Dbg User Manual

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Data** | **Author** | **Status** | **Remarks** |
| V0.0.1 | 2022.9.19 | Mingfen XIAO | Init | Init Version |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Contents

[1.Software Components Overview 4](#_Toc114515135)

[2.Software Function Enable Switch 4](#_Toc114515136)

[2.1 Description 4](#_Toc114515137)

[2.2 Use Rules 4](#_Toc114515138)

[2.2.1 Switch Definition 4](#_Toc114515139)

[2.2.2 Switch Description 4](#_Toc114515140)

[2.2.3 Switch Usage 5](#_Toc114515141)

[2.3 Example 5](#_Toc114515142)

[3.Debug Enable Switch 5](#_Toc114515143)

[3.1 Description 5](#_Toc114515144)

[3.2 Use Rules 5](#_Toc114515145)

[3.2.1 Switch Definition 5](#_Toc114515146)

[3.2.2 Debug Variable Definition 6](#_Toc114515147)

[3.2.3 Debug Function Definition 6](#_Toc114515148)

[3.2.4 Usage 6](#_Toc114515149)

[3.3 Example 6](#_Toc114515150)

[4.Software Integration Test Enable Switch 6](#_Toc114515151)

[4.1 Description 6](#_Toc114515152)

[4.2 Use Rules 7](#_Toc114515153)

[4.2.1 Switch Definition 7](#_Toc114515154)

[4.2.2 Integration Test Variable Definition 7](#_Toc114515155)

[4.2.3 Integration Test Function Definition 7](#_Toc114515156)

[4.2.4 Usage 7](#_Toc114515157)

[4.3 Example 8](#_Toc114515158)

[5.Software Unit Test Enable Switch 8](#_Toc114515159)

[5.1 Description 8](#_Toc114515160)

[5.2 Use Rules 8](#_Toc114515161)

[5.2.1 Switch Definition 8](#_Toc114515162)

[5.2.2 Unit Test Variable Definition 8](#_Toc114515163)

[5.2.3 Unit Test Function Definition 8](#_Toc114515164)

[5.2.4 Usage 9](#_Toc114515165)

[5.3 Example 9](#_Toc114515166)

[6.MISC Enable Switch 9](#_Toc114515167)

本文档旨在规范代码开发过程中的兼容性，包含软件模块子功能的开关配置，调试变量和函数的开关配置，集成/单元测试代码和量产代码的兼容等，在软件开发过程中，按照此规范进行执行。

对于软件开发者，在软件开发中，需要按照此规范进行代码的设计和开发，以提到较高的兼容性；

对于软件集成者，在软件集成中，需要根据实际项目中需要集成的内容，对相关的开关进行配置；

# 1.Software Components Overview

Dbg，Debugging，主要用于管理调试信息以及非标定可配置的软件子功能的使能配置。

此模块属于CSC\_Platform\_Common Composition中，相关的配置文件位于*01\_AsrConfig\source\Swc\_Appl\Platform\_Common\02\_Config\Dbg\_Cfg.h*

下述章节中所描述的开关，全部汇总中Dbg\_Cfg.h文件中。

# 2.Software Function Enable Switch

## 2.1 Description

Software Function Enable Switch用于对软件软件中各个子功能的开关配置。比如说，软件模块A中实现了三个独立的子功能分别为A1，A2，A3，可以定义三种宏，以实现三个子功能的开关控制。软件集成工程师可以根据软件发布对功能的要求，在软件集成时可以根据配置软件模块子功能的宏开关来实现对功能的使能。

## 2.2 Use Rules

### 2.2.1 Switch Definition

使用如下的规则对功能开关进行定义。

* Enable : *#define DBG\_<SWCs>\_<Function>\_ENABLE STD\_ON*
* Disable : *#define DBG\_<SWCs>\_<Function>\_ENABLE STD\_OFF*

### 2.2.2 Switch Description

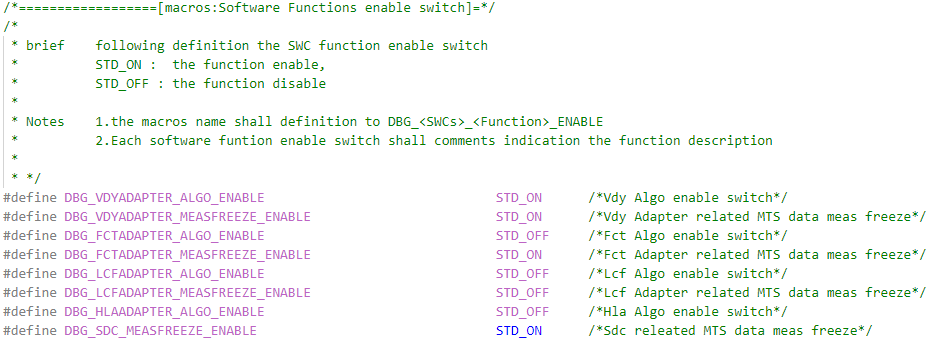
在定义功能使能开关时，必须增加注释对功能进行描述，用于指导软件工程师和集成工程师对功能的开关配置。

### 2.2.3 Switch Usage

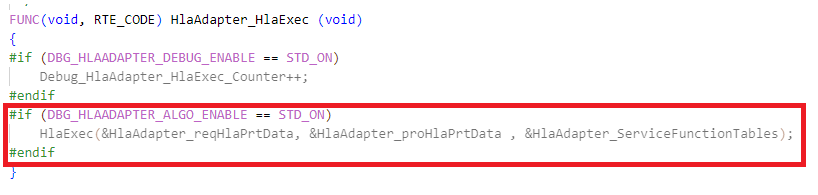
在代码实现时，需要通过条件编译的方式来实现对功能的开关切换。

## 2.3 Example

Switch Definition:



Switch Usage:



# 3.Debug Enable Switch

## 3.1 Description

Debug Enable Switch 用于软件模块对调试变量和函数进行条件编译的控制；

DBG\_MCU\_SW\_DEBUG\_ENABLE为Debug Enable Switch的总开关，只有此开关为STD\_ON时，软件模块开关才有效。

## 3.2 Use Rules

### 3.2.1 Switch Definition

使用如下规则对于Debug Enable Switch进行定义：

* Enable : *#define DBG\_<SWCs>\_DEBUG\_ENABLE STD\_ON*
* Disable : *#define DBG\_<SWCs>\_DEBUG\_ENABLE STD\_OFF*

### 3.2.2 Debug Variable Definition

专门用于Debug的变量定义，全部使用”Debug\_<SWCs>\_Var\_”作为前缀，使用如下规则：

*<data type> Debug\_<SWCs>\_Var\_<Variable Description>*

对于专门用于Debug的变量定义，需要使用条件编译的方式，以减少正式释放软件的RAM消耗，使用如下规则：

*#if (DBG\_<SWCs>\_DEBUG\_ENABLE == STD\_ON)*

*<data type> Debug\_<SWCs>\_Var\_<Variable Description>*

*#endif*

### 3.2.3 Debug Function Definition

专门用于Debug的函数定义，全部使用”Debug\_<SWCs>\_Func\_”作为前缀，使用如下规则：

*<data type> Debug\_<SWCs>\_Func\_<Variable Description>*

对于专门用于Debug的函数定义，需要使用条件编译的方式，以减少正式释放软件的ROM消耗，使用如下规则：

*#if (DBG\_<SWCs>\_DEBUG\_ENABLE == STD\_ON)*

*<data type> Debug\_<SWCs>\_Func\_<Variable Description>*

*#endif*

### 3.2.4 Usage

使用条件编译的方式来实现对Debug变量的使用和Debug函数的调用。

## 3.3 Example

# 4.Software Integration Test Enable Switch

## 4.1 Description

Software Integration Test Enable Switch 用于软件模块用于进行集成测试的变量和代码进行控制；

DBG\_MCU\_SW\_INTEGRATION\_TEST\_ENABLE为Software Integration Test Enable Switch的总开关，只有此开关为STD\_ON时，软件模块开关才有效。

## 4.2 Use Rules

### 4.2.1 Switch Definition

使用如下规则对于Integration Test Enable Switch进行定义：

* Enable : *#define DBG\_<SWCs>\_INTEGRATION\_TEST\_ENABLE STD\_ON*
* Disable : *#define DBG\_<SWCs>\_ INTEGRATION\_TEST\_ENABLE STD\_OFF*

### 4.2.2 Integration Test Variable Definition

专门用于集成测试的变量定义，全部使用”IT\_<SWCs>\_Var\_”作为前缀，使用如下规则：

*<data type> IT\_<SWCs>\_Var\_<Variable Description>*

对于专门用于集成测试的变量定义，需要使用条件编译的方式，以减少正式释放软件的RAM消耗，使用如下规则：

*#if (DBG\_<SWCs>\_INTEGRATION\_TEST\_ENABLE == STD\_ON)*

*<data type> IT\_<SWCs>\_Var\_<Variable Description>*

*#endif*

### 4.2.3 Integration Test Function Definition

专门用于集成测试的函数定义，全部使用”IT\_<SWCs>\_Func\_”作为前缀，使用如下规则：

*<data type> IT\_<SWCs>\_Func\_<Variable Description>*

对于专门用于集成测试的函数定义，需要使用条件编译的方式，以减少正式释放软件的ROM消耗，使用如下规则：

*#if (DBG\_<SWCs>\_INTEGRATION\_TEST\_ENABLE == STD\_ON)*

*<data type> IT\_<SWCs>\_Func\_<Variable Description>*

*#endif*

### 4.2.4 Usage

使用条件编译的方式来实现对集成测试相关的变量的使用和集成测试相关函数的调用。

## 4.3 Example

# 5.Software Unit Test Enable Switch

## 5.1 Description

Software Unit Test Enable Switch 用于软件模块用于进行单元测试的变量和代码进行控制；

DBG\_MCU\_SW\_UNIT\_TEST\_ENABLE为Software Unit Test Enable Switch的总开关，只有此开关为STD\_ON时，软件模块开关才有效。

## 5.2 Use Rules

### 5.2.1 Switch Definition

使用如下规则对于Unit Test Enable Switch进行定义：

* Enable : *#define DBG\_<SWCs>\_UNIT\_TEST\_ENABLE STD\_ON*
* Disable : *#define DBG\_<SWCs>\_ UNIT\_TEST\_ENABLE STD\_OFF*

### 5.2.2 Unit Test Variable Definition

专门用于单元测试的变量定义，全部使用”UT\_<SWCs>\_Var\_”作为前缀，使用如下规则：

*<data type> UT\_<SWCs>\_Var\_<Variable Description>*

对于专门用于单元测试的变量定义，需要使用条件编译的方式，以减少正式释放软件的RAM消耗，使用如下规则：

*#if (DBG\_<SWCs>\_ UNIT\_TEST\_ENABLE == STD\_ON)*

*<data type> UT\_<SWCs>\_Var\_<Variable Description>*

*#endif*

### 5.2.3 Unit Test Function Definition

专门用于单元测试的函数定义，全部使用”IT\_<SWCs>\_Func\_”作为前缀，使用如下规则：

*<data type> UT\_<SWCs>\_Func\_<Variable Description>*

对于专门用于Debug的函数定义，需要使用条件编译的方式，以减少正式释放软件的ROM消耗，使用如下规则：

*#if (DBG\_<SWCs>\_UNIT\_TEST\_ENABLE == STD\_ON)*

*<data type> UT\_<SWCs>\_Func\_<Variable Description>*

*#endif*

### 5.2.4 Usage

使用条件编译的方式来实现对单元测试相关的变量的使用和单元测试相关函数的调用。

## 5.3 Example

# 6.MISC Enable Switch

可以根据项目实际情况，对于上述描述的四种情况外的开关进行定义。比如由于DV使用监控的需要，增加部分监控状态信号通过CAN输出，可以在类型中定义如下开关：

*#define DBG\_MCU\_SW\_DV\_TEST\_ENABLE                               STD\_OFF*