To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

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

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

RENESAS

H8S Family

Using the 14-Bit PWM Function

Introduction

This application note presents an example usage of the 14-bit PWM function of the H8S/2100 Series to implement a D/A converter.

Target Device

H8S/2114

H8S/2110B

H8S/2140B

H8S/2141B

H8S/2160B

H8S/2161B

H8S/2145B

H8S/2189

H8S/2168

H8S/2148

H8S/2144

H8S/2138

H8S/2134

H8S/2128

Contents

1. Specifications

- The 14-bit PWM module outputs a PWM signal from the PWXi (i = 0, 1) pin. By passing this PWM output signal through a low-pass filter (RC network) as shown in figure 1, an analog output (D/A-converted signal) is produced. In this sample task, we use the output on the PWX0 pin.
- In 10-MHz operation, the internal clock frequency is selectable from among ϕ, ϕ/2, ϕ/64, ϕ/128, ϕ/256, ϕ/1024, ϕ/4096, and ϕ/16384. In this sample task, ϕ/2 is selected. For specific values, please see table 1.
- In this sample task, we measure the analog outputs corresponding to duty cycles in the range from 0/256 to 255/256. The relationship between duty cycle and analog voltage is given in table 2.

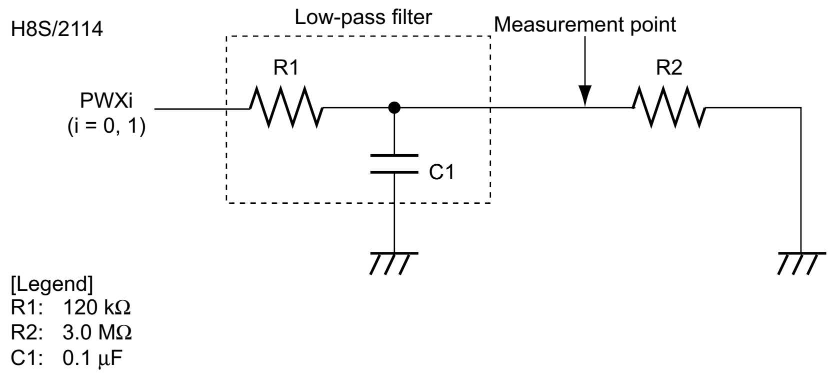


Figure 1 Example Circuit for Use as a D/A Converter

Table 1 Resolution, PWM Base Cycle, and PWM Conversion Cycle when ϕ = 10 MHz (CFS = 1 with 14-bit conversion accuracy)

Internal Clock	Resolution	PWM Base Cycle	PWM Conversion Cycle	Frequency
ϕ	0.1 μs	25.6 μs	1.64 ms	39.1 kHz
ϕ/2	0.2 μs	51.2 μs	3.28 ms	19.5 kHz
ϕ/64	6.4 μs	1.6 ms	102.4 ms	625 Hz
ϕ/128	12.8 μs	3.3 ms	211.2 ms	303 Hz
ϕ/256	25.6 μs	6.6 ms	422.4 ms	151.5 Hz
ϕ/1024	102.4 μs	26.2 ms	1.7 s	38.2 Hz
ϕ/4096	409.6 μs	104.9 ms	6.7 s	9.5 Hz
ϕ/16384	1638.4 μs	419.4 ms	26.8 s	2.4 Hz

Table 2 Relationship between Duty Cycle and Analog Output

No.	Duty Cycle	DADRA Setting	Analog Output [V]		
			Theoretical Value*1	(1)*2	(2)*2
1	0/256 (no additional pulse)	H'0003	0.00	0.01	0.00
2	1/256(no additional pulse)	H'0103	0.01	0.03	0.01
3	20/256(no additional pulse)	H'1403	0.26	0.25	0.26
4	40/256(no additional pulse)	H'2803	0.52	0.49	0.51
5	60/256(no additional pulse)	H'3C03	0.77	0.73	0.76
6	80/256(no additional pulse)	H'5003	1.03	0.99	1.04
7	100/256(no additional pulse)	H'6403	1.29	1.23	1.29
8	120/256(no additional pulse)	H'7803	1.55	1.49	1.48
9	140/256(no additional pulse)	H'8C03	1.81	1.73	1.79
10	160/256(no additional pulse)	H'A003	2.06	1.98	2.06
11	180/256(no additional pulse)	H'B403	2.32	2.23	2.31
12	200/256(no additional pulse)	H'C803	2.58	2.48	2.57
13	220/256(no additional pulse)	H'DC03	2.84	2.71	2.82
14	240/256(no additional pulse)	H'F003	3.09	2.95	3.06
15	250/256(no additional pulse)	H'FA03	3.22	3.09	3.21
16	255/256(no additional pulse)	H'FF03	3.29	3.15	3.28
17	255/256 (additional pulse: 63/63)*3	H'FFFF	—	3.16	3.29
18	PWM output fixed at high level (100% duty cycle)	—	3.30	3.17	3.29

Notes 1. The theoretical values of the analog output are calculated from the following formula:
Analog output (theoretical value) = Vcc × (DA13 to DA6)/256,
where Vcc = 3.3 V, 0 ≤ (DA13 to DA6) ≤ 255

2. Values of R1, R2, and C1 in figure 1:
(1) R1 = 120 kΩ, R2 = 3.0 MΩ, C1 = 0.1 μF
(2) R1 = 120 kΩ, R2 = no resistor connected (open), C1 = 0.1 μF
The R2 value in the figure is chosen on the assumption that the input impedance of an actually connected device is 3 MΩ.
The values shown here are reference values that suited our environment. Before actually using the circuit, evaluate the values on your system.

3. By superposing an additional pulse, the analog voltage can be set in finer steps. When the PWM function is used, the configuration of a 255/256 duty cycle with a 63/63 additional pulse leads to the maximum analog output that can be obtained through DADRA setting.

2. Conditions for Application

Table 3 Conditions for Application

Item	Description
Operating frequency	System clock (ϕ): 10 MHz
Operating mode	Mode 6 (MD2 = 1, MD1 = 1, MD0 = 0) Mode 2 (MD2 = 0, MD1 = 1, MD0 = 0)
Development tool	HEW: version 4.00.00.027
C/C++ compiler	H8S, H8/300 Series C/C++ Compiler: version 6.0.3.0 (from Renesas Technology Corp.)
Compiler options	-cpu = 2000A:24, -code = asmcode, -optimize = 1

3. Functional Description

This sample task applies the 14-bit PWM function to output pulses with a controlled duty cycle (0/256 to 255/256) from a PWM output pin. Figure 2 is a block diagram of the 14-bit PWM module, and is followed by a description of the module.

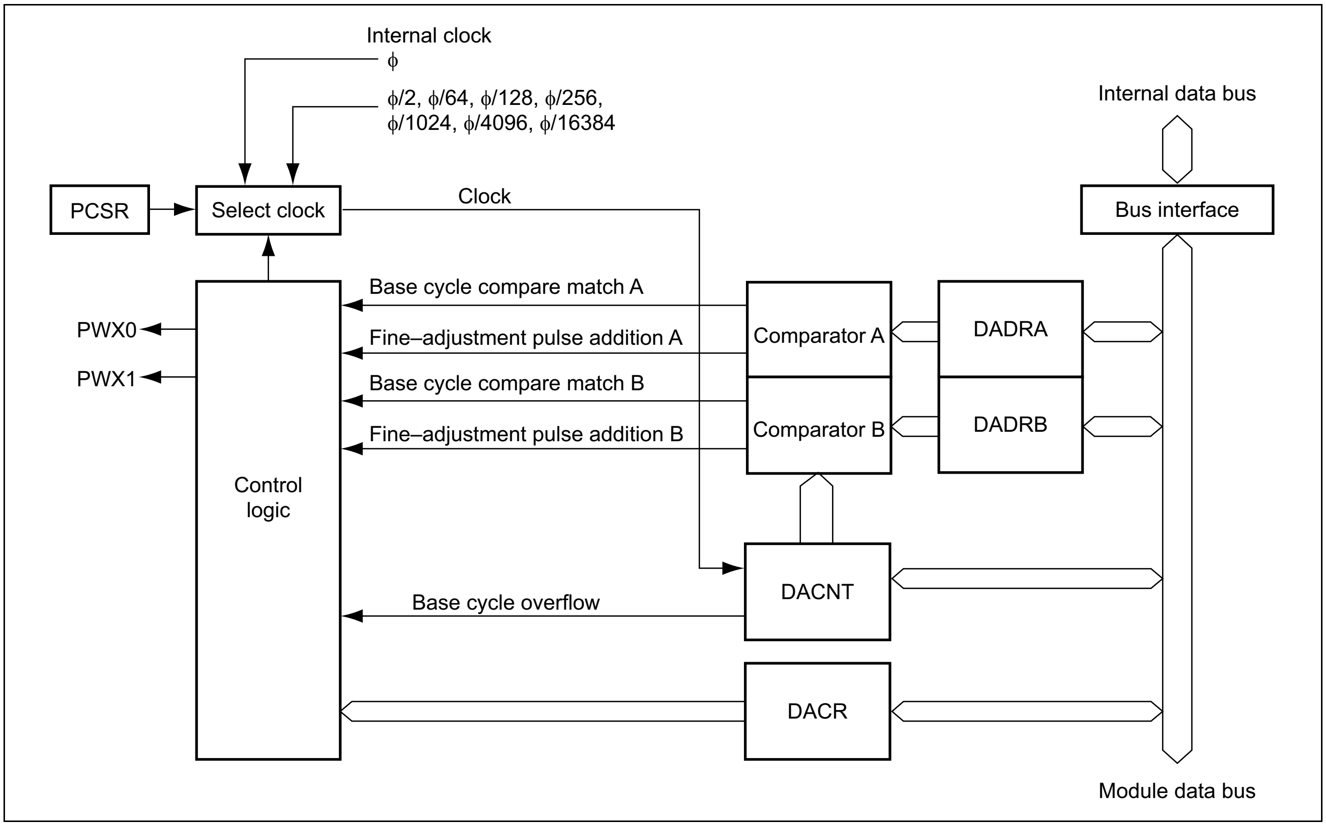


Figure 2 Block Diagram of 14-bit PWM Module

- For reduced ripple in analog output, the PWM module divides the pulse into multiple base cycles.
- Choice of eight resolutions: 1, 2, 64, 128, 256, 1024, 4096, or 16384 system clock cycles
- The base cycle can be selected as either T × 64 or T × 256, where T = resolution.
- Duty cycles ranging 0 to 100% can be set with 1/256 resolution (100% duty cycle is realizable as a port output).
- The input clock for the PWM timer is selectable from among ϕ, ϕ/2, ϕ/64, ϕ/128, ϕ/256, ϕ/1024, ϕ/4096, and ϕ/16384. The system clock (ϕ) is the reference clock used to drive the CPU and peripheral functions. The resolution, base cycle time, and conversion cycle time for PWM are calculated from the selected internal clock frequency by using the following formulae.

  Resolution (minimum pulse width) = 1/Internal clock frequency
  PWM base cycle = Resolution × 256
  PWM conversion cycle = PWM base cycle × 64

- PWMX (D/A) counter H, L (DACNTH, DACNTL)
  DACNT is a readable/writable14-bit up counter that counts the input clock pulses. The input clock is selected by the clock select bit (CKS) of the DACR register.
  DACNT provides the time base for the two-channel PWM (D/A). When operating the PWM function with 14-bit precision, all bits of DACNT are used. In operation with 12-bit precision, only the 12 lower-order bits are used; the two higher-order bits are ignored.

- PWMX (D/A) data registers A and B (DADRA, DADRB)
  DADR consists of a pair of 16-bit readable/writable registers, DADRA and DADRB, which correspond to channel A (PWM0) and channel B (PWM1), respectively, of the PWMX (D/A) module. The least significant bit of DADRA has no function, and the value read from this bit is always 1. The data for D/A conversion is set in the higher-order 14 bits of DADR. This value is continuously compared with the DACNT value to determine the duty cycle of the output waveform within each base cycle and also determines whether or not an additional pulse, which is equal in width to the resolution, is superposed. Such operation only proceeds if the DADR setting is within the range that corresponds to the current value of the carrier frequency select bit (CFS). If a value outside the range for the current CFS setting is selected, the PWM output is held constant.
  In operation with 12-bit precision, the two lower-order data bits (DA1 and DA0) are fixed to 0 and the higher-order 12 data bits are regarded as valid. The two lower-order bits correspond to the two higher-order bits of the counter (DACNT).

- PWMX (D/A) control register (DACR)
  DACR is an 8-bit readable/writable register used to enable the PWM outputs and select the output phase and resolution.

- Peripheral clock select register (PCSR)
  PCSR is an 8-bit readable/writable register that, in combination with the clock select bit (CKS) in DACR, selects the clock for the PWMX module.

Note: The register descriptions above apply to the H8S/2114 group. When using a device from another group of the H8S/2100 series, consult the corresponding datasheet.

4. Description of Operation

This section explains the operation of this sample task. Figure 3 illustrates the D/A converter operation using the 14-bit PWM function. The pulses output from the PWXi (i = 0, 1) pin are smoothed by the RC network (low-pass filter) to produce an analog output (D/A-converted output). For reference, figure 4 shows an example of a D/A-converted waveform generated by using the 14-bit PWM function.

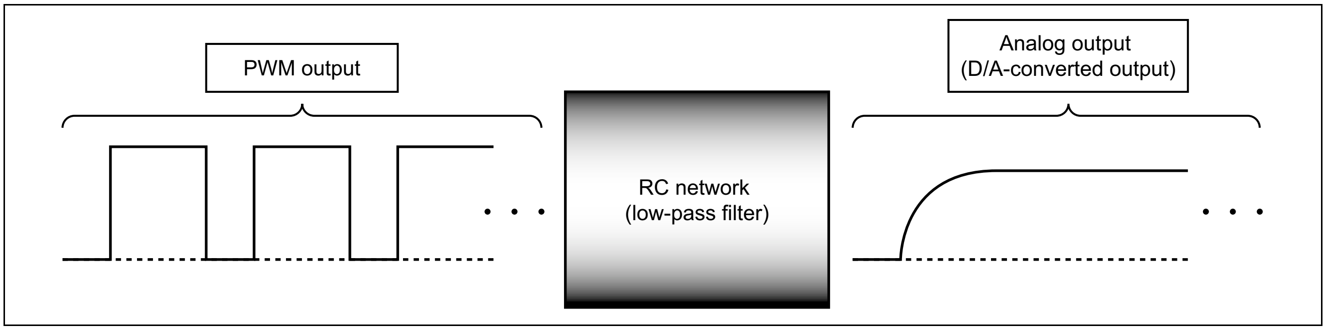


Figure 3 Operation of a D/A Converter Driven by the 14-bit PWM Function

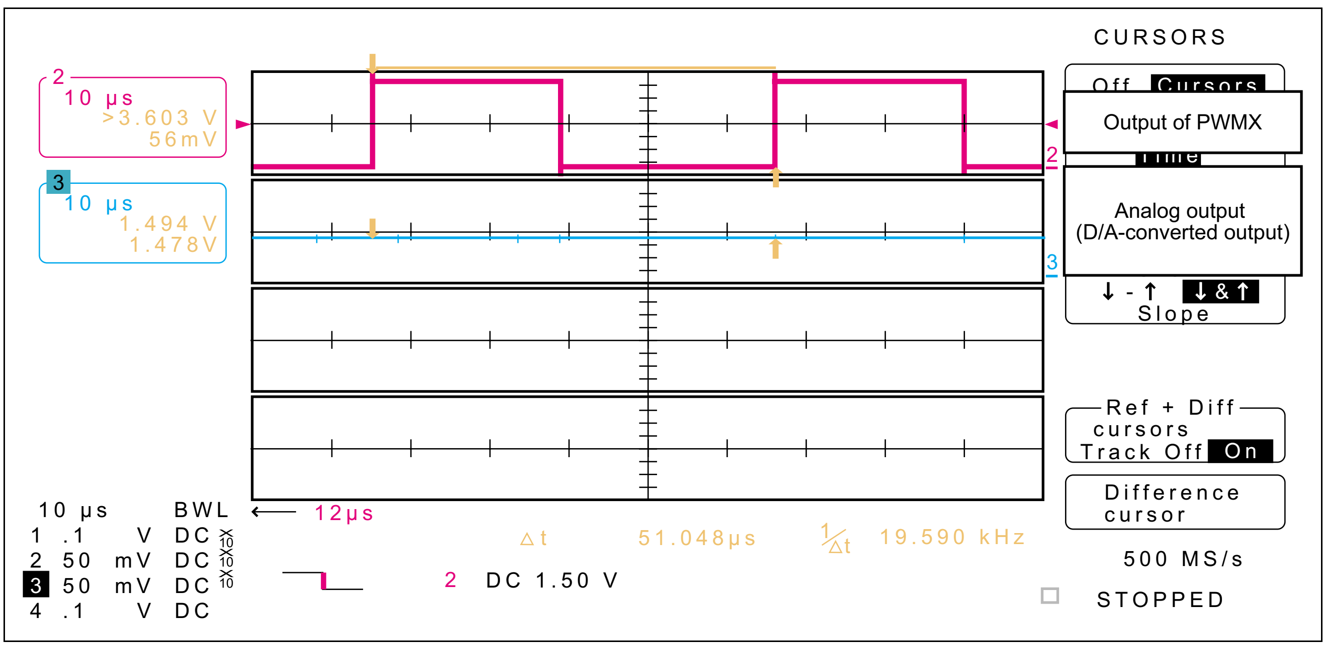


Figure 4 D/A-Converted Waveform Generated by Using the 14-bit PWM Function (for Reference)

5. Description of Software

5.1 Module

Table 4 describes the single module of this sample task.

Table 4 Description of Module

Module	Label	Function
Main Routine	main	Implements 14-bit PWM output on the PWX0 pin

5.2 Arguments

No argument is used in this sample task.

5.3 Internal Registers

Table 5 describes the internal registers used in this sample task.

Table 5 Description of Internal Registers

Register		Function	Address	Setting
DACR	OEA	PWM (D/A) Control Register (Output enable A) Enables or disables the output on channel A (PWX0) of the PWMX (D/A) module. 0: Output on channel A (PWX0) is disabled. 1: Output on channel A (PWX0) is enabled.	H'FFFFA0 Bit 2	1
	OS	PWM (D/A) Control Register (Output select) Selects the output phase of the PWMX (D/A) module. In this sample task, phase-inverted output is selected. 0: Direct PWM output 1: Inverted PWM output	H'FFFFA0 Bit 1	1
	CKS	PWM (D/A) Control Register (Clock select) In combination with the PCSR register, selects one of eight resolutions as that for the PWMX (D/A) module. In this sample task, resolution = 2 system clock cycles (t_{cyc}) is selected. 0: PWMX (D/A) operates with resolution = one system clock cycle (t_{cyc}). 1: PWM (D/A) operates with resolution = 2, 64, 128, 256, 1024, 4096, or 16384 system clock cycles (t_{cyc}).	H'FFFFA0 Bit 0	1

Register		Function	Address	Setting
DACNT	REGS	PWMX (D/A) Counter (Register select) Registers DADRA and DACR are allocated to the same address, as are registers DADRB and DACNT. The REGS bit selects which register pair is accessible. 0: Access to DADRA and DADRB is enabled. 1: Access to DACR and DACNT is enabled.	H'FFFFA6 Bit 0	0/1
DADRA	DA13 to DA6	PWMX (D/A) Data Register A These 8 higher-order bits of the data for D/A conversion set the duty cycle of the PWM output.	H'FFFFA0 Bits 7 to 0	H'00 to H'FF
	DA5 to DA0	PWMX (D/A) Data Register A These 6 lower-order bits of the data for D/A conversion select whether or not an additional pulse is superposed on the output.	H'FFFFA1 Bits 7 to 2	H'00 to H'3F
	CFS	PWMX (D/A) Data Register A (Carrier frequency select) In this sample task, base cycle = resolution (T) × 256 is selected. 0: PWM (D/A) module operates with base cycle = resolution (T) × 64. The value of DA13 to DA0 ranges from H'0100 to H'3FFF. 1: PWM (D/A) module operates with base cycle = resolution (T) × 256. The value of DA13 to DA0 ranges from H'0040 to H'3FFF.	H'FFFFA0 Bit 1	1
PCSR	PWCKXB PWCKXA PWCKXC	Peripheral Clock Select Register (PWMX clock select) These bits select the clock for PWMX when the CKS bit in DACR is 1. For details, see table 6. In this sample task, resolution = 2 system clock cycles (t_{cyc}) is selected.	H'FFFF82 Bits 5, 4, 0	0 0 0
STCR	IICE	Serial Timer Control Register (I2C master enable) When the RELOCATE bit is clear, controls access to the PWMX registers (DADRAH/DACR, DADRL, DADRBH/DACNTH, DADRBL/DACNTL) by the CPU.	H'FFFFC3 Bit 4	1
P4DDR		Port 4 Data Direction Register Here, sets the PWX0/P46 pin as an output pin.	H'FFFFB5 Bit 6	1
MSTPCRH	MSTP11	Module Stop Control Register H (MSTP11) Here, used to take the 14-bit PWM timer (PWMX) out of the module stop mode. 0: Module stop mode is cancelled. 1: Module stop mode is set.	H'FFFF86 Bit 3	0

Table 6 Internal Clock Selection

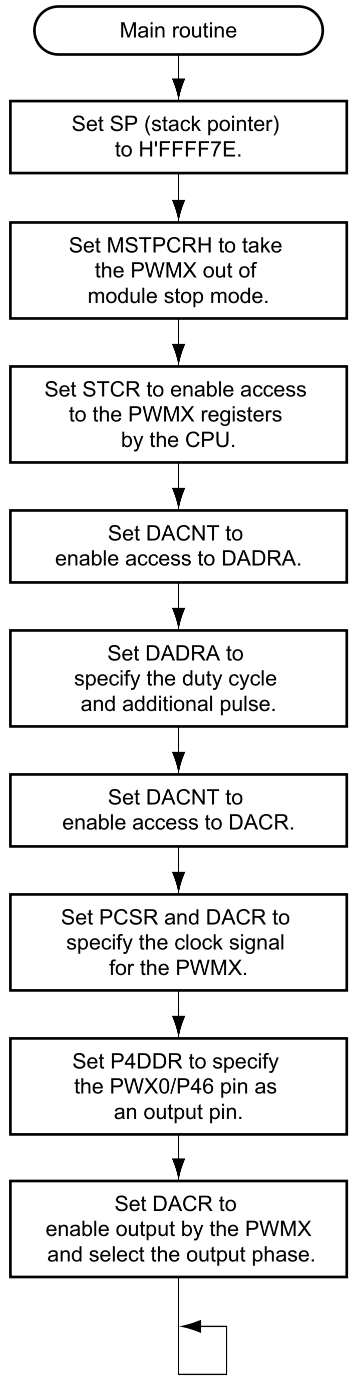
PWCKXC	PWCKXB	PWCKXA	Resolution (T)
0	0	0	Operates on system clock cycle (t_{cyc}) × 2
0	0	1	Operates on system clock cycle (t_{cyc}) × 64
0	1	0	Operates on system clock cycle (t_{cyc}) × 128
0	1	1	Operates on system clock cycle (t_{cyc}) × 256
1	0	0	Operates on system clock cycle (t_{cyc}) × 1024
1	0	1	Operates on system clock cycle (t_{cyc}) × 4096
1	1	0	Operates on system clock cycle (t_{cyc}) × 16384
1	1	1	Setting prohibited

5.4 RAM Usage

RAM is not used in this sample task.

6. Flowchart

6.1 Main routine



7. Program Listing

```
/*************************************************************************************/
/*                                                                                   */
/*  This program is 14bit PWM output program for H8S/2114 evaluation.                */
/*                                                                                   */
/*                                                           File name : pwm14.c     */
/*                                                           Frequency : 10MHz       */
/*                                                           CPU TYPE  : H8S/2114    */
/*************************************************************************************/

/*************************************************************************************/
*  Include                                                                           */
*************************************************************************************/

#include <stdio.h>                        /* Input/Output library file               */
#include <machine.h>                      /* Built-in function file                  */
#include "2114.h"                         /* H8S/2114 I/O register definition file   */

/*************************************************************************************/
*  Prototype                                                                         */
*************************************************************************************/

void main(void);                          /* Main routine                            */

/*************************************************************************************/
*  RAM allocation                                                                    */
*************************************************************************************/


/*************************************************************************************/
*  main : Main routine                                                               */
*************************************************************************************/

void main(void)

#pragma section

#pragma asm

    mov.l    #H'FFFF7E,sp                 ; Initialize stack pointer

#pragma endasm

{

/* Module stop mode reset */

    MSTPCR.BYTE.H = 0x37;                 /* Reset PWM module stop mode              */

/* Enable PWMX register access */

    STCR.BYTE = 0x10;
```

```
/* Enable DADRA,DADRB access */

PWMX.REGS1.ST_DACNT.BIT.REGS = 0;

/* Duty & Add pulse set */

PWMX.REGS0.ST_DADRA.WORD = 0x7803;                /* Duty cycle = 120/256, Add pulse = 0/63  */


/* Enable DACR,DACNT access */

    PWMX.REGS1.ST_DACNT.BIT.REGS = 1;

/* PWX0 clock select */

PWMX.REGS1.ST_DACR.BIT.CKS = 1;
PWM.PCSR.BYTE = 0x00;                             /* Resolution = tcyc x 16384               */

/* PWX0 output select */

PWMX.REGS1.ST_DACR.BIT.OS = 1;                    /* Select inverted output                  */

/* PWM output port set */

    P4.DDR = 0x40;                                /* Set PWX0(P46)                           */

/* PWX0 output enable */

    PWMX.REGS1.ST_DACR.BIT.OEA = 1;

    while(1);                                     /* End                                     */

}
```

Revision Record

Rev.	Date	Description	
		Page	Summary
1.00	Jul.22.05	—	First edition issued