



易控 DSM 上层接口说明

目录

| | | |
|-----|-------------------------------|---|
| 1 | 相关名词说明..... | 2 |
| 1.1 | 故障检查项..... | 2 |
| 1.2 | 异常水平 Fault Level | 2 |
| 1.3 | 故障水平 Debounce Level..... | 2 |
| 1.4 | 故障状态 Debounce State..... | 2 |
| 1.5 | 故障确认标志..... | 2 |
| 1.6 | FID Function Identifier | 3 |
| 2 | 上层接口函数说明..... | 3 |
| 2.1 | DFC_ReportFaultLevel..... | 3 |
| 2.2 | DFC_GetErrState | 3 |
| 2.3 | DFC_GetDebState | 4 |
| 2.4 | DINH_GetFIDState..... | 4 |
| 3 | 程序设计说明..... | 5 |

1 相关名词说明

1.1 故障检查项

在 DFC 模块中定义，每个检查项对应一个故障码。每个故障检查项都包含了异常水平、故障水平、故障确认标志等信息（详见 DSM 标定说明）。在 DSM 接口函数中，*DFC_ReportFaultLevel* / *DFC_GetErrState* / *DFC_GetDebState* 均为更新或请求故障检查项状态信息。

1.2 异常水平 Fault Level

取值范围为 0~15，通过 *DFC_ReportFaultLevel* 函数进行更新。当异常水平大于等于 7 时，即认为异常发生。

1.3 故障水平 Debounce Level

取值范围为 0~15，反应了 debounce 进行到 0~100%。与异常水平共同决定了故障状态。

1.4 故障状态 Debounce State

包含四种状态：

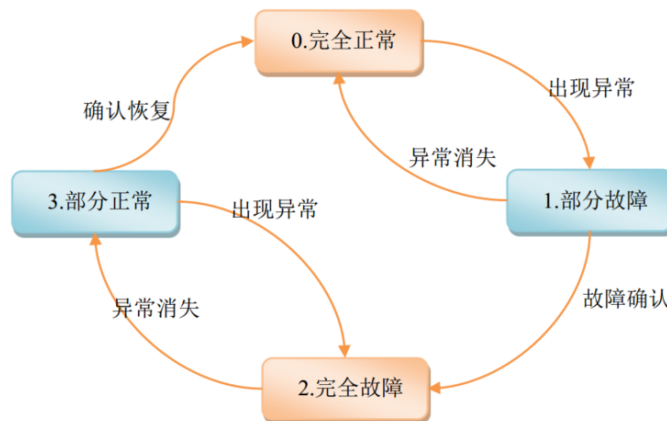


图 故障状态

1.5 故障确认标志

DFC 故障检查项状态之一，取值为 0（故障未确认）或 1（故障已确认）。

1.6 FID Function Identifier

功能识别号，当某一故障发生时，故障事件处理模块会禁止其对应的某些功能及其相关函数。

2 上层接口函数说明

上层程序与 DSM 之间接口函数共有四个，函数及相关宏定义均已在 HardwareLib.h 文件故障诊断部分声明。

2.1 DFC_ReportFaultLevel

```
void DFC_ReportFaultLevel (uint16 idxDFCUW, uint8 stFaultLevelUB)
```

功能：

DSM 模块入口函数。当上层算法检测到故障发生时，通过该函数通知 DSM 模块。该函数查询 DisableMask (`DFC_stDisblMskUB_CA[idxDFCUW]`) 确认该故障未被屏蔽后，开始进行 Debounce 计数，后续诊断功能才得以继续进行。

输入参数：

`idxDFCUW` - 故障检查项序号。故障检查项状态信息保存在底层 DFC 模块 `DFC_infoCheckUA` 中，使用该序号可查询 `DFC_infoCheckUA[idxDFCUW]` 保存的状态信息。每个故障所对应的故障码 (`DFES_numDTCBUW_CA`)、触发器 (`DSM_numOneHourUW_C`)、反抖方式 (`DDRC_idxDebTypeUB_CA`)、异常时禁用的 FID (`DINH_idxFIDUB_CM`) 等信息均可通过标定变量进行标定，详见《DSM 标定说明》。

`stFaultLevelUB` - 取值范围 0~15。0~6 为无异常状态，7~15 为异常状态。

2.2 DFC_GetErrState

```
uint8 DFC_GetErrState (uint16 idxDFCUW)
```

功能：

查询 DFC 当前故障是否被确认。当 `DFC_ReportFaultLevel` 报告故障后，DSM 开始按标定的方式 (`DDRC_idxDebTypeUB_CA`) 对该 DFC 进行 Debounce 计数，如果计数达到设定的限值 (`DDRC_ctDebDefUW_CA`)，则该 DFC 的 Error 状态置 1，故障解封时同理。调用该函数，可通过返回值判断 DFC 当前处于 Error 状态或者 OK 状态。

输入参数：

`idxDFCUW` - 故障检查项序号。故障检查项状态信息保存在底层 DFC 模块 `DFC_infoCheckUA` 中,使用该序号可查询 `DFC_infoCheckUA[idxDFCUW]`保存的状态信息。

返回值:

- 0 - 故障未确认
- 1 - 故障已确认

2.3 DFC_GetDebState

```
uint8 DFC_GetDebState(uint16 idxDFCUW)
```

功能:

查询 DFC 当前故障状态。DFC 存在四种故障状态:0.完全正常/1.部分故障/2.完全故障/3.部分正常。调用该函数,可通过返回值判断 DFC 当前故障水平。

输入参数:

`idxDFCUW` - 故障检查项序号。故障检查项状态信息保存在底层 DFC 模块 `DFC_infoCheckUA` 中,使用该序号可查询 `DFC_infoCheckUA[idxDFCUW]`保存的状态信息。

返回值:

- 0 - 完全正常
- 1 - 部分故障
- 2 - 完全故障
- 3 - 部分正常

2.4 DINH_GetFIDState

```
uint8 DINH_GetFIDState(uint8 idxFIDUB)
```

功能:

查询故障事件 FID 是否被禁止。当 DFC 的异常水平 (Fault Level) 和故障水平 (Debounce Level) 达到了标定的限值(`DINH_rLimLevelUB_CM`)时,与其相关的 FID 功能会被禁用或使能。调用该函数,可通过返回值查询该 FID 的禁用状态。

输入参数:

`idxFIDUB` - 功能识别号。某项功能及其相关函数对应的编号,用于在异常发生时禁用某些功能。故障检查项可通过 `DINH_idxFIDUB_CM` 变量进行标定。

返回值:

- 0 - FID 功能未禁止

1 - FID 功能已禁止

3 程序设计说明

上层程序与 DSM 模块之间通讯主要基于两个目的：一是上层程序根据设定条件监测运行出现异常，需要向 DSM 模块报告异常情况；二是需要查询某些故障的当前状态，并根据不同情况设计相应的控制算法。

以油门踏板控制程序 *AccPedCD.C* 为例，该模块中负责监控运行过程中是否出现相关异常的部分为函数 *AccPedCD_Monitor*。该函数中检测油门 1 和油门 2 信号是否出现过高或过低，并通过 *DFC_ReportFaultLevel* 将故障水平报告给 DSM。

在读取油门信号模拟值函数 *AccPedCD_DualAnalog/ AccPedCD_SingleAnalog* 中，读取信号之前需要先通过 *DFC_GetDebState* 读取油门信号相关 DFC 是否存在故障，并通过 *DINH_GetFIDState* 判断踏板信号故障失效模式是否开启。当确定有完全故障发生时，油门信号赋值为 0；当存在部分故障时，油门信号维持上一循环值不变；当状态完全正常时，才可以根据踏板实际位置对油门值进行更新。

该模块中没有使用到 *DFC_GetErrState* 函数，其功能与 *DFC_GetDebState* 函数类似。前者查询的是 DFC 故障是否被确认，后者反应了 DFC 的 Debounce 水平，可在开发过程中根据实际需要应用。