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.

32176 Group

Combination of DMAC and Serial Interface Reception

1. Overview

The reference sample program combined DMAC and serial interface reception for 32176 group appears on this document.

2. Introduction

These application examples in this document are used in the following microcomputers and conditions.

- Microcomputer: 32176 Group (M32176FnVFP, M32176FnTFP)
- Operating frequency: 20 to 40 MHz (The sample program is compiled assuming a frequency of 40 MHz.)
- Operating Board: Starter kit for 32176 Group

3. Sample program for Combination of DMAC and serial interface reception

3.1 Outline of Sample program

In this sample program, DMA4 is started by receiving serial interface0, received data is transferred on RAM. However DMAC transfer counter can not transfer more than 256 times with 8 bit, so in case more than 256 bit data is need to be received automatically DMA is had to be restarted. (by cascade connection of DMA channel or some way.)

All processing above are operated without software load.

During program operation, counter value of DMA4 is outputted to port 11.

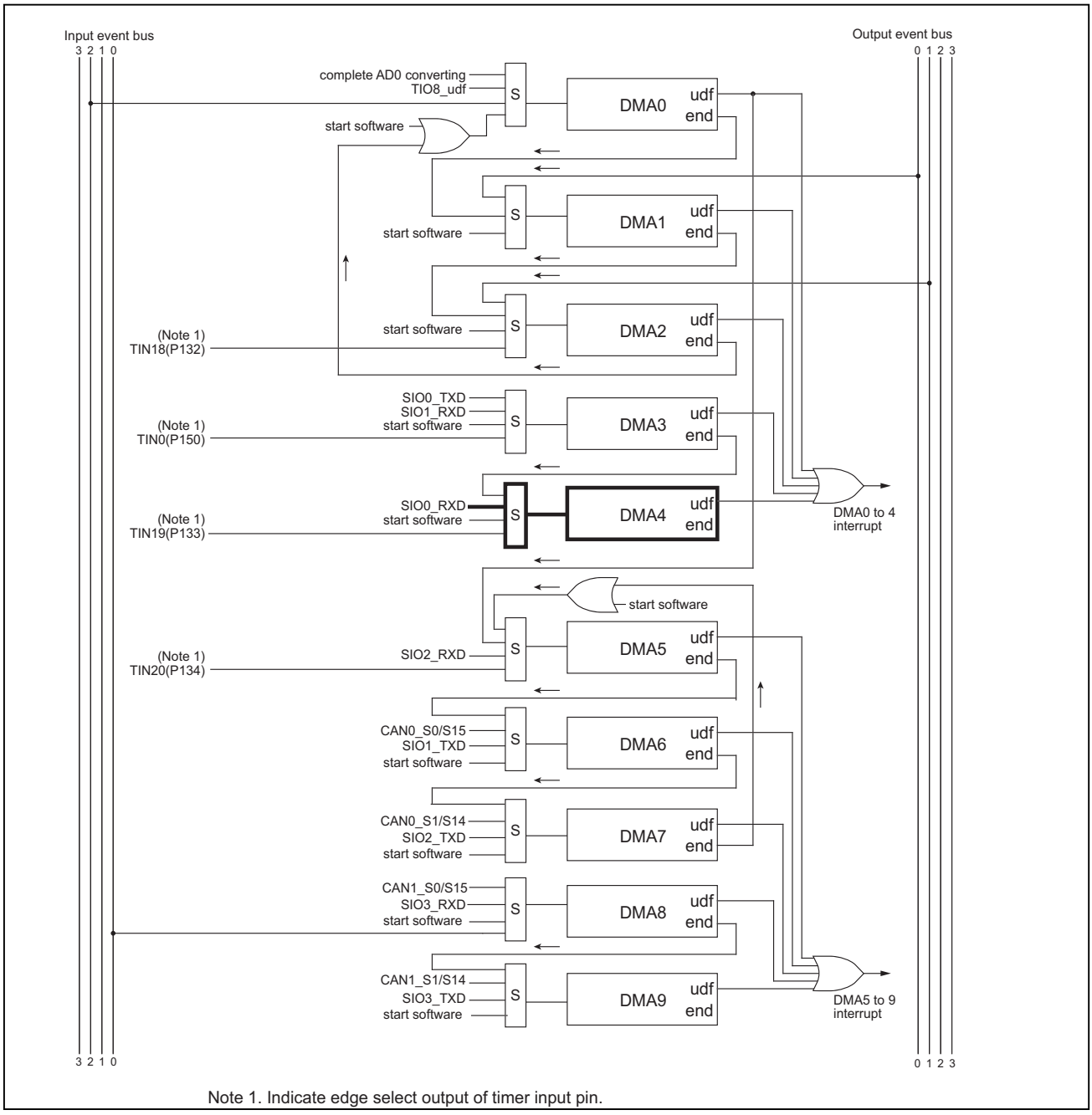

Figure3.1.1 Configuration of Sample for Combination of serial interface reception and DMAC

3.2 Description of a reference program

Note: The registers used are indicated as (register name: bit name).

3.2.1 Various initialization functions (init_func())

(1) Call the port initialization function
(2) Call the DMAC initialization function
(3) Call the serial interface initialization function

3.2.2 Port initialization functions (port_init())

(1) Initial setting of input and output port

- Set Port Input Permit bit of Port Input Special Function Control Register as permit inputting (PICNT: PIEN0)
- Initialize P11 Data register (P11DATA)
- Set P11 Direction Register as output mode. (P11DIR)
- Set P11 Operation Mode Register as input/output port. (P11MOD)

Note: If a Direction Register is set as output before setting up a Data Register, an unfixed value is outputted until writing will be performed to a Data Register.

3.2.3 DMAC initializing Function (dma_init())

(1) Interrupt Setting of DMA4

- Clear DMA4 interrupt request flag disable. (DM04ITST: DMITST4)
- Set DMA4 interrupt disable. (DM04ITMK: DMITMK4)

(2) Address Setting of DMA4

- Set SIO0 receive buffer address to DMA4 source address. (DM4SA)
- Set internal RAM address to DMA4 destination address. (DM4DA)
- Set 256 times for numbers of transfer times. (DM4TCT)

(3) Set DMA4 channel control register (DM0CNT)

- Set normal mode to DMA4 transfer mode.
- Clear DMA4 transmit request flag.
- Set SIO0 to DMA4 transfer request factor.
- Set DMA4 transfer as enable.
- Set DMA4 transfer size to 8 bit.
- Set address increment to DMA4 source address.
- Set address increment to DMA4 destination address.

3.2.4 Serial interface initializing Function (sio0rcv_init())

(1) Setting transfer control (S0RCNT, S0TCNT: CDIV, TEN)
- Set transmission paused.
- Set baud rate generator count source to f(BCLK)
- Set reception paused.

(2) Set port
- Set port P82 to TXD0, port P83 to RXD0.

(3) Set transfer data format (S0MOD, S0BAUR)
- Set to 8-bit UART.
- Set 1 stop bit to stop length.
- Set to no parity.
- Set baud rate as 19200bps. (At 19200bps when CPU clock is 40 MHz.)

(4) Interrupt setting
- Set as SIO0 transmit interrupt disable. (ISIO0RXCR: ILEVEL)
- Set as SIO0 receive complete interrupt. (SI03SEL: ISR0)
- Set as SIO0 transmit interrupt request disable. (SI03MASK: R0MASK)

(5) Start reception
- Set SIO0 to reception enable (S0RCNT: REN)

3.2.5 Main Function (main())

(1) Call the interrupt prohibitive function
(2) Call the various initialization functions
(3) Call the interrupt permit function
(4) Infinite loop outputs DMA transfer counter value to port P11.

3.3 Reference sample programs

The sample program for combination of DMAC and serial interface reception is shown below.

Note that the sample program below requires the SFR definition file. The latest SFR definition file can be downloaded from Renesas Technology website. When using the SFR definitions file, adjust the path setting to match the operating computer environment.

3.3.1 dma_sio_rcv.c

```
/*  FILE COMMENT   ************************************************
 *      M32R C Programming              Rev. 1.01
 *              < Sample Program for 32176 >
 *              < Serial I/O (UART) DMAC Receive >
 *
 *      Copyright (c) 2004 Renesas Technology Corporation
 *                      All Rights Reserved
 ************************************************************************/
/************************************************************************/
/*              Include file                                            */
/************************************************************************/

#include                "..\inc\sfr32176_pragma.h"

/************************************************************************/
/*              Definition of external reference                        */
/************************************************************************/

extern  void            DisInt( void );                 /* Interrupt disable function */
extern  void            EnInt( void );                  /* Interrupt enable function */

/************************************************************************/
/*              Function prototype declaration                          */
/************************************************************************/

        void            main(void);                     /* Main function */
        void            init_func(void);                /* Initial setup function */
        void            port_init(void);                /* Initialize port */
        void            dma_init(void);                 /* Initialize DMA */
        void            sio0rcv_init( void);            /* Initial setup serial I/O */

/************************************************************************/
/*              Define macro                                            */
/************************************************************************/

/* Setting serial IO */

                                                /* 0123 4567                                    */
#define P8MOD_SCI0      0x30u                   /* 0011 0000B   P8 operation mode register      */
                                                /*  |||| |||+--- P87                            */
                                                /*  |||| ||+---- P86                            */
                                                /*  |||| |+----- P85                            */
                                                /*  |||| +------ P84                            */
                                                /*  |||+-------- RXD0                          */
                                                /*  ||+--------- TXD0                          */
                                                /*  ++---------- don't care                    */

                                                /* 0123 4567                                    */
#define SnTCNT_INI      0x00                    /* 0000 0000B   SIOn transmit control register  */
                                                /*  |||| |||+--- Disable transmission          */
                                                /*  |||| +++---- don't care                    */
                                                /*  ||++-------- f(BCLK)                       */
                                                /*  ++---------- don't care                    */

                                                /* 0123 4567                                    */
#define SnMOD_INI       0x20                    /* 0010 0000B   SIOn mode register              */
                                                /*  |||| |||+--- Sleep function disabled       */
                                                /*  |||| ||+---- Parity inhibited              */
                                                /*  |||| |+----- don't care(odd)              */
                                                /*  |||| +------ 1 stop bit                    */
                                                /*  |||+-------- Internal clock                */
                                                /*  +++--------- 8-bit UART                   */

```

```
   64  /* Setting baud rate (Be sure to check actually set value when using) */
   65
   66  #define XIN             10                      /* 10MHz */
   67  #define BAUD_19200      (XIN * 2000000 / 16 / 19200 - 1)        /* 19200bps */
   68
   69  /* DMAC setting data */
   70
   71                                                  /* 0123 4567
*/
   72  #define DMA4_INIT       0x2d                    /* 0010 1101B   DMA4 channel control register       */
   73                                                  /* |||| |||+--- Destination address incremented     */
   74                                                  /* |||| ||+---- Source address fixed                */
   75                                                  /* |||| |+----- Transfer size of 8-bit              */
   76                                                  /* |||| +------ Transfer enabled                   */
   77                                                  /* ||++-------- 10: Start upon completion of serial I/O
0 reception     */
   78                                                  /* |+---------- No transfer request                */
   79                                                  /* +----------- Normal mode                        */
   80
   81  /************************************************************************/
   82  /*              Global variable                                         */
   83  /************************************************************************/
   84
   85          UCHAR   RcvBuf0[256];                                       /* Receive buffer */
   86
   87  /*""FUNC COMMENT""*************************************************************
   88   * Function name : init_func()
   89   *-----------------------------------------------------------------------
   90   * Description  : Initialize ICU
   91   *-----------------------------------------------------------------------
   92   * Argument     : -
   93   *-----------------------------------------------------------------------
   94   * Returns      : -
   95   *-----------------------------------------------------------------------
   96   * Notes        :
   97   *""FUNC COMMENT END""*********************************************************/
   98  void init_func(void)
   99  {
  100          port_init();                                                /* Initialize those related to port */
  101          dma_init();                                                 /* Initialize DMA                   */
  102          sio0rcv_init();                                             /* Initial setup serial I/O         */
  103  }
  104
  105  /*""FUNC COMMENT""*************************************************************
  106   * Function name :port_init()
  107   *-----------------------------------------------------------------------
  108   * Description  :Initialize port
  109   *-----------------------------------------------------------------------
  110   * Argument     : -
  111   *-----------------------------------------------------------------------
  112   * Returns      : -
  113   *-----------------------------------------------------------------------
  114   * Notes        :
  115   *""FUNC COMMENT END""*********************************************************/
  116  void port_init(void)
  117  {
  118          PICNT = PIEN0;                                              /* Enable port input */
  119
  120  /*** LED output port ***/
  121
  122          P11DATA = 0x00;                                             /* Output data (must be set prior to
mode) */
  123          P11DIR = 0xff;                                              /* P110-P117 : Output mode */
  124          P11MOD = 0x00;                                              /* P110-P117 : Input/output port */
  125  }
  126
  127  /*""FUNC COMMENT""*************************************************************
  128   * Function name : dma_init()
  129   *-----------------------------------------------------------------------
  130   * Description  : - Initialize DMAC
  131   *              :   DMA4: Transfer received data of the SIO0 receive
  132   *              :   buffer to the internal RAM
  133   *-----------------------------------------------------------------------
  134   * Argument     : -
  135   *-----------------------------------------------------------------------
  136   * Returns      : -
```

```
  137   *-----------------------------------------------------------------------
  138   * Notes        : Restart is required if the transfers size is 256 bytes or more
  139   *""FUNC COMMENT END""*************************************************/
  140  void dma_init(void)
  141  {
  142  /*** DMA4 initial setting ***/
  143
  144          DM04ITST = (~DMITST4) & 0xFFu;                          /* Clear DMA4 interrupt request */
  145          DM04ITMK |= DMITMK4;                                    /* Inhibit DMA4 interrupt */
  146
  147          DM4SA = (USHORT)&S0RXB_L;                               /* Source address -> SIO0 receive
buffer */
  148          DM4DA = (USHORT)&RcvBuf0;                               /* Destination address -> RAM buffer */
  149
  150          DM4TCT = 0xff;                                          /* Number of DMA4 transfer (256 times)
*/
  151          DM4CNT = DMA4_INIT;                                     /* Start upon the completion of the
serial I/O0 reception, enable transfer */
  152  }
  153
  154  /*""FUNC COMMENT""*******************************************************
  155   * Function name : sio0rcv_init()
  156   *-----------------------------------------------------------------------
  157   * Description  : Set 8-bit UART reception for SIO0
  158   *-----------------------------------------------------------------------
  159   * Argument     : -
  160   *-----------------------------------------------------------------------
  161   * Returns      : -
  162   *-----------------------------------------------------------------------
  163   * Notes        : Port input function must be enabled
  164   *              : No transmission setting has done
  165   *              : For M32R/E#1,2,3, PnMOD cannot be accessed for R/M/W
  166   *              : The function must be executed while interrupt is inhibited
  167   *""FUNC COMMENT END""*************************************************/
  168  void    sio0rcv_init( void)
  169  {
  170  /*** Setting transfer mode */
  171
  172          S0RCNT = 0x00;                                          /* Disable reception */
  173          S0TCNT = SnTCNT_INI;                                    /* f(BCLK) and disable transmission */
  174          P8MOD |= P8MOD_SCI0;                                    /* Set P8 for SCIO mode */
  175          S0MOD = SnMOD_INI;                                      /* Set data format */
  176          S0BAUR = BAUD_19200;                                    /* Set baud rate */
  177
  178  /*** interrupt related settings ***/
  179
  180          ISIO0RXCR = 7;                                          /* Set SIO0 receive interrupt priority
level */
  181          SI03SEL &= ~ISR0;                                       /* Select receive-finished interrupt */
  182          SI03MASK &= ~R0MASK;                                    /* Enable SIO0 receive interrupt
request */
  183
  184  /*** Starting reception ***/
  185
  186          S0RCNT |= REN;                                          /* Enable reception */
  187  }
  188
  189  /*""FUNC COMMENT""*******************************************************
  190   * Function name : main()
  191   *-----------------------------------------------------------------------
  192   * Description  : Serial receive data is stored in RAM by DMA transfer
  193   *              : - Start DMA4 upon reception of serial I/O0
  194   *              : - Receive data is stored in buffer by DMA
  195   *              : - Output DMA4's transfer counter
  196   *              :   (number of receive bytes) to LED (PORT11)
  197   *-----------------------------------------------------------------------
  198   * Argument     : -
  199   *-----------------------------------------------------------------------
  200   * Returns      : -
  201   *-----------------------------------------------------------------------
  202   * Notes        : Interrupt is not actually used
  203   *              : LED display is (255-number of receive bytes)
  204   *""FUNC COMMENT END""*************************************************/
  205  void main(void)
  206  {
  207  /*** Initialize microcomputer ***/
```

```
208
209         DisInt();                                           /* Disable interrupt */
210
211         init_func();
212
213         EnInt();                                            /* Enable interrupt */
214
215         while(1) {
216                 P11DATA = DM4TCT;
217         }
218  }
```

3.4 Timing of operation

Timing of operation in this reference program is shown below.

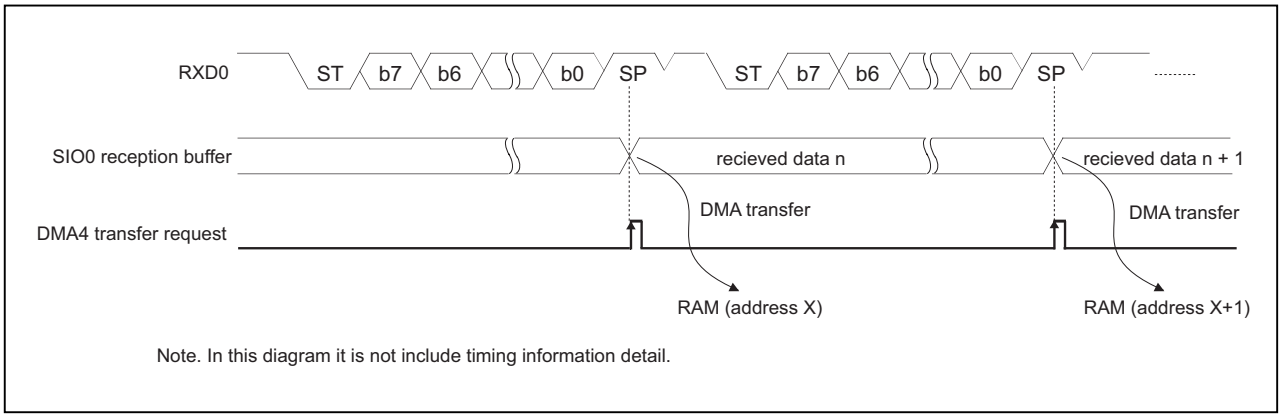

Figure3.4.1 Timing Diagram for combination sample of serial interface reception and DMAC

4. Reference Documents

- 32176 Group User's Manual (Rev.1.01)
- M32R Family Software Manual (Rev.1.20)
- M3T-CC32R V.4.30 User's Manual (Compiler)
- M3T-CC32R V.4.30 User's Manual (Assembler)

(Please get the latest one from Renesas Technology Corp. website.)

5. Website and Support Center

- Renesas Technology Corp. website

  http://www.renesas.com/

- Inquires for all Renesas products and technical inquiries for the M32R Family products:

  Customer Support Center: csc@renesas.com

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Dec.09.05	—	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.