

AC784xx_DFP CMU

5.1.0

Generated by Doxygen 1.8.13

Contents

1	File Index	1
1.1	File List	1
2	File Documentation	2
2.1	AC784xx_API_Reference_Manual_CMU.pdf File Reference	2
2.2	AC784xx_Cmu_Reg.h File Reference	2
2.2.1	Detailed Description	2
2.2.2	Function Documentation	3
2.2.2.1	Cmu_Reg_ClearStatus()	3
2.2.2.2	Cmu_Reg_Enable()	3
2.2.2.3	Cmu_Reg_GetEnable()	4
2.2.2.4	Cmu_Reg_GetHighThreshold()	4
2.2.2.5	Cmu_Reg_GetLowThreshold()	4
2.2.2.6	Cmu_Reg_GetStatus()	5
2.2.2.7	Cmu_Reg_GetWindowLen()	5
2.2.2.8	Cmu_Reg_SetHighThreshold()	6
2.2.2.9	Cmu_Reg_SetLowThreshold()	6
2.2.2.10	Cmu_Reg_SetWindowLen()	7
2.3	Cmu_Hal.c File Reference	7
2.3.1	Detailed Description	8
2.3.2	Macro Definition Documentation	8
2.3.2.1	CMU_MODULE_COUNT	8
2.3.2.2	CMU_THRESHOLD_MAX_VALUE	8
2.3.2.3	CMU_WAIT_TIMEOUT	8
2.3.2.4	CMU_WINLEN_MIN_VALUE	9

2.3.3	Enumeration Type Documentation	9
2.3.3.1	Cmu_StateType	9
2.3.4	Function Documentation	9
2.3.4.1	Cmu_Hal_ClearErrStatus()	9
2.3.4.2	Cmu_Hal_Deinit()	10
2.3.4.3	Cmu_Hal_Enable()	10
2.3.4.4	Cmu_Hal_GetErrInfo()	11
2.3.4.5	Cmu_Hal_Init()	11
2.4	Cmu_Hal.h File Reference	11
2.4.1	Detailed Description	12
2.4.2	Function Documentation	12
2.4.2.1	Cmu_Hal_ClearErrStatus()	12
2.4.2.2	Cmu_Hal_Deinit()	13
2.4.2.3	Cmu_Hal_Enable()	14
2.4.2.4	Cmu_Hal_GetErrInfo()	14
2.4.2.5	Cmu_Hal_Init()	15
	Index	17

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

AC784xx_API_Reference_Manual_CMU.pdf	2
AC784xx_Cmu_Reg.h	
Cmu access register inline function definition	2
Cmu_Hal.c	
Cmu hal source file	7
Cmu_Hal.h	
Cmu hal define	11

Chapter 2

File Documentation

2.1 AC784xx_API_Reference_Manual_CMU.pdf File Reference

2.2 AC784xx_Cmu_Reg.h File Reference

cmu access register inline function definition.

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

Functions

- LOCAL_INLINE void [Cmu_Reg_SetWindowLen](#) (uint8 Cmuldx, uint16 WinLen)
Set cmu window length.
- LOCAL_INLINE uint16 [Cmu_Reg_GetWindowLen](#) (uint8 Cmuldx)
Get cmu window length.
- LOCAL_INLINE void [Cmu_Reg_SetHighThreshold](#) (uint8 Cmuldx, uint32 Threshold)
Set cmu high threshold.
- LOCAL_INLINE uint32 [Cmu_Reg_GetHighThreshold](#) (uint8 Cmuldx)
Get cmu high threshold value.
- LOCAL_INLINE void [Cmu_Reg_SetLowThreshold](#) (uint8 Cmuldx, uint32 Threshold)
Set cmu low threshold.
- LOCAL_INLINE uint32 [Cmu_Reg_GetLowThreshold](#) (uint8 Cmuldx)
Get cmu low threshold value.
- LOCAL_INLINE void [Cmu_Reg_Enable](#) (uint8 Cmuldx, boolean IsEnable)
Enable cmu.
- LOCAL_INLINE boolean [Cmu_Reg_GetEnable](#) (uint8 Cmuldx)
Get cmu Enable.
- LOCAL_INLINE uint8 [Cmu_Reg_GetStatus](#) (uint8 Cmuldx)
Get cmu status.
- LOCAL_INLINE void [Cmu_Reg_ClearStatus](#) (uint8 Cmuldx)
Clear cmu status.

2.2.1 Detailed Description

cmu access register inline function definition.

2.2.2 Function Documentation

2.2.2.1 Cmu_Reg_ClearStatus()

```
LOCAL_INLINE void Cmu_Reg_ClearStatus (
    uint8 CmuIdx )
```

Clear cmu status.

Note

Function ID: DES_MCU_API_429

Parameters

in	<i>Cmuldx</i>	index in the CmuBase
----	---------------	----------------------

Returns

void

Definition at line 184 of file AC784xx_Cmu_Reg.h.

2.2.2.2 Cmu_Reg_Enable()

```
LOCAL_INLINE void Cmu_Reg_Enable (
    uint8 CmuIdx,
    boolean IsEnable )
```

Enable cmu.

Note

Function ID: DES_MCU_API_426

Parameters

in	<i>Cmuldx</i>	index in the CmuBase
in	<i>IsEnable</i>	true enable cmu; false disable cmu

Returns

void

Definition at line 148 of file AC784xx_Cmu_Reg.h.

2.2.2.3 Cmu_Reg_GetEnable()

```
LOCAL_INLINE boolean Cmu_Reg_GetEnable (
    uint8 CmuIdx )
```

Get cmu Enable.

Note

Function ID: DES_MCU_API_427

Parameters

in	<i>CmuIdx</i>	index in the CmuBase
----	---------------	----------------------

Returns

cmu is enable

Definition at line 160 of file AC784xx_Cmu_Reg.h.

2.2.2.4 Cmu_Reg_GetHighThreshold()

```
LOCAL_INLINE uint32 Cmu_Reg_GetHighThreshold (
    uint8 CmuIdx )
```

Get cmu high threshold value.

Note

Function ID: DES_MCU_API_423

Parameters

in	<i>CmuIdx</i>	index in the CmuBase
----	---------------	----------------------

Returns

cmu high threshold value

Definition at line 113 of file AC784xx_Cmu_Reg.h.

2.2.2.5 Cmu_Reg_GetLowThreshold()

```
LOCAL_INLINE uint32 Cmu_Reg_GetLowThreshold (
    uint8 CmuIdx )
```

Get cmu low threshold value.

Note

Function ID: DES_MCU_API_425

Parameters

in	<i>Cmuldx</i>	index in the CmuBase
----	---------------	----------------------

Returns

cmu low threshold value

Definition at line 136 of file AC784xx_Cmu_Reg.h.

2.2.2.6 Cmu_Reg_GetStatus()

```
LOCAL_INLINE uint8 Cmu_Reg_GetStatus (
    uint8 CmuIdx )
```

Get cmu status.

Note

Function ID: DES_MCU_API_428

Parameters

in	<i>Cmuldx</i>	index in the CmuBase
----	---------------	----------------------

Returns

cmu status

Definition at line 173 of file AC784xx_Cmu_Reg.h.

2.2.2.7 Cmu_Reg_GetWindowLen()

```
LOCAL_INLINE uint16 Cmu_Reg_GetWindowLen (
    uint8 CmuIdx )
```

Get cmu window length.

Note

Function ID: DES_MCU_API_421

Parameters

in	<i>Cmuldx</i>	index in the CmuBase
----	---------------	----------------------

Returns

cmu window length

Definition at line 90 of file AC784xx_Cmu_Reg.h.

2.2.2.8 Cmu_Reg_SetHighThreshold()

```
LOCAL_INLINE void Cmu_Reg_SetHighThreshold (
    uint8 CmuIdx,
    uint32 Threshold )
```

Set cmu high threshold.

Note

Function ID: DES_MCU_API_422

Parameters

in	<i>Cmuldx</i>	index in the CmuBase
in	<i>Threshold</i>	Threshold value

Returns

void

Definition at line 102 of file AC784xx_Cmu_Reg.h.

2.2.2.9 Cmu_Reg_SetLowThreshold()

```
LOCAL_INLINE void Cmu_Reg_SetLowThreshold (
    uint8 CmuIdx,
    uint32 Threshold )
```

Set cmu low threshold.

Note

Function ID: DES_MCU_API_424

Parameters

in	<i>Cmuldx</i>	index in the CmuBase
in	<i>Threshold</i>	Threshold value

Returns

void

Definition at line 125 of file AC784xx_Cmu_Reg.h.

2.2.2.10 Cmu_Reg_SetWindowLen()

```
LOCAL_INLINE void Cmu_Reg_SetWindowLen (
    uint8 CmuIdx,
    uint16 WinLen )
```

Set cmu window length.

Note

Function ID: DES_MCU_API_420

Parameters

in	<i>Cmuldx</i>	index in the CmuBase
in	<i>WinLen</i>	window length

Returns

void

Definition at line 79 of file AC784xx_Cmu_Reg.h.

2.3 Cmu_Hal.c File Reference

cmu hal source file.

```
#include "Cmu_Hal.h"
#include "AC784xx_Cmu_Reg.h"
#include "OsIf_Critical.h"
#include "OsIf_Irq.h"
```

Macros

- #define [CMU_MODULE_COUNT](#) (3U)
- #define [CMU_WINLEN_MIN_VALUE](#) (7U)
- #define [CMU_THRESHOLD_MAX_VALUE](#) (0xFFFFFU)
- #define [CMU_WAIT_TIMEOUT](#) (100000U)

Enumerations

- enum [Cmu_StateType](#) { [CMU_INITED](#) = 0x0U, [CMU_UNINITED](#), [CMU_RUNNING](#), [CMU_ABNORMAL](#) }
cmu state machine

Functions

- Hal_StatusType [Cmu_Hal_Init](#) (Cmu_ModuleIdType Cmuld, const Cmu_CtrlParamType *CtrlParam)
Initialize CMU module.
- void [Cmu_Hal_Deinit](#) (Cmu_ModuleIdType Cmuld)
Deinitialize CMU module.
- Hal_StatusType [Cmu_Hal_Enable](#) (Cmu_ModuleIdType Cmuld, boolean IsEnable)
Enable Cmu.
- void [Cmu_Hal_ClearErrStatus](#) (Cmu_ModuleIdType Cmuld)
Clear Cmu error flag.
- Hal_StatusType [Cmu_Hal_GetErrInfo](#) (Cmu_ModuleIdType Cmuld, Cmu_ErrInfoType *ErrInfo)
Get the error information of CMU.

2.3.1 Detailed Description

cmu hal source file.

2.3.2 Macro Definition Documentation

2.3.2.1 CMU_MODULE_COUNT

```
#define CMU_MODULE_COUNT (3U)
```

Definition at line 53 of file Cmu_Hal.c.

2.3.2.2 CMU_THRESHOLD_MAX_VALUE

```
#define CMU_THRESHOLD_MAX_VALUE (0xFFFFFFFU)
```

Definition at line 59 of file Cmu_Hal.c.

2.3.2.3 CMU_WAIT_TIMEOUT

```
#define CMU_WAIT_TIMEOUT (1000000U)
```

Definition at line 62 of file Cmu_Hal.c.

2.3.2.4 CMU_WINLEN_MIN_VALUE

```
#define CMU_WINLEN_MIN_VALUE (7U)
```

Definition at line 56 of file Cmu_Hal.c.

2.3.3 Enumeration Type Documentation

2.3.3.1 Cmu_StateType

```
enum Cmu_StateType
```

cmu state machine

Enumerator

CMU_INITED	
CMU_UNINITED	
CMU_RUNNING	
CMU_ABNORMAL	

Definition at line 68 of file Cmu_Hal.c.

2.3.4 Function Documentation

2.3.4.1 Cmu_Hal_ClearErrStatus()

```
void Cmu_Hal_ClearErrStatus (
    Cmu_ModuleIdType CmuId )
```

Clear Cmu error flag.

Note

Function ID: DES_MCU_API_404

Parameters

in	<i>CmuId</i>	CMU module id
----	--------------	---------------

Returns

void

Definition at line 245 of file Cmu_Hal.c.

2.3.4.2 Cmu_Hal_Deinit()

```
void Cmu_Hal_Deinit (
    Cmu_ModuleIdType CmuId )
```

Deinitialize CMU module.

Note

Function ID: DES_MCU_API_402

Parameters

in	<i>CmuId</i>	CMU module id
----	--------------	---------------

Returns

void

Definition at line 160 of file Cmu_Hal.c.

2.3.4.3 Cmu_Hal_Enable()

```
Hal_StatusType Cmu_Hal_Enable (
    Cmu_ModuleIdType CmuId,
    boolean IsEnable )
```

Enable Cmu.

Note

Function ID: DES_MCU_API_403

Parameters

in	<i>Cmuld</i>	CMU module id
in	<i>IsEnable</i>	true enable cmu module, false disable cmu module

Returns

Hal_StatusType: Enable success or not, the range is the STATUS_SUCCESS STATUS_ERROR

Definition at line 191 of file Cmu_Hal.c.

2.3.4.4 Cmu_Hal_GetErrInfo()

```
Hal_StatusType Cmu_Hal_GetErrInfo (
    Cmu_ModuleIdType CmuId,
    Cmu_ErrInfoType * ErrInfo )
```

Get the error information of CMU.

Note

Function ID: DES_MCU_API_405

Parameters

in	<i>CmuId</i>	CMU module id
out	<i>ErrInfo</i>	the pointer to the Cmu_ErrInfoType structure, save error message

Returns

Hal_StatusType: Is there an error, STATUS_SUCCESS stands for no error; STATUS_ERROR is a error

Definition at line 261 of file Cmu_Hal.c.

2.3.4.5 Cmu_Hal_Init()

```
Hal_StatusType Cmu_Hal_Init (
    Cmu_ModuleIdType CmuId,
    const Cmu_CtrlParamType * CtrlParam )
```

Initialize CMU module.

Note

Function ID: DES_MCU_API_401

Parameters

in	<i>CmuId</i>	CMU module id
in	<i>CtrlParam</i>	the pointer to the Cmu_CtrlParamType structure

Returns

Hal_StatusType: Initialize success or not, the range is the STATUS_SUCCESS STATUS_ERROR

Definition at line 124 of file Cmu_Hal.c.

2.4 Cmu_Hal.h File Reference

cmu hal define.

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

Functions

- Hal_StatusType [Cmu_Hal_Init](#) (Cmu_ModuleIdType Cmuld, const Cmu_CtrlParamType *CtrlParam)
Initialize CMU module.
- void [Cmu_Hal_Deinit](#) (Cmu_ModuleIdType Cmuld)
Deinitialize CMU module.
- Hal_StatusType [Cmu_Hal_Enable](#) (Cmu_ModuleIdType Cmuld, boolean IsEnable)
Enable Cmu.
- void [Cmu_Hal_ClearErrStatus](#) (Cmu_ModuleIdType Cmuld)
Clear Cmu error flag.
- Hal_StatusType [Cmu_Hal_GetErrInfo](#) (Cmu_ModuleIdType Cmuld, Cmu_ErrInfoType *ErrInfo)
Get the error information of CMU.

2.4.1 Detailed Description

cmu hal define.

2.4.2 Function Documentation

2.4.2.1 Cmu_Hal_ClearErrStatus()

```
void Cmu_Hal_ClearErrStatus (
    Cmu_ModuleIdType CmuId )
```

Clear Cmu error flag.

Note

Function ID: DES_MCU_API_304

Parameters

in	<i>Cmu↔ Id</i>	CMU module id
----	--------------------	---------------

Returns

void

Note

Function ID: DES_MCU_API_404

Parameters

in	<i>Cmu↔ Id</i>	CMU module id
----	--------------------	---------------

Returns

void

Definition at line 245 of file Cmu_Hal.c.

2.4.2.2 Cmu_Hal_Deinit()

```
void Cmu_Hal_Deinit (
    Cmu_ModuleIdType CmuId )
```

Deinitialize CMU module.

Note

Function ID: DES_MCU_API_302

Parameters

in	<i>Cmu↔ Id</i>	CMU module id
----	--------------------	---------------

Returns

void

Note

Function ID: DES_MCU_API_402

Parameters

in	<i>Cmu↔ Id</i>	CMU module id
----	--------------------	---------------

Returns

void

Definition at line 160 of file Cmu_Hal.c.

2.4.2.3 Cmu_Hal_Enable()

```
Hal_StatusType Cmu_Hal_Enable (
    Cmu_ModuleIdType CmuId,
    boolean IsEnable )
```

Enable Cmu.

Note

Function ID: DES_MCU_API_303

Parameters

in	<i>Cmuld</i>	CMU module id
in	<i>IsEnable</i>	true enable cmu module, false disable cmu module

Returns

Hal_StatusType: Enable success or not, the range is the STATUS_SUCCESS STATUS_ERROR

Note

Function ID: DES_MCU_API_403

Parameters

in	<i>Cmuld</i>	CMU module id
in	<i>IsEnable</i>	true enable cmu module, false disable cmu module

Returns

Hal_StatusType: Enable success or not, the range is the STATUS_SUCCESS STATUS_ERROR

Definition at line 191 of file Cmu_Hal.c.

2.4.2.4 Cmu_Hal_GetErrInfo()

```
Hal_StatusType Cmu_Hal_GetErrInfo (
    Cmu_ModuleIdType CmuId,
    Cmu_ErrInfoType * ErrInfo )
```

Get the error information of CMU.

Note

Function ID: DES_MCU_API_305

Parameters

in	<i>CmuId</i>	CMU module id
out	<i>ErrInfo</i>	the pointer to the Cmu_ErrInfoType structure, save error message

Returns

Hal_StatusType: Is there an error, STATUS_SUCCESS stands for no error; STATUS_ERROR is a error

Note

Function ID: DES_MCU_API_405

Parameters

in	<i>CmuId</i>	CMU module id
out	<i>ErrInfo</i>	the pointer to the Cmu_ErrInfoType structure, save error message

Returns

Hal_StatusType: Is there an error, STATUS_SUCCESS stands for no error; STATUS_ERROR is a error

Definition at line 261 of file Cmu_Hal.c.

2.4.2.5 Cmu_Hal_Init()

```
Hal_StatusType Cmu_Hal_Init (
    Cmu_ModuleIdType CmuId,
    const Cmu_CtrlParamType * CtrlParam )
```

Initialize CMU module.

Note

Function ID: DES_MCU_API_301

Parameters

in	<i>Cmuld</i>	CMU module id
in	<i>CtrlParam</i>	the pointer to the Cmu_CtrlParamType structure

Returns

Hal_StatusType: Initialize success or not, the range is the STATUS_SUCCESS STATUS_ERROR

Note

Function ID: DES_MCU_API_401

Parameters

in	<i>Cmuld</i>	CMU module id
in	<i>CtrlParam</i>	the pointer to the Cmu_CtrlParamType structure

Returns

Hal_StatusType: Initialize success or not, the range is the STATUS_SUCCESS STATUS_ERROR

Definition at line 124 of file Cmu_Hal.c.

Index

AC784xx_API_Reference_Manual_CMU.pdf, [2](#)

AC784xx_Cmu_Reg.h, [2](#)

 Cmu_Reg_ClearStatus, [3](#)

 Cmu_Reg_Enable, [3](#)

 Cmu_Reg_GetEnable, [3](#)

 Cmu_Reg_GetHighThreshold, [4](#)

 Cmu_Reg_GetLowThreshold, [4](#)

 Cmu_Reg_GetStatus, [5](#)

 Cmu_Reg_GetWindowLen, [5](#)

 Cmu_Reg_SetHighThreshold, [6](#)

 Cmu_Reg_SetLowThreshold, [6](#)

 Cmu_Reg_SetWindowLen, [7](#)

CMU_MODULE_COUNT

 Cmu_Hal.c, [8](#)

CMU_THRESHOLD_MAX_VALUE

 Cmu_Hal.c, [8](#)

CMU_WAIT_TIMEOUT

 Cmu_Hal.c, [8](#)

CMU_WINLEN_MIN_VALUE

 Cmu_Hal.c, [8](#)

Cmu_Hal.c, [7](#)

 CMU_MODULE_COUNT, [8](#)

 CMU_THRESHOLD_MAX_VALUE, [8](#)

 CMU_WAIT_TIMEOUT, [8](#)

 CMU_WINLEN_MIN_VALUE, [8](#)

 Cmu_Hal_ClearErrStatus, [9](#)

 Cmu_Hal_Deinit, [10](#)

 Cmu_Hal_Enable, [10](#)

 Cmu_Hal_GetErrInfo, [10](#)

 Cmu_Hal_Init, [11](#)

 Cmu_StateType, [9](#)

Cmu_Hal.h, [11](#)

 Cmu_Hal_ClearErrStatus, [12](#)

 Cmu_Hal_Deinit, [13](#)

 Cmu_Hal_Enable, [13](#)

 Cmu_Hal_GetErrInfo, [14](#)

 Cmu_Hal_Init, [15](#)

Cmu_Hal_ClearErrStatus

 Cmu_Hal.c, [9](#)

 Cmu_Hal.h, [12](#)

Cmu_Hal_Deinit

 Cmu_Hal.c, [10](#)

 Cmu_Hal.h, [13](#)

Cmu_Hal_Enable

 Cmu_Hal.c, [10](#)

 Cmu_Hal.h, [13](#)

Cmu_Hal_GetErrInfo

 Cmu_Hal.c, [10](#)

 Cmu_Hal.h, [14](#)

Cmu_Hal_Init

 Cmu_Hal.c, [11](#)

 Cmu_Hal.h, [15](#)

Cmu_Reg_ClearStatus

 AC784xx_Cmu_Reg.h, [3](#)

Cmu_Reg_Enable

 AC784xx_Cmu_Reg.h, [3](#)

Cmu_Reg_GetEnable

 AC784xx_Cmu_Reg.h, [3](#)

Cmu_Reg_GetHighThreshold

 AC784xx_Cmu_Reg.h, [4](#)

Cmu_Reg_GetLowThreshold

 AC784xx_Cmu_Reg.h, [4](#)

Cmu_Reg_GetStatus

 AC784xx_Cmu_Reg.h, [5](#)

Cmu_Reg_GetWindowLen

 AC784xx_Cmu_Reg.h, [5](#)

Cmu_Reg_SetHighThreshold

 AC784xx_Cmu_Reg.h, [6](#)

Cmu_Reg_SetLowThreshold

 AC784xx_Cmu_Reg.h, [6](#)

Cmu_Reg_SetWindowLen

 AC784xx_Cmu_Reg.h, [7](#)

Cmu_StateType

 Cmu_Hal.c, [9](#)