Bio Sensing Software Platform

Software Library for Measuring Skin Moisture

R11AN0325EJ0101
Rev.1.01
Sep 14. 2018

Abstract

This document describes the sample program to obtain the skin moisture by measuring skin conductance.

Target Device

RX231

SAIC101 (RAA730101)

Target Board

- Renesas Starter Kit for RX231 (R0K505231S000BE) (Renesas Electronics)
  Hereafter, it is abbreviated as RX231 RSK.
- RSK Option Board TSA-OP-IC101 (TESSERA TECHNOLOGY INC.)
  *Included Renesas Electronics SAIC101 (RAA730101)
- Training Potentiostat (TM-3000) (EC FRONTIER CO.,Ltd)

Contents

1. Introduction

This application note describes how to measure a skin conductance with RX231, SAIC101 and Potentiostat.

1.1 Terminology

Table 1-1 Terminology

Term	Meaning
AFE	Analog Front End circuit, pre-processing circuit before ADC
ADC	Analog to Digital signal converter
API	Application Programming Interface
DAC	Digital to Analog signal converter
ECG	ElectroCardioGram, the process of recording the electrical activity of the heart.
EGG	ElectroGastroGram, graphic records of the electrical signals that travel through the stomach muscles and control the muscles' contractions.
SPI, RSPI	Serial Peripheral Interface, Rapid SPI, one of common interface specification of communication between CPU and peripheral devices.
LPF	Low Pass Filter
PGA	Programable Gain Amplifier

1.2 Overview

This application note answers the following topics:

- Principle of skin conductance measurement using electrode

1.3 Devices

In this application note, the system is constructed following major devices:

- MCU: RX231 series 32bit microcontroller by Renesas Electronics.

  RX231 series CPU leverages a 32bit RXv2 CPU core with DSP/FPU and low power consumption technology to realize extreme power efficiency.

- ADC: SAIC101 16bit delta-sigma A/D converter with 4ch analog multiplexer by Renesas Electronics.

  SAIC101 is a flexibly change analog front-end settings in response to environmental changes.

2. Functional Purpose

The SMM realizes below functionalities:

- The system shall measure the skin conductance value
- The skin conductance shall be measured in "Micro Siemens".
- The skin conductance shall be measured at an interval of 1 minute averaged over a sampling rate.

3. Measurement Principle

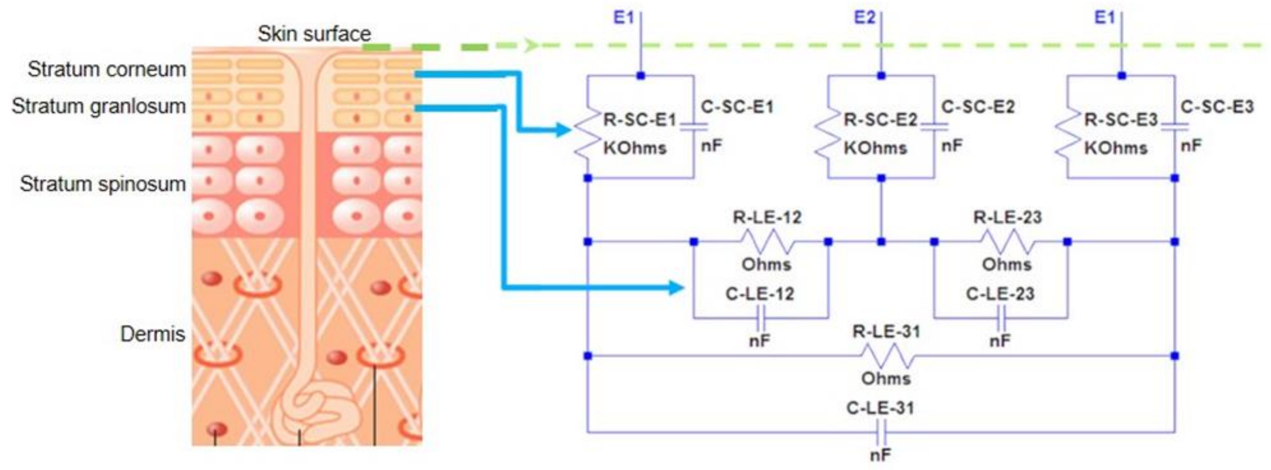
3.1 Electrical Model of Skin

3.1.1 Standard Electrical Model

The standard electrical model of the skin replaces the known physical properties of various layers of the skin by equivalent electrical components.

The stratum corneum is a hard, dense layer of dead skin cells and is modelled by a high value resistance in parallel with a low capacitance. The lower layers of the epidermis – also called the viable layers - contain spaced out live tissue cells, along with various glands. These layers are modelled by a low value resistance in parallel with a low value capacitance.

The following figure shows the electrical model for human skin in 3-D representation for 3 measuring points.



Where:
R-SC-Ex = Resistance of Stratum Corneum under Electrode x
C-SC-Ex = Capacitance of Stratum Corneum under Electrode x
R-LE-xy = Resistance of Lower Epidermis (viable skin) between Electrode x and y
C-LE-xy = Capacitance of Lower Epidermis (viable skin) between Electrode x and y

Figure 3-1 Standard Electrical Model of Human Skin

3.1.2 Simplified electrical model

For a healthy individual, the dynamic property of skin moisture, i.e. sweat, has two components – water and salts. The electrical behavior of water with dissolved salts is purely resistive. Hence, we can neglect the low capacitances and consider the skin moisture to be resistive in nature.

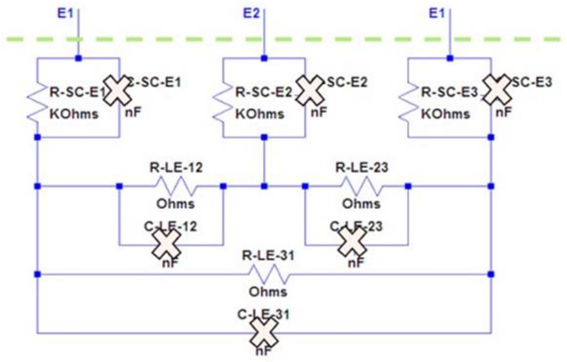
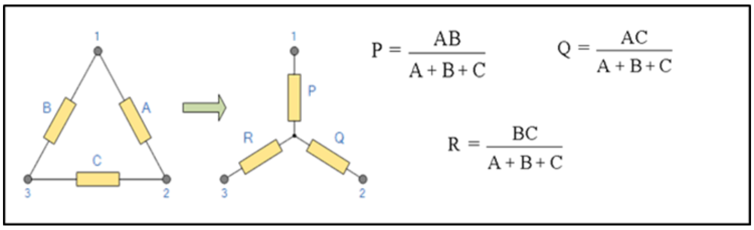


Figure 3-2 Neglecting the Capacitances

The transverse resistances of the lower epidermis are represented as a delta network between points 1, 2 and 3 in the above figure. Using delta-star conversion, we can replace the resistors by their equivalent star network values as per following reference image:



$$P = \frac{AB}{A+B+C} \qquad Q = \frac{AC}{A+B+C}$$

$$R = \frac{BC}{A+B+C}$$

Figure 3-3 Neglecting the Capacitances

Hence, the effective electrical model can be simplified as:

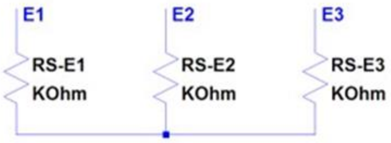


Figure 3-4 Simplified Model

Where:

RS-Ex = Effective Skin Resistance (sum of Stratum Corneum and Lower Epidermis resistances) under Electrode x

It is known that the skin resistivity is the value from 5 [Ohm-m] (at dermis) to 50,000 [Ohm-m] (at stratum corneum). For known values of electrode and layer thickness, the range of skin impedance will be 5k [Ohm] to 5000k [Ohm]. The measurement range of this application note should be 10k [Ohm] to 500k [Ohm] by considering device availability.

3.2 Detector

3.2.1 Potentiostat Technique

The Potentiostat is a method used to determine unknown electrode impedances using known impedance and an op-amp. The simplest form of its implementation is given below:

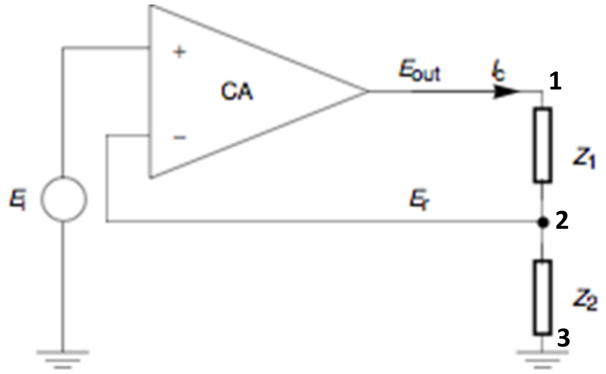


Figure 3-5 Basic Potentiostat

In this model, CA (control amplifier) is an ideal op-amp, Z1 is the known impedance and Z2 is the unknown impedance. Ei is the excitation input signal.

For an ideal op-amp, the signal at inverting terminal equals the signal at the non-inverting terminal. Hence:

Ei = Er

Also, for an ideal op-amp, no current flows into the inverting and non-inverting terminals. Hence:

Z2 = Er / Ic

Combining the two equations, we get

Z2 = Ei / Ic

Thus by measuring the value of Ic, we can calculate the value of the unknown impedance.

The points 1, 2 and 3 in Figure 3-5 indicate the electrode positions to be used in case of a physical system.

3.2.2 Driver Section Design

The driver section derived from the basic potentiostat model. The explanation for the driver section is given below:

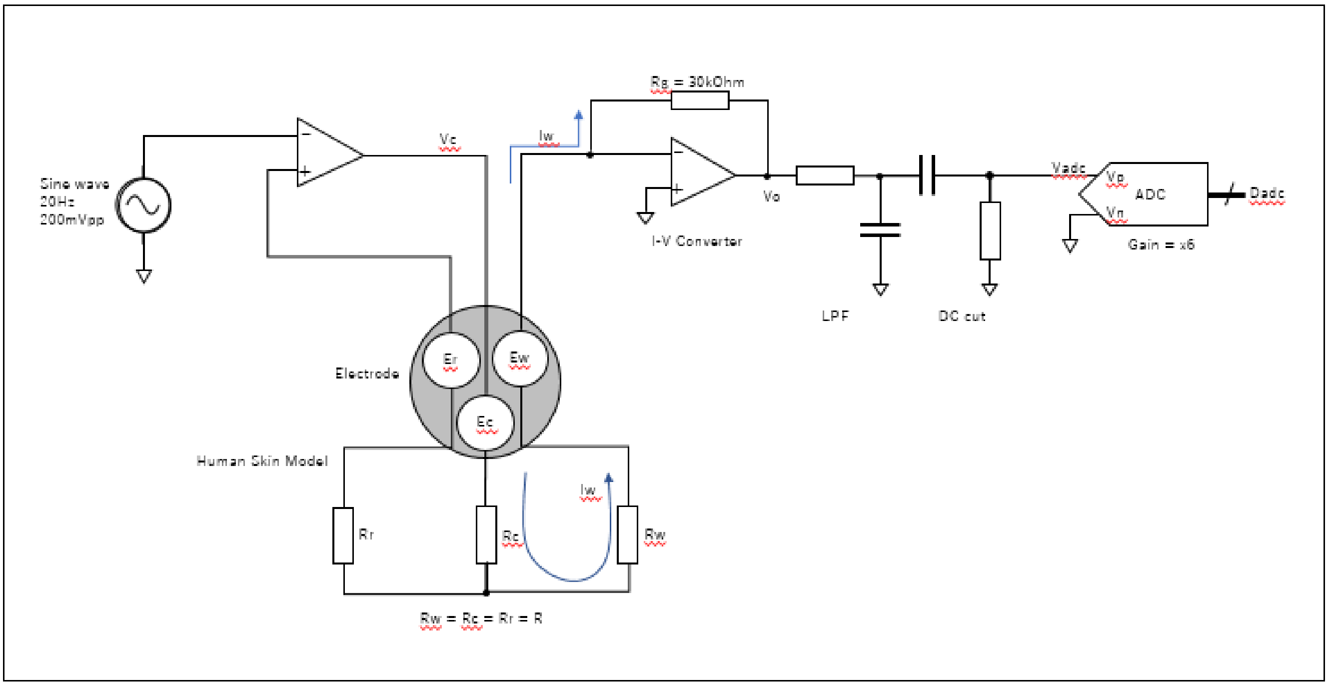


Figure 3-6 Driver Section interfaced with Skin

3.2.3 Potentiostat Explanation

(1) **Selection of V_REF**

The circuit is designed to function on a single-ended power supply. Hence a DC reference is generated and used through the skin conductance measurement circuit. The value of V_REF is chosen at 0.9V, so as to be at the center of the input range of SAIC101 which is limited at 0.2V to 1.6V.

(2) **Selection of V_IN**

The simplest way to measure resistance is by passing DC current and measure the potential drop. However, DC currents lead to issues like polarization of the electrodes and electrolyzation of the skin. IEC 60601-1, section 8.7.3 restricts the DC current to 10uA. Using AC currents has no such issues, but the measurement process is a little complex. IEC 60601-1, section 8.7.3 restricts the AC current to 100uA.

In choosing the AC frequency, we have to be careful because high frequency AC signals tend to penetrate the skin to a deeper level compared to low frequency signals of similar strength. This is due to the reduced impedance of the capacitive components of the skin impedance at higher frequencies. Since our requirement is to measure the DC component only, we restrict ourselves to low frequency AC signals.

The frequencies below 10Hz are very close to DC and there are chances of polarization and electrolysis. Frequencies above 30Hz make it difficult to filter out electrical (power-line) hum noise which is primarily in the 45-65 Hz band. The range near 20Hz is thus the automatic 'sweet spot' for our measurement.

The AC peak-to-peak amplitude of V_IN is restricted to 0.2V to limit the current flowing through the circuit. The AC signal is DC shifted by 0.9V V_REF using C5-R19 high pass filter combination. The 20KΩ current control resistors limit the short circuit current to 25uA.

Thus the peak of V_IN is at 1V.

(3) **Skin Conductance Formula**

The current Iw which path through the human body is

$$Iw = Vc / (Rc + Rw) = Vc / (2 * R)$$

It will be converted Vo by the I-V converter.

$$Vo = Iw * Rg$$

The next LPF's voltage gain = 1 / Sqrt(2)

Then Vadc which is input of ADC is

$$Vadc = Vo / Sqrt(2)$$

ADC has a x6 instrumantation amplifier and 16bit full scale range is 1.4V, So

$$Dadc = Vadc * 65536 / 1.4$$

Therefore, the total equation will be:

$$Dadc = ((Vc / (2 * R)) * Rg / Sqrt(2)) * 65536 / 1.4$$
$$Dadc = (Vc * Rg * 65536) / (2 * R * Sqrt(2) * 1.4)$$

Now, Rg = 30k Ohm, Sqrt(2) = 1.414 ADC gain is x6, and Vc = 100mV(200mVpp), So,

$$Dadc = ((100m * 30k * 65536) / (2 * R * 1.414 * 1.4)) * 6$$
$$Dadc = 297900000 * (1 / R)$$

The conductance (Cs) is a reciprocal number of R

$$Cs = 1 / R = Dadc / 297900000 = Dadc / 300000000$$

The unit of human skin conductance should be [uS], So

$$Cs = 1000000 / R$$
$$Cs = 1000000 * (Dadc / 300000000)$$
$$Cs = 0.003333 * Dadc$$

This equation should be implemented to the measurement software.

3.2.4 Receiver Section Deign

The receiver hardware is used to acquire the V_MEASURE signal using the SAIC101.

(1) 35Hz Low Pass Filter

A 35Hz first order low pass passive filter is implemented using R21 and C4 to remove the 45Hz-65Hz AC power line hum noise.

(2) Amplification

The output from the passive LPF is fed to the internal PGA of the SAIC101. The PGA is used in differential mode for better noise and gain performance with gain set at 6.

4. Libraries

4.1 API List

Each API functions' role is shown in the below table:

Table 4-1 API List

Functional Name	Function
R_SMM_StartSampling(void)	Starts the operation of MCU peripheral devices and initializes the skin conductance
R_SMM_StopSampling(void)	Stops the MCU peripheral devices' operation.
R_SMM_Calculate(void)	Reads the ADC sample values from the buffer, calculates average value of all the samples, calculates the skin conductance, and stores the values in the respective global variables.
R_SMM_InitializeWave(int16_t shifts)	Writes reference signal data (256 words) to the transfer source area of DTC0. The starting point of the waveform is passed as an argument.
R_SMM_IsSampleDataReady(void)	Returns the SampleDataReady flag. If the flag is set, the function returns true, otherwise returns false.
R_SMM_ContinueSampling(void)	Continues to collect the data samples.

4.2 Global Variables

The Skin Conductance are stored in the below global variables:

Table 4-2 List of Global Variables

Global Variables	Function
g_skin_conductance	floating point value indicating skin conductance in micro Siemens [uS]
g_samples_ready	flag which the sampling data was ready for calculating skin moisture

4.3 Memory Size

Table 4-3 Memory Size

Memory	Size
ROM	1,080 bytes for SMM library [Note1]
RAM	787 bytes for SMM Library [Note1]
User Stack	264 bytes
Interrupt Stack	48 bytes

Note1: Refer to “5.2.2 Software Architecture Overview”. It does not include device drivers.

5. Applications

5.1 Hardware

5.1.1 Hardware Design Policy

Hardware design policy of the SMM is

- The Hardware components used in the system shall be commercially available in the market. This helps the user to reproduce the development environment easily.
- TSA-OP-IC101 is used as Analog Front End which has high performance 16-bit delta-sigma ADCs with programmable gain instrumentation amplifier ideal for differential input sensors. The RAA730101 used in the TSA-OP-IC101 uses a 36-pin FBGA package, which enables a more compact set design. It has 256-byte flash memory for storing system configuration data. Also, using the serial communication (SPI or UART communication, which is selectable) each of the function blocks can be controlled from an external device and the measured data can be output to the external device. All these features provide the user flexibility and control to configure sensors for measurement.

5.1.2 Hardware Block Diagram

The SMM will be developed to run on RX231 RSK and TSA-OP-IC101.

The Block Diagram for the overall system is shown below:

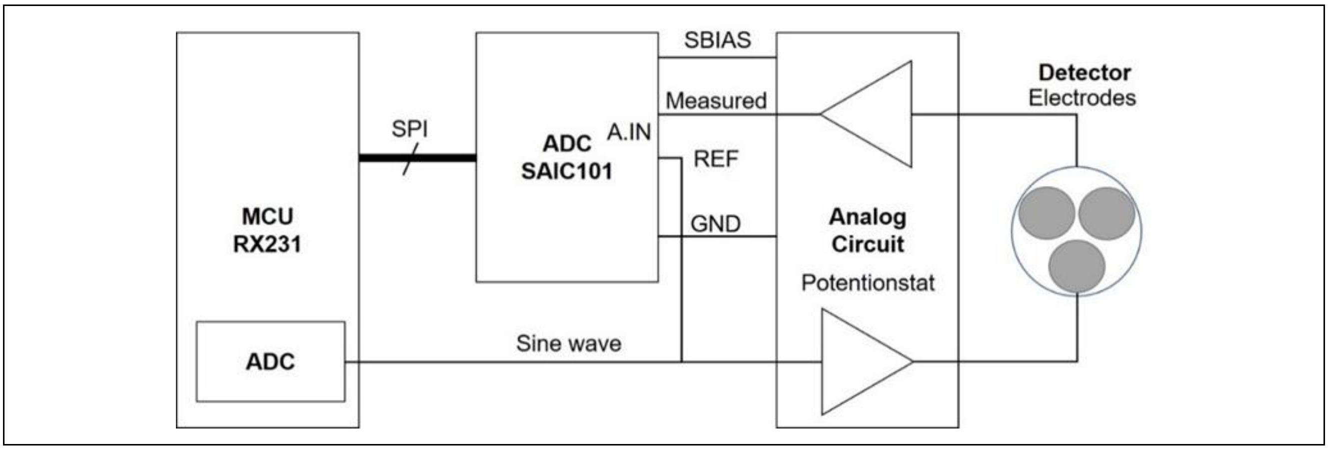


Figure 5-1 Hardware Block Diagram

The signal interconnects between RX231 RSK, TSA-OP-IC101 is shown in the figure below:

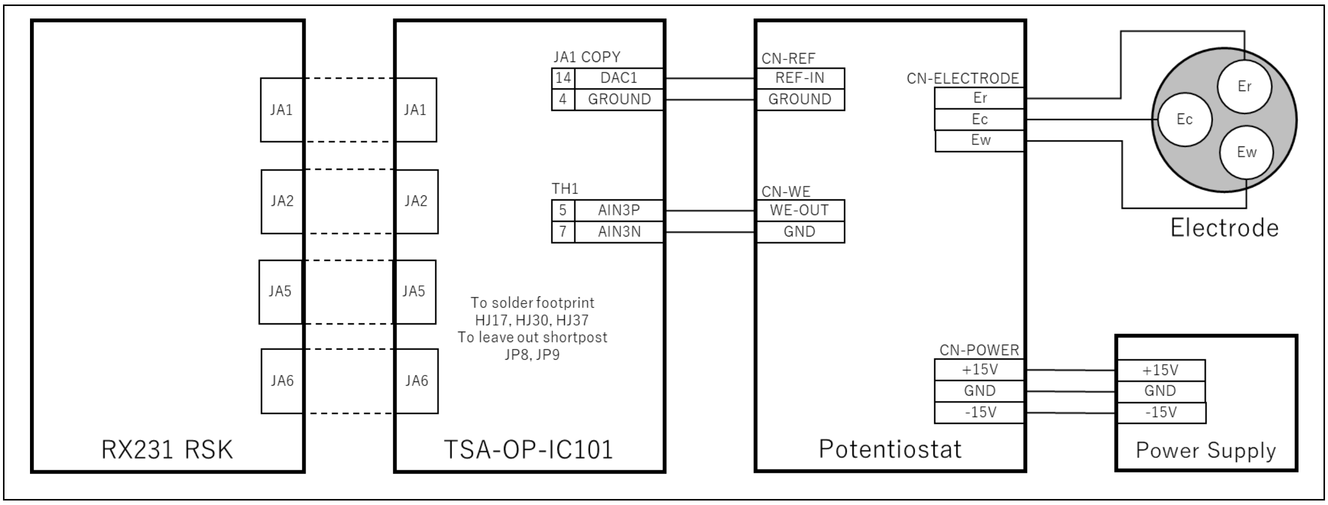


Figure 5-2 Wiring chart

The potentiostat circuit used in this application note is shown in the next figure. Training Potentiostat (TM-3000) (EC FRONTIER CO.,Ltd) is convenient as the base board.

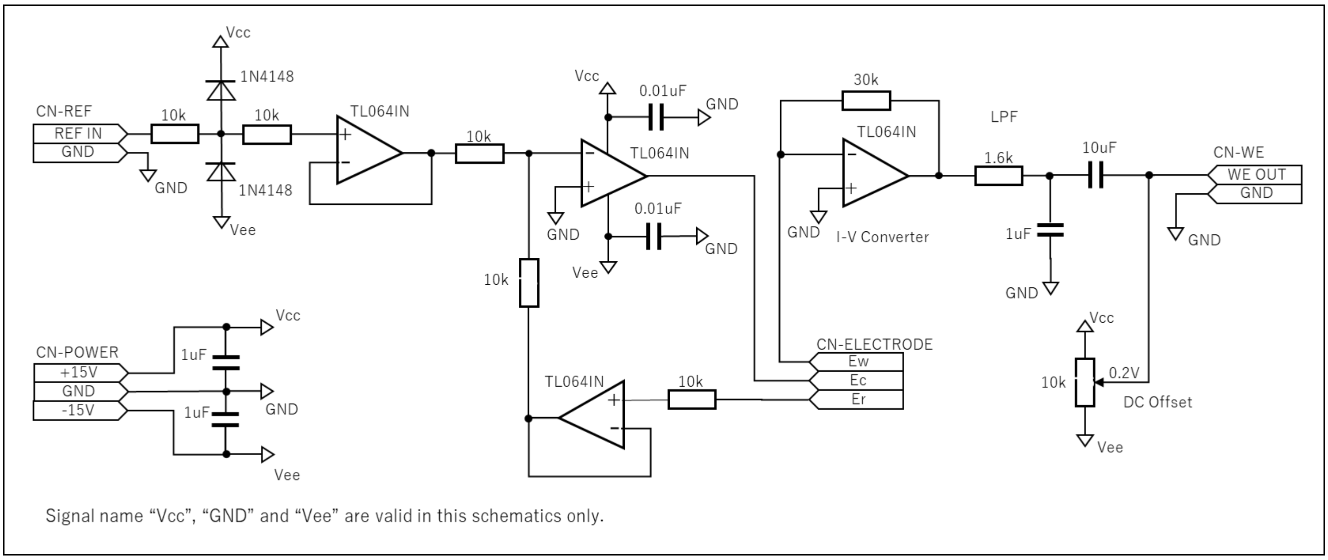


Figure 5-3 Potentiostat Circuit

5.1.3 RX231 MCU Digital Interconnect

The functional block diagram for the SMM library with the RX231 peripheral blocks is shown below:

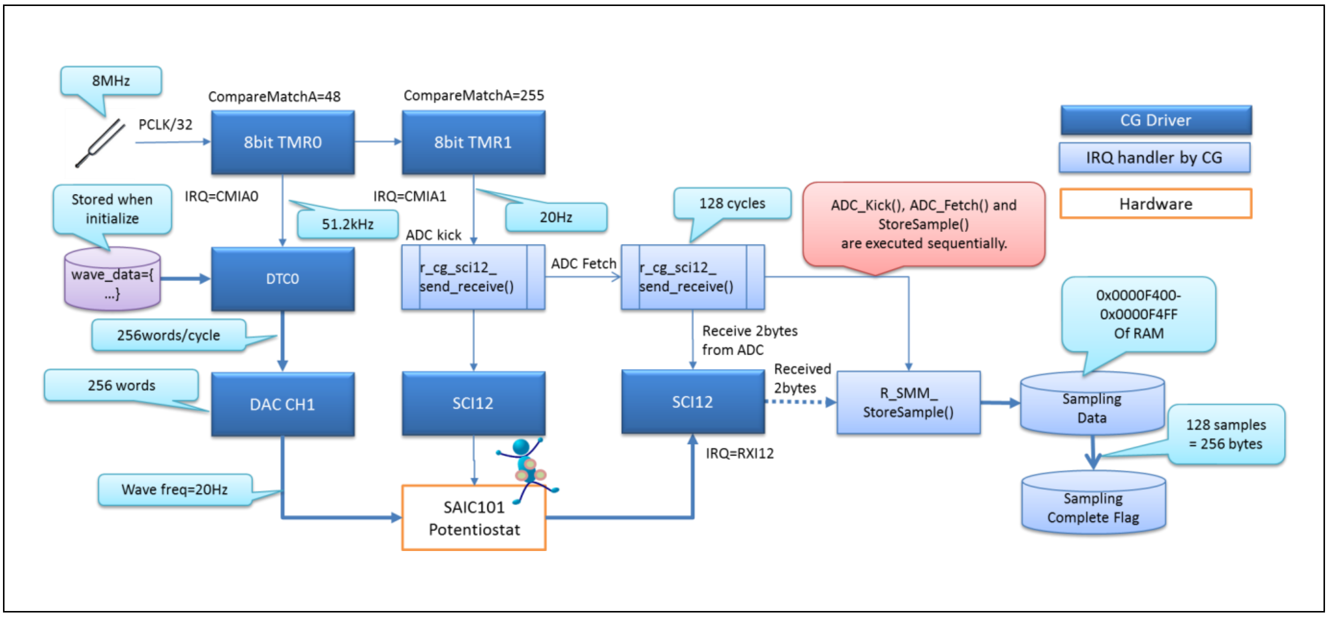


Figure 5-4 SMM Digital Interconnect

- The reference signal given to the SAIC101 potentiostat is a sinusoidal wave. The AD conversion must be executed when the reference signal is at the peak value. Since finding the peak value is easier in cosine wave than a sine wave, a cosine wave is used as reference signal. The sine wave data is phase shifted appropriately by shifting the start point of the reference wave data to get a cosine wave.
- Since the reference wave data is transferred to the DAC by the DTC, it should be configured as the transfer source area of DTC0 beforehand at the time of initialization. The waveform consists of 256 data points for one period, each data point of 16 bits length.
- The reference cosine signal sampling is controlled by a Timer (TMR0). The clock and compare match should be configured so as to sample the reference signal 256 times per cycle. For example, with input clock = 250 kHz, reference signal frequency = 20Hz, the compare match value of TMR0 should be configured as 50 counts. With this configuration, the timer TMR0 will generate an interrupt every time the timer count becomes around 50.
- For each Compare and Match Interrupt (CMAI0) generated by the timer TMR0, the DTC0 will transmit one word (2 bytes) to the DAC from the DTC0 transfer source area. Data transfer of 256 words is considered as one cycle.
- The DAC automatically converts the word to analog format every time the DTC transfer occurs and the analog value is output to the SAIC101 potentiostat.
- Every time the DTC0 completes transfer of 256 words, a DTC completion interrupt is generated and on this interrupt the SPI (SCI12) instructs the SAIC101 to perform Analog to Digital conversion (the 256th word becomes the peak of the waveform). The AD conversion must be executed when the reference signal is at the peak value. When the phase of the measurement signal deviates from the reference signal, the waveform information is shifted in advance.
- The SCI12 acquires the AD conversion value (one word, i.e., 2 bytes) from the SAIC101.
- The SCI12 acquired AD conversion value is stored to the buffer.
- This data sampling is continued for 128 cycles, on acquiring 128 samples the sampling data ready flag is set and skin conductance is calculated out of these 128 values.

5.2 Software

5.2.1 Software Design Policy

The Software Design Policy is:

- SMM Software is designed for Non-OS environment
- Code Generator Utility is used to create device drivers and hence the software framework is driven by the Code Generator Utility
- The Skin Conductance Data Sampling is automatically started on software initialization, immediately after all the peripherals are initialized and configured
- The sensor data is sampled continuously unless the system is powered off. The sampling can be stopped temporarily during the calculation, if needed to get better accuracy
- The data sampling and calculation of the Skin Conductance continuously using an infinite loop (while(1U) {}) in the main() function
- The calculate() function calculates Skin Conductance and stores the calculated values in global variables
- The measured values are not displayed on any display or stored in any file
- The software does not include power management
- The SMM Software Architecture is designed to be a simple Layered Architecture where each layer exposes a set of APIs to the layer above it. The Software Layered Architecture is described in the next section

5.2.2 Software Architecture Overview

The Software architecture is a layered architecture, where each layer provides a set of APIs for the above layers to access it.

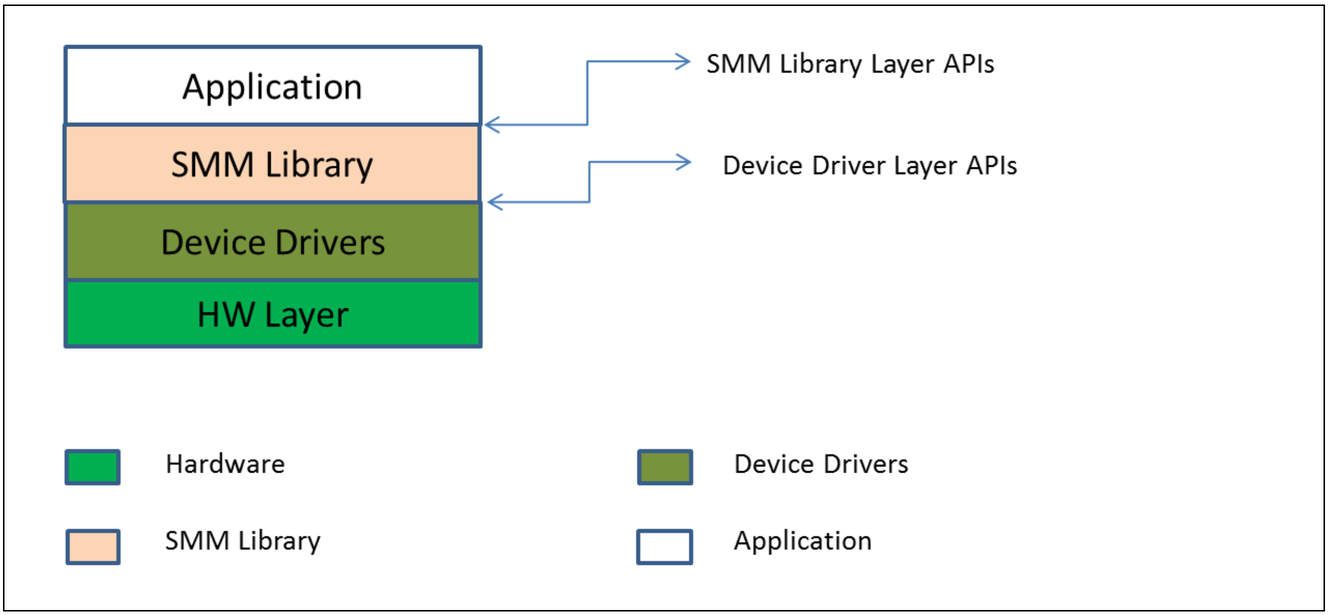


Figure 5-5 SMM Software Architecture

5.2.3 Measurement Signal Flow

The conductance of the skin is measured using surface electrodes to indicate the skin moisture level. The skin conductance measurement flow is shown in the diagram below:

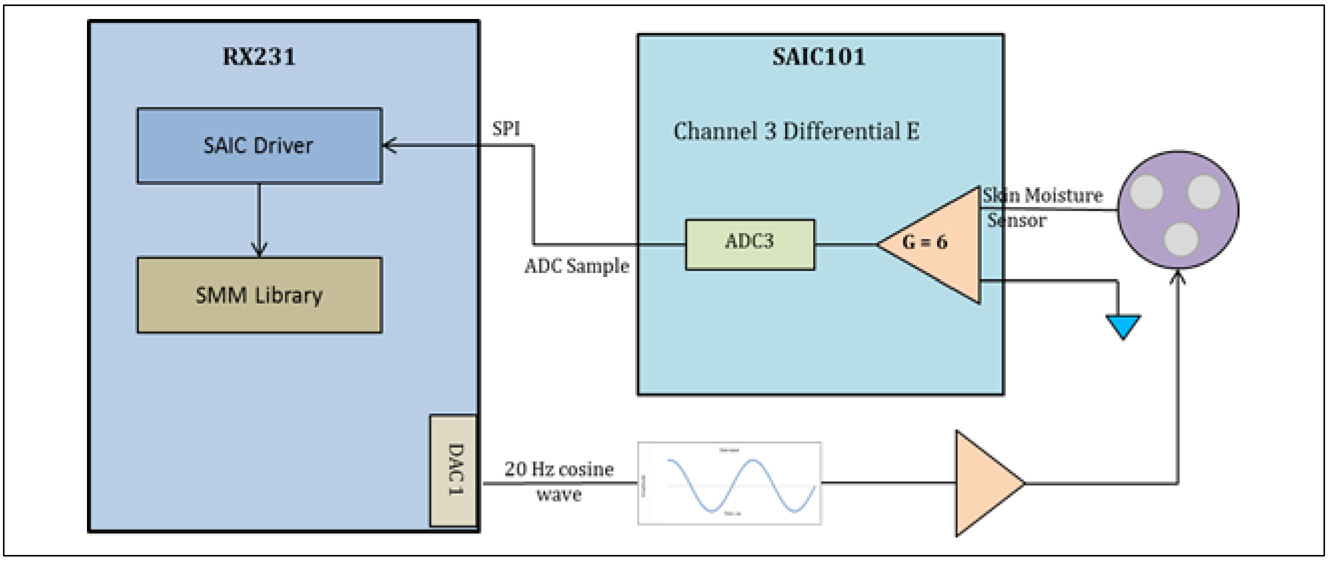


Figure 5-6 Interfacing of Skin Conductance Sensor with the MCU via SAIC101

5.2.4 Measurement Control Flow

SMM Software implements a simple application to use the SMM Library. The application is automatically started on reset. The Skin Conductance data sampling is done continuously until the device is powered off.

The Control Flow of the main application is shown in the flow chart below:

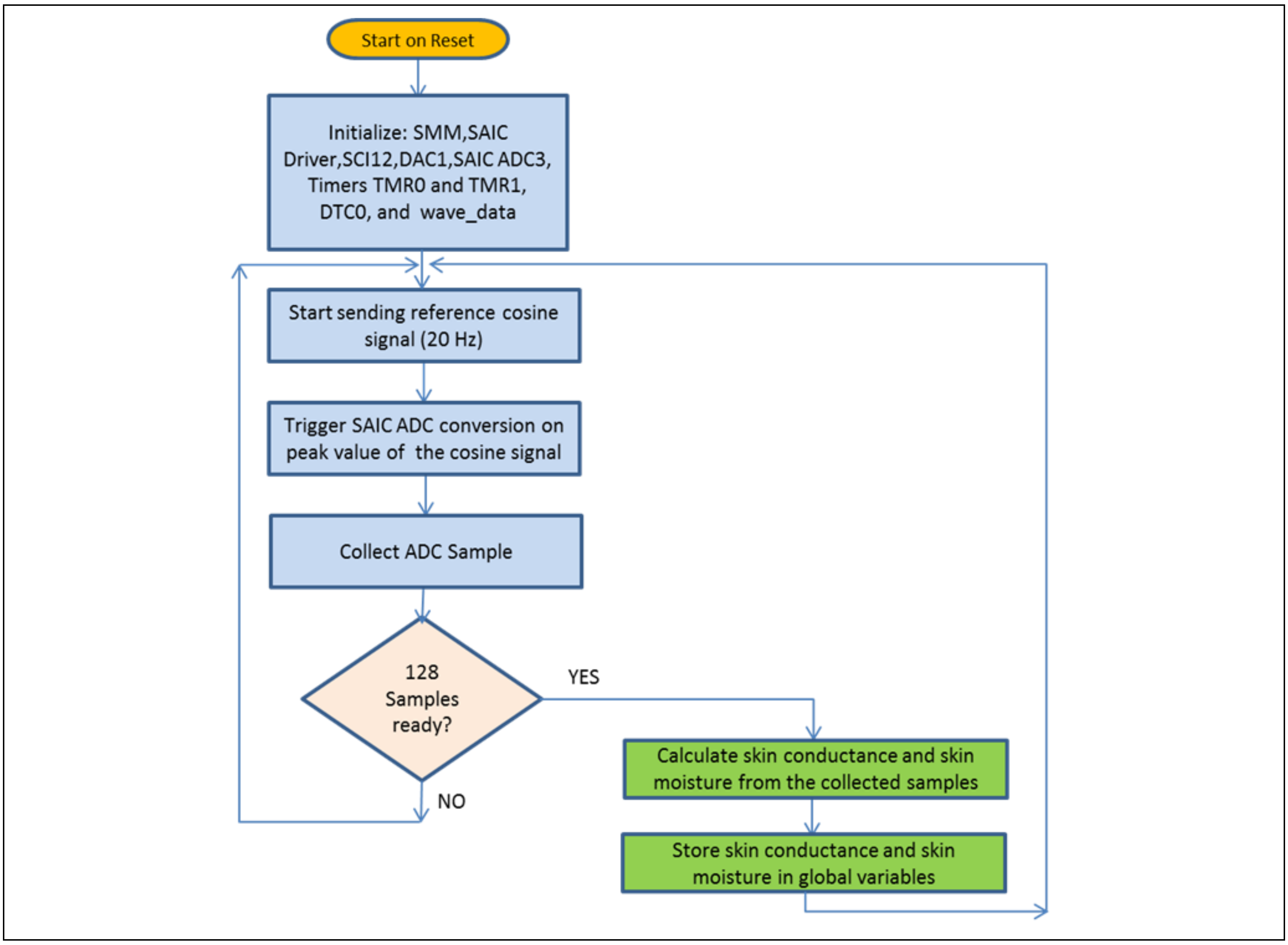


Figure 5-7 SMM Software Control Flow

5.3 Device Drivers

5.3.1 SAIC driver

(1) **Driver Function List**

Each API functions' role is shown in the below table:

Table 5-1 Driver Function List

Functional Name	Function
R_SAIC101_Initialize(void)	Initializes and sets the input mode, gain, offset, conversion rate, and interrupt setting of the SAIC101.
R_SAIC101_Done(void)	Sets the sampling data ready flag.
R_SAIC101_Kick(void)	Triggers SAIC101 A/D conversion.
R_SAIC101_Fetch(void)	Reads SAIC ADC result. The ADC result is 2 bytes (the lower and upper 8 bits of the 16-bit data) in length.
R_SAIC101_EndRTx(void)	This function disables the SAIC101.

(2) **Conditions**

Configuration of SAIC ADC for skin conductance is given below:

SAIC ADC configuration used is:

ADC ON_OFF = ON
ADC Input mode = Differential ended input mode
ADC offset = 0 mV (DC offset)
ADC Over sampling rate = 64 (15625.00 [samples per sec])
ADC Gain = 6(GSET1 = x3, GSET2 = x2, Total = x 6)
ADC count = 2 (Number of A/D conversions: 2)
ADC Sbias_voltage = 2.2 V

(3) **Algorithm**

Please refer to section 3.2.3 (3) Skin conductance formula.

5.3.2 Code Generator

Table 5-2 shows the used peripheral function.

Table 5-3 shows the functions generated by Code Generator.

Table 5-2 Used Peripheral function List

Peripheral function	Macro	Sub Macro	Setting	Status
Clock Generator	CGC		VCC setting	2.7 (V) = VCC = 5.5 (V)
			Main clock oscillation source	Resonator
			Main clock oscillation source Frequency	8(MHz)
			Oscillator wait time	8192cycles2048 (µs)
			Oscillation stop detection function	Disabled
			PLL circuit setting	
			Input frequency division ratio	x 1/2
			Frequency multiplication factor	x 8
			PLL Frequency	32 (MHz)
			Sub-clock oscillator drive capacity	Drive capacity for low CL
			Sub-clock oscillator and RTC (RTCSCLK) setting	32.768 (kHz)
			Low speed clock oscillator (LOCO) setting	4 (MHz)
			Clock source	Main clock oscillator
			System clock (ICLK)	x 18 (MHz)
			Peripheral module clock (PCLKA)	x 18 (MHz)
			Peripheral module clock (PCLKB)	x 18 (MHz)
			Peripheral module clock (PCLKD)	x 18 (MHz)
			External bus clock (BCLK)	x 18 (MHz)
			Flash IF clock (FCLK)	x 18 (MHz)
			BCLK pin output setting	Clock output source BCLK
I/O Ports	Port3	P33	Mode	Out
			CMOS output	Used (for debugging)
			output value	1
Data Transfer Controller	Dtc	BaseAddress	Transfer data read skip	Disable
			Address mode	Full-address mode (32 bits)
			DTC vector base address	0x0000FC00
		DtcChannel0	Transfer data 0	Used
			Activation source	TMR0 (CMIA0 vect=174)
			Transfer mode setting	Repeat mode
			Transfer data size setting	16 bits
			Interrupt setting	An interrupt request to the CPU is generated each time DTC data transfer is performed
			Block / Repeat area setting	Transfer source
			Source address	0x0000FA00(Address incremented)
			Destination address	0x00088042(Address fixed)
			Count	256

8-bit Timer	Tmr0	TmrChannel0	TMR0	8-bit count mode
			Clock source	PCLK/32 250 (kHz)
			Counter clear	Cleared by compare match A
			Compare match A value (TCORA)	48 count (Actual value: 48)
			Compare match B value (TCORB)	48 count (Actual value: 48)
			Enable TMO0 output	Used P22
			Output at compare match A	Toggle output
			Output at compare match B	No change
			Enable TCORA compare match interrupt (CMIA0)	Used
			Priority	Level 10
			TMR0	8-bit count mode
			Clock source	PCLK/32 250 (kHz)
			Counter clear	Cleared by compare match A
			Compare match A value (TCORA)	48 count (Actual value: 48)
			Compare match B value (TCORB)	48 count (Actual value: 48)
			Enable TMO0 output	Used P22
		TmrChannel1	TMR1	8-bit count mode
			Clock source	Count at TMR0 compare match A
			Counter clear	Disabled
			Compare match A value (TCORA)	255 count (Actual value: 255)
			Compare match B value (TCORB)	256 count (Actual value: 256)
			Enable TCORA compare match interrupt (CMIA1)	Used
			Priority	Level 12
Serial Communications Interface	SCI12		Function setting	Simple SPI bus (Master transmit/receive)
			SMOSI12	PE1
			SMISO12	PE2
			Transfer direction setting	MSB-first
			Data inversion setting	Normal
			Transfer clock	Internal clock
			Bit rate	100000 (bps)
			SCK12 pin function selection	Clock output
			SCK12	PE0
			Clock delay	Clock is not delayed
			Transmit data handling	Data handled in interrupt service routine
			Receive data handling	Data handled in interrupt service routine
			TXI12, RXI12, TEI12, ERI12 priority	Level 15 (highest)
			Enable error interrupt (ERI12)	Used
			Transmission end	Used
			Reception end	Used
			Reception error	Used
12-bit D/A Converter	DA		D/A converter operation setting	Used
			Use DA1	Used
			Data format	Right-alignment
			VREF select	VREFH/VREFL

Table 5-3 Functions Generated by Code Generator

Peripheral function	File	Macro	Function
Common	r_cg_main.c		void main(void)
			void R_MAIN_UserInit(void)
	r_cg_dbsct.c		-
	r_cg_intprg.c		void r_privileged_exception(void)
			void r_floatingpoint_exception(void)
			void r_access_exception(void)
			void r_undefined_exception(void)
			void r_reserved_exception(void)
			void r_nmi_exception(void)
			void r_brk_exception(void)
	r_cg_resetprg.c		void PowerON_Reset_PC(void)
	r_cg_sbrk.c		-
	r_cg_vecttbl.c		-
	r_cg_sbrk.h		-
	r_cg_stacksct.h		-
	r_cg_vect.h		-
	r_cg_hardware_setup.c		void R_Systeminit(void)
			void HardwareSetup(void)
	r_cg_macrodriver.h		-
	r_cg_userdefine.h		-
Clock Generator	r_cg_cgc.c		void R_CGC_Create(void)
	r_cg_cgc_user.c		-
	r_cg_cgc.h		-
I/O Ports	r_cg_port.c		void R_PORT_Create(void)
	r_cg_port_user.c		-
	r_cg_port.h		-
Data Transfer Controller	r_cg_dtc.c		void R_DTC_Create(void)
		DTC0	void R_DTC0_Start(void)
			void R_DTC0_Stop(void)
	r_cg_dtc_user.c		-
	r_cg_dtc.h		-
8 Bit Timer	r_cg_tmr.c		void R_TMR_Create(void)
		TMR0	void R_TMR0_Start(void)
			void R_TMR0_Stop(void)
		TMR1	void R_TMR1_Start(void)
			void R_TMR1_Stop(void)
	r_cg_tmr_user.c	TMR0	static void r_tmr_cmia0_interrupt(void)
		TMR1	static void r_tmr_cmia1_interrupt(void)
	r_cg_tmr.h		-

Serial Communication Interface	r_cg_sci.c	SCI12	void R_SCI12_Create(void)
			void R_SCI12_Start(void)
			void R_SCI12_Stop(void)
			MD_STATUS R_SCI12_SPI_Master_Send_Receive(uint8_t * const tx_buf, uint16_t tx_num, uint8_t * const rx_buf, uint16_t rx_num)
	r_cg_sci_user.c	SCI12	static void r_sci12_transmit_interrupt(void)
			static void r_sci12_transmitend_interrupt(void)
			static void r_sci12_receive_interrupt(void)
			static void r_sci12_receiveerror_interrupt(void)
			void r_sci12_callback_transmitend(void)
			void r_sci12_callback_receiveend(void)
			void r_sci12_callback_receiveerror(void)
	r_cg_sci.h		-
12 Bit D / A Converter	r_cg_r12da.c		void R_R12DA_Create(void)
			void R_R12DA1_Start(void)
			void R_R12DA1_Stop(void)
			void R_R12DA1_Set_ConversionValue(uint16_t reg_value)
	r_cg_r12da_user.c		-
	r_cg_r12da.h		-

For details, refer to the following files.

- Function.html

  It is stored in the “an-r11an0325ej0100-bsspf-apl/workspace/SMM/doc” folder
- Macro.html

  It is stored in the “an-r11an0325ej0100-bsspf-apl/workspace/SMM/doc” folder

5.4 Application Framework

The application framework generated by Code Generator is shown in the flow chart below:

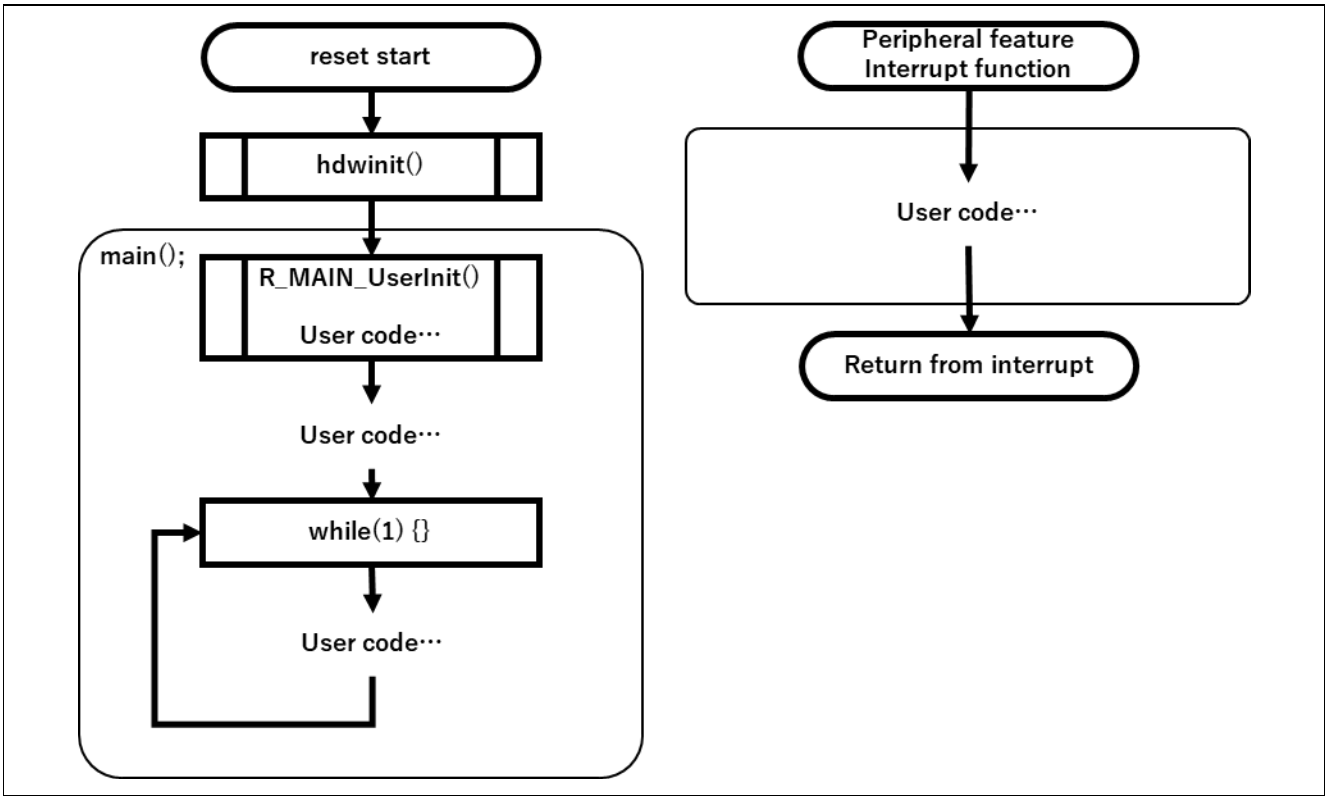


Figure 5-8 Application Framework by the Code Generator

The application program is constructed on this framework generated by the Code Generator.

In the main routine, all the peripherals of the MCU will be initialized before the main () function is executed. The hdwinit() is automatically invoked from the compiler on reset and all the peripherals are initialized according to the Code Generator Configuration.

The main () function will have three sections:

1. R_MAIN_UserInit () – This function is invoked in the beginning of main () function. User code to initialize all the devices outside the MCU, such as SAIC shall be implemented in R_MAIN_UseInit () function.
2. User Code Section - User code to start each of the device operation shall be implemented here.
3. While loop – main () function in the application framework will have an infinite while loop. User code can be implemented in this loop.

The Code Generator generates Peripheral Interrupt Framework for each of the Interrupt configured in the Code Generator. The user code to handle each of the enabled interrupts shall be implemented in the respective interrupt template generated by the code generator.

5.5 Application Flow

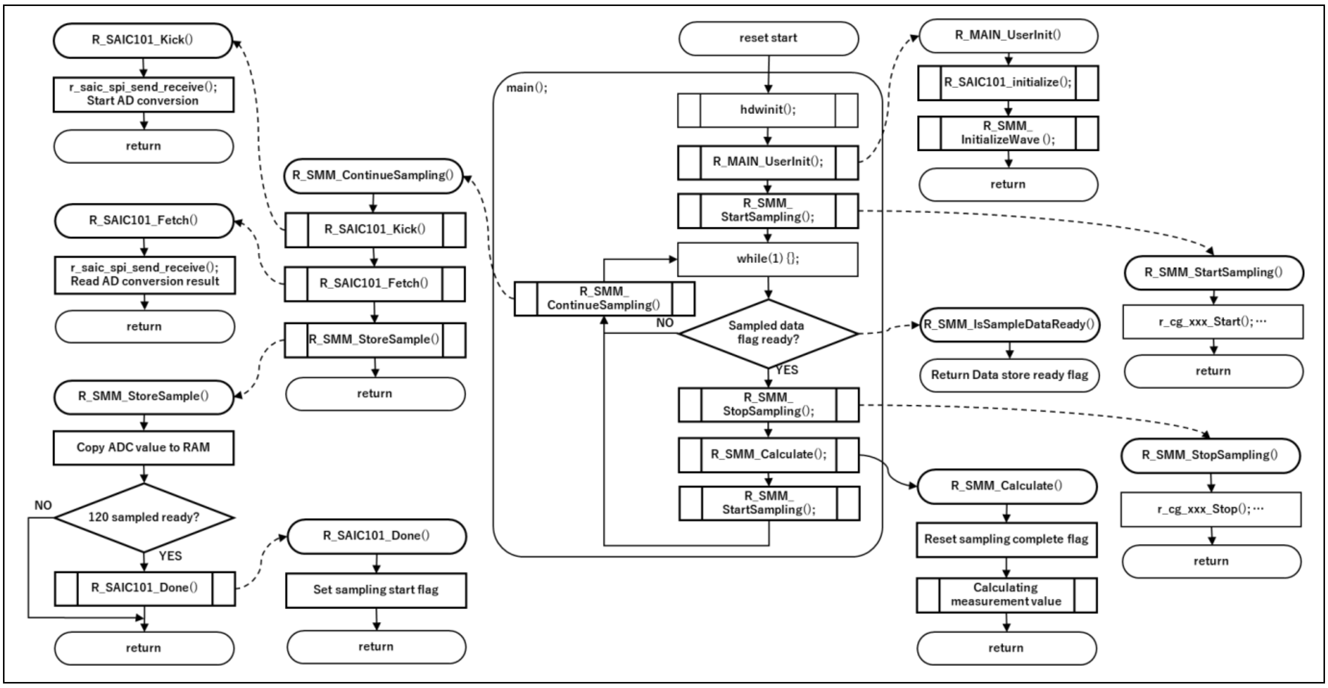


Figure 5-9 Software Flow Chart

The application does the below operations:

- Initialize the MCU, Peripherals and the SAIC on reset.
- Do below operations repeatedly

    Start Data Sampling

    Wait for required number of samples

    Stop Sampling once the required number of samples are collected

    Calculate Skin Conductance

    Start Data Sampling again

5.6 File Configurations

Figure 5-10 and Figure 5-11 show the file structure.

```
an-r11an0325ej0101-bsspf-apl
|    r11an0325ej0101-bsspf.pdf                      : This application note
|
+--- workspace                                      : Workspace folder
     |
     +--- SMM                                       : Project folder
          |    .cproject                            : ProjectDescription
          |    .project                             : ProjectDescription
          |    SM HardwareDebug.launch              : Launch Configuration
          |
          +--- .settings                            : Configuration folder of e2studio (Omit details)
          |
          +--- demo
          |         r11an0325ej0101- bsspf-smm.zip  : Archived file of this project
          |
          +--- doc
          |    |    Function.html                   : Function Table file for CG
          |    |    Macro.html                      : Macro Table file for CG
          |    |
          |    +--- Macro.files                     : Macro.files folder (Omit details)
          |
          +--- generate                             : generate folder
          |         iodefine.h                      : IO definition file
          |
          +--- src                                  : Source folder
               |
               +--- cg_src                          : CG Source folder
               |         r_cg_cgc.c
               |         r_cg_cgc.h
               |         r_cg_cgc_user.c
               |         r_cg_dbsct.c
               |         r_cg_dtc.c
               |         r_cg_dtc.c
               |         r_cg_dtc_user.c
               |         r_cg_hardware_setup.c
               |         r_cg_icu.c
               |         r_cg_icu.h
               |         r_cg_icu_user.c
               |         r_cg_intprg.c
               |         r_cg_macrodriver.h
               |         r_cg_main.c
               |         r_cg_port.c
               |         r_cg_port.h
               |         r_cg_port_user.c
```

Figure 5-10 File Structure (1/2)

```
|        r_cg_r12da.c
|        r_cg_r12da.h
|        r_cg_r12da_user.c
|        r_cg_resetprg.c
|        r_cg_sbrk.c
|        r_cg_sbrk.h
|        r_cg_sci.c
|        r_cg_sci.h
|        r_cg_sci_user.c
|        r_cg_stacksct.c
|        r_cg_tmr.sc
|        r_cg_tmr.h
|        r_cg_tmr_user.c
|        r_cg_userdefine.h
|        r_cg_vect.h
|        r_cg_vecttbl.h
|
+--- r_smm_src                                  : SMM Source folder
     |   r_smm_if.h
     |
     +--- src
              r_smm.c
              r_smm_cgw.h
              r_smm_driver.c
              r_smm_driver.h
              r_smm_private.h
```

Figure 5-11 File Structure (2/2)

5.7 System Requirement

The following are required for the execution of the sample project:

- e2studio version: 6.2.0 or above
- RX Family C/C++Compiler Package version: CC-RX 2.07.00 or later
- Language Configuration: C(C99) (-lang=c99)

5.8 Procedure to Execute the Sample Application

The following describes the steps to build and execute the sample application.

(1) **Import the sample project into e2studio workspace by clicking "Import" from File Menu.**

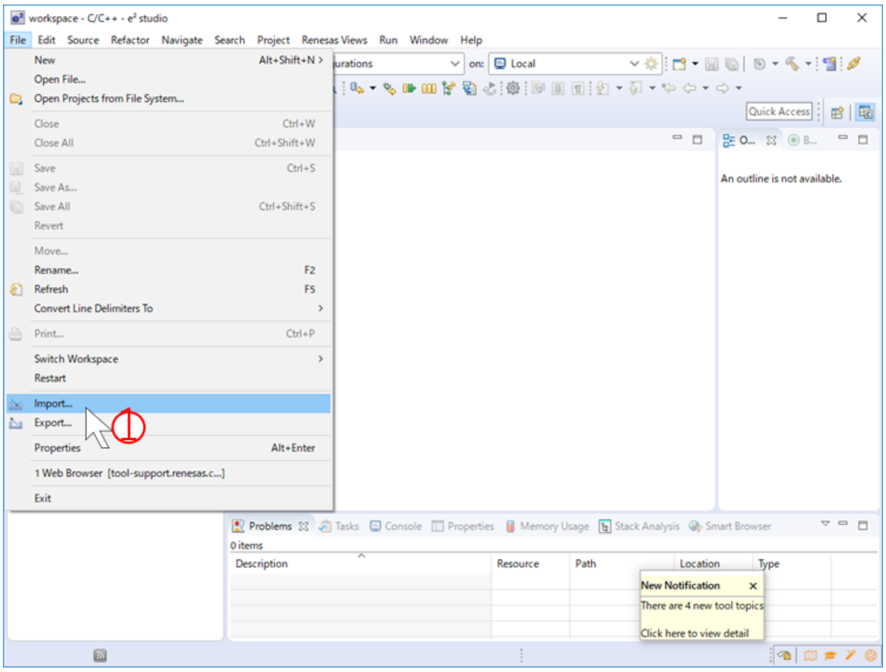


Figure 5-12 Select "Import" Menu

(2) **Select "Existing Projects into Workspace".**

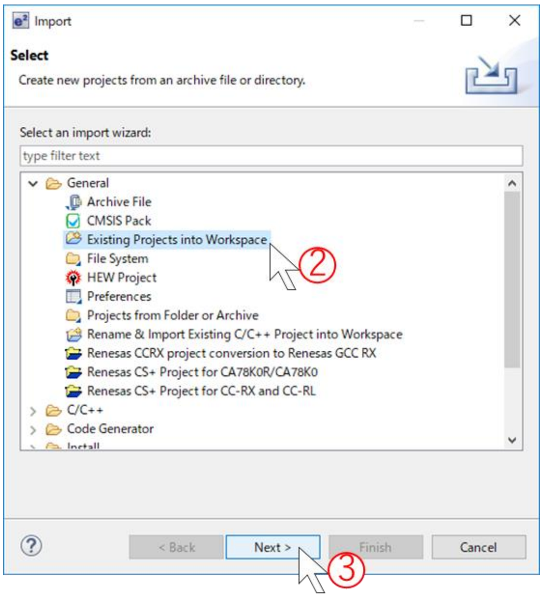
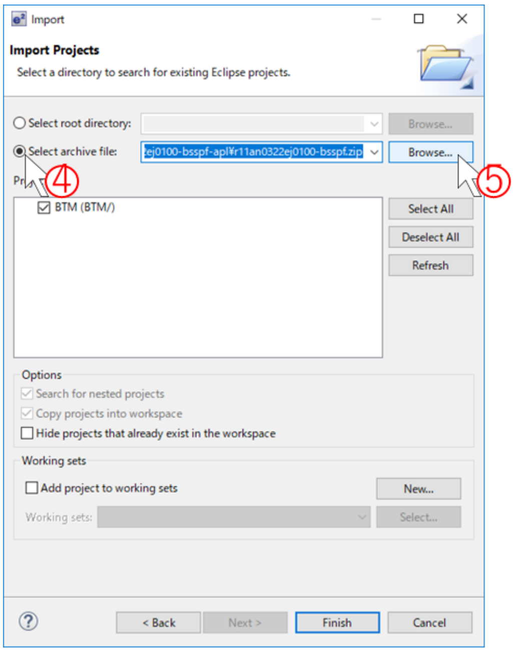


Figure 5-13 Select "Existing Projects into Workspace"

(3) **In the next window, choose “Select archive file:” and browse to the directory of “r11an0325ej0101-bsspf-smm.zip”.**



Import

Import Projects

Select a directory to search for existing Eclipse projects.

○ Select root directory:

● Select archive file: ej0100-bsspf-apl¥r11an0322ej0100-bsspf.zip

☑ BTM (BTM/)

Options

☑ Search for nested projects

☑ Copy projects into workspace

☐ Hide projects that already exist in the workspace

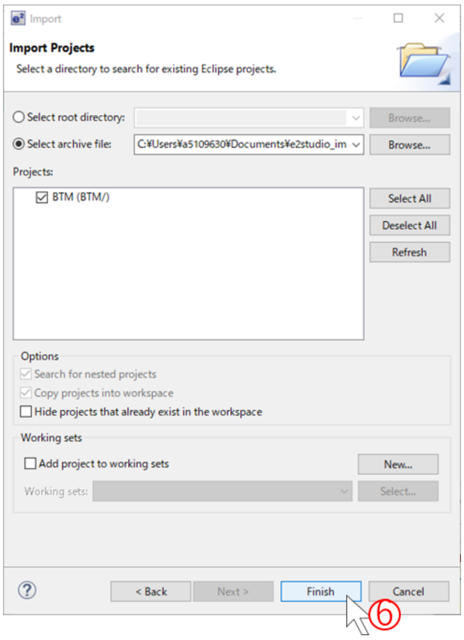
Working sets

☐ Add project to working sets

Working sets:

Figure 5-14 Choose “Select archive file:” and Select the Archived File

(4) **After selecting the archive file, the projects it contains will be listed down as shown. Click “Finish” to finish the importing.**



Import

Import Projects

Select a directory to search for existing Eclipse projects.

○ Select root directory:

● Select archive file: C:¥Users¥a5109630¥Documents¥e2studio_im

Projects:

☑ BTM (BTM/)

Options

☑ Search for nested projects

☑ Copy projects into workspace

☐ Hide projects that already exist in the workspace

Working sets

☐ Add project to working sets

Working sets:

Figure 5-15 Click “Finish”

6. References

- User’s Manual for RX231:

  The latest version can be downloaded from the Renesas Electronics website.
- User’s Manual for Renesas Starter Kit for RX231 (R0K505231S020BE):

  The latest version can be downloaded from the Renesas Electronics website.
- Datasheet for SAIC101 (RAA730101):

  The latest version can be downloaded from the Renesas Electronics website.
- User’s Manual Datasheet for SAIC101 (RAA730101):

  https://www.tessera.co.jp/Download/TSA_OP_IC101_UM_E_V1_00.pdf
- Equivalent Circuit Model to Simulate the Neuromuscular Electrical Stimulation
- Bioimpedance and Bioelectricity Basics by S. Grimnes and O. Martinsen (Academic Press, 2008)
- Inquiry for Training Potentiostat (TM-3000):

  http://www.ec-frontier.co.jp/

  Please contact info@ec-frontier.co.jp.

Website and Support

Renesas Electronics Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/contact/

Revision History

Rev.	Date	Description Page	Summary
1.00	Jul 31, 2018	-	1st Released
1.01	Sep 14, 2018	25 to 26	Changed “5.6 File Configurations”.
		26	Added “5.7 System Requirement” and “5.8 Import procedure”.
		27 to 28	Updated SALES OFFICE page.
		29	Updated “6. References”

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Handling of Unused Pins

   Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.

   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment when power is supplied.

   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

   Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

   — The characteristics of Microprocessing unit or Microcontroller unit products in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. 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 or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving 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, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
6. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, 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 and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall 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. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
11. This document shall not be reprinted, 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.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

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

(Rev.4.0-1 November 2017)

RENESAS

SALES OFFICES

Renesas Electronics Corporation

http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics Corporation
TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

Renesas Electronics America Inc.
1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709 Quantum Plaza, No.27 ZhichunLu, Haidian District, Beijing, 100191 P. R. China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, 200333 P. R. China
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338

Colophon 7.2