User Manual

for S32K3 GDU Driver

Document Number: UM34GDUASRR21-11 Rev0000R3.0.0 Rev. 1.0

Chapter 1

Revision History

Revision	Date	Author	Description
1.0	31.03.2023	NXP RTD Team	S32K3 Real-Time Drivers AUTOSAR 4.4 & R21-11 Version 3.0.0

Chapter 2

Introduction

- Supported Derivatives

- Overview

- About This Manual

- Acronyms and Definitions

- Reference List

This integration manual describes the integration requirements for GDU CDD Driver for S32K3 microcontrollers.

2.1 Supported Derivatives

The software described in this document is intended to be used with the following microcontroller devices of NXP Semiconductors:

- s32k310_mqfp100

- s32k310_lqfp48

- s32k311_mqfp100 / MWCT2015S_mqfp100

- s32k311_lqfp48

- s32k312_mqfp100 / MWCT2016S_mqfp100

- s32k312_mqfp172 / MWCT2016S_mqfp172

- s32k314_mqfp172

- s32k314_mapbga257

- s32k322_mqfp100 / MWCT2D16S_mqfp100

- s32k322_mqfp172 / MWCT2D16S_mqfp172

- s32k324_mqfp172 / MWCT2D17S_mqfp172

- s32k324_mapbga257

- s32k341_mqfp100

- s32k341_mqfp172

- s32k342_mqfp100

- s32k342_mqfp172

- s32k344_mqfp172

- s32k344_mapbga257

- s32k394_mapbga289

- s32k396_mapbga289

- s32k358_mqfp172

- s32k358_mapbga289

- s32k328_mqfp172

- s32k328_mapbga289

- s32k338_mqfp172

- s32k338_mapbga289

- s32k348_mqfp172

- s32k348_mapbga289

- s32m274_lqfp64

- s32m276_lqfp64

All of the above microcontroller devices are collectively named as S32K3.

Note: MWCT part numbers contain NXP confidential IP for Qi Wireless Power.

2.2 Overview

AUTOSAR (AUTomotive Open System ARchitecture) is an industry partnership working to establish standards for software interfaces and software modules for automobile electronic control systems.

AUTOSAR:

- paves the way for innovative electronic systems that further improve performance, safety and environmental friendliness.

- is a strong global partnership that creates one common standard: "Cooperate on standards, compete on implementation".

- is a key enabling technology to manage the growing electrics/electronics complexity. It aims to be prepared for the upcoming technologies and to improve cost-efficiency without making any compromise with respect to quality.

- facilitates the exchange and update of software and hardware over the service life of the vehicle.

2.3   About This Manual

This Technical Reference employs the following typographical conventions:

- **Boldface** style: Used for important terms, notes and warnings.

- *Italic* style: Used for code snippets in the text. Note that C language modifiers such "const" or "volatile" are sometimes omitted to improve readability of the presented code.

Notes and warnings are shown as below:

Note

    This is a note.

Warning

    This is a warning

2.4 Acronyms and Definitions

Term	Definition
API	Application Programming Interface
ASM	Assembler
BSMI	Basic Software Make file Interface
C/CPP	C and C++ Source Code
CDD	Complex Device Driver
DEM	Diagnostic Event Manager
DET	Development Error Tracer
ECU	Electronic Control Unit
LSB	Least Signifigant Bit
MCU	Micro Controller Unit
MIDE	Multi Integrated Development Environment
MRC	Memory Region Controller
MSB	Most Significant Bit
N/A	Not Applicable
RAM	Random Access Memory
GDU	Gate Drive Unit
SIU	Systems Integration Unit
SWS	Software Specification
XML	Extensible Markup Language

2.5 Reference List

#	Title	Version
1	S32M27x Reference Manual	Rev. 2, Draft A, 2/2023
2	S32M2xx Data Sheet	Rev. 2 RC, 12/2022

Chapter 3

Driver

3.1 Requirements

Gate Drive Unit is a Complex Device Driver (CDD), so there are no AUTOSAR requirements regarding this module.

It has vendor-specific requirements and implementation.

3.2 Driver Design Summary

The GDU module provides pre-drivers to control a three-phase DC motor via external FETs. To support this control, it includes a charge pump and a boost converter.

3.3 Hardware Resources

#	Hardware IP	Description
1	GDU	Gate Drive Unit

3.4 Deviations from Requirements

Since this is a CDD Module, there are no AUTOSAR requirements for the functionality.

3.5 Driver Limitations

- None

3.6 Driver usage and configuration tips

The GDU IP does not require any initialization (it works right out of reset with the default values found in the reference manual), but in order to use the GDU driver you must initialize it first. There is no deinitialization API, but the Init function allows reinitializations.

When IMPLEMENTATION_CONFIG_VARIANT is configured as VariantPreCompile and there is no more than one variant defined, Gdu_Init:

- expects a null pointer as an argument

- extracts the configuration structure from the first element of Gdu_ConfigVariantPredefined, which must have external linkage.
  Else, Gdu_Init accepts a pointer to a configuration structure (of type Gdu_ConfigType) as an argument.

Before configuring the GDU, Gdu_Init:

- enables safe state, cfg mode and register writes

- disables the boost converter and the charge pump to allow their configuration

After configuring the GDU, Gdu_Init:

- enables the charge pump

- enables the boost converter if the user enabled it

- disables safe state, cfg mode and register writes

3.6.0.1 Delay measurement

In order to measure the delay between the edge of the input PWM signal and the transition of the corresponding output gate, Gdu_StartDelayMeasurement can be used to start a new measurement on a certain PWM channel. The result is to be obtained by the user with a subsequent call to Gdu_ReadDelayMeasurement.

3.6.0.2 Slew rate waveform

GDU supports configuration of the slew rate with three zones for every transition, as shown in the Reference Manual, Figure 324.

3.6.1 HLD usage

Prior usage of the GDU CDD in an application, the configuration files must be generated with the configurator.

3.6.2 LLD usage

GDU doesn't support IP layer with EB tresos.

3.7 Runtime errors

The driver does not trigger any DEM runtime errors, but triggers the runtime DET errors listed in the table below:

Function	Error Code	Condition triggering the error
- Gdu_SetMode - Gdu_SetSafeState - Gdu_StartDelayMeasurement - Gdu_ReadDelayMeasurement - Gdu_AdjustSlewRate - Gdu_SwIrefTrimming - Gdu_OffStateDiagnostics	GDU_DET_ERROR_↩ UNINITIALIZED	The operation cannot be performed while the driver is uninitialized.
- Gdu_Init - Gdu_SetMode - Gdu_SetSafeState - Gdu_StartDelayMeasurement - Gdu_ReadDelayMeasurement - Gdu_AdjustSlewRate - Gdu_SwIrefTrimming - Gdu_GetVersionInfo	GDU_DET_ERROR_↩ PARAMETER	The provided arguments are not valid.

3.8 Symbolic Names Disclaimer

All containers having symbolicNameValue set to TRUE in the AUTOSAR schema will generate defines like:

#define <Mip>Conf_<Container_ShortName>_<Container_ID>

For this reason it is forbidden to duplicate the names of such containers across the RTD configurations or to use names that may trigger other compile issues (e.g. match existing `#ifdefs` arguments).

Chapter 4

Tresos Configuration Plug-in

This chapter describes the Tresos configuration plug-in for the driver. All the parameters are described below.

- Module Gdu
    - Container GduGeneral
        * Parameter GduVersionInfoApi
        * Parameter GduDevErrorDetect
        * Parameter GduNotification
        * Container GduCtl
            · Parameter GduBoostConverterEn
            · Parameter GduCustomIrefTrimming
        * Container GduIntEn
            · Parameter GduHighVoltageIE
            · Parameter GduDesaturationHs0IE
            · Parameter GduDesaturationHs1IE
            · Parameter GduDesaturationHs2IE
            · Parameter GduDesaturationLs0IE
            · Parameter GduDesaturationLs1IE
            · Parameter GduDesaturationLs2IE
        * Container GduCfg
            · Parameter GduSyncPwmClock
            · Parameter GduVoltageDividerFactor
            · Parameter GduOvervoltageThreshold
        * Container GduEaCfg
            · Parameter GduDesaturationAction
            · Parameter GduOvercurrentAction
            · Parameter GduOvervoltageAction
        * Container GduBoostCfg
            · Parameter GduBoostCoilCurrent
            · Parameter GduBoostClockDivider
            · Parameter GduBoostDutyCyle
        * Container GduBtCfg
            · Parameter GduBlankingTime

* Container GduHssrOn
  · Parameter GduHsTurnOnTicks1
  · Parameter GduHsTurnOnTicks2
  · Parameter GduHsTurnOnSlew1
  · Parameter GduHsTurnOnSlew2
  · Parameter GduHsTurnOnSlew3
* Container GduHssrOff
  · Parameter GduHsTurnOffTicks1
  · Parameter GduHsTurnOffTicks2
  · Parameter GduHsTurnOffSlew1
  · Parameter GduHsTurnOffSlew2
  · Parameter GduHsTurnOffSlew3
* Container GduLssrOn
  · Parameter GduLsTurnOnTicks1
  · Parameter GduLsTurnOnTicks2
  · Parameter GduLsTurnOnSlew1
  · Parameter GduLsTurnOnSlew2
  · Parameter GduLsTurnOnSlew3
* Container GduLssrOff
  · Parameter GduLsTurnOffTicks1
  · Parameter GduLsTurnOffTicks2
  · Parameter GduLsTurnOffSlew1
  · Parameter GduLsTurnOffSlew2
  · Parameter GduLsTurnOffSlew3
* Container GduDsCfg
  · Parameter GduDesaturationFilterHs
  · Parameter GduDesaturationFilterLs
  · Parameter GduDesaturationLevelHs
  · Parameter GduDesaturationLevelLs
* Container GduCpCfg
  · Parameter GduChargePumpDischarge
  · Parameter GduChargePumpVcpOn
  · Parameter GduChargePumpClockDivider
* Container GduSupCfg
  · Parameter GduDriveLsFunctionalReset
* Container GduIrt
  · Parameter GduIrefTrimmingHg0
  · Parameter GduIrefTrimmingLg0
  · Parameter GduIrefTrimmingHg1
  · Parameter GduIrefTrimmingLg1
  · Parameter GduIrefTrimmingHg2
  · Parameter GduIrefTrimmingLg2
− Container CommonPublishedInformation
  * Parameter ArReleaseMajorVersion
  * Parameter ArReleaseMinorVersion
  * Parameter ArReleaseRevisionVersion
  * Parameter ModuleId

∗ Parameter SwMajorVersion

∗ Parameter SwMinorVersion

∗ Parameter SwPatchVersion

∗ Parameter VendorApiInfix

∗ Parameter VendorId

4.1 Module Gdu

Vendor specific: Configuration of the Gdu module.

Included containers:

- GduGeneral

- CommonPublishedInformation

Property	Value
type	ECUC-MODULE-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantSupport	true
supportedConfigVariants	VARIANT-POST-BUILD, VARIANT-PRE-COMPILE

4.2 Container GduGeneral

Vendor specific: Configuration of general Gdu parameters.

Included subcontainers:

- GduCtl

- GduIntEn

- GduCfg

- GduEaCfg

- GduBoostCfg

- GduBtCfg

- GduHssrOn

- GduHssrOff

- GduLssrOn

- [GduLssrOff](#)
- [GduDsCfg](#)
- [GduCpCfg](#)
- [GduSupCfg](#)
- [GduIrt](#)

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.3 Parameter GduVersionInfoApi

Toggles the availability of Gdu_GetVersionInfo API.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	AUTOSAR_ECUC
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.4 Parameter GduDevErrorDetect

Vendor specific:

Switches the Development Error Detection and Notification on or off.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	true

4.5   Parameter GduNotification

Function pointer to callback function.

Prototype:

void Gdu_Callback(uint8 Event);

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-FUNCTION-NAME-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: PRE-COMPILE
defaultValue	NULL_PTR

4.6   Container GduCtl

Vendor specific: Controls operation of GDU.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.7   Parameter GduBoostConverterEn

Vendor specific:

Enable the boost converter circuit.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.8   Parameter GduCustomIrefTrimming

Vendor specific:

Enables writing of Iref trimming values (IRT). When disabled, GDU loads device-specific trimming values into the Iref trimming fields and does not allow software to do so. When enabled, GDU does not load the trimming fields and allows software to trim the Iref current. IRT is also proctected by RWEN . In other words, for software to trim the Iref current, both RWEN and IRTSW must be 1.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.9   Container GduIntEn

Vendor specific: Enables assertion of interrupts when errors set the corresponding interrupt flags.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.10   Parameter GduHighVoltageIE

Vendor specific:

Enables the high HD interrupt (INTF[HDHVDIF]).

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.11   Parameter GduDesaturationHs0IE

Vendor specific:

Enables the desaturation error interrupt (INTF[DHSIF0]) on the high-side FET of phase 0.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.12   Parameter GduDesaturationHs1IE

Vendor specific:

Enables the desaturation error interrupt (INTF[DHSIF1]) on the high-side FET of phase 1.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.13   Parameter GduDesaturationHs2IE

Vendor specific:

Enables the desaturation error interrupt (INTF[DHSIF2]) on the high-side FET of phase 2.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.14   Parameter GduDesaturationLs0IE

Vendor specific:

Enables the desaturation error interrupt (INTF[DLSIF0]) on the low-side FET of phase 0.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.15   Parameter GduDesaturationLs1IE

Vendor specific:

Enables the desaturation error interrupt (INTF[DLSIF1]) on the low-side FET of phase 1.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.16   Parameter GduDesaturationLs2IE

Vendor specific:

Enables the desaturation error interrupt (INTF[DLSIF2]) on the low-side FET of phase 2.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.17 Container GduCfg

Vendor specific: Configures driver operation.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.18 Parameter GduSyncPwmClock

Vendor specific:

Enables PWM synchronization to the module clock.

true: Enabled.

false: Disabled.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.19   Parameter GduVoltageDividerFactor

Vendor specific:

Selects the voltage divider factor on HD and HSx to measure them by the ADC. For the exact values of the divider factors, see the data sheet electrical specs.

0b - GDU_IP_DIVIDER_LOW. If the MCU operating supply is 3.3V, divide by ADIVlow3p3. If the MCU operating supply is 5V, divide by ADIVlow5p0.

1b - GDU_IP_DIVIDER_HIGH. If the MCU operating supply is 3.3V, divide by ADIVhigh3p3. If the MCU operating supply is 5V, divide by ADIVhigh5p0.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	False
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_IP_DIVIDER_LOW
literals	['GDU_IP_DIVIDER_LOW', 'GDU_IP_DIVIDER_HIGH']

4.20   Parameter GduOvervoltageThreshold

Vendor specific:

Selects the voltage thresholds of overvoltage detection on the HD pin. For the exact value of the threshold, see the data sheet electrical specs.

0b - GDU_IP_VOLTAGE_LOW voltage. Selects VHVHDLA and VHVHDLD.

1b - GDU_IP_VOLTAGE_HIGH voltage. Selects VHVHDHA and VHVHDHD.

Note: Implementation Specific Parameter.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	False
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_IP_VOLTAGE_LOW
literals	['GDU_IP_VOLTAGE_LOW', 'GDU_IP_VOLTAGE_HIGH']

4.21   Container GduEaCfg

Vendor specific: Configures the GDU reaction to the different error events.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.22   Parameter GduDesaturationAction

Vendor specific:

Selects the action in response to a desaturation error on any FET.

true: Turn off all FETs.

false: Turn off only the desaturated FET.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.23   Parameter GduOvercurrentAction

Vendor specific:

Selects the action in response to too much current out of the motor control phases on any external FET.

true: Low-side gate drivers off.

false: Low-side gate drivers on.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.24   Parameter GduOvervoltageAction

Vendor specific:

Selects the action in response to an overvoltage fault or a safe-state request from AEC. The following conditions cause an overvoltage fault:.

Overvoltage on the HD pin.

Overvoltage from the supply (faults coming from PMC).

true: Turn off all low-side gate drivers.

false: Turn on all low-side gate drivers.

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	false

4.25   Container GduBoostCfg

Vendor specific: Configures boost operation.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.26 Parameter GduBoostCoilCurrent

Vendor specific:

Specifies an adjustment to the boost coil current limit ICOIL0,16 on the BST pin.

GDU_ICOIL_0: Boost coil current limit (lowest setting).

...

GDU_ICOIL_7: Boost coil current limit (highest setting).

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_ICOIL_0
literals	['GDU_ICOIL_0', 'GDU_ICOIL_1', 'GDU_ICOIL_2', 'GDU_ICOIL_3', 'GDU_ICOIL_4', 'GDU_ICOIL_5', 'GDU_ICOIL_6', 'GDU_ICOIL_7']

4.27 Parameter GduBoostClockDivider

Vendor specific:

Specifies the clock divider applied to the module clock to obtain the boost converter clock.

Desired boost clock frequency fBOOST, f_module = 42 MHz

Clock division

BOOSTCFG[BOCD]

62.5 KHz

k = 672

6

125 KHz

k = 336

5

250 KHz

k = 168

4

500 KHz

k = 84

3

1 MHz

k = 42

2

2 MHz

k = 21

1

2.5 MHz

k = 16

0

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	6
min	0

4.28   Parameter GduBoostDutyCyle

Vendor specific:

Selects the duty cycle of the boost converter clock.

GDU_DUTY_CYCLE_1_DIV_4: 25%.

GDU_DUTY_CYCLE_1_DIV_2: 50%.

GDU_DUTY_CYCLE_3_DIV_4: 75%.

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_DUTY_CYCLE_1_DIV_2
literals	['GDU_DUTY_CYCLE_1_DIV_4', 'GDU_DUTY_CYCLE_1_DIV_2', 'GDU_DUTY_CYCLE_3_DIV_4']

4.29   Container GduBtCfg

Vendor specific: Specifes the blanking time.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.30   Parameter GduBlankingTime

Vendor specific:

Specifies the blanking time for hiding desaturation errors according to this equation:

tBLANK = BT * module clock cycles.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	511
min	0

4.31   Container GduHssrOn

Vendor specific: Configures the high-side slew rate.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.32   Parameter GduHsTurnOnTicks1

Vendor specific:

Specifies the time, in module clock cycles, when GDU sets the gate current for the high-side drivers to the reference specified in High Side Turn On Current Time Point 1

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	127
min	0

4.33   Parameter GduHsTurnOnTicks2

Vendor specific:

Specifies the time, in module clock cycles, when GDU sets the gate current for the high-side drivers to the reference specified in High Side Turn On Current Time Point 2

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	127
min	0

4.34   Parameter GduHsTurnOnSlew1

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.35   Parameter GduHsTurnOnSlew2

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.36   Parameter GduHsTurnOnSlew3

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.37   Container GduHssrOff

Vendor specific: Configures the high-side slew rate.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.38   Parameter GduHsTurnOffTicks1

Vendor specific:

Specifies the time, in module clock cycles, when GDU sets the gate current for the high-side drivers to the reference specified in High Side Turn Off Current Time Point 1

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	127
min	0

4.39   Parameter GduHsTurnOffTicks2

Vendor specific:

Specifies the time, in module clock cycles, when GDU sets the gate current for the high-side drivers to the reference specified in High Side Turn Off Current Time Point 2

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	127
min	0

4.40   Parameter GduHsTurnOffSlew1

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.41   Parameter GduHsTurnOffSlew2

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.42   Parameter GduHsTurnOffSlew3

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.43   Container GduLssrOn

Vendor specific: Configures the low-side slew rate.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.44   Parameter GduLsTurnOnTicks1

Vendor specific:

Specifies the time, in module clock cycles, when GDU sets the gate current for the low-side drivers to the reference specified in Low Side Turn On Current Time Point 1

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	127
min	0

4.45   Parameter GduLsTurnOnTicks2

Vendor specific:

Specifies the time, in module clock cycles, when GDU sets the gate current for the low-side drivers to the reference specified in Low Side Turn On Current Time Point 2

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	127
min	0

4.46   Parameter GduLsTurnOnSlew1

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.47 Parameter GduLsTurnOnSlew2

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.48 Parameter GduLsTurnOnSlew3

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4
max	31
min	0

4.49 Container GduLssrOff

Vendor specific: Configures the low-side slew rate.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.50 Parameter GduLsTurnOffTicks1

Vendor specific:

Specifies the time, in module clock cycles, when GDU sets the gate current for the low-side drivers to the reference specified in Low Side Turn Off Current Time Point 1

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP

Property	Value
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	127
min	0

4.51   Parameter GduLsTurnOffTicks2

Vendor specific:

Specifies the time, in module clock cycles, when GDU sets the gate current for the low-side drivers to the reference specified in Low Side Turn Off Current Time Point 2

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	127
min	0

4.52   Parameter GduLsTurnOffSlew1

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.53   Parameter GduLsTurnOffSlew2

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	0
max	31
min	0

4.54   Parameter GduLsTurnOffSlew3

Vendor specific:

Specifies the slew, from 0 (minimum) to 31 (maximum).

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	4
max	31
min	0

4.55 Container GduDsCfg

Vendor specific: Specifies the desaturation level adjustement.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.56 Parameter GduDesaturationFilterHs

Vendor specific:

Specifies the desaturation filter characteristic adjustment for the three high-side FET gate drivers.

GDU_DS_FILTER_200ns

GDU_DS_FILTER_600ns

GDU_DS_FILTER_1000ns

GDU_DS_FILTER_1400ns

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_DS_FILTER_200ns
literals	['GDU_DS_FILTER_200ns', 'GDU_DS_FILTER_600ns', 'GDU_DS_↵ FILTER_1000ns', 'GDU_DS_FILTER_1400ns']

4.57   Parameter GduDesaturationFilterLs

Vendor specific:

Specifies the desaturation filter characteristic adjustment for the three low-side FET gate drivers.

GDU_DS_FILTER_200ns

GDU_DS_FILTER_600ns

GDU_DS_FILTER_1000ns

GDU_DS_FILTER_1400ns

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_DS_FILTER_200ns
literals	['GDU_DS_FILTER_200ns', 'GDU_DS_FILTER_600ns', 'GDU_DS_↵ FILTER_1000ns', 'GDU_DS_FILTER_1400ns']

4.58   Parameter GduDesaturationLevelHs

Vendor specific:

Specifies the desaturation level adjustment (V desaths) for the three high-side FET gate drivers.

GDU_DS_LEVEL_150mV

GDU_DS_LEVEL_250mV

GDU_DS_LEVEL_350mV

GDU_DS_LEVEL_450mV

GDU_DS_LEVEL_700mV

GDU_DS_LEVEL_950mV

GDU_DS_LEVEL_1200mV

GDU_DS_LEVEL_1450mV

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_DS_LEVEL_150mV
literals	['GDU_DS_LEVEL_150mV', 'GDU_DS_LEVEL_250mV', 'GDU_DS_↵ LEVEL_350mV', 'GDU_DS_LEVEL_450mV', 'GDU_DS_LEVEL_700mV', 'GDU_DS_LEVEL_950mV', 'GDU_DS_LEVEL_1200mV', 'GDU_DS_↵ LEVEL_1450mV']

4.59   Parameter GduDesaturationLevelLs

Vendor specific:

Specifies the desaturation level adjustment (V desaths) for the three low-side FET gate drivers.

GDU_DS_LEVEL_150mV

GDU_DS_LEVEL_250mV

GDU_DS_LEVEL_350mV

GDU_DS_LEVEL_450mV

GDU_DS_LEVEL_700mV

GDU_DS_LEVEL_950mV

GDU_DS_LEVEL_1200mV

GDU_DS_LEVEL_1450mV

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_DS_LEVEL_150mV
literals	['GDU_DS_LEVEL_150mV', 'GDU_DS_LEVEL_250mV', 'GDU_DS_↵LEVEL_350mV', 'GDU_DS_LEVEL_450mV', 'GDU_DS_LEVEL_700mV', 'GDU_DS_LEVEL_950mV', 'GDU_DS_LEVEL_1200mV', 'GDU_DS_↵LEVEL_1450mV']

4.60   Container GduCpCfg

Vendor specific: Configures the charge pump.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.61   Parameter GduChargePumpDischarge

Vendor specific:

Specifies the desaturation filter characteristic adjustment for the three high-side FET gate drivers.

GDU_CP_DISCHARGE_250ns

GDU_CP_DISCHARGE_500ns

GDU_CP_DISCHARGE_750ns

GDU_DS_FILTER_1000ns

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_CP_DISCHARGE_500ns
literals	['GDU_CP_DISCHARGE_250ns', 'GDU_CP_DISCHARGE_500ns', 'GDU↩_CP_DISCHARGE_750ns', 'GDU_DS_FILTER_1000ns']

4.62   Parameter GduChargePumpVcpOn

Vendor specific:

Specifies the number of module clock cycles to wait before closing the switch that connects the charge pump voltage (VCP) to VBSx.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

Property	Value
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	1
max	63
min	0

4.63   Parameter GduChargePumpClockDivider

Vendor specific:

Specifies the half period, in module clock cycles, of the charge pump clock (fCP), according to this equation:

Tcharge_pump = 2 x (CPCD x Tmodule_clock)

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	167
max	1023
min	0

4.64   Container GduSupCfg

Vendor specific: Specifies the state of the low-side driver on functional reset.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF

Property	Value
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.65 Parameter GduDriveLsFunctionalReset

Vendor specific:

Selects the state of the low-side driver upon functional reset. This field configuration applies during functional reset and when (CTL[SSTEN] = 1).

A POR turns off the high-side and low-side drivers (SUPCFG[SUP] = 1).

If you turn startup on (write 0 to SUPCFG[SUP]), then the next functional reset turns off the high side but leaves the low side turned on with a slew rate value of 00100b.

false: ON

true: OFF

Property	Value
type	ECUC-BOOLEAN-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	true

4.66 Container GduIrt

Vendor specific: Specifies the trimming percentage to apply to the 10 ?A reference current for the the corresponding gate drivers. GDU loads these values from OTP at startup. For each trimming field, 100% (011b) should provide 10 ?A at nominal process.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.67   Parameter GduIrefTrimmingHg0

Vendor specific:

Trims the reference current of the high-side gate driver HG0.

GDU_IP_IREF_TRIM_000: 62.5% (minimum current).

GDU_IP_IREF_TRIM_001: 75%.

GDU_IP_IREF_TRIM_010: 87.5%.

GDU_IP_IREF_TRIM_011: 100% (10 ?A at nominal process).

GDU_IP_IREF_TRIM_100: 112.5%.

GDU_IP_IREF_TRIM_101: 125%.

GDU_IP_IREF_TRIM_110: 137.5%.

GDU_IP_IREF_TRIM_111: 150% (maximum current).

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_IP_IREF_TRIM_000
literals	['GDU_IP_IREF_TRIM_000', 'GDU_IP_IREF_TRIM_001', 'GDU_IP_IREF_TRIM_010', 'GDU_IP_IREF_TRIM_011', 'GDU_IP_IREF_TRIM_100', 'GDU_IP_IREF_TRIM_101', 'GDU_IP_IREF_TRIM_110', 'GDU_IP_IREF_TRIM_111']

4.68   Parameter GduIrefTrimmingLg0

Vendor specific:

Trims the reference current of the low-side gate driver LG0.

GDU_IP_IREF_TRIM_000: 62.5% (minimum current).

GDU_IP_IREF_TRIM_001: 75%.

GDU_IP_IREF_TRIM_010: 87.5%.

GDU_IP_IREF_TRIM_011: 100% (10 ?A at nominal process).

GDU_IP_IREF_TRIM_100: 112.5%.

GDU_IP_IREF_TRIM_101: 125%.

GDU_IP_IREF_TRIM_110: 137.5%.

GDU_IP_IREF_TRIM_111: 150% (maximum current).

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_IP_IREF_TRIM_000
literals	['GDU_IP_IREF_TRIM_000', 'GDU_IP_IREF_TRIM_001', 'GDU_IP_↩IREF_TRIM_010', 'GDU_IP_IREF_TRIM_011', 'GDU_IP_IREF_TRIM↩_100', 'GDU_IP_IREF_TRIM_101', 'GDU_IP_IREF_TRIM_110', 'GDU↩_IP_IREF_TRIM_111']

4.69   Parameter GduIrefTrimmingHg1

Vendor specific:

Trims the reference current of the high-side gate driver HG1.

GDU_IP_IREF_TRIM_000: 62.5% (minimum current).

GDU_IP_IREF_TRIM_001: 75%.

GDU_IP_IREF_TRIM_010: 87.5%.

GDU_IP_IREF_TRIM_011: 100% (10 ?A at nominal process).

GDU_IP_IREF_TRIM_100: 112.5%.

GDU_IP_IREF_TRIM_101: 125%.

GDU_IP_IREF_TRIM_110: 137.5%.

GDU_IP_IREF_TRIM_111: 150% (maximum current).

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_IP_IREF_TRIM_000
literals	['GDU_IP_IREF_TRIM_000', 'GDU_IP_IREF_TRIM_001', 'GDU_IP_↩IREF_TRIM_010', 'GDU_IP_IREF_TRIM_011', 'GDU_IP_IREF_TRIM↩_100', 'GDU_IP_IREF_TRIM_101', 'GDU_IP_IREF_TRIM_110', 'GDU↩_IP_IREF_TRIM_111']

4.70   Parameter GduIrefTrimmingLg1

Vendor specific:

Trims the reference current of the low-side gate driver LG1.

GDU_IP_IREF_TRIM_000: 62.5% (minimum current).

GDU_IP_IREF_TRIM_001: 75%.

GDU_IP_IREF_TRIM_010: 87.5%.

GDU_IP_IREF_TRIM_011: 100% (10 ?A at nominal process).

GDU_IP_IREF_TRIM_100: 112.5%.

GDU_IP_IREF_TRIM_101: 125%.

GDU_IP_IREF_TRIM_110: 137.5%.

GDU_IP_IREF_TRIM_111: 150% (maximum current).

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_IP_IREF_TRIM_000
literals	['GDU_IP_IREF_TRIM_000', 'GDU_IP_IREF_TRIM_001', 'GDU_IP_↩IREF_TRIM_010', 'GDU_IP_IREF_TRIM_011', 'GDU_IP_IREF_TRIM↩_100', 'GDU_IP_IREF_TRIM_101', 'GDU_IP_IREF_TRIM_110', 'GDU↩_IP_IREF_TRIM_111']

4.71   Parameter GduIrefTrimmingHg2

Vendor specific:

Trims the reference current of the high-side gate driver HG2.

GDU_IP_IREF_TRIM_000: 62.5% (minimum current).

GDU_IP_IREF_TRIM_001: 75%.

GDU_IP_IREF_TRIM_010: 87.5%.

GDU_IP_IREF_TRIM_011: 100% (10 ?A at nominal process).

GDU_IP_IREF_TRIM_100: 112.5%.

GDU_IP_IREF_TRIM_101: 125%.

GDU_IP_IREF_TRIM_110: 137.5%.

GDU_IP_IREF_TRIM_111: 150% (maximum current).

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A

Property	Value
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_IP_IREF_TRIM_000
literals	['GDU_IP_IREF_TRIM_000', 'GDU_IP_IREF_TRIM_001', 'GDU_IP_↵IREF_TRIM_010', 'GDU_IP_IREF_TRIM_011', 'GDU_IP_IREF_TRIM↵_100', 'GDU_IP_IREF_TRIM_101', 'GDU_IP_IREF_TRIM_110', 'GDU↵_IP_IREF_TRIM_111']

4.72   Parameter GduIrefTrimmingLg2

Vendor specific:

Trims the reference current of the low-side gate driver LG2.

GDU_IP_IREF_TRIM_000: 62.5% (minimum current).

GDU_IP_IREF_TRIM_001: 75%.

GDU_IP_IREF_TRIM_010: 87.5%.

GDU_IP_IREF_TRIM_011: 100% (10 ?A at nominal process).

GDU_IP_IREF_TRIM_100: 112.5%.

GDU_IP_IREF_TRIM_101: 125%.

GDU_IP_IREF_TRIM_110: 137.5%.

GDU_IP_IREF_TRIM_111: 150% (maximum current).

Property	Value
type	ECUC-ENUMERATION-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	true
valueConfigClasses	VARIANT-PRE-COMPILE: PRE-COMPILE
	VARIANT-POST-BUILD: POST-BUILD
defaultValue	GDU_IP_IREF_TRIM_000
literals	['GDU_IP_IREF_TRIM_000', 'GDU_IP_IREF_TRIM_001', 'GDU_IP_↵IREF_TRIM_010', 'GDU_IP_IREF_TRIM_011', 'GDU_IP_IREF_TRIM↵_100', 'GDU_IP_IREF_TRIM_101', 'GDU_IP_IREF_TRIM_110', 'GDU↵_IP_IREF_TRIM_111']

4.73   Container CommonPublishedInformation

Vendor specific:

Common container, aggregated by all modules. It contains published information about vendor and versions.

Included subcontainers:

- None

Property	Value
type	ECUC-PARAM-CONF-CONTAINER-DEF
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A

4.74   Parameter ArReleaseMajorVersion

Vendor specific:

Major version number of AUTOSAR specification on which the appropriate implementation is based on.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
defaultValue	4
max	4
min	4

4.75   Parameter ArReleaseMinorVersion

Vendor specific:

Minor version number of AUTOSAR specification on which the appropriate implementation is based on.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
defaultValue	7
max	7
min	7

4.76   Parameter ArReleaseRevisionVersion

Vendor specific:

Revision version number of AUTOSAR specification on which the appropriate implementation is based on.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0

4.77   Parameter ModuleId

Vendor specific:

Module ID of this module from Module List.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
defaultValue	123
max	123
min	123

4.78   Parameter SwMajorVersion

Vendor specific:

Major version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
defaultValue	3
max	3
min	3

4.79   Parameter SwMinorVersion

Vendor specific:

Minor version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0

4.80   Parameter SwPatchVersion

Vendor specific:

Patch level version number of the vendor specific implementation of the module. The numbering is vendor specific.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
defaultValue	0
max	0
min	0

4.81   Parameter VendorApiInfix

Vendor specific:

In driver modules which can be instantiated several times on a single ECU, BSW00347 requires

that the name of APIs is extended by the VendorId and a vendor specific name.

This parameter is used to specify the vendor specific name. In total, the

implementation specific name is generated as follows: <ModuleName>__>VendorId>__<VendorApiInfix>.

E.g. assuming that the VendorId of the implementor is 123 and the implementer chose a VendorApiInfix of "v11r456" a api name Can_Write defined in the SWS will translate to Can_123_v11r456Write.

This parameter is mandatory for all modules with upper multiplicity > 1. It shall not be used for modules with upper multiplicity =1.

Property	Value
type	ECUC-STRING-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	0
upperMultiplicity	1
postBuildVariantMultiplicity	false
multiplicityConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
defaultValue	

4.82   Parameter VendorId

Vendor specific:

Vendor ID of the dedicated implementation of this module according to the AUTOSAR vendor list.

Property	Value
type	ECUC-INTEGER-PARAM-DEF
origin	NXP
symbolicNameValue	false
lowerMultiplicity	1
upperMultiplicity	1
postBuildVariantMultiplicity	N/A
multiplicityConfigClasses	N/A
postBuildVariantValue	false
valueConfigClasses	VARIANT-PRE-COMPILE: PUBLISHED-INFORMATION
	VARIANT-POST-BUILD: PUBLISHED-INFORMATION
defaultValue	43
max	43
min	43

Chapter 5

Module Index

5.1 Software Specification

Here is a list of all modules:

Chapter 6

Module Documentation

6.1 CDD_GDU

6.1.1 Detailed Description

Macros

- #define GDU_RET_ERROR
- #define GDU_RET_INVALID
- #define GDU_RET_DET_PARAM
- #define GDU_RET_DET_UNINIT
- #define GDU_DET_SERVICE_INITIALIZATION
- #define GDU_DET_SERVICE_SETMODE
- #define GDU_DET_SERVICE_SETSAFESTATE
- #define GDU_DET_SERVICE_GETSTATUS
- #define GDU_DET_SERVICE_START_DELAYMEAS
- #define GDU_DET_SERVICE_READ_DELAYMEAS
- #define GDU_DET_SERVICE_ADJUSTSLEWRATE
- #define GDU_DET_SERVICE_SWIREFTRIMMING
- #define GDU_DET_SERVICE_OFFSTATEDIAG
- #define GDU_DET_ERROR_PARAMETER
- #define GDU_DET_ERROR_UNINITIALIZED

Function Reference

- Std_ReturnType Gdu_Init (const Gdu_Ip_ConfigType ∗Configuration)

  GDU initialization API.
- Std_ReturnType Gdu_SetMode (Gdu_Ip_ModeType Mode)

  Control the operation mode (enable/disable).
- Std_ReturnType Gdu_SetSafeState (Gdu_Ip_SafeStateType SafeState)

  Control the safe state (on/off).
- Std_ReturnType Gdu_GetStatus (Gdu_Ip_ErrorStatusType ∗ErrorStatus)

> *Get the status of the error conditions present in the INTF register.*

- Std_ReturnType Gdu_StartDelayMeasurement (Gdu_Ip_PwmChannelType PwmChannel, Gdu_Ip_EdgeType Edge)

    Start the delay measurement between the PWM edge and the feedback signal coming from the FET gate driver.

- Std_ReturnType Gdu_ReadDelayMeasurement (uint8 ∗Delay)

    Read the delay measurement results.

- Std_ReturnType Gdu_AdjustSlewRate (Gdu_Ip_SlewRateSelectType SlewRateSelect, const Gdu_Ip_SlewRateConfigTy ∗SlewRateConfig)

    Adjust the slew rate of the FET gate drivers.

- Std_ReturnType Gdu_SwIrefTrimming (Gdu_Ip_IrefTrimmingType IrefTrimming)

    Configure the Iref trimming percentages.

- Std_ReturnType Gdu_OffStateDiagnostics (Gdu_Ip_OffStateDiagType OffStateDiagnostics)

    Connect the HS pins towards HD or VSS in order to detect open loads, loads shorted to HD, or loads shorted to GND.

6.1.2 Macro Definition Documentation

6.1.2.1 GDU_RET_ERROR

```
#define GDU_RET_ERROR
```

Enumerators defined to be used with Std_ReturnType (see SWS_Std_00011). Their meaning is detailed in the description of each API. Make sure to not use b7 and b6, as they are reserved by the RTE. Return status code

Definition at line 121 of file CDD_Gdu.h.

6.1.2.2 GDU_RET_INVALID

```
#define GDU_RET_INVALID
```

Invalid status code

Definition at line 122 of file CDD_Gdu.h.

6.1.2.3 GDU_RET_DET_PARAM

```
#define GDU_RET_DET_PARAM
```

Return status code

Definition at line 123 of file CDD_Gdu.h.

6.1.2.4 GDU_RET_DET_UNINIT

#define GDU_RET_DET_UNINIT

Return status code

Definition at line 124 of file CDD_Gdu.h.

6.1.2.5 GDU_DET_SERVICE_INITIALIZATION

#define GDU_DET_SERVICE_INITIALIZATION

Service IDs for Det error reporting Gdu_Init

Definition at line 127 of file CDD_Gdu.h.

6.1.2.6 GDU_DET_SERVICE_SETMODE

#define GDU_DET_SERVICE_SETMODE

Gdu_SetMode

Definition at line 128 of file CDD_Gdu.h.

6.1.2.7 GDU_DET_SERVICE_SETSAFESTATE

#define GDU_DET_SERVICE_SETSAFESTATE

Gdu_SetSafeState

Definition at line 129 of file CDD_Gdu.h.

6.1.2.8   GDU_DET_SERVICE_GETSTATUS

`#define GDU_DET_SERVICE_GETSTATUS`

Gdu_GetStatus

Definition at line 130 of file CDD_Gdu.h.

6.1.2.9   GDU_DET_SERVICE_START_DELAYMEAS

`#define GDU_DET_SERVICE_START_DELAYMEAS`

Gdu_StartDelayMeasurement

Definition at line 131 of file CDD_Gdu.h.

6.1.2.10   GDU_DET_SERVICE_READ_DELAYMEAS

`#define GDU_DET_SERVICE_READ_DELAYMEAS`

Gdu_ReadDelayMeasurement

Definition at line 132 of file CDD_Gdu.h.

6.1.2.11   GDU_DET_SERVICE_ADJUSTSLEWRATE

`#define GDU_DET_SERVICE_ADJUSTSLEWRATE`

Gdu_AdjustSlewRate

Definition at line 133 of file CDD_Gdu.h.

6.1.2.12   GDU_DET_SERVICE_SWIREFTRIMMING

```
#define GDU_DET_SERVICE_SWIREFTRIMMING
```

Gdu_SwIrefTrimming

Definition at line 134 of file CDD_Gdu.h.

6.1.2.13   GDU_DET_SERVICE_OFFSTATEDIAG

```
#define GDU_DET_SERVICE_OFFSTATEDIAG
```

Gdu_OffStateDiagnostics

Definition at line 135 of file CDD_Gdu.h.

6.1.2.14   GDU_DET_ERROR_PARAMETER

```
#define GDU_DET_ERROR_PARAMETER
```

Error codes for Det error reporting The provided arguments are not valid.

Definition at line 141 of file CDD_Gdu.h.

6.1.2.15   GDU_DET_ERROR_UNINITIALIZED

```
#define GDU_DET_ERROR_UNINITIALIZED
```

The operation cannot be performed while the driver is uninitialized.

Definition at line 142 of file CDD_Gdu.h.

6.1.3   Function Reference

6.1.3.1   Gdu_Init()

```
Std_ReturnType Gdu_Init (
            const Gdu_Ip_ConfigType * Configuration )
```

GDU initialization API.
Call this API to configure the driver.

Parameters

in	*Configuration*	Contains the address of the configuration structure.

Returns

the status of the command.

Return values

E_OK	command processed successfully
E_NOT_OK	failure from the IP layer
GDU_RET_DET_PARAM	one of the provided arguments is invalid: error reported to Det

Precondition

The 6 PWM lines routed to this module are OFF.

6.1.3.2 Gdu_SetMode()

```
Std_ReturnType Gdu_SetMode (
            Gdu_Ip_ModeType Mode )
```

Control the operation mode (enable/disable).

Following the recommendations from the manual, charge pump [CPEN] follows the mode of the GDU [GDUEN].

Parameters

in	*Mode*	specifies the desired mode

Returns

the status of the command.

Return values

E_OK	command processed successfully
E_NOT_OK	failure from the IP layer
GDU_RET_DET_UNINIT	API called before driver initialization: error reported to Det
GDU_RET_DET_PARAM	one of the provided arguments is invalid: error reported to Det

Precondition

CDD_GDU is initialized.

6.1.3.3 Gdu_SetSafeState()

```
Std_ReturnType Gdu_SetSafeState (
            Gdu_Ip_SafeStateType SafeState )
```

Control the safe state (on/off).

Call this API to enable safe state before entering low power mode.

Parameters

in	*SafeState*	specifies the desired safe state

Returns

the status of the command.

Return values

E_OK	command processed successfully
E_NOT_OK	failure from the IP layer
GDU_RET_DET_UNINIT	API called before driver initialization: error reported to Det
GDU_RET_DET_PARAM	one of the provided arguments is invalid: error reported to Det

Precondition

CDD_GDU is initialized.

6.1.3.4 Gdu_GetStatus()

```
Std_ReturnType Gdu_GetStatus (
            Gdu_Ip_ErrorStatusType * ErrorStatus )
```
Get the status of the error conditions present in the INTF register.

Parameters

out	*ErrorStatus*	structure containing bit-fields that indicate if an error is present or not The user needs to create and pass an address to this type of structure only if they are interested in the status of each error condition; otherwise, the overall return status may be just enough.

Returns

the status of the command.

Return values

E_OK	no error conditions are active at the moment
E_NOT_OK	failure from the IP layer, *ErrorStatus was not written
GDU_RET_ERROR	one of the error conditions is active
GDU_RET_DET_UNINIT	API called before driver initialization: error reported to Det

Precondition

CDD_GDU is initialized.

6.1.3.5 Gdu_StartDelayMeasurement()

```
Std_ReturnType Gdu_StartDelayMeasurement (
```

```
            Gdu_Ip_PwmChannelType PwmChannel,
            Gdu_Ip_EdgeType Edge )
```
Start the delay measurement between the PWM edge and the feedback signal coming from the FET gate driver.

Parameters

in	*PwmChannel*	the measurement is performed on a specific PWM channel
in	*Edge*	the measurement is performed between two edges of the signal: rising or falling

Returns

the status of the command.

Return values

E_OK	command processed successfully
E_NOT_OK	failure from the IP layer
GDU_RET_DET_UNINIT	API called before driver initialization: error reported to Det
GDU_RET_DET_PARAM	one of the provided arguments is invalid: error reported to Det

Precondition

CDD_GDU is initialized.

Make sure that the previous measurement had time to either finish or expire, because: "Writting to DLYMCFG while delay measurement is enabled can corrupt the delay measurement."

6.1.3.6 Gdu_ReadDelayMeasurement()

```
Std_ReturnType Gdu_ReadDelayMeasurement (
            uint8 * Delay )
```
Read the delay measurement results.

Parameters

out	*Delay*	delay measured in clock cycles

Returns

the status of the command.

Return values

E_OK	command processed successfully
E_NOT_OK	failure from the IP layer
GDU_RET_ERROR	measurement result is unavailable or corrupted
GDU_RET_DET_UNINIT	API called before driver initialization: error reported to Det
GDU_RET_DET_PARAM	one of the provided arguments is invalid: error reported to Det

Precondition

CDD_GDU is initialized.

6.1.3.7 Gdu_AdjustSlewRate()

```
Std_ReturnType Gdu_AdjustSlewRate (
            Gdu_Ip_SlewRateSelectType SlewRateSelect,
            const Gdu_Ip_SlewRateConfigType * SlewRateConfig )
```
Adjust the slew rate of the FET gate drivers.

Parameters

in	*SlewRateSelect*	selects the HS/LS FETs, and the turn on / turn off waveforms
in	*SlewRateConfig*	pointer to a structure containing the configuration values

Returns

the status of the command.

Return values

E_OK	command processed successfully
E_NOT_OK	failure from the IP layer
GDU_RET_DET_UNINIT	API called before driver initialization: error reported to Det
GDU_RET_DET_PARAM	one of the provided arguments is invalid: error reported to Det

Precondition

CDD_GDU is initialized.

Stop the PWMs inputs.

Put the GDU in safe state before making big changes to these values.

6.1.3.8 Gdu_SwIrefTrimming()

```
Std_ReturnType Gdu_SwIrefTrimming (
            Gdu_Ip_IrefTrimmingType IrefTrimming )
```
Configure the Iref trimming percentages.

Parameters

in	*IrefTrimming*	structure containing trimming percentages for each of the six gates

Returns

the status of the command.

Return values

E_OK	command processed successfully

Return values

E_NOT_OK	failure from the IP layer
GDU_RET_DET_UNINIT	API called before driver initialization: error reported to Det
GDU_RET_DET_PARAM	one of the provided arguments is invalid: error reported to Det

Precondition

> CDD_GDU is initialized.

6.1.3.9 Gdu_OffStateDiagnostics()

```
Std_ReturnType Gdu_OffStateDiagnostics (
          Gdu_Ip_OffStateDiagType OffStateDiagnostics )
```
Connect the HS pins towards HD or VSS in order to detect open loads, loads shorted to HD, or loads shorted to GND.
Beware that pulling the same pin to both HD and VSS may cause a short circuit.

Parameters

`in`	*OffStateDiagnostics*	structure containing boolean settings for each FET, and each direction

Returns

> the status of the command.

Return values

E_OK	command processed successfully
E_NOT_OK	failure from the IP layer
GDU_RET_DET_UNINIT	API called before driver initialization: error reported to Det

Precondition

> CDD_GDU is initialized.

6.2   IPV_GDU

6.2.1   Detailed Description

Data Structures

- struct Gdu_Ip_ErrorStatusType
- struct Gdu_Ip_SlewRateConfigType
- struct Gdu_Ip_OffStateDiagType

Macros

- #define GDU_IP_REG_SIZE_8b
- #define GDU_IP_REG_SIZE_16b

Module Documentation

- #define GDU_IP_REG_SIZE_32b
- #define GDU_IP_REG_INTF
- #define GDU_IP_REG_INTEN
- #define GDU_IP_REG_STAT
- #define GDU_IP_REG_CTL
- #define GDU_IP_REG_CFG
- #define GDU_IP_REG_EACFG
- #define GDU_IP_REG_BOOSTCFG
- #define GDU_IP_REG_BTCFG
- #define GDU_IP_REG_HSSRON
- #define GDU_IP_REG_HSSROFF
- #define GDU_IP_REG_LSSRON
- #define GDU_IP_REG_LSSROFF
- #define GDU_IP_REG_OFFSDCFG
- #define GDU_IP_REG_DSCFG
- #define GDU_IP_REG_CPCFG
- #define GDU_IP_REG_DLYMR
- #define GDU_IP_REG_DLYMCFG
- #define GDU_IP_REG_SUPCFG
- #define GDU_IP_REG_IRT

Types Reference

- typedef GDU_AE_Type Gdu_Ip_ConfigType
- typedef void(∗ Gdu_Ip_NotificationType) (uint32 InterruptFlags)

Enum Reference

- enum Gdu_Ip_StatusType
- enum Gdu_Ip_ModeType
- enum Gdu_Ip_SafeStateType
- enum Gdu_Ip_PwmChannelType
- enum Gdu_Ip_EdgeType
- enum Gdu_Ip_SlewRateSelectType
- enum Gdu_Ip_IrefTrimType

Function Reference

- Gdu_Ip_StatusType Gdu_Ip_Init (const Gdu_Ip_ConfigType ∗const Configuration)

  IPV_GDU initialization API.
- Gdu_Ip_StatusType Gdu_Ip_SetMode (Gdu_Ip_ModeType Mode)

  Control the operation mode (enable/disable).
- Gdu_Ip_StatusType Gdu_Ip_SetSafeState (Gdu_Ip_SafeStateType SafeState)

  Control the safe state (on/off).
- Gdu_Ip_StatusType Gdu_Ip_GetStatus (Gdu_Ip_ErrorStatusType ∗ErrorStatus)

  Get the status of the error conditions present in the INTF register.
- Gdu_Ip_StatusType Gdu_Ip_StartDelayMeasurement (Gdu_Ip_PwmChannelType PwmChannel, Gdu_Ip_EdgeType Edge)

  Start the delay measurement between the PWM edge and the feedback signal coming from the FET gate driver.
- Gdu_Ip_StatusType Gdu_Ip_ReadDelayMeasurement (uint8 ∗Delay)

  Read the delay measurement results.

- Gdu_Ip_StatusType   Gdu_Ip_AdjustSlewRate   (Gdu_Ip_SlewRateSelectType   SlewRateSelect,   const Gdu_Ip_SlewRateConfigType ∗SlewRateConfig)

    Adjust the slew rate of the FET gate drivers.
- Gdu_Ip_StatusType Gdu_Ip_SwIrefTrimming (Gdu_Ip_IrefTrimmingType IrefTrimming)

    Configure the Iref trimming percentages.
- Gdu_Ip_StatusType Gdu_Ip_OffStateDiagnostics (Gdu_Ip_OffStateDiagType OffStateDiagnostics)

    Connect the HS pins towards HD or GND in order to detect open loads, loads shorted to HD, or loads shorted to GND.
- void Gdu_Ip_ISR (void)

    ISR for GDU_INT.
- ISR (Gdu_Ip_Irq_ISR)

    ISR for GDU_INT.

6.2.2   Data Structure Documentation

6.2.2.1   struct Gdu_Ip_ErrorStatusType

Structure containing bit-fields for each of the possible error conditions found in the INTF register.
Is it ok to define single-bit fields as boolean? It's important that they do not have a signed bit: boolean is defined in PlatformTypes.h as either unsigned char or bool, so the following extract from the C99 standard applies if the members are defined as bool: "If the value 0 or 1 is stored into a nonzero-width bit-field of type _Bool, the value of the bit-field shall compare equal to the value stored."
Definition at line 243 of file Gdu_Ip_Types.h.

6.2.2.2   struct Gdu_Ip_SlewRateConfigType

Structure type that maps to one of the registers below; they contain configuration values for the slew rate of the gate driver signals that correspond to:

- HSSRON: FETs on the high side (valid for the turn on part)

- HSSROFF: FETs on the high side (valid for the turn off part)

- LSSRON: FETs on the low side (valid for the turn on part)

- LSSROFF: FETs on the low side (valid for the turn off part)

Definition at line 261 of file Gdu_Ip_Types.h.

6.2.2.3   struct Gdu_Ip_OffStateDiagType

Structure containing off state diagnostics settings
Definition at line 281 of file Gdu_Ip_Types.h.

Data Fields

Type	Name	Description
boolean	PullHs0Down: 1	Pull down HS0
boolean	PullHs1Down: 1	Pull down HS1
boolean	PullHs2Down: 1	Pull down HS2
boolean	PullHs0Up: 1	Pull up HS0

Data Fields

Type	Name	Description
boolean	PullHs1Up: 1	Pull up HS1
boolean	PullHs2Up: 1	Pull up HS2

6.2.3 Macro Definition Documentation

6.2.3.1 GDU_IP_REG_SIZE_8b

`#define GDU_IP_REG_SIZE_8b`
Data transfer size for an 8-bit SPI transaction [in bits]
Definition at line 102 of file Gdu_Ip.h.

6.2.3.2 GDU_IP_REG_SIZE_16b

`#define GDU_IP_REG_SIZE_16b`
Data transfer size for a 16-bit SPI transaction [in bits]
Definition at line 103 of file Gdu_Ip.h.

6.2.3.3 GDU_IP_REG_SIZE_32b

`#define GDU_IP_REG_SIZE_32b`
Data transfer size for a 32-bit SPI transaction [in bits]
Definition at line 104 of file Gdu_Ip.h.

6.2.3.4 GDU_IP_REG_INTF

`#define GDU_IP_REG_INTF`
Offset between IP_GDU_AE_BASE and INTF

Definition at line 106 of file Gdu_Ip.h.

6.2.3.5 GDU_IP_REG_INTEN

`#define GDU_IP_REG_INTEN`
Offset between IP_GDU_AE_BASE and INTEN

Definition at line 107 of file Gdu_Ip.h.

6.2.3.6 GDU_IP_REG_STAT

`#define GDU_IP_REG_STAT`
Offset between IP_GDU_AE_BASE and STAT

Definition at line 108 of file Gdu_Ip.h.

6.2.3.7   GDU_IP_REG_CTL

`#define GDU_IP_REG_CTL`
Offset between IP_GDU_AE_BASE and CTL

Definition at line 109 of file Gdu_Ip.h.

6.2.3.8   GDU_IP_REG_CFG

`#define GDU_IP_REG_CFG`
Offset between IP_GDU_AE_BASE and CFG

Definition at line 110 of file Gdu_Ip.h.

6.2.3.9   GDU_IP_REG_EACFG

`#define GDU_IP_REG_EACFG`
Offset between IP_GDU_AE_BASE and EACFG

Definition at line 111 of file Gdu_Ip.h.

6.2.3.10   GDU_IP_REG_BOOSTCFG

`#define GDU_IP_REG_BOOSTCFG`
Offset between IP_GDU_AE_BASE and BOOSTCFG
Definition at line 112 of file Gdu_Ip.h.

6.2.3.11   GDU_IP_REG_BTCFG

`#define GDU_IP_REG_BTCFG`
Offset between IP_GDU_AE_BASE and BTCFG

Definition at line 113 of file Gdu_Ip.h.

6.2.3.12   GDU_IP_REG_HSSRON

`#define GDU_IP_REG_HSSRON`
Offset between IP_GDU_AE_BASE and HSSRON

Definition at line 114 of file Gdu_Ip.h.

6.2.3.13   GDU_IP_REG_HSSROFF

`#define GDU_IP_REG_HSSROFF`
Offset between IP_GDU_AE_BASE and HSSROFF

Definition at line 115 of file Gdu_Ip.h.

6.2.3.14 GDU_IP_REG_LSSRON

`#define GDU_IP_REG_LSSRON`
Offset between IP_GDU_AE_BASE and LSSRON

Definition at line 116 of file Gdu_Ip.h.

6.2.3.15 GDU_IP_REG_LSSROFF

`#define GDU_IP_REG_LSSROFF`
Offset between IP_GDU_AE_BASE and LSSROFF

Definition at line 117 of file Gdu_Ip.h.

6.2.3.16 GDU_IP_REG_OFFSDCFG

`#define GDU_IP_REG_OFFSDCFG`
Offset between IP_GDU_AE_BASE and OFFSDCFG
Definition at line 118 of file Gdu_Ip.h.

6.2.3.17 GDU_IP_REG_DSCFG

`#define GDU_IP_REG_DSCFG`
Offset between IP_GDU_AE_BASE and DSCFG

Definition at line 119 of file Gdu_Ip.h.

6.2.3.18 GDU_IP_REG_CPCFG

`#define GDU_IP_REG_CPCFG`
Offset between IP_GDU_AE_BASE and CPCFG

Definition at line 120 of file Gdu_Ip.h.

6.2.3.19 GDU_IP_REG_DLYMR

`#define GDU_IP_REG_DLYMR`
Offset between IP_GDU_AE_BASE and DLYMR

Definition at line 121 of file Gdu_Ip.h.

6.2.3.20 GDU_IP_REG_DLYMCFG

`#define GDU_IP_REG_DLYMCFG`
Offset between IP_GDU_AE_BASE and DLYMCFG

Definition at line 122 of file Gdu_Ip.h.

6.2.3.21 GDU_IP_REG_SUPCFG

`#define GDU_IP_REG_SUPCFG`
Offset between IP_GDU_AE_BASE and SUPCFG

Definition at line 123 of file Gdu_Ip.h.

6.2.3.22 GDU_IP_REG_IRT

`#define GDU_IP_REG_IRT`
Offset between IP_GDU_AE_BASE and IRT

Definition at line 124 of file Gdu_Ip.h.

6.2.4 Types Reference

6.2.4.1 Gdu_Ip_ConfigType

`typedef GDU_AE_Type Gdu_Ip_ConfigType`
The configuration structure follows the layout of the register set structure
Definition at line 227 of file Gdu_Ip_Types.h.

6.2.4.2 Gdu_Ip_NotificationType

`typedef void(* Gdu_Ip_NotificationType) (uint32 InterruptFlags)`
Type definition for the notification function
Definition at line 230 of file Gdu_Ip_Types.h.

6.2.5 Enum Reference

6.2.5.1 Gdu_Ip_StatusType

`enum Gdu_Ip_StatusType`
The significance of each return code is explained in the function descriptions
Definition at line 102 of file Gdu_Ip_Types.h.

6.2.5.2 Gdu_Ip_ModeType

`enum Gdu_Ip_ModeType`
Controls GDUEN

Enumerator

GDU_IP_MODE_MIN	lower boundary for consecutive enumerators
GDU_IP_MODE_ENABLE	GDU is enabled
GDU_IP_MODE_DISABLE	GDU is disabled
GDU_IP_MODE_MAX	upper boundary for consecutive enumerators

Definition at line 111 of file Gdu_Ip_Types.h.

6.2.5.3 Gdu_Ip_SafeStateType

enum `Gdu_Ip_SafeStateType`
Controls SSTEN

Enumerator

GDU_IP_SAFESTATE_MIN	lower boundary for consecutive enumerators
GDU_IP_SAFESTATE_ON	put the GDU in safe state
GDU_IP_SAFESTATE_OFF	pull the GDU out from safe state
GDU_IP_SAFESTATE_MAX	upper boundary for consecutive enumerators

Definition at line 120 of file Gdu_Ip_Types.h.

6.2.5.4 Gdu_Ip_PwmChannelType

enum `Gdu_Ip_PwmChannelType`
Enumeration of the input PWM channels

Enumerator

GDU_IP_PWM_CHANNEL_MIN	lower boundary for consecutive enumerators
GDU_IP_PWM_CHANNEL_0	input PWM channel 0
GDU_IP_PWM_CHANNEL_1	input PWM channel 1
GDU_IP_PWM_CHANNEL_2	input PWM channel 2
GDU_IP_PWM_CHANNEL_3	input PWM channel 3
GDU_IP_PWM_CHANNEL_4	input PWM channel 4
GDU_IP_PWM_CHANNEL_5	input PWM channel 5
GDU_IP_PWM_CHANNEL_MAX	upper boundary for consecutive enumerators

Definition at line 129 of file Gdu_Ip_Types.h.

6.2.5.5 Gdu_Ip_EdgeType

enum `Gdu_Ip_EdgeType`
Enumeration of a signal's edge types

Enumerator

GDU_IP_EDGE_MIN	lower boundary for consecutive enumerators
GDU_IP_EDGE_RISING	rising edge
GDU_IP_EDGE_FALLING	falling edge

Enumerator

GDU_IP_EDGE_MAX	upper boundary for consecutive enumerators

Definition at line 142 of file Gdu_Ip_Types.h.

6.2.5.6 Gdu_Ip_SlewRateSelectType

enum Gdu_Ip_SlewRateSelectType
Enumeration of slew rate configuration registers

Enumerator

GDU_IP_SLEWRATE_MIN	lower boundary for consecutive enumerators
GDU_IP_SLEWRATE_HS_ON	HSSRON register
GDU_IP_SLEWRATE_HS_OFF	HSSROFF register
GDU_IP_SLEWRATE_LS_ON	LSSRON register
GDU_IP_SLEWRATE_LS_OFF	LSSROFF register
GDU_IP_SLEWRATE_MAX	upper boundary for consecutive enumerators

Definition at line 151 of file Gdu_Ip_Types.h.

6.2.5.7 Gdu_Ip_IrefTrimType

enum Gdu_Ip_IrefTrimType
Enumeration of the possible trimming percentages

Enumerator

GDU_IP_IREF_TRIM_MIN	lower boundary for consecutive enumerators
GDU_IP_IREF_TRIM_000	62.5% (minimum current)
GDU_IP_IREF_TRIM_001	75%
GDU_IP_IREF_TRIM_010	87.5%
GDU_IP_IREF_TRIM_011	100% (10 A at nominal process)
GDU_IP_IREF_TRIM_100	112.5%
GDU_IP_IREF_TRIM_101	125%
GDU_IP_IREF_TRIM_110	137.5%
GDU_IP_IREF_TRIM_111	150% (maximum current)
GDU_IP_IREF_TRIM_MAX	upper boundary for consecutive enumerators

Module Documentation

Definition at line 162 of file Gdu_Ip_Types.h.

6.2.6 Function Reference

6.2.6.1 Gdu_Ip_Init()

Gdu_Ip_StatusType Gdu_Ip_Init (
            const Gdu_Ip_ConfigType *const *Configuration*)

IPV_GDU initialization API.
Call this API to configure the IP driver.

Parameters

in	*Configuration*	Contains the address of the configuration structure.

Returns

Returns the status of the driver initialization.

Return values

GDU_IP_STATUS_OK	command processed successfully
GDU_IP_STATUS_ERROR	failure from Aec_Ip_SpiWrite

Precondition

The 6 PWM lines routed to this module are OFF.

6.2.6.2 Gdu_Ip_SetMode()

Gdu_Ip_StatusType Gdu_Ip_SetMode (
            Gdu_Ip_ModeType *Mode*)

Control the operation mode (enable/disable).
Following the recommendations from the manual, charge pump [CPEN] follows the state of the GDU [GDUEN].

Parameters

in	*Mode*	specifies the desired mode

Returns

Returns the status of the command.

Return values

GDU_IP_STATUS_OK	command processed successfully
GDU_IP_STATUS_ERROR	failure from Aec_Ip_SpiWrite

Precondition

    IPV_GDU is initialized.

6.2.6.3 Gdu_Ip_SetSafeState()

Gdu_Ip_StatusType Gdu_Ip_SetSafeState (
            Gdu_Ip_SafeStateType *SafeState*)

Control the safe state (on/off).
Call this API to enable safe state before entering low power mode.

Parameters

in	*SafeState*	specifies the desired safe state

Returns

    Returns the status of the command.

Return values

GDU_IP_STATUS_OK	command processed successfully
GDU_IP_STATUS_ERROR	failure from Aec_Ip_SpiWrite

Precondition

    IPV_GDU is initialized.

6.2.6.4 Gdu_Ip_GetStatus()

Gdu_Ip_StatusType Gdu_Ip_GetStatus (
            Gdu_Ip_ErrorStatusType * *ErrorStatus*)

Get the status of the error conditions present in the INTF register.

Parameters

out	*ErrorStatus*	structure containing bit-fields that indicate if an error is present or not The user needs to create and pass an address to this type of structure only if they are interested in the status of each error condition; otherwise, the overall return status may be just enough.

Returns

    Returns the status of the command.

Return values

GDU_IP_STATUS_OK	no error conditions are active at the moment
GDU_IP_STATUS_ERROR	failure from Aec_Ip_SpiWrite, *ErrorStatus was not written
GDU_IP_STATUS_ERRORCONDITION	one of the error conditions is active

Precondition

    IPV_GDU is initialized.

6.2.6.5 Gdu_Ip_StartDelayMeasurement()

```
Gdu_Ip_StatusType Gdu_Ip_StartDelayMeasurement (
            Gdu_Ip_PwmChannelType PwmChannel,
            Gdu_Ip_EdgeType Edge )
```
Start the delay measurement between the PWM edge and the feedback signal coming from the FET gate driver.

Parameters

in	*PwmChannel*	the measurement is performed on a specific PWM channel
in	*Edge*	the measurement is performed between two edges of the signal: rising or falling

Returns

    Returns the status of the command.

Return values

GDU_IP_STATUS_OK	command processed successfully
GDU_IP_STATUS_ERROR	failure from Aec_Ip_SpiWrite

Precondition

    IPV_GDU is initialized.

    Make sure that the previous measurement had time to either finish or expire, because: "Writting to DLYMCFG while delay measurement is enabled can corrupt the delay measurement."

6.2.6.6 Gdu_Ip_ReadDelayMeasurement()

```
Gdu_Ip_StatusType Gdu_Ip_ReadDelayMeasurement (
            uint8 * Delay )
```
Read the delay measurement results.

Parameters

out	*Delay*	delay measured in clock cycles

Returns

    Returns the status of the command.

Return values

GDU_IP_STATUS_OK	command processed successfully
GDU_IP_STATUS_ERROR	failure from Aec_Ip_SpiWrite
GDU_IP_STATUS_INVALID	measurement result is unavailable or corrupted

Precondition

IPV_GDU is initialized.

6.2.6.7 Gdu_Ip_AdjustSlewRate()

Gdu_Ip_StatusType Gdu_Ip_AdjustSlewRate (
        Gdu_Ip_SlewRateSelectType *SlewRateSelect,*
        const Gdu_Ip_SlewRateConfigType * *SlewRateConfig*)

Adjust the slew rate of the FET gate drivers.

Parameters

in	*SlewRateSelect*	selects the HS/LS FETs, and the turn on / turn off waveforms
in	*SlewRateConfig*	pointer to a structure containing the configuration values

Returns

Returns the status of the command.

Return values

GDU_IP_STATUS_OK	command processed successfully
GDU_IP_STATUS_ERROR	failure from Aec_Ip_SpiWrite

Precondition

IPV_GDU is initialized.

Stop the PWMs inputs.

Put the GDU in safe state before making big changes to these values.

6.2.6.8 Gdu_Ip_SwIrefTrimming()

Gdu_Ip_StatusType Gdu_Ip_SwIrefTrimming (
        Gdu_Ip_IrefTrimmingType *IrefTrimming*)

Configure the Iref trimming percentages.

Parameters

in	*IrefTrimming*	structure containing trimming percentages for each of the six gates

Returns

Returns the status of the command.

Return values

GDU_IP_STATUS_OK	command processed successfully
GDU_IP_STATUS_ERROR	failure from Aec_Ip_SpiWrite

Precondition

IPV_GDU is initialized.

6.2.6.9 Gdu_Ip_OffStateDiagnostics()

`Gdu_Ip_StatusType` `Gdu_Ip_OffStateDiagnostics (`

`Gdu_Ip_OffStateDiagType` *OffStateDiagnostics* `)`

Connect the HS pins towards HD or GND in order to detect open loads, loads shorted to HD, or loads shorted to GND.
Beware that pulling the same pin to both HD and VSS may cause a short circuit.

Parameters

in	*OffStateDiagnostics*	structure containing boolean settings for each FET, and each direction

Returns

Returns the status of the command.

Return values

GDU_IP_STATUS_OK	command processed successfully
GDU_IP_STATUS_ERROR	failure from Aec_Ip_SpiWrite

Precondition

IPV_GDU is initialized. "In low-power mode it is not possible to connect the HSn pin to either the HD pin or the VSS pin."

6.2.6.10 Gdu_Ip_ISR()

`void Gdu_Ip_ISR (`

`void )`

ISR for GDU_INT.
If the module is initialized and at least one interrupt is enabled, then

- call GduNotification

- W1C the triggered interrupts

Precondition

No preconditions for this API.

6.2.6.11 ISR()

`ISR (`

`Gdu_Ip_Irq_ISR )`

ISR for GDU_INT.
Simply call Gdu_Ip_ISR and handle everything from there.

Precondition