RH850/U2A-EVA

PWM Output/Diagnostic

Introduction

This document describes an implementation of the PWM signal generation with automatic diagnosis on the RH850/U2A-EVA microcontrollers.

It should be used in conjunction with the corresponding user's manual of RH850/U2A-EVA series.

Target Device

This application note is intended to describe the PWM output/diagnostic application on RH850/U2A-EVA series.

In this document, the RH850/U2A-EVA device R7F702Z19AEDBG is employed to implement the example application. Still, the concept described in this document applies also to other members of the RH850/U2A-EVA series.

The RH850/U2A-EVA series has following variants:

RH850/U2A-EVA	FBGA-516	R7F702Z19AEDBG
	FBGA-516	R7F702Z19BFDBG
RH850/U2A16	FBGA-516	R7F702300EBBG
	FBGA-373	R7F702300EBBB
	FBGA-292	R7F702300EABA
	FBGA-516	R7F702300AEBBG-C
	FBGA-373	R7F702300AEBBB-C
	FBGA-292	R7F702300AFABA-C
RH850/U2A8	FBGA-373	R7F702301EBBA
	FBGA-292	R7F702301EABG
	FBGA-373	R7F702301AEBBA-C
	FBGA-292	R7F702301AFABG-C
RH850/U2A6	FBGA-292	R7F702302FABB-C
	HLQFP-176	R7F702302FAFK-C
	FBGA-156	R7F702302FABD-C
	HLQFP-144	R7F702302FABD-C

Disclaimer

Contents

Contents of Figures

1. Introduction

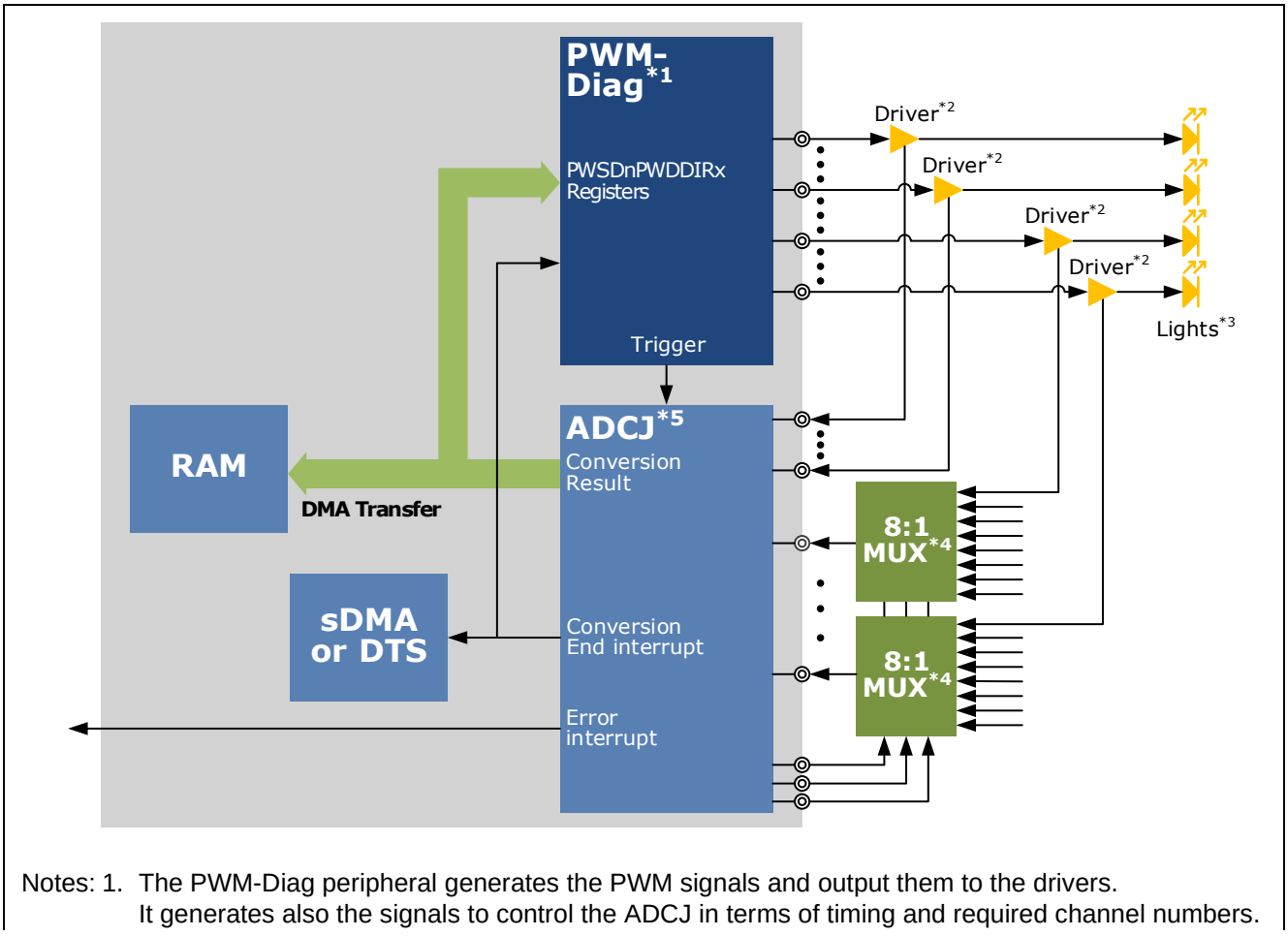
The primary targets for the PWM-Output/Diagnostic module are lighting applications that require a large quantity of PWM signals with the following requirements:

- Typically, two different kinds of light sources are used: Traditional bulbs and LEDs.
  As the brightness must be adjusted according to the application needs, both light sources are controlled with the same type of signal: A PWM signal. But due to their different characteristic in terms of turn-on/-off time, different period frequencies are applied for a flicker free appearance. For bulbs PWM signals of about 100Hz[*1] are used, whereas LEDs are controlled with PWM signals of 200-400Hz[*1].

Note 1: In order to avoid any visual interference with stationary lights operating at 50Hz/60Hz, multiples of these frequencies are not used. Still, for a simple and demonstrative calculation, a frequency f1 = 100Hz will be used for bulbs lighting sources and a frequency f2 = 200Hz will be used for LED lighting sources in this document.

- In addition to the generation of the PWM signals, also the error situation (e.g. in case of a broken light source) must be detected with minimum efforts of the CPU.
  The detection of the (wrong) operating state is done by measuring the operating current of the light source and comparing it to a known good reference value. The measurement must take place within the on- (high-) period of the PWM signal. In case the resulting measurement value mismatches a reference value an alert (interrupt) is issued to the application.
- The complete process of the PWM signal generation and diagnostic shall take place with minimum interaction of the CPU.

A block diagram of the described application is shown in Figure 1.1.



Notes: 1. The PWM-Diag peripheral generates the PWM signals and output them to the drivers.
It generates also the signals to control the ADCJ in terms of timing and required channel numbers.

2. The drivers amplify the PWM signals from the device to the light sources. An additional feedback output is provided to measure the current flowing through each driver.
3. These are light sources of the application. Typically LED's and bulbs are used.
4. To provide the required number of analog input, ADCJ supports external analog multiplexers with up to eight analog inputs to one analog output. Selection of the analog input is controlled by the ADCJ automatically.
5. The ADCJ converts the analog values into digital values.
   The conversion end interrupt is output to PWM-Diag to allow the next trigger, this interrupt can also be used as a DMA transfer (via sDMA or DTS) trigger for the result transfer from ADCJ to RAM.
   The conversion results are transferred to the related PWSDnPWDDIRx registers on the same time.
   In case the converted value does not match an expected result, an error interrupt is issued to the CPU.

Figure 1.1 Example Connection

The entire PWM generation and diagnostics function operates with the following modules of the device:

- PWM output/diagnostic (PWM-Diag)
- A/D converter (ADCJ)
- Optionally the sDMA or DTS can be applied for result handling.

The PWM-Diag peripheral is comprised of the following units:

- PWBA: PWM period timing generation
- PWGC: PWM signal generation generator
- PWSD: PWM ADCJ control functions

The detailed function and operation of each unit is explained in the next sections.

2. Reference Documents

This chapter contains information about the device reference documentation.

2.1 User's Manual

The Hardware User's Manual provides information about the functional and electrical behavior of the device.

At the release time of this document the following manual versions are available:

- RH850/U2A-EVA User's Manual (Rev. 1.10): R01UH0864EJ0110

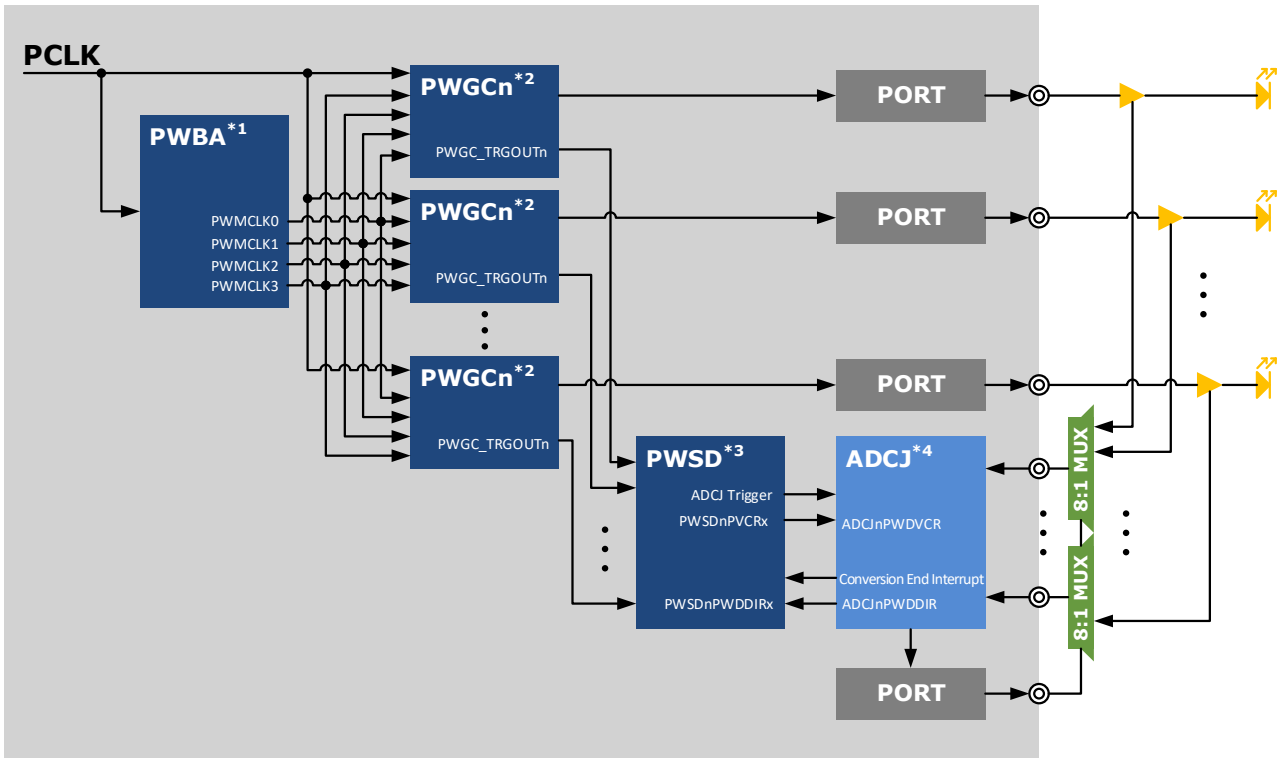
3. PWM Output/Diagnostic Function

3.1 Overview

The PWM Output/Diagnostic function is implemented with three related units:

- PWBA: PWM period timing generation
- PWGC: PWM signal generation generator
- PWSD: PWM ADCJ control functions

Figure 3.1 illustrates the relationship among the mentioned functions for the general PWM diagnosis operation:



Notes: 1. PWBA generates up to 4 PWM period clocks out of the supplied peripheral clock.

2. PWGC selects one of the 4 period clocks and generates a PWM signal with selectable counter period, duty cycle, period delay and ADCJ trigger. Additionally an ADCJ conversion trigger can be set within the PWM period.
   Each of the available PWM channels can be configured via a dedicated PWGC unit.
   Up to 96 PWM signals are available (depending on device implementation).
3. PWSD holds the information about the ADCJ channel to be used for diagnosis of each PWM signal. This includes information about the selected ADCJ unit, the physical channel number, the upper/lower limit usage and multiplexer information.
   Upon reception of am ADCJ conversion trigger form PWGC, the ADCJ information related to that channel is transferred to the ADCJ and an ADCJ conversion is triggered.
   The end of the conversion is signaled from ADCJ to PWSD unit. The conversion results are automatically transferred to the dedicated result registers for each channel in PWSD.
   The PWSD unit holds a FIFO queue with 8 entries for buffering of ADCJ triggers in case the ADCJ is busy.
4. ADCJ performs A/D conversion of the lamp driver, measures the analogue signal which is proportional to the lamp current. The start trigger of ADCJ is generated by PWSD unit.
   On completion of the A/D conversion, ADCJ provides PWSD a conversion end signal.

Figure 3.1 Block diagram of PWM Output/Diagnostics with Related Functions

Throughout this document, the following indices are employed for the abbreviation of the peripheral units and registers.

- Index “n” identifies the individual units of PWM-Diag functions or ADCJ function.
  For example, the PWBAnTE register, the ADCJnPWDVCR register.
- Index “m” identifies the PWBA cycle configuration and PWGC period setting registers; m = 0 to 3.
  For example, the PWBAnBRSm registers.
- Index “x” identifies the individual PWM channels; x = 00 to 95.
  For example, the PWSDnPVCRx registers and PWSDnPWDDIR registers.
- Index “j” identifies the registers storing trigger channel numbers; j = 0 to 7.
  For example, the PWSDnQUEj register.
- Index “k”, “h” and “q” identifies the registers with the same function to set amount of different PWM channels; k = 0 to 2, h = 0 to 3, q = 0 to 5.
  For example, the PWGC synchronous trigger registers SLPWGCk register, the PWGCINTFhk register and the PWGCPRDSLq register.

Table 3-1 lists the number of units on PWM-Diag and physical channels on ADCJ functions for the RH850/U2A-EVA products.

For further information, please refer to the device User’s Manual.

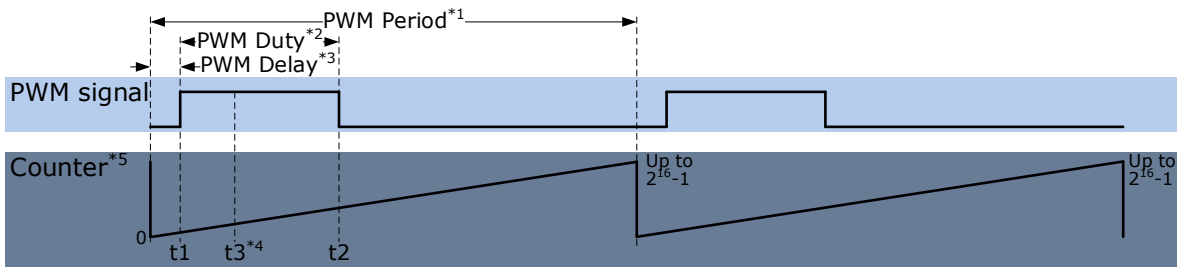
Table 3-1 Number of units on PWM-Diag and physical channels[*1] on ADCJ for RH850/U2A-EVA devices

Product name		**Units of PWM Output/Diagnostic**			**Channels of A/D Converter**		
		PWBAn	**PWGCn**	**PWSDn**	**ADCJ0**	**ADCJ1**	**ADCJ2**
RH850/U2A-EVA	516-PIN	1	96	1	30	34	30
RH850/U2A16	516-PIN	1	96	1	30	34	30
	373-PIN	1	96	1	30	34	30
	292-PIN	1	96	1	30	34	15
RH850/U2A8	373-PIN	1	96	1	30	34	30
	292-PIN	1	96	1	30	34	15
RH850/U2A6	292-PIN	1	96	1	30	34	15
	176-PIN	1	76	1	24	28	12
	156-PIN	1	50	1	14	15	0
	144-PIN	1	64	1	13	20	6

Notes: 1. The physical channels includes high accuracy inputs with analog ports and low accuracy inputs with digital/analog ports. For detailed information, please refer to the attachment “PWM_Diagnostic External Input_Output and Pin Assignment.xlsx”.

3.2 Generation of PWM Signals

As is shown in Figure 3.2, the PWM signals are related to 3 important parameters: PWM period, PWM duty and PWM delay.



Notes: 1. The PWM Period is the repeating interval of the PWM signals.
2. The PWM Duty is the active portion of a signal. For the PWM generation described in this document, the active portion is the high (‘1’) level of the signal.
In Figure 3.2 the active portion of the PWM signal is starting at t1 and ending at t2.
3. The PWM Delay is the interval between the rising edges of a signal relative to the beginning of the reference period.
A PWM delay is used to avoid the simultaneous rising edge of multiple PWM signals at the same time.
In Figure 3.2 the delay is the time between the start of PWM period (counter equals 0) and t1.
4. t3 is the trigger signal to start the A/D conversion
For detailed information, please refer to section 3.3.
5. Counter is selectable up to 16 bits.
For detailed information, please refer to section 3.2.

Figure 3.2 Concepts of PWM Signal Generation

The PWM signals are generated in 2 steps:

1. The frame clock of the PWM signal is generated
For details please refer to the section 3.2.1 below.
2. The period, delay, duty and trigger of PWM signal are generated.
For details please refer to the section 3.2.2 below.

3.2.1 Generation of PWM Frame Clock

(1) Clock Supply of PWM-Diag

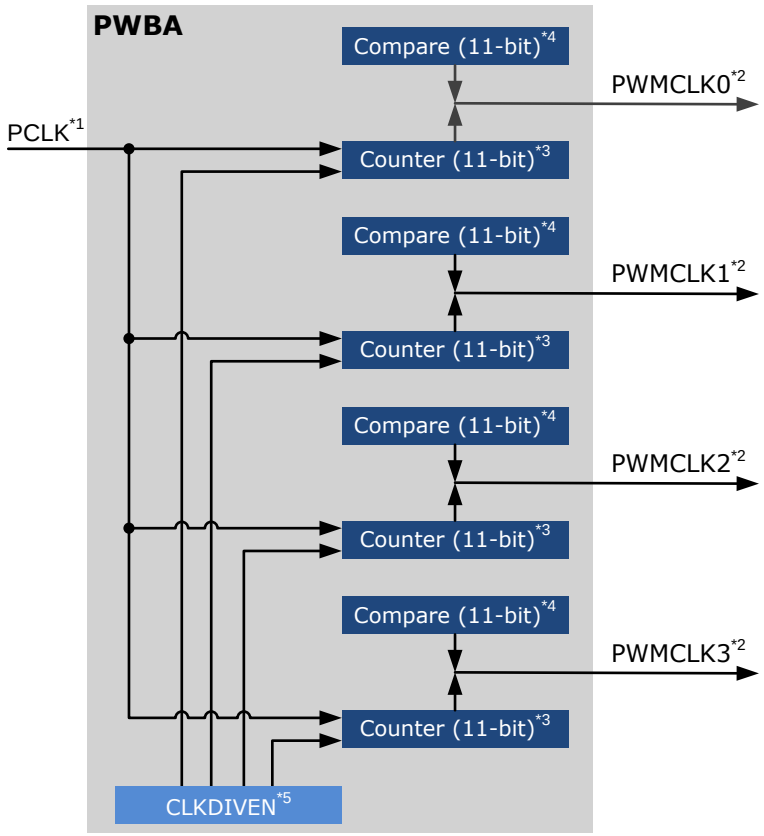
Clock supply for PWM-Diag module is CLK_LSB.

For detailed information, please refer to device User’s Manual, section 13 ‘Clock Controller’

(2) Generation of PWM Frame Clock

PWM frame clock is generated by the PWBA unit.

Figure 3.3 shows the block diagram for PWM frame clock generation.



Notes: 1. PCLK is the input clock of PWBA. Throughout this application note this input clock is 40 MHz.
2. PWMCLK0 to 3 are the generated frame clocks.
Up to four frame clocks can be generated.
3. Each 11-bit counter is the counter for generation of the frame clocks.
4. Each 11-bit compare register is employed to hold the required compare value for generation of the frame clocks out of the PCLK. The value can be set with a resolution of 11 bits. The 4 Compare Registers are referred to as PWBAnBRSm, where m = 0 to 3.
5. CLKDIVEN is a control block to enable the counters individually.

Figure 3.3 Block Diagram for PWM Frame Clock Generation

The peripheral clock signal PCLK supplies the common clock for all counter registers. The counter register is compared with the value in the related compare register on every PCLK cycle. In case of a compare match, the PWM period clock signal PWMCLK toggles and the related counter is cleared to 0.

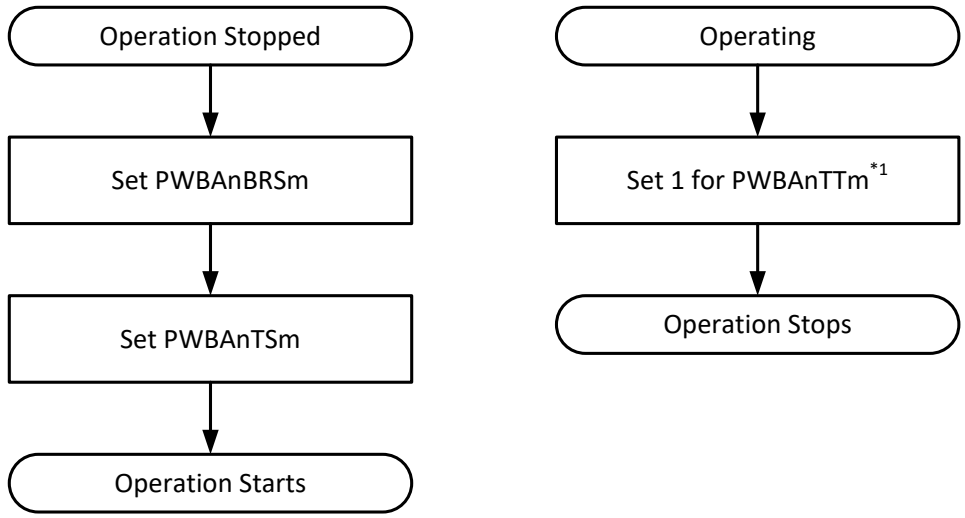
The PWM frame clocks are the outputs of the PWBA unit, each frequency of the clocks is related to the compare register PWBAnBRSm, for the values except 0:

$$f_{PWMCLKm} = f_{PCLK} / (\, set\ value\ of\ PWBAnBRSm \times 2) \quad (1).$$

For set value 0 of PWBAnBRSm, the frequency of the related PWMCLKm is set as the PCLK of 40 MHz.

The four PWM clocks can be enabled by setting the related bit of PWBAnTS to 1, and disabled by setting the related PWBAnTT bit to 1.

Figure 3.4 shows the procedures for setting when starting and stopping operation.

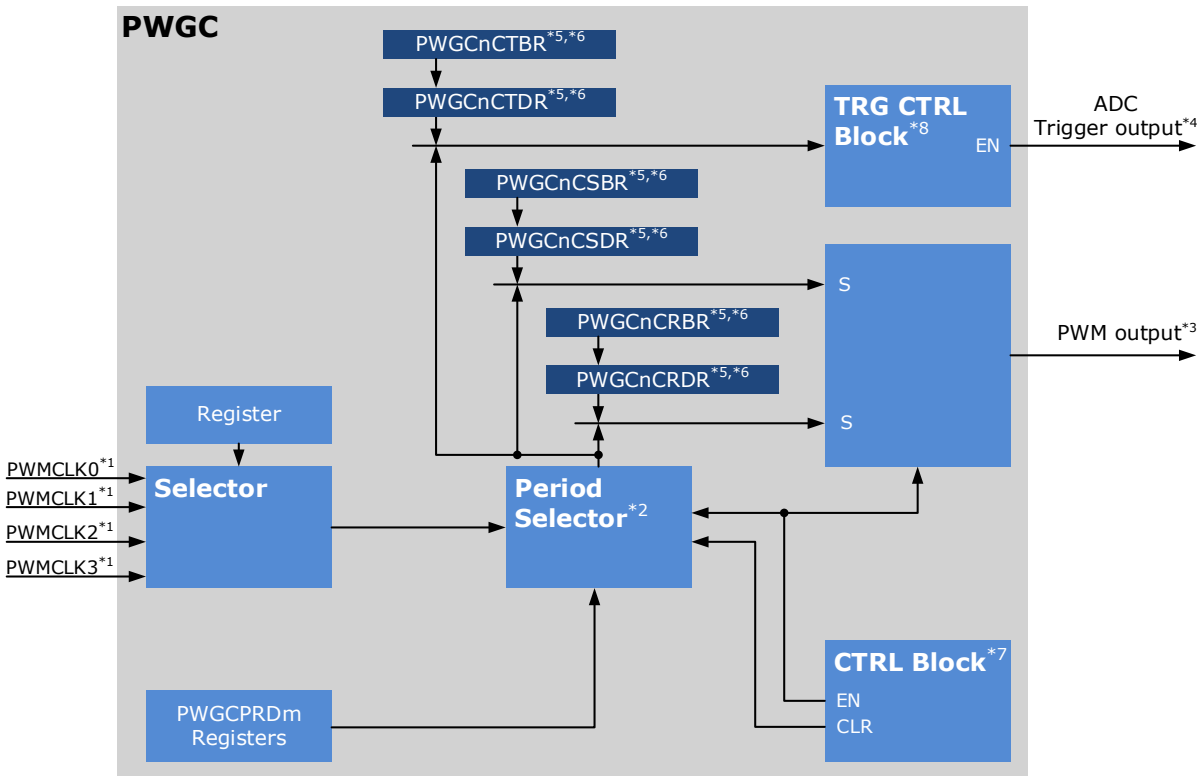


Notes: 1. Setting the PWBAnTTm bit **immediately** stops the related PWM frame clock. The stop is not synchronized to the current states of the PWM signals.

Figure 3.4 Starting and Stopping Operation of PWBA

3.2.2 Generation of PWM Waveform and Trigger

The PWM signal is generated by the PWGC units. For each PWM signal an individual PWGC unit is available. Figure 3.5 shows the Block diagram for PWM signal generation.



Notes: 1. PWMCLK0 to 3 are PWM period clock inputs generated by the PWBA unit. One of the four clocks is chosen by the Selector.

2. The period selector specifies the PWGC_PERIOD for each PWGC channel based on the PWGCPRDm registers.
The selected counter is clocked by the selected PWM period clock. After related ticks of the clock

the counter is cleared and starts from 0. The transient count cycle can be read in PWGCnCNT register.
3. The PWM signal is defined by its rising and falling edge (see note 5 and 6).
4. ADCJ Trigger Output is a signal of timing information for A/D conversion.
   The trigger signal is output to PWSD unit for A/D converter control. For details please refer to section 3.3.1.
5. C**S**xR registers hold the timing information for the rising edge of the PWM signal.
   C**R**xR registers hold the timing information for the falling edge of the PWM signal.
   C**T**xR registers hold the timing information for the ADCJ trigger.
6. The Cx**B**R registers are the **b**uffer registers that are accessible by the CPU.
   The Cx**D**R registers are the **d**ata registers for internal operation of the unit.
   The setting value of Cx**D**R registers is reflected to the related Cx**B**R registers at the start of PWGC operation or after the execution of simultaneous rewrite.
7. The Ctrl Block includes a function to start and stop PWGC.
8. An ADCJ trigger output can be spacified, en- and disabled by the Trg Ctrl Block.

Figure 3.5 Block Diagram for PWM Signal Generation

(1) Generation of PWM Period

As is illustrated in Figure 3.5, the clock signal flows through a period selector in PWGC, the PWGC_PERIOD selection can be configured in PWGC period selection registers PWGCPRDSLq.

To select its counter, each PWGC channel has 2 related bits in the PWGCPRDSLq registers. The default value 00 means the counter configured by PWGCPRD0 is selected for the PWGC_PERIOD for the related PWGC channel.

The PWM period frequencies are calculated using the formula below:

$$f_{PWM} = f_{PWMCLKm} / (PWGC\_PERIOD + 1) \quad (2).$$

Here are some examples of resulting PWM period frequencies with different values of the clock divider or counter settings in PWBA and PWGC:

Table 3-2 Examples of PWM Period Frequency Settings

PWBAnBRSm	PWGCPRD0	PWGCPRD1	PWGCPRD2	PWGCPRD3	Channel-Related Bits in PWGCPRDSLq	PWGC_PERIOD	PWM Period Frequency
0000_H	$0FFF_H$	0001_H	$000F_H$	$FFFF_H$	00_H	$0FFF_H$	9764Hz
					01_H	0001_H	20MHz
					11_H	$FFFF_H$	610Hz
$07FF_H$					00_H	$0FFF_H$	2Hz
					01_H	0001_H	4885Hz
					10_H	$000F_H$	610Hz
					11_H	$FFFF_H$	0.15Hz

The maximal value of PWBAnBRSm is 2047 (11-bit resolution), while the minimal value is 0. The minimum PWM period frequency is 0.15 Hz; the maximum period frequency is 20MHz.

To drive typical 100Hz bulbs or 200Hz LEDs, an option is to select $0FFF_H$ as the PWGC_PERIOD value and

set the register PWBAnBRSm to 30_H or 18_H.

(2) Generation of PWM Duty, Delay and Trigger

PWM Delay:

PWM delay is the time between a start of the PWM period and the next rising edge of the PWM signal.

A PWM period starts at the counter value of 000_H, the rising edge of the PWM signal is at the match of the counter value with the CSDR register value. The delay can be calculated according to the following formula:

$$PWM\ Delay\ [\%] = (PWGCnCSDR + 1) \times 100 / (PWGC\_PERIOD + 1) \quad (3).$$

The PWGCnCSDR register value can be calculated like this:

$$PWGCnCSDR = [PWM\ Delay\ [\%] \times (PWGC\_PERIOD + 1) / 100] - 1 \quad (4).$$

PWM Duty:

The PWM duty is the time between a rising edge of the PWM signal (value of the CSDR register) and the next falling edge (value of the CRDR register). The duty can be calculated according to this formula:

$$PWM\ Duty\ [\%] = (PWGCnCRDR - PWGCnCSDR + 1) \times 100 / (PWGC\_PERIOD + 1) \quad (5).$$

The falling edge register value can be calculated like this:

$$PWGCnCRDR = PWGCnCSDR + [PWM\ Duty\ [\%] \times (PWGC\_PERIOD + 1) / 100] - 1 \quad (6).$$

Example 1:

Let's assume the following PWM signal:

PWM Duty = 25%

PWM Delay = 5%

PWM Trigger = 60%

PWGC_PERIOD = $7FF_H$

According to the formula (4), the value of register PWGCnCSDR is:

$$\left(\frac{5\ [\%]}{100} \times 2048\right) - 1 = 065_H$$

According to the formula (6) the value of register PWGCnCRDR is:

$$065_H + \frac{25\ [\%] \times 2048}{100} - 1 = 264_H$$

If the relationship between set values in one interval is PWGCnCSDR > PWGCnCRDR, falling edge in that interval is meaningless, and the falling edge in the next interval is valid.

In this case, the PWM duty can be calculated according to the following formula:

$$PWM\ Duty\ [\%] = (PWGC\_PERIOD - PWGCnCRDR + PWGCnCSDR + 1) \times 100 / (PWGC\_PERIOD + 1) \quad (7).$$

For two special conditions:

- The PWM 0% output is generated when the register value of the PWM rising edge matches the register value of falling edge:
  PWGCnCSDR (PWGCnCSBR) = PWGCnCRDR (PWGCnCRBR).

The 0% PWM output can also be configured by selecting the output level for forcible low level, using the PWGCnFOT register and the bit PWGCnFOS of the PWGCnCTL register.

If the interrupt is enabled, the interrupt request signal PWGC_INTn is generated upon a match of the PWGCnCNT and PWGCnCRDR.

- The PWM 100% output is generated if the bit 16 of PWGCnCRDR is set to 1:
  PWGCnCRDR (PWGCnCRBR) = 1XXXX$_H$.

  The 100% PWM can also be configured by selecting the output level for forcible high level, using the PWGCnFOT register and the bit PWGCnFOS of the PWGCnCTL register.

  If the interrupt is enabled, the PWGC_INTn signal is generated when PWGCnCNT is cleared.

A PWGC_INTn interrupt request signal is generated at the falling edge of the PWM signal.
Each 32 interrupt sources are assigned in a group to the same interrupt channel, up to 3 interrupt channels are assigned for up to 96 channels.

PWGC interrupt factor register PWGCINTFhk can be read to check the exact interrupt sources.

For detailed information, please refer to the device User's Manual, section 6.2.3.3 'EI Level Maskable Interrupts'

ADCJ Trigger:
If a diagnostics function for the feedback signal is expected for a PWGC channel, the related trigger signal for A/D conversion can be configured using following registers:

- PWGC_**T**RGOUTn **C**ontrol register PWGCnTCR
- PWGC **C**ontrol register PWGCnCTL
- PWGC_**T**RGOUTn Generation Condition register PWGCnCTDR

The generation of a trigger for ADCJ is required in case a diagnostic function shall be applied to the PWM signal. To enable this trigger to ADCJ, the PWGCnTCR.PWGCnTOE bit must be set to '1'.
Still, in some cases a PWM signal shall be generated without the diagnostic function. In this case the generation of the ADCJ trigger can be disabled by clearing PWGCnTCR.PWGCnTOE to '0'.

The trigger signals PWGC_TRGOUTn for ADCJ are generated in the PWGC unit, when PWGCnCNT matches PWGCnCTBR register.

The trigger signal can be calculated like in Formula (7).

$$PWM\ Trigger\ [\%] = (PWGAnCTDR - PWGAnCSDR + 1) \times 100/(PWGAnCRDR - PWGAnCSDR + 1) \quad (8).$$

The setting value of register PWGCnCTDR is:

$$PWGAnCTDR = PWGAnCSDR + [PWM\ Trigger\ [\%] \times (PWGAnCRDR - PWGAnCSDR + 1) / 100] - 1 \quad (9).$$

Example 2:

Let's assume the following PWM signal:

PWM Duty = 25%

PWM Delay = 5%

PWM Trigger = 60%

PWGC_PERIOD = 7FF$_H$

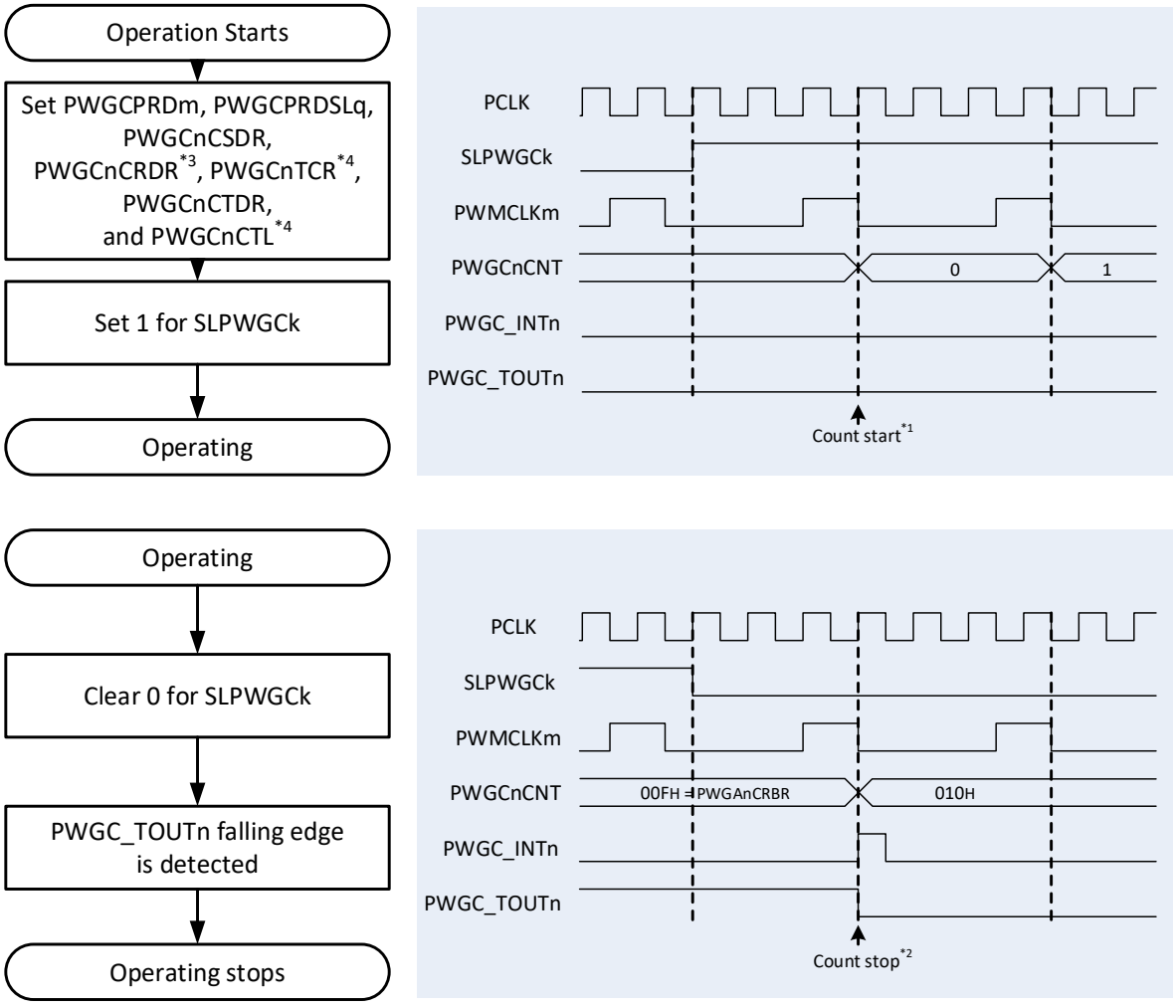
According to Example 1, the value of register PWGCnCSDR is 065$_H$, the value of register PWGCnCRDR is 264$_H$.

According to the formula (8) the value of register PWGCnCTDR is:

$$065_H + \frac{60\,[\%] \times (264_H - 065_H + 1)}{100} - 1 = 197_H$$

The PWGC unit starts and stops the PWM signals simultaneously in accordance with the reference bit of the SLPWGCk Registers.

The procedures for starting and stopping the PWM signal are illustrated in Figure 3.6.



Notes: 1. After the SLPWGCk is set to 1, PWGCnCNT starts the count operation at the next falling edge of the corresponding PWMCLK.
2. After the SLPWGCk is cleared to 0, PWGCnCNT stops operation when the next PWGC_INTn interrupt request is generated.
3. If an update of the PWM toggle during the operation cycle is necessary, simultaneous rewrite is executed by resetting the PWGCnCSDR and PWGCnCRDR registers, and setting PWGCnRDT register to 1. The new value can be written into the CSBR registers by the CPU after the next falling edge of the PWM signal.
4. These registers include the trigger control settings.

Figure 3.6 Starting and Stopping Operation of PWM Waveform

(3) Simultaneous Rewrite Procedure

For variable phase change during PWM output, PWGC also provides a possibility for simultaneous rewrite option.

Simultaneous rewrite is executed by re-setting the data registers, then setting either the PWGCnRDT or SLPWGCk register.

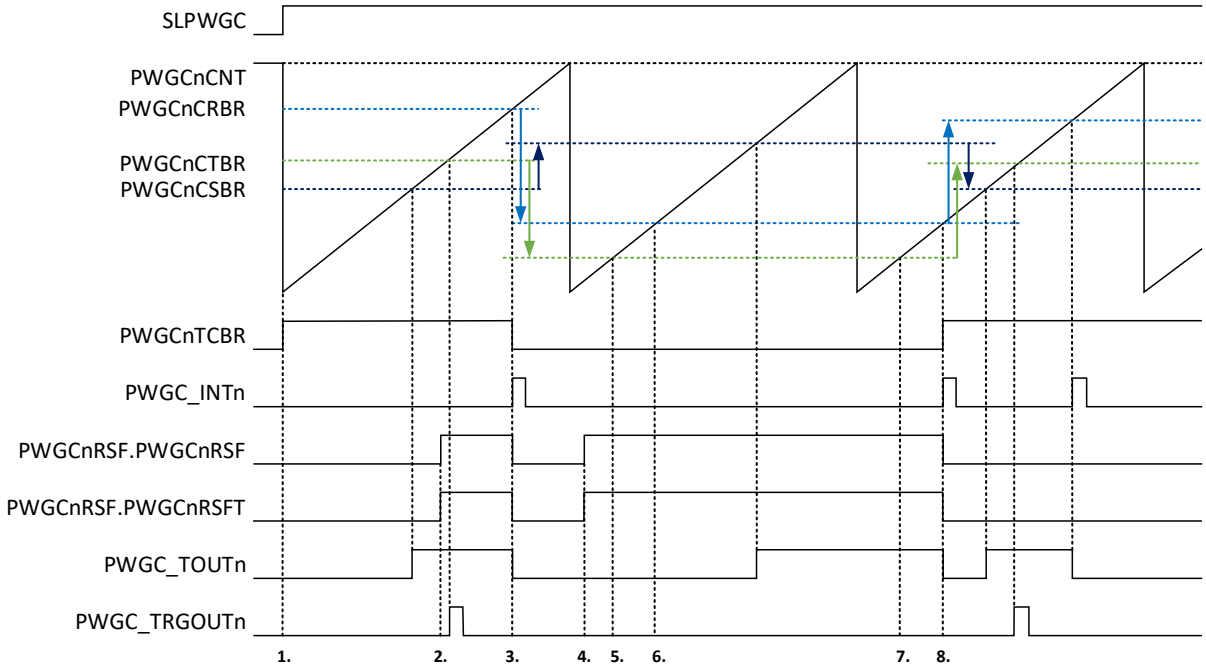
If a rewrite is triggered, the values in data registers are reflected simultaneously to buffer registers at the next valid falling edge of the PWM output.

In the RH850/U2A-EVA devices also the contents of the PWGCnTCR register is simultaneously reflected to the PWGCnTCBR register.

The PWGCnCTL.PWGCnTCUT bit specifies the timing of this simultaneous rewrite to the PWGCnTCBR register.

Figure 3.7 shows an example of simultaneous rewrite.

For detailed information, please refer to device User's Manual, section 43.3.16 'PWGCnRDT – Buffer Register Reload Trigger Register'.



Notes: 1. PWGC output simultaneous start, buffer registers are updated to the current value in data registers.
2. PWGCnRDT.PWGCnRDT bit is set to 1, simultaneous rewrite request for buffer registers is triggered. The data registers PWGCnCxDR are reset before this trigger.
PWGCnRDT.PWGCnRDTT bit is set to 1, simultaneous rewrite request for PWGCnTCBR is triggered. The register PWGCnTCR is reset to 0 before this trigger.
3. Buffer registers PWGCnCSBR and PWGCnCRBR are updated on the next falling edge of the current signal.
PWGCnCTL.PWGCnTCUT[1:0] is set to 10, therefore PWGCnTCBR is updated on the falling edge.
The simultaneous rewrite processes are finished.
4. Please refer to Note 2.
5. TRGOUT does not occur if PWGCnCTL.PWGCnOCL bit remains 0.
6. PWGC_INTn does not occur in this interval.
7. According to Note 2., PWGCnTCBR register is updated to 0 at the latest falling edge, the PWGC_TRGOUTn is disabled.
8. Please refer to Note 3.

Figure 3.7 Waveform of Simultaneous Rewrite (PWGCnCTL.PWGCnTCUT[1:0] = 10)

3.2.3 Port Configuration for PWM Output

To output the PWM waveform from PWGC, the corresponding pins must operate in software I/O control alternative mode, which is enabled by setting the following 2 registers:

- Port Mode Control register PMCn:
  The register specifies the operation mode of the corresponding pin.
  - For the related bit PMCn.PMCn_m = 0, the pin is switched to port mode;
  - For the related bit PMCn.PMCn_m = 1, the pin is switched to alternative mode.
- Port IP Control register PIPCn:
  The register specifies the I/O control mode.
  - For PIPCn.PIPCn_m = 0, the I/O mode of the relevant pin is selected by PMn register;
  - For PIPCn.PIPCn_m = 1, the I/O mode is selected by the peripheral function.

In this mode, the pins operate as alternative functions. The I/O direction is selected by setting the PMn_m bit of the PMn register:

- The pin operates in alternative output mode when PMn_m = 0,
- The pin operates in alternative input mode when PMn_m = 1.

Table 3-3 shows the register setups for the 5 alternative functions, which can be selected using the port function control registers below:

- PFCn: port function control register,
- PFCEn: port function control expansion register,
- PFCAEn: port function control additional expansion register.

Table 3-3 Overview of Port Alternative Mode Selection

Alternative-Function	Bit Name					
	PMCn_m	PIPCn_m*1	PMn_m	PFCAEn_m	PFCEn_m	PFCn_m
Output Mode 1	1	0	0	0	0	0
Input Mode 1			1			
Output Mode 2			0		0	1
Input Mode 2			1			
Output Mode 3			0		1	0
Input Mode 3			1			
Output Mode 4			0		1	1
Input Mode 4			1			
Output Mode 5			0	1	0	0
Input Mode 5			1			
Output Mode 6			0		0	1
Input Mode 6			1			
Output Mode 7			0		1	0
Input Mode 7			1			
Output Mode 8			0		1	1
Input Mode 8			1			

Note: 1. If PIPCn.PIPCn_m=1, the I/O direction is directly controlled by the peripheral (alternative function) and PM is ignored.

The necessary information to enable the PWGC output ports, i.e. port functions and pin connections in RH850/U2A-EVA devices can be found in the attachment “External Input_Output and Pin Assignment for PWM Output_Diagnostics.xlsx”.

For further details of device pin function, please see the device User’ Manual, section 2 ‘Pin Functions’.

3.3 Trigger and Settings for the Diagnostic Function

If a diagnostic function is required, the PWSD unit provides the required settings to the A/D converter (ADCJ). This includes the control data as well as the trigger information for the A/D conversion.

The PWM feedback signals are input to the A/D converter, which performs A/D conversion.

3.3.1 Control data for the A/D conversion

According to Figure 3.1 and section 3.2.2 (2), the PWGC unit generates and outputs the triggers to PWSD unit, from which the conversion settings and triggers are sent to the ADCJ.

For an A/D conversion, the following converter setups are required:

- Selection of the ADCJ unit
- Settings of the required physical channel number
- En-/disabling an external multiplexer
- The address of the enabled multiplexer
- Selection of an error detection limit
- The wait time before executing a virtual channel
- All the above settings are specified in the PWSDnPVCRx registers.

3.3.2 Trigger control for A/D conversion

Upon the reception of a trigger from a PWGC channel, the related channel number is stored in the PWSDnQUEj (j = 0 to 7) register. This FIFO keeps up to 8 channel numbers to be converted.

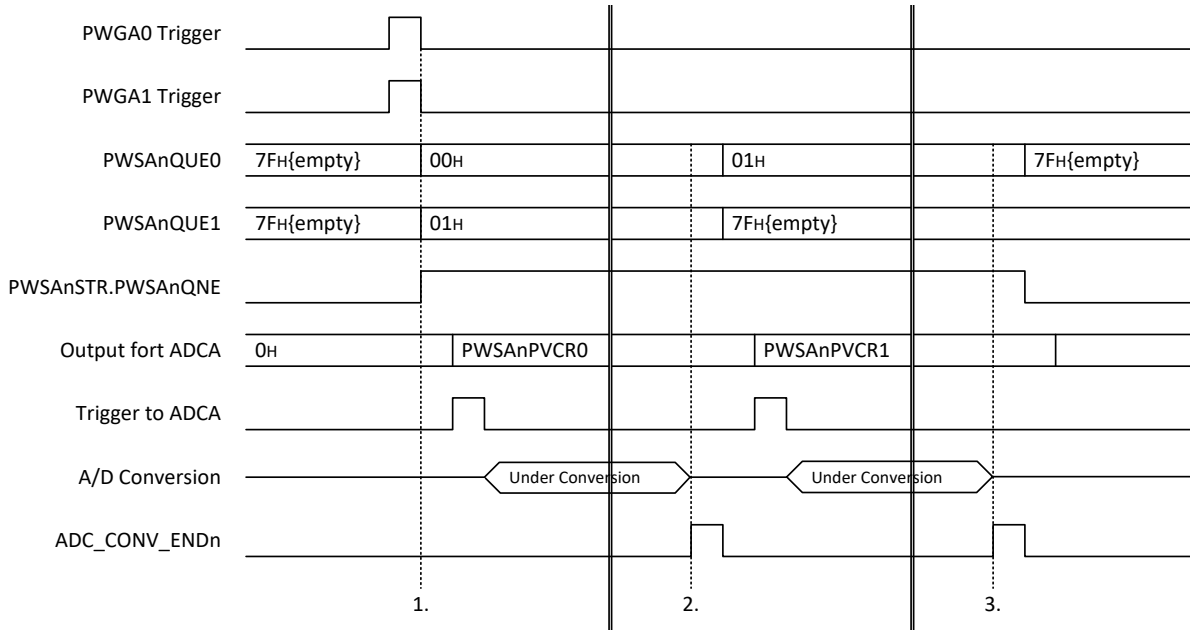
The PWSD outputs a trigger to the ADCJ. At the same time, the setup information for the A/D conversion of the channel, which is stored at the lowest queue position (PWSDnQUE0 register) is transferred to the corresponding ADCJ unit.

The output of the ADCJ setup information is kept until the next trigger is generated.

When the ADCJ unit completes the conversion, the following processes are executed:

- A conversion end interrupt request signal ADC_CONV_ENDn is generated by the ADCJ and output to the PWSD unit.
  Meanwhile, the conversion result is transferred to the corresponding PWSDnPWDDIRz register, if PWSD storing control is enabled (PWSDnCTL.PWSDnARSE bit is set to 1).
- The lowest element of the trigger queue is removed from the queue and the remaining elements are downshifted by one element.
  E.g. the information stored in register PWSDnQUE2 is written in PWSDnQUE1, the information stored in register PWSDnQUE1 is written in PWSDnQUE0
- Upon the next PWSD trigger the next conversion stored in the PWSDnQUE0 register will be initiated.

Figure 3.8 shows the waveforms of the related signals during the PWSD operation.



Notes: 1. The triggers occur simultaneously in channel 0 and channel 1 of PWGC. Channel 0 with the smaller channel number is stored in PWSDnQUE0, and channel 1 with the larger channel number is stored in PWSDnQUE1. A trigger is output to the ADCJ.
At the same time, the PWSDnPVCR0 information corresponding to the value stored in PWSDnQUE0 is transmitted to the A/D converter.
The A/D conversion for channel 0 starts and the conversion for channel 1 is in the waiting state.
The PWSDnQNE bit of register PWSDnSTR is set.

2. On completion of A/D conversion executed in step 1, a conversion end interrupt is generated and output to PWSD.
The channel number of PWSDnQUE1 shifts to PWSDnQUE0 and PWSDnQUE1 enters the empty state.
As similar to step1, a trigger is output to ADCJ, and the PWSDnPVCR1 information corresponding to the value in PWSDnQUE0 is transmitted to the ADCJ.

3. On completion of A/D conversion executed in step 2, a conversion end interrupt is generated and output to PWSD, PWSDnQUE0 enters the empty state.
The PWSDnQNE bit of register PWSDnSTR is cleared.

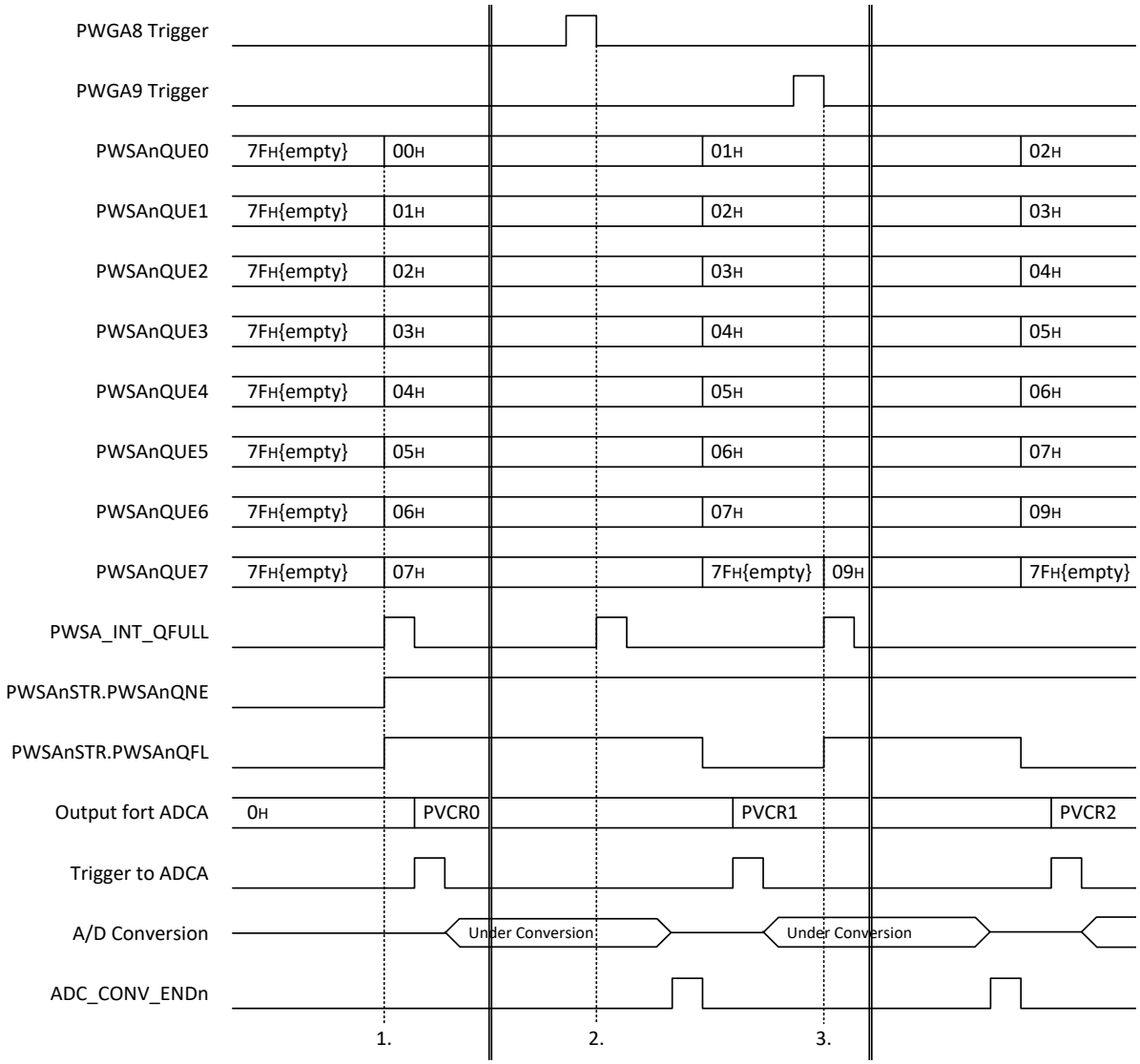
Figure 3.8 Example of PWSD operation

A PWSD_INT_QFULL interrupt request signal is generated in response to the PWGC_TRGOUTn signal on any of the following conditions:

- A trigger number is written to PWSDnQUE7.
- The PWSDnQUE7 position is already filled and no more write destinations in the queue are available.

In the QFULL state, the input PWGC trigger has no effect to PWSDnQUE register.

Figure 3.9 illustrates the PWSD operation when the trigger queue is full:



Notes: 1. A PWSD_INT_QFULL signal is generated, when the 7th trigger is written to PWSDnQUE7.
The PWSDnQNE and PWSDnQFL bits of the register PWSDnSTR are set to high level.
PWSD outputs a trigger signal to ADCJ, and transmits the conversion information related to the channel number stored in PWSDnQUE0 register.
ADCJ starts the conversion of PWGC channel 0, while the other channels are in the wait queue.

2. A PWGC trigger is output to PWSD during the conversion, and there is no more write destination available in the queue.
PWSD generates a QFULL interrupt request, and remains the values in the queue. The new PWGC trigger is ignored.
ADCJ generates an ADC_CONV_ENDn interrupt request at the completion of the A/D conversion.
The values in PWSDnQUE register are shifted; the PWSDnQUE7 register enters the empty state.
The PWSDnQFL bit of the register PWSDnSTR is cleared to 0.
PWSD triggers the ADCJ and outputs the corresponding data like in step 1.

3. A PWGC trigger is output to PWSD during the conversion, and PWSDnQUE7 is empty. Thus the trigger information is written to PWSDnQUE7, and a QFULL interrupt is generated.
The PWSDnQFL bit of the register PWSDnSTR is set to 1.

Figure 3.9 Example of PWSD operation when trigger Queue is full

4. A/D Conversion for the Diagnostics Function

This section describes the configuration of the A/D converter for the PWM diagnostics function.

4.1 Basic A/D converter configuration

For use of ADCJ units the following setups are required for their operation:

- ADCJ clock configuration
- Selection of the suspend mode
- Selection of 10-bit or 12-bit resolution
- Selection of the conversion result alignment control
- Configurations of the upper limit/lower limits; overwrite check for data registers; read and clear function for data registers.

4.1.1 Configuration of the clock supply

The clock supply for the employed ADCJ units can be configured regarding to the selector and divider registers. Besides, the module standby mode for these corresponding units should also be disabled, so that the clock connection to these units is enabled.

Table 4-1 lists the related registers.

Table 4-1 ADCJ Clock Configuration

Function	Register	Bit Name	Bit Position
Selection of the clock supply for ADCJ2	CKSC_AADCC	AADCSCSID[1:0]	1 to 0
Selection of the divider for ADCJ2 clock	CLKD_AADCC	AADCDCSID	0
Selection of the clock supply for ADCJ0 to 1	CKSC_ADC	ADCSCSID	0
Enable/disable the write access to clock registers	CLKKCPROT1	KCPROT[31:1]	31 to 1
		KCE	0
Enable/disable the clock supply to ADCJ2	MSR_ADCJ_AWO	MS_ADCJ_2	0
Enable/disable the clock supply to ADCJ 0 to 1	MSR_ADCJ_AWO	MS_ADCJ_1	1
		MS_ADCJ_0	0
Enable/disable the write access to MSR registers	MSRKCPROT	KCPROT[31:1]	31 to 1
		KCE	0

For a stable operation of the diagnostic function it is recommended to select the same clock supply for A/D converter and PWSD unit.

For detailed setups of these registers in the RH850/U2A-EVA devices, please refer to the device User's Manual, section 13.5.5 'Clock Selector / Divider Control Registers'.

4.1.2 ADCJ configurations for the PWM-Diag Function

Table 4-2 lists the registers and related bit positions in which are also necessary for the A/D conversion settings of the diagnostics:

Table 4-2 Basic ADCJ Configuration for the Diagnostics Function

Function	Register	Bit Name	Bit Position
PWM-Diag for SG4 disable/enable	ADCJnPWDCR	PWE	0
PWM-Diag HW triggers disable/enable	ADCJnPWDSGCR	TRGME	0
Selection of suspend mode	ADCJnADCR1	SUSMTD[1:0]	1 to 0
Selection of data format	ADCJnADCR2	DFMT[2:0]	6 to 4
Trigger Overlap Check Error Interrupt enable/disable	ADCJnSFTCR	TOCEIE	6
Read and clear disable/enable		RDCLRE	4
Overwrite Error Interrupt disable/enable		OWEIE	2
Parity Error Interrupt disable/enable		PEIE	1
ID Error Interrupt disable/enable		IDEIE	0
Upper/Lower Limit Check status clear	ADCJnPWVCLMSCR	PWVCLMSC	0
Upper/Lower Limit Check Interrupt disable/enable	ADCJnPWVCLMINTER	PWADULIE	0
Buffer amplifier enable/disable	ADCJnSMPCR	BUFAMPD	15
Selection of sampling period		SMPTS	12
Configuration of sampling time		SMPT[7:0]	7 to 0
Configuration of upper limit	ADCJnVCULLMTBR0 to 7	ULMTB[11:0]	31 to 20
Configuration of lower limit		LLMTB[11:0]	15 to 4
Configuration of wait time before executing a virtual channel	ADCJnWAITTR0 to 7	WAITTIME[13:0]	13 to 0

For setup details of the device RH850/U2A-EVA, please see the device User's Manual, section 43.3 'Registers (ADCJ)'.

Besides, the ADCJ provides following read-only registers to check the setups and status of the diagnostics function:

- When the PWM diagnostics function is running, the settings of the channel under conversion is transmitted from the corresponding PWSDnPVCRx register to PWM-Diag virtual channel register ADCJnPWDVCR of the related ADCJ.
- After completion of the A/D conversion, the conversion result can be checked by reading the following 2 registers:
  - PWM diagnostics data register ADCJnPWDDR,
  - PWM diagnostics data supplementary information register ADCJnPWDDIR.
  For an additional SW processing of the conversion results, the data in those registers can be transmitted to RAM using DTS or sDMA. See section 4.3.2 for detailed information.
- The comparison of the conversion result can be performed in HW using the available Upper Limit/Lower Limit detection function (ULL).
  In case the conversion result is out of the set limits, the register ADCJnPWVCLMSR stores occurrence of an ULL error.
  An upper/lower error interrupt request signal INT_UL is generated if an ULL error occurs.

4.2 Pin configuration of A/D converter

The PWM feedback signals are connected to ADCJ input pins.

The ADCJ inputs are special alternative I/O pins that are permanently connected to the A/D module. Thus, the ADCJ inputs are specified by directly connecting to the corresponding pins.

The pin assignments of the ADCJ input channels and their related configuration used in this application note are listed in the attachment "External Input_Output and Pin Assignment for PWM Output_Diagnostics.xlsx".

4.3 Conversion result handling

This section describes the handling of the conversion result values.

Even though a comparison of the conversion result using the Upper Limit/Lower Limit function of the A/D converter can be used, additional SW processing could be required in dedicated applications.

As only a single PWM Diagnostic conversion result value is stored within the ADCJ units, RH850/U2A-EVA devices provide 2 possibilities for the conversion result handling.
Whenever a conversion is completed, an ADC_CONV_ENDn interrupt signal is generated. This interrupt signal is output to PWSD as well as the DMA controller, and is possible to trigger the transmission of the current conversion result to the:

- PWSD unit
- a buffer located in RAM

As these processes are performed without any CPU interaction (directly performed by HW), the transfer is finished before the next conversion result is stored in the register.

4.3.1 Transmission to PWSD

If the current converted PWM channel is expected immediately for further application, the conversion results can be transferred back to PWSD.

Up to 96 channels dedicated PWSDnPWDDIRx registers are able to hold the conversion result from the ADCJ.

To enable this transmission, PWSD storing control bit PWSDnCTL.PWSDnARSE must be configured to 1.

4.3.2 Transmission to RAM

If a RAM buffer is considered as the results storage, a DMA transfer is required. For RH850/U2A-EVA devices, both DTS and sDMA module can be implemented to transmit the conversion result to RAM.

For details please refer to the HW User’s Manual Section 7 “sDMA Controller” and Section 8 “DTS Controller”.

5. Starting and stopping the Diagnostics

Before starting the diagnosis operation, the PWM-Diag, ADCJ and DMA peripherals (if DMA transfer is required) must be initialized. Figure 3.9 illustrate the starting and stopping process.

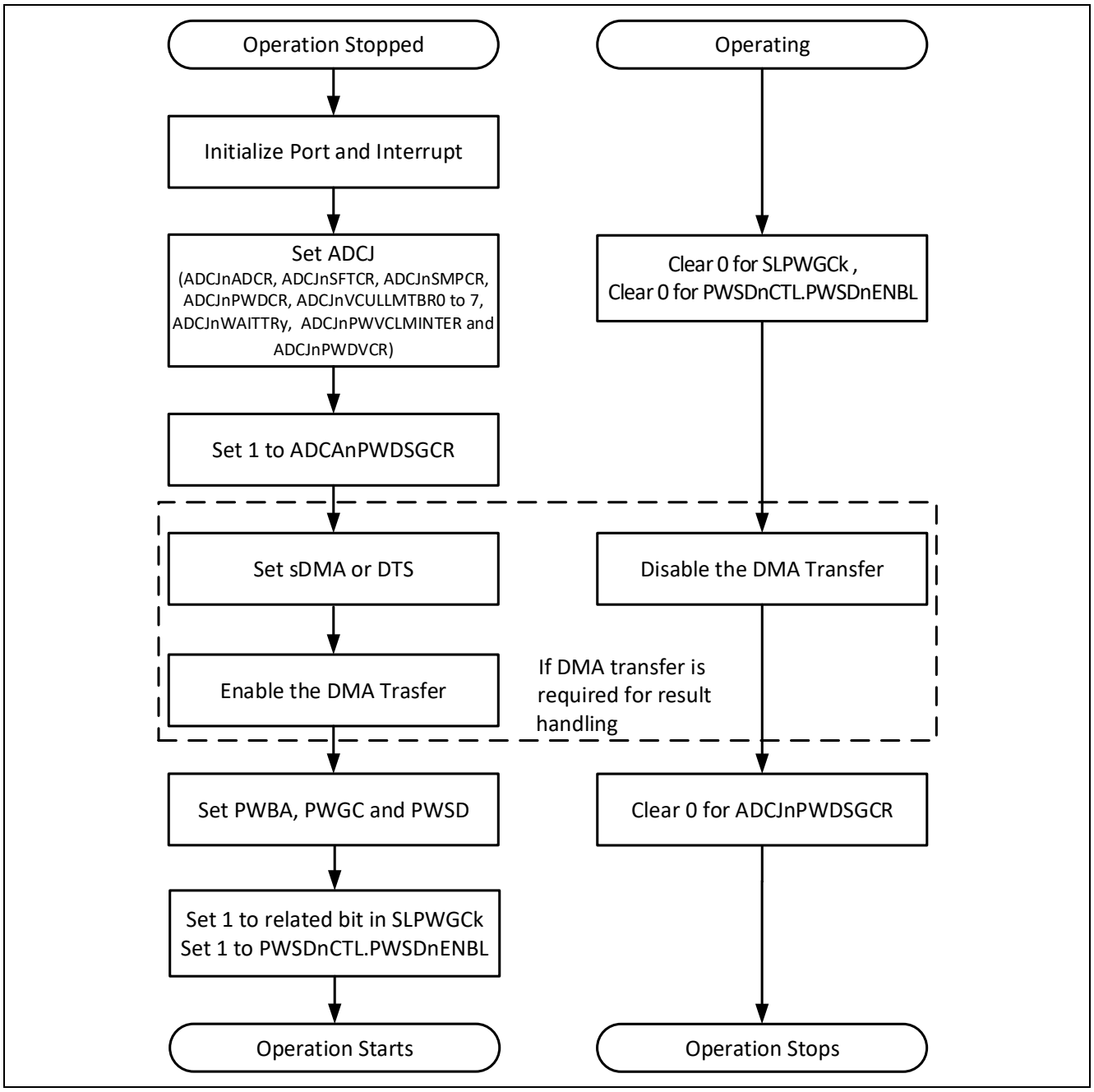


Figure 5.1 Starting and Stopping Operation of PWM Diagnostics

6. Interrupts

The provided PWM diagnostics application employs the following interrupts:

- PWGC_INTn interrupt request signals, each 32 interrupt sources are assigned in PWGCINTFhk registers that are connected to interrupt controller;
- PWSD Queue Full interrupt request signal PWSD_INT_QFULL, connected to the INTQFULL signal in the interrupt controller;
- A/D Converter Conversion End interrupt request signal ADC_CONV_ENDn for the PWM-Diag scan group, connected to PWSD;
- ADCJ Error interrupts INT_UL, connected to the INTADCJnERR in interrupt controller.

7. Sample software

The corresponding sample software of this document can be found in the attachment file: *r01an5463ed0110_rh850_u2a_sw.7z*.

The target device of the software is the RH850/U2A-EVA device R7F702300EABA.

The target board of the sample software is Piggyback Board Y-RH850-U2A-292PIN-PB-T1-Vx.

The sample software is executed from core 0, it contains the configuration of CPU clock, PWM-Diag functions, ADCJ, ports and the main function.

7.1 Clock Configuration

If the preprocessor symbol is defined as USE_BOARD=Y_RH850_U2A, the corresponding clock configuration can be executed automatically by checking the application board.

The source code of the clock configuration is located in …\ *r01an5463ed0110_rh850_u2a_sw* \device.h.

7.2 PWMDIAG.c and PWMDIAG.h

The location of these files is in folder: …\ *r01an5463ed0110_rh850_u2a_sw\src_mca*.

PWMDIAG.c contains the configuration for the PWM-Diag module:

- Configure PWMCLK: PWM_Setbrs(t_PWBA_CH PWBANr, uint16_t Brs).
- Configure PWM output:
  - PWM_SetOutput (uint8_t PWGANr, t_PWBA_CH PWBANr, t_PWM_PRD prdsel, uint16_t prd, uint32_t delay, uint32_t duty);
  - PWM_SetSpecialOutput (uint8_t PWGANr, t_PWM_LEVEL level).
- Configure PWM trigger:
  - PWM_SetTrigger (uint8_t PWGANr, uint16_t trigger, t_PWM_TRGOUT atwhere, t_PWGC_TRG_UPD update);
  - PWM_SetDiag (uint8_t PWGANr, t_PWSD_ADC_SEL ADCNr, t_PWSD_ADC_WAIT wait, uint32_t PhyCh, uint32_t MPXEn, uint32_t MPXAdd, t_PWSD_ADC_ULL ull).
- Start the PWM output and trigger:
  - PWM_EnOutput (t_PWM_DIS_EN output);
  - PWM_EnDiag (t_PWM_DIS_EN diag).

PWMDIAG.h is the header file of PWMDIAG.c, in which the parameters and settings of the PWM-Diag module are defined.

7.3 ADCJ.c and ADCJ.h

The location of these files is in folder: …\ *r01an5463ed0110_rh850_u2a_sw\src_mca*.

ADCJ.c contains the configuration for the ADCJ module:

- Configure ADCA clock: ADCJ_SetClk (uint8_t ADCNr, t_ADCJ_CLK clk, t_ADCJA_CLK aclk).
- Configure ADCA for PWM-Diag function:
  - ADCJ_PwmCfg (uint8_t ADCNr, t_SUSPEND_MODE smode, t_DATA_FORMAT dataf, t_SMP_PERIOD smpt, t_ADCJ_DIS_EN ErrInt);
  - ADCJ_AssignULL (uint8_t ADCNr, uint8_t RegNr, uint32_t UpperL, uint32_t LowerL).

ADCJ.h is the header file of ADCJ.c, in which the parameters and settings of the ADCJ module are defined.

7.4 PORT.c and PORT.h

The location of these files is in folder: …\ *r01an5463ed0110_rh850_u2a_sw\src_mca*.

PORT.c contains the configuration for port functions:

- Set port protection: PORT_SetProtection (t_PORT_GROUP portNr, t_PORT_PROTECTION onOff).
- Set port alternative function: PORT_SetAF (t_PORT_GROUP portNr, t_PORT_PIN pinNr, t_PORT_AF afNr, t_PORT_DIRECTION dir).
- Set port drive strength: PORT_SetAF (t_PORT_GROUP portNr, t_PORT_PIN pinNr, t_PORT_AF afNr, t_PORT_DIRECTION dir).

PORT.h is the header file of PORT.c, in which the variables and enumerated types for the configuration are defined.

7.5 main_pe0.c

The location of these files is in the folder: …\ *r01an5463ed0110_rh850_u2a_sw\src_mca*.

The sample software is for core0. The main_pe0.c contains the main function of this application.

The main function calls other local / global functions to implement the application. Some of the dedicated module configuration functions, such as PwmCfg () and AdcjCfg (), are included in main_pe0.c.

8. Summary

The PWM diagnostics function (PWM_Diag) provides the means to automatically monitor the resulting currents of a PWM load with the need for any CPU interaction.

This application note describes the configuration of the related peripherals (such as PWM-Diag, ADCJ), their operation, and the handling of the diagnostic values.

Problems can be detected by both HW (ADCJ error interrupts) and SW (diagnostic value handling) methods.

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Dec 15, 2020	-	Internal release
1.10	Jul 04, 2022	1	Update the product line
		7	Update of the reference document
		9	Update of Table 8-1 Number of units on PWM-Diag and physical channels on ADCJ for RH850/U2A-EVA devices
		-	Update of the attached document “External Input_Output and Pin Assignment for PWM Output_Diagnostics.xlsx”
		27 - 28	Added updated sample SW as attachment file, and updated the description of the sample SW.

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. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time 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 time 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 time 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 time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. 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 the 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.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is 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. Additionally, 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.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of 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.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.4.0-1 November 2017)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.