# Module –

# High-Level Description

This module implements functionality with respect to ES-34B ePWM. This module implements the ePWM-related register initialization and the motor control and ADC SOCA configuration update subfunctions.

# Figures

## Component Diagram

# Variable Data Dictionary

For details on module input / output variable, refer to the Data Dictionary for the application. Input / output variable names are listed here for reference.

|  |  |  |
| --- | --- | --- |
| Module Inputs | Module Outputs | |
| PWMPeriod\_Cnt\_u16 | | ePWM1CMPA\_Cnt\_u16 |
| DCPhsAComp\_Cnt\_u16 | | ePWM1CMPB\_Cnt\_u16 |
| DCPhsBComp\_Cnt\_u16 | | ePWM2CMPA\_Cnt\_u16 |
| DCPhsCComp\_Cnt\_u16 | | ePWM2CMPB\_Cnt\_u16 |
| ePWM4CMPB\_Cnt\_u16 | | ePWM3CMPA\_Cnt\_u16 |
|  | | ePWM3CMPB\_Cnt\_u16 |
|  | | ePWM4CMPA\_Cnt\_u16 |
|  | | ePWM4CMPB\_Cnt\_u16 |

## Module Internal Variables

This section identifies the name, range and resolutions for module specific data created by this module. If there are no range restrictions on the variable, the term “FULL” is placed into the table for legal range.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Variable Name | Resolution | Legal Range  (min) | Legal Range  (max) | Software Segment |
| None |  |  |  |  |

### User defined typedef definition/declaration

This section documents any user types uniquely used for the module.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Typedef Name | Element Name | User Defined Type | Legal Range  (min) | Legal Range  (max) |

# Constant Data Dictionary

## Calibration Constants

This section lists the calibrations used by the module. For details on calibration constants, refer to the Data Dictionary for the application.

|  |
| --- |
| Constant Name |
| k\_ADCTrig1Offset\_Cnt\_s16 |
| k\_PwmDeadBand\_Cnt\_u16  k\_PwmRelay\_Cnt\_u16 |

## Program(fixed) Constants

### Embedded Constants

All embedded constants whose values are provided in Eng units will be evaluated to the equivalent counts by using the FPM\_InitFixedPoint\_m() macro within the #define statement.

#### Local

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Constant Name | Resolution | Legal Range  (min) | Legal Range  (max) | Value |
| D\_DUTYCYCLESHIFT\_CNT\_U16 | 1 | 535U | 535U | 535U |

#### Global

This section lists the global constants used by the module. For details on global constants, refer to the Data Dictionary for the application.

|  |
| --- |
| Constant Name |
| None |

### Module specific Lookup Tables Constants

|  |  |  |  |
| --- | --- | --- | --- |
| Constant Name | Resolution | Value | Software Segment |
| None |  |  |  |

# Functions/Macros used by the Sub-Modules

## Library Functions / Macros

The library and functions / Macros that are called by the various sub modules are identified below,

1. ePWM\_Read\_PWMPeriod\_u16
2. ePWM\_Read\_DCPhsAComp\_u16
3. ePWM\_Read\_DCPhsBComp\_u16
4. ePWM\_Read\_DCPhsCComp\_u16
5. ePWM\_Write\_ePWM1CMPA\_Cnt\_u16
6. ePWM\_Write\_ePWM1CMPB\_Cnt\_u16
7. ePWM\_Write\_ePWM2CMPA\_Cnt\_u16
8. ePWM\_Write\_ePWM2CMPB\_Cnt\_u16
9. ePWM\_Write\_ePWM3CMPA\_Cnt\_u16
10. ePWM\_Write\_ePWM3CMPB\_Cnt\_u16
11. ePWM\_Write\_ePWM4CMPA\_Cnt\_u16
12. ePWM\_Write\_ePWM4CMPB\_Cnt\_u16

## Data Hiding Functions

1. None

## Global Functions/Macros Defined by this Module

### Global Functions #1

(For detailed info regarding values assigned to registers refer Reference Pdf attached below)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Function Name** | ePWM\_Init1 | Type | Min | Max | UTP Tol. |
| **Arguments Passed** | None |  |  |  |  |
|  |  |  |  |  |  |
| **Return Value** | None |  |  |  |  |

#### Description

**EPWM1**



**EPWM2**



**EPWM3**



**EPWM4**



**EPWM7**



### Global Functions #2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Function Name** | ePWM\_Per1 | Type | Min | Max | UTP Tol. |
| **Arguments Passed** | None |  |  |  |  |
|  |  |  |  |  |  |
| **Return Value** | None |  |  |  |  |

#### Design Rationale

PWM Period computed by Pwm\_Cdd is read and adjusted. Adjusted value is used for computations of the motor control PWM CMPA and CMPB values, and for the ADC SOCA update (ePWM4CMPA).

ADC SOCB update is done in the 2ms loop (see integration manual) then value (ePWM4CMPB) is read by this function and written to the appropriate buffer.

This function is called by the motor control ISR and so data cannot be transferred via the RTE; therefore data is read and written via macros and global variables.

Due to timing constraints of the motor control ISR processing, outputs are not limited to their defined ranges before writing.

#### Store Module Inputs to Local copies

ePWM\_Read\_PWMPeriod\_u16(&PWMPeriod\_Cnt\_T\_u16)

ePWM\_Read\_DCPhsAComp\_u16(&DCPhsAComp\_Cnt\_T\_u16)

ePWM\_Read\_DCPhsBComp\_u16(&DCPhsBComp\_Cnt\_T\_u16)

ePWM\_Read\_DCPhsCComp\_u16(&DCPhsCComp\_Cnt\_T\_u16)

ePWM\_Read\_ePWM4CMPB\_Cnt\_u16(&ePWM4CMPB\_Cnt\_T\_u16)

#### Processing



#### Store Local copy of outputs into Module Outputs

ePWM\_Write\_ePWM1CMPA\_Cnt\_u16(ePWM1CompA\_Cnt\_T\_u16);

ePWM\_Write\_ePWM1CMPB\_Cnt\_u16(ePWM1CompB\_Cnt\_T\_u16);

ePWM\_Write\_ePWM2CMPA\_Cnt\_u16(ePWM2CompA\_Cnt\_T\_u16);

ePWM\_Write\_ePWM2CMPB\_Cnt\_u16(ePWM2CompB\_Cnt\_T\_u16);

ePWM\_Write\_ePWM3CMPA\_Cnt\_u16(ePWM3CompA\_Cnt\_T\_u16);

ePWM\_Write\_ePWM3CMPB\_Cnt\_u16(ePWM3CompB\_Cnt\_T\_u16);

ePWM\_Write\_ePWM4CMPA\_Cnt\_u16((uint16)(((sint16)halfPWMPeriod\_Cnt\_T\_u16) - k\_ADCTrig1Offset\_Cnt\_s16 - D\_DUTYCYCLESHIFT\_CNT\_U16));

ePWM\_Write\_ePWM4CMPB\_Cnt\_u16(ePWM4CMPB\_Cnt\_T\_u16);

## Local Functions/Macros Used by this MDD only

### Local Macro #1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Function Name** | ePWM\_DisableOutputs | Type | Min | Max | UTP Tol. |
| **Arguments Passed** | None |  |  |  |  |
|  |  |  |  |  |  |
| **Return Value** | None |  |  |  |  |

#### Description

Disables the motor control EPWM outputs; forces A and B outputs low on all three motor control ePWMs (ePWM1, ePWM2, ePWM3) (Refer the included register reference for more details of register)

ePWM1->AQCSFRC = 5U

ePWM1->DBCTL &= 0xFFFFFFFCU

ePWM2->AQCSFRC = 5U

ePWM2->DBCTL &= 0xFFFFFFFCU

ePWM3->AQCSFRC = 5U

ePWM3->DBCTL &= 0xFFFFFFFCU

### Local Macro #2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Function Name** | ePWM\_EnableOutputs | Type | Min | Max | UTP Tol. |
| **Arguments Passed** | None |  |  |  |  |
|  |  |  |  |  |  |
| **Return Value** | None |  |  |  |  |

#### Description

Enables the motor control EPWM outputs; allows A and B outputs to be driven high/low as configured in register initialization (core initialization subfunction) and using the CMPA and CMPB values computed in the motor control configuration subfunction. (Refer the included register reference for more details of register)

ePWM1->AQCSFRC = 0U;

ePWM1->DBCTL |= 3U;

ePWM2->AQCSFRC = 0U;

ePWM2->DBCTL |= 3U;

ePWM3->AQCSFRC = 0U;

ePWM3->DBCTL |= 3U;

# Software Module Implementation

## Runtime Environment (RTE) Initial Values

This section lists the initial values of data written by this module but controlled by the RTE. After RTE initialization, the data in this table will contain these values.

|  |  |
| --- | --- |
| Data | Value |
| None |  |

## Initialization Functions

### Init:

#### Design Rationale

This component does not have an RTE Init function, however ePWM pin mux settings are configured in the integration project and initialized in (patched) generated code. See the integration manual.

#### Module Outputs

None

#### Module Internal

None

#### Initialize EPWM Direction Register

## Periodic Functions

None

## Fault Recovery Functions

None

## Shutdown Functions

None

## Interrupt Functions

None

## Serial Communication Functions

None

## Transition Functions

None

# Execution Requirements

## Execution Rates for sub-modules called by the Subroutine

This table serves as reference for the Scheduler design

|  |  |  |
| --- | --- | --- |
| Global Function Name | Calling Frequency | Function in which the function is called |
| ePWM\_Init1 | On Event | ECU start up |
| ePWM\_Per1 | On Event | Motor Control ISR subroutine |

## Execution Requirements for Serial Communication Functions

|  |  |
| --- | --- |
| Function Name | Sub-Module called by (Serial Comm Function Name) |
| None |  |

# Memory Map Definition Requirements

## Sub Modules (Functions)

This table identifies the software segments for functions identified in this module.

|  |  |
| --- | --- |
| Name of Sub Module | Software Segment |
| ePWM\_Init1 | EPWM\_START\_SEC\_CODE |
| ePWM\_Per1 | EPWM\_START\_SEC\_CODE |

## Local Functions

This table identifies the software segments for local functions identified in this module.

|  |  |
| --- | --- |
| Name of Sub Module | Software Segment |
| None |  |

# Known Issues / Limitations With Design

1. QAC for the generated files not corrected.
2. In the FDD DataDictionary full range is used for the signals ‘PWMDutyCycleA\_Cnts’, ‘PWMDutyCycleB\_Cnts’ and ‘PWMDutyCycleC\_Cnts’ which is not correct. Corrected range (i.e. 0 to 6000 given by the FDD author) is used in the software DataDictionary.
3. In the FDD DataDictionary the max value used for signal ’PWM\_Period ‘ is not correct. Corrected max value (i.e. 6000 given by the FDD author) is uded in the software DataDictionary.
4. Unit test consideration: Don’t use any combination of the input signals where the value of ‘PWMDutyCycleA\_Cnts’/’PWMDutyCycleB\_Cnts’/’ PWMDutyCycleC\_Cnts’ is greater than the signal ‘PWM\_Period’.

# Reference

Register Reference



# Revision Control Log

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev #** | **Change Description** | **Date** | **Author Initials** |
| 1.0 | Initial Version ( FDD 34B) –Shutdown Mech not included | 18-Feb-13 | Selva |
| 2 | Anomaly 4605 – changed active state of ePWM modules 1-3 | 11-Mar-13 | OT |
| 3 | Updated to FDD 34B v003 | 18-Jun-13 | OT/Selva |
| 4 | Updated to FDD 34B v005 | 07-Apr-14 | Selva |
| 5 | Updated to ES-34B v008 | 25-Jan-15 | KMC |
| 6 | Updated to ES-34B v009 | 25-Nov-15 | Rijvi |