Bio Sensing Software Platform

Software Library for Measuring Respiratory Rate

R11AN0324EJ0101
Rev.1.01
Sep 14, 2018

Abstract

This document describes the sample program to obtain the respiratory rate using accelerometer.

Target Device

RX231

Target Board

- Renesas Starter Kit for RX231 (R0K505231S000BE) (Renesas Electronics)

  Hereafter, it is abbreviated as RX231 RSK.
- Accelerometer module: Pmod™ ACL2 (Digilent Inc.)

  SPI interface module included Analog Devices ADXL362

Contents

1. Introduction

This application note describes how to measure the respiratory rate with RX231 and accelerometer chip.

The respiratory rate means the respiratory count per minute at rest.

1.1 Terminology

Table 1-1 Terminology

Term	Meaning
API	Application Programming Interface
BPM	Beat Per Minute
DSP	Digital Signal Processor.
FIFO	First-In First-Out type buffer
FPU	Floating-Point number processing Uni
HPF	High Pass Filter
LPF	Low Pass Filter
MEMS	Micro Electro Mechanical Systems
LSB	The Least Significant Bit
MSB	The Most Significant Bit
RRM	Respiratory Rate Measuring

1.2 Overview

This application note answers the following topics:

- Mechanism of respiration.
- How to use an accelerometer device for respiration acquisition.
- Algorithm of calculation of respiratory rate.

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.

- Sensor: ADXL362 by Analog Devices, Inc.

  ADXL362 is an ultralow power, 3-axis MEMS accelerometer.

  - 1.8[uA] at 100[Hz] ODR, 2.0[V] supply
  - 1[mg/LSB]
  - Adjustable threshold sleep/wake mode for motion activation
  - Autonomous interrupt processing
  - Deep embedded FIFO
  - Low noise down to 175[ug/√Hz]
  - Selectable measurement range
  - External trigger for sampling
  - SPI digital interface

2. Functional Purpose

Respiration rate is the number of times a person’s breaths per minute. The rate is usually measured when a person is at rest and simply involves counting the number of breaths for one minute by counting how many times the chest rises. Respiration rates may increase when an individual is indulged in some physical activity such as exercise, walking, swimming etc.

In this application note, the system will cover the range of rate from 12 BPM to 100 BPM in a stationary state.

3. Measurement Principle

3.1 Measurement Value

The Respiratory Rate Module calculates Respiration Rate per minute.

The Respiration Rate value will be stored in global variable.

3.1.1 Equations

Considering the case where the accelerometer is worn on the chest as following.

During Respiration, average circumferential chest expansion is 4 mm to 8 mm. However, outward expansion of chest during respiration is approximately 2 mm to 4 mm.

Accelerometer ADXL362 has ability to measure displacements [mm] in all three axes (X, Y and Z axis). In this Application note, to measure the respiration rate Accelerometer ADXL362 is placed on the chest. Thus the outward expansion i.e. to and from displacement of chest is denoted by Acceleration due to Respiration (Ar) along Z-axis.

The ADXL362 has selectable measurement ranges of ±2G, ±4G, and ±8G, with a resolution of 1 mG/LSB on the ±2G range. Acceleration samples are always converted by a 12-bit ADC; therefore, sensitivity scales with g range. In ADXL362 1 LSB corresponds to 5mm displacement at 0.1 Hz frequency of acceleration.

Following is the relationship between displacement and acceleration:

$$D = (G*A)/(2\pi^2*f^2)$$

Where:

D - Displacement [mm]
G - Constant (9.80 [m/sec^2])
A - Acceleration Recorded [g]
F - Frequency of Acceleration [Hz]

For this Application,

Considering values for the parameters,

G= 9.8 m/sec^2
A - 0.001 G
F - 0.3 Hz

Displacement i.e. outward expansion of chest during respiration along Z-axis is measured as 5.52 mm.

3.1.2 Biquad Filter for LPF and HPF

Direct form 1 Biquad filter is used for LPF and HPF. The flow graph of direct form 1 Biquad filter is given below,

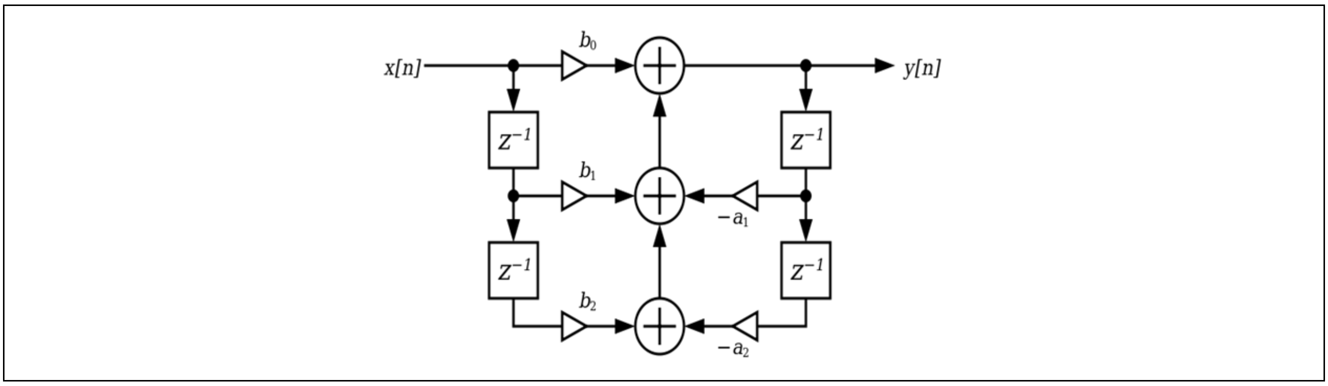

Figure 3-1 Flow graph of Biquad filter (Direct Form 1)

(1) **Biquad Filter Equation**

The normalized direct form 1 equation is given below,

$$y[n] = b0x[n] + b1x[n-1] + b2x[n-2] - a1y[n-1] - a2y[n-2]$$

Where,

b0, b1 and b2 are the coefficients determining zero
a1 and a2 are the coefficients determining poles

(2) **Filter Coefficient Calculation**

Butterworth filter has been chosen due to its low ripple in the pass band and stop band region than other filter types.

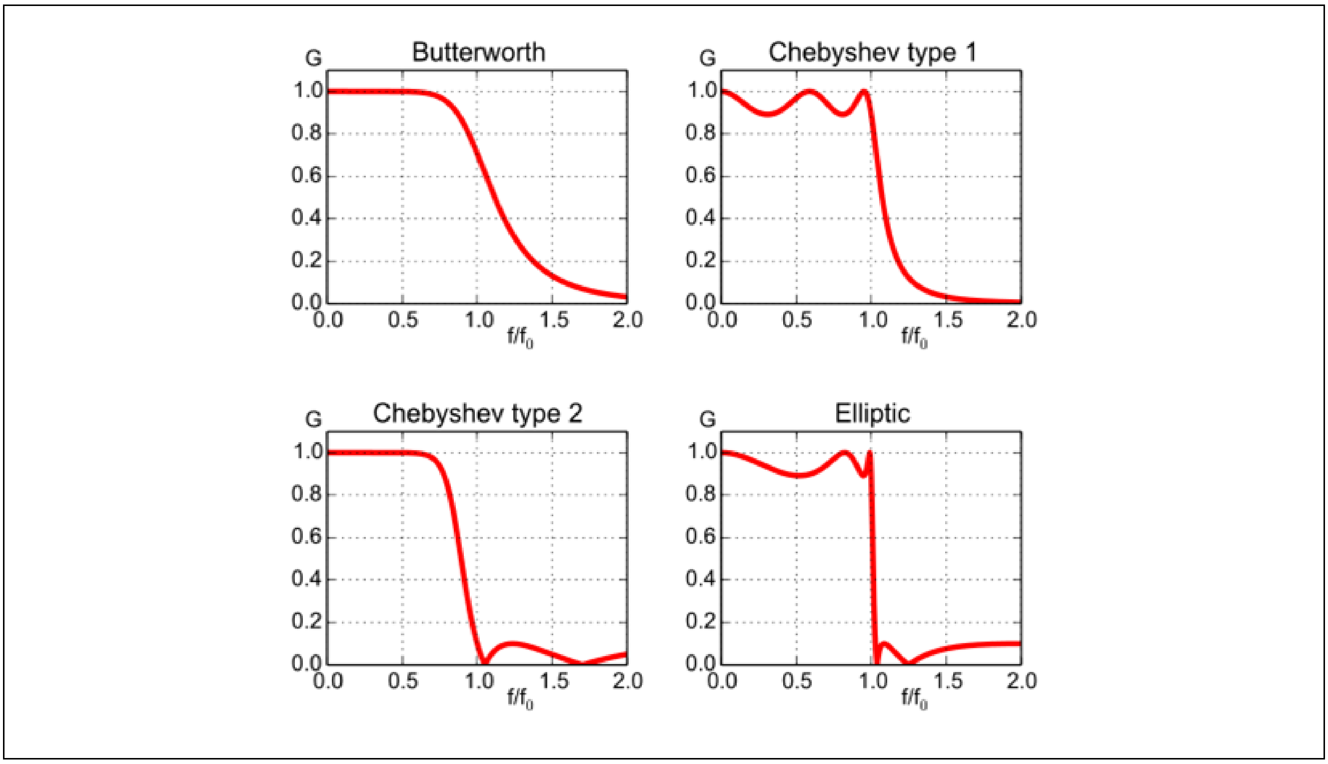

Figure 3-2 Gain distribution of discrete-time filters

The Butterworth filter coefficients can be calculated by below formula,

For LPF,

w0 = 2*pi*f0/Fs
b0 = (1 – cos (w0))/2
b1 = 1 – cos (w0)
b2 = (1 – cos (w0))/2
a0 = 1 + (sin (w0)/ (2*Q))
a1 = -2*cos (w0)
a2 = 1 – (sin (w0)/ (2*Q))

For HPF,

w0 = 2*pi*f0/Fs
b0 = (1 + cos (w0))/2
b1 = - (1 + cos (w0))
b2 = (1 + cos (w0))/2
a0 = 1 + (sin (w0)/ (2*Q))
a1 = -2*cos (w0)
a2 = 1 – (sin (w0)/ (2*Q))

Where,

f0 = cut-off frequency
Fs = sampling frequency
Q = Quality factor (0.7071)

The acceleration data on Z-Axis with LPF and HPF applied is below,

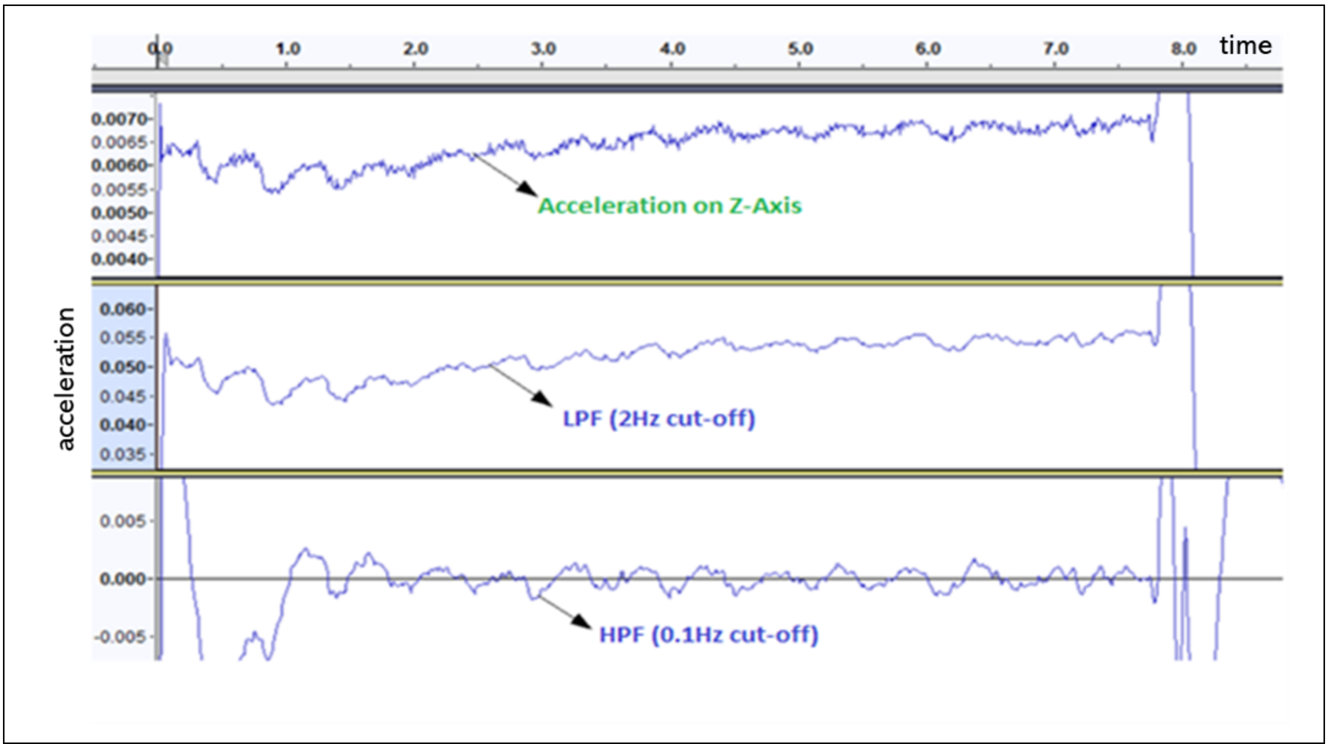

Figure 3-3 Acceleration samples representation

In the HPF result the number of peaks crossing the 0 baseline can be considering as a respiration.

(3) **Filter Details**

(a) **LPF Details:**

ORDER: 2nd order IIR Biquad filter

Type: Butterworth filter

Cut-off: 2 Hz

ADXL Sampling rate: 12.5 Hz (Ultra-low noise mode)

ADXL FILTER: Low Pass (Antialiasing) Filter with cut-off 3.125 Hz, -3 dB Corner

(b) **HPF Details:**

ORDER: 2nd order IIR Biquad filter

Type: Butterworth filter

Cut-off: 0.1 Hz

ADXL Sampling rate: 12.5 Hz (Ultra-low noise mode)

In the HPF result, this system measures the difference of peak from bottom value of one cycle. And if the difference is greater than 0.0018[g], respiration counted is incremented. This threshold value was founded by the experiments.

3.1.3 Noise in the Acceleration Data

The below picture shows the noise present in the acceleration data even after applying LPF,

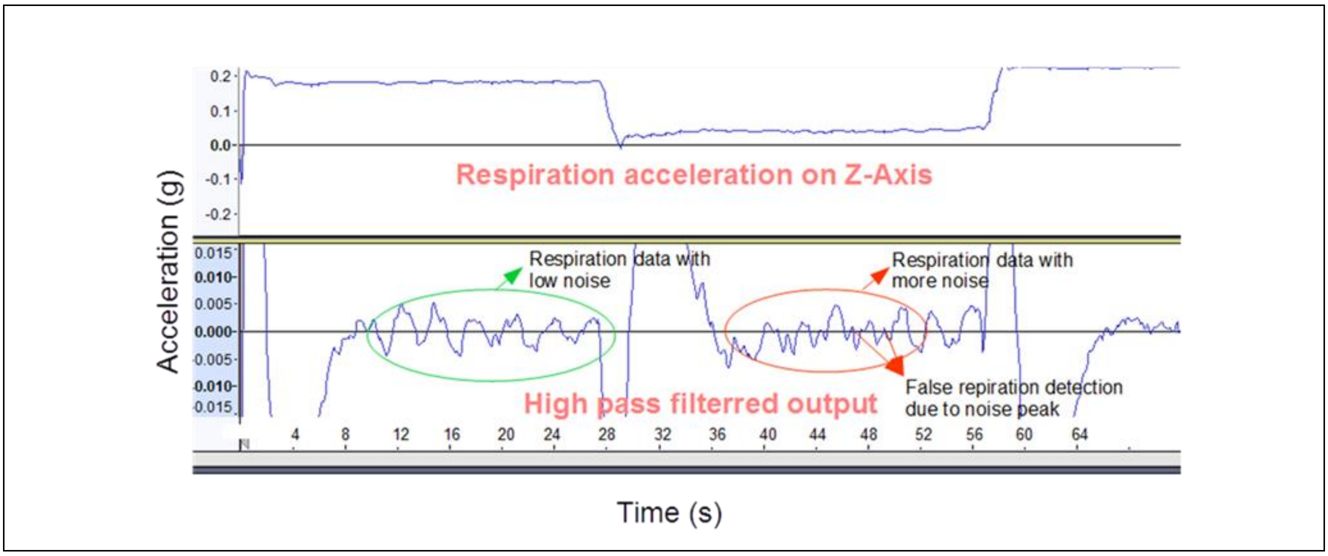

Figure 3-4 False respiration detection due to noise peaks

Since these noise peaks have a such respiration acceleration characteristics, these are detected as respiration which leads to inaccuracy in respiratory measurements.

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_RRM_StartSampling(void)	Starts the operation of MCU peripheral devices.
R_RRM_StopSampling(void)	Stops the MCU peripheral devices' operation. It is used only if required.
R_RRM_Calculate(void)	Reset sampling complete flag. Calculates the Respiration Rate and stores the values in global variables.
R_RRM_Calculation_Init(void)	Initializes filter configuration.
R_RRM_IsSampleDataReady(void)	Returns the sampling complete flag. If the flag is set, the function returns true indicating the samples ready for calculation, otherwise returns false.

4.2 Global Variables

The Respiration Rate values are stored in the below global variable:

Table 4-2 List of Global Variables

Global Variables	Function
g_respiration_rate	integer value indicating respiration rate [breaths per minute]
g_sample_data_ready	Flag which indicates data sampling was completed for calculation
g_bytes_read	Total size of acquired acceleration data
g_handle_lpf	Parameters for software LPF
g_handle_hpf	Parameters for software HPF
g_sample_buffer[]	Array of sampled acceleration data arranged each axis by axis

4.3 Memory Size

Table 4-3 Memory Size

Memory	Size
ROM	1,226 bytes for RRM library [Note1]
RAM	5,958 bytes for RRM Library [Note1]
User Stack	96 bytes
Interrupt Stack	116 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 RRM 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.
- Accelerometer ADXL362, an ultralow power, 3-axis MEMS accelerometer that consumes less than 2 µA at a 100 Hz output data rate, 2.0 V supply and 270 nA when in motion triggered wake-up mode is used for activity detection. In addition to its ultra-low power operation, it provides below features which makes it suitable for the activity detection:
  - High resolution: 1 mg/LSB
  - Adjustable threshold sleep/wake modes for motion activation
  - Autonomous interrupt processing, without need for microcontroller intervention
  - Deep embedded FIFO minimizes host processor load
  - Acceleration sample synchronization via external trigger – host can control the sampling timing
- A Bi-directional Voltage level translator circuit is inserted between the RX231 RSK and the ADXL362 to give an external clock to the ADXL362 to maintain accuracy and to translate the voltage levels of the interface signals to the power supply voltage.

5.1.2 Hardware Block Diagram

The RRM will be developed to run on RX231 RSK (Renesas Starter Kit for RX231) and Accelerometer ADXL362.

The Block Diagram for the overall system is shown below:

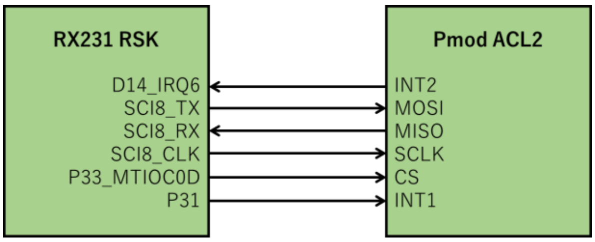

Figure 5-1 Hardware Block Diagram

The Specifications of the RX231 RSK and ADXL362 Accelerometer listed in the below tables respectively:

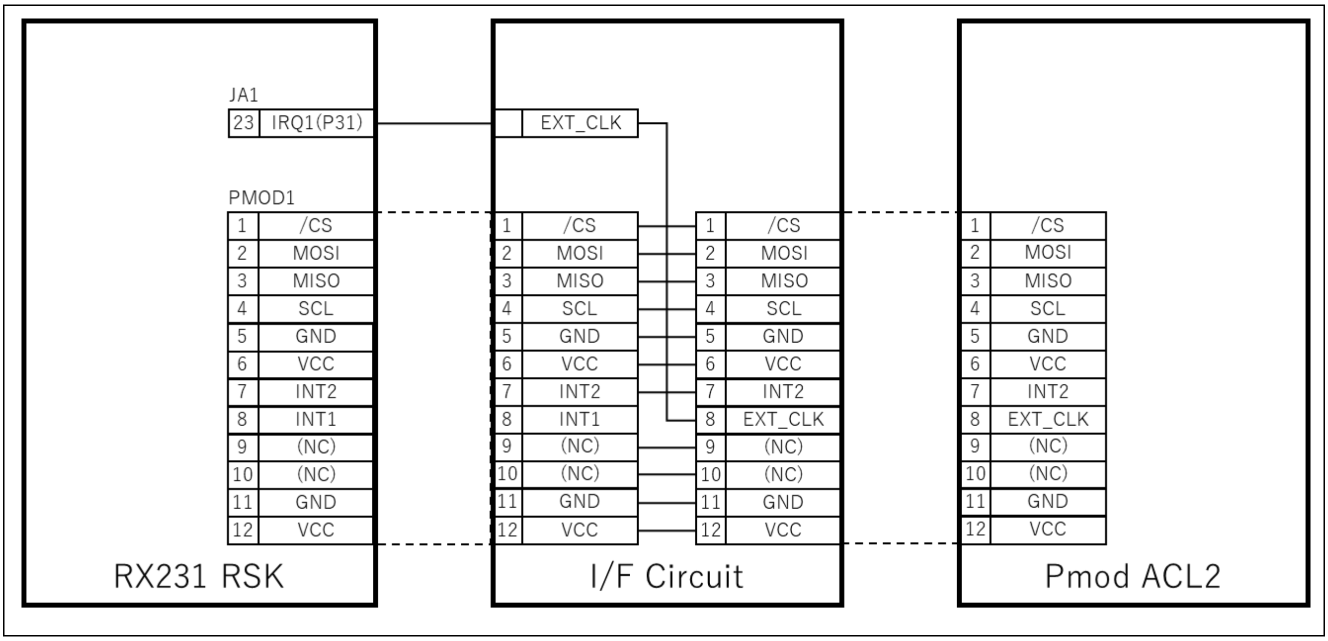

Figure 5-2 Hardware wiring chart

5.1.3 RX231 MCU Digital Interconnect

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

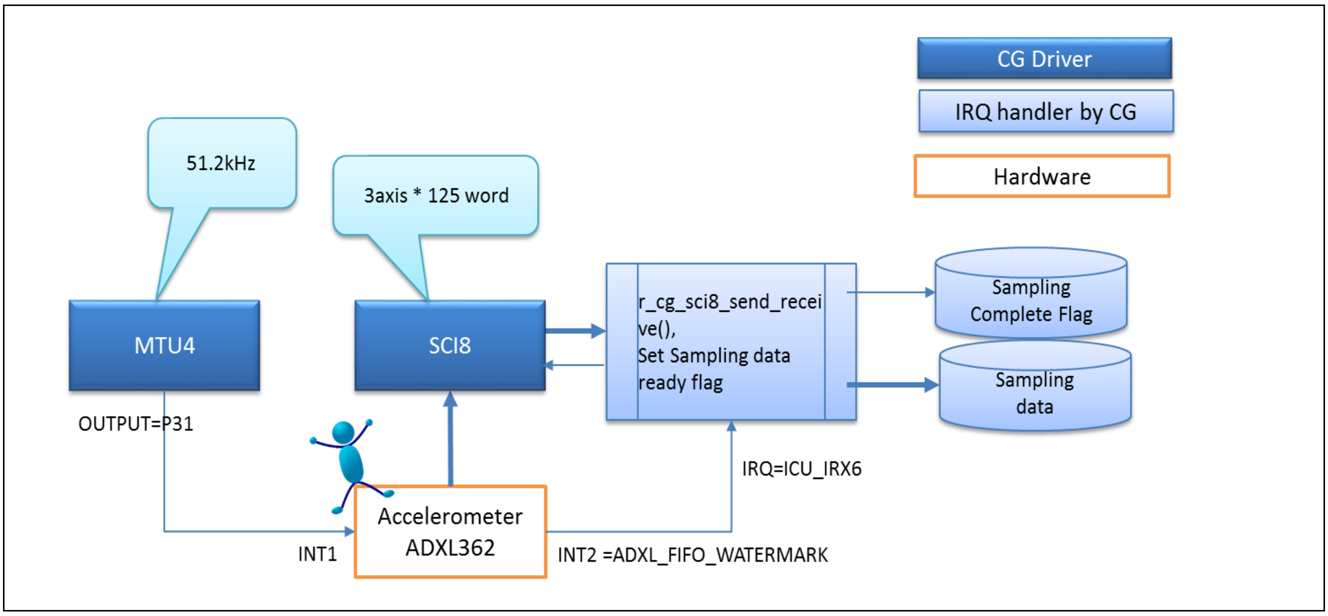

Figure 5-3 Respiratory Rate Measuring Digital Interconnect

- The EXTCLK for ADXL362 is oscillated by MTU4 which creates oscillation of 51.2 kHz.
- ADXL362 will be configured with Range=±2G, Sampling Rate=12.5Hz.
- ADXL362 will be configured to use Z axis value. X, Y axis and T (Temperature) is not used.

- ADXL362 will use FIFO for storing Data, WatermarkIRQ (10sec) for raising interrupt and will use ETXCLK for data Synchronization.
- Set Power Mode = Measure.
- ADXL362 is connected to SCI8 channel.
- When WatermarkIRQ occurs, SCI8 sends Read FIFO Command to ADXL362.
- On reading FIFO sampling data ready flag is set.

5.2 Software

5.2.1 Software Design Policy

The Software Design Policy is:

- RRM 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 Accelerometer 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 Respiration Rate is executed continuously using an infinite loop (while(1U) {}) in the main () function
- The calculate () function calculates Respiration Rate 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 RRM 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

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

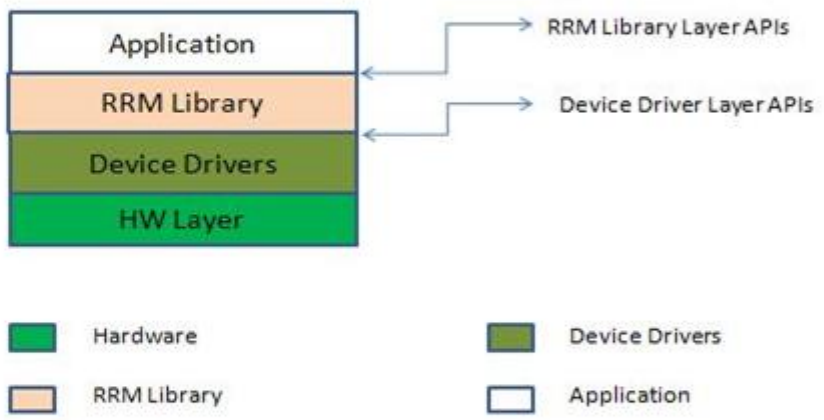

Figure 5-2 RRM Software Architecture

5.2.3 Measurement Signal Flow

The Respiratory Rate is measured using ADXL362 to indicate the respiration rate. The respiratory rate measurement flow is shown in the diagram below:

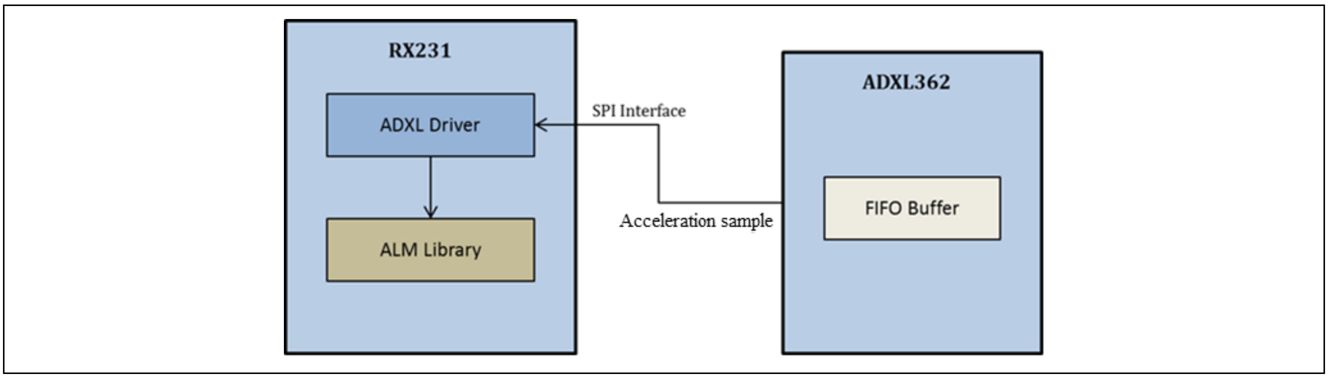

Figure 5-3 Interface of the ADXL362 Sensor with the MCU

5.2.4 Measurement Control Flow

The RRM Software implements a simple application to use the RRM Library. The application is automatically started on reset. The Respiratory rate 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:

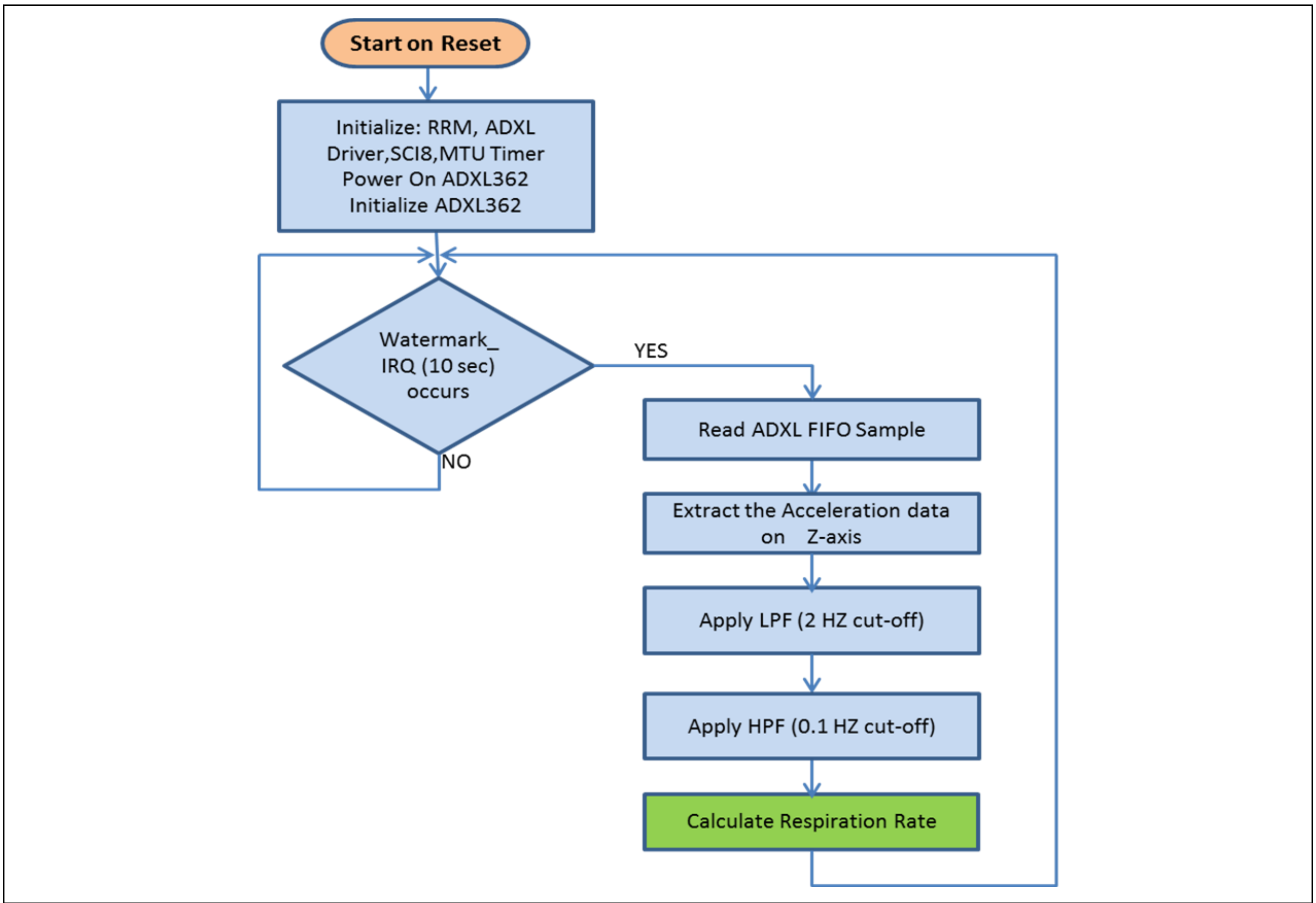

Figure 5-4 RRM Software Control Flow

5.3 Device Drivers

5.3.1 ADXL 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_ADXL362_Initialize(void)	Initializes the accelerometer sensor (ADXL362) by setting range, sampling rate, watermark interrupt etc.
R_ADXL362_Fetch(void)	Receives the sample data through SCI and Sets the sampling complete flag.

(2) **Conditions**

For ADXL Configuration, refer to "5.1.3 RX231 MCU Digital Interconnect".

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)

Interrupt Controller Unit	ICU	IRQ6	Pin	PE6
			Digital filter	No filter 0(MHz)
			Valid edge	Falling
			Priority	Level 15 (highest)
I/O Ports	Port3	P33	Mode	Out
			CMOS output	Used
			output value	1
Multi-Function Timer Pulse Unit 2	MTU2_U0	MTU4	Counter clock selection	PCLK/4
			Clock edge setting	Rising edge
			Counter clear source	TGRD4 compare match/input capture
			TGRA4 (Output compare register)	9.765625µs, (Actual value: 10)
			TGRB4 (Output compare register)	9.765625µs, (Actual value: 10)
			TGRC4 (Output compare register)	9.765625µs, (Actual value: 10)
			TGRD4 (Output compare register)	19.5µs, (Actual value: 19.5)
			MTIOC4A pin (PA0)	MTIOC4A pin output disabled
			MTIOC4B pin (P30)	MTIOC4B pin output disabled
			MTIOC4C pin (PB1)	MTIOC4C pin output disabled
			MTIOC4D pin (P31)	Initial output of MTIOC4D pin is 0. Toggle output at compare match.
			Enable TGRA4 input capture/compare match interrupt (TGIA4)	Used
			(TGIA/TGIB/TGIC/TGID) Priority	Level 15 (highest)
Serial Communications Interface	SCI8		Function setting	Simple SPI bus (Master transmit/receive)
			SMOSI8	PC7
			SMISO8	PC6
			Transfer direction setting	MSB-first
			Data inversion setting	Normal
			Transfer clock	Internal clock
			Bit rate	5000 (bps)
			SCK8 pin function selection	Clock output
			SCK8	PC5
			Clock delay	Clock is not delayed
			Transmit data handling	Data handled in interrupt service routine
			Receive data handling	Data handled in interrupt service routine
			TXI8, RXI8, TEI8, ERI8 priority	Level 15 (highest)
			Enable error interrupt (ERI8)	Used
			Transmission end	Used
			Reception end	Used
			Reception error	Used

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		-
Interrupt Controller Unit	r_cg_icu.c		void R_ICU_Create(void)
		IRQ6	void R_ICU_IRQ6_Start(void)
			void R_ICU_IRQ6_Stop(void)
	r_cg_icu_user.c	IRQ6	static void r_icu_irq6_interrupt(void)
	r_cg_icu.h		-
I/O Ports	r_cg_port.c		void R_PORT_Create(void)
	r_cg_port_user.c		-
	r_cg_port.h		-
Multifunction timer pulse unit 2	r_cg_mtu2.c		void R_MTU2_Create(void)
		MTU4	void R_MTU2_C4_Start(void)
			void R_MTU2_C4_Stop(void)
	r_cg_mtu2_user.c	MTU4	static void r_mtu2_tgia4_interrupt(void)
	r_cg_mtu2.h		-
Serial Communication Interface	r_cg_sci.c	SCI8	void R_SCI8_Create(void)
			void R_SCI8_Start(void)
			void R_SCI8_Stop(void)
			MD_STATUS R_SCI8_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	SCI8	static void r_sci8_transmit_interrupt(void)
			static void r_sci8_transmitend_interrupt(void)
			static void r_sci8_receive_interrupt(void)
			static void r_sci8_receiveerror_interrupt(void)
			void r_sci8_callback_transmitend(void)
			void r_sci8_callback_receiveend(void)
			void r_sci8_callback_receiveerror(void)
	r_cg_sci.h		-

For details, refer to the following files.

- Function.html
  It is stored in the “an-r11an0324ej0100-bsspf-apl/workspace/RRM/doc” folder
- Macro.html
  It is stored in the “an-r11an0324ej0100-bsspf-apl/workspace/RRM/doc” folder

5.4 Application Framework

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

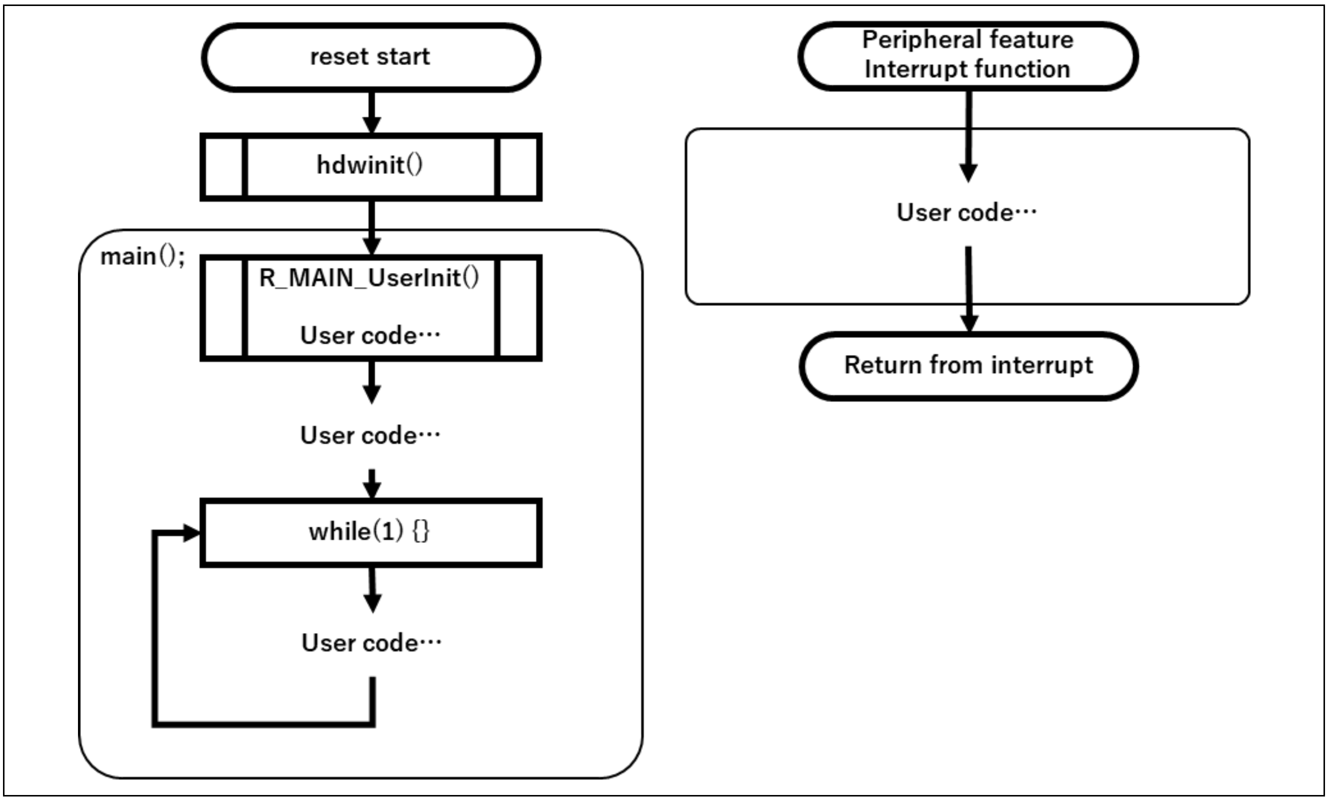

Figure 5-5 Application Framework by the Code Generator

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 ADXL362 will be implemented in R_MAIN_UseInit () function.
2. User Code Section - User code to start each of the device operation will 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

The application flow is shown in the flow chart below:

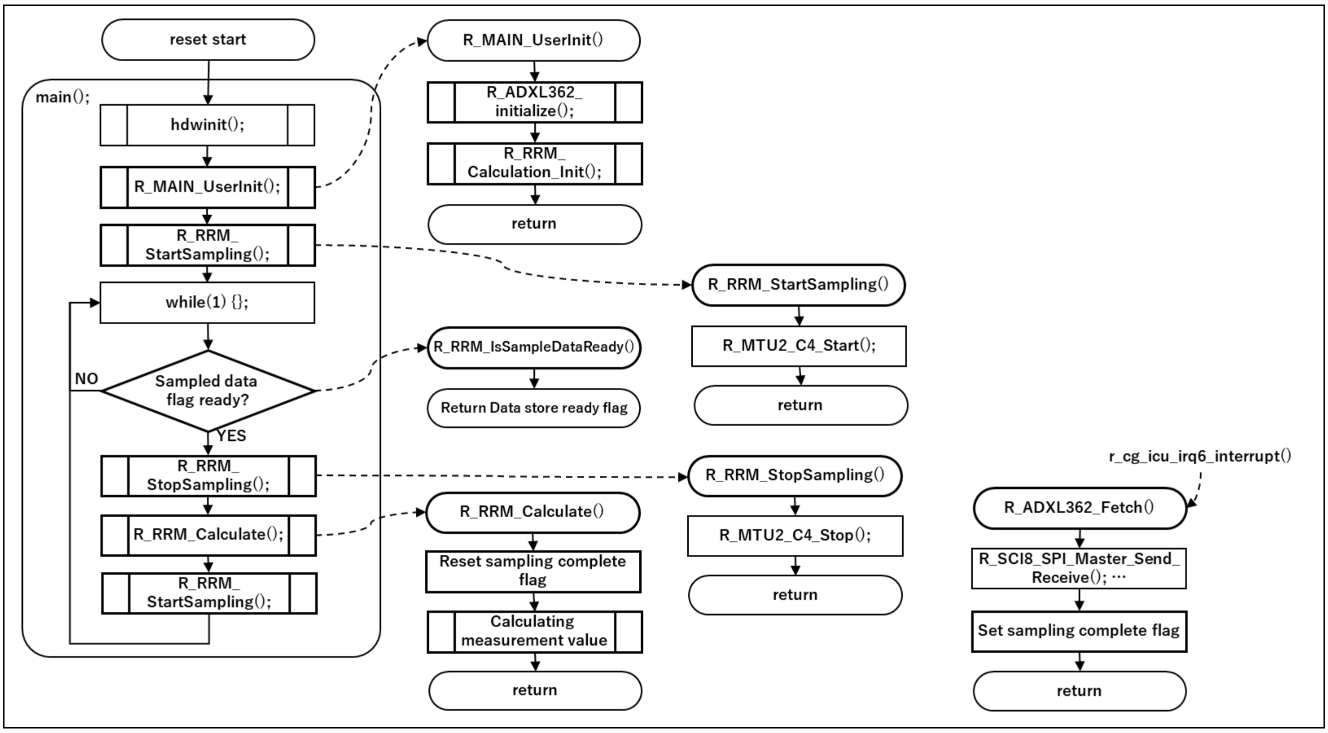

Figure 5-6 Software Flow Chart

The application does the below operations:

- Initialize the MCU, Peripherals and the ADXL362 on reset.
- Do below operations repeatedly

  Start Data Sampling

  Wait for Sampled data ready status (sampling complete flag to set)

  Stop Sampling once the Sampled data is ready, if required

  Reset Sampling complete flag

  Calculate Respiration Rate from the collected samples

  Start Data Sampling again

5.6 File Configurations

Figure 5-7 and Figure 5-8 show the file structure.

```
an-r11an0324ej0101-bsspf-apl
|     r11an0324ej0101-bsspf.pdf                 : This application note
|
+--- workspace                                  : Workspace folder
      |
      +--- RRM                                  : Project folder
            |     .cproject                     : ProjectDescription
            |     .project                      : ProjectDescription
            |     SM HardwareDebug.launch       : Launch Configuration
            |
            +--- .settings                      : Configuration folder of e2studio (Omit details)
            |
            +--- demo
            |         r11an0324ej0101-bsspf-rrm.zip : Archived file of this project (Omit details)
            |
            +--- doc
            |         Function.html             : Function Table file for CG
            |         Macro.html                : Macro Table file for CG
            |
            +--- generate                       : generate folder
            |         iodefine.h                : IO definition file
            |
            +--- src
                  |
                  +--- cg_src                   : CG Source folder
                  |         r_cg_cgc.c
                  |         r_cg_cgc.h
                  |         r_cg_cgc_user.c
                  |         r_cg_dbsct.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_mtu2.c
                  |         r_cg_mtu2.h
                  |         r_cg_mtu2_user.c
                  |         r_cg_port.c
                  |         r_cg_port.h
                  |         r_cg_port_user.c
                  |         r_cg_resetprg.c
                  |         r_cg_sbrk.c
                  |         r_cg_sbrk.h
```

Figure 5-7 File Structure (1/2)

```
|          r_cg_sci.c
|          r_cg_sci.h
|          r_cg_sci_user.c
|          r_cg_stacksct.c
|          r_cg_userdefine.h
|          r_cg_vect.h
|          r_cg_vecttbl.h
|
+--- r_rrm_src                                  : RRM Source folder
     |     r_rrm_if.h
     |
     +--- src
               r_rrm.c
               r_rrm_cgw.h
               r_rrm_driver.c
               r_rrm_driver.h
               r_rrm_private.h
```

Figure 5-8 File Structure (1/2)

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.**

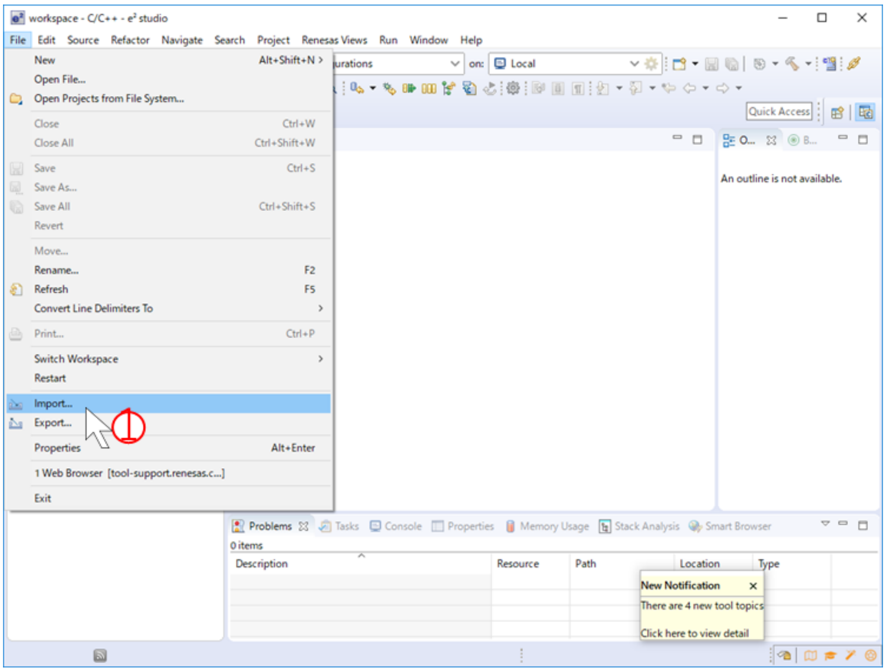

Figure 5-9 Select "Import" Menu

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

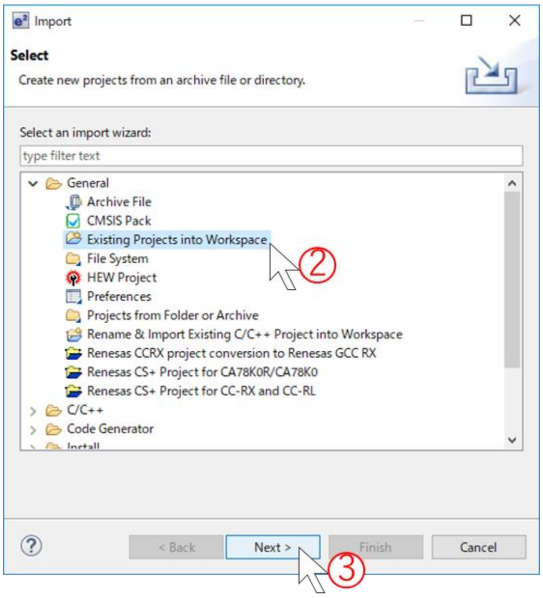

Figure 5-10 Select "Existing Projects into Workspace"

(3) **In the next window, choose "Select archive file:" and browse to the directory of "r11an0324ej0101-bsspf-rrm.zip".**

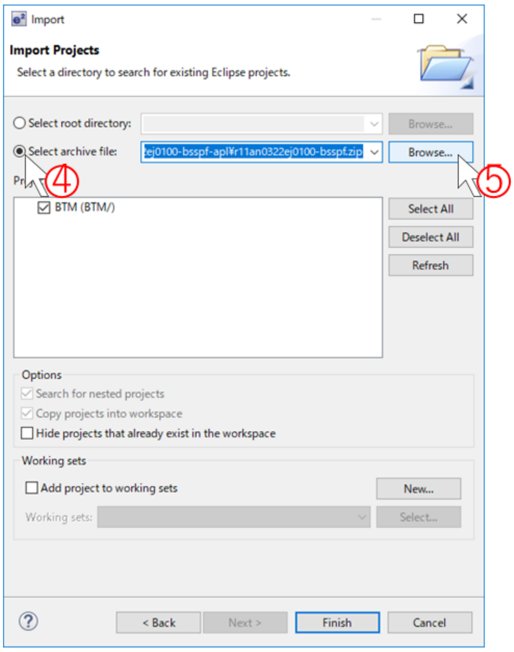

Figure 5-11 Choose "Select archive file:" and Select the Archived File

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

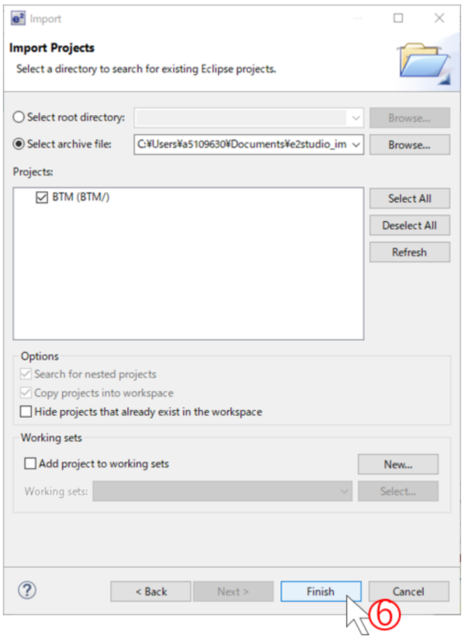

Figure 5-12 Click "Finish"

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.
- Pmod ACL2:

  https://store.digilentinc.com/pmod-acl2-3-axis-mems-accelerometer/
- ADXL362 Datasheet:

  http://www.analog.com/media/en/technical-documentation/data-sheets/ADXL362.pdf

Website and Support

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

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

PmodTM is a trademark of Digilent Inc.

All trademarks and registered trademarks are the property of their respective owners.

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Jul 31, 2018	-	1st Released
1.01	Sep 14, 2018	21 to 22	Changed “5.6 File Configurations”.
		22	Added “5.7 System Requirement” and “5.8 Import procedure”.
		23 to 24	Updated SALES OFFICE page.

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)

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

© 2018 Renesas Electronics Corporation. All rights reserved.
Colophon 7.2