the activation of start bit. The serial bus is normally in the mark state (high level). When a space (low level) is detected, identify as the start bit, the incoming serial data will be sampled. In SCI3 of the SLP MCU, the data is sampled on the 8th pulse of clock with the frequency that is 16 times larger than the bit rate (Data latched at the center of bit).

It will receive (sample) different patterns of data from the transmitter when the transmitter is set to a different baud rate.

1.1.1 Baud >= 1200 and Baud <= 9600

The diagram below will illustrate the read data when 0x0D (RETURN) is send at various baud rates.



The transmitter's serial data stream, which is based on the different baud rate, will be fed into the 9600-bps receiver. Data will be sampled at this rate. The first data bit will be the LSB (least significant bit).

Based on the above illustration, when data is transmitted at 4800 bps (half slower than that of the receiver of 9600 bps), the initial start bit of transmitter will be lengthened. This will be treated as the start bit and first data bit in the receiver of 9600-bps frame.

Thus, the following data are received at 9600 bps:

- At 9600 bps, received data is 0x0D.
- At 4800 bps, received data is 0xE6.
- At 2400 bps, received data is 0x78.
- At 1200 bps, received data is 0x80.

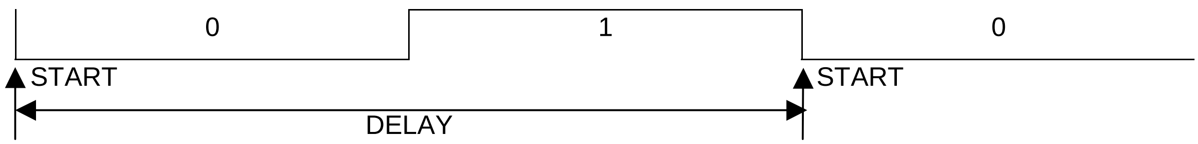
1.1.2 Baud < 1200

For the baud rate below 1200 (600 & 300 baud), the receiver will read the same data pattern of 0x00. This will not prevent the detection procedure. The <RETURN> (0x0D) has several transitions from high to low, which can signify a mark (start) of newly received byte at the higher baud rate receiver (In this case, 9600 bps).

In other words, the high baud rate receiver will be able to receive two more bytes of data from the low baud rate transmitter.

If the timing delay of the "new" byte is measured, the baud rate of transmitter can be predicted.

The following illustrates the initial bit stream of transmitter and the calculation of delay:



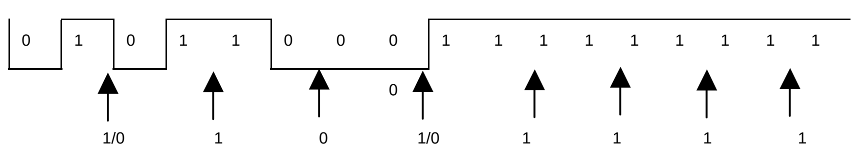
600 bps	16 '0'	16 '1'	16 '0'
	32 bit delay at 9600 Baud = 32* 1/9600 = 3.33 ms		
300 bps	32 '0'	32 '1'	32 '0'
	64 bit delay at 9600 Baud = 64* 1/9600 = 6.66ms		

1.1.3 Baud > 9600

For the baud rate above 9600 (19200 baud), the receiver will read different patterns, as the sampling window will capture the data transitions. The transitions of (0 to 1) and (1 to 0) may be interpret as '0' or '1'.

Thus, in this case,

TX Baud Rate	RX Sample Data
19200	F2, F3, FA, or FB



The sampled data can fall into few possibilities:

- 0xF2
- 0xF3
- 0xFA
- 0xFB

1.2 MCU SCI Setting

The serial communication interface (SCI) is set to

- 9600 baud
- 1 start bit
- 8 data bits
- 1 stop bit
- No parity bit

When the baud rate is determined, the SCI will have to set the following:

- CKS1 (bit 1) & CKS0 (bit 0) of SMR (Serial Mode Register) (**n**) [determine the input clock to the SCI]
- BRR (Bit Rate Register) (**N**) [Baud rate generator] (0<N<255)

n	Clock	SMR Setting	
		CKS1 (bit 1)	CKS0 (bit 0)
0	ϕ	0	0
0	$\phi w/2\phi w$	0	1
2	$\phi/16$	1	0
3	$\phi/64$	1	1

$$N = \frac{\phi OSC}{(64 \times 2^{2n} \times B)} - 1$$

whereby,

B — Bit Rate (bit/s)

N — Baud Rate Generator BRR setting (0<N<255)

OSC — Value of OSC (Hz)

n — Baud Rate Generator Input Clock Number (n = 0, 2, 3)

The above theory proves that detecting different baud rates is possible. However, due to the input clock selection, certain baud rates may not be feasible to be generated. The error rate may be too high (recommend < 1%).

The error can be calculated as below:

$$\text{Error (\%)} = \frac{\text{B (rate obtained from n, N, OSC)} - \text{R (desired bit rate)}}{\text{R (desired bit rate)}} \times 100$$

The following is the recommended generated settings for n & N, based on the main input clock.

Frequency OSC (MHz)	n	Desired bps B	Calculated N	Selected N	New bps B	Error %
9.8304	0	1200	127.00	127	1200.00	0.00%
	0	2400	63.00	63	2400.00	0.00%
	0	4800	31.00	31	4800.00	0.00%
	0	9600	15.00	15	9600.00	0.00%
	0	19200	7.00	7	19200.00	0.00%
	0	31250	3.92	4	30720.00	-1.70%
	2	1200	7.00	7	1200.00	0.00%
10	0	1200	129.21	129	1201.92	0.16%
	0	2400	64.10	64	2403.85	0.16%
	0	4800	31.55	31	4882.81	1.73%
	0	9600	15.28	15	9765.63	1.73%
	0	19200	7.14	7	19531.25	1.73%
	0	31250	4.00	4	31250.00	0.00%
	2	1200	7.14	7	1220.70	1.73%
16	0	1200	207.33	207	1201.92	0.16%
	0	2400	103.17	103	2403.85	0.16%
	0	4800	51.08	51	4807.69	0.16%
	0	9600	25.04	25	9615.38	0.16%
	0	19200	12.02	12	19230.77	0.16%
	0	31250	7.00	7	31250.00	0.00%
	2	1200	12.02	12	1201.92	0.16%

For other frequency settings, please refer to the MCU hardware manual.

1.3 Alternative Solution

The simple alternative is to create a protocol so that the master controller will continue to send a series of characters such as 0xA, 0xB, and 0xC at the desired baud rate and wait for response. On the slave end, the MCU will switch itself to different baud rates, in order to capture the correct data stream, and send acknowledgment when the pattern of data is recognized.

2. Operation

Objective: The SLP will be able to switch to the respective baud rates (1200, 4800, 9600, & 19200 bps), based on the single character (Return- 0x0D) send from the HyperTerminal on PC.

2.1 Environment Setup

The setup is illustrated in figure 1.

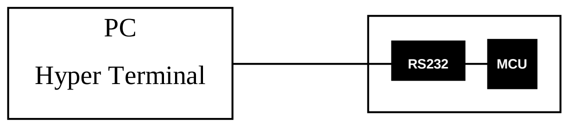
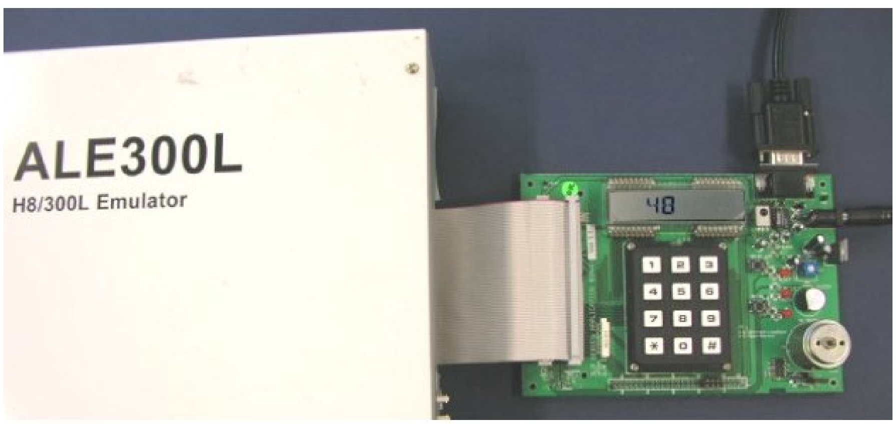


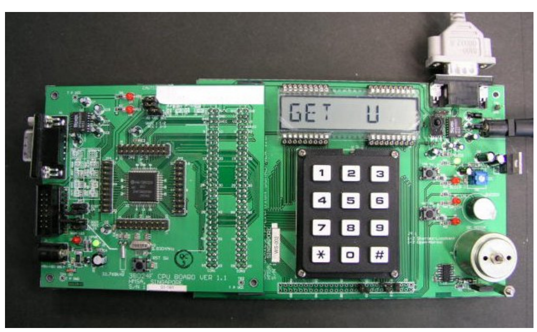
Figure 1 Basic Block Diagram

The setup of tools is shown in figure 2.



Detection of

Baud Rate - 4800 bps



Detection of

Character – ‘U’ from PC

Figure 2 Use of ALE300L Emulator or the SLP CPU Board with the General Application Board

If the general application board is not available, a simple serial driver will have to be built to condition the signal level between the SLP MCU & the PC serial port. Users can confirm the data on the LCD panel in the application board.

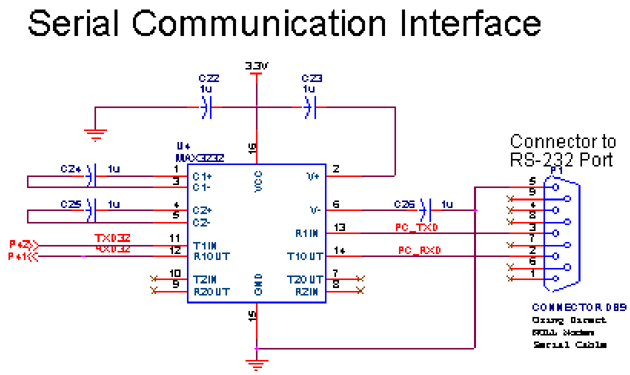


Figure 3 RS232C Serial Connection

The HyperTerminal setting on PC is shown in figure 4.

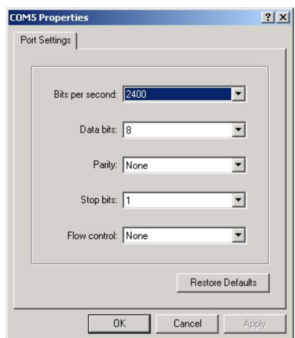


Figure 4 The HyperTerminal setting on PC

2.2 Operation & Observation

1. Set the PC to any baud rate (1200, 4800, 9600, or 19200)
2. Hit the <RETURN> key
3. The LCD display shows the baud rate detection, and a stream of data (“BAUD DETECT”) is sent back to the PC.
4. Press any key
   a. The key pressed will be displayed on the LCD display and sent back to the PC.
5. Press ‘a’ to abort
6. Go to step 1
   a. Click on [Call/ Disconnect] in the HyperTerminal.
   b. Click on [File/ Property/Configure] to re-configure to a new baud rate.

3. Code Listing

The attached code is generated using the HEW project generator targeting at the H8/38024 SLP MCU. The tool chain used is the free SLP/TINY tool chain.

The main routines of the auto baud rate detection are attached as follows.

In summary, the code provides a basic framework for users to have a quick start. The code is made to be readable, and thus it is not optimized. The Bprintf() function (customized printf()) is detailed in the application note "Writing a printf function to LCD & serial port."

```
// autobaud.c
#include "iodefine.h"
#include "auto_baud.h"
#include <machine.h>

int lcd_cursor_pos=8;

void main(void)
{
   init_io();
   init_lcd();

   while(1)
   {
      init_sci();
      auto_detect();
   }
}

unsigned char auto_detect(void)
{
   static unsigned char sequence='1';
   unsigned char b_data, b_data2;
   unsigned int baudrate,i;

   // Start up message with sequence numering
   lcd_cursor_pos=8;
   Bprintf("BAUD %c ", (BYTE)sequence,(DWORD)SPACE);
   sequence++;

   b_data              = sci_charget();
   P_SCI3.SCR3.BIT.RE  = 0x0;            //disable
   P_SCI3.SSR.BYTE     = 0x00;           //84 clear error

   if (b_data==0x00)                     //low baud rate
   {
      // start timer
      // watch for sci_charget()
      // Measure time
      // Determine new Baud
      lcd_cursor_pos=8;
      Bprintf("LOW BR", (BYTE)baudrate,(DWORD)b_data);
      P_SCI3.SCR3.BIT.RE = 0x1;          //enable
      P_SCI3.SSR.BYTE = 0x00;            //84 clear error

      //clear leftover data
      for(i=0;i<10000;i++);              // delay
      if ((P_SCI3.SSR.BIT.RDRF) == 0)
         b_data2=P_SCI3.RDR;
      return(0);
   }

   //to prevent latching in of the leftover data
   //dummy read to clear the initial unwanted data
```

```
for(i=0;i<10000;i++);                // delay
if ((P_SCI3.SSR.BIT.RDRF) == 0)
   b_data2=P_SCI3.RDR;

if (b_data == 0x80)                  // 1200 bps
{  baudrate = BR1200;
   P_SCI3.SMR.BYTE |=BR12_CKS;
   P_SCI3.BRR     = BR12_BRR;
}

else if (b_data == 0x78)             //2400 bps
{  baudrate = BR2400;
   P_SCI3.SMR.BYTE |=BR24_CKS;
   P_SCI3.BRR     = BR24_BRR;
}

else if (b_data == 0xE6)             // 4800 bps
{  baudrate = BR4800;
   P_SCI3.SMR.BYTE |=BR48_CKS;
   P_SCI3.BRR     = BR48_BRR;
}

else if (b_data == 0x0D)             // 9600 bps
{  baudrate = BR9600;
   P_SCI3.SMR.BYTE |=BR96_CKS;
   P_SCI3.BRR     = BR96_BRR;
}

else if (b_data == 0xF2   ||
       b_data == 0xF3      ||
       b_data == 0xFA     ||
       b_data == 0xFB )              // 19200 bps
{  baudrate = BR19200;
   P_SCI3.SMR.BYTE |=BR192_CKS;
   P_SCI3.BRR     = BR192_BRR;
}
else                                 // unknown Baud
{  baudrate = 0xFF;
   lcd_cursor_pos=8;
   Bprintf("X %x ", (BYTE)b_data,(DWORD)SPACE);
   P_SCI3.SCR3.BIT.RE = 0x1; //enable
   P_SCI3.SSR.BYTE = 0x00; //84 clear error
   return(0);
}

lcd_cursor_pos=8;
Bprintf("%x      ", (BYTE)baudrate,(DWORD)SPACE);

P_SCI3.SCR3.BIT.RE = 0x1;            //enable
P_SCI3.SSR.BYTE = 0x00;              //84 clear error

sci_putstr("BAUD ");
sci_charput('D');
sci_charput('E');
```

```
    sci_charput('T');
    sci_charput('E');
    sci_charput('C');
    sci_charput('T');
    sci_charput(' ');

    while(1)    // send back receive character based on new Baudrate
    {
       b_data= sci_charget();
       lcd_cursor_pos=8;
       Bprintf("GET  %c%c ", (BYTE)b_data,(DWORD)SPACE);
       sci_charput(b_data);

       if(b_data=='a' || b_data=='A')
          break;
    }
}

void sci_charput(char OutputChar)      //Serial Port
{
    while ((P_SCI3.SSR.BIT.TDRE) == 0);
    P_SCI3.TDR = OutputChar;
    P_SCI3.SSR.BIT.TDRE = 0;
}


unsigned char sci_charget(void)        //Serial Port
{
    while ((P_SCI3.SSR.BIT.RDRF) == 0);
    return(P_SCI3.RDR);
}

void sci_putstr(char *str)
{
     while(*str !='\0')
       sci_charput(*str++);
}

void init_sci(void)
{
    P_SCI3.SCR3.BYTE  = 0x30;
    P_SCI3.SMR.BYTE   = 0x00;
    P_SCI3.BRR        = BR96_BRR;
    P_SCI3.SPCR.BYTE  = 0xE0;
    P_SCI3.SSR.BYTE   = 0x84;
}
```

```
// autobaud.h
///////////////////////DEFINE////////////////////////////
// For auot baud rate
//#define OSC_16M
#define OSC_98304M

#define BR19200   0x19
#define BR9600    0x96
#define BR4800    0x48
#define BR2400    0x24
#define BR1200    0x12

#ifdef OSC_16M
#define BR192_CKS 0
#define BR192_BRR 12
#define BR96_CKS  0
#define BR96_BRR  25
#define BR48_CKS  0
#define BR48_BRR  51
#define BR24_CKS  0
#define BR24_BRR  103
#define BR12_CKS  0
#define BR12_BRR  207
#endif

#ifdef OSC_10M
#define BR192_CKS 0
#define BR192_BRR 12
#define BR96_CKS  0
#define BR96_BRR  0
#define BR48_CKS  0
#define BR48_BRR  0
#define BR24_CKS  0
#define BR24_BRR  64
#define BR12_CKS  0
#define BR12_BRR  129
#endif

#ifdef OSC_98304M
#define BR192_CKS 0
#define BR192_BRR 7
#define BR96_CKS  0
#define BR96_BRR  15
#define BR48_CKS  0
#define BR48_BRR  31
#define BR24_CKS  0
#define BR24_BRR  63
#define BR12_CKS  0
#define BR12_BRR  127
#endif
```

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Sep.10.04	—	First edition issued

Keep safety first in your circuit designs!

1. Renesas Technology Corp. 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 Corp. 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 Corp. or a third party.
2. Renesas Technology Corp. 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 Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. 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 Corp. 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 Corp. by various means, including the Renesas Technology Corp. 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 Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. 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 Corp. or an authorized Renesas Technology Corp. 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 Corp. 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 Corp. for further details on these materials or the products contained therein.