# 系统可靠性

## LVD

LVD模块包括LVD和LVR，分别是低电压检测和低电压复位的意思，这里的电压指的是MCU供电电压。LVD通常用于电平下降或上升时产生中断事件。LVR一旦使能，则会在电压下降到某个电平值后，使芯片进入复位状态，直到电压重新回到设定值以上。无论是LVR还是LVD，存在一个迟滞电压。具体参数请参考芯片的数据手册。

LVD和LVR可以设置为不同的电平值。LVD和LVR可以同时工作。

### API列表

Table 4‑1 系统可靠性 CSI接口函数

|  |  |  |
| --- | --- | --- |
| API | 说明 | 函数位置 |
| csi\_lvd\_int\_enable | 配置低电压检测的电平值，并使能中断 | reliability.c |
| csi\_lvd\_disable | 关闭低电压检测功能 |
| csi\_get\_lvdlevel | 读取当前低电压检测的电平值 |
| csi\_lvr\_enable | 配置低电压复位的电平值 |
| csi\_lvr\_disable | 关闭低电压复位功能 |
| csi\_get\_lvrlevel | 读取当前低电压复位的电平值 |

### API详细说明

#### csi\_lvd\_int\_enable

void csi\_lvd\_int\_enable(csi\_lvd\_pol\_e ePol, csi\_lvd\_level\_e eLvl)

##### 功能描述

该函数实现了3个功能：配置低电压检测的电平值、设置中断事件发生极性（下降到预设值或上升到预设值），并使能中断。

##### 参数/返回值说明

|  |  |  |
| --- | --- | --- |
| 参数 | 说明 | 枚举量位置 |
| ePol | 可选值为csi\_lvd\_pol\_e枚举值中的一个。  typedef enum {      LVD\_INTF = 1,   //电压下降到预设值时产生事件      LVD\_INTR,       //电压上升到预设值时产生事件      LVD\_INTFR       //电压下降或上升到预设值时产生事件  }clvd\_pol\_e; | reliability.h |
| eLvl | 可选值为csi\_lvd\_level\_e枚举值中的一个。  typedef enum{      LVD\_24 = 0,      LVD\_21,      LVD\_27,      LVD\_30,      LVD\_33,      LVD\_36,      LVD\_39,  }clvd\_level\_e; |

#### csi\_lvd\_disable

void csi\_lvd\_disable(void)

##### 功能描述

关闭芯片的低电压检测功能。注意，因为LVD和LVR实际是一个模块，所以关闭LVD的同时LVR功能也会被关闭。

##### 参数/返回值说明

无。

#### csi\_lvr\_enable

void csi\_lvr\_enable(csi\_lvr\_level\_e eLvl)

##### 功能描述

该函数用于配置触发复位的低电压值。

##### 参数/返回值说明

1. 参数：

eLvl：低电压复位的门限电平值。

1. 返回值：无。
2. 参数说明表

|  |  |  |
| --- | --- | --- |
| 参数 | 说明 | 位置 |
| eLvl | 可选值为csi\_lvr\_level\_e枚举值中的一个。  typedef enum {      LVR\_19 = 0,      LVR\_22,      LVR\_25,      LVR\_28,      LVR\_31,      LVR\_34,      LVR\_37,      LVR\_40  }clvr\_level\_e; | reliability.h |

## EMOSC 时钟监测

EMOSC时钟监测模块一旦使能，会持续监控外部EMOSC的状态。如果检测到振荡异常，可以有两种动作：复位，自动切换系统时钟到IMOSC，同时可以配置产生中断。

### API列表

|  |  |  |
| --- | --- | --- |
| API | 说明 | 函数位置 |
| csi\_emcm\_2\_imosc\_int | 使能EMCM功能，当监测到EMOSC异常时，将系统时钟切换到IMOSC，且触发中断。 | reliability.c |
| csi\_emcm\_rst | 使能EMCM功能，当监测到EMOSC异常时，系统复位。 |
| csi\_emcm\_disable | 关闭EMCM功能。 |

### API详细说明

略。见API列表中的说明。

## 内存检验

内存校验模块一旦使能，可以对SRAM和Flash的内容进行实时的检测。可以设置允许的检验错误次数上限。如果SRAM内容错误次数超过上限，可以有两种动作：复位和中断事件。如果Flash内容错误次数超过上限，芯片会直接复位。

### API列表

|  |  |  |
| --- | --- | --- |
| API | 说明 | 函数位置 |
| csi\_sramcheck\_set\_times | 设置SRAM校验允许的错误次数上限。复位值为0xffff。 | reliability.c |
| csi\_sramcheck\_rst | 设置SRAM校验错误次数超过设置的上限时，复位芯片。 |
| csi\_sramcheck\_int | 设置SRAM校验错误次数超过设置的上限时，产生中断。 |
| csi\_sramcheck\_disable | 禁止SRAM校验功能。 |
| csi\_flashcheck\_set\_times | 设置Flash校验允许的错误次数上限。复位值为0xffffff。 |
| csi\_flashcheck\_rst | 设置Flash校验错误次数超过设置的上限时，复位芯片。 |

### API详细说明

略。见API列表中的说明。

## 复位源

芯片在每次复位的时候都会对复位的原因进行记录。

### API列表

|  |  |  |
| --- | --- | --- |
| API | 说明 | 函数位置 |
| csi\_get\_rst\_reason | 获取上次复位的原因。 | reliability.c |
| csi\_clr\_rst\_reason | 清除复位信息 |

### API详细说明

#### csi\_get\_rst\_reason

uint16\_t csi\_get\_rst\_reason(void)

##### 功能描述

返回上次复位的原因。该功能可用于定位导致异常复位的原因，也可以是系统可靠运行的一部分：不同的复位条件转向不同的复位流程。

##### 参数/返回值说明

1. 参数：无
2. 返回值：

返回chip复位原因，复位信息的MASK值，BIT0~14，复位源详情请参考csi\_rsr\_src\_e枚举中定义。

1. 返回值说明表

|  |  |  |
| --- | --- | --- |
| 返回值 | 说明 | 概述 |
| 复位信息 | uint16\_t 类型数值，复位信息的MASK值；具体复位信息，请参考csi\_rsr\_src\_e中具体定义 |  |

#### csi\_clr\_rst\_reason

void csi\_clr\_rst\_reason(uint16\_t hwRstSrc)

##### 功能描述

清除chip复位信息。

##### 参数/返回值说明

1. 参数

hwRstSrc：复位信息的MASK值，BIT0~14，复位源详情请参考csi\_rsr\_src\_e枚举中定义。

1. 返回值：无
2. 参数说明

|  |  |  |
| --- | --- | --- |
| 参数 | 说明 | 概述 |
| 复位信息 | uint16\_t 类型数值，复位信息的MASK值 |  |

## CHIP软件复位

### API列表

|  |  |  |
| --- | --- | --- |
| API | 说明 | 函数位置 |
| csi\_sys\_swrst | Chip软件复位 | reliability.c |

### API详细说明

#### csi\_sys\_swrst

void csi\_sys\_swrst(void)

##### 功能描述

软件复位mcu。

##### 参数/返回值说明

1. 参数：无
2. 返回值：无

## UREG操作

### API列表

|  |  |  |
| --- | --- | --- |
| API | 说明 | 函数位置 |
| csi\_ureg\_write | 写用户寄存器 | reliability.c |
| csi\_ureg\_read | 读取用户寄存器 |

### API详细说明

#### csi\_ureg\_write

csi\_error\_t csi\_ureg\_write(csi\_user\_reg\_e eUreg, uint32\_t wValue)

##### 功能描述

写用户寄存器，除POR复位外，其余原因导致的复位，寄存器值保持不变。

##### 参数/返回值说明

1. 参数

eUreg：用户寄存器选项，详见枚举定义csi\_user\_reg\_e

wValue：用户写入寄存器值，eUreg = USER\_REG0/ USER\_REG1,为32位值，USER\_REG2时，为16位值。

1. 返回值：

CSI\_OK：初始化成功。

CSI\_ERROR：初始化失败。

1. 参数/返回值说明

|  |  |  |
| --- | --- | --- |
| 参数/返回值 | 说明 | 枚举量位置 |
| eUreg |  | reliability.h |
| wValue | 写入寄存器值，USER\_REG0/ USER\_REG1 为32位，其余为16位 |  |
| csi\_error\_t | csi\_error\_t 中定义值 | 在common.h中定义 |

#### csi\_ureg\_read

uint32\_t csi\_ureg\_read(csi\_user\_reg\_e eUreg)

##### 功能描述

读取用户寄存器存储值，除POR复位外，其余原因导致的复位，寄存器值保持不变。

##### 参数/返回值说明

1. 参数

eUreg：用户寄存器选项，详见枚举定义csi\_user\_reg\_e

1. 返回值：寄存器中存储值
2. 参数/返回值说明

|  |  |  |
| --- | --- | --- |
| 参数/返回值 | 说明 | 枚举量位置 |
| eUreg |  | reliability.h |
| 返回值 | 寄存器存储值，USER\_REG0/ USER\_REG1 为32位，其余为16位 |  |