

SH7254R グループ

R01AN1166JJ0100

Rev.1.00

2012.05.08

割り込み変換による A/D 変換

要旨

本アプリケーションノートでは SH72546R の割り込み変換による A/D 変換動作例を示します。

動作確認デバイス

- ・ SH72546R

適用条件

- ・ マイコン : SH72546R
- ・ 動作周波数 : 内部クロック 200MHz
: 周辺クロック 40MHz
- ・ 動作モード : シングルチップモード
- ・ 統合開発環境 : ルネサス エレクトロニクス製
High-performance Embedded Workshop Ver.4.09.00
- ・ C コンパイラ : ルネサス エレクトロニクス製 SuperH RISC engine ファミリ
C/C++ コンパイラパッケージ Ver.9.04.00 Release 00
- ・ コンパイルオプション High-performance Embedded Workshop でのデフォルト設定
-cpu=sh2afpu -object="\$(CONFIGDIR)¥\$(FILELEAF).obj"
-debug -gbr=auto -chgincpath -errorpath -global_volatile=0
-opt_range=all -infinite_loop=0 -del_vacant_loop=0 -struct_alloc=1
-nologo

目次

1. 仕様	2
2. 使用機能説明	2
3. 動作説明	4
4. ソフトウェア説明	5
5. プログラムリスト	8

1. 仕様

本タスク例では図 1 に示すように、AN0～AN5,AN7～AN27 に印加される電圧を SH72546R の A/D 変換器 (AD_A) を使用し連続スキャンモードで測定します。スキャン変換はソフトウェアトリガにて開始し、測定結果はスキャン変換終了割り込み(ADIO)処理にて RAM に格納します。

また、AN6 に印加される電圧は割り込み変換を使用し測定します。割り込み変換はタイマトリガにて 5ms 毎に開始し、測定結果は割り込み変換終了割り込み(ADID6)処理にて RAM に格納します。

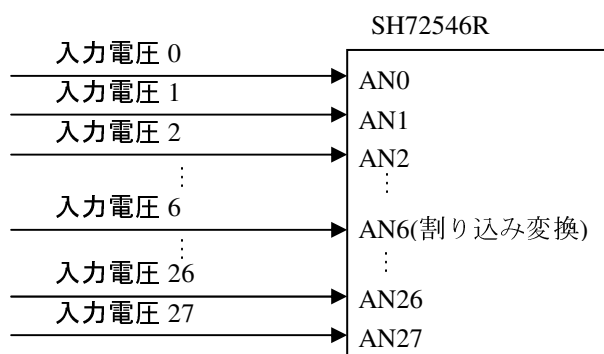


図 1 SH72546R による電圧の測定ブロック図

2. 使用機能説明

表 1、表 2、表 3 に本タスク例で用いる各周辺機能の使用リソースを示します。

表 1 A/D 変換器使用リソース

A/D 変換器機能		機能
端子	AN0～AN27	アナログ電圧を印加します。
レジスタ	ADR0 ┆ ADR27	AN0～AN27 を A/D 変換した結果を格納する 16 ビットの読み出し専用レジスタです。
	ADCSR0	ADST ADCS ADIE スキャン変換の開始/停止を制御します。 スキャン変換のモードを選択します。 A/D スキャン変換終了割り込みの発生を許可/禁止します。
	ADCER0	ADRFMT CKS A/D データレジスタのフォーマットを選択します。 A/D 変換時間を選択します。
	ADANS0	ANS0 ┆ ANS5 ANS7 ┆ ANS15 スキャン変換で変換するチャネルの選択をします。
	ADANS1	ANS16 ┆ ANS27
ADREF0	ADF	スキャン変換でスキャンが終了するたびに 1 がセットされます。ADF ビットに 1 がセットされたときにスキャン変換終了割り込みか DMAC への DMA 転送要求のどちらかを発生することができます。

	ADTRE0	ADTRGE6	ADTRGE6 ビットを 1 にセットすると、そのビットに対応した AN6 の割り込み変換要求を許可します。
	ADTRS0	ADTRS6	ADTRS6 ビットを 0 にセットし、さらに A/D 割り込みトリガイネーブルレジスタの ADTRGE6 ビットを 1 にしたときは、ATU-Ⅲのタイマトリガが入るとエッジ検出して AN6 の割り込み変換を開始します。ADTRS6 ビットを 1 にセットしたときは、A/D 割り込みソフトトリガレジスタの ADSTRG6 ビットに 1 をライトするとエッジを検出して AN6 の割り込み変換を開始します。
	ADTRF0	ADTF6	割り込み変換が終了したことを示すステータスフラグです。割り込み変換終了時に 1 がセットされます。
	ADTRD0	ADIDE6	AN6 の割り込み変換終了割り込みの発生を許可/禁止します。

表 2 ATU-Ⅲ使用リソース

ATU-Ⅲ機能		機能	
レジスタ	ATUENR	TDE PSCE	タイマ D のカウントを動作させるか、停止させるかを設定します。プリスケアラのカウンタを動作させるか、停止させるかを設定します。
	PSCR0	PSC0	プリスケアラの分周比を設定します。
	TSTRD	STRD0	タイマカウンタ 1D0,2D0,タイマダウンカウンタを動作させるか、停止させるかを設定します。
	TCRD0	CKSEL1D0	TCNT1D0 のカウントアップクロックを選択します。
	TIOR1D0	IOAD03	アウトプットコンペアレジスタの機能を選択します。
	TSRD0	CMFAD03	サブブロック D0,チャンネル3のアウトプットコンペアレジスタのコンペアマッチ発生を示すステータスフラグです。
	TIERD	CMEAD03	アウトプットコンペアレジスタのコンペアマッチ発生による割り込み要求の許可/禁止を設定します。
	CMPOD0	CMPAD03	コンペアマッチ A 検出パルスを出力するか否かを設定します。
	TCNT1D0	-	選択されたカウントクロックによりカウント動作を行います。
	OCRD03	-	TCNT1D0 の値との一致を検出し、次の Pφクロックに同期して TSRD0 レジスタの CMFAD03 ビットを 1 にセットします。

表 3 INTC 使用リソース

INTC 機能		機能	
レジスタ	IPR10	-	CMID00 ~ CMID03, OVI1D0 ~ OVI2D0, UDID00 ~ UDID03, CMID10 ~ CMID13 の割り込み優先レベルを設定します。
	IPR22	-	ADIO, ADI1, ADID0 ~ ADID7 の割り込み優先レベルを設定します。

3. 動作説明

図 2 に本タスク例の動作をハードウェア処理及びソフトウェアの処理に分けて説明します。

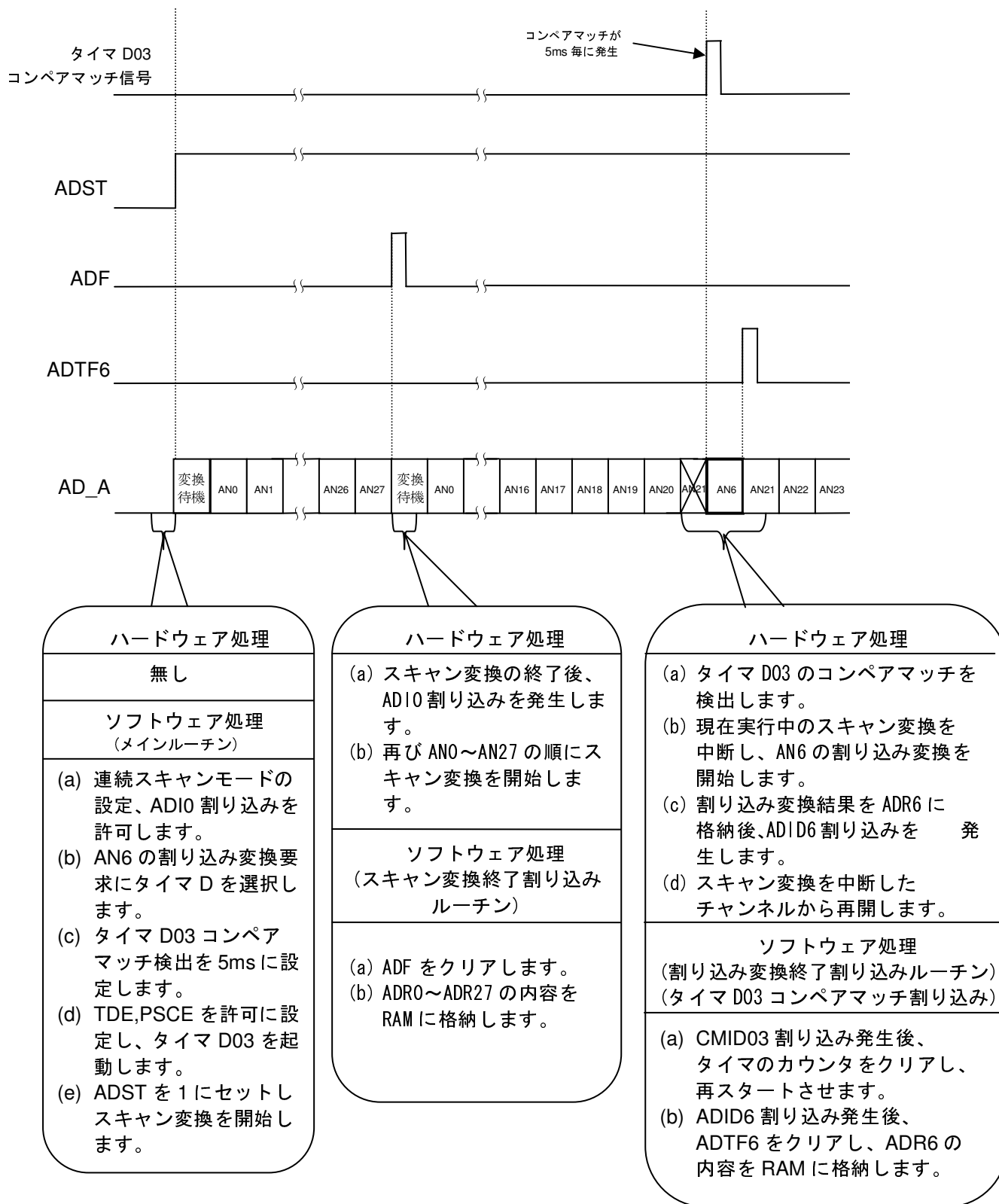


図 2 A/D 変換動作説明

4. ソフトウェア説明

4.1 モジュール説明

表 4 に本タスク例のモジュール説明を示します。

表 4 モジュール説明

モジュール名	ラベル名	機能
メインルーチン	main	ADC,ATU-Ⅲ,INTC の初期設定を行います。
スキャン変換終了 割り込みルーチン	Int_Adi0	ADI0 により起動し、スキャン変換結果を RAM に格納します。
割り込み変換 終了割り込みルーチン	Int_Adid6	ADID6 により起動し、割り込み変換結果を RAM に格納します。
タイマ D03 コンペア マッチ割り込み	Int_Cmid3	CMID03 により起動し、タイマカウンタをクリアします。

4.2 使用変数説明

表 5 に本タスク例で使用する変数の説明を示します。

表 5 使用変数の説明

ラベル名	機能	データ長	使用モジュール
DATA0~5 DATA7~27	AN0~AN5,AN7~AD27 に印加した電圧のスキャン変換データを格納します。	unsigned short	スキャン変換終了 割り込みルーチン
DATA6	AN6 に印加した電圧の割り込み変換データを格納します。	unsigned short	割り込み変換終了 割り込みルーチン

4.3 レジスタ説明

表 6 に本タスク例で使用するレジスタを示します。

表 6 使用レジスタ説明

レジスタ名	機能	設定値	使用モジュール名
ADCSR0	スキャン変換の開始/停止を制御します。 連続スキャンモードを選択します。 スキャン変換終了割り込みの発生を許可にします。	0x50 0xD0	メインルーチン
ADCER0	A/D データレジスタのフォーマットを左詰めにします。 A/D 変換時間を 50 ステートにします。	0x0000	
ADANS0	スキャン変換で変換するチャンネルに AN0~AN5,AN7~AN15 を選択します。	0xFFBF	
ADANS1	スキャン変換で変換するチャンネルに AN16~AN27 を選択をします。	0x0FFF	
ADTRE0	AN6 に対して ATU-Ⅲのタイマまたはソフトトリガからの割り込み変換要求許可にします。	0x0040	
ADTRS	AN6 の割り込み変換要求として ATU-Ⅲのタイマ D を選択します。	0x0000	
ADTRD	AN6 の割り込み変換終了後の割り込み要求を許可にします。	0x0040	
PSCR0	プリスケアラの分周比を 1/20 に設定します。	0x0013	
TCRD0	クロックバス 0 で TCNT1D0 をアップカウントする選択をします。	0x0000	
TIOR1D0	コンペアマッチで 0 出力する設定にします。	0x0040	
CMPOD0	コンペアマッチ A 検出パルス出力 ON に設定します。	0x08	
TIERD	CMFAD03 による割り込み要求を許可します。	0x0080	
IPR22	AD10 の割り込み優先レベルを 15 に設定します。 AD1D4~AD1D7 の割り込み優先レベルを 15 に設定します。	0xF00F	
IPR10	CM1D00~CM1D03 の割り込み優先レベルを 15 に設定します。	0xF000	
ATUENR	タイマ D のカウント動作を許可します。 プリスケアラのカウント動作を許可します。	0x0011	
TSTRD	TCNT1D0,TCNT2D0,DCNTD30 のカウント動作を許可します。	0x01 0x00	メインルーチン タイマ D03 コンペア マッチ割り込み
OCRD03	アウトプットコンペアの値を 10000 に設定します。	0x00271000	
ADREF0	スキャン変換終了毎に ADF がセットされ、スキャン変換終了割り込み要求が発生します。	ADF をクリア	スキャン変換終了 割り込みルーチン
ADR0 ┆ ADR5 ADR7 ┆ ADR27	AN0~AN5,AN7~AN27 を A/D 変換した結果を格納する 16 ビットの読み出し専用レジスタです。	A/D 変換値 を書き込み	
ADTRF0	割り込み変換が終了したことを示すステータスフラグです。	ADTF6 を クリア	

ADR6	AN6 を割り込み変換した結果を格納する 16 ビットの読み出し専用レジスタです。	A/D 変換値を書き込み	割り込みルーチン
TSRD0	サブブロック D0, チャンネル 3 のアウトプットコンペアレジスタのコンペアマッチ発生を示すステータスフラグです。	CMFAD3 をクリア	タイマ D03 コンペアマッチ割り込み
TCNT1D0	タイマカウンタをクリアします。	0x00000000	

5. プログラムリスト

```

/*****
* DISCLAIMER
* This software is supplied by Renesas Electronics Corporation and is only
* intended for use with Renesas products. No other uses are authorized. This
* software is owned by Renesas Electronics Corporation and is protected under
* all applicable laws, including copyright laws.
* THIS SOFTWARE IS PROVIDED "AS IS" AND RENESAS MAKES NO WARRANTIES REGARDING
* THIS SOFTWARE, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING BUT NOT
* LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
* AND NON-INFRINGEMENT. ALL SUCH WARRANTIES ARE EXPRESSLY DISCLAIMED.
* TO THE MAXIMUM EXTENT PERMITTED NOT PROHIBITED BY LAW, NEITHER RENESAS
* ELECTRONICS CORPORATION NOR ANY OF ITS AFFILIATED COMPANIES SHALL BE LIABLE
* FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR
* ANY REASON RELATED TO THIS SOFTWARE, EVEN IF RENESAS OR ITS AFFILIATES HAVE
* BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
* Renesas reserves the right, without notice, to make changes to this software
* and to discontinue the availability of this software. By using this software,
* you agree to the additional terms and conditions found by accessing the
* following link:
* http://www.renesas.com/disclaimer *
* Copyright (C) 2011 Renesas Electronics Corporation. All rights reserved.
*****/
/*****
* File Name      : SH7254R.c
* Version        : 1.00
* Device(s)      : SH72546R
* Tool-Chain     : High-performance Embedded Workshop (Ver.4.09.00.007).
*                 : C/C++ compiler package for the SuperH RISC engine family
*                 : (Ver.9.04 Release00).
* OS             : None
* H/W Platform   : SH725xEVB04-C(CPU board)
* Description     : This is the main tutorial code.
* Operation      : A/D conversion data transfer by A/D conversion interrupt.
*****/
/*****
* History : DD.MM.YYYY Version Description
*          : 23.12.2011 1.00   First Release
*****/

/*****
Includes <System Includes> , "Project Includes"
*****/
#include <machine.h>          /* Built-in function header      */
#include "iodefine.h"        /* I/O header                      */

/*****
Macro definitions
*****/
volatile struct st_adc
{
    unsigned short DATA0; /* Scan conversion data0      */
    unsigned short DATA1; /* Scan conversion data1      */
    unsigned short DATA2; /* Scan conversion data2      */
    unsigned short DATA3; /* Scan conversion data3      */
    unsigned short DATA4; /* Scan conversion data4      */
    unsigned short DATA5; /* Scan conversion data5      */
}

```



```

unsigned short DATA6; /* Conversion interrupt data 6 */
unsigned short DATA7; /* Scan conversion data7 */
unsigned short DATA8; /* Scan conversion data8 */
unsigned short DATA9; /* Scan conversion data9 */
unsigned short DATA10; /* Scan conversion data10 */
unsigned short DATA11; /* Scan conversion data11 */
unsigned short DATA12; /* Scan conversion data12 */
unsigned short DATA13; /* Scan conversion data13 */
unsigned short DATA14; /* Scan conversion data14 */
unsigned short DATA15; /* Scan conversion data15 */
unsigned short DATA16; /* Scan conversion data16 */
unsigned short DATA17; /* Scan conversion data17 */
unsigned short DATA18; /* Scan conversion data18 */
unsigned short DATA19; /* Scan conversion data19 */
unsigned short DATA20; /* Scan conversion data20 */
unsigned short DATA21; /* Scan conversion data21 */
unsigned short DATA22; /* Scan conversion data22 */
unsigned short DATA23; /* Scan conversion data23 */
unsigned short DATA24; /* Scan conversion data24 */
unsigned short DATA25; /* Scan conversion data25 */
unsigned short DATA26; /* Scan conversion data26 */
unsigned short DATA27; /* Scan conversion data27 */
};
#define ADC (*(struct st_adc *)0xFFFF8F000)

/*****
Private global variables and functions
*****/
void main(void); /* Main function */
void Int_Adi0(void); /* Scan conversion end interrupt function */
void Int_Adid6(void); /* Interrupt of conversion end function */
void Int_Cmid03(void); /* Interrupt of timer D03 compare match */

/*****
* Function Name: main
* Description : The main loop
* Argument : none
* Return Value : none
*****/
void main(void)
{
/* ADC Initialize */
/* Configure ADCSR
b7 ADST - Scan Conversion start - Scan Conversion stop
b6 ADCS - Scan Conversion mode select - Continuous scan mode
b5 Reserved - Read only bit
b4 ADIE - ADI Interrupt Enable - Enable
b3:b2 Reserved - Read only bit
b1 TRGE - External or ATU-III Trigger Enable - Enable
b0 EXTRG - Trigger Select - Select ATU-III trigger */
ADCA.ADCSR.BYTE = 0x50;

/* Configure ADCER
b15 ADRFMT - A/D Data Register Format Sselect - Selects left-alignment
b14:b12 Reserved - Read only bit
b11 DIAGM - Self Test Enable - Disable
b10 DIAGLD - Self Test Mode Select - Self-test is performed with automatic
rotation

```

```

b9:b8  DIAGVAL - Self Test Voltage Select - Reserved
b7     CKS - Clock Select - A/D conversion time = 50state
b6:b1  Reserved - Read only bit
b0     ITTRGS - Expanded Interrupt Conversion Trigger Source Selection -
        AN0 interrupt conversion is triggered by the timer D00A in the ATU-III*/
ADCA.ADCER.WORD = 0x0000;

/* Configure ADANS0
b15:b7 ANS15 to 7 - A/D Channel 15 to 7 Select - Selected
b6     ANS6 - A/D Channel 6 Select - Not selected
b5:b0  ANS5 to 0 - A/D Channel 5 to 0 Select - Selected*/
ADCA.ADANS0.WORD = 0xFFBF; /* Select AN0 to AN5 and AN7 to AN15*/

/* Configure ADANS1
b15:b12 Reserved - Read only bit
b11:b0 ANS 27 to 16 - A/D Channel 27 to 16 Select - Selected */
ADCA.ADANS1.WORD = 0x0FFF; /* Select AN16 to An27 */

/* Configure ADTRE
b15:b7 ADTRGE15-7 - AN15-7 Conversion Interrupt Request - Disable
b6     ADTRGE6 - AN6 Conversion Interrupt Request - Enable
b5:b0  ADTRGE5-0 - AN5-0 Conversion Interrupt Request - Disable */
ADCA.ADTRE.WORD = 0x0040; /* Enable : Interrupt request from AN6 */

/* Configure ADTRS
b15:b1 ADTRSn - Select Interrupt Trigger Source - Select ATU-III timer D
        for ANn interrupt request
b0     ADTRS0 - Select Interrupt Trigger Source - Select ATU-III timer D00A
        or timer G1 */
ADCA.ADTRS.WORD = 0x0000; /* Select TimerD to Conversion interrupt request*/

/* Configure ADTRD
b15:b7 ADIDE15-7 - Interrupt Conversion End Interrupt Enable - Disable ADID 15-7
interrupt request
b6     ADIDE6 - Interrupt Conversion End Interrupt Enable - Enable ADID 6 interrupt
request
b5:b1  ADIDE5-1 - Interrupt Conversion End Interrupt Enable - Disable ADID 5-1
interrupt request
b0     ADIDE0 - Interrupt Conversion End Interrupt Enable - Disable ADID0 interrupt
request or DMA transfer request */
ADCA.ADTRD.WORD = 0x0040;

/* ATU-III Initialize */
/* Configure PSCR0
b15:b10 Reserved - Read only bit
b9:b0  PSCn - Devision Ratio - Set to 1/20 of the prescaler division ratio*/
ATUCTRL.PSCR0.WORD = 0x0013;

/* Configure of TCRD
b15,11,7,3
        Reserved - Read only bit
b14    OBRED - Timer Offset Base Register enable - Enable
b13    C2CED - Counter 2 Clear Enable - Disable
b12    C1CED - Counter 1 Clear Enable - Disable
b10:b8 TCNT2D0 - TCNT2D0 Clock Select - Clock bus 0 up count
b6:b4  TCNT1D0 - TCNT1D0 Clock Select - Clock bus 0 up count
b2:b0  DCSELD0 - DCNTD0m Clock Select - Clock bus 0 down count */
ATUD.SUBBLOCKA[0].TCRD.WORD = 0x0000;

```

```
/* Configure TIOR1D
b15:b14 OSSD03 - Compare Match Output Source Select - Output level is not changed
b13:b12 OSSD02 - Compare Match Output Source Select - Output level is not changed
b11:b10 OSSD01 - Compare Match Output Source Select - Output level is not changed
b9:b8  OSSD00 - Compare Match Output Source Select - Output level is not changed
b7:b6  IOAD03 - I/O Control - Output level on compare match is 0
b5:b4  IOAD02 - I/O Control - Compare match is not performed
b3:b2  IOAD01 - I/O Control - Compare match is not performed
b1:b0  IOAD00 - I/O Control - Compare match is not performed */
ATUD.SUBBLOCKB[0].TIOR1D.WORD = 0x0040;

/* Configure of CMPOD0
b7:b4  CMPBD07-04 - Compare Match B Pulse Output Control - Compare match B pulse
is not output
b3:    CMPAD03 - Compare Match A Pulse Output Control - Compare match A pulse
is output (pulse width = cycle of the Pφ clock)
b2:b0  CMPAD02-00 - Compare Match A Pulse Output Control - Compare match A pulse
is not output */
ATUD.SUBBLOCKA[0].CMPOD.BYTE = 0x08;

/* Configure of OCRD
b31:b8 Compare Match Value : 10000
b7:b0  Reserved - Read only bit */
ATUD.SUBBLOCKB[0].OCRD[3] = 0x00271000;

/* Configure of TIERD
b15:b14 Reserved - Read only bit
b13    OVE2D0 - Overflow interrupt enable 2D0 - Disable
b12    OVE1D0 - Overflow interrupt enable 1D0 - Disable
b11:b8 UEDDD03-00 - Underflow interrupt enable D03-00 Disable
b7:b4  CMEAD03-00 - Compare Match A interrupt enable D03-00 - Interrupt request
by CMEAD03 Enable
b3:b0  CMEBD03-00 - Compare Match B interrupt enable D03-00 - Disable */
ATUD.SUBBLOCKB[0].TIERD.WORD = 0x0080;

/* INTC Initialize */
/* Configure IPR22 - Interrupt Priority Configuration Register
b15:b12 IPR_ADI0 - ADI0 Interrupt Priority Level - Level 15
b11:b8  IPR_ADI1 - ADI1 Interrupt Priority Level - Level 0
b7:b4   IPR_ADID0_3 - ADID 0 to 3 Interrupt Priority Level - Level 0
b3:b0   IPR_AIDI4_7 - ADID 4 to 7 Interrupt Priority Level - Level 15 for ADID6
*/
INTC.IPR22.WORD = 0xF00F;

/* Configure IPR10 - Interrupt Priority Configuration Register
b15:b12 IPR_CMID00_03 - ATU-D0 Interrupt Priority Level - Level 15
b11:b8  IPR_OVI1D0,OVI2D0 - ATU-D0 Interrupt Priority Level - Level 0
b7:b4   IPR_UDID00_03 - ATU-D0 0 to 3 Interrupt Priority Level - Level 0
b3:b0   IPR_CMID10_13 - ATU-D1 4 to 7 Interrupt Priority Level - Level 0 */
INTC.IPR10.WORD = 0xF000;

set_imask(0x0); /* Set Interrupt Mask Level to 0 */

/* Configure of TSTRD
b7:b4  Reserved - Read only bit
b3:b0  STRDN - TCNT1Dn,TCNT2Dn,DCNTDnm Counter Start - Enable TCNT1D0 counting*/
ATUD.TSTRD.BYTE = 0x01;
```

```

/* Configure ATUENR
b15:b10 Reserved - Read only bit
b9:b5, b3:b1      TJE, T[H-E]E, T[C-A] - Timer J, H-E, C-A Enable - Stop
b4                TDE - Timer D Enable - Enable
b0                PSCE - Prescaler Enable - Prescaler enable */
ATUCTRL.ATUENR.WORD = 0x0011;

/* Configure ADCSR
b7      ADST - Scan Conversion Start - Scan conversion start
b6      ADCS - Scan Conversion Mode Select - Continuous scan mode
b5      Reserved - Read only bit
b4      ADIE - ADI Interrupt Enable - Enable
b3:b2   Reserved - Read only bit
b1      TRGE - External or ATU-III Trigger Enable - Enable
b0      EXTRG - Trigger Select - Select ATU-III trigger      */
ADCA.ADCSR.BYTE = 0xD0;      /* Start scan conversion      */

while(1)
{
    /* Interrupt pending      */
}
} /* End of function main() */

/*****
* Function Name: Int_Adi0
* Description : The end of scan conversion
* Argument : none
* Return Value : none
*****/
void Int_Adi0(void)
{
    /* Configure ADRF
b7      ADSCACT - Scan Conversion Status
b6      ADITACT - Interrupt Conversion Status
b5:b1   Reserved - Read only bit
b0      ADF - Single Scan End Flag - Clear Single cycle scan end flag      */
ADCA.ADRF.BIT.ADF = 0;

    /* Store scan conversion result      */
    *(short*)&ADC.DATA0 = ADCA.ADR0;
    *(short*)&ADC.DATA1 = ADCA.ADR1;
    *(short*)&ADC.DATA2 = ADCA.ADR2;
    *(short*)&ADC.DATA3 = ADCA.ADR3;
    *(short*)&ADC.DATA4 = ADCA.ADR4;
    *(short*)&ADC.DATA5 = ADCA.ADR5;
    *(short*)&ADC.DATA7 = ADCA.ADR7;
    *(short*)&ADC.DATA8 = ADCA.ADR8;
    *(short*)&ADC.DATA9 = ADCA.ADR9;
    *(short*)&ADC.DATA10 = ADCA.ADR10;
    *(short*)&ADC.DATA11 = ADCA.ADR11;
    *(short*)&ADC.DATA12 = ADCA.ADR12;
    *(short*)&ADC.DATA13 = ADCA.ADR13;
    *(short*)&ADC.DATA14 = ADCA.ADR14;
    *(short*)&ADC.DATA15 = ADCA.ADR15;
    *(short*)&ADC.DATA16 = ADCA.ADR16;
    *(short*)&ADC.DATA17 = ADCA.ADR17;
    *(short*)&ADC.DATA18 = ADCA.ADR18;
    *(short*)&ADC.DATA19 = ADCA.ADR19;

```

```

*(short*)&ADC.DATA20 = ADCA.ADR20;
*(short*)&ADC.DATA21 = ADCA.ADR21;
*(short*)&ADC.DATA22 = ADCA.ADR22;
*(short*)&ADC.DATA23 = ADCA.ADR23;
*(short*)&ADC.DATA24 = ADCA.ADR24;
*(short*)&ADC.DATA25 = ADCA.ADR25;
*(short*)&ADC.DATA26 = ADCA.ADR26;
*(short*)&ADC.DATA27 = ADCA.ADR27;

/* Initialize A/D data register */
ADCA.ADR0 = 0;
ADCA.ADR1 = 0;
ADCA.ADR2 = 0;
ADCA.ADR3 = 0;
ADCA.ADR4 = 0;
ADCA.ADR5 = 0;
ADCA.ADR7 = 0;
ADCA.ADR8 = 0;
ADCA.ADR9 = 0;
ADCA.ADR10 = 0;
ADCA.ADR11 = 0;
ADCA.ADR12 = 0;
ADCA.ADR13 = 0;
ADCA.ADR14 = 0;
ADCA.ADR15 = 0;
ADCA.ADR16 = 0;
ADCA.ADR17 = 0;
ADCA.ADR18 = 0;
ADCA.ADR19 = 0;
ADCA.ADR20 = 0;
ADCA.ADR21 = 0;
ADCA.ADR22 = 0;
ADCA.ADR23 = 0;
ADCA.ADR24 = 0;
ADCA.ADR25 = 0;
ADCA.ADR26 = 0;
ADCA.ADR27 = 0;
} /* End of function Int_Adi0() */

/*****
* Function Name: Int_Adid6
* Description : Interrupt of conversion end
* Argument : none
* Return Value : none
*****/
void Int_Adid6(void)
{
    /* Configure of ADTRF
    b15:b0 ADT15-0 - Interrupt Conversion End Flag - Clear ADT6 flag */
    ADCA.ADTRF.BIT.ADTF6 &= 0;

    *(short*)&ADC.DATA6 = ADCA.ADR6;
    /* Store interrupt conversion result to RAM */
    while(1){}; /* Interrupt pending */
} /* End of function Int_Adid6() */

/*****
* Function Name: Int_Cmid03

```

```
* Description : Timer D03 Interrupt
* Argument : none
* Return Value : none
*****/
void Int_Cmid03(void)
{
    /* Configure of TSRD
    b15:b14 Reserved - Read only bit
    b13:b12 OVF2D0,OVF1D0 - Overflow Flag
    b11:b8  UDFD0N - Underflow Flag
    b7:b4   CMFAD0N - Compare Match A Flag - Clear CMFAD03 flag
    b3:b0   CMFBD0N - Compare Match B Flag */
    ATUD.SUBBLOCKB[0].TSRD.BIT.CMFAD3 &= 0;

    /* Configure of TSTRD
    b7:b4   Reserved - Read only bit
    b3:b0   STRDN - TCNT1Dn, TCNT2Dn, DCNTDnm Counter Start - Disable TCNT1D0 counting*/
    ATUD.TSTRD.BYTE = 0x00;

    /* Configure of TCNT1D
    b31:b8 Timer Counter - Clear counter (H'00000000)
    b7:b0   Reserved - Read only bit */
    ATUD.SUBBLOCKA[0].TCNT1D = 0x00000000;

    /* Configure of OCRD
    b31:b8 Compare Match Value : 10000
    b7:b0   Reserved - Read only bit */
    ATUD.SUBBLOCKB[0].OCRD[3] = 0x00271000;

    /* Configure of TSTRD
    b7:b4   Reserved - Read only bit
    b3:b0   STRDN - TCNT1Dn, TCNT2Dn, DCNTDnm Counter Start - Enable TCNT1D0 counting*/
    ATUD.TSTRD.BYTE = 0x01;
} /* End of function Int_Cmid03() */
```

ホームページとサポート窓口

- ルネサス エレクトロニクスホームページ
<http://japan.renesas.com/>
- お問い合わせ先
<http://japan.renesas.com/inquiry>

すべての商標および登録商標は、それぞれの所有者に帰属します

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本文を参照してください。なお、本マニュアルの本文と異なる記載がある場合は、本文の記載が優先するものとします。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI周辺のノイズが印加され、LSI内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSIの内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレスのアクセス禁止

【注意】リザーブアドレスのアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレスがあります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、事前に問題ないことをご確認下さい。

同じグループのマイコンでも型名が違くと、内部メモリ、レイアウトパターンの相違などにより、特性が異なる場合があります。型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害に関し、当社は、何らの責任を負うものではありません。当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を改造、改変、複製等しないでください。かかる改造、改変、複製等により生じた損害に関し、当社は、一切その責任を負いません。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。
標準水準： コンピュータ、OA機器、通信機器、計測機器、AV機器、
家電、工作機械、パーソナル機器、産業用ロボット等
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、
防災・防犯装置、各種安全装置等
当社製品は、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（原子力制御システム、軍事機器等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じても、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 当社製品をご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は耐放射線設計については行っていません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制するRoHS指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。お客様がかかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
9. 本資料に記載されている当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。また、当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事情報に使用しないでください。当社製品または技術を輸出する場合は、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。
10. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にてご負担して頂きますのでご了承ください。
11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。

注1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社とその総株主の議決権の過半数を直接または間接に保有する会社をいいます。

注2. 本資料において使用されている「当社製品」とは、注1において定義された当社の開発、製造製品をいいます。



ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

<http://www.renesas.com>

※営業お問合せ窓口の住所・電話番号は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス販売株式会社 〒100-0004 千代田区大手町2-6-2（日本ビル）

(03)5201-5307

■技術的なお問合せおよび資料のご請求は下記へどうぞ。

総合お問合せ窓口：<http://japan.renesas.com/contact/>