|  |
| --- |
|  |
|  |
| DTC故障策略模块  设计与应用文档 |
|  |
|  |

|  |
| --- |
| 文件状态：  [ ] 草稿  [√] 正式发布  [ ] 正在修改 |
| 文件起草分工：   1. 郭春吉 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **编制：** | **郭春吉** | **签名：** | **郭春吉** | **日期：** | **2013-7-15** |
| **审核：** |  | **签名：** |  | **日期：** |  |
| **批准：** |  | **签名：** |  | **日期：** |  |

**更改历史**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 更改描述 | 更改日期 | 更改人 |
| 1.0 | 初始版本 | 2013-7-15 | 郭春吉 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**目 录**

[1 概述 1](#_Toc361733726)

[1.1 目的 1](#_Toc361733727)

[1.2 适用范围 1](#_Toc361733728)

[1.3 读者对象 1](#_Toc361733729)

[1.4 术语和缩写 1](#_Toc361733730)

[2 概述 1](#_Toc361733731)

[3 功能描述 1](#_Toc361733732)

[3.1 状态位描述 1](#_Toc361733733)

[3.2 扩展数据描述 2](#_Toc361733734)

[3.2.1 默认扩展数据 2](#_Toc361733735)

[3.2.2 用户指定扩展数据 2](#_Toc361733736)

[3.3 冻结帧描述 2](#_Toc361733737)

[4 文件结构 3](#_Toc361733738)

[5 配置接口 3](#_Toc361733739)

[5.1 Dtc\_Cfg.c文件配置项 3](#_Toc361733740)

[5.1.1 DTC故障码初始值配置表 3](#_Toc361733741)

[5.1.2 DTC故障码属性表配置 3](#_Toc361733742)

[5.1.3 冻结帧配置表 5](#_Toc361733743)

[5.1.4 动态DID信息配置表 6](#_Toc361733744)

[5.2 Dtc\_Cfg.h宏配置文件 6](#_Toc361733745)

[6 集成说明 7](#_Toc361733746)

[6.1 头文件包含 7](#_Toc361733747)

[6.2 函数集成 7](#_Toc361733748)

[7 故障变量接口 8](#_Toc361733749)

[8 DtcStatusManage.h文件变量与宏定义 8](#_Toc361733750)

[9 函数描述 11](#_Toc361733751)

[9.1 故障预初始化函数 11](#_Toc361733752)

[9.2 故障初始化函数 11](#_Toc361733753)

[9.3 DTC模块主函数 12](#_Toc361733754)

[9.4 故障信息清除勾子函数 12](#_Toc361733755)

[9.5 DTC状态位上电初始化逻辑 13](#_Toc361733756)

[9.6 DTC状态位更新函数 13](#_Toc361733757)

[9.7 DTC数据存储函数 13](#_Toc361733758)

[9.8 当前故障防抖处理逻辑 14](#_Toc361733759)

[9.9 历史故障计数逻辑 14](#_Toc361733760)

[9.10 故障状态位更新逻辑 15](#_Toc361733761)

[9.11 初始化故障检测函数 15](#_Toc361733762)

[9.12 扩展数据更新逻辑 16](#_Toc361733763)

[9.13 冻结帧数据更新逻辑 16](#_Toc361733764)

[9.14 数据拷贝函数 16](#_Toc361733765)

1. 概述
   1. 目的

本文档的目的是对DTC策略模块的设计与集成配置进行描述。

* 1. 适用范围

本文档适用于维护DTC策略模块程序的人员和使用该模块的人员进行参考。

* 1. 读者对象

本文档预期读者对象为：

1. 顾客代表
2. 项目成员
3. 高层经理
   1. 术语和缩写
4. DTC: Diagnostic Trouble Code
5. FreezeFrame: 冻结帧
6. ExtendData: 扩展数据
7. 概述

本文描述了DTCcode及附属的8个故障状态位、故障扩展数据记录、故障相关的冻结帧数据的存储更新逻辑。描述了模块的配置接口和集成说明。

1. 功能描述

DTC故障策略管理模块主要功能是管理控制器的DTC Code，运行过程中根据应用层提供的实时监测结果更新对应故障的8个故障状态位信息、计算默认的扩展数据记录和用户指定的扩展数据信息以及实时捕捉部分故障的冻结帧数据信息，并对这些故障信息进行存储管理。

* 1. 状态位描述

|  |  |  |
| --- | --- | --- |
| **Status Bits** | **ISO Name** | **Description (Display)** |
| Bit0 | testFailed | 0 - DTC is no longer failed at the time of the request. 1 - DTC failed at the time of the request. |
| Bit1 | testFailedThisMonitoringCycle | 0 - DTC never failed on the current operation cycle. 1 - DTC failed on the current operation cycle. |
| Bit2 | pendingDTC | 0 - DTC was not failed on the current or previous operation cycle. 1 - DTC failed on the current or previous operation cycle. |
| Bit3 | confirmedDTC | 0 - DTC is not confirmed at the time of the request. 1 - DTC is confirmed at the time of the request. |
| Bit4 | testNotCompletedSinceLastClear | 0 - DTC test has been completed since the last code clear. 1 - DTC test has not been completed since the last code clear. |
| Bit5 | testFailedSinceLastClear | 0 - DTC test never failed since last code clear. 1 - DTC test failed at least once since last code clear. |
| Bit6 | testNotCompletedThisMonitoringCycle | 0 - DTC test completed this operation cycle. 1 - DTC test has not completed this operation cycle. |
| Bit7 | warningIndicatorRequested | 0 - Server is not requesting warningIndicator to be active. 1 - Server is requesting warningIndicator to be active |

* 1. 扩展数据描述
     1. 默认扩展数据
* 故障出现的点火周期数量；
  + 面向所有故障；
  + 出现故障时数值累加，且每个点火周期内只累加一次；
  + 在成功收到14请求时清除数据信息；
  + 故障自恢复完成时，不能清除该计数器信息；
  + 通过19-06（服务-子功能）读取并解析；
  + 点火周期结束时存入EEPROM；
* 历史故障老化计数器值；
  + 面向所有故障；
  + 历史故障bit3置位后，立即启动计数，一个点火周期内如果未出现故障，则会累加1；
  + 在成功收到14请求时清除计数器信息；
  + 在老化计数器累计到恢复阈值时，自动清除计数器信息；
  + 在某一点火周期内再次出现故障时，清除计数器信息；
  + 通过19-06（服务-子功能）读取并解析；
  + 点火周期结束时存入EEPROM；
    1. 用户指定扩展数据
* 默认无，根据客户需要进行定义；
  1. 冻结帧描述

冻结帧用于记录故障发生瞬间，控制器实时捕捉到的整车环境信息。这些信息可以帮助还原汽车故障发生时的真实环境，从而对于分析和解决故障提供数据支持。可能需要记录的参量如下：系统电压、车速、电机转速、电机温度、电流等。

* 特定故障需要冻结帧支持；
* 故障发生瞬间，应用层负责捕捉相关信息；
* DTC策略模块获取环境信息后将之冻结；
* 在冻结帧被强制清除之前，故障再次发生时不应导致冻结帧更新；
* 故障自恢复完成时，不允许清除对应冻结帧信息；
* 冻结帧信息的清除只有14服务手动清除；
* 所有冻结帧采用统一编码方式；
* 每个冻结帧包含1个或多个动态DID，每个DID对应一个环境参量；
* 通过19-04（服务-子功能）读取并解析；
* 点火周期结束时存入EEPROM；

1. 文件结构

DtcStatusManage.c

--DTC故障策略模块逻辑实现

DtcStatusManage.h

--DTC故障策略模块逻辑实现头文件

Dtc\_Cfg.c

--DTC故障策略模块配置文件

Dtc\_Cfg.h

--DTC故障策略模块配置文件头文件

Dtc\_Fault\_Def.c

--DTC故障接口变量定义文件，应用层提供

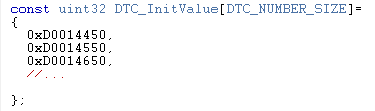
Dtc\_Fault\_Def.h

--DTC故障接口变量定义数据结构头文件

Dtc\_Types.h

--DTC数据类型与通用宏定义文件

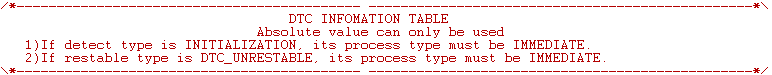
1. 配置接口
   1. Dtc\_Cfg.c文件配置项
      1. DTC故障码初始值配置表

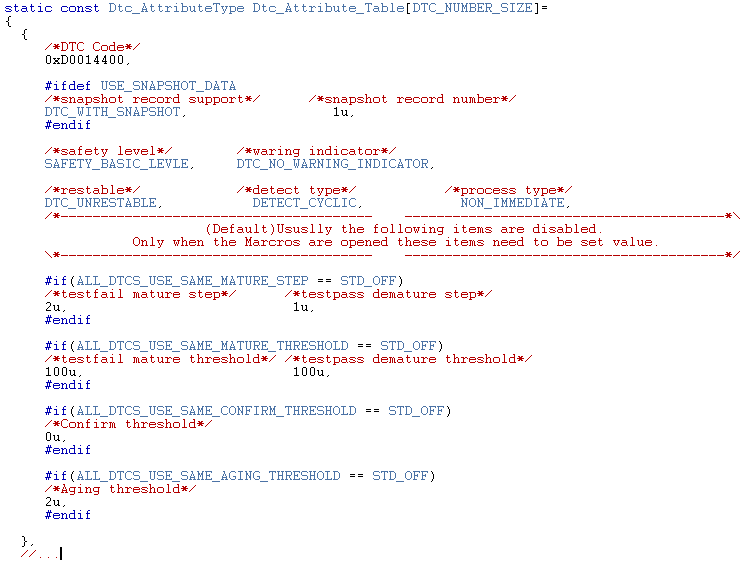


将控制器所有的故障码配置在此表中，其索引顺序需要与故障变量接口定义中的故障标志变量结构成员顺序严格一致，两者一一对应。

故障码采用4bytes故障码—High、Middle、Low bytes 及 状态位字节。

* + 1. DTC故障码属性表配置





* DTC码初始值；
* 是否支持冻结帧

DTC\_WITH\_SNAPSHOT：当前故障支持冻结帧；

DTC\_NO\_SNAPSHOT： 当前故障不支持冻结帧；

* 冻结帧编码

根据诊断规范中定义的统一编码配置对应的冻结帧编号；

0x01~0xFE 用于表示冻结帧编码；

* 安全等级

根据诊断规范中定义的故障安全等级进行配置；

SAFETY\_S\_LEVEL /\*UnRestable\*/

SAFETY\_A\_LEVEL /\*UnRestable\*/

SAFETY\_B\_LEVEL

SAFETY\_C\_LEVEL

SAFETY\_D\_LEVEL

* 故障指示灯

DTC\_NO\_WARNING\_INDICATOR 该故障不支持故障灯指示

DTC\_WITH\_WARNING\_INDICATOR 该故障支持故障灯指示

* 故障可重置性

DTC\_UNRESTABLE 故障完全不可重置

DTC\_PARTIALRESTABLE 故障部分可重置

DTC\_RESTABLE 故障可重置

* 故障检测机制

DETECT\_CYCLIC 始终周期监测

DETECT\_ON\_DEMAND 基于命令/条件的周期监测

DETECT\_IN\_INITIALIZATION 初始化过程监测

* 故障处理模式

IMMEDIATE 立即模式

NON\_IMMEDIATE 防抖模式

/\*If use DTC\_UNRESTABLE, IMMEDIATE mode will be used.

If use DTC\_ DETECT\_IN\_INITIALIZATION, IMMEDIATE mode will be used.\*/

* 故障置位防抖计数器步长

8位无符号整型数，default value: 2u;

* 故障复位防抖计数器步长

8位无符号整型数, default value: 1u;

* 故障置位防抖计数器阈值

16位无符号整型数

* 故障复位防抖计数器阈值

16位无符号整型数

* 历史故障置位计数器阈值

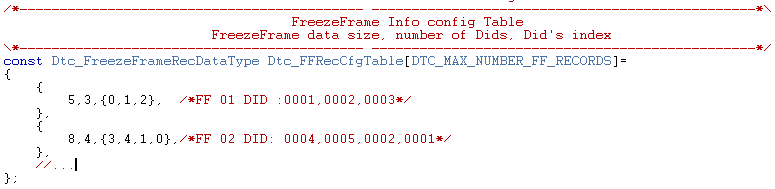
16位无符号整型数

配置为0u表示，历史故障位与当前故障位同时置位；

* 历史故障老化计数器阈值

16位无符号整型数

* + 1. 冻结帧配置表



* 冻结帧元素信息

冻结帧数据大小；

所包含Did的数量；

Did编码；

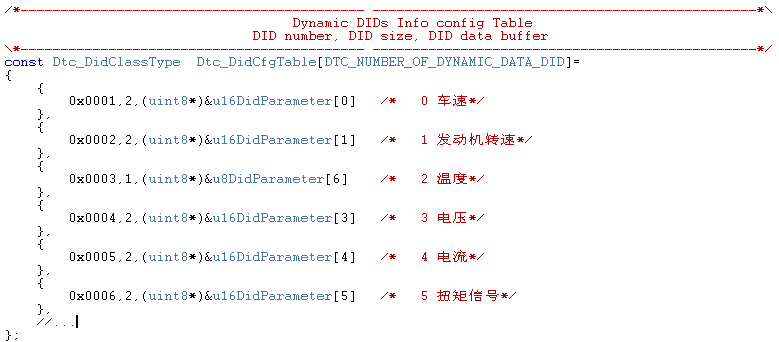
* 冻结帧编码信息

数组元素索引号即为冻结帧编号；

* 冻结帧组合原则

按每个故障码的实际需要，确保每个需要存储冻结帧的DTC码有且仅有一个与己对应的冻结帧；

* + 1. 动态DID信息配置表



* DID元素信息

动态DID编码（参考企业诊断规范或ISO14229）；

DID数据大小；

数据存储地址；

* 1. Dtc\_Cfg.h宏配置文件

|  |  |
| --- | --- |
| 宏定义 | 描述 |
| ROLLING\_MODE | 如果定义该宏，表示DTC策略模块会作为独立的任务被循环调度；默认采用该模式； |
| REALTIME\_CALL\_MODE | 如果定义该宏，表示DTC策略模块会以接口函数的方式为各个故障检测模块调用； |
| DTC\_TASK\_TIME | 轮询模式中，DTC策略任务的调度周期 |
| DTC\_DELAY\_ON\_BOOT | 上电启动后，DTC策略任务延迟调度时间； |
| DTC\_STATUS\_MASK | 状态位掩码信息 |
| USE\_SNAPSHOT\_DATA | 模块是否需要支持冻结帧功能 |
| DTC\_NUMBER\_SIZE | 故障码数量 |
| DTC\_CODE\_STORE\_SIZE | 故障码存储大小 |
| DEM\_DEFAULT\_EXTDATA\_RECORD\_SIZE | 默认扩展数据大小 |
| DEM\_USER\_DEFINE\_EXTDATA\_RECORD\_SIZE | 用户定义扩展数据大小 |
| DTC\_FREEZEFRAME\_AMOUNT | 冻结帧的数量 |
| DTC\_SNAPSHOT\_TYPE\_SIZE | 冻结帧数据结构大小 |
| DTC\_NUMBER\_OF\_DYNAMIC\_DATA\_DID | 动态数据DID的数量 |
| DTC\_MAX\_NUMBER\_FF\_RECORDS | 冻结帧的数量 |
| DTC\_MAX\_NUMBER\_DID\_IN\_FF\_RECORDS | 所有冻结帧中包含最多DID的数量 |
| ALL\_DTCS\_USE\_SAME\_MATURE\_STEP | 如果打开，表明所有故障使用相同的防抖步长 |
| ALL\_DTCS\_USE\_SAME\_MATURE\_THRESHOLD | 如果打开，表明所有故障使用相同的防抖阈值 |
| ALL\_DTCS\_USE\_SAME\_CONFIRM\_THRESHOLD | 如果打开，表明所有故障使用相同的故障确认阈值 |
| ALL\_DTCS\_USE\_SAME\_AGING\_THRESHOLD | 如果打开，表明所有故障使用相同的老化计数器阈值 |
| DTC\_FAIL\_MATURE\_STEP | 置位防抖步长 |
| DTC\_PASS\_DEMATURE\_STEP | 复位防抖步长 |
| DTC\_FAIL\_MATURE\_THRESHOLD | 置位防抖阈值 |
| DTC\_PASS\_DEMATURE\_THRESHOLD | 复位防抖阈值 |
| DTC\_FAIL\_CONFIRM\_THRESHOLD | 历史故障置位阈值 |
| DTC\_AGING\_THRESHOLD | 历史故障自恢复阈值 |

* 数据类型

Dtc\_DidClassType， 动态DID数据类型

Dtc\_FreezeFrameRecDataType 冻结帧数据类型

Dtc\_AttributeType DTC属性类型；

1. 集成说明
   1. 头文件包含

* 在main.h中或自定义的主调度文件头文件中包含

"DtcStatusManage.h"；

* 在诊断应用头文件App\_Dcm.h中包含

"DtcStatusManage.h"；

* 在各个故障监测模块头文件中包含

"Dtc\_Fault\_Def.h" 和"DtcStatusManage.h"；TBD

（包含后者的目的在于获取故障等级信息）

* 在诊断策略模块头文件中包含

"Dtc\_Fault\_Def.h" 和"DtcStatusManage.h"；TBD

* 1. 函数集成
* Dtc\_PreInit()

该函数置于系统初始化过程中，并且必须放置在所有“初始化故障检测”（即初始化过程中进行故障检测的模块）模块之前。

* Dtc\_Init()

该函数置于系统初始化过程中，并且必须放置在所有“初始化故障检测”（即初始化过程中进行故障检测的模块）模块之后。

* Dtc\_MainTask()

该函数作为任务需要周期调度，如可置于5ms或10ms定时器任务中调度，调度周期务必与Dtc\_Cfg.h中配置的DTC\_TASK\_TIME一致。

* Dtc\_Status\_14Service\_Hook(uint32 groupofDTC)

该函数置于App\_Dcm.c中14 Service的功能函数

App\_ClearDiagnosticInformation(Dcm\_MsgContextType\* pMsgContext)，由其调用。

* Dtc\_Save(void)

该函数在点火切换到OFF后或者检测到异常掉电后调用。

1. 故障变量接口

TBD

1. DtcStatusManage.h文件变量与宏定义

|  |  |
| --- | --- |
| 宏定义 | 描述 |
| 通用宏定义 | |
| EXD\_DATA\_INITIAL | 扩展数据初始值，0x00000000u |
| FREEZE\_DATA\_INITIAL | 冻结帧数据初始值，0xFFFFFFFFu |
| 故障监控逻辑宏定义 | |
| DTC\_TEST\_FAILED | 当前测试有故障 |
| DTC\_TEST\_PASSED | 当前测试无故障 |
| DTC\_CONFIRMED | 确认为历史故障 |
| DTC\_NOT\_CONFIRMED | 历史故障未确认 |
| DTC\_TEST\_FAILED\_COUNTER\_CLEAR | 当前故障防抖计数器清零 |
| DTC\_TEST\_FAILED\_COUNTER\_START | 当前故障计数器启动 |
| DTC\_CONFIRM\_COUNTER\_CLEAR | 历史故障计数器清零 |
| DTC\_CONFIRM\_COUNTER\_START | 历史故障计数器启动 |
| DTC\_AGING\_COUNTER\_CLEAR | 故障老化计数器清零 |
| DTC\_AGING\_COUNTER\_START | 故障老化计数器启动 |
| DTC\_HAS\_CLEAR | 故障已被14清除 |
| DTC\_NOT\_CLEAR | 故障未被14清除 |
| MONITOR\_CYCLE\_NOTCOMPLETE | 当前故障监控周期未结束 |
| MONITOR\_CYCLE\_END\_IN\_PASS | 当前故障监控结果通过 |
| MONITOR\_CYCLE\_END\_IN\_FAIL | 当前故障监控结果失败 |
| DTC\_TRUE | -- |
| DTC\_FALSE | -- |
| IN\_FAULT | 故障标志位使用，置位 |
| NO\_FAULT | 故障标志位使用，清零 |
| DTC\_ENABLE | DTC检测使能 |
| DTC\_DISABLE | DTC检测禁能 |
| DTC 状态位逻辑宏定义 | |
| DTC\_BIT0 | Bit0掩码 |
| DTC\_BIT2 | … |
| DTC\_BIT3 | … |
| DTC\_BIT4 | … |
| DTC\_BIT5 | … |
| DTC\_BIT6 | … |
| DTC\_BIT7 | … |
| DTC\_NOT\_BIT0 | Bit0掩码反码 |
| DTC\_NOT\_BIT1 | … |
| DTC\_NOT\_BIT2 | … |
| DTC\_NOT\_BIT3 | … |
| DTC\_NOT\_BIT4 | … |
| DTC\_NOT\_BIT5 | … |
| DTC\_NOT\_BIT6 | … |
| DTC\_NOT\_BIT7 | … |
| DTC\_BIT0\_SUPPORT | judge if bit0 is supported |
| DTC\_BIT1\_SUPPORT | …. |
| DTC\_BIT2\_SUPPORT | …. |
| DTC\_BIT3\_SUPPORT | ….. |
| DTC\_BIT4\_SUPPORT | …. |
| DTC\_BIT5\_SUPPORT | …. |
| DTC\_BIT6\_SUPPORT | …. |
| DTC\_BIT7\_SUPPORT | …. |
| BIT0\_SET\_OPERRAND | 状态位置位操作数 |
| BIT1\_SET\_OPERRAND |
| BIT2\_SET\_OPERRAND |
| BIT3\_SET\_OPERRAND |
| BIT4\_SET\_OPERRAND |
| BIT5\_SET\_OPERRAND |
| BIT6\_SET\_OPERRAND |
| BIT7\_SET\_OPERRAND |
| BIT0\_RESET\_OPERRAND | 状态位复位操作数 |
| BIT1\_RESET\_OPERRAND |
| BIT2\_RESET\_OPERRAND |
| BIT3\_RESET\_OPERRAND |
| BIT4\_RESET\_OPERRAND |
| BIT5\_RESET\_OPERRAND |
| BIT6\_RESET\_OPERRAND |
| BIT7\_RESET\_OPERRAND |
| BIT0\_SET(dtc) | 状态位置位 |
| BIT1\_SET(dtc) |
| BIT2\_SET(dtc) |
| BIT3\_SET(dtc) |
| BIT4\_SET(dtc) |
| BIT5\_SET(dtc) |
| BIT6\_SET(dtc) |
| BIT7\_SET(dtc) |
| BIT0\_RESET(dtc) | 状态位复位 |
| BIT1\_RESET(dtc) |
| BIT2\_RESET(dtc) |
| BIT3\_RESET(dtc) |
| BIT4\_RESET(dtc) |
| BIT5\_RESET(dtc) |
| BIT6\_RESET(dtc) |
| BIT7\_RESET(dtc) |
| DTC数据结构宏定义 | |
| DTC\_CODE(x) | 全局DTC code数组 |
| DTC\_STATUS\_BYTE(x) | 全局DTC code状态字节数组 |
| DTC\_STATUS\_BIT0(x) | 全局DTC code 状态位数组 |
| DTC\_STATUS\_BIT1(x) |
| DTC\_STATUS\_BIT2(x) |
| DTC\_STATUS\_BIT3(x) |
| DTC\_STATUS\_BIT4(x) |
| DTC\_STATUS\_BIT5(x) |
| DTC\_STATUS\_BIT6(x) |
| DTC\_STATUS\_BIT7(x) |
| DTC\_EXT\_DATA(x) | 扩展数据全局数组 |
| DTC\_EXT\_DATA\_BYTE(x) |
| DTC\_EXT\_AGING\_COUNTER\_DATA(x) |
| DTC\_EXT\_NUM\_FAILED\_CYCLE\_DATA(x) |
| DTC\_SNAPSHOT\_DATA(x) | 冻结帧全局数组 |
| DTC\_SNAPSHOT\_DATA\_16BIT(x) |
| DTC\_SNAPSHOT\_DATA\_BYTE(x) |

1. 函数描述
   1. 故障预初始化函数

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | void Dtc\_PreInit() | |
| 功能描述 | 所有故障接口变量标志位初始化 | |
| 逻辑描述 | 清零逻辑 | |
| 注意事项 | 该函数需在所有**初始化故障检测**模块之前调用； | |
| 输入参数 | 定义 | 取值范围 |
| 无 | —— | —— |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. 故障初始化函数

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | void Dtc\_Init() | |
| 功能描述 | 读取EEPROM故障信息或将默认故障信息并存入故障信息的RAM buffer中；  调用Dtc\_Status\_PowerOn\_Init()完成部分状态位信息更新；  调用InitializationFault\_DTCStatusUpdate()完成初始化故障的逻辑判断与更新； | |
| 注意事项 | 该函数需在所有**初始化故障检测**模块之后调用； | |
| 逻辑描述 | * 1. 读取EEPROM中存储的故障数据；   2. 如果数据为空，则使用默认初始值初始化故障数据的RAM空间;   3. 如果数据不为空，则根据各个数据结构的特点分别完成DTC码RAM空间、扩展数据RAM空间以及冻结帧数据RAM空间的初始化；   4. 确定上个周期是否发生过故障；   5. 调用状态位初始化逻辑函数；   6. 调用初始化过程故障检测函数； | |
| 输入参数 | 定义 | 取值范围 |
| 无 | —— | —— |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. DTC模块主函数

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | void Dtc\_MainTask(void) | |
| 功能描述 | 判断延时调度信息，调用DTC状态更新函数进行故障状态更新 | |
| 逻辑描述 | 略 | |
| 注意事项 | only be appropriate for ROLLING MODE | |
| 输入参数 | 定义 | 取值范围 |
| 无 | —— | —— |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. 故障信息清除勾子函数

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | void Dtc\_Status\_14Service\_Hook(uint32 groupofDTC) | |
| 功能描述 | 执行14服务对故障信息的清除操作 | |
| 逻辑描述 | 略 | |
| 注意事项 | 无 | |
| 输入参数 | 定义 | 取值范围 |
| groupofDTC | uint32 groupofDTC | 0x00000000u~  0xFFFFFF00u |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. DTC状态位上电初始化逻辑

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | void Dtc\_Status\_PowerOn\_Init() | |
| 功能描述 | 状态位初始化 | |
| 逻辑描述 | 1.各状态位初始化逻辑；  2.历史故障更新、老化判断； | |
| 注意事项 | 无 | |
| 输入参数 | 定义 | 取值范围 |
| 无 | —— | —— |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. DTC状态位更新函数

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | void Dtc\_StatusUpdate() | |
| 功能描述 | 部分状态位的初始化，历史故障位的逻辑操作 | |
| 逻辑描述 | 调用主要的三个状态位判断逻辑完成状态位更新； | |
| 注意事项 | 为DTC主函数调用；执行非初始化故障的检测； | |
| 输入参数 | 定义 | 取值范围 |
| 无 | —— | —— |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. DTC数据存储函数

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | uint8 Dtc\_Save(void) | |
| 功能描述 | 将故障信息打包存储到EEPROM | |
| 逻辑描述 | 1. 获得当前的故障码状态； 2. 获得当前的扩展数据； 3. 获得当前的冻结帧数据； 4. 解析不同的数据结构buffer打包至EEPROM存储buffer; 5. 调用EEPROM写入函数完成数据存储； | |
| 注意事项 | 点火周期结束或掉电时调用该函数 | |
| 输入参数 | 定义 | 取值范围 |
| 无 | —— | —— |
| 返回值/类型 | 定义 | 取值范围 |
| Uint8 | Flag | 0u，1u |

* 1. 当前故障防抖处理逻辑

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | static uint8 Dtc\_MatureCounterState(uint8 aIndex, uint8 testResult) | |
| 功能描述 | 进行监控周期monitor cycle的逻辑处理，给出监控结果； | |
| 逻辑描述 | 根据故障测定结果   1. 如果发生故障，分别处理故障类型为不可重置，故障为可重置且为immediate模式，故障为可重置且为非immediate模式的逻辑； 2. 如果没有故障，递减相关计数器，逐步完成一个monitor cycle. | |
| 注意事项 | 无 | |
| 输入参数 | 定义 | 取值范围 |
| uint8 | aIndex | 0~255 |
| uint8 | testResult | 0~1  0:No fault  1:In fault |
| 返回值/类型 | 定义 | 取值范围 |
| Uint8 | Dtc\_MonitorState | 0:MONITOR\_CYCLE\_NOTCOMPLETE  1:MONITOR\_CYCLE\_END\_IN\_PASS 2:MONITOR\_CYCLE\_END\_IN\_FAIL |

* 1. 历史故障计数逻辑

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | static uint8 Dtc\_ConfirmedCounterState(uint8 aIndex, uint8 monitorResult) | |
| 功能描述 | 进行历史故障逻辑处理，返回检测结果； | |
| 逻辑描述 | 根据监控周期的监控结果，更新并判断历史故障计数器，并据此更新历史故障位； | |
| 注意事项 | 无 | |
| 输入参数 | 定义 | 取值范围 |
| uint8 | aIndex | 0~255 |
| uint8 | monitorResult | 0~2  0:MONITOR\_CYCLE\_NOTCOMPLETE  1:MONITOR\_CYCLE\_END\_IN\_PASS 2:MONITOR\_CYCLE\_END\_IN\_FAIL |
| 返回值/类型 | 定义 | 取值范围 |
| Uint8 | Dtc\_ConfirmState | 0: DTC\_NOT\_CONFIRMED  1:DTC\_CONFIRMED |

* 1. 故障状态位更新逻辑

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | void Dtc\_StatusBitsUpdate(uint8 aIndex,uint8 monitorState,uint8 confirmState) | |
| 功能描述 | 对DTC状态位进行更新。 | |
| 逻辑描述 | 根据DTC的可重置性、Monitor Cycle的监控结果、14服务的清楚标志、历史故障的监控结果及状态位之间的关联进行状态位的确认和更新； | |
| 注意事项 | 无 | |
| 输入参数 | 定义 | 取值范围 |
| uint8 | aIndex | 0~255 |
| uint8 | monitorResult | 0~2  0:MONITOR\_CYCLE\_NOTCOMPLETE  1:MONITOR\_CYCLE\_END\_IN\_PASS 2:MONITOR\_CYCLE\_END\_IN\_FAIL |
| uint8 | confirmState | 0: DTC\_NOT\_CONFIRMED  1:DTC\_CONFIRMED |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. 初始化故障检测函数

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | void InitializationFault\_DTCStatusUpdate() | |
| 功能描述 | 初始化故障检测的总调度函数 | |
| 逻辑描述 | 略 | |
| 注意事项 | 被Dtc\_Init()调用，必须在Dtc\_Status\_PowerOn\_Init()之后调用 | |
| 输入参数 | 定义 | 取值范围 |
| 无 | —— | —— |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. 扩展数据更新逻辑

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | static void Dtc\_ExtDataUpdate(uint8 aIndex) | |
| 功能描述 | 更新扩展数据 | |
| 逻辑描述 | 故障发生点火周期计数的更新逻辑； | |
| 注意事项 | 无 | |
| 输入参数 | 定义 | 取值范围 |
| uint8 | aIndex | 0~255 |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. 冻结帧数据更新逻辑

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | static void Dtc\_FFDataUpdate(uint8 aIndex) | |
| 功能描述 | 更新冻结帧数据 | |
| 逻辑描述 | 根据故障对应的冻结帧索引，获得其所辖动态数据DID索引、DID数量，根据DID索引得到DID编码及各自的大小和buffer.  得到冻结帧数据； | |
| 注意事项 | 无 | |
| 输入参数 | 定义 | 取值范围 |
| uint8 | aIndex | 0~255 |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

* 1. 数据拷贝函数

|  |  |  |
| --- | --- | --- |
| 基本信息 | 内容 | |
| 函数名称 | void Dtc\_DataCopy( uint8\* Dest,const uint8\* Src,uint16 Cnt) | |
| 功能描述 | 数据拷贝函数 | |
| 逻辑描述 | 略 | |
| 注意事项 | 无 | |
| 输入参数 | 定义 | 取值范围 |
| uint8\* | Dest | -- |
| const uint8\* | Src | -- |
| uint16 | Cnt | 0~65535 |
| 返回值/类型 | 定义 | 取值范围 |
| 无 | —— | —— |

----------------------------完----------------------------