RZ/A1H Group

TFT Backlight Control using Motor Control PWM Timer Peripheral

R30AN0230EG0100
Rev.1.00
16 Oct 2015

Introduction

This application note describes how to use the Motor Control PWM Timer Peripheral of the RZ/A1H microcontroller to provide a single variable duty-cycle PWM output. In this example, the PWM output is used to dynamically adjust the brightness of a TFT Display attached to an RSK+RZ/A1H demonstration/development board. This application note is supported by an embedded C project, developed in the e2 studio IDE.

Target Device

RZ/A1H

When applying the sample program covered in this application note to another microcontroller, modify the program according to the specifications for the target microcomputer and conduct an extensive evaluation of the modified program.

Contents

1. Specification

In this application, the Motor Control PWM Timer peripheral of the RZ/A1H microcontroller is configured to produce a 32kHz PWM waveform to the backlight control pin of the TFT display that connects to the RSK board. This PWM signal is dynamically controlled either by adjusting the potentiometer RV1 or by the switches SW1 – SW3 onboard the RSK. The application is set up to put the Renesas logo onto the display and a simple demonstration of the touch capabilities of the display.

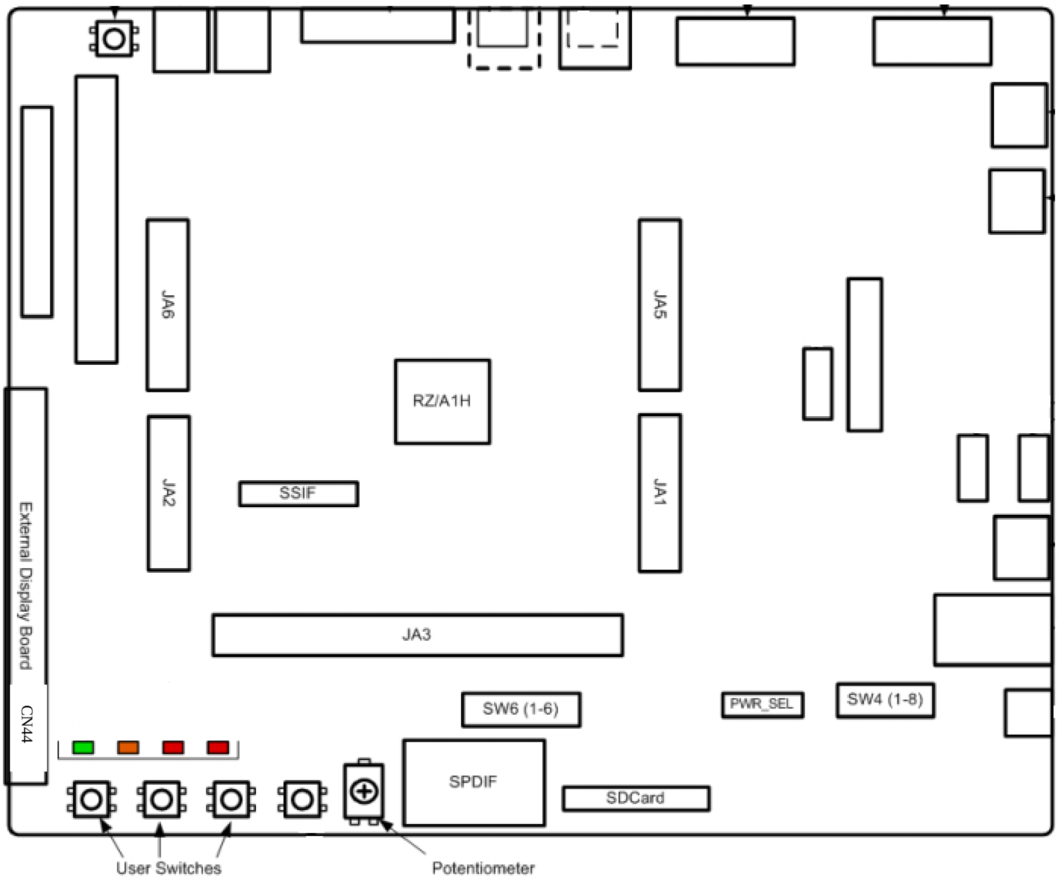

Figure 1-1 Board Layout

2. Application Environment

The sample code accompanying this application note has been run and confirmed under the conditions below.

Peripheral Function	Application
MCU used	RZ/A1H
Operating frequency	CPU clock (Iφ): 400 MHz Image processing clock (Gφ): 266.67MHz Internal bus clock (Bφ): 133.33MHz Peripheral clock 1 (P1φ): 66.67MHz Peripheral clock 0 (P0φ): 33.33MHz
Operating voltage	Power supply voltage (I/O): 3.3V Power supply voltage (internal): 1.18V
Integrated development environment (IDE)	e^2 studio (version 4.0.1.007)
C compiler	KPIT GNUARM-NONE-EABI Toolchain v.14.02
Debugger	Segger J-Link
Board used	RSK+RZ/A1H board and TFT display

Table 3.1 Application Environment

The peripherals used for the PWM backlight control and their function within the application are described in Table 2-1.

Peripheral	Function
Motor PWM Control Timer	This peripheral is configured to output a PWM signal from the pin PWM2B, operating from timer channel two.
Peripheral Clock 0 (P0φ) (32MHz)	Clock supply for motor control PWM timer peripheral
Interrupt Controller Compare Match Interrupt (INTC_ID_CMI2)	This interrupt is generated at the start of the PWM period. It is used to update the motor control PWM duty cycle.

Table 2-1 Required Peripheral Function and their Use

3. Hardware

3.1 Hardware Scheme

The PWM signal is generated by the MCU on port 3-1 (Pin AA7). This is routed to the backlight pin of the display, pin 34 of connector CN44, via an option link on the RSK. See section 3.2.3 for details of resistor link settings required for correct sample operation.

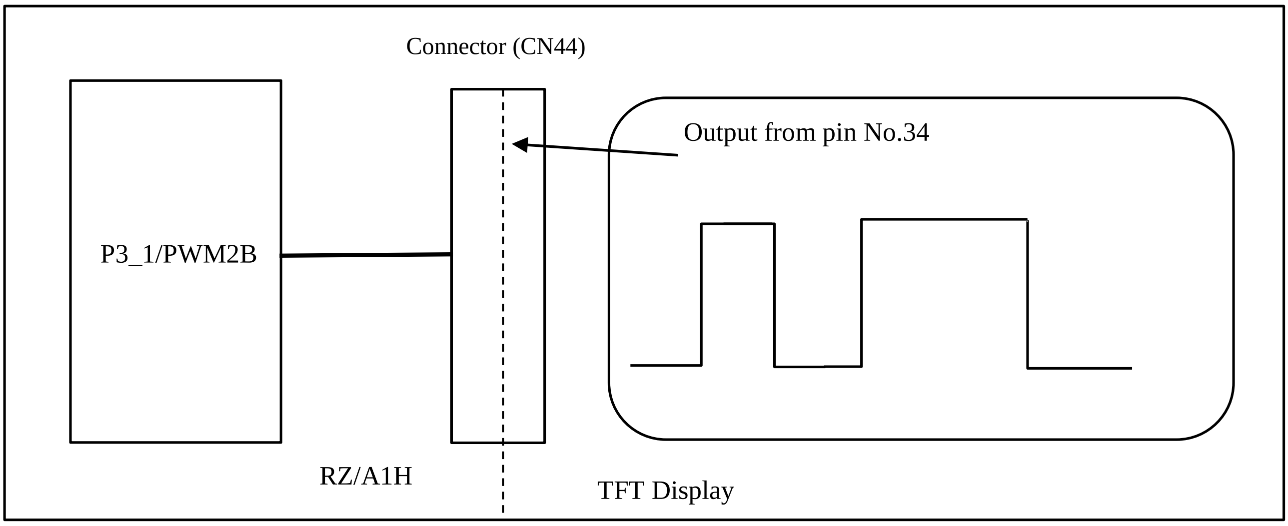

Figure 3-1 Connection of Backlight Control from MCU to Display

3.2 RSK Configuration

Please make sure that the RZA1H being used with this sample code has the following settings.

3.2.1 Switches

Please make sure that SW4 has the settings listed in Table 3-1.

SW4	SW4-1	SW4-2	SW4-3	SW4-4	SW4-5	SW4-6	SW4-7	SW4-8
State	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF

Table 3-1 SW4 Settings

Please ensure that SW6 has the settings specified in Table 3-2.

SW6	SW6-1	SW6-2	SW6-3	SW6-4	SW6-5	SW6-6
State	OFF	ON	OFF	ON	ON	ON

Table 3-2 SW6 Settings

3.2.2 Jumpers

Please ensure that the following only the following jumper are fitters on the board in use.

- JP11 1-2 connected
- JP12 1-2 connected
- JP21 1-2 connected
- JP18 1-2 connected
- PWR_SEL 2-3 connected if power supplied from 5V, 1-2 connected if power supplied from 12V. Please refer to RSK+RZ/A1H User's Manual section 2.1 for further information if required.

It is essential that if a 12V supply is used that PWR_SEL is ***NOT*** linked on pins 2-3 or an overvoltage will be applied to the MCU and associated devices, resulting in likely destruction of the whole board

3.2.3 Resistor modifications

Some resistor modifications are needed for the RSK+RZA1H in order to achieve the correct operation of the sample program, these are:

- Remove R106 (Disconnects IRQ_AUDIO signal from RZA1 port pin P3_1)
- Ensure that R107 is not fitted (not fitted as default). (Disconnects IRQ_PORTX signal from RZA1 port pin P3_1)
- Fit R186 (Connects BL_PWM_CTRL to RZA1H port pin P3_1)

The resistors locations are shown in Figure 3-2 below:

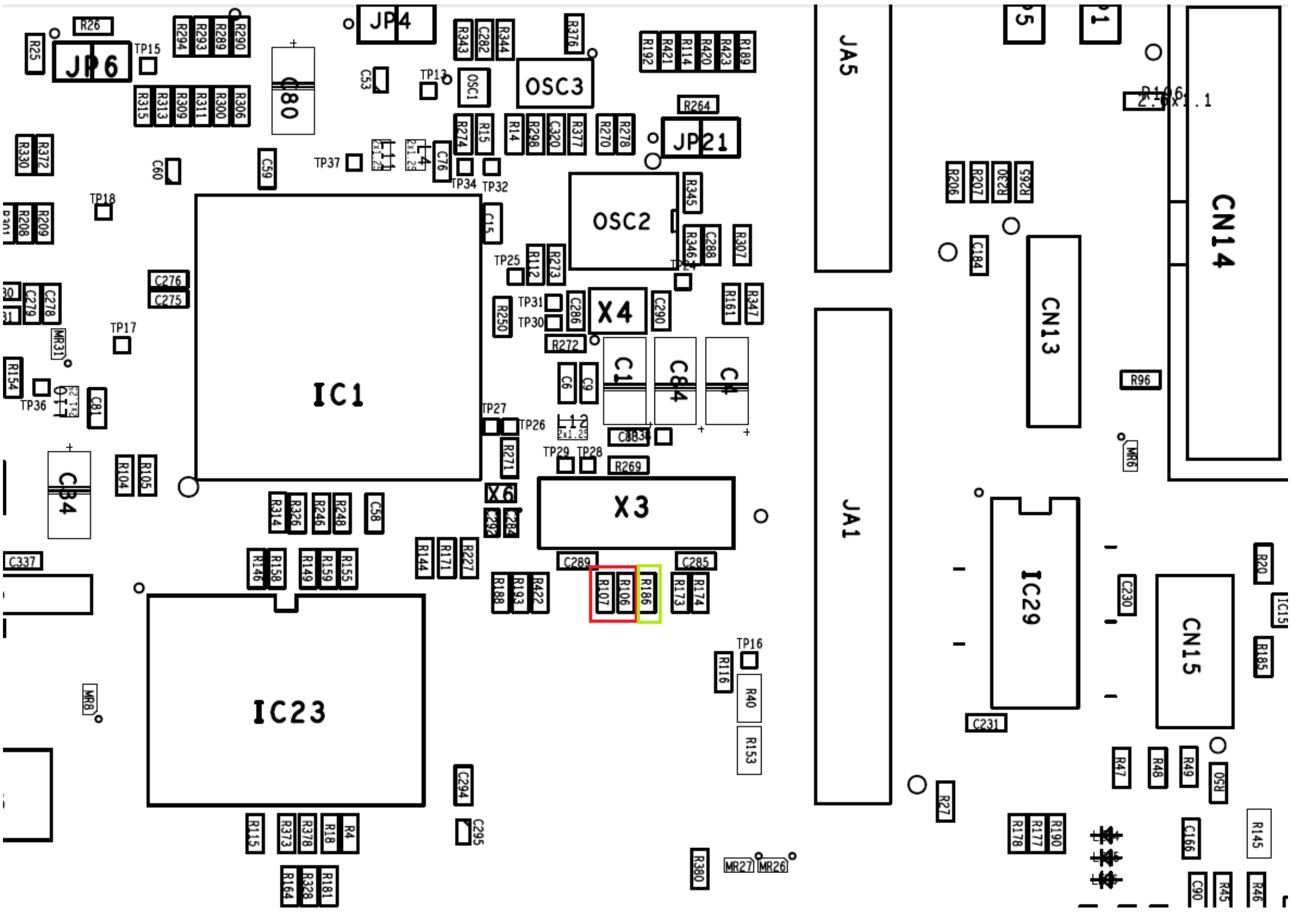

Figure 3-2 Location of modified resistors

4. Software

4.1 User Operation of Sample

Refer to the RSK Quickstart Guide (QSG) for instructions on how to import the project into the e^2 studio workspace. The sample code can be built and loaded on the RZA1H in either HardwareDebug or Release mode. When the program is running, the panel records up to 5 discrete low pressure touches. The brightness of the LCD panel can be controlled using the onboard potentiometer(RV1) and the switches (SW2 and SW3). SW1 can be used to used to toggle between controlling the brightness using the switches and the potentiometer.

4.2 Operation Overview

The Motor Control PWM Timer channel is configured to use channel two (PWM2B) to output a variable PWM waveform to the TFT display via port 3_1. Channel two is set to generate a compare match interrupt at the start of the PWM cycle. This interrupt is used to dynamically update the duty cycle of the PWM signal being delivered to the backlight control pin. As the Timer is 10bit, the programmable duty cycle varies from 0 to 1023 (0% to 100%). Table 4-1 provides an illustration of the relationship between them.

PWM Timer Duty Setting	LCD Screen Brightness (Duty Cycle) (%)
102	10
512	50
1023	100

Table 4-1 Relationship between PWM Output Ratios LCD Brightness

4.2.1 Operation Description

After the system initialisation, the program flow enters the main function in main.c, which calls the R_PWM_SampleMain function (in r_pwm_user.c). This initialises the PWM Motor Timer peripheral and then enters an infinite loop. Control of the PWM duty cycle, switch handling are carried out entirely within interrupt functions. Updating the screen is carried out in the background loop.

Duty cycle control is carried out in function “pwm_vary_duty” in r_pwm_user.c which is called when the PWM timer compare match interrupt is generated. If the code is in Pot mode, the current voltage on the pot RV1 is sampled here by the ADC, and applied to the PWM duty cycle. If the code is in Switch mode, the interrupt function checks for any switch presses and updates the duty cycle accordingly.

Switch control is carried out in function “pwm_controller” in r_pwm_user.c which is called when any of the IRQ interrupts are generated. It also incorporates a 10ms time-out delay, to minimize switch-bouncing effects. The interrupt function sets flags according to which switch was pressed, and these flags are picked up by the PWM timer interrupt function when it is updating the duty cycle.

Code for initialising and controlling the Motor Control PWM Timer is contained in the file r_pwm_motor.c.

4.2.2 Setting Cycle and Duty

The PWM output (PWB2B) is selected by setting OTS bit in PWDTR_2A register to 1. When compare match happens between PWCNT_2 (counter) and PWCYR_2 (period), the output pin PWM2B is set high, at the start of the cycle. PWM output goes low when another compare match occurs between PWCNT_2 (counter) and PWBFR_2A (duty register). In this sample code, the polarity of the PWM output has been set to direct output in the PWPR_2 register. Figure 4-1 shows graphically the operation of the period register PWCYR, the counter register PWCNT and the duty register PWBFR to generate the PWM waveform on PWM2B.

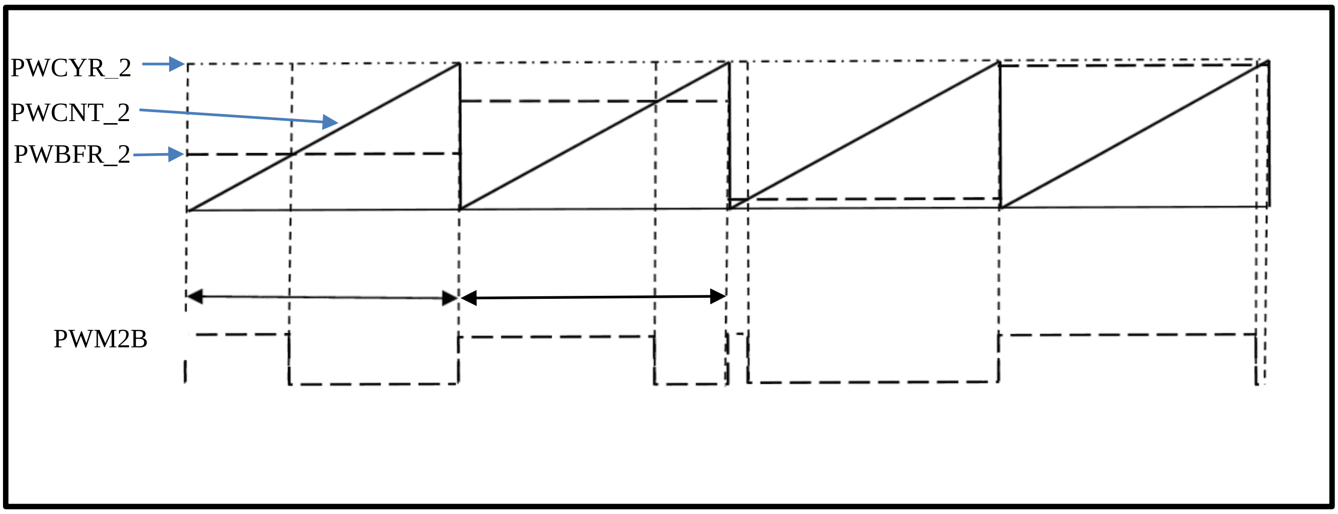

Figure 4-1 Description of PWM Operation

4.3 Peripheral Function Settings

Table 4-2 lists the peripheral settings.

Module	Settings
Motor Control PWM Timer	Channel 2 Count clock frequency: P0φ/1 = 33.33MHz PWM signal frequency: P0φ/1/1023 = 32.58kHz PWM signal period: 1023/33.33MHz = 30.7us Source of counter clear: PWCYR_2
Clock Pulse Generator (CPG)	Clock is supplied to Motor Control PWM Timer
Ports	Port 3_1 configured as peripheral controlled output PWM2B
Interrupt Controller	IRQ3 configured for SW1 (interrupt ID 35). IRQ2 configured for SW2 (interrupt ID 34). IRQ5 configured for SW3 (interrupt ID 37). Motor Control PWM Timer channel 2 compare match interrupt (CMI2, Interrupt ID: 165).

Table 4-2 Peripheral Function Setting

4.4 Interrupts

Table 4-3 lists interrupts used in the sample code.

Interrupt Source (Interrupt ID)	Priority	Processing Overview
CMI2 (165)	8	Interrupt generated by PWCYR_2 compare match. Set the value of the next duty cycle in the PWBFR_2A register.
ADI (ID:170)	5	Processing outline sets the flag for signaling the end of A/D conversion and clears the interrupt request ADF. Processing time: 964.8 ns (maximum)
IRQ2 (ID:34)	15	Interrupt generated when SW2 is pressed.
IRQ3 (ID: 35)	15	Interrupt generated when SW1 is pressed.
IRQ5 (ID: 37)	15	Interrupt generated when SW3 is pressed.

Table 4-3 Interrupts Used in Sample Code

4.5 Fixed Width Variables

Table 4-4 lists fixed-width variables used in sample code.

Symbol	Description
char_t	8-bit character
bool_t	Logical type. The value is true (1) or false (0).
int_t	High-speed integer, signed 32-bit integers are used in this sample code
int8_t	8-bit integer, signed (defined in the standard library)
int16_t	16-bit integer, signed (defined in the standard library)
int32_t	32-bit integer, signed (defined in the standard library)
int32_t	64-bit integer, signed (defined in the standard library)
uint8_t	8-bit integer, unsigned (defined in the standard library)
uint16_t	16-bit integer, unsigned (defined in the standard library)
uint32_t	32-bit integer, unsigned (defined in the standard library)
uint64_t	64-bit integer, unsigned (defined in the standard library)
float32_t	32-bit floating point number (defined in the standard library when "__ARM_NEON__" is specified)
float64_t	64-bit floating point number (defined in the standard library) (defined in the standard library when "__ARM_NEON__" is specified)
float128_t	128-bit floating point number

Table 4-4 Fixed-width Variables Used in Sample Code

4.6 List of Constants/Error Codes

Table 4-5 lists the constants used in the sample program and Table 4-6 lists the error codes.

Constant	Setting	Description
PWM_INT_PRIORITY1 PWM_INT_PRIORITY2	8u	PWM interrupt priority
PWM_SRC_CLK_MAX	5ul	Total number of different clock frequencies
PWM_CH1	0L	Channel 1 of the PWM unit
PWM_CH2	1L	Channel 2 of the PWM unit
POT_MODE	0u	To control the duty cycle using the potentiometer
SWITCH_MODE	1u	To control the duty cycle using SW2 and SW3
PWM_CNT_START	0x08u	PWM counter start
PWM_IE_ENABLE	0x20u	PWM interrupt enable
PWM_CMF_CLEAR	0x10u	Clear PWM CMF flag
PWM_CLK1	0x00u	P0φ
PWM_CLK2	0x10u	P0φ/2
PWM_CLK3	0x20u	P0φ/4
PWM_CLK4	0x30u	P0φ/8
PWM_CLK5	0x40u	P0φ/16
PWM_CLK_NG	0xFFu	Not a register value of counter clock
PWM_POLARITY_A	0x01u	PWM inverse output of A
PWM_POLARITY_B	0x02u	PWM inverse output of B
PWM_POLARITY_C	0x03u	PWM inverse output of C
PWM_POLARITY_D	0x04u	PWM inverse output of D
PWM_POLARITY_E	0x05u	PWM inverse output of E
PWM_POLARITY_F	0x06u	PWM inverse output of F
PWM_POLARITY_G	0x07u	PWM inverse output of G
PWM_POLARITY_H	0x08u	PWM inverse output of H
PWM_CLK_SUPPLY	0x01u	Halt PWM clock supply
PWM_OTS_A	0x0000u	Output terminal select PWMnA
PWM_OTS_B	0x1000u	Output terminal select PWMnB
PWM_OTS_C	0x0000u	Output terminal select PWMnC
PWM_OTS_D	0x1000u	Output terminal select PWMnD
PWM_OTS_E	0x0000u	Output terminal select PWMnE
PWM_OTS_F	0x1000u	Output terminal select PWMnF
PWM_OTS_G	0x0000u	Output terminal select PWMnG
PWM_OTS_H	0x1000u	Output terminal select PWMnH

Constant	Setting	Description
PWM_BUFFER_MASK	0x03FFu	Mask the control and reserved bits of register PWBFR
PWM_PERIOD_MASK	0x03FFu	Mask for period register
PWM_BTC1A_DISABLE	0x01u	Disable data transfer from register PWBFR_1A
PWM_BTC1C_DISABLE	0x02u	Disable data transfer from register PWBFR_1C
PWM_BTC1E_DISABLE	0x04u	Disable data transfer from register PWBFR_1E
PWM_BTC1G_DISABLE	0x08u	Disable data transfer from register PWBFR_1G
PWM_BTC2A_DISABLE	0x10u	Disable data transfer from register PWBFR_2A
PWM_BTC2C_DISABLE	0x20u	Disable data transfer from register PWBFR_2C
PWM_BTC2E_DISABLE	0x40u	Disable data transfer from register PWBFR_2E
PWM_BTC2G_DISABLE	0x80u	Disable data transfer from register PWBFR_2G
PWM_CYCLE	1023u	Value of PWM period
MAX_DUTY_CYCLE	1023u	Maximum value of PWM duty cycle
MIN_DUTY_CYCLE	10u	Minimum value of PWM duty cycle
INITIAL_DUTY_CYCLE	512u	Value to be used during initialization of PWM

Table 4-5 List of Constants

Constant	Setting	Description
MOTOR_PWM_SUCCESS	0L	Value to be returned on successful completion of code execution
MOTOR_PWM_ERROR_CLK	-1L	Non-existent clock error value
MOTOR_PWM_ERROR_UNSUPPORT_CLK	-2L	Wrong clock register value error value

Table 4-6 Error Codes from Sample Code

4.7 List of Structures/Unions

```
/*== the kind of clock count ==*/
typedef enum clock_count
{
    CLOCK_COUNT_1,                     /* internal clock P0/1  */
    CLOCK_COUNT_2,                     /* internal clock P0/2  */
    CLOCK_COUNT_4,                     /* internal clock P0/4  */
    CLOCK_COUNT_8,                     /* internal clock P0/8  */
    CLOCK_COUNT_16                     /* internal clock P0/16 */
} clock_count_t;


/*== the structure of a channel and a clock ==*/
typedef struct ch_clock
{
    clock_count_t clock;                   /* the kind of counter clock */

    /* register data of each channel at the time of clock */
    uint8_t       reg_data[PWM_CNT_CLK_MAX];
} ch_clock_t;
```

4.8 Function Specifications

This section contains the specifications for the functions that implement the backlight PWM controlin the sample code.

4.8.1 R_PWM_SampleMain

Synopsis	PWM sample program’s main processing section
Header	r_pwm_user.h
Source Location	r_pwm_user.c
Declaration	int32_t R_PWM_SampleMain(void)
Description	The main processing section of the PWM sample program. Sets up PWM control and switch handling, then runs graphics handling in background.
Arguments	none
Return	int32_t

4.8.2 R_PWM_Initialise

Synopsis	Initialise Motor Control PWM Timer unit
Header	r_pwm_motor.h
Source Location	r_pwm_motor.c
Declaration	int32_t R_PWM_Initialise(const clock_count_t clock_count1, const clock_count_t clock_count2)
Description	This function initialises the Motor Control PWM timer as follows: 1. Sets the PWM output to direct polarity 2. Sets the PWM clock to value from function parameters 3. Set up and enable PWM interrupts 4. Enable data transfer on compare match 5. Configure output port.
Arguments	`clock_count_t` clock_count1, `clock_count_t` clock_count2. Timer clock frequency. See 0 for details of the typedef.
Return	Result: 1. 0 - initialization successful 2. -1 – clock error 3. -2 – clock register error

4.8.3 R_PWM_SetDuty2A

Synopsis	Set PWM duty cycle for channel 2A
Header	r_pwm_motor.h
Source Location	r_pwm_motor.c
Declaration	void R_PWM_SetDuty (uint16_t pwm_duty)
Description	This function sets PWM duty cycle for channel 2A
Arguments	pwm_duty – 0 to 1023
Return	none

4.8.4 R_PWM_SetPeriod2

Synopsis	Set PWM period for channel 2
Header	r_pwm_motor.h
Source Location	r_pwm_motor.c
Declaration	void R_PWM_SetPeriod2 (uint16_t period)
Description	This function sets PWM period for counter2
Arguments	period - 0 to 1023
Return	none

4.8.5 R_PWM_StartTimer2

Synopsis	Start PWM counter 2
Header	r_pwm_motor.h
Source Location	r_pwm_motor.c
Declaration	void R_PWM_StartTimer (void)
Description	This function starts the PWM counter 2
Arguments	none
Return	none

4.8.6 R_PWM_StopTimer2

Synopsis	Stop PWM counter
Header	r_pwm_motor.h
Source Location	r_pwm_motor.c
Declaration	void R_PWM_StopTimer2 (void)
Description	This function stops PWM counter2
Arguments	none
Return	none

4.8.7 pwm_interrupt_cmf2

Synopsis	PWM interrupt handler for timer 2
Header	r_pwm_motor.h
Source Location	r_pwm_motor.c
Declaration	static void pwm_interrupt_cmf2 (uint32_t cmf)
Description	This function handles the compare-match ISR then calls the callback function.
Arguments	int32_t cmf
Return	None

4.8.8 R_PWM_CM2InterruptCallback

Synopsis	Registers a function to be called when a PWM interrupt is generated.
Header	r_pwm_motor.h
Source Location	r_pwm_motor.c
Declaration	static void R_PWM_CM2InterruptCallback(void)
Description	Registers a function to be called when an interrupt is generated for timer 2 compare-match
Arguments	void (*func) (void)
Return	None

4.8.9 pwm_controller

Synopsis	Handles switch presses
Header	r_pwm_user.h
Source Location	r_pwm_user.c
Declaration	static void pwm_controller (void)
Description	Run from switch IRQ interrupts, this function toggles between switches and potentiometer to control duty cycle, and controls flags to increment or decrement duty cycle based on switch presses
Arguments	None
Return	None

4.8.10 pwm_vary_duty

Synopsis	Alters the value of PWM duty cycle
Header	r_pwm_user.h
Declaration	static void pwm_vary_duty (void)
Description	Run from PWM Timer compare-match interrupt, this function alters the value of duty cycle based on switch presses or potentiometer ADC reading.
Arguments	None
Return	None

4.9 Flowcharts

4.9.1 Peripheral Initialisation

Run from then main function, the function init_periph initializes the RIIC, switches, LEDs and the ADC.

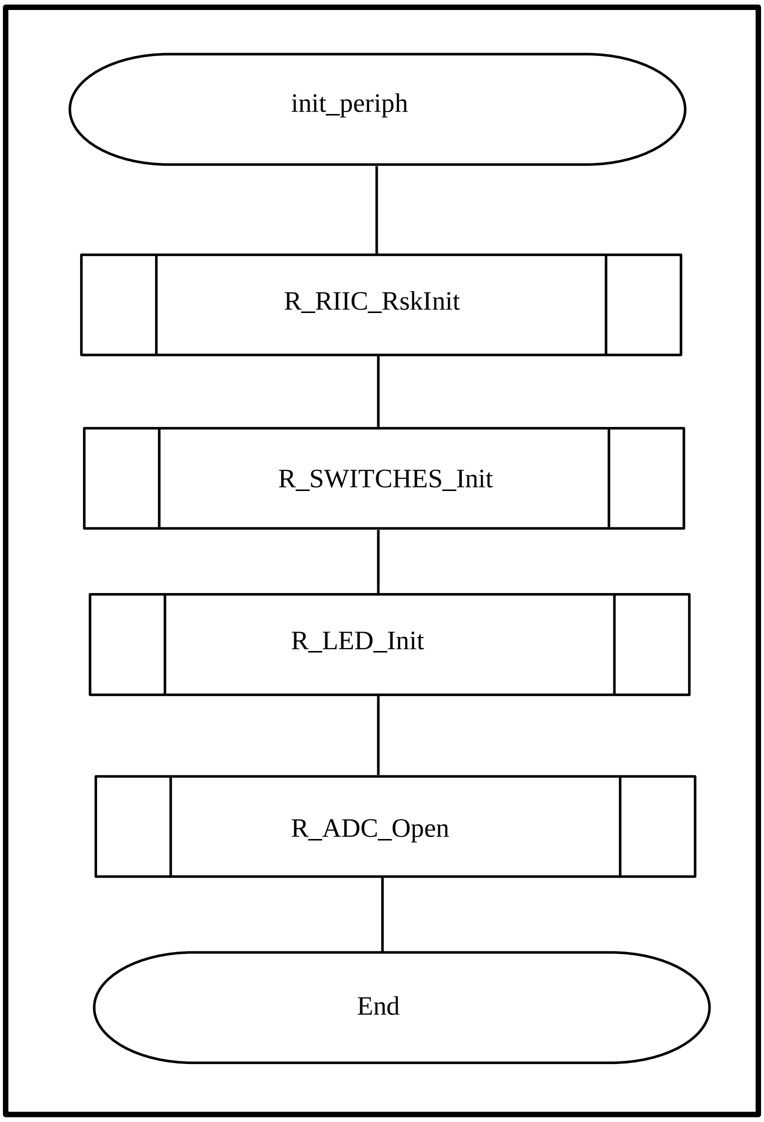

Figure 4-2 Peripheral Initialisation Flow Chart

4.9.2 Main Processing

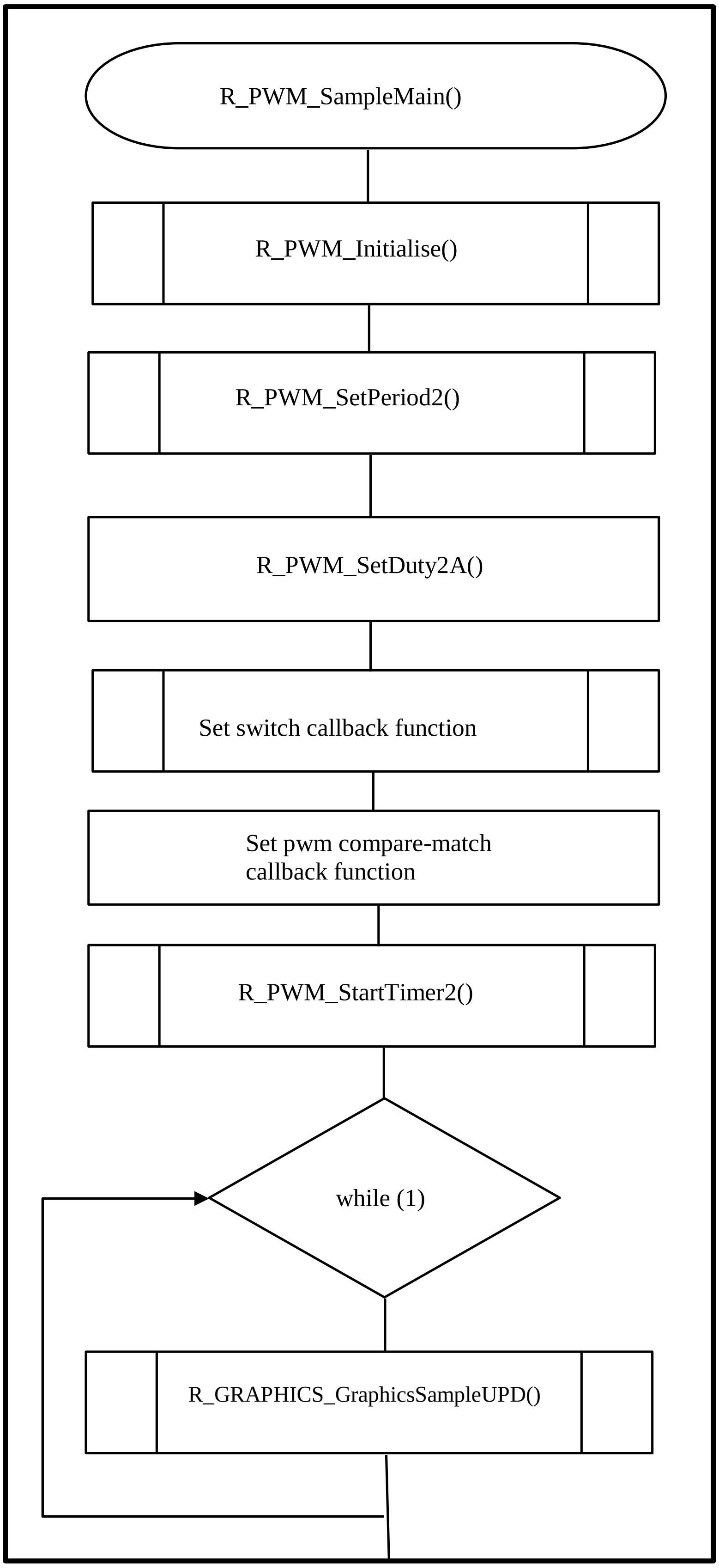

Figure 4-3 Backlight Background Program Flow

5. Reference Documents

User's Manual: Hardware

RZ/A1H Group User's Manual: Hardware Rev.1.00 R01UH0403EJ

The latest version can be downloaded from the Renesas Electronics website.

Application Notes:

RZ/A1H Group I/O definition header file <iodefine.h> R01AN1860EJ

RZ/A1H Group Example of Initialization (R01AN1864EJ)

RSK+ RZ/A1H User Manual R20UT3007EG

RSK+ RZ/A1H Tutorial Manual R20UT3008EG

The latest versions can be downloaded from the Renesas Electronics website.

Technical Update/Technical News

The latest information can be downloaded from the Renesas Electronics website.

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	16 Oct 2015		

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

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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended 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; and industrial robots etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
   Renesas Electronics products are neither intended nor 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 damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or 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. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

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

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

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

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

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: +44-1628-585-100, Fax: +44-1628-585-900

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, P. R. China 200333
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, HALII Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

© 2015 Renesas Electronics Corporation. All rights reserved.

Colophon 5.0