

GLOBAL_ADC

Home

Data Structures

Data Structure Index

Data Fields

Data Structures

Here are the data structures with brief descriptions:

 [GLOBAL_ADC](#)

**Configuration Data structure of
GLOBAL_ADC APP**

 [GLOBAL_ADC_GROUP](#)

**Structure to hold the configuration
information of a group**

GLOBAL_ADC

[Home](#)

[Data Structures](#)

[Data Structure Index](#)

[Data Fields](#)

[Data Fields](#)

GLOBAL_ADC Struct Reference

Detailed Description

Configuration Data structure of **GLOBAL_ADC APP**.

Definition at line **142** of file **GLOBAL_ADC.h**.

```
#include <GLOBAL_ADC.h>
```

Data Fields

`GLOBAL_ADC_GROUP_t *const group_ptrs_array
[XMC_VADC_MAXIML`

`const XMC_VADC_GLOBAL_CONFIG_t
*const global_config_handle`

`XMC_VADC_GLOBAL_t *const module_ptr`

`XMC_VADC_GLOBAL_SHS_t *const global_shs_ptr`

`XMC_VADC_GLOBAL_SHS_CONFIG_t *const global_shscfg`

`GLOBAL_ADC_STATUS_t init_state`

`const bool enable_startup_calib`

Field Documentation

const bool GLOBAL_ADC::enable_startup_calibration

Enable startup calibration for all the converters

Definition at line [157](#) of file **GLOBAL_ADC.h**.

Referenced by **GLOBAL_ADC_Init()**.

const XMC_VADC_GLOBAL_CONFIG_t* const GLOBAL_ADC::glob

This is the pointer to the Global LLD Handle.

Definition at line [147](#) of file **GLOBAL_ADC.h**.

Referenced by **GLOBAL_ADC_Init()**.

XMC_VADC_GLOBAL_SHS_t* const GLOBAL_ADC::global_shs_pt

This is the sample and hold structure pointer

Definition at line [152](#) of file **GLOBAL_ADC.h**.

Referenced by **GLOBAL_ADC_Init()**.

XMC_VADC_GLOBAL_SHS_CONFIG_t* const GLOBAL_ADC::glob

This is the sample and hold structure pointer

Definition at line [153](#) of file **GLOBAL_ADC.h**.

GLOBAL_ADC_GROUP_t* const GLOBAL_ADC::group_ptrs_array

This is an array of pointers to the ADC Groups

Definition at line [145](#) of file **GLOBAL_ADC.h**.

Referenced by **GLOBAL_ADC_Init()**.

GLOBAL_ADC_STATUS_t GLOBAL_ADC::init_state

This hold the State of the **GLOBAL_ADC** APP

Definition at line [155](#) of file **GLOBAL_ADC.h**.

Referenced by **GLOBAL_ADC_Init()**.

XMC_VADC_GLOBAL_t* const GLOBAL_ADC::module_ptr

This is the register structure pointer to the VADC kernel.

Definition at line [149](#) of file **GLOBAL_ADC.h**.

Referenced by **GLOBAL_ADC_Init()**.

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

- **GLOBAL_ADC.h**
-

GLOBAL_ADC

[Home](#)

[Data Structures](#)

[Data Structure Index](#)

[Data Fields](#)

[Data Fields](#)

GLOBAL_ADC_GROUP Struct Reference

[Data structures](#)

Detailed Description

Structure to hold the configuration information of a group.

Definition at line [128](#) of file [GLOBAL_ADC.h](#).

```
#include <GLOBAL\_ADC.h>
```

Data Fields

XMC_VADC_GROUP_t *const **group_handle**

const XMC_VADC_GROUP_CONFIG_t
*const **group_config_handle**

const bool **post_calibration**

GLOBAL_ADC_STATUS_t state

Field Documentation

const XMC_VADC_GROUP_CONFIG_t* const GLOBAL_ADC_GRO

This is the pointer to the Handle of the Group APP.

Definition at line [132](#) of file [GLOBAL_ADC.h](#).

Referenced by [GLOBAL_ADC_Init\(\)](#).

XMC_VADC_GROUP_t* const GLOBAL_ADC_GROUP::group_hanc

This holds the VADC group Registers.

Definition at line [130](#) of file [GLOBAL_ADC.h](#).

Referenced by [GLOBAL_ADC_Init\(\)](#).

const bool GLOBAL_ADC_GROUP::post_calibration

This enables the post calibration for a specific group

Definition at line [134](#) of file [GLOBAL_ADC.h](#).

Referenced by [GLOBAL_ADC_Init\(\)](#).

GLOBAL_ADC_STATUS_t GLOBAL_ADC_GROUP::state

This enumerates the state of the APP.

Definition at line [136](#) of file [GLOBAL_ADC.h](#).

Referenced by [GLOBAL_ADC_Init\(\)](#).

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

- [**GLOBAL_ADC.h**](#)
-



GLOBAL_ADC

Home

Data Structures

Data Structure Index

Data Fields

Data Structure Index

G

G

GLOBAL_ADC_GROUP

GLOBAL_ADC

G



GLOBAL_ADC

Home	Data Structures	Data Structure Index	Data Fields
All	Variables		

Here is a list of all documented struct and union fields with links to the struct/union documentation for each field:

- enable_startup_calibration : [GLOBAL_ADC](#)
- global_config_handle : [GLOBAL_ADC](#)
- global_shs_ptr : [GLOBAL_ADC](#)
- global_shscfg : [GLOBAL_ADC](#)
- group_config_handle : [GLOBAL_ADC_GROUP](#)
- group_handle : [GLOBAL_ADC_GROUP](#)
- group_ptrs_array : [GLOBAL_ADC](#)
- init_state : [GLOBAL_ADC](#)
- module_ptr : [GLOBAL_ADC](#)
- post_calibration : [GLOBAL_ADC_GROUP](#)
- state : [GLOBAL_ADC_GROUP](#)

GLOBAL_ADC

Home	Data Structures	Data Structure Index	Data Fields
All	Variables		

- enable_startup_calibration : GLOBAL_ADC
 - global_config_handle : GLOBAL_ADC
 - global_shs_ptr : GLOBAL_ADC
 - global_shscfg : GLOBAL_ADC
 - group_config_handle : GLOBAL_ADC_GROUP
 - group_handle : GLOBAL_ADC_GROUP
 - group_ptrs_array : GLOBAL_ADC
 - init_state : GLOBAL_ADC
 - module_ptr : GLOBAL_ADC
 - post_calibration : GLOBAL_ADC_GROUP
 - state : GLOBAL_ADC_GROUP
-

GLOBAL_ADC

Home

File List

Globals

File List

Here is a list of all documented files with brief descriptions:

 [GLOBAL_ADC.c](#)

 [GLOBAL_ADC.h](#)

GLOBAL_ADC

Home	
File List	Globals
Code	

[Functions](#)

GLOBAL_ADC.c File Reference

Detailed Description

Date

2017-06-24

NOTE: This file is generated by DAVE. Any manual modification done to this file will be lost when the code is regenerated.

Definition in file [GLOBAL_ADC.c](#).

Functions

DAVE_APP_VERSION_t [**GLOBAL_ADC_GetAppVersion**](#) (void)
Get **GLOBAL_ADC** APP version.
[More...](#)

GLOBAL_ADC_STATUS_t [**GLOBAL_ADC_Init**](#) (**GLOBAL_ADC_t**
*const handle_ptr)
Initializes the ADC global as per user
configured values. [More...](#)

Function Documentation

GLOBAL_ADC_STATUS_t GLOBAL_ADC_Init (GLOBAL_ADC_t *config)

Initializes the ADC global as per user configured values.

This function initializes all instances of the ADC Global APP and low level app.

Definition at line [116](#) of file **GLOBAL_ADC.c**.

References [GLOBAL_ADC::enable_startup_calibration](#),
[GLOBAL_ADC_SUCCESS](#), [GLOBAL_ADC_UNINITIALIZED](#),
[GLOBAL_ADC::global_config_handle](#),
[GLOBAL_ADC::global_shs_ptr](#),
[GLOBAL_ADC_GROUP::group_config_handle](#),
[GLOBAL_ADC_GROUP::group_handle](#),
[GLOBAL_ADC::group_ptrs_array](#), [GLOBAL_ADC::init_state](#),
[GLOBAL_ADC::module_ptr](#),
[GLOBAL_ADC_GROUP::post_calibration](#), and
[GLOBAL_ADC_GROUP::state](#).

Go to the source code of this file.

GLOBAL_ADC

Home

File List

Code

Globals

Data Structures

GLOBAL_ADC.h File Reference

Detailed Description

Date

2016-08-18

NOTE: This file is generated by DAVE. Any manual modification done to this file will be lost when the code is regenerated.

Definition in file [GLOBAL_ADC.h](#).

Data Structures

struct **GLOBAL_ADC_GROUP**

Structure to hold the configuration information of a group.

[More...](#)

struct **GLOBAL_ADC**

Configuration Data structure of **GLOBAL_ADC APP**. [More...](#)

TypeDefs

typedef struct **GLOBAL_ADC_GROUP** **GLOBAL_ADC_GROUP_t**

Structure to hold the configuration information of a group.

typedef struct **GLOBAL_ADC** **GLOBAL_ADC_t**

Configuration Data structure of **GLOBAL_ADC APP**.

Functions

DAVE_APP_VERSION_t [**GLOBAL_ADC_GetAppVers**](#)
(void)
Get **GLOBAL_ADC** APP
version. More...

GLOBAL_ADC_STATUS_t [**GLOBAL_ADC_Init**](#)
(**GLOBAL_ADC_t** *const
handle_ptr)
Initializes the ADC global as p
user configured values. More..

enum [**GLOBAL_ADC_STATUS**](#) {
GLOBAL_ADC_SUCCESS =
GLOBAL_ADC_FAILURE,
GLOBAL_ADC_UNINITIALIZ
}
GLOBAL_ADC state
information. More...

typedef enum **GLOBAL_ADC_STATUS** **GLOBAL_ADC_STATUS_t**
GLOBAL_ADC state
information.

Go to the source code of this file.

GLOBAL_ADC

Home					
File List	Globals				
All	Functions	Typedefs	Enumerations	Enumerator	

Here is a list of all documented functions, variables, defines, enums, and typedefs with links to the documentation:

- GLOBAL_ADC_FAILURE : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_GetAppVersion() : [GLOBAL_ADC.c](#) , [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_GROUP_t : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_Init() : [GLOBAL_ADC.c](#) , [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_STATUS : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_STATUS_t : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_SUCCESS : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_t : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_UNINITIALIZED : [GLOBAL_ADC.h](#)
-
-
-

GLOBAL_ADC

Home	
File List	Globals
All	Functions

- GLOBAL_ADC_GetAppVersion() : [GLOBAL_ADC.c](#) , [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_Init() : [GLOBAL_ADC.c](#) , [GLOBAL_ADC.h](#)
-



GLOBAL_ADC

Home					
File List	Globals				
All	Functions	Typedefs	Enumerations	Enumerator	

- GLOBAL_ADC_GROUP_t : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_STATUS_t : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_t : [GLOBAL_ADC.h](#)
-



GLOBAL_ADC

Home					
File List	Globals				
All	Functions	Typedefs	Enumerations	Enumerator	

- GLOBAL_ADC_STATUS : [GLOBAL_ADC.h](#)
-

GLOBAL_ADC

Home				
File List	Globals			
All	Functions	Typedefs	Enumerations	Enumerator

- GLOBAL_ADC_FAILURE : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_SUCCESS : [GLOBAL_ADC.h](#)
 - GLOBAL_ADC_UNINITIALIZED : [GLOBAL_ADC.h](#)
-



GLOBAL_ADC

Home	
File List	Globals
Code	

GLOBAL_ADC.h

[Go to the documentation of this file.](#)

```
1
78 #ifndef GLOBAL_ADC_H
79 #define GLOBAL_ADC_H
80
81 /*****
82 * HEADER FILES
83
*****@
84 #include <xmc_vadc.h>
85 #include "global_adc_conf.h"
86 #include <DAVE_Common.h>
87
88
*****@
89 * MACROS
90
*****@
91 #if (!(XMC_LIB_MAJOR_VERSION == 2U) && \
92 (XMC_LIB_MINOR_VERSION >= 1U) && \
93 (XMC_LIB_PATCH_VERSION >= 14U)))
94 #error "GLOBAL_ADC requires XMC Peripheral Library v2.0.0 or
higher"
95 #endif
96
*****@
97 * ENUMS
```

98

```
*****
107 typedef enum GLOBAL_ADC_STATUS
108 {
109     GLOBAL_ADC_SUCCESS = 0,
110    GLOBAL_ADC_FAILURE,
111   GLOBAL_ADC_UNINITIALIZED
112 } GLOBAL_ADC_STATUS_t;
113
117 ****
118 * DATA STRUCTURES
119 ****
127 #if XMC_VADC_GROUP_AVAILABLE == 1U
128 typedef struct GLOBAL_ADC_GROUP
129 {
130     XMC_VADC_GROUP_t *const group_handle;
132     const XMC_VADC_GROUP_CONFIG_t* const
133     group_config_handle;
134     const bool post_calibration;
136     GLOBAL_ADC_STATUS_t state;
137 } GLOBAL_ADC_GROUP_t;
138 #endif
139
142 typedef struct GLOBAL_ADC
143 {
144 #if XMC_VADC_GROUP_AVAILABLE == 1U
145     GLOBAL_ADC_GROUP_t* const
146     group_ptrs_array[XMC_VADC_MAXIMUM_NUM_GROUPS];
147     const XMC_VADC_GLOBAL_CONFIG_t* const
148     global_config_handle;
149     XMC_VADC_GLOBAL_t* const module_ptr;
151 #if(XMC_VADC_SHS_AVAILABLE == 1U)
152     XMC_VADC_GLOBAL_SHS_t* const global_shs_ptr;
153     XMC_VADC_GLOBAL_SHS_CONFIG_t* const global_shscfg;
154 #endif
155     GLOBAL_ADC_STATUS_t init_state;
```

```
157 const bool enable_startup_calibration;
158 } GLOBAL_ADC_t;
159
169 #ifdef __cplusplus
170 extern "C" {
171 #endif
172 ****
173 * API Prototypes
174 ****
175
210 DAVE_APP_VERSION_t GLOBAL_ADC_GetAppVersion(void);
211
234 GLOBAL_ADC_STATUS_t GLOBAL_ADC_Init(GLOBAL_ADC_t
*const handle_ptr);
235
236 #include "global_adc_extern.h"
237
242 #ifdef __cplusplus
243 }
244 #endif
245
246 #endif /* GLOBAL_ADC_H_ */
```

GLOBAL_ADC

Home

Data Structures

Data structures

Data Structures

struct **GLOBAL_ADC_GROUP**
Structure to hold the
configuration information of
a group. [More...](#)

typedef struct **GLOBAL_ADC_GROUP** **GLOBAL_ADC_GROUP_t**
Structure to hold the
configuration information of
a group.

typedef struct **GLOBAL_ADC** **GLOBAL_ADC_t**
Configuration Data structure
of **GLOBAL_ADC** APP.

Detailed Description



GLOBAL_ADC

Home

Enumerations

enum **GLOBAL_ADC_STATUS** {
 GLOBAL_ADC_SUCCESS =
 GLOBAL_ADC_FAILURE,
 GLOBAL_ADC_UNINITIALIZ
}

GLOBAL_ADC state
information. More...

typedef enum **GLOBAL_ADC_STATUS** **GLOBAL_ADC_STATUS_t**
GLOBAL_ADC state
information.

Detailed Description

Enumeration Type Documentation

enum GLOBAL_ADC_STATUS

GLOBAL_ADC state information.

Enumerator	
<i>GLOBAL_ADC_SUCCESS</i>	APP is in INITIALIZED state after execution of the Init function
<i>GLOBAL_ADC_FAILURE</i>	Initialization failed returns this as status
<i>GLOBAL_ADC_UNINITIALIZED</i>	This is the default state after power on reset.

Definition at line 107 of file **GLOBAL_ADC.h**.

GLOBAL_ADC

Home	
File List	Globals
Code	

GLOBAL_ADC.c

[Go to the documentation of this file.](#)

```
1
78 /*****
79 * HEADER FILES
80 ****
81
83 #include "global_adc.h"
84
85 /*****
86 * MACROS
87 ****
88
89 /*****
90 * LOCAL DATA
91 ****
92
93 /*****
94 * LOCAL ROUTINES
95 ****
96
97 ****
98 * API IMPLEMENTATION
```

```
99 ****
100
101 /*This function returns the version of the GLOBAL_ADC APP*/
102 DAVE_APP_VERSION_t GLOBAL_ADC_GetAppVersion(void)
103 {
104     DAVE_APP_VERSION_t version;
105
106     version.major = (uint8_t) GLOBAL_ADC_MAJOR_VERSION;
107     version.minor = (uint8_t) GLOBAL_ADC_MINOR_VERSION;
108     version.patch = (uint8_t) GLOBAL_ADC_PATCH_VERSION;
109
110     return version;
111 }
112 /*-----
116 GLOBAL_ADC_STATUS_t GLOBAL_ADC_Init(GLOBAL_ADC_t
*const handle_ptr)
117 {
118     XMC_ASSERT("GLOBAL_ADC_Init:Invalid handle_ptr",
119     (handle_ptr != NULL))
120     #if (XMC_VADC_GROUP_AVAILABLE == 1U)
121         uint32_t group_index;
122     #endif
123     if (GLOBAL_ADC_UNINITIALIZED == handle_ptr->init_state)
124     {
125         /* Initialize an instance of Global hardware */
126         XMC_VADC_GLOBAL_Init(handle_ptr->module_ptr, handle_ptr-
127         >global_config_handle);
128         /* Initialize all the Groups */
129         #if (XMC_VADC_GROUP_AVAILABLE == 1U)
130             for(group_index = (uint32_t)0; group_index <
131                 XMC_VADC_MAXIMUM_NUM_GROUPS; group_index++)
131             {
132                 /*Initialize Group*/
133                 XMC_VADC_GROUP_Init(handle_ptr-
```

```

>group_ptrs_array[group_index]->group_handle,
134 handle_ptr->group_ptrs_array[group_index]-
>group_config_handle);
135
136 /* Switch on the converter of the Group[group_index]*/
137 XMC_VADC_GROUP_SetPowerMode(handle_ptr-
>group_ptrs_array[group_index]->group_handle,
138 XMC_VADC_GROUP_POWERMODE_NORMAL);
139
140 /* Disable the post calibration option for the respective group*/
141 if ((bool)false == handle_ptr->group_ptrs_array[group_index]-
>post_calibration)
142 {
143 XMC_VADC_GLOBAL_DisablePostCalibration(handle_ptr-
>module_ptr,group_index);
144 }
145
146 #if(XMC_VADC_SHS_AVAILABLE == 1U)
147 XMC_VADC_GLOBAL_SHS_EnableAcceleratedMode(handle_ptr-
>global_shs_ptr, (XMC_VADC_GROUP_INDEX_t)group_index);
148 #endif
149
150 handle_ptr->group_ptrs_array[group_index]->state =
GLOBAL_ADC_SUCCESS;
151 }
152 #if(XMC_VADC_SHS_AVAILABLE == 1U)
153 XMC_VADC_GLOBAL_SHS_SetClockDivider(handle_ptr-
>global_shs_ptr, (uint8_t) GLOBAL_ADC_DIVS_VALUE);
154 #endif
155 #endif /* _XMC_VADC_GROUP_AVAILABLE_ */
156 #if(XMC_VADC_SHS_AVAILABLE == 1U)
157 XMC_VADC_GLOBAL_SHS_SetAnalogReference(handle_ptr-
>global_shs_ptr, GLOBAL_ADC_AREF_VALUE);
158 #endif
159 if((bool)true == handle_ptr->enable_startup_calibration)
160 {
161 XMC_VADC_GLOBAL_StartupCalibration(handle_ptr-

```

```
>module_ptr);  
162 }  
163 handle_ptr->init_state = GLOBAL_ADC_SUCCESS;  
164 }  
165 return (handle_ptr->init_state);  
166 }
```



GLOBAL_ADC

Home

Code >

Code Directory Reference

Files

file **GLOBAL_ADC.c** [code]

file **GLOBAL_ADC.h** [code]

GLOBAL_ADC

Home

Methods

DAVE_APP_VERSION_t **GLOBAL_ADC_GetAppVersion** (void)

Get **GLOBAL_ADC** APP version.

[More...](#)

GLOBAL_ADC_STATUS_t

GLOBAL_ADC_Init (**GLOBAL_ADC_t**

*const handle_ptr)

Initializes the ADC global as per user
configured values. [More...](#)

Detailed Description

Methods

Function Documentation

DAVE_APP_VERSION_t GLOBAL_ADC_GetAppVersion (void)

Get **GLOBAL_ADC** APP version.

Returns

DAVE_APP_VERSION_t APP version information (major, minor and patch number)

Description:

The function can be used to check application software compatibility with a specific version of the APP.

Example Usage:

```
#include <DAVE.h>
int main(void) {
    DAVE_STATUS_t init_status;
    DAVE_APP_VERSION_t version;
    // Initialize GLOBAL_ADC APP:
    // GLOBAL_ADC_Init() is called from within DAVE_Init().
    init_status = DAVE_Init();
    version = GLOBAL_ADC_GetAppVersion();
    if (version.major != 1U) {
        // Probably, not the right version.
    }
    // More code here
    while(1) {
    }
    return (0);
}
```

Inclusion of header file

Definition at line **102** of file **GLOBAL_ADC.c**.

GLOBAL_ADC_STATUS_t GLOBAL_ADC_Init (GLOBAL_ADC_t *c

Initializes the ADC global as per user configured values.

Returns

void

Description:

Initializes the VADC peripheral. Invokes various VADC LLD drivers to initialize the VADC peripheral. This would invoke The XMC_VADC_GLOBAL_Init(), XMC_VADC_GROUP_Init(). It also invokes XMC_VADC_GROUP_SetPowerMode() to power on available groups.

Example Usage:

```
#include <DAVE.h>
int main (void)
{
    DAVE_Init(); //GLOBAL_ADC_Init is called within DAVE_Init
    while(1);
    return 0;
}
```

This function initializes all instances of the ADC Global APP and low level app.

Definition at line [116](#) of file **GLOBAL_ADC.c**.

References [GLOBAL_ADC::enable_startup_calibration](#),
[GLOBAL_ADC_SUCCESS](#), [GLOBAL_ADC_UNINITIALIZED](#),
[GLOBAL_ADC::global_config_handle](#),
[GLOBAL_ADC::global_shs_ptr](#),
[GLOBAL_ADC_GROUP::group_config_handle](#),
[GLOBAL_ADC_GROUP::group_handle](#),
[GLOBAL_ADC::group_ptrs_array](#), [GLOBAL_ADC::init_state](#),
[GLOBAL_ADC::module_ptr](#),

GLOBAL_ADC_GROUP::post_calibration, and
GLOBAL_ADC_GROUP::state.