

AC784xx_DFP GPIO

7.1.0

Generated by Doxygen 1.8.13

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	2
2.1	File List	2
3	Class Documentation	3
3.1	Gpio_Hal_SettingsConfigType Struct Reference	3
3.1.1	Detailed Description	3
3.1.2	Member Data Documentation	3
3.1.2.1	ChannelId	3
3.1.2.2	Direction	4
3.1.2.3	DriveStrength	4
3.1.2.4	InitValue	4
3.1.2.5	IntConfig	4
3.1.2.6	Mux	4
3.1.2.7	PinLock	5
3.1.2.8	PortId	5
3.1.2.9	PullConfig	5
3.2	Port_Hal_DigitalFilterCfgType Struct Reference	5
3.2.1	Detailed Description	5
3.2.2	Member Data Documentation	6
3.2.2.1	Clock	6
3.2.2.2	PinMask	6
3.2.2.3	Width	6

4 File Documentation	7
4.1 AC784xx_API_Reference_Manual_GPIO.pdf File Reference	7
4.2 AC784xx_Gpio_Reg.h File Reference	7
4.2.1 Detailed Description	8
4.2.2 Function Documentation	9
4.2.2.1 Gpio_Reg_GetPIDR()	9
4.2.2.2 Gpio_Reg_GetPIER()	9
4.2.2.3 Gpio_Reg_GetPIOR()	10
4.2.2.4 Gpio_Reg_GetPODR()	10
4.2.2.5 Gpio_Reg_GetPOER()	10
4.2.2.6 Gpio_Reg_GetPROR()	11
4.2.2.7 Gpio_Reg_GetPSOR()	11
4.2.2.8 Gpio_Reg_ModifyPIER()	12
4.2.2.9 Gpio_Reg_ModifyPODR()	12
4.2.2.10 Gpio_Reg_ModifyPOER()	13
4.2.2.11 Gpio_Reg_SetPIDR()	13
4.2.2.12 Gpio_Reg_SetPIER()	14
4.2.2.13 Gpio_Reg_SetPIOR()	14
4.2.2.14 Gpio_Reg_SetPODR()	15
4.2.2.15 Gpio_Reg_SetPOER()	15
4.2.2.16 Gpio_Reg_SetPROR()	16
4.2.2.17 Gpio_Reg_SetPSOR()	16
4.2.2.18 Port_Reg_ClearInterruptFlag()	17
4.2.2.19 Port_Reg_EnableDigitalFilter()	17
4.2.2.20 Port_Reg_GetDFCR()	18
4.2.2.21 Port_Reg_GetDFER()	18
4.2.2.22 Port_Reg_GetDFWR()	19
4.2.2.23 Port_Reg_GetInterruptFlag()	19
4.2.2.24 Port_Reg_GetISFR()	20
4.2.2.25 Port_Reg_GetPCR()	20
4.2.2.26 Port_Reg_SetDFCR()	21

4.2.2.27	Port_Reg_SetDFER()	21
4.2.2.28	Port_Reg_SetDFWR()	22
4.2.2.29	Port_Reg_SetDriverStrength()	22
4.2.2.30	Port_Reg_SetInterruptMode()	23
4.2.2.31	Port_Reg_SetISFR()	23
4.2.2.32	Port_Reg_SetPCR()	24
4.2.2.33	Port_Reg_SetPinLock()	24
4.2.2.34	Port_Reg_SetPinMux()	25
4.2.2.35	Port_Reg_SetPullConfig()	25
4.3	Gpio_Hal.c File Reference	26
4.3.1	Detailed Description	27
4.3.2	Macro Definition Documentation	27
4.3.2.1	CHANNEL_NUM_MAX	28
4.3.2.2	MODIFY_VALUE	28
4.3.2.3	PIN_NUM_MAX	28
4.3.3	Function Documentation	28
4.3.3.1	Gpio_Hal_Channellnit()	28
4.3.3.2	Gpio_Hal_ClearIntStatus()	29
4.3.3.3	Gpio_Hal_ClearPins()	29
4.3.3.4	Gpio_Hal_ConfigDigitalFilter()	30
4.3.3.5	Gpio_Hal_EnableDigitalFilter()	30
4.3.3.6	Gpio_Hal_GetIntStatus()	31
4.3.3.7	Gpio_Hal_GetPinDirection()	31
4.3.3.8	Gpio_Hal_GetPinIntSel()	32
4.3.3.9	Gpio_Hal_GetPinOutputLevel()	32
4.3.3.10	Gpio_Hal_GetPinsDirection()	33
4.3.3.11	Gpio_Hal_GetPinsOutputLevel()	33
4.3.3.12	Gpio_Hal_InstallCallback()	34
4.3.3.13	Gpio_Hal_ReadPin()	34
4.3.3.14	Gpio_Hal_ReadPins()	35
4.3.3.15	Gpio_Hal_SetHighZ()	35

4.3.3.16	Gpio_Hal_SetMuxMode()	36
4.3.3.17	Gpio_Hal_SetPinDirection()	36
4.3.3.18	Gpio_Hal_SetPinIntSel()	37
4.3.3.19	Gpio_Hal_SetPins()	37
4.3.3.20	Gpio_Hal_SetPinsDirection()	38
4.3.3.21	Gpio_Hal_SetPullSel()	39
4.3.3.22	Gpio_Hal_TogglePin()	39
4.3.3.23	Gpio_Hal_TogglePins()	40
4.3.3.24	Gpio_Hal_WritePin()	40
4.3.3.25	Gpio_Hal_WritePins()	41
4.3.3.26	ISR() [1/5]	41
4.3.3.27	ISR() [2/5]	41
4.3.3.28	ISR() [3/5]	42
4.3.3.29	ISR() [4/5]	42
4.3.3.30	ISR() [5/5]	42
4.4	Gpio_Hal.h File Reference	42
4.4.1	Detailed Description	44
4.4.2	Macro Definition Documentation	44
4.4.2.1	PORTID_A	44
4.4.2.2	PORTID_B	44
4.4.2.3	PORTID_C	44
4.4.2.4	PORTID_D	44
4.4.2.5	PORTID_E	45
4.4.3	Function Documentation	45
4.4.3.1	Gpio_Hal_Channellnit()	45
4.4.3.2	Gpio_Hal_ClearIntStatus()	45
4.4.3.3	Gpio_Hal_ClearPins()	46
4.4.3.4	Gpio_Hal_ConfigDigitalFilter()	46
4.4.3.5	Gpio_Hal_EnableDigitalFilter()	47
4.4.3.6	Gpio_Hal_GetIntStatus()	47
4.4.3.7	Gpio_Hal_GetPinDirection()	48

4.4.3.8	Gpio_Hal_GetPinIntSel()	48
4.4.3.9	Gpio_Hal_GetPinOutputLevel()	49
4.4.3.10	Gpio_Hal_GetPinsDirection()	49
4.4.3.11	Gpio_Hal_GetPinsOutputLevel()	50
4.4.3.12	Gpio_Hal_InstallCallback()	50
4.4.3.13	Gpio_Hal_ReadPin()	51
4.4.3.14	Gpio_Hal_ReadPins()	51
4.4.3.15	Gpio_Hal_SetHighZ()	52
4.4.3.16	Gpio_Hal_SetMuxMode()	52
4.4.3.17	Gpio_Hal_SetPinDirection()	53
4.4.3.18	Gpio_Hal_SetPinIntSel()	53
4.4.3.19	Gpio_Hal_SetPins()	54
4.4.3.20	Gpio_Hal_SetPinsDirection()	55
4.4.3.21	Gpio_Hal_SetPullSel()	55
4.4.3.22	Gpio_Hal_TogglePin()	56
4.4.3.23	Gpio_Hal_TogglePins()	56
4.4.3.24	Gpio_Hal_WritePin()	57
4.4.3.25	Gpio_Hal_WritePins()	57
4.5	Gpio_Hal_Types.h File Reference	58
4.5.1	Typedef Documentation	59
4.5.1.1	Gpio_Hal_ChannelType	59
4.5.1.2	Port_CallbackType	59
4.5.2	Enumeration Type Documentation	59
4.5.2.1	Gpio_Hal_LevelType	59
4.5.2.2	Port_Hal_DataDirectionType	60
4.5.2.3	Port_Hal_DigitalFilterClockType	60
4.5.2.4	Port_Hal_DriveStrengthType	60
4.5.2.5	Port_Hal_InterruptConfigType	61
4.5.2.6	Port_Hal_MuxType	61
4.5.2.7	Port_Hal_PullConfigType	61

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Gpio_Hal_SettingsConfigType	
Defines the gpio configuration structure	3
Port_Hal_DigitalFilterCfgType	
The digital filter configuration	5

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

AC784xx_API_Reference_Manual_GPIO.pdf	7
AC784xx_Gpio_Reg.h	
This file provides gpio hardware integration interface	7
Gpio_Hal.c	
This file provides gpio integration functions	26
Gpio_Hal.h	
This file provides extern Hal Gpio api	42
Gpio_Hal_Types.h	58

Chapter 3

Class Documentation

3.1 Gpio_Hal_SettingsConfigType Struct Reference

Defines the gpio configuration structure.

```
#include <Gpio_Hal_Types.h>
```

Public Attributes

- uint8 [PortId](#)
- uint8 [ChannelId](#)
- [Port_Hal_PullConfigType](#) [PullConfig](#)
- [Port_Hal_DriveStrengthType](#) [DriveStrength](#)
- [Port_Hal_MuxType](#) [Mux](#)
- boolean [PinLock](#)
- [Port_Hal_InterruptConfigType](#) [IntConfig](#)
- [Port_Hal_DataDirectionType](#) [Direction](#)
- [Gpio_Hal_LevelType](#) [InitValue](#)

3.1.1 Detailed Description

Defines the gpio configuration structure.

Definition at line 164 of file Gpio_Hal_Types.h.

3.1.2 Member Data Documentation

3.1.2.1 ChannelId

```
uint8 Gpio_Hal_SettingsConfigType::ChannelId
```

Pin index.

Definition at line 167 of file Gpio_Hal_Types.h.

3.1.2.2 Direction

`Port_Hal_DataDirectionType` `Gpio_Hal_SettingsConfigType::Direction`

Configures the port data direction.

Definition at line 173 of file `Gpio_Hal_Types.h`.

3.1.2.3 DriveStrength

`Port_Hal_DriveStrengthType` `Gpio_Hal_SettingsConfigType::DriveStrength`

Configures the drive strength.

Definition at line 169 of file `Gpio_Hal_Types.h`.

3.1.2.4 InitValue

`Gpio_Hal_LevelType` `Gpio_Hal_SettingsConfigType::InitValue`

Initial value

Definition at line 174 of file `Gpio_Hal_Types.h`.

3.1.2.5 IntConfig

`Port_Hal_InterruptConfigType` `Gpio_Hal_SettingsConfigType::IntConfig`

Interrupt generation condition.

Definition at line 172 of file `Gpio_Hal_Types.h`.

3.1.2.6 Mux

`Port_Hal_MuxType` `Gpio_Hal_SettingsConfigType::Mux`

Pin mux selection.

Definition at line 170 of file `Gpio_Hal_Types.h`.

3.1.2.7 PinLock

```
boolean Gpio_Hal_SettingsConfigType::PinLock
```

Lock pin control register or not.

Definition at line 171 of file Gpio_Hal_Types.h.

3.1.2.8 PortId

```
uint8 Gpio_Hal_SettingsConfigType::PortId
```

Port index.

Definition at line 166 of file Gpio_Hal_Types.h.

3.1.2.9 PullConfig

```
Port_Hal_PullConfigType Gpio_Hal_SettingsConfigType::PullConfig
```

Internal pull-up/down resistor selection.

Definition at line 168 of file Gpio_Hal_Types.h.

The documentation for this struct was generated from the following file:

- [Gpio_Hal_Types.h](#)

3.2 Port_Hal_DigitalFilterCfgType Struct Reference

The digital filter configuration.

```
#include <Gpio_Hal_Types.h>
```

Public Attributes

- [Port_Hal_DigitalFilterClockType](#) Clock
- uint8 Width
- uint32 PinMask

3.2.1 Detailed Description

The digital filter configuration.

Definition at line 153 of file Gpio_Hal_Types.h.

3.2.2 Member Data Documentation

3.2.2.1 Clock

[Port_Hal_DigitalFilterClockType](#) Port_Hal_DigitalFilterCfgType::Clock

The digital filter clock for port

Definition at line 155 of file Gpio_Hal_Types.h.

3.2.2.2 PinMask

uint32 Port_Hal_DigitalFilterCfgType::PinMask

Mask of pins for which digital filter is enabled

Definition at line 157 of file Gpio_Hal_Types.h.

3.2.2.3 Width

uint8 Port_Hal_DigitalFilterCfgType::Width

The digital filter width value

Definition at line 156 of file Gpio_Hal_Types.h.

The documentation for this struct was generated from the following file:

- [Gpio_Hal_Types.h](#)

Chapter 4

File Documentation

4.1 AC784xx_API_Reference_Manual_GPIO.pdf File Reference

4.2 AC784xx_Gpio_Reg.h File Reference

This file provides gpio hardware integration interface.

```
#include "Device_Register.h"
```

Functions

- LOCAL_INLINE void [Port_Reg_SetPCR](#) (uint8 PortId, uint32 ChannelId, uint32 Value)
Set the Pin control register.
- LOCAL_INLINE uint32 [Port_Reg_GetPCR](#) (const uint8 PortId, uint32 ChannelId)
Get the Pin control register.
- LOCAL_INLINE void [Port_Reg_SetPullConfig](#) (uint8 PortId, uint32 ChannelId, uint32 Value)
Set the up/down pull of pin.
- LOCAL_INLINE void [Port_Reg_SetDriverStrength](#) (uint8 PortId, uint32 ChannelId, uint32 Value)
Set the driver strength of pin.
- LOCAL_INLINE void [Port_Reg_SetPinMux](#) (uint8 PortId, uint32 ChannelId, uint32 Value)
Set the Pin mux mode.
- LOCAL_INLINE void [Port_Reg_SetPinLock](#) (uint8 PortId, uint32 ChannelId, uint32 Value)
Set the Pin lock.
- LOCAL_INLINE void [Port_Reg_SetInterruptMode](#) (uint8 PortId, uint32 ChannelId, uint32 Value)
Set the Pin interrupt mode .
- LOCAL_INLINE void [Port_Reg_ClearInterruptFlag](#) (uint8 PortId, uint32 ChannelId)
Clear the port control interrupt status flag register.
- LOCAL_INLINE uint32 [Port_Reg_GetInterruptFlag](#) (uint8 PortId, uint32 ChannelId)
get the port control interrupt status flag register.
- LOCAL_INLINE void [Port_Reg_SetISFR](#) (uint8 PortId, uint32 Value)
Set the port interrupt status flag register.
- LOCAL_INLINE uint32 [Port_Reg_GetISFR](#) (const uint8 PortId)
Get the port interrupt status flag register.
- LOCAL_INLINE void [Port_Reg_EnableDigitalFilter](#) (uint8 PortId, uint32 ChannelId, boolean En)
Set the port digital filter enable register.

- LOCAL_INLINE void [Port_Reg_SetDFER](#) (uint8 PortId, uint32 Value)
Set the port digital filter enable register.
- LOCAL_INLINE uint32 [Port_Reg_GetDFER](#) (uint8 PortId)
Get the port digital filter enable register.
- LOCAL_INLINE void [Port_Reg_SetDFCR](#) (uint8 PortId, uint32 Value)
Set the port digital filter clock register.
- LOCAL_INLINE uint32 [Port_Reg_GetDFCR](#) (uint8 PortId)
Get the port digital filter clock register.
- LOCAL_INLINE void [Port_Reg_SetDFWR](#) (uint8 PortId, uint32 Value)
Set the port digital filter width register.
- LOCAL_INLINE uint32 [Port_Reg_GetDFWR](#) (uint8 PortId)
Get the port digital filter width register.
- LOCAL_INLINE void [Gpio_Reg_SetPODR](#) (uint8 GpioId, uint32 Value)
write the port output data register
- LOCAL_INLINE uint32 [Gpio_Reg_GetPODR](#) (const uint8 GpioId)
Get the port output data register.
- LOCAL_INLINE void [Gpio_Reg_SetPSOR](#) (uint8 GpioId, uint32 Value)
Set the port set output data register.
- LOCAL_INLINE uint32 [Gpio_Reg_GetPSOR](#) (const uint8 GpioId)
Get the port set output data register.
- LOCAL_INLINE void [Gpio_Reg_SetPROR](#) (uint8 GpioId, uint32 Value)
Set the port clear output data register.
- LOCAL_INLINE uint32 [Gpio_Reg_GetPROR](#) (const uint8 GpioId)
Get the port clear output data register.
- LOCAL_INLINE void [Gpio_Reg_SetPIOR](#) (uint8 GpioId, uint32 Value)
Set the port toggle output data register.
- LOCAL_INLINE uint32 [Gpio_Reg_GetPIOR](#) (const uint8 GpioId)
Get the port toggle output data register.
- LOCAL_INLINE void [Gpio_Reg_SetPIDR](#) (uint8 GpioId, uint32 Value)
Set the port input data register.
- LOCAL_INLINE uint32 [Gpio_Reg_GetPIDR](#) (const uint8 GpioId)
Get the port input data register.
- LOCAL_INLINE void [Gpio_Reg_SetPOER](#) (uint8 GpioId, uint32 Value)
Set the port output enable register.
- LOCAL_INLINE uint32 [Gpio_Reg_GetPOER](#) (const uint8 GpioId)
Get the port output enable register.
- LOCAL_INLINE void [Gpio_Reg_SetPIER](#) (uint8 GpioId, uint32 Value)
Set the port input enable register.
- LOCAL_INLINE uint32 [Gpio_Reg_GetPIER](#) (const uint8 GpioId)
Get the port input enable register.
- LOCAL_INLINE void [Gpio_Reg_ModifyPIER](#) (uint8 GpioId, uint32 ChannelId, uint32 Value)
write the port input enable register
- LOCAL_INLINE void [Gpio_Reg_ModifyPOER](#) (uint8 GpioId, uint32 ChannelId, uint32 Value)
write the port output enable register
- LOCAL_INLINE void [Gpio_Reg_ModifyPODR](#) (uint8 GpioId, uint32 ChannelId, uint32 Value)
write the port output data register

4.2.1 Detailed Description

This file provides gpio hardware integration interface.

4.2.2 Function Documentation

4.2.2.1 Gpio_Reg_GetPIDR()

```
LOCAL_INLINE uint32 Gpio_Reg_GetPIDR (
    const uint8 GpioId )
```

Get the port input data register.

Note

Function ID : DES_GPIO_API_327

Parameters

in	<i>GpioId</i>	Gpio id
----	---------------	---------

Returns

uint32 Each bit represents one Pin input Value high(1) or low(0) (LSB is Pin 0, MSB is Pin 31).

Definition at line 471 of file AC784xx_Gpio_Reg.h.

4.2.2.2 Gpio_Reg_GetPIER()

```
LOCAL_INLINE uint32 Gpio_Reg_GetPIER (
    const uint8 GpioId )
```

Get the port input enable register.

Note

Function ID : DES_GPIO_API_331

Parameters

in	<i>GpioId</i>	Gpio id
----	---------------	---------

Returns

uint32 Each bit represents one Pin input enable(1) or disable(0) (LSB is Pin 0, MSB is Pin 31).

Definition at line 529 of file AC784xx_Gpio_Reg.h.

4.2.2.3 Gpio_Reg_GetPIOR()

```
LOCAL_INLINE uint32 Gpio_Reg_GetPIOR (
    const uint8 GpioId )
```

Get the port toggle output data register.

Note

Function ID : DES_GPIO_API_325

Parameters

in	<i>Gpio↔ Id</i>	Gpio id
----	---------------------	---------

Returns

uint32 Each bit represents one Pin toggle Value toggle(1) or not effect(0) (LSB is Pin 0, MSB is Pin 31).

Definition at line 442 of file AC784xx_Gpio_Reg.h.

4.2.2.4 Gpio_Reg_GetPODR()

```
LOCAL_INLINE uint32 Gpio_Reg_GetPODR (
    const uint8 GpioId )
```

Get the port output data register.

Note

Function ID : DES_GPIO_API_319

Parameters

in	<i>Gpio↔ Id</i>	Gpio id
----	---------------------	---------

Returns

uint32 Each bit represents one Pin output Value high(1) or low(0) (LSB is Pin 0, MSB is Pin 31).

Definition at line 355 of file AC784xx_Gpio_Reg.h.

4.2.2.5 Gpio_Reg_GetPOER()

```
LOCAL_INLINE uint32 Gpio_Reg_GetPOER (
    const uint8 GpioId )
```

Get the port output enable register.

Note

Function ID : DES_GPIO_API_329

Parameters

in	<i>GpioId</i>	Gpio id
----	---------------	---------

Returns

uint32 Each bit represents one Pin output enable(1) or disable(0) (LSB is Pin 0, MSB is Pin 31).

Definition at line 500 of file AC784xx_Gpio_Reg.h.

4.2.2.6 Gpio_Reg_GetPROR()

```
LOCAL_INLINE uint32 Gpio_Reg_GetPROR (  
    const uint8 GpioId )
```

Get the port clear output data register.

Note

Function ID : DES_GPIO_API_323

Parameters

in	<i>GpioId</i>	Gpio id
----	---------------	---------

Returns

uint32 Each bit represents one Pin reset Value reset(1) or not effect(0) (LSB is Pin 0, MSB is Pin 31).

Definition at line 413 of file AC784xx_Gpio_Reg.h.

4.2.2.7 Gpio_Reg_GetPSOR()

```
LOCAL_INLINE uint32 Gpio_Reg_GetPSOR (  
    const uint8 GpioId )
```

Get the port set output data register.

Note

Function ID : DES_GPIO_API_321

Parameters

in	<i>GpioId</i>	Gpio id
----	---------------	---------

Returns

uint32 Each bit represents one Pin set Value set high(1) or not effect(0) (LSB is Pin 0, MSB is Pin 31).

Definition at line 384 of file AC784xx_Gpio_Reg.h.

4.2.2.8 Gpio_Reg_ModifyPIER()

```
LOCAL_INLINE void Gpio_Reg_ModifyPIER (
    uint8 GpioId,
    uint32 ChannelId,
    uint32 Value )
```

write the port input enable register

Note

Function ID : DES_GPIO_API_332

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Value</i>	1 means enable , 0 means disable

Returns

void

Definition at line 544 of file AC784xx_Gpio_Reg.h.

4.2.2.9 Gpio_Reg_ModifyPODR()

```
LOCAL_INLINE void Gpio_Reg_ModifyPODR (
    uint8 GpioId,
    uint32 ChannelId,
    uint32 Value )
```

write the port output data register

Note

Function ID : DES_GPIO_API_334

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Channel↔ Id</i>	channel id of port
in	<i>Value</i>	1 means high level , 0 means low level

Returns

void

Definition at line 576 of file AC784xx_Gpio_Reg.h.

4.2.2.10 Gpio_Reg_ModifyPOER()

```
LOCAL_INLINE void Gpio_Reg_ModifyPOER (
    uint8 GpioId,
    uint32 ChannelId,
    uint32 Value )
```

write the port output enable register

Note

Function ID : DES_GPIO_API_333

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Channel↔ Id</i>	channel id of port
in	<i>Value</i>	1 means enable , 0 means disable

Returns

void

Definition at line 560 of file AC784xx_Gpio_Reg.h.

4.2.2.11 Gpio_Reg_SetPIDR()

```
LOCAL_INLINE void Gpio_Reg_SetPIDR (
    uint8 GpioId,
    uint32 Value )
```

Set the port input data register.

Note

Function ID : DES_GPIO_API_326

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Value</i>	the port Value to set.Each bit represents one Pin input Value high(1) or low(0) (LSB is Pin 0, MSB is Pin 31).

Returns

void

Definition at line 457 of file AC784xx_Gpio_Reg.h.

4.2.2.12 Gpio_Reg_SetPIER()

```
LOCAL_INLINE void Gpio_Reg_SetPIER (
    uint8 GpioId,
    uint32 Value )
```

Set the port input enable register.

Note

Function ID : DES_GPIO_API_330

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Value</i>	the port Value to set.Each bit represents one Pin input enable(1) or disable(0) (LSB is Pin 0, MSB is Pin 31).

Returns

void

Definition at line 515 of file AC784xx_Gpio_Reg.h.

4.2.2.13 Gpio_Reg_SetPIOR()

```
LOCAL_INLINE void Gpio_Reg_SetPIOR (
    uint8 GpioId,
    uint32 Value )
```

Set the port toggle output data register.

Note

Function ID : DES_GPIO_API_324

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Value</i>	the port Value to set.Each bit represents one Pin toggle Value toggle(1) or not effect(0) (LSB is Pin 0, MSB is Pin 31).

Returns

void

Definition at line 428 of file AC784xx_Gpio_Reg.h.

4.2.2.14 Gpio_Reg_SetPODR()

```
LOCAL_INLINE void Gpio_Reg_SetPODR (
    uint8 GpioId,
    uint32 Value )
```

write the port output data register

Note

Function ID : DES_GPIO_API_318

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Value</i>	the port Value to set.Each bit represents one Pin output Value high(1) or low(0) (LSB is Pin 0, MSB is Pin 31).

Returns

void

Definition at line 341 of file AC784xx_Gpio_Reg.h.

4.2.2.15 Gpio_Reg_SetPOER()

```
LOCAL_INLINE void Gpio_Reg_SetPOER (
    uint8 GpioId,
    uint32 Value )
```

Set the port ouput enable register.

Note

Function ID : DES_GPIO_API_328

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Value</i>	the port Value to set.Each bit represents one Pin output enable(1) or disable(0) (LSB is Pin 0, MSB is Pin 31).

Returns

void

Definition at line 486 of file AC784xx_Gpio_Reg.h.

4.2.2.16 Gpio_Reg_SetPROR()

```
LOCAL_INLINE void Gpio_Reg_SetPROR (
    uint8 GpioId,
    uint32 Value )
```

Set the port clear output data register.

Note

Function ID : DES_GPIO_API_322

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Value</i>	the port Value to set.Each bit represents one Pin reset Value reset(1) or not effect(0) (LSB is Pin 0, MSB is Pin 31).

Returns

void

Definition at line 399 of file AC784xx_Gpio_Reg.h.

4.2.2.17 Gpio_Reg_SetPSOR()

```
LOCAL_INLINE void Gpio_Reg_SetPSOR (
    uint8 GpioId,
    uint32 Value )
```

Set the port set output data register.

Note

Function ID : DES_GPIO_API_320

Parameters

in	<i>GpioId</i>	Gpio id
in	<i>Value</i>	the port Value to set.Each bit represents one Pin set Value set high(1) or not effect(0) (LSB is Pin 0, MSB is Pin 31).

Returns

void

Definition at line 370 of file AC784xx_Gpio_Reg.h.

4.2.2.18 Port_Reg_ClearInterruptFlag()

```
LOCAL_INLINE void Port_Reg_ClearInterruptFlag (
    uint8 PortId,
    uint32 ChannelId )
```

Clear the port control interrupt status flag register.

Note

Function ID : DES_GPIO_API_308

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id

Returns

void

Definition at line 179 of file AC784xx_Gpio_Reg.h.

4.2.2.19 Port_Reg_EnableDigitalFilter()

```
LOCAL_INLINE void Port_Reg_EnableDigitalFilter (
    uint8 PortId,
    uint32 ChannelId,
    boolean En )
```

Set the port digital filter enable register.

Note

Function ID : DES_GPIO_API_311

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id
in	<i>En</i>	enable/disable digital filter

Returns

void

Definition at line 239 of file AC784xx_Gpio_Reg.h.

4.2.2.20 Port_Reg_GetDFCR()

```
LOCAL_INLINE uint32 Port_Reg_GetDFCR (
    uint8 PortId )
```

Get the port digital filter clock register.

Note

Function ID : DES_GPIO_API_315

Parameters

in	<i>Port↔ Id</i>	port id
----	---------------------	---------

Returns

uint32

Definition at line 297 of file AC784xx_Gpio_Reg.h.

4.2.2.21 Port_Reg_GetDFER()

```
LOCAL_INLINE uint32 Port_Reg_GetDFER (
    uint8 PortId )
```

Get the port digital filter enable register.

Note

Function ID : DES_GPIO_API_313

Parameters

in	<i>Port↔ Id</i>	port id
----	---------------------	---------

Returns

uint32 Each bit represents one Pin digital filter enable(1) or disable(0) (LSB is Pin 0, MSB is Pin 31).

Definition at line 268 of file AC784xx_Gpio_Reg.h.

4.2.2.22 Port_Reg_GetDFWR()

```
LOCAL_INLINE uint32 Port_Reg_GetDFWR (
    uint8 PortId )
```

Get the port digital filter width register.

Note

Function ID : DES_GPIO_API_317

Parameters

in	<i>Port↔ Id</i>	port id
----	---------------------	---------

Returns

uint32 digital filter width

Definition at line 326 of file AC784xx_Gpio_Reg.h.

4.2.2.23 Port_Reg_GetInterruptFlag()

```
LOCAL_INLINE uint32 Port_Reg_GetInterruptFlag (
    uint8 PortId,
    uint32 ChannelId )
```

get the port control interrupt status flag register.

Note

Function ID : DES_GPIO_API_335

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id

Returns

uint32

Definition at line 194 of file AC784xx_Gpio_Reg.h.

4.2.2.24 Port_Reg_GetISFR()

```
LOCAL_INLINE uint32 Port_Reg_GetISFR (
    const uint8 PortId )
```

Get the port interrupt status flag register.

Note

Function ID : DES_GPIO_API_310

Parameters

in	<i>Port↔ Id</i>	port id
----	---------------------	---------

Returns

uint32 Each bit represents one Pin interrupt status interrupted(1) or no interrupt(0) (LSB is Pin 0, MSB is Pin 31)

Definition at line 223 of file AC784xx_Gpio_Reg.h.

4.2.2.25 Port_Reg_GetPCR()

```
LOCAL_INLINE uint32 Port_Reg_GetPCR (
    const uint8 PortId,
    uint32 ChannelId )
```

Get the Pin control register.

Note

Function ID : DES_GPIO_API_302

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id

Returns

uint32

Definition at line 84 of file AC784xx_Gpio_Reg.h.

4.2.2.26 Port_Reg_SetDFCR()

```
LOCAL_INLINE void Port_Reg_SetDFCR (
    uint8 PortId,
    uint32 Value )
```

Set the port digital filter clock register.

Note

Function ID : DES_GPIO_API_314

Parameters

in	<i>Port↔ Id</i>	port id
in	<i>Value</i>	port digital filter clock Value to set

Returns

void

Definition at line 283 of file AC784xx_Gpio_Reg.h.

4.2.2.27 Port_Reg_SetDFER()

```
LOCAL_INLINE void Port_Reg_SetDFER (
    uint8 PortId,
    uint32 Value )
```

Set the port digital filter enable register.

Note

Function ID : DES_GPIO_API_312

Parameters

in	<i>PortId</i>	port id
in	<i>Value</i>	Each bit represents one Pin digital filter enable(1) or disable(0) (LSB is Pin 0, MSB is Pin 31)

Returns

void

Definition at line 254 of file AC784xx_Gpio_Reg.h.

4.2.2.28 Port_Reg_SetDFWR()

```
LOCAL_INLINE void Port_Reg_SetDFWR (
    uint8 PortId,
    uint32 Value )
```

Set the port digital filter width register.

Note

Function ID : DES_GPIO_API_316

Parameters

in	<i>PortId</i>	port id
in	<i>Value</i>	port digital filter width Value to set

Returns

void

Definition at line 312 of file AC784xx_Gpio_Reg.h.

4.2.2.29 Port_Reg_SetDriverStrength()

```
LOCAL_INLINE void Port_Reg_SetDriverStrength (
    uint8 PortId,
    uint32 ChannelId,
    uint32 Value )
```

Set the driver strength of pin.

Note

Function ID : DES_GPIO_API_304

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id
in	<i>Value</i>	driver strength Value to set

Returns

void

Definition at line 116 of file AC784xx_Gpio_Reg.h.

4.2.2.30 Port_Reg_SetInterruptMode()

```
LOCAL_INLINE void Port_Reg_SetInterruptMode (
    uint8 PortId,
    uint32 ChannelId,
    uint32 Value )
```

Set the Pin interrupt mode .

Note

Function ID : DES_GPIO_API_307

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id
in	<i>Value</i>	interrupt mode Value to set

Returns

void

Definition at line 164 of file AC784xx_Gpio_Reg.h.

4.2.2.31 Port_Reg_SetISFR()

```
LOCAL_INLINE void Port_Reg_SetISFR (
    uint8 PortId,
    uint32 Value )
```

Set the port interrupt status flag register.

Note

Function ID : DES_GPIO_API_309

Parameters

in	<i>PortId</i>	port id
in	<i>Value</i>	Each bit represents one Pin clear interrupt status clear(1) or not effect(0) (LSB is Pin 0, MSB is Pin 31)

Returns

void

Definition at line 209 of file AC784xx_Gpio_Reg.h.

4.2.2.32 Port_Reg_SetPCR()

```
LOCAL_INLINE void Port_Reg_SetPCR (
    uint8 PortId,
    uint32 ChannelId,
    uint32 Value )
```

Set the Pin control register.

Note

Function ID : DES_GPIO_API_301

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id
in	<i>Value</i>	port Value to set

Returns

void

Definition at line 69 of file AC784xx_Gpio_Reg.h.

4.2.2.33 Port_Reg_SetPinLock()

```
LOCAL_INLINE void Port_Reg_SetPinLock (
    uint8 PortId,
    uint32 ChannelId,
    uint32 Value )
```

Set the Pin lock.

Note

Function ID : DES_GPIO_API_306

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id
in	<i>Value</i>	lock Value to set

Returns

void

Definition at line 148 of file AC784xx_Gpio_Reg.h.

4.2.2.34 Port_Reg_SetPinMux()

```
LOCAL_INLINE void Port_Reg_SetPinMux (
    uint8 PortId,
    uint32 ChannelId,
    uint32 Value )
```

Set the Pin mux mode.

Note

Function ID : DES_GPIO_API_305

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id
in	<i>Value</i>	mux mode Value to set

Returns

void

Definition at line 132 of file AC784xx_Gpio_Reg.h.

4.2.2.35 Port_Reg_SetPullConfig()

```
LOCAL_INLINE void Port_Reg_SetPullConfig (
    uint8 PortId,
    uint32 ChannelId,
    uint32 Value )
```

Set the up/down pull of pin.

Note

Function ID : DES_GPIO_API_303

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id
in	<i>Value</i>	up/down pull Value to set

Returns

void

Definition at line 100 of file AC784xx_Gpio_Reg.h.

4.3 Gpio_Hal.c File Reference

This file provides gpio integration functions.

```
#include "Gpio_Hal.h"
#include "Core_Hal.h"
#include "AC784xx_Gpio_Reg.h"
```

Macros

- `#define PIN_NUM_MAX (153U)`
pin max number of current board
- `#define CHANNEL_NUM_MAX (32U)`
channel max number of current board
- `#define MODIFY_VALUE MODIFY_REG32`
MODIFY_REG32 redefine to MODIFY_VALUE to modify variable.

Functions

- `ISR (PORTA_IRQHandler)`
- `ISR (PORTB_IRQHandler)`
- `ISR (PORTC_IRQHandler)`
- `ISR (PORTD_IRQHandler)`
- `ISR (PORTE_IRQHandler)`
- void `Gpio_Hal_ChannelInit` (uint8 PinCount, const `Gpio_Hal_SettingsConfigType` Config[])
Initializes the gpio with the given configuration structure.
- void `Gpio_Hal_SetPullSel` (uint8 PortId, uint8 ChannelId, `Port_Hal_PullConfigType` PullConfig)
Configures the internal pull-up/down resistor.
- void `Gpio_Hal_SetMuxMode` (uint8 PortId, uint8 ChannelId, `Port_Hal_MuxType` Mux)
Configures the pin Mux.
- Hal_StatusType `Gpio_Hal_SetPinIntSel` (uint8 PortId, uint8 ChannelId, `Port_Hal_InterruptConfigType` IntConfig)
Configures the port pin interrupt/DMA request.
- `Port_Hal_InterruptConfigType` `Gpio_Hal_GetPinIntSel` (uint8 PortId, uint8 ChannelId)
Gets the current port pin interrupt/DMA request configuration.
- void `Gpio_Hal_EnableDigitalFilter` (uint8 PortId, uint8 ChannelId, boolean En)
Enables digital filter for digital pin Mux.

- void [Gpio_Hal_ConfigDigitalFilter](#) (uint8 PortId, const [Port_Hal_DigitalFilterCfgType](#) *Config)
Configures digital filter for port with given configuration.
- [Port_Hal_DataDirectionType](#) [Gpio_Hal_GetPinDirection](#) (uint8 PortId, uint8 ChannelId)
Get the pins directions configuration for a pin.
- [Gpio_Hal_ChannelType](#) [Gpio_Hal_GetPinsDirection](#) (uint8 PortId)
Get the pins directions configuration for a port.
- void [Gpio_Hal_SetPinDirection](#) (uint8 PortId, uint8 ChannelId, [Port_Hal_DataDirectionType](#) Direction)
Configure the direction for a certain pin from a port.
- void [Gpio_Hal_SetPinsDirection](#) (uint8 PortId, uint32 PinsDir)
Set the pins directions configuration for a port.
- void [Gpio_Hal_WritePin](#) (uint8 PortId, uint8 ChannelId, [Gpio_Hal_LevelType](#) Value)
Write a pin of a port with a given value.
- void [Gpio_Hal_WritePins](#) (uint8 PortId, [Gpio_Hal_ChannelType](#) Pins)
Write all pins of a port.
- [Gpio_Hal_LevelType](#) [Gpio_Hal_ReadPin](#) (uint8 PortId, uint8 ChannelId)
Read a pin input level.
- [Gpio_Hal_ChannelType](#) [Gpio_Hal_ReadPins](#) (uint8 PortId)
Read pins input level.
- void [Gpio_Hal_TogglePin](#) (uint8 PortId, uint8 ChannelId)
Toggle a pin output level.
- void [Gpio_Hal_TogglePins](#) (uint8 PortId, [Gpio_Hal_ChannelType](#) Pins)
Toggle pins output level.
- [Gpio_Hal_LevelType](#) [Gpio_Hal_GetPinOutputLevel](#) (uint8 PortId, uint8 ChannelId)
Get the current output level configuration from a pin.
- [Gpio_Hal_ChannelType](#) [Gpio_Hal_GetPinsOutputLevel](#) (uint8 PortId)
Get the current output level configuration from a port.
- void [Gpio_Hal_SetHighZ](#) (uint8 PortId, uint8 ChannelId, boolean En)
Set a pin to HighZ.
- void [Gpio_Hal_InstallCallback](#) (uint8 PortId, const [Port_CallbackType](#) Function)
Install GPIO interrupt callback function.
- void [Gpio_Hal_ClearIntStatus](#) (uint8 PortId, uint8 ChannelId)
Clear Gpio interrupt status.
- uint8 [Gpio_Hal_GetIntStatus](#) (uint8 PortId, uint8 ChannelId)
Get Gpio interrupt status.
- void [Gpio_Hal_ClearPins](#) (uint8 PortId, [Gpio_Hal_ChannelType](#) Pins)
Clear Gpio Pins.
- void [Gpio_Hal_SetPins](#) (uint8 PortId, [Gpio_Hal_ChannelType](#) Pins)
Set Gpio Pins.

4.3.1 Detailed Description

This file provides gpio integration functions.

4.3.2 Macro Definition Documentation

4.3.2.1 CHANNEL_NUM_MAX

```
#define CHANNEL_NUM_MAX (32U)
```

channel max number of current board

Definition at line 49 of file Gpio_Hal.c.

4.3.2.2 MODIFY_VALUE

```
#define MODIFY_VALUE MODIFY_REG32
```

MODIFY_REG32 redefine to MODIFY_VALUE to modify variable.

Definition at line 51 of file Gpio_Hal.c.

4.3.2.3 PIN_NUM_MAX

```
#define PIN_NUM_MAX (153U)
```

pin max number of current board

Definition at line 47 of file Gpio_Hal.c.

4.3.3 Function Documentation

4.3.3.1 Gpio_Hal_ChannelInit()

```
void Gpio_Hal_ChannelInit (
    uint8 PinCount,
    const Gpio_Hal_SettingsConfigType Config[] )
```

Initializes the gpio with the given configuration structure.

Note

Function ID : DES_PORT_API_202

Service ID : NA

Parameters

in	<i>PinCount</i>	The number of configured pins in structure
in	<i>Config</i>	The configuration structure

Returns

void

Definition at line 97 of file Gpio_Hal.c.

4.3.3.2 Gpio_Hal_ClearIntStatus()

```
void Gpio_Hal_ClearIntStatus (
    uint8 PortId,
    uint8 ChannelId )
```

Clear Gpio interrupt status.

Note

Function ID : DES_PORT_API_223

Parameters

in	PortId	Gpio PortId
in	ChannelId	Gpio channel id

Returns

void

Definition at line 470 of file Gpio_Hal.c.

4.3.3.3 Gpio_Hal_ClearPins()

```
void Gpio_Hal_ClearPins (
    uint8 PortId,
    Gpio_Hal_ChannelType Pins )
```

Clear Gpio Pins.

Note

Function ID : DES_PORT_API_225

Parameters

in	PortId	Gpio PortId
in	Pins	Pin mask of bits to be Clear. Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit: <ul style="list-style-type: none">0: corresponding pin is unaffected1: corresponding pin is toggled
Generated by Doxygen		

Returns

void

Definition at line 488 of file Gpio_Hal.c.

4.3.3.4 Gpio_Hal_ConfigDigitalFilter()

```
void Gpio_Hal_ConfigDigitalFilter (
    uint8 PortId,
    const Port_Hal_DigitalFilterCfgType * Config )
```

Configures digital filter for port with given configuration.

Note

Function ID : DES_PORT_API_208

Parameters

in	<i>PortId</i>	port id
in	<i>Config</i>	The digital filter configuration struct

Returns

void

Definition at line 214 of file Gpio_Hal.c.

4.3.3.5 Gpio_Hal_EnableDigitalFilter()

```
void Gpio_Hal_EnableDigitalFilter (
    uint8 PortId,
    uint8 ChannelId,
    boolean En )
```

Enables digital filter for digital pin Mux.

Note

Function ID : DES_PORT_API_207

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id
in	<i>En</i>	Digital Filter enable/disalbe 0:diabile 1:enable

Returns

void

Definition at line 205 of file Gpio_Hal.c.

4.3.3.6 Gpio_Hal_GetIntStatus()

```
uint8 Gpio_Hal_GetIntStatus (
    uint8 PortId,
    uint8 ChannelId )
```

Get Gpio interrupt status.

Note

Function ID : DES_PORT_API_224

Parameters

in	<i>PortId</i>	Gpio PortId
in	<i>ChannelId</i>	Gpio channel id

Returns

Int status

Definition at line 478 of file Gpio_Hal.c.

4.3.3.7 Gpio_Hal_GetPinDirection()

```
Port_Hal_DataDirectionType Gpio_Hal_GetPinDirection (
    uint8 PortId,
    uint8 ChannelId )
```

Get the pins directions configuration for a pin.

Note

Function ID : DES_PORT_API_209

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id

Returns

the pin direction input or onput

Definition at line 233 of file Gpio_Hal.c.

4.3.3.8 Gpio_Hal_GetPinIntSel()

```
Port_Hal_InterruptConfigType Gpio_Hal_GetPinIntSel (
    uint8 PortId,
    uint8 ChannelId )
```

Gets the current port pin interrupt/DMA request configuration.

Note

Function ID : DES_PORT_API_206

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id

Returns

Interrupt configuration

Definition at line 189 of file Gpio_Hal.c.

4.3.3.9 Gpio_Hal_GetPinOutputLevel()

```
Gpio_Hal_LevelType Gpio_Hal_GetPinOutputLevel (
    uint8 PortId,
    uint8 ChannelId )
```

Get the current output level configuration from a pin.

Note

Function ID : DES_PORT_API_219

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id

Returns

the pin output level

Definition at line 401 of file Gpio_Hal.c.

4.3.3.10 Gpio_Hal_GetPinsDirection()

```
Gpio_Hal_ChannelType Gpio_Hal_GetPinsDirection (
    uint8 PortId )
```

Get the pins directions configuration for a port.

Note

Function ID : DES_PORT_API_210

Parameters

in	<i>Port↔ Id</i>	port id
----	---------------------	---------

Returns

GPIO directions: Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit:

- 0: corresponding pin is set to input
- 1: corresponding pin is set to output

Definition at line 258 of file Gpio_Hal.c.

4.3.3.11 Gpio_Hal_GetPinsOutputLevel()

```
Gpio_Hal_ChannelType Gpio_Hal_GetPinsOutputLevel (
    uint8 PortId )
```

Get the current output level configuration from a port.

Note

Function ID : DES_PORT_API_220

Parameters

in	<i>Port↔ Id</i>	Port id
----	---------------------	---------

Returns

GPIO outputs level configuration: Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit:

- 0: corresponding pin is set to low
- 1: corresponding pin is set to high

Definition at line 415 of file Gpio_Hal.c.

4.3.3.12 Gpio_Hal_InstallCallback()

```
void Gpio_Hal_InstallCallback (
    uint8 PortId,
    const Port_CallbackType Function )
```

Install GPIO interrupt callback function.

Note

Function ID : DES_PORT_API_222

Parameters

in	<i>PortId</i>	GPIO PortId
in	<i>Function</i>	The GPIO interrupt callback function to be installed

Returns

void

Definition at line 462 of file Gpio_Hal.c.

4.3.3.13 Gpio_Hal_ReadPin()

```
Gpio_Hal_LevelType Gpio_Hal_ReadPin (
    uint8 PortId,
    uint8 ChannelId )
```

Read a pin input level.

Note

Function ID : DES_PORT_API_215

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id

Returns

GPIO inputs level

Definition at line 343 of file Gpio_Hal.c.

4.3.3.14 Gpio_Hal_ReadPins()

```
Gpio_Hal_ChannelType Gpio_Hal_ReadPins (
    uint8 PortId )
```

Read pins input level.

Note

Function ID : DES_PORT_API_216

Parameters

in	<i>PortId</i>	PortId id
----	---------------	-----------

Returns

GPIO inputs level: Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit:

- 0: corresponding pin is read as low
- 1: corresponding pin is read as high

Definition at line 369 of file Gpio_Hal.c.

4.3.3.15 Gpio_Hal_SetHighZ()

```
void Gpio_Hal_SetHighZ (
    uint8 PortId,
    uint8 ChannelId,
    boolean En )
```

Set a pin to HighZ.

Note

Function ID : DES_PORT_API_221

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id
in	<i>En</i>	HighZ enable/disalbe 0:diable 1:enable

Returns

void

Definition at line 427 of file Gpio_Hal.c.

4.3.3.16 Gpio_Hal_SetMuxMode()

```
void Gpio_Hal_SetMuxMode (
    uint8 PortId,
    uint8 ChannelId,
    Port_Hal_MuxType Mux )
```

Configures the pin Mux.

Note

Function ID : DES_PORT_API_204

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id
in	<i>Mux</i>	Pin Mux selection

Returns

void

Definition at line 158 of file Gpio_Hal.c.

4.3.3.17 Gpio_Hal_SetPinDirection()

```
void Gpio_Hal_SetPinDirection (
    uint8 PortId,
    uint8 ChannelId,
    Port_Hal_DataDirectionType Direction )
```

Configure the direction for a certain pin from a port.

Note

Function ID : DES_PORT_API_211

Parameters

in	<i>PortId</i>	port id
----	---------------	---------

Parameters

in	<i>Channel↔ Id</i>	port channel id
in	<i>Direction</i>	The pin direction: <ul style="list-style-type: none"> • GPIO_INPUT_DIRECTION: corresponding pin is set to input • GPIO_OUTPUT_DIRECTION: corresponding pin is set to output • GPIO_UNSPECIFIED_DIRECTION: corresponding pin is not set input and output

Returns

void

Definition at line 268 of file Gpio_Hal.c.

4.3.3.18 Gpio_Hal_SetPinIntSel()

```
Hal_StatusType Gpio_Hal_SetPinIntSel (
    uint8 PortId,
    uint8 ChannelId,
    Port_Hal_InterruptConfigType IntConfig )
```

Configures the port pin interrupt/DMA request.

Note

Function ID : DES_PORT_API_205

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id
in	<i>IntConfig</i>	Interrupt configuration

Returns

function execution status 0 means SUCCESS, other means fail

Definition at line 167 of file Gpio_Hal.c.

4.3.3.19 Gpio_Hal_SetPins()

```
void Gpio_Hal_SetPins (
    uint8 PortId,
    Gpio_Hal_ChannelType Pins )
```

Set Gpio Pins.

Note

Function ID : DES_PORT_API_226

Parameters

in	<i>PortId</i>	Gpio PortId
in	<i>Pins</i>	Pin mask of bits to be Set. Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit: <ul style="list-style-type: none">• 0: corresponding pin is unaffected• 1: corresponding pin is toggled

Returns

void

Definition at line 494 of file Gpio_Hal.c.

4.3.3.20 Gpio_Hal_SetPinsDirection()

```
void Gpio_Hal_SetPinsDirection (
    uint8 PortId,
    uint32 PinsDir )
```

Set the pins directions configuration for a port.

Note

Function ID : DES_PORT_API_212

Parameters

in	<i>PortId</i>	Port id
in	<i>PinsDir</i>	Pin mask where each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit: <ul style="list-style-type: none">• 0: corresponding pin is set to input• 1: corresponding pin is set to output

Returns

void

Definition at line 278 of file Gpio_Hal.c.

4.3.3.21 Gpio_Hal_SetPullSel()

```
void Gpio_Hal_SetPullSel (
    uint8 PortId,
    uint8 ChannelId,
    Port_Hal_PullConfigType PullConfig )
```

Configures the internal pull-up/down resistor.

Note

Function ID : DES_PORT_API_203
Service ID : NA

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id
in	<i>PullConfig</i>	The pull pull-up/down resistor configuration

Returns

void

Definition at line 149 of file Gpio_Hal.c.

4.3.3.22 Gpio_Hal_TogglePin()

```
void Gpio_Hal_TogglePin (
    uint8 PortId,
    uint8 ChannelId )
```

Toggle a pin output level.

Note

Function ID : DES_PORT_API_217

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id

Returns

void

Definition at line 381 of file Gpio_Hal.c.

4.3.3.23 Gpio_Hal_TogglePins()

```
void Gpio_Hal_TogglePins (
    uint8 PortId,
    Gpio_Hal_ChannelType Pins )
```

Toggle pins output level.

Note

Function ID : DES_PORT_API_218

Parameters

in	<i>PortId</i>	PortId id
in	<i>Pins</i>	Pin mask of bits to be toggled. Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit: <ul style="list-style-type: none">• 0: corresponding pin is unaffected• 1: corresponding pin is toggled

Returns

void

Definition at line 393 of file Gpio_Hal.c.

4.3.3.24 Gpio_Hal_WritePin()

```
void Gpio_Hal_WritePin (
    uint8 PortId,
    uint8 ChannelId,
    Gpio_Hal_LevelType Value )
```

Write a pin of a port with a given value.

Note

Function ID : DES_PORT_API_213

Parameters

in	<i>PortId</i>	Port id
in	<i>ChannelId</i>	port channel id
in	<i>Value</i>	Pin value to be written <ul style="list-style-type: none">• 0: corresponding pin is set to low• 1: corresponding pin is set to high

Returns

void

Definition at line 316 of file Gpio_Hal.c.

4.3.3.25 Gpio_Hal_WritePins()

```
void Gpio_Hal_WritePins (
    uint8 PortId,
    Gpio_Hal_ChannelType Pins )
```

Write all pins of a port.

Note

Function ID : DES_PORT_API_214

Parameters

in	<i>PortId</i>	Port id
in	<i>Pins</i>	Pin mask to be written <ul style="list-style-type: none">• 0: corresponding pin is set to low• 1: corresponding pin is set to high

Returns

void

Definition at line 335 of file Gpio_Hal.c.

4.3.3.26 ISR() [1/5]

```
ISR (
    PORTA_IRQHandler )
```

Definition at line 501 of file Gpio_Hal.c.

4.3.3.27 ISR() [2/5]

```
ISR (
    PORTB_IRQHandler )
```

Definition at line 506 of file Gpio_Hal.c.

4.3.3.28 ISR() [3/5]

```
ISR (
    PORTC_IRQHandler )
```

Definition at line 511 of file Gpio_Hal.c.

4.3.3.29 ISR() [4/5]

```
ISR (
    PORTD_IRQHandler )
```

Definition at line 516 of file Gpio_Hal.c.

4.3.3.30 ISR() [5/5]

```
ISR (
    PORTE_IRQHandler )
```

Definition at line 521 of file Gpio_Hal.c.

4.4 Gpio_Hal.h File Reference

This file provides extern Hal Gpio api.

```
#include "Gpio_Hal_Types.h"
```

Macros

- #define [PORTID_A](#) (0u)
PORTA of instances module.
- #define [PORTID_B](#) (1u)
PORTB of instances module.
- #define [PORTID_C](#) (2u)
PORTC of instances module.
- #define [PORTID_D](#) (3u)
PORTD of instances module.
- #define [PORTID_E](#) (4u)
PORTE of instances module.

Functions

- void [Gpio_Hal_Channellnit](#) (uint8 PinCount, const [Gpio_Hal_SettingsConfigType](#) Config[])
Initializes the gpio with the given configuration structure.
- void [Gpio_Hal_SetPullSel](#) (uint8 PortId, uint8 ChannelId, [Port_Hal_PullConfigType](#) PullConfig)
Configures the internal pull-up/down resistor.
- void [Gpio_Hal_SetMuxMode](#) (uint8 PortId, uint8 ChannelId, [Port_Hal_MuxType](#) Mux)
Configures the pin Mux.
- Hal_StatusType [Gpio_Hal_SetPinIntSel](#) (uint8 PortId, uint8 ChannelId, [Port_Hal_InterruptConfigType](#) IntConfig)
Configures the port pin interrupt/DMA request.
- [Port_Hal_InterruptConfigType](#) [Gpio_Hal_GetPinIntSel](#) (uint8 PortId, uint8 ChannelId)
Gets the current port pin interrupt/DMA request configuration.
- void [Gpio_Hal_EnableDigitalFilter](#) (uint8 PortId, uint8 ChannelId, boolean En)
Enables digital filter for digital pin Mux.
- void [Gpio_Hal_ConfigDigitalFilter](#) (uint8 PortId, const [Port_Hal_DigitalFilterCfgType](#) *Config)
Configures digital filter for port with given configuration.
- [Port_Hal_DataDirectionType](#) [Gpio_Hal_GetPinDirection](#) (uint8 PortId, uint8 ChannelId)
Get the pins directions configuration for a pin.
- [Gpio_Hal_ChannelType](#) [Gpio_Hal_GetPinsDirection](#) (uint8 PortId)
Get the pins directions configuration for a port.
- void [Gpio_Hal_SetPinDirection](#) (uint8 PortId, uint8 ChannelId, [Port_Hal_DataDirectionType](#) Direction)
Configure the direction for a certain pin from a port.
- void [Gpio_Hal_SetPinsDirection](#) (uint8 PortId, uint32 PinsDir)
Set the pins directions configuration for a port.
- void [Gpio_Hal_WritePin](#) (uint8 PortId, uint8 ChannelId, [Gpio_Hal_LevelType](#) Value)
Write a pin of a port with a given value.
- void [Gpio_Hal_WritePins](#) (uint8 PortId, [Gpio_Hal_ChannelType](#) Pins)
Write all pins of a port.
- [Gpio_Hal_LevelType](#) [Gpio_Hal_GetPinOutputLevel](#) (uint8 PortId, uint8 ChannelId)
Get the current output level configuration from a pin.
- [Gpio_Hal_ChannelType](#) [Gpio_Hal_GetPinsOutputLevel](#) (uint8 PortId)
Get the current output level configuration from a port.
- void [Gpio_Hal_TogglePin](#) (uint8 PortId, uint8 ChannelId)
Toggle a pin output level.
- void [Gpio_Hal_TogglePins](#) (uint8 PortId, [Gpio_Hal_ChannelType](#) Pins)
Toggle pins output level.
- [Gpio_Hal_LevelType](#) [Gpio_Hal_ReadPin](#) (uint8 PortId, uint8 ChannelId)
Read a pin input level.
- [Gpio_Hal_ChannelType](#) [Gpio_Hal_ReadPins](#) (uint8 PortId)
Read pins input level.
- void [Gpio_Hal_SetHighZ](#) (uint8 PortId, uint8 ChannelId, boolean En)
Set a pin to HighZ.
- void [Gpio_Hal_InstallCallback](#) (uint8 PortId, const [Port_CallbackType](#) Function)
Install GPIO interrupt callback function.
- void [Gpio_Hal_ClearIntStatus](#) (uint8 PortId, uint8 ChannelId)
Clear Gpio interrupt status.
- uint8 [Gpio_Hal_GetIntStatus](#) (uint8 PortId, uint8 ChannelId)
Get Gpio interrupt status.
- void [Gpio_Hal_ClearPins](#) (uint8 PortId, [Gpio_Hal_ChannelType](#) Pins)
Clear Gpio Pins.
- void [Gpio_Hal_SetPins](#) (uint8 PortId, [Gpio_Hal_ChannelType](#) Pins)
Set Gpio Pins.

4.4.1 Detailed Description

This file provides extern Hal Gpio api.

4.4.2 Macro Definition Documentation

4.4.2.1 PORTID_A

```
#define PORTID_A (0u)
```

PORTA of instances module.

Definition at line 52 of file Gpio_Hal.h.

4.4.2.2 PORTID_B

```
#define PORTID_B (1u)
```

PORTB of instances module.

Definition at line 54 of file Gpio_Hal.h.

4.4.2.3 PORTID_C

```
#define PORTID_C (2u)
```

PORTC of instances module.

Definition at line 56 of file Gpio_Hal.h.

4.4.2.4 PORTID_D

```
#define PORTID_D (3u)
```

PORTD of instances module.

Definition at line 58 of file Gpio_Hal.h.

4.4.2.5 PORTID_E

```
#define PORTID_E (4u)
```

PORTE of instances module.

Definition at line 60 of file Gpio_Hal.h.

4.4.3 Function Documentation

4.4.3.1 Gpio_Hal_ChannelInit()

```
void Gpio_Hal_ChannelInit (
    uint8 PinCount,
    const Gpio_Hal_SettingsConfigType Config[] )
```

Initializes the gpio with the given configuration structure.

Note

Function ID : DES_PORT_API_202
Service ID : NA

Parameters

in	<i>PinCount</i>	The number of configured pins in structure
in	<i>Config</i>	The configuration structure

Returns

void

Definition at line 97 of file Gpio_Hal.c.

4.4.3.2 Gpio_Hal_ClearIntStatus()

```
void Gpio_Hal_ClearIntStatus (
    uint8 PortId,
    uint8 ChannelId )
```

Clear Gpio interrupt status.

Note

Function ID : DES_PORT_API_223

Parameters

in	<i>PortId</i>	Gpio PortId
in	<i>Channel↔ Id</i>	Gpio channel id

Returns

void

Definition at line 470 of file Gpio_Hal.c.

4.4.3.3 Gpio_Hal_ClearPins()

```
void Gpio_Hal_ClearPins (
    uint8 PortId,
    Gpio_Hal_ChannelType Pins )
```

Clear Gpio Pins.

Note

Function ID : DES_PORT_API_225

Parameters

in	<i>Port↔ Id</i>	Gpio PortId
in	<i>Pins</i>	Pin mask of bits to be Clear. Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit: <ul style="list-style-type: none">• 0: corresponding pin is unaffected• 1: corresponding pin is toggled

Returns

void

Definition at line 488 of file Gpio_Hal.c.

4.4.3.4 Gpio_Hal_ConfigDigitalFilter()

```
void Gpio_Hal_ConfigDigitalFilter (
    uint8 PortId,
    const Port_Hal_DigitalFilterCfgType * Config )
```

Configures digital filter for port with given configuration.

Note

Function ID : DES_PORT_API_208

Parameters

in	<i>PortId</i>	port id
in	<i>Config</i>	The digital filter configuration struct

Returns

void

Definition at line 214 of file Gpio_Hal.c.

4.4.3.5 Gpio_Hal_EnableDigitalFilter()

```
void Gpio_Hal_EnableDigitalFilter (
    uint8 PortId,
    uint8 ChannelId,
    boolean En )
```

Enables digital filter for digital pin Mux.

Note

Function ID : DES_PORT_API_207

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id
in	<i>En</i>	Digital Filter enable/disalbe 0:diable 1:enable

Returns

void

Definition at line 205 of file Gpio_Hal.c.

4.4.3.6 Gpio_Hal_GetIntStatus()

```
uint8 Gpio_Hal_GetIntStatus (
    uint8 PortId,
    uint8 ChannelId )
```

Get Gpio interrupt status.

Note

Function ID : DES_PORT_API_224

Parameters

in	<i>PortId</i>	Gpio PortId
in	<i>Channel↔ Id</i>	Gpio channel id

Returns

Int status

Definition at line 478 of file Gpio_Hal.c.

4.4.3.7 Gpio_Hal_GetPinDirection()

```
Port_Hal_DataDirectionType Gpio_Hal_GetPinDirection (
    uint8 PortId,
    uint8 ChannelId )
```

Get the pins directions configuration for a pin.

Note

Function ID : DES_PORT_API_209

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id

Returns

the pin direction input or onput

Definition at line 233 of file Gpio_Hal.c.

4.4.3.8 Gpio_Hal_GetPinIntSel()

```
Port_Hal_InterruptConfigType Gpio_Hal_GetPinIntSel (
    uint8 PortId,
    uint8 ChannelId )
```

Gets the current port pin interrupt/DMA request configuration.

Note

Function ID : DES_PORT_API_206

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↵ Id</i>	port channel id

Returns

Interrupt configuration

Definition at line 189 of file Gpio_Hal.c.

4.4.3.9 Gpio_Hal_GetPinOutputLevel()

```
Gpio_Hal_LevelType Gpio_Hal_GetPinOutputLevel (
    uint8 PortId,
    uint8 ChannelId )
```

Get the current output level configuration from a pin.

Note

Function ID : DES_PORT_API_219

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↵ Id</i>	port channel id

Returns

the pin output level

Definition at line 401 of file Gpio_Hal.c.

4.4.3.10 Gpio_Hal_GetPinsDirection()

```
Gpio_Hal_ChannelType Gpio_Hal_GetPinsDirection (
    uint8 PortId )
```

Get the pins directions configuration for a port.

Note

Function ID : DES_PORT_API_210

Parameters

in	<i>Port↔ Id</i>	port id
----	---------------------	---------

Returns

GPIO directions: Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit:

- 0: corresponding pin is set to input
- 1: corresponding pin is set to output

Definition at line 258 of file Gpio_Hal.c.

4.4.3.11 Gpio_Hal_GetPinsOutputLevel()

```
Gpio_Hal_ChannelType Gpio_Hal_GetPinsOutputLevel (
    uint8 PortId )
```

Get the current output level configuration from a port.

Note

Function ID : DES_PORT_API_220

Parameters

in	<i>Port↔ Id</i>	Port id
----	---------------------	---------

Returns

GPIO outputs level configuration: Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit:

- 0: corresponding pin is set to low
- 1: corresponding pin is set to high

Definition at line 415 of file Gpio_Hal.c.

4.4.3.12 Gpio_Hal_InstallCallback()

```
void Gpio_Hal_InstallCallback (
    uint8 PortId,
    const Port_CallbackType Function )
```

Install GPIO interrupt callback function.

Note

Function ID : DES_PORT_API_222

Parameters

in	<i>PortId</i>	GPIO PortId
in	<i>Function</i>	The GPIO interrupt callback function to be installed

Returns

void

Definition at line 462 of file Gpio_Hal.c.

4.4.3.13 Gpio_Hal_ReadPin()

```
Gpio_Hal_LevelType Gpio_Hal_ReadPin (
    uint8 PortId,
    uint8 ChannelId )
```

Read a pin input level.

Note

Function ID : DES_PORT_API_215

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id

Returns

GPIO inputs level

Definition at line 343 of file Gpio_Hal.c.

4.4.3.14 Gpio_Hal_ReadPins()

```
Gpio_Hal_ChannelType Gpio_Hal_ReadPins (
    uint8 PortId )
```

Read pins input level.

Note

Function ID : DES_PORT_API_216

Parameters

in	<i>PortId</i>	PortId id
----	---------------	-----------

Returns

GPIO inputs level: Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit:

- 0: corresponding pin is read as low
- 1: corresponding pin is read as high

Definition at line 369 of file Gpio_Hal.c.

4.4.3.15 Gpio_Hal_SetHighZ()

```
void Gpio_Hal_SetHighZ (
    uint8 PortId,
    uint8 ChannelId,
    boolean En )
```

Set a pin to HighZ.

Note

Function ID : DES_PORT_API_221

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id
in	<i>En</i>	HighZ enable/disalbe 0:disable 1:enable

Returns

void

Definition at line 427 of file Gpio_Hal.c.

4.4.3.16 Gpio_Hal_SetMuxMode()

```
void Gpio_Hal_SetMuxMode (
    uint8 PortId,
    uint8 ChannelId,
    Port_Hal_MuxType Mux )
```

Configures the pin Mux.

Note

Function ID : DES_PORT_API_204

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id
in	<i>Mux</i>	Pin Mux selection

Returns

void

Definition at line 158 of file Gpio_Hal.c.

4.4.3.17 Gpio_Hal_SetPinDirection()

```
void Gpio_Hal_SetPinDirection (
    uint8 PortId,
    uint8 ChannelId,
    Port_Hal_DataDirectionType Direction )
```

Configure the direction for a certain pin from a port.

Note

Function ID : DES_PORT_API_211

Parameters

in	<i>PortId</i>	port id
in	<i>Channel↔ Id</i>	port channel id
in	<i>Direction</i>	The pin direction: <ul style="list-style-type: none"> • GPIO_INPUT_DIRECTION: corresponding pin is set to input • GPIO_OUTPUT_DIRECTION: corresponding pin is set to output • GPIO_UNSPECIFIED_DIRECTION: corresponding pin is not set input and output

Returns

void

Definition at line 268 of file Gpio_Hal.c.

4.4.3.18 Gpio_Hal_SetPinIntSel()

```
Hal_StatusType Gpio_Hal_SetPinIntSel (
    uint8 PortId,
```

```
uint8 ChannelId,  
Port_Hal_InterruptConfigType IntConfig )
```

Configures the port pin interrupt/DMA request.

Note

Function ID : DES_PORT_API_205

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id
in	<i>IntConfig</i>	Interrupt configuration

Returns

function execution status 0 means SUCCESS, other means fail

Definition at line 167 of file Gpio_Hal.c.

4.4.3.19 Gpio_Hal_SetPins()

```
void Gpio_Hal_SetPins (  
    uint8 PortId,  
    Gpio_Hal_ChannelType Pins )
```

Set Gpio Pins.

Note

Function ID : DES_PORT_API_226

Parameters

in	<i>PortId</i>	Gpio PortId
in	<i>Pins</i>	Pin mask of bits to be Set. Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit: <ul style="list-style-type: none">• 0: corresponding pin is unaffected• 1: corresponding pin is toggled

Returns

void

Definition at line 494 of file Gpio_Hal.c.

4.4.3.20 Gpio_Hal_SetPinsDirection()

```
void Gpio_Hal_SetPinsDirection (
    uint8 PortId,
    uint32 PinsDir )
```

Set the pins directions configuration for a port.

Note

Function ID : DES_PORT_API_212

Parameters

in	<i>PortId</i>	Port id
in	<i>PinsDir</i>	Pin mask where each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit: <ul style="list-style-type: none">• 0: corresponding pin is set to input• 1: corresponding pin is set to output

Returns

void

Definition at line 278 of file Gpio_Hal.c.

4.4.3.21 Gpio_Hal_SetPullSel()

```
void Gpio_Hal_SetPullSel (
    uint8 PortId,
    uint8 ChannelId,
    Port_Hal_PullConfigType PullConfig )
```

Configures the internal pull-up/down resistor.

Note

Function ID : DES_PORT_API_203

Service ID : NA

Parameters

in	<i>PortId</i>	port id
in	<i>ChannelId</i>	port channel id
in	<i>PullConfig</i>	The pull pull-up/down resistor configuration

Returns

void

Definition at line 149 of file Gpio_Hal.c.

4.4.3.22 Gpio_Hal_TogglePin()

```
void Gpio_Hal_TogglePin (
    uint8 PortId,
    uint8 ChannelId )
```

Toggle a pin output level.

Note

Function ID : DES_PORT_API_217

Parameters

in	PortId	port id
in	ChannelId	port channel id

Returns

void

Definition at line 381 of file Gpio_Hal.c.

4.4.3.23 Gpio_Hal_TogglePins()

```
void Gpio_Hal_TogglePins (
    uint8 PortId,
    Gpio_Hal_ChannelType Pins )
```

Toggle pins output level.

Note

Function ID : DES_PORT_API_218

Parameters

in	PortId	PortId id
in	Pins	Pin mask of bits to be toggled. Each bit represents one pin (LSB is pin 0, MSB is pin 31). For each bit: <ul style="list-style-type: none">0: corresponding pin is unaffected1: corresponding pin is toggled
Generated by Doxygen		

Returns

void

Definition at line 393 of file Gpio_Hal.c.

4.4.3.24 Gpio_Hal_WritePin()

```
void Gpio_Hal_WritePin (
    uint8 PortId,
    uint8 ChannelId,
    Gpio_Hal_LevelType Value )
```

Write a pin of a port with a given value.

Note

Function ID : DES_PORT_API_213

Parameters

in	<i>PortId</i>	Port id
in	<i>ChannelId</i>	port channel id
in	<i>Value</i>	Pin value to be written <ul style="list-style-type: none">• 0: corresponding pin is set to low• 1: corresponding pin is set to high

Returns

void

Definition at line 316 of file Gpio_Hal.c.

4.4.3.25 Gpio_Hal_WritePins()

```
void Gpio_Hal_WritePins (
    uint8 PortId,
    Gpio_Hal_ChannelType Pins )
```

Write all pins of a port.

Note

Function ID : DES_PORT_API_214

Parameters

in	<i>Port↔ Id</i>	Port id
in	<i>Pins</i>	Pin mask to be written <ul style="list-style-type: none"> • 0: corresponding pin is set to low • 1: corresponding pin is set to high

Returns

void

Definition at line 335 of file Gpio_Hal.c.

4.5 Gpio_Hal_Types.h File Reference

```
#include "Device_Register.h"
```

Classes

- struct [Port_Hal_DigitalFilterCfgType](#)
The digital filter configuration.
- struct [Gpio_Hal_SettingsConfigType](#)
Defines the gpio configuration structure.

Typedefs

- typedef uint32 [Gpio_Hal_ChannelType](#)
Type of a gpio channel.
- typedef void(* [Port_CallbackType](#)) (uint8 PortId, uint32 Status)
Type of port interrupt callback.

Enumerations

- enum [Gpio_Hal_LevelType](#) { [GPIO_LOW_LEVEL](#) = 0x0U, [GPIO_HIGH_LEVEL](#) = 0x1U, [GPIO_MAX_LEVEL](#) = 0x2U }
Type of a gpio pin levels.
- enum [Port_Hal_DataDirectionType](#) { [GPIO_INPUT_DIRECTION](#) = 0x0U, [GPIO_OUTPUT_DIRECTION](#) = 0x1U, [GPIO_UNSPECIFIED_DIRECTION](#) = 0x2U }
Type of gpio data direction.
- enum [Port_Hal_PullConfigType](#) { [PORT_INTERNAL_PULL_NOT_ENABLED](#) = 0U, [PORT_INTERNAL_PULL_UP_ENABLED](#) = 1U, [PORT_INTERNAL_PULL_DOWN_ENABLED](#) = 2U }
Internal pull-up/down resistor.
- enum [Port_Hal_DriveStrengthType](#) { [PORT_LOW_DRIVE_STRENGTH](#) = 0U, [PORT_HIGH_DRIVE_STRENGTH](#) = 1U }
Configures the drive strength.

- enum `Port_Hal_MuxType` {
`PORT_MUX_ALTO = 0U`, `PORT_MUX_AS_GPIO = 1U`, `PORT_MUX_ALT2 = 2U`, `PORT_MUX_ALT3 = 3U`,
`PORT_MUX_ALT4 = 4U`, `PORT_MUX_ALT5 = 5U`, `PORT_MUX_ALT6 = 6U`, `PORT_MUX_ALT7 = 7U` }
Configures the pin mux.
- enum `Port_Hal_InterruptConfigType` {
`PORT_DMA_INT_DISABLED = 0x0U`, `PORT_DMA_RISING_EDGE = 0x1U`, `PORT_DMA_FALLING_EDGE = 0x2U`,
`PORT_DMA_EITHER_EDGE = 0x3U`,
`PORT_INT_RISING_EDGE = 0x9U`, `PORT_INT_FALLING_EDGE = 0xAU`, `PORT_INT_EITHER_EDGE = 0xBU` }
Configures the interrupt generation condition.
- enum `Port_Hal_DigitalFilterClockType` { `PORT_DIGITAL_FILTER_BUS_CLOCK = 0U`, `PORT_DIGITAL_FILTER_LSI_128K_CLOCK = 1U` }
Clock source for the digital input filters.

4.5.1 Typedef Documentation

4.5.1.1 Gpio_Hal_ChannelType

```
typedef uint32 Gpio_Hal_ChannelType
```

Type of a gpio channel.

Definition at line 56 of file `Gpio_Hal_Types.h`.

4.5.1.2 Port_CallbackType

```
typedef void(* Port_CallbackType) (uint8 PortId, uint32 Status)
```

Type of port interrupt callback.

Definition at line 59 of file `Gpio_Hal_Types.h`.

4.5.2 Enumeration Type Documentation

4.5.2.1 Gpio_Hal_LevelType

```
enum Gpio_Hal_LevelType
```

Type of a gpio pin levels.

Enumerator

<code>GPIO_LOW_LEVEL</code>	General level 0.
<code>GPIO_HIGH_LEVEL</code>	General level 1.
<code>GPIO_MAX_LEVEL</code>	General unspecified level.

Definition at line 64 of file Gpio_Hal_Types.h.

4.5.2.2 Port_Hal_DataDirectionType

```
enum Port_Hal_DataDirectionType
```

Type of gpio data direction.

Enumerator

GPIO_INPUT_DIRECTION	General purpose input direction.
GPIO_OUTPUT_DIRECTION	General purpose output direction.
GPIO_UNSPECIFIED_DIRECTION	General purpose unspecified direction.

Definition at line 74 of file Gpio_Hal_Types.h.

4.5.2.3 Port_Hal_DigitalFilterClockType

```
enum Port_Hal_DigitalFilterClockType
```

Clock source for the digital input filters.

Enumerator

PORT_DIGITAL_FILTER_BUS_CLOCK	Select bus clock
PORT_DIGITAL_FILTER_LSI_128K_CLOCK	Select LSI 128k clock

Definition at line 132 of file Gpio_Hal_Types.h.

4.5.2.4 Port_Hal_DriveStrengthType

```
enum Port_Hal_DriveStrengthType
```

Configures the drive strength.

Enumerator

PORT_LOW_DRIVE_STRENGTH	low drive strength
PORT_HIGH_DRIVE_STRENGTH	high drive strength

Definition at line 94 of file Gpio_Hal_Types.h.

4.5.2.5 Port_Hal_InterruptConfigType

```
enum Port_Hal_InterruptConfigType
```

Configures the interrupt generation condition.

Enumerator

PORT_DMA_INT_DISABLED	Interrupt/DMA request is disabled.
PORT_DMA_RISING_EDGE	DMA request on rising edge.
PORT_DMA_FALLING_EDGE	DMA request on falling edge.
PORT_DMA_EITHER_EDGE	DMA request on either edge.
PORT_INT_RISING_EDGE	Interrupt on rising edge.
PORT_INT_FALLING_EDGE	Interrupt on falling edge.
PORT_INT_EITHER_EDGE	Interrupt on either edge.

Definition at line 118 of file Gpio_Hal_Types.h.

4.5.2.6 Port_Hal_MuxType

```
enum Port_Hal_MuxType
```

Configures the pin mux.

Enumerator

PORT_MUX_ALT0	gpio is disabled or as an analog funciton
PORT_MUX_AS_GPIO	gpio is configured as GPIO
PORT_MUX_ALT2	chip-specific
PORT_MUX_ALT3	chip-specific
PORT_MUX_ALT4	chip-specific
PORT_MUX_ALT5	chip-specific
PORT_MUX_ALT6	chip-specific
PORT_MUX_ALT7	chip-specific

Definition at line 103 of file Gpio_Hal_Types.h.

4.5.2.7 Port_Hal_PullConfigType

```
enum Port_Hal_PullConfigType
```

Internal pull-up/down resistor.

Enumerator

PORT_INTERNAL_PULL_NOT_ENABLED	internal pull-down or pull-up resistor is not enabled.
PORT_INTERNAL_PULL_UP_ENABLED	internal pull-down resistor is enabled.
PORT_INTERNAL_PULL_DOWN_ENABLED	internal pull-up resistor is enabled.

Definition at line 84 of file Gpio_Hal_Types.h.

Index

AC784xx_API_Reference_Manual_GPIO.pdf, [7](#)

AC784xx_Gpio_Reg.h, [7](#)

 Gpio_Reg_GetPIDR, [9](#)

 Gpio_Reg_GetPIER, [9](#)

 Gpio_Reg_GetPIOR, [9](#)

 Gpio_Reg_GetPODR, [10](#)

 Gpio_Reg_GetPOER, [10](#)

 Gpio_Reg_GetPROR, [11](#)

 Gpio_Reg_GetPSOR, [11](#)

 Gpio_Reg_ModifyPIER, [12](#)

 Gpio_Reg_ModifyPODR, [12](#)

 Gpio_Reg_ModifyPOER, [13](#)

 Gpio_Reg_SetPIDR, [13](#)

 Gpio_Reg_SetPIER, [14](#)

 Gpio_Reg_SetPIOR, [14](#)

 Gpio_Reg_SetPODR, [15](#)

 Gpio_Reg_SetPOER, [15](#)

 Gpio_Reg_SetPROR, [16](#)

 Gpio_Reg_SetPSOR, [16](#)

 Port_Reg_ClearInterruptFlag, [17](#)

 Port_Reg_EnableDigitalFilter, [17](#)

 Port_Reg_GetDFCR, [18](#)

 Port_Reg_GetDFER, [18](#)

 Port_Reg_GetDFWR, [19](#)

 Port_Reg_GetISFR, [20](#)

 Port_Reg_GetInterruptFlag, [19](#)

 Port_Reg_GetPCR, [20](#)

 Port_Reg_SetDFCR, [21](#)

 Port_Reg_SetDFER, [21](#)

 Port_Reg_SetDFWR, [22](#)

 Port_Reg_SetDriverStrength, [22](#)

 Port_Reg_SetISFR, [23](#)

 Port_Reg_SetInterruptMode, [23](#)

 Port_Reg_SetPCR, [24](#)

 Port_Reg_SetPinLock, [24](#)

 Port_Reg_SetPinMux, [25](#)

 Port_Reg_SetPullConfig, [25](#)

CHANNEL_NUM_MAX

 Gpio_Hal.c, [27](#)

ChannelId

 Gpio_Hal_SettingsConfigType, [3](#)

Clock

 Port_Hal_DigitalFilterCfgType, [6](#)

Direction

 Gpio_Hal_SettingsConfigType, [3](#)

DriveStrength

 Gpio_Hal_SettingsConfigType, [4](#)

Gpio_Hal.c, [26](#)

 CHANNEL_NUM_MAX, [27](#)

 Gpio_Hal_ChannelInit, [28](#)

 Gpio_Hal_ClearIntStatus, [29](#)

 Gpio_Hal_ClearPins, [29](#)

 Gpio_Hal_ConfigDigitalFilter, [30](#)

 Gpio_Hal_EnableDigitalFilter, [30](#)

 Gpio_Hal_GetIntStatus, [31](#)

 Gpio_Hal_GetPinDirection, [31](#)

 Gpio_Hal_GetPinIntSel, [32](#)

 Gpio_Hal_GetPinOutputLevel, [32](#)

 Gpio_Hal_GetPinsDirection, [33](#)

 Gpio_Hal_GetPinsOutputLevel, [33](#)

 Gpio_Hal_InstallCallback, [34](#)

 Gpio_Hal_ReadPin, [34](#)

 Gpio_Hal_ReadPins, [35](#)

 Gpio_Hal_SetHighZ, [35](#)

 Gpio_Hal_SetMuxMode, [36](#)

 Gpio_Hal_SetPinDirection, [36](#)

 Gpio_Hal_SetPinIntSel, [37](#)

 Gpio_Hal_SetPins, [37](#)

 Gpio_Hal_SetPinsDirection, [38](#)

 Gpio_Hal_SetPullSel, [38](#)

 Gpio_Hal_TogglePin, [39](#)

 Gpio_Hal_TogglePins, [39](#)

 Gpio_Hal_WritePin, [40](#)

 Gpio_Hal_WritePins, [41](#)

ISR, [41](#), [42](#)

MODIFY_VALUE, [28](#)

PIN_NUM_MAX, [28](#)

Gpio_Hal.h, [42](#)

 Gpio_Hal_ChannelInit, [45](#)

 Gpio_Hal_ClearIntStatus, [45](#)

 Gpio_Hal_ClearPins, [46](#)

 Gpio_Hal_ConfigDigitalFilter, [46](#)

 Gpio_Hal_EnableDigitalFilter, [47](#)

 Gpio_Hal_GetIntStatus, [47](#)

 Gpio_Hal_GetPinDirection, [48](#)

 Gpio_Hal_GetPinIntSel, [48](#)

 Gpio_Hal_GetPinOutputLevel, [49](#)

 Gpio_Hal_GetPinsDirection, [49](#)

 Gpio_Hal_GetPinsOutputLevel, [50](#)

 Gpio_Hal_InstallCallback, [50](#)

 Gpio_Hal_ReadPin, [51](#)

 Gpio_Hal_ReadPins, [51](#)

 Gpio_Hal_SetHighZ, [52](#)

 Gpio_Hal_SetMuxMode, [52](#)

 Gpio_Hal_SetPinDirection, [53](#)

 Gpio_Hal_SetPinIntSel, [53](#)

 Gpio_Hal_SetPins, [54](#)

 Gpio_Hal_SetPinsDirection, [54](#)

 Gpio_Hal_SetPullSel, [55](#)

 Gpio_Hal_TogglePin, [56](#)

 Gpio_Hal_TogglePins, [56](#)

 Gpio_Hal_WritePin, [57](#)

- Gpio_Hal_WritePins, [57](#)
- PORTID_A, [44](#)
- PORTID_B, [44](#)
- PORTID_C, [44](#)
- PORTID_D, [44](#)
- PORTID_E, [44](#)
- Gpio_Hal_ChannelInit
 - Gpio_Hal.c, [28](#)
 - Gpio_Hal.h, [45](#)
- Gpio_Hal_ChannelType
 - Gpio_Hal_Types.h, [59](#)
- Gpio_Hal_ClearIntStatus
 - Gpio_Hal.c, [29](#)
 - Gpio_Hal.h, [45](#)
- Gpio_Hal_ClearPins
 - Gpio_Hal.c, [29](#)
 - Gpio_Hal.h, [46](#)
- Gpio_Hal_ConfigDigitalFilter
 - Gpio_Hal.c, [30](#)
 - Gpio_Hal.h, [46](#)
- Gpio_Hal_EnableDigitalFilter
 - Gpio_Hal.c, [30](#)
 - Gpio_Hal.h, [47](#)
- Gpio_Hal_GetIntStatus
 - Gpio_Hal.c, [31](#)
 - Gpio_Hal.h, [47](#)
- Gpio_Hal_GetPinDirection
 - Gpio_Hal.c, [31](#)
 - Gpio_Hal.h, [48](#)
- Gpio_Hal_GetPinIntSel
 - Gpio_Hal.c, [32](#)
 - Gpio_Hal.h, [48](#)
- Gpio_Hal_GetPinOutputLevel
 - Gpio_Hal.c, [32](#)
 - Gpio_Hal.h, [49](#)
- Gpio_Hal_GetPinsDirection
 - Gpio_Hal.c, [33](#)
 - Gpio_Hal.h, [49](#)
- Gpio_Hal_GetPinsOutputLevel
 - Gpio_Hal.c, [33](#)
 - Gpio_Hal.h, [50](#)
- Gpio_Hal_InstallCallback
 - Gpio_Hal.c, [34](#)
 - Gpio_Hal.h, [50](#)
- Gpio_Hal_LevelType
 - Gpio_Hal_Types.h, [59](#)
- Gpio_Hal_ReadPin
 - Gpio_Hal.c, [34](#)
 - Gpio_Hal.h, [51](#)
- Gpio_Hal_ReadPins
 - Gpio_Hal.c, [35](#)
 - Gpio_Hal.h, [51](#)
- Gpio_Hal_SetHighZ
 - Gpio_Hal.c, [35](#)
 - Gpio_Hal.h, [52](#)
- Gpio_Hal_SetMuxMode
 - Gpio_Hal.c, [36](#)
 - Gpio_Hal.h, [52](#)
- Gpio_Hal_SetPinDirection
 - Gpio_Hal.c, [36](#)
 - Gpio_Hal.h, [53](#)
- Gpio_Hal_SetPinIntSel
 - Gpio_Hal.c, [37](#)
 - Gpio_Hal.h, [53](#)
- Gpio_Hal_SetPins
 - Gpio_Hal.c, [37](#)
 - Gpio_Hal.h, [54](#)
- Gpio_Hal_SetPinsDirection
 - Gpio_Hal.c, [38](#)
 - Gpio_Hal.h, [54](#)
- Gpio_Hal_SetPullSel
 - Gpio_Hal.c, [38](#)
 - Gpio_Hal.h, [55](#)
- Gpio_Hal_SettingsConfigType, [3](#)
 - ChannelId, [3](#)
 - Direction, [3](#)
 - DriveStrength, [4](#)
 - InitValue, [4](#)
 - IntConfig, [4](#)
 - Mux, [4](#)
 - PinLock, [4](#)
 - PortId, [5](#)
 - PullConfig, [5](#)
- Gpio_Hal_TogglePin
 - Gpio_Hal.c, [39](#)
 - Gpio_Hal.h, [56](#)
- Gpio_Hal_TogglePins
 - Gpio_Hal.c, [39](#)
 - Gpio_Hal.h, [56](#)
- Gpio_Hal_Types.h, [58](#)
 - Gpio_Hal_ChannelType, [59](#)
 - Gpio_Hal_LevelType, [59](#)
 - Port_CallbackType, [59](#)
 - Port_Hal_DataDirectionType, [60](#)
 - Port_Hal_DigitalFilterClockType, [60](#)
 - Port_Hal_DriveStrengthType, [60](#)
 - Port_Hal_InterruptConfigType, [60](#)
 - Port_Hal_MuxType, [61](#)
 - Port_Hal_PullConfigType, [61](#)
- Gpio_Hal_WritePin
 - Gpio_Hal.c, [40](#)
 - Gpio_Hal.h, [57](#)
- Gpio_Hal_WritePins
 - Gpio_Hal.c, [41](#)
 - Gpio_Hal.h, [57](#)
- Gpio_Reg_GetPIDR
 - AC784xx_Gpio_Reg.h, [9](#)
- Gpio_Reg_GetPIER
 - AC784xx_Gpio_Reg.h, [9](#)
- Gpio_Reg_GetPIOR
 - AC784xx_Gpio_Reg.h, [9](#)
- Gpio_Reg_GetPODR
 - AC784xx_Gpio_Reg.h, [10](#)
- Gpio_Reg_GetPOER
 - AC784xx_Gpio_Reg.h, [10](#)
- Gpio_Reg_GetPROR
 - AC784xx_Gpio_Reg.h, [11](#)
- Gpio_Reg_GetPSOR
 - AC784xx_Gpio_Reg.h, [11](#)
- Gpio_Reg_ModifyPIER
 - AC784xx_Gpio_Reg.h, [12](#)
- Gpio_Reg_ModifyPODR

- AC784xx_Gpio_Reg.h, [12](#)
- Gpio_Reg_ModifyPOER
 - AC784xx_Gpio_Reg.h, [13](#)
- Gpio_Reg_SetPIDR
 - AC784xx_Gpio_Reg.h, [13](#)
- Gpio_Reg_SetPIER
 - AC784xx_Gpio_Reg.h, [14](#)
- Gpio_Reg_SetPIOR
 - AC784xx_Gpio_Reg.h, [14](#)
- Gpio_Reg_SetPODR
 - AC784xx_Gpio_Reg.h, [15](#)
- Gpio_Reg_SetPOER
 - AC784xx_Gpio_Reg.h, [15](#)
- Gpio_Reg_SetPROR
 - AC784xx_Gpio_Reg.h, [16](#)
- Gpio_Reg_SetPSOR
 - AC784xx_Gpio_Reg.h, [16](#)
- ISR
 - Gpio_Hal.c, [41](#), [42](#)
- InitValue
 - Gpio_Hal_SettingsConfigType, [4](#)
- IntConfig
 - Gpio_Hal_SettingsConfigType, [4](#)
- MODIFY_VALUE
 - Gpio_Hal.c, [28](#)
- Mux
 - Gpio_Hal_SettingsConfigType, [4](#)
- PIN_NUM_MAX
 - Gpio_Hal.c, [28](#)
- PORTID_A
 - Gpio_Hal.h, [44](#)
- PORTID_B
 - Gpio_Hal.h, [44](#)
- PORTID_C
 - Gpio_Hal.h, [44](#)
- PORTID_D
 - Gpio_Hal.h, [44](#)
- PORTID_E
 - Gpio_Hal.h, [44](#)
- PinLock
 - Gpio_Hal_SettingsConfigType, [4](#)
- PinMask
 - Port_Hal_DigitalFilterCfgType, [6](#)
- Port_CallbackType
 - Gpio_Hal_Types.h, [59](#)
- Port_Hal_DataDirectionType
 - Gpio_Hal_Types.h, [60](#)
- Port_Hal_DigitalFilterCfgType, [5](#)
 - Clock, [6](#)
 - PinMask, [6](#)
 - Width, [6](#)
- Port_Hal_DigitalFilterClockType
 - Gpio_Hal_Types.h, [60](#)
- Port_Hal_DriveStrengthType
 - Gpio_Hal_Types.h, [60](#)
- Port_Hal_InterruptConfigType
 - Gpio_Hal_Types.h, [60](#)
- Port_Hal_MuxType
 - Gpio_Hal_Types.h, [61](#)
- Port_Hal_PullConfigType
 - Gpio_Hal_Types.h, [61](#)
- Port_Reg_ClearInterruptFlag
 - AC784xx_Gpio_Reg.h, [17](#)
- Port_Reg_EnableDigitalFilter
 - AC784xx_Gpio_Reg.h, [17](#)
- Port_Reg_GetDFCR
 - AC784xx_Gpio_Reg.h, [18](#)
- Port_Reg_GetDFER
 - AC784xx_Gpio_Reg.h, [18](#)
- Port_Reg_GetDFWR
 - AC784xx_Gpio_Reg.h, [19](#)
- Port_Reg_GetISFR
 - AC784xx_Gpio_Reg.h, [20](#)
- Port_Reg_GetInterruptFlag
 - AC784xx_Gpio_Reg.h, [19](#)
- Port_Reg_GetPCR
 - AC784xx_Gpio_Reg.h, [20](#)
- Port_Reg_SetDFCR
 - AC784xx_Gpio_Reg.h, [21](#)
- Port_Reg_SetDFER
 - AC784xx_Gpio_Reg.h, [21](#)
- Port_Reg_SetDFWR
 - AC784xx_Gpio_Reg.h, [22](#)
- Port_Reg_SetDriverStrength
 - AC784xx_Gpio_Reg.h, [22](#)
- Port_Reg_SetISFR
 - AC784xx_Gpio_Reg.h, [23](#)
- Port_Reg_SetInterruptMode
 - AC784xx_Gpio_Reg.h, [23](#)
- Port_Reg_SetPCR
 - AC784xx_Gpio_Reg.h, [24](#)
- Port_Reg_SetPinLock
 - AC784xx_Gpio_Reg.h, [24](#)
- Port_Reg_SetPinMux
 - AC784xx_Gpio_Reg.h, [25](#)
- Port_Reg_SetPullConfig
 - AC784xx_Gpio_Reg.h, [25](#)
- PortId
 - Gpio_Hal_SettingsConfigType, [5](#)
- PullConfig
 - Gpio_Hal_SettingsConfigType, [5](#)
- Width
 - Port_Hal_DigitalFilterCfgType, [6](#)