**IEP PWM PRU Firmware**

Applies to Firmware Release Version: 01.00.00.00

Publication Date: July 1, 2020

**Document License**

This work is licensed under the Creative Commons Attribution-NoDerivs 3.0 Unported License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nd/3.0/ or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

**Contributors to this document**

Copyright (C) 2019 Texas Instruments Incorporated - http://www.ti.com/



Texas Instruments, Incorporated  
12500 TI Blvd  
Dallas, TX 75243 USA

Texas Instruments, Incorporated  
20450 Century Boulevard  
Germantown, MD 20874 USA

**This document is intended for users interested the IEP PWM firmware design. It discusses details of the firmware design and implementation, and includes information concerning the memory maps, structures, state machines, and software flow of the firmware. Those users only concerned with using the firmware may not need to read this document.**

**The PWM firmware was developed on the AM654x EVM based on AM6548 SR1.0 silicon. There is no intent or commitment for updates to the firmware, and the firmware can be used for customer reference in their own development and maintenance.**

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Description of changes** |
| 0.1 | 04-13-19 | Initial Draft |
| 0.5 | 05-18-19 | Added detailed descriptions of FW registers.  Added FW register bit field type descriptions.  Added FW register bit field default values.  Added PWM\_CTRL & PWM\_STAT FW registers.  Added details on configurability (Init, Post-Init) of PWM parameters.  Added Initialization flow chart.  Updated State Machine. |
| 0.6 | 05-22-19 | Added references & definitions.  Added introduction & feature set.  Added details on PWM Module behavior.  Added PWM signal levels & action tables.  Added details on Host & FW communication.  Added State Machine flow charts.  Added PRU memory requirements.  Added EVM support to Test Plan section.  Added FW source files. |
| 0.7 | 08-19-19 | Updated Duty Cycle count calculation in description of for IEPx\_PWMm\_DC\_COUNT.  Added FW version information.  Add PWM\_CTRL:PWM\_EN & PWM\_STAT:PWM\_EN\_ACK to allow Host reconfiguration of FW registers before FW initialization is started. |
| 1.0 | 07-01-20 | Editorial update to cover page and footer for publication |

**TABLE OF CONTENTS**

LIST OF FIGURES 5

LIST OF TABLES 6

1 References 7

2 Definitions 7

3 Introduction 7

4 Feature Set 8

5 Design Description 9

5.1 Firmware Registers 9

5.1.1 Register Memory Map 9

5.1.2 Register Descriptions 12

5.1.2.1 Firmware Magic Number 12

5.1.2.2 Firmware Type 12

5.1.2.3 Firmware Version 12

5.1.2.4 Firmware Feature 12

5.1.2.5 Firmware Extended Feature 13

5.1.2.6 PWM\_CTRL 13

5.1.2.7 PWM\_STAT 13

5.1.2.8 IEPx\_PWM\_RECFG 14

5.1.2.9 IEP*x*\_PWM\_MODE 14

5.1.2.10 IEP*x*\_PWM\_EN 16

5.1.2.11 IEP*x*\_PWM\_PRD\_COUNT 17

5.1.2.12 IEP*x*\_PWM*m*\_DC\_COUNT 18

5.1.2.13 IEP*x*\_PWM*n*\_*n*+1\_DB\_COUNT 18

5.2 PWM Module Behavior 18

5.3 Design Details 19

5.3.1 PWM Signal Levels 21

5.3.2 PWM Action Tables 21

5.3.2.1 Initialization 22

5.3.2.2 Normal Execution 23

5.3.3 Host & PRU Firmware Communication 25

5.3.3.1 Initialization 25

5.3.3.2 Reconfiguration 28

5.3.4 Firmware State Machine 29

5.4 PRU Resource Usage 43

5.4.1 PRU Cycles 43

5.4.2 PRU Memory 43

5.5 Firmware Source Code 44

6 Test Plan 44

6.1 EVM Support 44

6.1.1 AM654x EVM + Breakout Board 44

LIST OF FIGURES

[Figure 1. Host & Firmware Design Layers 8](#_Toc44508201)

[Figure 2. Symmetric PWM Signals 20](#_Toc44508202)

[Figure 3. Method for PWM Disable/Enable & DC = 0% 20](#_Toc44508203)

[Figure 4. Method for DC = 100% & DC != 100% 21](#_Toc44508204)

[Figure 5. Firmware Initialization 27](#_Toc44508205)

[Figure 6. FW Register Bits for FW Initialization 28](#_Toc44508206)

[Figure 7. FW Register Bits for FW Reconfiguration 29](#_Toc44508207)

[Figure 8. Firmware State Machine 31](#_Toc44508208)

[Figure 9. SM INIT State, Top-Level Flow Chart 32](#_Toc44508209)

[Figure 10. SM INIT State, Initialize SNGL/CMPL PWM Flow Charts 33](#_Toc44508210)

[Figure 11. SM INIT State, Initialize SPWM Flow Chart 34](#_Toc44508211)

[Figure 12. SM LHS State, Top-Level Flow Chart 35](#_Toc44508212)

[Figure 13. SM RHS State, Top-Level Flow Chart 36](#_Toc44508213)

[Figure 14. SM LHS\_RECFG State, Top-Level Flow Chart 37](#_Toc44508214)

[Figure 15. SM LHS\_RECFG State, Reconfigure Period Flow Chart 38](#_Toc44508215)

[Figure 16. SM LHS\_RECFG State, Reconfigure PWM Enable or DC Flow Chart 39](#_Toc44508216)

[Figure 17. SM LHS\_RECFG State, Reconfigure DB Flow Chart 40](#_Toc44508217)

[Figure 18. SM LHS\_RECFG State, Finalize Period Reconfiguration Flow Chart 40](#_Toc44508218)

[Figure 19. SM RHS\_RECFG State, Top-Level Flow Chart 41](#_Toc44508219)

[Figure 20. SM RHS\_RECFG State, Execute RHS Stash for SNGL PWM Flow Chart 42](#_Toc44508220)

[Figure 21. SM RHS\_RECFG State, Execute RHS Stash for CMPL PWM Flow Chart 43](#_Toc44508221)

LIST OF TABLES

[Table 1. Referenced Materials 7](#_Toc44508222)

[Table 2. Definitions 7](#_Toc44508223)

[Table 3. Firmware Register Memory Map 11](#_Toc44508224)

[Table 4. Firmware Magic Number FW Register Description 12](#_Toc44508225)

[Table 5. Firmware Type FW Register Description 12](#_Toc44508226)

[Table 6. Firmware Version FW Register Description 12](#_Toc44508227)

[Table 7. Firmware Feature FW Register Description 12](#_Toc44508228)

[Table 8. Firmware Extended Feature FW Register Description 13](#_Toc44508229)

[Table 9. PWM\_CTRL FW Register Description 13](#_Toc44508230)

[Table 10. PWM\_STAT FW Register Description 14](#_Toc44508231)

[Table 11. IEPx\_PWM\_RECFG FW Register Description 14](#_Toc44508232)

[Table 12. IEP*x*\_PWM\_MODE FW Register Description 16](#_Toc44508233)

[Table 13. IEP*x*\_PWM\_EN FW Register Description 17](#_Toc44508234)

[Table 14. IEP*x*\_PWM\_PRD\_COUNT FW Register Description 17](#_Toc44508235)

[Table 15. IEPx\_PWM*m*\_DC\_COUNT FW Register Description 18](#_Toc44508236)

[Table 16. IEP*x*\_PWM*n*\_*n*+1\_DB\_COUNT FW Register Description 18](#_Toc44508237)

[Table 17. PWM Action Table Terms 22](#_Toc44508238)

[Table 18. Action Table, Normal PWM Initialization 22](#_Toc44508239)

[Table 19. Action Table, Sacrificial PWM Initialization 22](#_Toc44508240)

[Table 20. Action Table, No PWM Enable Reconfiguration & PWM Disabled 23](#_Toc44508241)

[Table 21.Action Table, No PWM Enable Reconfiguration & PWM Enabled 23](#_Toc44508242)

[Table 22. Action Table, PWM Enable Reconfiguration & PWM Enable 24](#_Toc44508243)

[Table 23. Action Table, PWM Enable Reconfiguration & PWM Disable 25](#_Toc44508244)

[Table 24. PWM Parameter Configurability 29](#_Toc44508245)

[Table 25. State Machine State Descriptions 30](#_Toc44508246)

[Table 26. FW Memory Usage 43](#_Toc44508247)

[Table 27. Firmware Source Files 44](#_Toc44508248)

[Table 28. AM654x PWMs 44](#_Toc44508249)

# References

The following references are related to the feature described in this document and shall be consulted as necessary.

|  |  |  |
| --- | --- | --- |
| No | Referenced Document | Link |
| 1 | * [AM65x/DRA80xM Processors Technical Reference Manual (Rev. C)](http://www.ti.com/lit/pdf/spruid7" \t "_blank) | <http://www.ti.com/lit/pdf/spruid7> |

Table . Referenced Materials

# Definitions

| **Acronym** | **Description** |
| --- | --- |
| API | Application Programming Interface |
| BB | Breakout Board |
| CMPL | Complementary |
| DB | Deadband |
| DC | Duty Cycle |
| FW | Firmware |
| IEP | Industrial Ethernet Peripheral |
| LHS | Left Hand Side |
| NEG | Complementary PWM Negative Signal |
| POS | Complementary PWM Positive Signal |
| PRU\_ICSSG | Programmable Real-Time Unit Subsystem and Industrial Communication Subsystem – Gigabit |
| PWM | Pulse Width Modulation |
| RHS | Right Hand Side |
| SM | State Machine |
| SNGL | Single-Ended PWM |
| SPWM | “Sacrificial” PWM |

Table . Definitions

# Introduction

PWM signals have several uses, including:

* Dimming an LED
* Providing analog output; if the digital output is filtered, it will provide an analog voltage between 0% and 100%
* Generating audio signals
* Providing variable speed control for motors
* Generating a modulated signal, for example to drive an infrared LED for a remote control.

The PRU\_ICSSG integrates one PWM module. The PWM module uses IEP0 and IEP1 compare events to generate PWM outputs. IEP0 and IEP1 each support 2 PWM sets of 3 phased motor control PWM outputs with 6 primary (positive) and 6 complimentary (negative) programmable PWM outputs per set. Additional details of the PWM module are contained in [1].

The IEP PWM PRU firmware (FW) provides Host application/driver software access to the PRU\_ICSSG PWM module. The FW implements a Host API which abstracts the PWM module hardware. This abstraction simplifies utilization of the PWM module by Host application/driver software since details of PWM hardware configuration and servicing are hidden by the Host API. Figure 1 shows the Host & FW design layers.



Figure . Host & Firmware Design Layers

# Feature Set

The IEP PWM FW supplies the following configuration options:

* Mode for each POS/NEG signal pair, where each pair can be configured as 2 Single-Ended PWMs or 1 Complementary PWM.
* Disable/Enable for all Single-Ended and Complementary PWMs.
* PWM Period, tested up to 1 msec. (1 kHz PWM frequency).
* PWM Duty Cycle, tested up to 8-bit (256 level) resolution.
* PWM Dead Band time for Complementary PWMs.

PWM Mode can only be configured at FW initialization time, i.e. once set PWM Mode can only be reconfigured by re-downloading the FW to PRU\_ICSSG PRU memories. All other PWM FW parameters (Disable/Enable, PWM period, etc.) can be dynamically reconfigured.

# Design Description

## Firmware Registers

### Register Memory Map

The register memory map for the FW executing on a particular PRU is located in the data memory associated with the PRU, i.e. the map is located in DRAM0 for PRU0 and DRAM1 for PRU1.

| **Memory offset** | **Size (Bytes)** | **Register name** | **Description** |
| --- | --- | --- | --- |
| 0x00000000 | 4 | Firmware Magic Number | Magic number for ICSSG PWM PRU firmware |
| 0x00000004 | 4 | Firmware Type | Firmware type information |
| 0x00000008 | 4 | Firmware Version | Firmware version information |
| 0x0000000C | 4 | Firmware Feature | Firmware feature information |
| 0x00000010 | 4 | Firmware Extended Feature | Firmware extended feature information |
| 0x00000014 | 4 | PWM\_CTRL | PWM Control |
| 0x00000018 | 4 | PWM\_STAT | PWM Status |
| 0x0000001C | 4 | IEP0\_PWM\_RECFG | IEP0 PWM Reconfiguration. Flags for controlling which PWM parameters are reconfigured.   * Host writes ‘1’ to request update of associated parameter. * Firmware clears flag to ‘0’ to inform Host update of parameter is complete. |
| 0x00000020 | 4 | IEP0\_PWM\_MODE | IEP0 PWM Mode. Flags for selecting PWM mode (Single-Ended or Complementary). |
| 0x00000024 | 4 | IEP0\_PWM\_EN | IEP0 PWM Enable. Flags for enabling / disabling PWMs. |
| 0x00000028 | 4 | IEP0\_PWM\_PRD\_COUNT | IEP0 PWM Period Count. IEP counter value necessary for desired PWM frequency. |
| 0x0000002C | 4 | IEP0\_PWM0\_DC\_COUNT | IEP0 PWM0 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000030 | 4 | IEP0\_PWM1\_DC\_COUNT | IEP0 PWM1 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000034 | 4 | IEP0\_PWM2\_DC\_COUNT | IEP0 PWM2 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000038 | 4 | IEP0\_PWM3\_DC COUNT | IEP0 PWM3 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x0000003C | 4 | IEP0\_PWM4\_DC COUNT | IEP0 PWM4 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000040 | 4 | IEP0\_PWM5\_DC COUNT | IEP0 PWM5 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000044 | 4 | IEP0\_PWM6\_DC COUNT | IEP0 PWM6 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000048 | 4 | IEP0\_PWM7\_DC COUNT | IEP0 PWM7 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x0000004C | 4 | IEP0\_PWM8\_DC COUNT | IEP0 PWM8 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000050 | 4 | IEP0\_PWM9\_DC COUNT | IEP0 PWM9 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000054 | 4 | IEP0\_PWM10\_DC COUNT | IEP0 PWM10 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000058 | 4 | IEP0\_PWM11\_DC COUNT | IEP0 PWM11 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x0000005C | 2 | IEP0\_PWM0\_1\_DEADBAND | IEP0 PWM0 & 1 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x0000005E | 2 | IEP0\_PWM2\_3\_DEADBAND | IEP0 PWM2 & 3 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x00000060 | 2 | IEP0\_PWM4\_5\_DEADBAND | IEP0 PWM4 & 5 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x00000062 | 2 | IEP0\_PWM6\_7\_DEADBAND | IEP0 PWM6 & 7 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x00000064 | 2 | IEP0\_PWM8\_9\_DEADBAND | IEP0 PWM8 & 9 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x00000066 | 2 | IEP0\_PWM10\_11\_DEADBAND | IEP0 PWM10 & 11 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x00000068 | 4 | IEP1\_PWM\_RECFG | IEP1 PWM Reconfiguration. Flags for controlling which PWM parameters are reconfigured.   * Host writes ‘1’ to request update of associated parameter. * Firmware clears flag to ‘0’ to inform Host update of parameter is complete. |
| 0x0000006C | 4 | IEP1\_PWM\_MODE | IEP1 PWM Mode. Flags for selecting PWM mode (Single-Ended or Complementary). |
| 0x00000070 | 4 | IEP1\_PWM\_EN | IEP1 PWM Enable. Flags for enabling / disabling PWMs. |
| 0x00000074 | 4 | IEP1\_PWM\_PRD\_COUNT | IEP1 PWM Period Count. IEP counter value necessary for desired PWM frequency. |
| 0x00000078 | 4 | IEP1\_PWM0\_DC\_COUNT | IEP1 PWM0 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x0000007C | 4 | IEP1\_PWM1\_DC COUNT | IEP1 PWM1 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000080 | 4 | IEP1\_PWM2\_DC COUNT | IEP1 PWM2 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000084 | 4 | IEP1\_PWM3\_DC COUNT | IEP1 PWM3 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000088 | 4 | IEP1\_PWM4\_DC COUNT | IEP1 PWM4 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x0000008C | 4 | IEP1\_PWM5\_DC COUNT | IEP1 PWM5 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000090 | 4 | IEP1\_PWM6\_DC COUNT | IEP1 PWM6 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000094 | 4 | IEP1\_PWM7\_DC COUNT | IEP1 PWM7 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x00000098 | 4 | IEP1\_PWM8\_DC COUNT | IEP1 PWM8 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x0000009C | 4 | IEP1\_PWM9\_DC COUNT | IEP1 PWM9 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x000000A0 | 4 | IEP1\_PWM10\_DC COUNT | IEP1 PWM10 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x000000A4 | 4 | IEP1\_PWM11\_DC COUNT | IEP1 PWM11 Duty Cycle Count. IEP counter value necessary for desired PWM duty cycle. |
| 0x000000A8 | 2 | IEP1\_PWM0\_1\_DEADBAND | IEP1 PWM0 & 1 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x000000AA | 2 | IEP1\_PWM2\_3\_DEADBAND | IEP1 PWM2 & 3 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x000000AC | 2 | IEP1\_PWM4\_5\_DEADBAND | IEP1 PWM4 & 5 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x000000AE | 2 | IEP1\_PWM6\_7\_DEADBAND | IEP1 PWM6 & 7 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x000000B0 | 2 | IEP1\_PWM8\_9\_DEADBAND | IEP1 PWM8 & 9 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x000000B2 | 2 | IEP1\_PWM10\_11\_DEADBAND | IEP1 PWM10 & 11 Dead Band Count. IEP counter value necessary for desired complementary mode dead band. |
| 0x000000B4 | Remaining | Reserved |  |

Table . Firmware Register Memory Map

### Register Descriptions

#### Firmware Magic Number

The table below contains descriptions of the Firmware Magic Number FW register bit fields.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-24 | Magic Number Byte 3 | R-4Dh | ‘M’ |
| 23-16 | Magic Number Byte 2 | R-57h | ‘W’ |
| 15-8 | Magic Number Byte 1 | R-50h | ‘P’ |
| 7-0 | Magic Number Byte 0 | R-47h | ‘G’ |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . Firmware Magic Number FW Register Description

#### Firmware Type

The table below contains descriptions of the Firmware Type FW register bit fields.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-24 | Firmware ICSS version | R-01h | 01h – ICSS Revision 2 |
| 23-8 | Firmware Protocol Type | R-0002h | 0002h – Control Class |
| 7-0 | Firmware Protocol Type Version | R-01h | 01h – ICSSG PWM version (arbitrary, no PWM specification) |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . Firmware Type FW Register Description

#### Firmware Version

The table below contains descriptions of the Firmware Version FW register bit fields.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31 | Firmware Release or Internal Version | R-1b | 0b – Release version  1b – Internal version |
| 30-24 | Firmware Version Major | R-0000001b | Major version number |
| 23-8 | Firmware Version Minor | R-00h | Minor version number |
| 7-0 | Firmware Version Build | R-00h | Build version number |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . Firmware Version FW Register Description

#### Firmware Feature

The table below contains descriptions of the Firmware Feature FW register bit fields.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-13 | RESERVED | - | - |
| 12-9 | Firmware IEP1 Number of Supported PWMs | R-1100b | Firmware supports supports 12 PWMs on IEP1. |
| 8-5 | Firmware IEP0 Number of Supported PWMs | R-1100b | Firmware supports supports 12 PWMs on IEP0. |
| 4-0 | Firmware Number of Supported PWMs | R-11000b | Firmware supports 24 PWMs |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . Firmware Feature FW Register Description

#### Firmware Extended Feature

The table below contains descriptions of the Firmware Extended Feature FW register bit fields.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-0 | Offset to extended feature structure for future use | R-00000000h | Reserved for future use |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . Firmware Extended Feature FW Register Description

#### PWM\_CTRL

The table below contains descriptions of the PWM\_CTRL FW register bit fields.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-3 | RESERVED | - | - |
| 2 | IEP1\_PWM\_GBL\_EN | R/W-0b | IEP1 PWMs global enable. Controls whether IEP1 PWMs are disabled or enabled. This flag is only read during firmware initialization time.  0b – IEP1 PWMs disabled.  1b – IEP1 PWMs enabled. |
| 1 | IEP0\_PWM\_GBL\_EN | R/W-0b | IEP0 PWMs global enable. Controls whether IEP1 PWMs are disabled or enabled. This flag is only read during firmware initialization time.  0b – IEP0 PWMs disabled.  1b – IEP0 PWMs enabled. |
| 0 | PWM\_EN | R/W-0b | PWM enable. FW continuously polls this bit in a loop at start of initialization until the bit is set to ‘1’. This allows Host driver software to configure FW registers with non-default values before FW initialization commences.  0b – PWMs disabled.  1b – PWMs enabled. |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . PWM\_CTRL FW Register Description

#### PWM\_STAT

The table below contains descriptions of the PWM\_STAT FW register bit fields.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-4 | RESERVED | - | - |
| 3 | FW\_INIT | R-0b | Firmware initialized flag:  0b – Firmware initialization not complete.  1b – Firmware initialization complete. After enabling PRU, Host must wait for firmware initialization complete before writing to any firmware registers. |
| 2 | IEP1\_PWM\_GBL\_EN\_ACK | R-0b | IEP1 PWMs global enable acknowledge. Acknowledge flag corresponding to IEP1 enable control.  0b – IEP1 PWMs disabled.  1b – IEP1 PWMs enabled. |
| 1 | IEP0\_PWM\_GBL\_EN\_ACK | R-0b | IEP0 PWMs global enable acknowledge. Acknowledge flag corresponding to IEP0 enable control.  0b – IEP0 PWMs disabled.  1b – IEP0 PWMs enabled. |
| 0 | PWM\_EN\_ACK | R-0b | PWM enable acknowledge. Acknowledge flag corresponding to PWM enable control.  0b – PWMs disabled.  1b – PWMs enabled. |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . PWM\_STAT FW Register Description

#### IEPx\_PWM\_RECFG

The table below contains descriptions of the IEP*x*\_PWM\_MODE FW register bit fields, where *x* = 0, 1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-20 | RESERVED | - | - |
| 19-14 | RECFG\_IEPx\_PWM\_DB\_COUNT | R/W-000000b | Reconfigure IEPx PWM Deadband Count. Bit *n* in field corresponds with PWM 2*n\_2n+1*, e.g. Bit 0 is for PWM0\_1.  For any bit in field:  0 – no reconfiguration requested  1 – reconfiguration requested |
| 13-2 | RECFG\_IEPx\_PWM\_DC\_COUNT | R/W-000h | Reconfigure IEPx PWM Duty Cycle Count. Bit *n* in field corresponds with PWM *n*, e.g. Bit 0 is for PWM 0.  For any bit in field:  0 – no reconfiguration requested  1 – reconfiguration requested |
| 1 | RECFG\_IEPx\_PWM\_PRD\_COUNT | R/W-0b | Reconfigure IEPx PWM Period Count:  0 – no reconfiguration requested  1 – reconfiguration requested |
| 0 | RECFG\_IEPx\_PWM\_EN | R/W-0b | Reconfigure IEPx PWM Enable:  0 – no reconfiguration requested  1 – reconfiguration requested |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . IEPx\_PWM\_RECFG FW Register Description

#### IEP*x*\_PWM\_MODE

The table below contains descriptions of the IEP*x*\_PWM\_MODE FW register bit fields, where *x* = 0, 1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-6 | RESERVED | - | - |
| 5 | PWM10\_11\_MODE | R/W-0b | PWM 10 & 11 mode select:  0 – Single-Ended mode  1 – Complementary mode  In Single-Ended mode, PWM10 and PWM11 are enabled/disabled individually, i.e.:   * IEPx\_PWM\_EN:PWM10\_EN = <0 or 1> * IEPx\_PWM\_EN:PWM11\_EN = <0 or 1>   In Complementary mode, PWM10 and PWM11 are enabled/disabled as a pair, i.e.:   * IEPx\_PWM\_EN:PWM10\_EN = 0,  IEPx\_PWM\_EN:PWM11\_EN = X : PWM10 & 11 disabled * IEPx\_PWM\_EN:PWM10\_EN = 1, IEPx\_PWM\_EN:PWM11\_EN = X : PWM10 & 11 enabled |
| 4 | PWM8\_9\_MODE | R/W-0b | PWM 8 & 9 mode select:  0 – Single-Ended mode  1 – Complementary mode  In Single-Ended mode, PWM6 and PWM7 are enabled/disabled individually, i.e.:   * IEPx\_PWM\_EN:PWM8\_EN = <0 or 1> * IEPx\_PWM\_EN:PWM9\_EN = <0 or 1>   In Complementary mode, PWM8 and PWM9 are enabled/disabled as a pair, i.e.:   * IEPx\_PWM\_EN:PWM8\_EN = 0,  IEPx\_PWM\_EN:PWM9\_EN = X : PWM8 & 9 disabled * IEPx\_PWM\_EN:PWM8\_EN = 1, IEPx\_PWM\_EN:PWM9\_EN = X : PWM8 & 9 enabled |
| 3 | PWM6\_7\_MODE | R/W-0b | PWM 6 & 7 mode select:  0 – Single-Ended mode  1 – Complementary mode  In Single-Ended mode, PWM6 and PWM7 are enabled/disabled individually, i.e.:   * IEPx\_PWM\_EN:PWM6\_EN = <0 or 1> * IEPx\_PWM\_EN:PWM7\_EN = <0 or 1>   In Complementary mode, PWM6 and PWM7 are enabled/disabled as a pair, i.e.:   * IEPx\_PWM\_EN:PWM6\_EN = 0,  IEPx\_PWM\_EN:PWM7\_EN = X : PWM6 & 7 disabled * IEPx\_PWM\_EN:PWM6\_EN = 1, IEPx\_PWM\_EN:PWM7\_EN = X : PWM6 & 7 enabled |
| 2 | PWM4\_5\_MODE | R/W-0b | PWM 4 & 5 mode select:  0 – Single-Ended mode  1 – Complementary mode  In Single-Ended mode, PWM4 and PWM5 are enabled/disabled individually, i.e.:   * IEPx\_PWM\_EN:PWM4\_EN = <0 or 1> * IEPx\_PWM\_EN:PWM5\_EN = <0 or 1>   In Complementary mode, PWM4 and PWM5 are enabled/disabled as a pair, i.e.:   * IEPx\_PWM\_EN:PWM4\_EN = 0,  IEPx\_PWM\_EN:PWM5\_EN = X : PWM4 & 5 disabled * IEPx\_PWM\_EN:PWM4\_EN = 1, IEPx\_PWM\_EN:PWM5\_EN = X : PWM4 & 5 enabled |
| 1 | PWM2\_3\_MODE | R/W-0b | PWM 2 & 3 mode select:  0 – Single-Ended mode  1 – Complementary mode  In Single-Ended mode, PWM2 and PWM2 are enabled/disabled individually, i.e.:   * IEPx\_PWM\_EN:PWM2\_EN = <0 or 1> * IEPx\_PWM\_EN:PWM3\_EN = <0 or 1>   In Complementary mode, PWM0 and PWM1 are enabled/disabled as a pair, i.e.:   * IEPx\_PWM\_EN:PWM2\_EN = 0,  IEPx\_PWM\_EN:PWM3\_EN = X : PWM2 & 3 disabled * IEPx\_PWM\_EN:PWM2\_EN = 1, IEPx\_PWM\_EN:PWM3\_EN = X : PWM2 & 3 enabled |
| 0 | PWM0\_1\_MODE | R/W-0b | PWM 0 & 1 mode select:  0 – Single-Ended mode  1 – Complementary mode  In Single-Ended mode, PWM0 and PWM1 are enabled/disabled individually, i.e.:   * IEPx\_PWM\_EN:PWM0\_EN = <0 or 1> * IEPx\_PWM\_EN:PWM1\_EN = <0 or 1>   In Complementary mode, PWM0 and PWM1 are enabled/disabled as a pair, i.e.:   * IEPx\_PWM\_EN:PWM0\_EN = 0,  IEPx\_PWM\_EN:PWM1\_EN = X : PWM0 & 1 disabled * IEPx\_PWM\_EN:PWM0\_EN = 1, IEPx\_PWM\_EN:PWM1\_EN = X : PWM0 & 1 enabled |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . IEP*x*\_PWM\_MODE FW Register Description

#### IEP*x*\_PWM\_EN

The table below contains descriptions of the IEP*x*\_PWM\_EN FW register bit fields, where *x* = 0, 1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-12 | RESERVED | - | - |
| 11 | PWM11\_EN | R/W-0b | PWM 11 enable:  0 – disabled  1 – enabled |
| 10 | PWM10\_EN | R/W-0b | PWM 10 enable:  0 – disabled  1 – enabled |
| 9 | PWM9\_EN | R/W-0b | PWM 9 enable:  0 – disabled  1 – enabled |
| 8 | PWM8\_EN | R/W-0b | PWM 8 enable:  0 – disabled  1 – enabled |
| 7 | PWM7\_EN | R/W-0b | PWM 7 enable:  0 – disabled  1 – enabled |
| 6 | PWM6\_EN | R/W-0b | PWM 6 enable:  0 – disabled  1 – enabled |
| 5 | PWM5\_EN | R/W-0b | PWM 5 enable:  0 – disabled  1 – enabled |
| 4 | PWM4\_EN | R/W-0b | PWM 4 enable:  0 – disabled  1 – enabled |
| 3 | PWM3\_EN | R/W-0b | PWM 3 enable:  0 – disabled  1 – enabled |
| 2 | PWM2\_EN | R/W-0b | PWM 2 enable:  0 – disabled  1 – enabled |
| 1 | PWM1\_EN | R/W-0b | PWM 1 enable:  0 – disabled  1 – enabled |
| 0 | PWM0\_EN | R/W-0b | PWM 0 enable:  0 – disabled  1 – enabled |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . IEP*x*\_PWM\_EN FW Register Description

#### IEP*x*\_PWM\_PRD\_COUNT

The table below contains descriptions of the IEP*x*\_PWM\_EN FW register bit fields, where *x* = 0,1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-0 | PRD\_COUNT | R/W-0007A120h | IEP counter value necessary for desired PWM period.  PRD\_COUNT = (IEP clock freq)/(PWM freq)/2\*5, where 5 is default IEP counter increment value.  E.g. PWM freq = 1 kHz:   * IEP clock freq = 250 MHz: PRD\_COUNT = (250e6)/(1e3)/2\*5 = 625e3. * IEP clock freq = 200 MHz: PRD\_COUNT = (200e6)/(1e3)/2\*5 = 500e3. |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . IEP*x*\_PWM\_PRD\_COUNT FW Register Description

#### IEP*x*\_PWM*m*\_DC\_COUNT

The table below contains descriptions of the IEP*x*\_PWM*m*\_DC\_COUNT FW register bit fields, where *x* = 0, 1 and *m* = 0…11.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 31-0 | DC\_COUNT | R/W-0007A120h | IEP counter value necessary for desired PWM duty cycle.  DC\_COUNT = round(DC% \*(2\*PRD\_COUNT)), where DC% is the Duty Cycle percentage.  E.g. PWM freq = 1 kHz, DC% = 50:   * IEP clock freq = 250 MHz: DC\_COUNT = round(0.5 \* (2\*625e3)) = 6.25e3. * IEP clock freq = 200 MHz: DC\_COUNT = round(0.5 \* (2\*500e3)) = 500e3. |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . IEPx\_PWM*m*\_DC\_COUNT FW Register Description

#### IEP*x*\_PWM*n*\_*n*+1\_DB\_COUNT

The table below contains descriptions of the IEP*x*\_PWM*n*\_*n*+1\_DB\_COUNT FW register bit fields, where *x* = 0, 1 and *n* = 0, 2, 4, 6, 8, 10.

|  |  |  |  |
| --- | --- | --- | --- |
| **Bit** | **Field** | **Type** | **Description** |
| 15-0 | DB\_COUNT | R/W-0A00h | IEP counter value necessary for desired complementary mode deadband.  DB\_COUNT = (IEP clock freq)\*(Dead band time)\*5  E.g. Dead band time = 2.56 usec.:   * IEP clock freq = 250 MHz: DB\_COUNT = (250e6)\*(2.56e-6)\*5 = 3200 * IEP clock freq = 200 MHz: DB\_COUNT = (200e6)\*(2.56e-6)\*5 = 2560 |

LEGEND: R/W = Read/Write; R = Read Only; -n = value after firmware load

Table . IEP*x*\_PWM*n*\_*n*+1\_DB\_COUNT FW Register Description

## PWM Module Behavior

The PWM module exhibits the following behaviors for each IEP PWM set CMP*x*, *x* = 1…5:

1. The first CMP*x* event for the set causes the set to transition from Init to Active State, but the Active State update (High, Low, Toggle) for the PWM signal associated with the CMP*x* event doesn’t occur.
2. The IEP CMP*x* event which triggers the Active State in the IEP CMP0 period must be cleared by software before the CMP*x* event can occur in next IEP CMP0 period.

Generating proper output signals from the PWM module requires software to account for these behaviors. Different software design approaches can be used for this purpose, each with different tradeoffs. The design selected for implementation of IEP PWM FW was chosen because it provides the highest number of features from among the design methods considered. Specifically, the selected design provides symmetric (vs. left justified) PWMs together with:

1. Operation of all PWM module output signals.
2. Complementary PWMs.

The cost of providing these features is the FW must:

1. Continuously clear CMP*x*, resulting in a tight real-time deadline.
2. Write the CMP*x* Shadow Register each IEP CMP0 (1/2 PWM) period.
3. Implement PWM Enable/Disable, DC = 0%, and DC = 100% by moving the CMP*x* Shadow Register "inside" and "outside" the CMP0 period according to table of PWM state transitions.

## Design Details

The FW design approach is summarized below.

* For each ICSSG PWM set *m*, *m* = 0, 1, 2, 3 (2 sets/IEP \* 2 IEP = 4): set ICSSG\_PWM*m***:**PWM*m*\_TRIP\_CMP0\_EN = 0 so that (Active ⇒ Initial) State transition does not occur on CMP0 event.
* For each ICSSG PWM set *m*, *m* = 0, 1, 2, 3: Select 1 (of 6) PWMs in each PWM set as “Sacrificial” PWM (SPWM), where SPWM handles missing Active State update in first IEP CMP0 period:
  + Set Active State Signal for all PWMs to Toggle.
  + For SPWM: set CMP PWM inside CMP0 period & Initial State Signal to opposite other PWMs.
  + For other PWMs: set CMP outside CMP0 period.
  + Upon completion of first IEP CMP period: set CMP for other PWMs inside CMP0 period.
* Continuously clear CMP*x* for each IEP. Worst-case deadline before next CMP*x*: Deadline = (PWM period)/(PWM resolution), e.g. 1 msec./256 = 3.91 usec.
* Write CMP*x* Shadow Registers for enabled PWMs each IEP CMP0 period.
* Handle PWM Enable/Disable, DC = 0% & DC = 100% by moving CMP*x* “inside” and “outside” CMP0 period.

The symmetric PWM signals generated using this design approach are show in Figure 2. As can be observed, the symmetric PWM period is twice the IEP CMP0 period. Each PWM period starts with the “Left” or LHS (first IEP0 CMP0 period) followed by the “Right” or RHS (second IEP0 CMP0 period).

Figure 3 shows the method for handling PWM Enable/Disable and DC = 0%, while the method for handling DC = 100% and DC != 100% is shown in Figure 4.



Figure . Symmetric PWM Signals



Figure . Method for PWM Disable/Enable & DC = 0%



Figure . Method for DC = 100% & DC != 100%

### PWM Signal Levels

The PWM signal levels are fixed in the current FW design. PWM signal levels are described below.

Initial signal levels for PWM

* SNGL PWM : High
* CMPL PWM POS/NEG : High/Low

Initial signal levels for SPWM

* SNGL PWM : Low
* CMPL PWM POS/NEG : Low/High

Active signal levels for CMPL PWMs

* POS : High
* NEG : Low

Active signal levels for disabled PWMs

* SNGL PWM : Low
* CMPL PWM POS/NEG : Low/High

### PWM Action Tables

The FW must take certain actions to prepare for the next LHS or RHS of the PWM signal. These actions are a function of the current PWM enable and DC, together with any new PWM enable and DC received from reconfiguration requests. The required actions also depend on whether PWMs are being initialized (i.e. during first time initialization or period reconfiguration), or updated as part of normal (post-initialization) execution.

The actions required in all circumstances are presented in Table 17 - Table 22 below. The terms used in these tables are described in Table 16.

|  |  |  |
| --- | --- | --- |
| **Term Column(s)** | **Term** | **Description** |
| Latch | En\_new = <value> | Latch new PWM enable setting <value> received from enable reconfiguration request to internal FW state |
| Latch | DC\_new = <value> | Latch new PWM DC setting <value> received from DC reconfiguration request to internal FW state |
| LHS / RHS prepare action | CMP SR = EV | Set Compare Shadow Register to Early Value inside IEP0 Period, e.g. 5. |
| LHS / RHS prepare action | CMP SR = PRD | Set Compare Shadow Register to value outside CMP0 Period |
| LHS / RHS prepare action | CMP SR = PRD-DC\_LHS\_X | Set Compare Shadow Register to (CMP0 Period minus LHS value for DC = X) |
| LHS / RHS prepare action | CMP SR = PRD-DC\_LHS\_Y | Set Compare Shadow Register to (CMP0 Period minus LHS value for DC = Y) |
| LHS / RHS prepare action | Enable SR update | Enable write to CMP Shadow Register during LHS / RHS preparation |
| LHS / RHS prepare action | Disable SR update | Disable write to CMP Shadow Register during LHS / RHS preparation |

Table . PWM Action Table Terms

#### Initialization

|  |  |  |  |
| --- | --- | --- | --- |
| **En\_new** | **DC\_cur** | **LHS prepare action** | **RHS prepare action** |
| 0 | <any> | - | - |
| 1 | 0 | - | - |
| 1 | 100 | CMP SR = EV  & Enable SR update | CMP SR = PRD  & Disable SR update |
| 1 | X | CMP SR = PRD-DC\_LHS\_X  & Enable SR update | - |

Table . Action Table, Normal PWM Initialization

|  |  |  |  |
| --- | --- | --- | --- |
| **En\_new** | **DC\_cur** | **LHS prepare action** | **RHS prepare action** |
| 0 | <any> | CMP SR = PRD  & Disable SR update | **-** |
| 1 | 0 | CMP SR = PRD  & Disable SR update | **-** |
| 1 | 100 | - | CMP SR = PRD  & Disable SR update |
| 1 | X | CMP SR = PRD-DC\_LHS\_X  & Enable SR update | **-** |

Table . Action Table, Sacrificial PWM Initialization

#### Normal Execution

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **En\_old** | **En\_new** | **DC\_cur** | **DC\_new** | **PWM State**  **cur** | **PWM State**  **new** | **Latch** | **LHS prepare**  **Action** | **RHS prepare**  **action** |
| 0 | 0 | X | X | L/H | L/H | - | - | - |
| 0 | 0 | X | Y | L/H | L/H | DC\_new = Y | - | - |
| 0 | 0 | X | 0 | L/H | L/H | DC\_new = 0 | - | - |
| 0 | 0 | X | 100 | L/H | L/H | DC\_new = 100 | - | - |
| 0 | 0 | 0 | Y | L/H | L/H | DC\_new = Y | - | - |
| 0 | 0 | 0 | 0 | L/H | L/H | - | - | - |
| 0 | 0 | 0 | 100 | L/H | L/H | DC\_new = 100 | - | - |
| 0 | 0 | 100 | Y | L/H | L/H | DC\_new = Y | - | - |
| 0 | 0 | 100 | 0 | L/H | L/H | DC\_new = 0 | - | - |
| 0 | 0 | 100 | 100 | L/H | L/H | - | - | - |

Table . Action Table, No PWM Enable Reconfiguration & PWM Disabled

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **En\_old** | **En\_new** | **DC\_cur** | **DC\_new** | **PWM State**  **cur** | **PWM State**  **new** | **Latch** | **LHS prepare**  **Action** | **RHS prepare**  **Action** |
| 1 | 1 | X | X | DC\_x | DC\_x | - | - | - |
| 1 | 1 | X | Y | DC\_x | DC\_y | DC\_new = Y | CMP SR =  PRD−DC\_LHS\_Y | - |
| 1 | 1 | X | 0 | DC\_x | L/H | DC\_new = 0 | CMP SR = PRD  & Disable SR update | - |
| 1 | 1 | X | 100 | DC\_x | H/L | DC\_new = 100 | - | CMP SR = PRD  & Disable SR update |
| 1 | 1 | 0 | Y | L/H | DC\_y | DC\_new = Y | CMP SR = PRD−DC\_LHS\_Y  & Enable SR update | - |
| 1 | 1 | 0 | 0 | L/H | L/H | - | - | - |
| 1 | 1 | 0 | 100 | L/H | H/L | DC\_new = 100 | CMP SR = EV  & Enable SR update | CMP SR = PRD  & Disable SR update |
| 1 | 1 | 100 | Y | H/L | DC\_y | DC\_new = Y | - | CMP SR =  DC\_RHS\_Y  & Enable SR update |
| 1 | 1 | 100 | 0 | H/L | L/H | DC\_new = 0 | CMP SR = EV  & Enable SR update | CMP SR = PRD  & Disable SR update |
| 1 | 1 | 100 | 100 | H/L | H/L | - | - | - |

Table .Action Table, No PWM Enable Reconfiguration & PWM Enabled

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **En\_old** | **En\_new** | **DC\_cur** | **DC\_new** | **PWM State**  **cur** | **PWM State**  **new** | **Latch** | **LHS prepare**  **Action** | **RHS prepare**  **action** |
| 0 | 1 | X | X | L/H | DC\_x | En\_new = 1 | CMP SR =  PRD−DC\_LHS\_X  & Enable SR update | - |
| 0 | 1 | X | Y | L/H | DC\_y | En\_new = 1,  DC\_new = Y | CMP SR =  PRD−DC\_LHS\_Y  & Enable SR update | - |
| 0 | 1 | X | 0 | L/H | L/H | En\_new = 1,  DC\_new = 0 | - | - |
| 0 | 1 | X | 100 | L/H | H/L | En\_new = 1,  DC\_new = 100 | CMP SR = EV  & Enable SR update | CMP SR = PRD  & Disable SR update |
| 0 | 1 | 0 | Y | L/H | DC\_y | En\_new = 1,  DC\_new = Y | CMP SR = PRD−DC\_LHS\_Y  & Enable SR update | - |
| 0 | 1 | 0 | 0 | L/H | L/H | En\_new = 1 | - | - |
| 0 | 1 | 0 | 100 | L/H | H/L | En\_new = 1,  DC\_new = 100 | CMP SR = EV  & Enable SR update | CMP SR = PRD  & Disable SR update |
| 0 | 1 | 100 | Y | L/H | DC\_y | En\_new = 1,  DC\_new = Y | CMP SR =  PRD−DC\_LHS\_Y  & Enable SR update | - |
| 0 | 1 | 100 | 0 | L/H | L/H | En\_new = 1,  DC\_new = 0 | - | - |
| 0 | 1 | 100 | 100 | L/H | H/L | En\_new = 1 | CMP SR = EV  & Enable SR update | CMP SR = PRD  & Disable SR update |

Table . Action Table, PWM Enable Reconfiguration & PWM Enable

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **En\_old** | **En\_new** | **DC\_cur** | **DC\_new** | **PWM State**  **Cur** | **PWM State**  **new** | **Latch** | **LHS prepare**  **Action** | **RHS prepare**  **action** |
| 1 | 0 | X | X | DC\_x | L/H | En\_new = 0 | CMP SR = PRD  & Disable SR update | - |
| 1 | 0 | X | Y | DC\_x | L/H | En\_new = 0,  DC\_new = Y | CMP SR = PRD  & Disable SR update | - |
| 1 | 0 | X | 0 | DC\_x | L/H | En\_new = 0,  DC\_new = 0 | CMP SR = PRD  & Disable SR update | - |
| 1 | 0 | X | 100 | DC\_x | L/H | En\_new = 0,  DC\_new = 100 | CMP SR = EV  & Enable SR update | - |
| 1 | 0 | 0 | Y | L/H | L/H | En\_new = 0,  DC\_new = Y | - | - |
| 1 | 0 | 0 | 0 | L/H | L/H | En\_new = 0 | - | - |
| 1 | 0 | 0 | 100 | L/H | L/H | En\_new = 0,  DC\_new = 100 | - | - |
| 1 | 0 | 100 | Y | H/L | L/H | En\_new = 0,  DC\_new = Y | CMP SR = EV  & Enable SR update | CMP SR = PRD  & Disable SR update |
| 1 | 0 | 100 | 0 | H/L | L/H | En\_new = 0,  DC\_new = 0 | CMP SR = EV  & Enable SR update | CMP SR = PRD  & Disable SR update |
| 1 | 0 | 100 | 100 | H/L | L/H | En\_new = 0 | CMP SR = EV  & Enable SR update | CMP SR = PRD  & Disable SR update |

Table . Action Table, PWM Enable Reconfiguration & PWM Disable

### Host & PRU Firmware Communication

#### Initialization

Host and PRU interaction for FW initialization is shown below in Figure 5.

The primary responsibilities of the Host for PRU FW initialization include: (1) configuring the IEP clock; (2) loading the FW PRU IMEM and DMEM images; (3) writing PRU FW registers with desired non-default values; and (4) initiating execution of the PRU FW. The FW registers are initialized to default values on load of PRU FW DMEM image (FW register default values are described below in Section 5.1), and any non-default values must be written after load of the FW DMEM image.

The PRU FW parses the information provided by the Host in the PRU FW registers and applies the requested initial configuration. The following parameters are only parsed (and associated configuration applied) at FW initialization time:

* PWM\_CTRL FW register, IEP0 & IEP1 Global Enable: IEP and PWM registers are only configured when the associated Global Enable bit is set. This allows the FW to use IEP0 PWMs, IEP1 PWMs, or IEP0 and IEP1 PWMs. Further, the IEP PWM FW image can be simultaneously executed from PRU0 and PRU1, provided the Global Enable settings are consistent (e.g. PRU0 sets IEP0 Global Enable, while PRU1 sets IEP1 Global Enable). The FW informs the Host concerning observed Global Enable settings using the PWM\_STAT Global Enable Acknowledge bits.
* IEPx\_PWM\_MODE FW register, IEP0 & IEP1 Mode: Each POS/NEG PWM signal pair is configured as 2 Single-Ended PWMs or 1 Complementary PWM.

The FW informs the Host that FW initialization is complete by setting the PWM\_STAT FW\_INIT flag bit. Upon observing this bit, the Host can issue commands to dynamically reconfigure PWMs.

The PWM\_CTRL and PWM\_STAT bits are shown below in Figure 6. FW Register Bits for .



Figure . Firmware Initialization



Figure . FW Register Bits for FW Initialization

#### Reconfiguration

The Host can dynamically reconfigure all PWM parameters except IEP0 & IEP1 Global Enable and IEP0 & IEP1 PWM Mode. PWM parameter reconfiguration is achieved as follows:

1. Host writes updated parameter value to FW register
2. Host triggers reconfiguration by setting associated reconfiguration request bit in IEPx\_PWM\_RECFG.

For example, to reconfigure the IEP0 PWM Period, the Host:

1. Writes new PWM period to IEP0\_PWM\_PRD\_COUNT
2. Writes ‘1’ IEP0\_PWM\_RECFG: RECFG\_IEP0\_PWM\_PRD\_COUNT.

The PRU FW services reconfiguration requests on PWM period boundaries in the LHS\_RECFG (Host) reconfiguration State Machine state. The FW State Machine is described below in Section 5.3.4.

The IEPx\_PWM\_RECFG bits are shown below in Figure 7.



Figure . FW Register Bits for FW Reconfiguration

PWM parameter (re-)configuration is summarized in Table 23.

|  |  |  |
| --- | --- | --- |
| **Configuration Parameter** | **Initialization, Dynamic** | **Affects** |
| IEP*x*\_GBL\_EN | Initialization | All PWMs in 2 IEP PWM sets |
| IEP*x*\_PWM\_MODE [5:0] | Initialization | Each bit: 2 SNGL / 1 CMPL PWM |
| IEP*x*\_PWM\_EN [11:0] | Dynamic | Each bit: 1 SNGL / 1 CMPL PWM |
| IEP*x*\_PWM\_PRD\_COUNT | Dynamic | All PWMs in 2 IEP PWM sets |
| IEP*x*\_PWM*m*\_DC\_COUNT | Dynamic | 1 SNGL / 1 CMPL PWM |
| IEP*x*\_PWM*n*\_*n*+1\_DB\_COUNT | Dynamic | 1 CMPL PWM |

Table . PWM Parameter Configurability

### Firmware State Machine

After One-Time Initialization and Application of the Initial Configuration (see Figure 5), the FW begins execution of the State Machine (SM). The SM includes five states, details of which are presented in Table 24. A diagram of the SM is presented in Figure 8, and detailed flow charts for all SM states are presented in Figure 9 - Figure 21.

|  |  |  |
| --- | --- | --- |
| **State** | **Event Trigger** | **Description** |
| INIT | IEPx CMP0 | Initialize IEPx CMP Shadow Registers for PWM initialization. Set LHS/RHS actions according to configuration in FW registers & Initialization Action Tables. |
| LHS | IEPx CMP0 | Write LHS DC values to CMP Shadow Registers for enabled SNGL / CMPL PWMs. |
| LHS\_RECFG | None, executes after LHS | Check for Host reconfiguration requests. For all reconfiguration requests:   * Determine LHS & RHS actions using Execution Action Tables * Execute LHS actions * Stash RHS actions.   Note the SM transitions to the INIT state in case of a PWM Period reconfiguration request. |
| RHS | IEPx CMP0 | Write RHS DC values to CMP Shadow Registers for enabled SNGL / CMPL PWMs. |
| RHS\_RECFG | None, executes after RHS | Execute stashed (pending) RHS actions determined in LHS\_RECFG state. |

Table . State Machine State Descriptions



Figure . Firmware State Machine



Figure . SM INIT State, Top-Level Flow Chart



Figure . SM INIT State, Initialize SNGL/CMPL PWM Flow Charts



Figure . SM INIT State, Initialize SPWM Flow Chart



Figure . SM LHS State, Top-Level Flow Chart



Figure . SM RHS State, Top-Level Flow Chart



Figure . SM LHS\_RECFG State, Top-Level Flow Chart



Figure . SM LHS\_RECFG State, Reconfigure Period Flow Chart



Figure . SM LHS\_RECFG State, Reconfigure PWM Enable or DC Flow Chart



Figure . SM LHS\_RECFG State, Reconfigure DB Flow Chart



Figure . SM LHS\_RECFG State, Finalize Period Reconfiguration Flow Chart



Figure . SM RHS\_RECFG State, Top-Level Flow Chart



Figure . SM RHS\_RECFG State, Execute RHS Stash for SNGL PWM Flow Chart



Figure . SM RHS\_RECFG State, Execute RHS Stash for CMPL PWM Flow Chart

## PRU Resource Usage

The FW is written entirely in C and no effort has been invested in optimizing the code to reduce PRU cycle or memory requirements.

### PRU Cycles

Not Measured

### PRU Memory

FW memory requirements are shown in Table 25. Memory types not listed in the table (i.e. Scratch-Pad Memory and Shared Memory) are not used by the FW.

|  |  |
| --- | --- |
| **Memory Type** | **Required Memory (Bytes)** |
| IMEM | 7412 |
| DMEM | 1072 |

Table . FW Memory Usage

## Firmware Source Code

Table 26 lists the firmware source code files.

|  |  |
| --- | --- |
| **File** | **Description** |
| **AM654x\_PRU.cmd** | Linker command file for firmware build |
| **icssg\_iep\_pwm.h** | Header file, contains Host API FW register definitions |
| **iepPwm.c** | Contains code for State Machine and supporting functions |
| **iepPwm.h** | Header file, contains definitions for State Machine and supporting functions |
| **iepPwmFwRegs.c** | Contains code for FW default register settings |
| **iepPwmFwRegs.h** | Header file, contains FW register definitions |
| **iepPwmHwRegs.h** | Header file, contains hardware register definitions |
| **main.c** | Contains main function with top-level flow |

Table . Firmware Source Files

# ICSSG Release Target

## EVM Platform

The PWM firmware was developed on the AM654x EVM based on AM6548 SR1.0 silicon.

### AM654x EVM + Breakout Board

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ICSSG** | **PRU** | **Instance** | **PWM Set** | **PRU GPIO Pin** | **PRU PWM Pin** | **BB Port** | **BB Pin** |
| ICSSG1 | PRU0 | IEP0 PWM | 0 | PRG1\_PRU0\_GPO12 | PRG1\_PWM0\_A0 | J8 | 5 |
| 0 | PRG1\_PRU0\_GPO13 | PRG1\_PWM0\_B0 | J8 | 7 |
| 0 | PRG1\_PRU0\_GPO14 | PRG1\_PWM0\_A1 | J8 | 9 |
| 0 | PRG1\_PRU0\_GPO15 | PRG1\_PWM0\_B1 | J8 | 11 |
| 0 | PRG1\_PRU0\_GPO16 | PRG1\_PWM0\_A2 | J8 | 13 |
| 0 | PRG1\_PRU0\_GPO17 | PRG1\_PWM0\_B2 | J8 | 15 |
| 1 | PRG1\_PRU1\_GPO12 | PRG1\_PWM1\_A0 | J10 | 5 |
| 1 | PRG1\_PRU1\_GPO13 | PRG1\_PWM1\_B0 | J10 | 7 |
| 1 | PRG1\_PRU1\_GPO14 | PRG1\_PWM1\_A1 | J10 | 9 |
| 1 | PRG1\_PRU1\_GPO15 | PRG1\_PWM1\_B1 | J10 | 11 |
| 1 | PRG1\_PRU1\_GPO16 | PRG1\_PWM1\_A2 | J10 | 13 |
| 1 | PRG1\_PRU1\_GPO17 | PRG1\_PWM1\_B2 | J10 | 15 |
| IEP1 PWM | 2 | PRG1\_PRU0\_GPO2 | PRG1\_PWM2\_A0 | J7 | 5 |
| 2 | PRG1\_PRU0\_GPO4 | PRG1\_PWM2\_B0 | J7 | 7 |
| 2 | PRG1\_PRU0\_GPO8 | PRG1\_PWM2\_A1 | J7 | 9 |
| 2 | PRG1\_PRU0\_GPO10 | PRG1\_PWM2\_B1 | J8 | 1 |
| 2 | PRG1\_PRU1\_GPO2 | PRG1\_PWM2\_A2 | J9 | 5 |
| 2 | PRG1\_PRU1\_GPO4 | PRG1\_PWM2\_B2 | J9 | 9 |
| 3 | PRG1\_PRU0\_GPO0 | PRG1\_PWM3\_A0 | J7 | 1 |
| 3 | PRG1\_PRU0\_GPO1 | PRG1\_PWM3\_B0 | J7 | 3 |
| 3 | PRG1\_PRU0\_GPO6 | PRG1\_PWM3\_A1 | J7 | 13 |
| 3 | PRG1\_PRU0\_GPO7 | PRG1\_PWM3\_B1 | J7 | 15 |
| 3 | PRG1\_PRU0\_GPO3 | PRG1\_PWM3\_A2 | J7 | 7 |
| 3 | PRG1\_PRU0\_GPO5 | PRG1\_PWM3\_B2 | J7 | 11 |

Table . AM654x PWMs