

# MC56F84xxx和MC56F82xxx DSC上的eFlexPWM故障处理

作者: Pavel Sustek

## 1 简介

在外部故障情况下，需要对电机驱动或电源管理应用中的功率晶体管进行保护以防止失效。设计中选择的功率器件一般情况下可以可靠工作。然而，在故障情况下，器件可能会面临非常高的浪涌电流，只有通过内部或外部方法及时处理故障，才能避免开关器件失效。这里在可能出现系统故障的应用里面，采用了一种故障保护来检测故障，并通过关断栅极驱动来关闭晶体管。

典型的逆变器故障状态有：

- 过流
- 过压
- 过温

飞思卡尔MC56F84xxx和MC56F82xxx数字信号控制器（DSC）对脉宽调制（PWM）输出就具有这种故障保护功能。

## 目录

1	简介 .....	1
2	eFlexPWM故障保护 .....	2
2.1	概述 .....	2
2.2	故障输入 .....	4
2.3	故障毛刺拉伸逻辑 .....	5
2.4	故障有效电平 .....	5
2.5	PWM输出故障状态 .....	6
2.6	故障测试 .....	6
2.7	故障清除 .....	6
2.8	故障引脚滤波器 .....	10
2.9	故障中断 .....	12
3	应用程序注意事项 .....	12
3.1	PWM故障寄存器初始化 .....	12
3.2	故障源 .....	12
4	应用示例 .....	13
5	结语 .....	14
6	参考 .....	14
7	首字母缩略词和缩写词 .....	14
8	修订历史记录 .....	15

本应用笔记配有应用示例代码，提供即用型PWM模块故障配置。本文档扩展了MC56F84xxx和MC56F82xxx参考手册中的故障逻辑说明，并主要讨论应用案例。

## 2 eFlexPWM故障保护

### 2.1 概述

脉宽调制器（PWM）模块包含多个PWM子模块，每个子模块用于控制单个半桥功率电路，并提供故障入口。飞思卡尔MC56F84xxx和MC56F82xxx DSC均实现了故障保护逻辑。

故障保护可以控制任意PWM输出引脚组合，任意FAULTx引脚上的逻辑“1”或“0”都可以产生故障。每个FAULTx引脚均可映射在任意 PWM 输出上。当故障保护硬件禁止PWM输出时，PWM发生器会继续运行，故障逻辑和禁止映射寄存器（DISMAPn）选定相应的PWM引脚，这些引脚被故障解码器禁止输出，输出状态位强制设为逻辑“1”，逻辑“0”或三态。下图为故障禁止逻辑的一个示例。

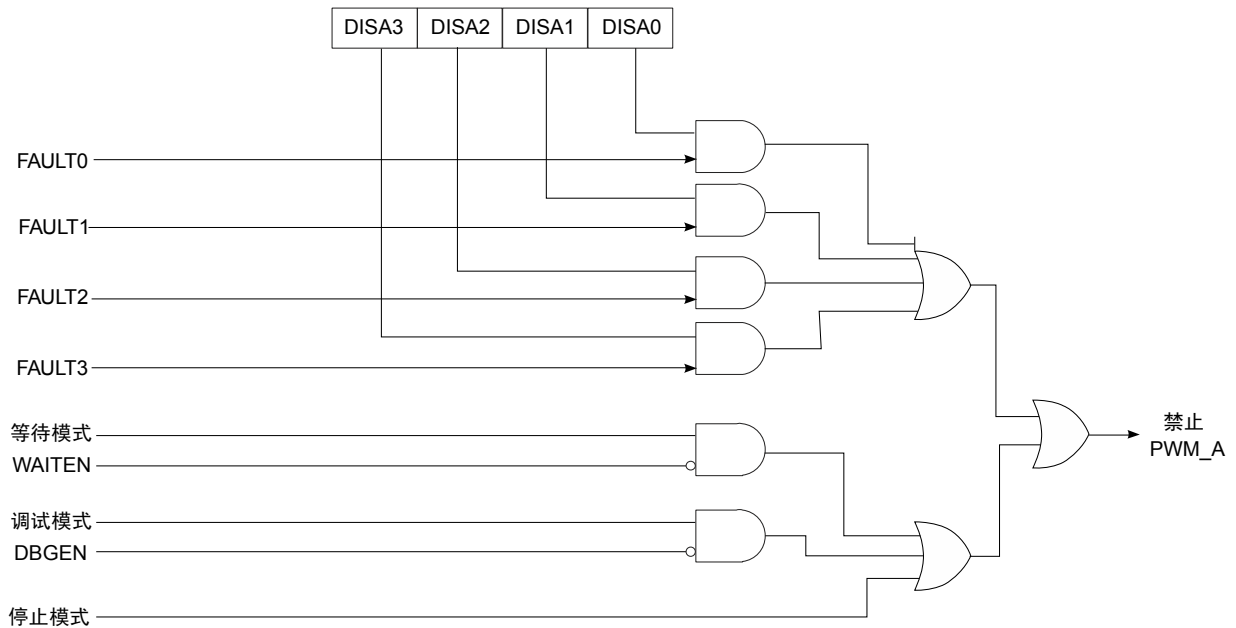


图1. PWM故障逻辑

MC56F84xxx和MC56F82xxx上的每个PWM模块包含两组专用于PWM故障设定的寄存器。第一组后缀为“0”，对应故障0-3，第二组后缀为“1”，对应故障4-7。Table 1显示了PWM故障相关的寄存器。每4个BIT对应于相应PWM输出的故障输入通道。例如，FIE位在PWMA\_FCTRL0中对应于故障3-0，在PWMA\_FCTRL1中的对应于故障7-4。

表1. PWM故障寄存器

寄存器名称	全称	域以及位的位置
PWMA(B)_SMxDISMAP0(1)	故障禁止映射	DIS0(1)A [3-0]—故障禁止屏蔽 DIS0(1)B [7-4]—故障禁止屏蔽 DIS0(1)X [11-8]—故障禁止屏蔽
PWMA(B)_SMxOCTRL	输出控制	PWMXFS [1-0]—PWM_X故障状态 PWMBFS [3-2]—PWM_B故障状态 PWMAFS [5-4]—PWM_A故障状态
PWMA(B)_FCTRL0(1)	故障控制	FIE [3-0]—故障中断开启 FSAFE [7-4]—故障安全模式 FAUTO [11-8]—自动故障清除 FLVL [15-12]—故障有效电平
PWMA(B)_FSTS0(1)	故障状态	FFLAG [3-0]—故障标志 FFULL [7-4]—故障周期 FFPIN [11-8]—过滤故障引脚 FHALF [15-12]—半周期故障恢复
PWMA(B)_FFILT0(1)	故障滤波器	FILT_PER [7-0]—故障滤波器周期 FILT_CNT [10-8]—故障滤波器计数 GSTR [15]—故障毛刺拉伸开启
PWMA(B)_FTST0(1)	故障测试	FTEST [0]—故障测试

eFlexPWM故障可通过图形配置工具（GCT）轻松配置，该工具是QuickStart工具 [3]中的一部分。下图显示了GCT面板提供的故障设定。

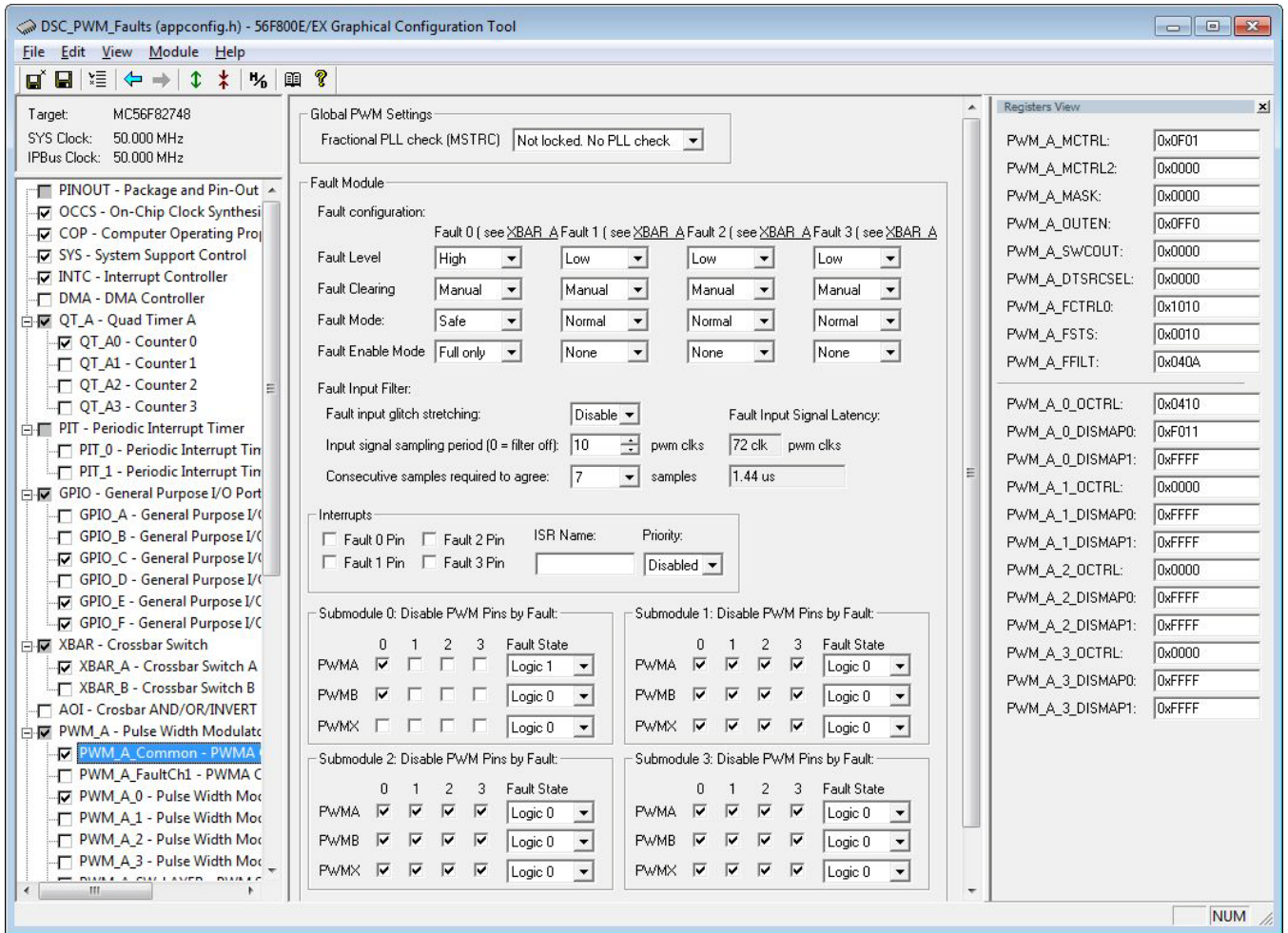


图2. 故障配置GCT面板

## 2.2 故障输入

eFlexPWM包含最多8个故障输入，可分配用于控制多个PWM输出。故障源可以是内部和外部的。内部故障源通过XBAR通道连接到PWM外设，外部故障源通过专用GPIO引脚连接。下列表格显示了MC56F84xxx（包含两个PWM模块）和MC56F82xxx（包含一个PWM模块）上可用的故障源。一些外部故障源在低引脚数器件封装上不可用。

表2. eFlexPWM故障寄存器

故障编号	MC56F84xxx		MC56F82xxx
	PWMA	PWMB	PWMA
FAULT0	XBAR_OUT29 / GPIO_E8	XBAR_OUT29 / GPIO_F14	XBAR_OUT29
FAULT1	XBAR_OUT30 / GPIO_E9	XBAR_OUT30 / GPIO_F13	XBAR_OUT30

FAULT2	XBAR_OUT31 / GPIO_G4	XBAR_OUT31 / GPIO_F12	XBAR_OUT31
FAULT3	XBAR_OUT32	XBAR_OUT32	XBAR_OUT32
FAULT4	GPIO_G6	GPIO_G6	GPIO_C14
FAULT5	GPIO_G7	GPIO_G7	GPIO_C15
FAULT6	GPIO_F10	GPIO_F10	GPIO_F4
FAULT7	GPIO_F9	GPIO_F9	GPIO_F5

GPIO功能（作为PWM FAULT输入）通过专门的外设选择寄存器（SIM\_GPSxx）进行选择。在XBAR\_OUTxx和GPIO\_xx（仅MC56F84xxx）之间的故障源选择通过内部外设选择寄存器（SIM\_IPS0）进行设定。

经过过滤的FAULTx的实际状态通过只读位FFPINx来反映，无论故障有效电平（FLVLx）如何，故障发生该位被置高，即逻辑1表示故障条件存在于过滤FAULTx引脚上。如果禁止滤波器，则FFPINx直接反映FAULTx引脚上的变化。

## 2.3 故障毛刺拉伸逻辑

故障毛刺拉伸逻辑代表PWMA\_FFILT<sub>x</sub>寄存器中的GSTR域。该逻辑确保较窄的故障毛刺被拉伸为至少两个IPBus时钟周期宽度。在某些情况下，由于PWM输出关断/重新激活时间较短，窄的故障输入可能导致问题。

例如，在FAULTx引脚发生转换后，将FFLAGx位设定为在两个IPBus时钟周期内。拉伸逻辑确保故障输入上的毛刺（故障滤波器禁止时）将被记录到故障标志中。

## 2.4 故障有效电平

故障输入的有效逻辑电平配置通过PWMA(B)\_FCTRL0（故障0–3）和PWMA(B)\_FCTRL1（故障4–7）寄存器中的FLVL域。

- 如果FLVL<sub>x</sub> = 0，故障输入FAULT<sub>x</sub>上的逻辑0表示故障。
- 如果FLVL<sub>x</sub> = 1，故障输入FAULT<sub>x</sub>上的逻辑1表示故障。

在故障状态和STOP模式期间，定义PWM输出故障状态的位位于PWMA(B)\_SMnOCTRL寄存器（n是PWM子模块的编号）。可在PWMxFS位域中分别设置每个PWM子模块及其输出A、B和X的状态。

## 2.5 PWM输出故障状态

位设定决定了以下PWM输出状态：

位设定	PWM输出故障状态
00	在考虑输出极性控制之前，输出强制为逻辑0状态。
01	在考虑输出极性控制之前，输出强制为逻辑1状态。
10	输出为三态。
11	输出为三态。

输出PWM的状态受POLx位（PWMA\_SMnOCTRL）中设置的输出极性的影响。如果输出极性反向（POLx为逻辑1），则实际PWM输出故障状态也将反向（三态除外）。

## 2.6 故障测试

故障测试（FTEST）域允许模拟故障状态。在难以产生合适故障信号时，用于调试和应用测试。

- 设定该域会导致向所有故障过滤器发送模拟故障，置位故障标志和其他可能的位，PWM输出取决于DISMAPn的设定。
- 清零该域将移除模拟故障。

## 2.7 故障清除

模块支持两种清除现有故障条件的模式。

- 自动故障清除
- 手动故障清除

故障清除模式对应PWMA(B)\_FCTRL0（故障0–3）和PWMA(B)\_FCTRL1（故障4–7）寄存器中的FAUTO域。

- FAUTOx = 0：手动故障清除
- FAUTOx = 1：自动故障清除

PWM输出可在PWM半周期（FHALFx位置1）或PWM全周期（FFULLx位置1）开始处开启。至少一个控制位必须置1以开启PWM输出，否则不可能实现故障恢复。如果FHALFx和FFULLx位均置1，则故障恢复在全周期开始处和半周期开始处发生（由VAL0定义）。

## 2.7.1 自动故障清除

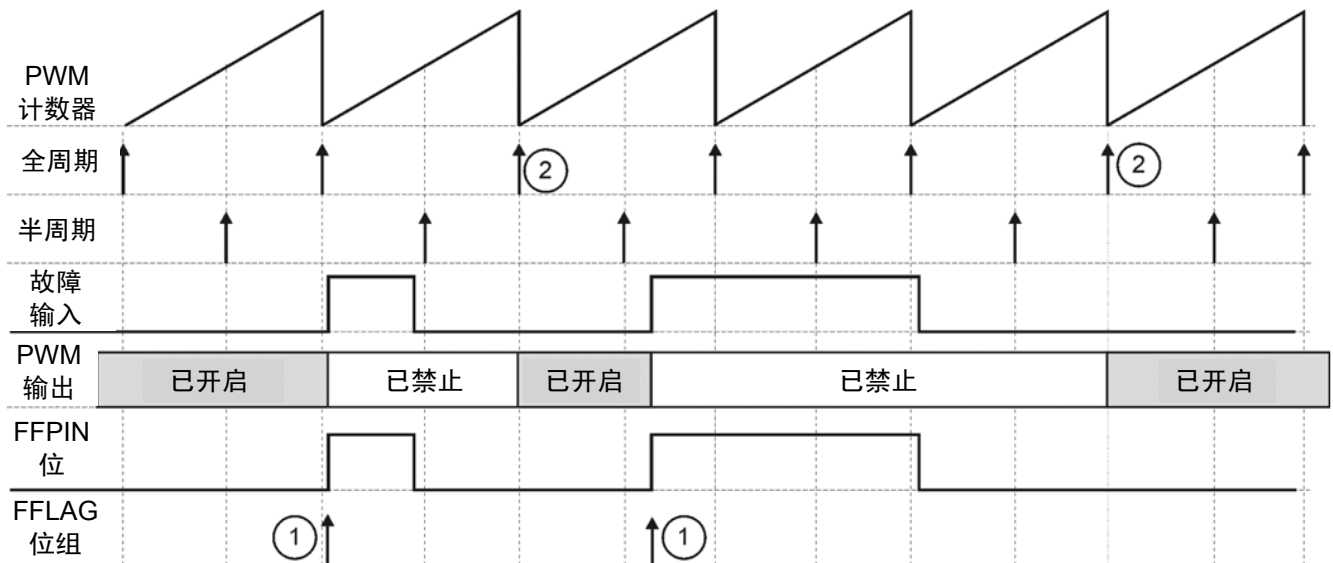
当FAUTO<sub>x</sub>置1时，在以下情况下开启已禁止的PWM输出引脚：

- FAULT<sub>x</sub>输入无故障信号，FAULT<sub>x</sub>引脚返回逻辑1（FLVL<sub>x</sub> = 0）或逻辑0（FLVL<sub>x</sub> = 1）
- 新PWM从全周期FFULL<sub>x</sub> = 1（Figure 3）或半周期FHALF<sub>x</sub> = 1（Figure 4）开始

当FAUTO<sub>x</sub>置1时，清零FFLAG<sub>x</sub>位不影响禁止PWM引脚。

下图中显示的信号如下所示。

- PWM计数器—计数PWM输入时钟，从INIT值到VAL1值
- 全周期—比较计数器溢出处的值（VAL1寄存器）
- 半周期—典型的PWM半周期由VAL0寄存器表示
- 故障输入—来自内部或外部源的故障信号
- PWM输出—PWM输出状态，可由故障输入禁止（DISMAP<sub>x</sub>寄存器）
- FFPIN位—反映经过过滤后故障输入状态的位
- FFLAG位—FFPIN故障状态切换的故障标志



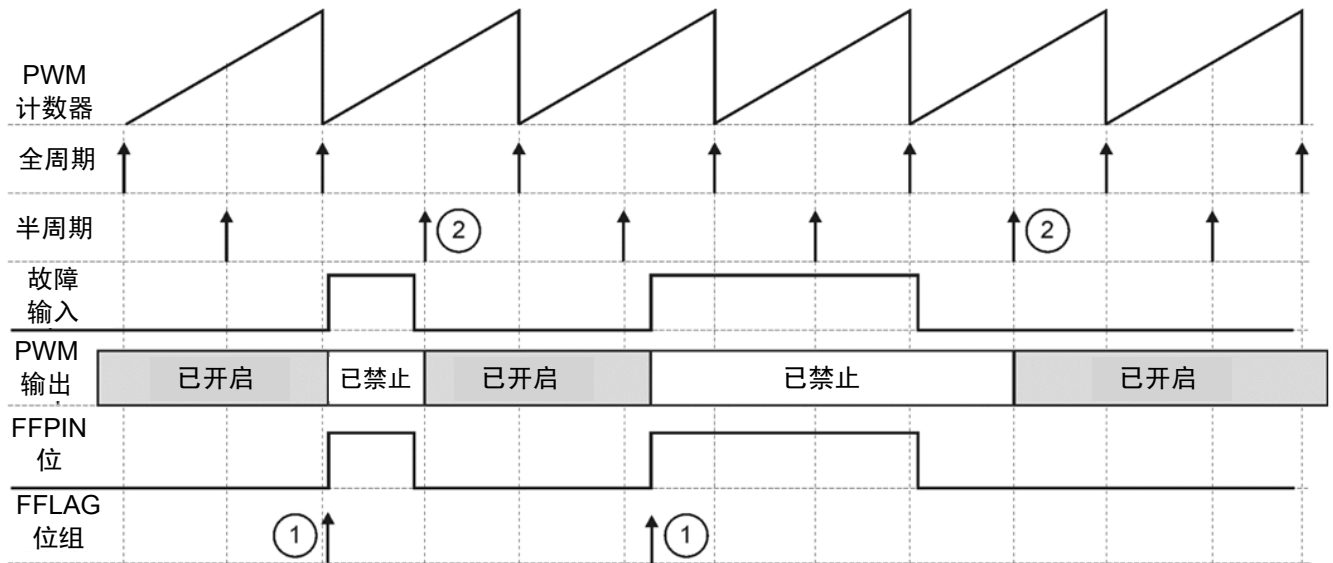
1 故障输入信号显示故障，禁止PWM输出，故障标志置1

2 故障输入信号返回低电平之后的第一个全周期，重新开启PWM输出

图3. 全周期下的自动故障清除

这些代码行设置自动故障清除模式。

```
PWM_A_FCTRL0 = 0x1100 /* FLVL = 1, FAUTO = 1, FSAFE = 0 */
PWM_A_FSTS = 0x0010 /* FFULL = 1, FHALF = 0 */
PWM_A_FFILT = 0x0000 /* FILT_COUNT = 0*/
```



1 故障输入信号显示故障，禁止PWM输出  
 2 故障输入信号返回低电平之后的第一个半周期，重新开启PWM输出

图4. 半周期下的自动故障清除

这些代码行设置PWM半周期下的自动故障清除。

```
PWM_A_FCTRL0 = 0x1100 /* FLVL = 1, FAUTO = 1, FSAFE = 0 */
PWM_A_FSTS = 0x1000 /* FFULL = 0, FHALF = 1 */
PWM_A_FFILT = 0x0000 /* FILT_COUNT = 0*/
```

## 2.7.2 手动故障清除

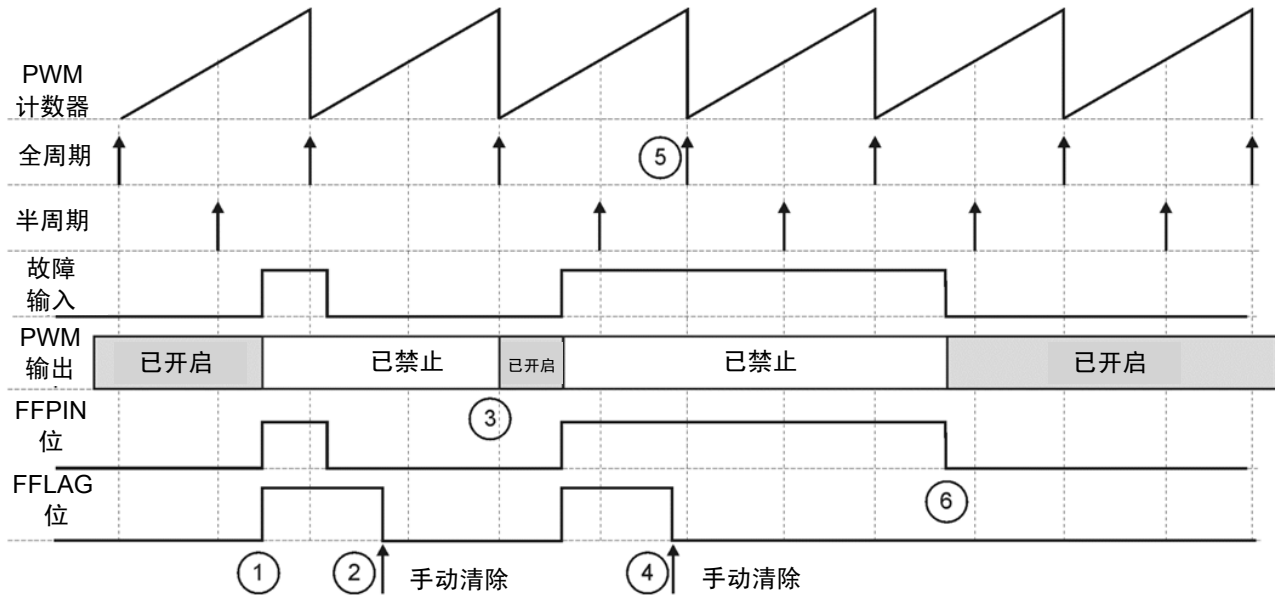
清零自动清除模式位（FAUTO<sub>x</sub> = 0），设置手动模式。

如果禁止安全模式（FSAFE<sub>x</sub> = 0），则在故障恢复时不会检查FFPIN<sub>x</sub>的状态。在以下情况下开启已禁止的PWM输出引脚：

- 软件清除了相应的FFLAG<sub>x</sub>标志
- 新PWM全周期（FFULL<sub>x</sub> = 1）或半周期（FHALF<sub>x</sub> = 1）开始

更多详细信息，请见下图。





- 1 故障输入信号显示故障，禁止PWM输出
- 2 手动清除FFLAG
- 3 在第一个即将到来的全周期重新开启PWM输出
- 4 在故障输入信号显示故障之后手动清除FFLAG，保持禁止PWM输出
- 5 在禁止安全模式的情况下，在FFLAG清除之后的半周期或全周期，无论FFPIN状态如何，均开启PWM输出。但是，如果故障输入信号显示故障，模块逻辑中仍存在禁止PWM输出的组合路径。因此，保持禁止PWM输出。
- 6 故障输入信号返回低电平，立即重新开启PWM输出

**图5. 全周期下的手动故障清除**

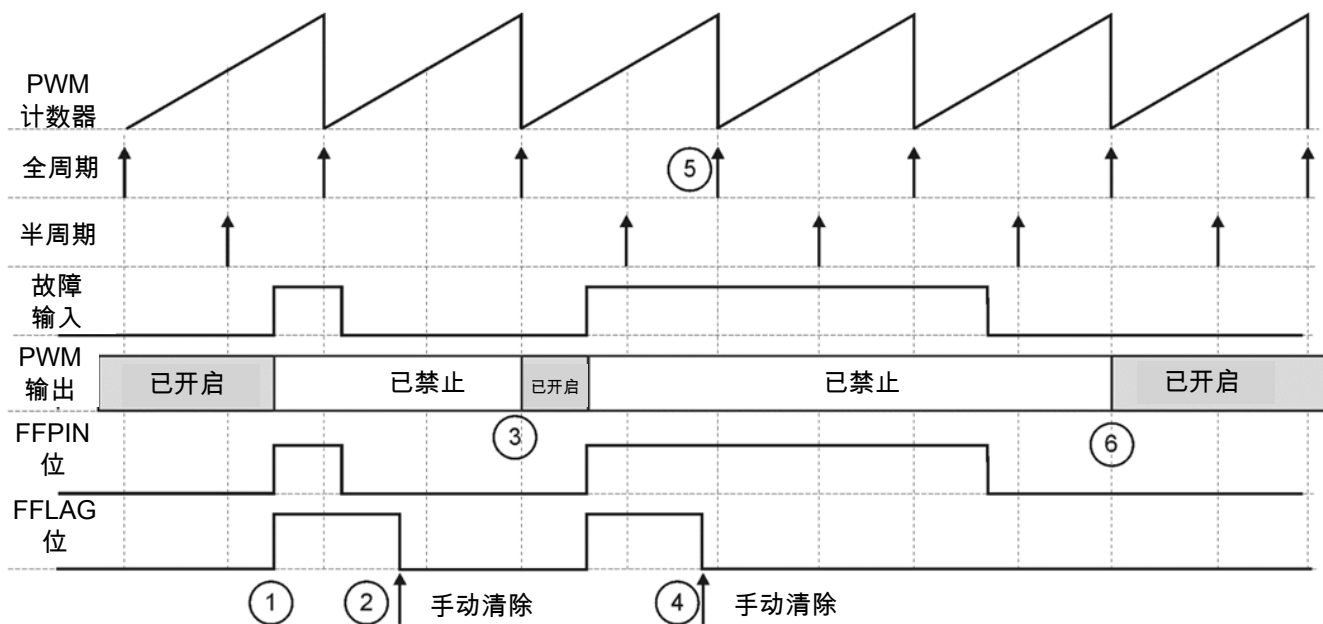
这些代码行设置PWM全周期下的手动故障清除。

```
PWM_A_FCTRL0 = 0x1000 /* FLVL = 1, FAUTO = 0, FSAFE = 0 */
PWM_A_FSTS = 0x0010 /* FFULL = 1, FHALF = 0 */
PWM_A_FFILT = 0x0000 /* FILT_COUNT = 0*/
```

如果开启安全模式（FSAFE<sub>x</sub> = 1），则在故障恢复时必须清零FFPIN<sub>x</sub>的状态。在以下情况下开启已禁止的PWM输出引脚：

- 软件清除了相应的FFLAG<sub>x</sub>标志
- 新PWM全周期（FFULL<sub>x</sub> = 1）或半周期（FHALF<sub>x</sub> = 1）开始
- 在恢复瞬间及下一个PWM全/半周期边界开始处，FAULT<sub>x</sub>引脚上不会检测到故障信号

更多详细信息，请见下图。



- 1 故障输入信号显示故障，禁止PWM输出
- 2 手动清除FFLAG
- 3 在第一个即将到来的全周期重新开启PWM输出
- 4 在故障输入信号显示故障之后手动清除FFLAG，保持禁止PWM输出
- 5 在开启安全模式的情况下，检查FFLAG和FFPIN以便在全周期清除。PWM输出仍保持禁止
- 6 在全周期FFLAG = 0，FFPIN = 0，重新开启PWM输出

图6. 全周期下的手动故障清除，开启安全模式

这些代码行设置PWM全周期下，开启安全模式的手动故障清除。

```
PWM_A_FCTRL0 = 0x1010 /* FLVL = 1, FAUTO = 0, FSAFE = 1 */
PWM_A_FSTS = 0x0010 /* FFULL = 1, FHALF = 0 */
PWM_A_FFILT = 0x0000 /* FILT_COUNT = 0*/
```

## 2.8 故障引脚滤波器

每个故障引脚都有一个可旁路的可编程滤波器。必须注意的是，即使开启了滤波器，还是有组合途径可以禁止PWM输出。这可以确保对故障条件做出快速响应，也可以在PWM模块丢失时钟的情况下确保故障响应。这意味着在FAULTx输入信号指示故障期间PWM输出始终禁止，无论PWM滤波器是否开启（见Figure 7）。

故障滤波器延迟了PWM故障中断的产生，但是在滤波器周期内不会消除PWM输出禁止。

故障滤波器的设置由两个阈值组成：FILT\_PER和FILT\_CNT。两个域均位于PWMx\_FCTRL寄存器。

- PWMx\_FCTRL[FILT\_PER]—设置滤波器的采样周期。应将值设置为采样周期大于预期噪声周期。这样一个噪声尖峰只会影响一次采样。对于给定的FAULTx引脚，将FILT\_PER设为全0以禁止输入滤波器。当修改FILT\_PER的值从一个非零值修改为另一非零值时，首先要写入0来清零滤波器。设置FILT\_PER域的IPBus时钟周期的最大数值为255。
- PWMx\_FCTRL[FILT\_CNT]—设置连续采样的数量，所有这些连续采样值必须有效才能产生一个有效的输入故障跳变。选择的值应降低噪声采样导致识别错误转换的可能性。采样数量为该域的十进制值+3：0–7的位域值分别表示3–10次采样。

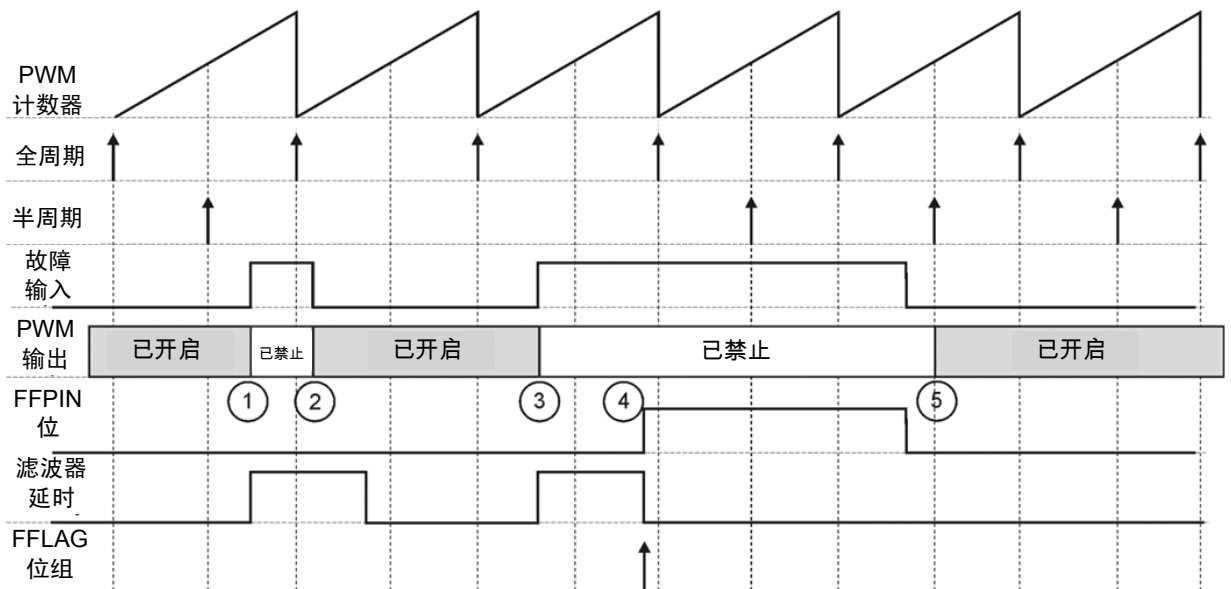
识别输入故障转换的延迟可通过以下公式计算。

$$\text{FAULT FILT LATENCY} = (\text{FILT CNT} + 4) * \text{FILT PER} * \text{IPBUS clock period}$$

滤波器引入的延迟将在设置PWMx\_FSTS寄存器的FFLAG和FFPIN域时体现。

如果开启故障滤波器（PWMx\_FCTRL[FILT\_PER] > 0），在检测到过滤FAULTx引脚上的逻辑0后（或FLVLx置1时为逻辑1）；并且在由滤波器延迟定义的时间之后，相应的FFPINx位置1，故障标志位FFLAGx也置1。

FFPINx在过滤后FAULTx引脚为0（或FLVLx置1时为逻辑1）期间始终置1。向FFLAGx写入逻辑1进行清零。



- 1 故障输入信号显示故障，禁止PWM输出
- 2 滤波器延时长于故障脉冲，但是在故障输入信号显示故障时禁止PWM输出
- 3 故障输入信号显示故障
- 4 滤波器传播延迟，FFPIN = 1，FFLAG = 1
- 5 半周期下的自动故障清除，开启PWM输出

图7. 半周期下的自动故障清除，滤波器开启

这些代码行设置PWM半周期下，开启滤波器的自动故障清除。

```
PWM_A_FCTRL0 = 0x1010 /* FLVL = 1, FAUTO = 1, FSAFE = 0 */
PWM_A_FSTS = 0x0010 /* FFULL = 1, FHALF = 0 */
PWM_A_FFILT = 0x011E /* FILT_FILT_COUNT = 4, FILT_PER = 30 */
```

## 2.9 故障中断

故障中断开启（FIE）位开启由FAULT<sub>x</sub>引脚产生的CPU中断请求（FFLAG<sub>x</sub>置1）。FIE<sub>x</sub>位位于PWMA(B)\_FCTRL0（故障0–3）和PWMA(B)\_FCTRL1（故障4–7）寄存器。

- 如果FIE<sub>x</sub> = 0，FAULT<sub>x</sub> CPU中断请求禁止。
- 如果FIE<sub>x</sub> = 1，FAULT<sub>x</sub> CPU中断请求开启。

如果相应FAULT<sub>x</sub>引脚的中断开启位在FIE<sub>x</sub>位域中置1，

FFLAG<sub>x</sub>会产生CPU中断请求。中断请求会被锁存直到满足以下任一条件：

- 软件通过向此位写入逻辑1将FFLAG<sub>x</sub>清零。
- 软件通过向其写入逻辑0来清除FIE<sub>x</sub>位。
- 发生复位。

故障保护电路独立于FIE<sub>x</sub>位，并且始终运行。若检测出故障，将根据禁止映射寄存器PWMA(B)\_SMxDISMAP0(1)禁止PWM输出。

## 3 应用程序注意事项

### 3.1 PWM故障寄存器初始化

PWM模块逻辑在复位后，将所有FAULT输入的FFLAG<sub>x</sub>标志（寄存器FSTS0为故障0-3，FSTS1为故障4-7）设置为逻辑1。在应用程序初始化期间，建议在开启中断前向FFLAG<sub>x</sub>标志写入逻辑1以清零这些位。否则，在FIE<sub>x</sub>中的相应位置1后会立即产生相应的PWM故障中断。

建议仅将DIS0(1)<sub>x</sub>位（PWMA(B)\_SMnDISMAP<sub>x</sub>寄存器）置1，该位对应于应用程序中使用的FAULT<sub>x</sub>。DIS0(1)<sub>x</sub>位的复位值为逻辑1。

### 3.2 故障源

产生故障条件的信号如下。

- 内部—通过XBAR外设连接到PWM FAULT<sub>x</sub>输入
- 外部—从特定的GPIO引脚连接到PWM FAULT<sub>x</sub>输入

故障输入的内部源通常为比较器、程序延时模块（PDB）、四通道定时器(QTIMER)或AOI逻辑模块之类的外设。

可将内部比较器用作过流保护，其中一路输入连接到内部参考（DAC），另一路输入连接到反映测量电流的信号。比较器输出通过XBAR通道连接到PWM故障输入，见下图。

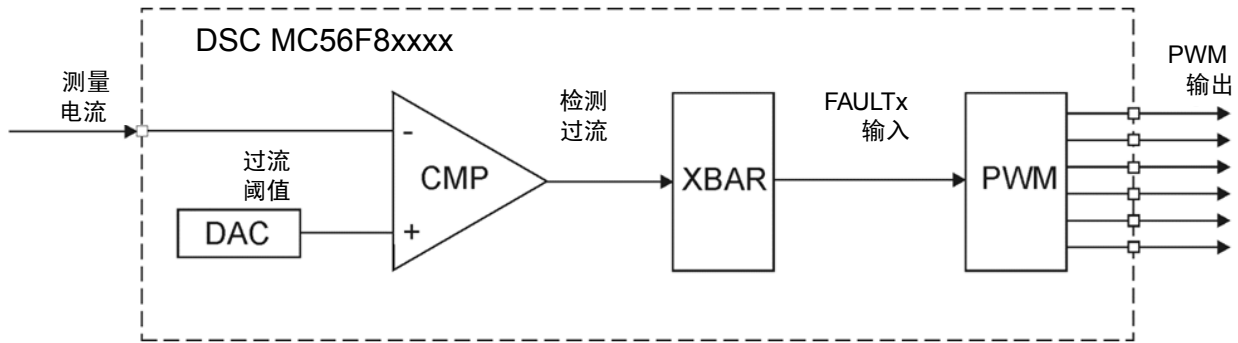


图8. 内部故障源连接

外部源通常为比较器、温度传感器等。建议将外部RC滤波器用于故障信号，以避免由于电路板上的噪声产生的无用故障信号。虽然PWM模块包含一个故障滤波器，不过当故障在FAULTx输入上生效期间，仍有组合途径来禁止PWM输出（见Figure 5和Figure 7）。

## 4 应用示例

应用笔记附带的示例应用程序提供了PWM故障的典型配置，包括内部和外部故障源信号。

代码基于CodeWarrior v.10.4、QuickStart 2.6和FreeMASTER 1.3.15。使用TWR-56F8200作为硬件电路板。应用示例的跳线设置如下表所示。

表3. TWR-56F8200的跳线设置

跳线名称	位置
J4	1-2 3
J5	1 2-3
J8	1 2 3 4 5
J9	1 2 3 4 5

内部故障信号由内部比较器HSCMP\_A产生。比较器比较了由6位DAC设置的0.567 V参考信号和12位DAC产生的三角波形。该周期性的比较器输出脉冲通过XBAR通道30连接到PWM故障输入1。

故障有效电平设置为高（FLVL1 = 1）且禁止PWM\_A1和PWM\_B1输出，将其设置为逻辑0。故障清除设置为自动模式，恢复时间为半周期。

外部故障信号通过按下SW1按钮产生。故障信号通过XBAR通道29连接到PWM故障输入0。故障有效电平设为低（FLVL0 = 0）且禁止PWM\_A0和PWM\_B0输出，将其设置为逻辑0。如果PWM\_A0输出设置为低电平有效，所以当输出被故障禁止时，PWM\_A0输出会变成高电平。故障标志产生中断请求。中断服务程序Extern\_Fault\_ISR()中的代码会点亮LED8，直到按下SW2按钮，清除故障标志，开启PWM\_A0和PWM\_B0输出，关闭LED8。

PWM输出A0和B0连接到LED0和LED1，A1和B1连接到LED2和LED3。外设配置在GCT中完成，GCT可以配置所有的外设寄存器。

## 5 结语

该应用笔记从应用角度描述了eFlexPWM故障逻辑的详细配置。逆变器需要合适的故障保护设置，以避免出现危险状态。应用笔记提供的应用程序源代码支持使用最新工具进行客户应用程序开发。

## 6 参考

可从[freescale.com](http://freescale.com)获取以下文档。

1. MC56F827xx 参考手册（文档MC56F8277XRM）
2. MC56F847xx 参考手册（文档MC56F847XXRM）
3. DSC56800EX 快速入门用户指南（文档DSC56800EXQSUG）

## 7 首字母缩略词和缩写词

该表包含了本应用笔记中使用的缩写词。

表4. 首字母缩略词和缩写词

术语	含义
LED	发光二极管
DSC	数字信号控制器
IPBus	内部外设总线
PWM	脉宽调制
XBAR	外设间交叉开关

## 8 修订历史记录

修订版本号	日期	重大变更
0	2013/08	首次发行

**How to Reach Us:**

**Home Page:**

[freescale.com](http://freescale.com)

**Web Support:**

[freescale.com/support](http://freescale.com/support)

本文档中的信息仅供系统和软件实施方使用 Freescale 产品。本文并未明示或暗示授予利用本文档信息进行设计或者加工集成电路的版权许可。飞思卡尔保留对本文档中所述任何产品进行更改的权利，恕不另行通知。

飞思卡尔对其产品在任何特定用途方面的适用性不做任何担保、表示或保证，也不承担因为应用或使用产品或电路所产生的任何责任，明确拒绝承担包括但不限于因果性或附带损害在内的所有责任。飞思卡尔数据表和/或技术规格中所提供的“典型”参数在不同应用中可能，并且确实不同，实际性能会随时间而有所变化。所有操作参数，包括“典型值”在内，必须由客户的技术专家根据每个客户应用进行验证。飞思卡尔未转让与其专利权及其他权利相关的许可。Freescale 销售产品时遵循以下网址中包含的标准销售条款和条件：  
[freescale.com/SalesTermsandConditions](http://freescale.com/SalesTermsandConditions).

Freescale, the Freescale logo, and CodeWarrior are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. Tower is a trademark of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

© 2013 Freescale Semiconductor, Inc.

Document Number: AN4795

Revision 0, August 2013