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.

H8S/2200 Series

Measuring Voltages by A/D Conversion

Introduction

An analog input signal is converted into a digital value using the H8S/2215's A/D converter in single mode or scan mode.

Target Device

H8S/2215

Contents

1. Overview

The A/D conversion function of the H8S/2215 is used to convert an analog input signal into a digital value.

2. Configuration

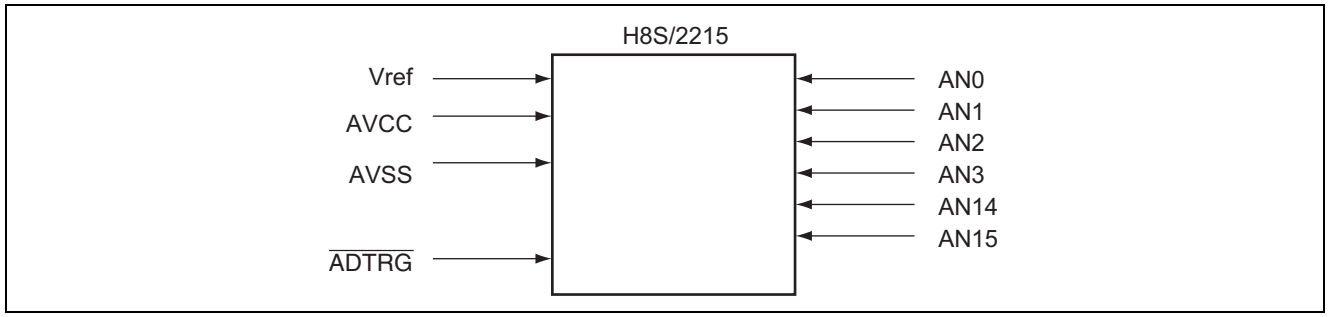


Figure 1 Pins Used for Voltage Measurement by A/D Conversion

Table 1 Pin Configuration

Pin Name	Symbol	Setting	Rating
Analog power supply pin	AVCC	3.3 V	2.7 to 3.6 V
Analog ground pin	AVSS	Ground	
Reference voltage pin	Vref	3.3 V	0 V to AVCC
Analog input pin 0	AN0	0 to Vref	0 to Vref
Analog input pin 1	AN1	0 to Vref	0 to Vref
Analog input pin 2	AN2	0 to Vref	0 to Vref
Analog input pin 3	AN3	0 to Vref	0 to Vref
Analog input pin 14	AN14	0 to Vref	0 to Vref
Analog input pin 15	AN15	0 to Vref	0 to Vref
A/D external trigger input pin	$\overline{\text{ADTRG}}$	Trigger on the falling edge	External trigger input pin for starting A/D conversion

3. Sample Programs

3.1 Functions

1. Sets the contents specified by a structure (adc_parm) in the corresponding resisters in the H8 microcomputer.
2. Starts and stops A/D conversion.
3. Checks the end of A/D conversion.
4. Reads the A/D-converted data.

3.2 Program Incorporation

1. Incorporate sample program 2-A: #define definitions.
2. Incorporate sample program 2-B: common variable definitions.
3. Incorporate sample program 2-C: prototype declarations.
4. Sample program 2-D
   A. Add ADI reset vector.
   B. Add common subroutines.
   C. Add interrupt handling routines.

3.3 Modifications to Sample Programs

Without modifications to the sample program, the system may not run. Modifications must be made according to your program and system environment.

1. The sample programs can be used without further changes if you use the I/O register structure definition file, which is available free of charge from the following Renesas web site:
   http://www.renesas.com/eng/products/mpumcu/tool/crosstool/iodef/index.html
   When you create structure definitions by yourself, modify the I/O register structures used in the sample program as appropriate.

3.4 Using the Sample Programs

Subroutines provided in the sample programs are described below. These subroutines should be used in combination according to the A/D conversion function usages. Examples of usage are shown in section 3.5, Description of Operation.

1. The contents specified by a structure (adc_parm) are set into the corresponding registers in the H8 microcomputer.

- Subroutine name: int com_adc_set_parm ()

Return Value	Description
NORMAL_END (0)	Normal termination
ADC_BUSY (-1)	Cannot set because A/D conversion is being performed.
ADC_PARM_ERR (-2)	Parameter error • In single mode, a value from 4 to 13 was attempted to be set in adc_parm.ch_no. • In scan mode, a value of 4 or greater was attempted to be set in adc_parm.ch_no.

- Before the above subroutine is called, A/D conversion mode should be specified in structure adc_parm.

Variable Name	Setting
adc_parm.ch_no	Specifies the channel(s) for A/D conversion. When adc_parm.mode = ADC_SINGLE_MODE (single mode): **ADC_CH_AN0** (0): Analog input channel AN0 **ADC_CH_AN1** (1): Analog input channel AN1 **ADC_CH_AN2** (2): Analog input channel AN2 **ADC_CH_AN3** (3): Analog input channel AN3 **ADC_CH_AN14** (14): Analog input channel AN14 **ADC_CH_AN15** (15): Analog input channel AN15 When adc_parm.mode = ADC_SCAN_MODE (scan mode): **ADC_CH_AN0** (0): Analog input channel AN0 **ADC_CH_AN0_1** (1): Analog input channels AN0, 1 **ADC_CH_AN0_2** (2): Analog input channels AN0, 1, 2 **ADC_CH_AN0_3** (3): Analog input channels AN0, 1, 2, 3
adc_parm.mode	Specifies A/D conversion mode. **ADC_SINGLE_MODE** (0): Single mode A/D conversion is performed only once for an analog input channel specified by adc_parm.ch_no. **ADC_SCAN_MODE** (1): Scan mode A/D conversion is performed repeatedly for the analog input channel(s) specified by adc_parm.ch_no. A/D conversion continues until stopped by subroutine com_adc_control (ADC_STOP). When multiple analog input channels are specified, A/D conversion is performed starting from the channel with the smallest number. Example: when adc_parm.ch_no = ADC_CH_AN0_1: A/D conversion is performed in this order: AN0, AN1, AN0, …….
adc_parm.action	Specifies the method of stating A/D conversion. **ADC_MANUAL_TRG** (0): Start by software. A/D conversion is started by subroutine com_adc_control (ADC_START). **ADC_TPU_TRG** (1): Start by TPU timer interrupt A/D conversion is started if the TTGE bit in TIER is set to 1 when the TGFA flag in TSR is set to 1 due to occurrence of TGRA input capture/compare-match in TPU0, TPU1, or TPU2. The methods of setting the TPU registers are omitted in this Application Note. See the H8S/2215 Hardware Manual. **ADC_TMR_TRG** (2): Start by TMR timer interrupt A/D conversion is only started by TMR_0 compare-match A. A/D conversion is started if the ADTE bit is set to 1 when CMFA flag in TCSR_0 is set to 1 due to occurrence of TMR_0 compare-match A. The methods of setting the TMR registers are omitted in this Application Note. See the H8S/2215 Hardware Manual. **ADC_EXT_TRG** (3): Start by external trigger (ADTRG) A/D conversion is started when the signal level on the external input pin (ADTRG) changes from high to low.
Adc_parm.end_int	Enables/disables generation of an interrupt at the end of A/D conversion. ADC_DISABLE (0): Generates an interrupt. ADC_ENABLE (1): Does not generate an interrupt.

Variable Name	Setting
adc_parm.ad_time	Sets A/D conversion time. **ADC_530_STATE** (0): 530 clock cycles **ADC_266_STATE** (1): 266 clock cycles **ADC_134_STATE** (2): 134 clock cycles **ADC_68_STATE** (3): 68 clock cycles This variable should be set so that the A/D conversion time will be 8.4 μs or longer according to the operating frequency of the microcomputer. Example: When φ = 16 MHz: A/D conversion time = (8.4 μs / 62.5 ns) cycles = 134.4 clock cycles → Setting value = ADC_266_STATE

2. Starts or stops A/D conversion.

- Subroutine name: void com_adc_control (unsigned char control kind)

Argument	Setting
control_kind	**ADC_START** (0): Start A/D conversion. **ADC_STOP** (1): Stop A/D conversion. The A/D conversion end flag (ADCSR (ADF)) is reset when AD conversion is stopped.

3. Checks if the A/D conversion has ended.

- Subroutine name: int com_adc_check (void)

Return Value	Description
ADC_NO_EXE (0)	A/D conversion is not performed or A/D conversion is in progress.
ADC_END (1)	A/D conversion has ended.

4. Reads the A/D converted data.

- Subroutine name: unsigned int com_adc_read_data (unsigned char ch_no)

Argument	Setting
ch_no	Specifies an analog input channel to be read out. **ADC_CH_AN0** (0): analog input channel AN0 **ADC_CH_AN1** (1): analog input channel AN1 **ADC_CH_AN2** (2): analog input channel AN2 **ADC_CH_AN3** (3): analog input channel AN3 **ADC_CH_AN14** (14): analog input channel AN14 **ADC_CH_AN15** (15): analog input channel AN15

Return Value	Description
16-bit data	A/D converted data of the specified channel (upper 6 bits are '0').

3.5 Description of Operation

3.5.1 Coding Examples

There may be various cases of A/D conversion processing depending on the combination of conversion mode, starting method, and use of the A/D conversion-end interrupt. Coding examples are shown below along with description of operation.

1. When A/D conversion for the analog input channel (AN0) is started by software, and an end interrupt is not used:

```
int ret ;
unsigned int ad0_data ;

// Set specifications of A/D conversion processing
adc_parm.ch_no   = ADC_CH_AN0 ;
adc_parm.mode    = ADC_SINGLE_MODE ;
adc_parm.action  = ADC_MANUAL_TRG ;
adc_parm.end_int = ADC_DISABLE ;
adc_parm.ad_time = ADC_266_STATE ;

// Place the above parameter values into the corresponding ADC registers
ret = com_adc_set_parm() ;

if (ret!=0){                                          // Setting error
// Add error processing here
}
else {
    com_adc_control(ADC_START) ;                      // Start A/D conversion
    while(com_adc_check() == ADC_NO_EXE){             // Wait until A/D conversion ends
    }
    com_adc_control(ADC_STOP) ;                       // Stop A/D conversion
    // In single mode, A/D converter automatically stops when A/D conversion ends.
    // Since the ADF bit in the ADCSR register, which indicates that A/D conversion has ended,
    // is not reset, this subroutine is called to reset ADF.

  ad0_data = com_adc_read_data(ADC_CH_AN0) ;          // Read A/D converted data
}
```

2. When A/D conversion for the analog input channel (AN0) is started by the external trigger (ADTRG), and an end interrupt is used:

```
int ret ;
// Set specifications of A/D conversion processing
adc_parm.ch_no   = ADC_CH_AN0 ;
adc_parm.mode    = ADC_SINGLE_MODE ;
adc_parm.action  = ADC_EXT_TRG ;
adc_parm.end_int = ADC_ENABLE ;
adc_parm.ad_time = ADC_266_STATE ;

// Place the above parameter values into the corresponding ADC registers
ret = com_adc_set_parm() ;

if (ret!=0){ // Setting error
// Add error processing here
}
// If the setting is correct, wait for external trigger (ADTRG) input.
// A/D conversion is started when the ADTRG pin level changes from high to low.

ADI interrupt handling routine
// When A/D conversion ends, an ADI interrupt is generated and execution
// jumps to the ADI interrupt handling routine directed by the interrupt vector.
#pragma interrupt(h8s_adi)
void h8s_adi(void)
{
unsigned int ad0_data ;

com_adc_control(ADC_STOP) ; // Stop A/D conversion
    // In single mode, A/D converter automatically stops when A/D conversion ends.
    // Since the ADF bit in the ADCSR register, which indicates that A/D conversion has ended,
    // is not reset, this subroutine is called to reset ADF.

  ad0_data = com_adc_read_data(ADC_CH_AN0) ; // Read A/D converted data
}
```

3. When A/D conversion for multiple analog input channels (AN0, 1) is started by software and repeated with use of an end interrupt:

```
int ret ;
// Set specifications of A/D conversion processing
adc_parm.ch_no = ADC_CH_AN0_1 ;
adc_parm.mode = ADC_SCAN_MODE ;
adc_parm.action = ADC_MANUAL_TRG ;
adc_parm.end_int = ADC_ENABLE ;
adc_parm.ad_time = ADC_266_STATE ;

// Place the above parameter values into the corresponding ADC registers
ret = com_adc_set_parm() ;

if (ret!=0){ // Setting error
// Add error processing here
}
else {
  com_adc_control(ADC_START) ; // Start A/D conversion
}

ADI interrupt handling routine
// When A/D conversion ends, an ADI interrupt is generated and execution jumps to
// the ADI interrupt handling routine directed by the interrupt vector.
// In scan mode, A/D conversion is repeated, and ADI interrupt handling routine is
// always called when A/D conversion ends.
// To stop A/D conversion, call subroutine com_adc_control(ADC_STOP).
#pragma interrupt(h8s_adi)
void h8s_adi(void)
{
unsigned int ad0_data , ad1_data;

  ad0_data = com_adc_read_data(ADC_CH_AN0) ; // Read A/D converted data
  ad1_data = com_adc_read_data(ADC_CH_AN1) ; // Read A/D converted data
}
```

3.5.2 Output Value of A/D-Converted Results

The output value of the A/D-converted results for an analog input voltage can be given by the following formula.

A/D-converted value = (analog input voltage / Vref) * 1024

For conversion accuracy, see sections 16.7 and 24.6 of the H8S/2215 Hardware Manual.

3.5.3 A/D Conversion Time

The A/D conversion time should be set in the structure (adc_parm.ad_time) so as to be 8.4 μs or longer according to the operating frequency of the microcomputer.

For details on the conversion time, see sections 16.5.3 and 24.6 of the H8S/2215 Hardware Manual.

3.5.4 A/D Converter Usage Notes

Since the A/D converter deals with analog signals, there are many things to be noted when designing a board.

For details, see section 16.8 of the H8S/2215 Hardware Manual.

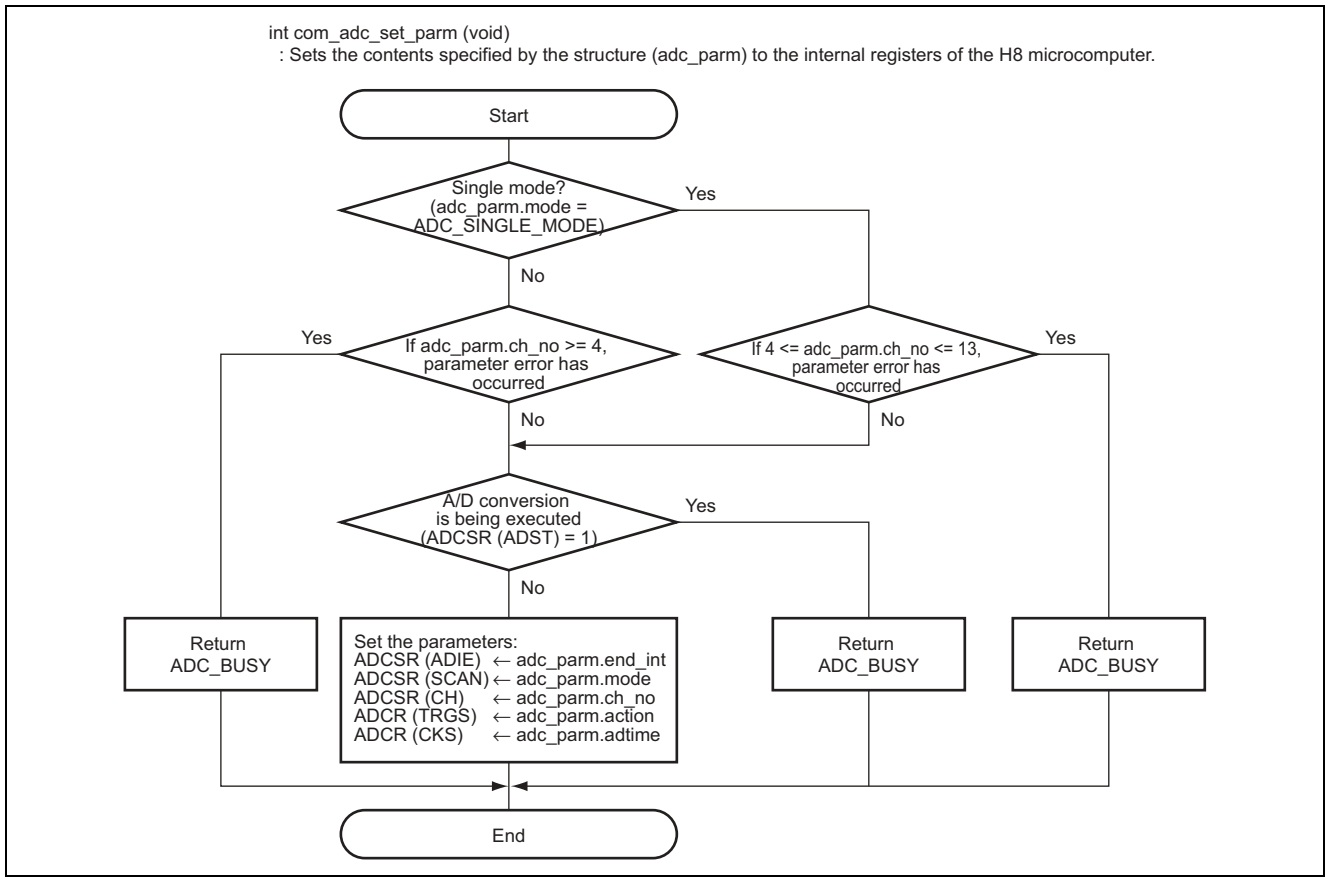
3.6 List of Registers Used

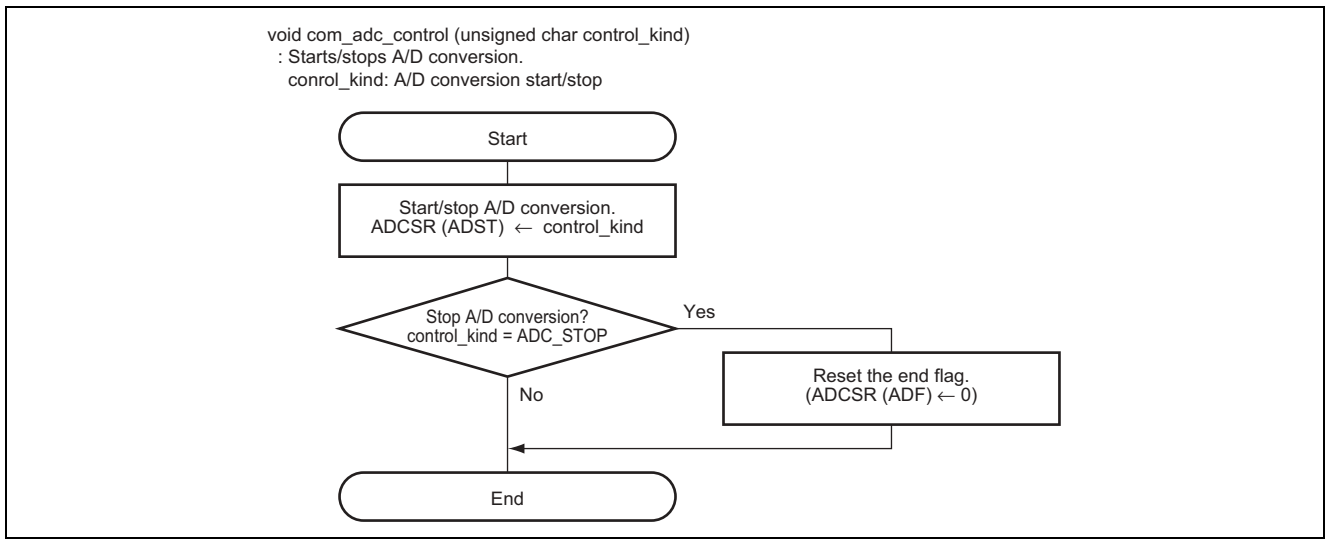
The internal registers of the H8 microcomputer used in the sample program are listed below. For detailed information, refer to the H8S/2215 Hardware Manual.

Name	Summary
A/D data registers (ADDRA to ADDRD)	• Area for storing A/D conversion results
A/D control/status register (ADCSR)	• Indicates that A/D conversion has ended. • Enables A/D conversion interrupt. • Starts A/D conversion. • Specifies A/D conversion mode. • Selects the channel(s) for which A/D conversion is performed.
A/D control register (ADCR)	• Selects A/D conversion starting method. • Sets A/D conversion time.

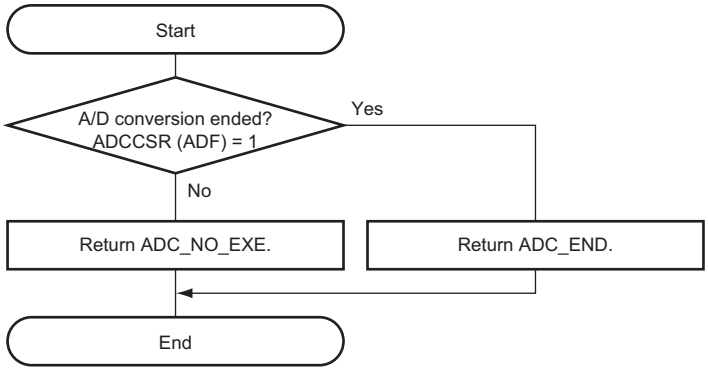
3.7 Flowcharts

Processing flows of the sample program are shown below.

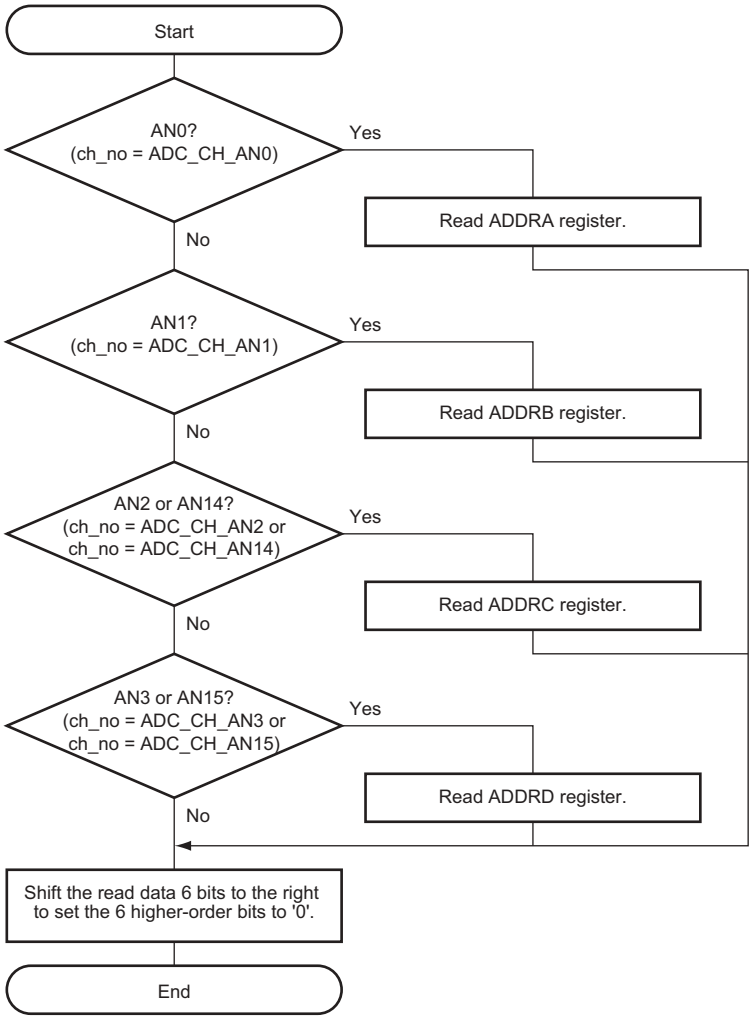




int com_adc_check (void)
: Checks the end of A/D conversion.



unsigned int com_adc_read_data (unsigned char ch_no)
: Reads out A/D converted data.
Ch_no: Specifies an analog input channel to be read.



4. Reference Document

- H8S/2215 Series Hardware Manual (published by Renesas Technology Corp.)

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Mar.16, 2004	—	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.