



通用输入输出端口



通用输入输出端口

人的身体通过眼睛、鼻子、手、脚等器官来同外界打交道，接收外界的信息并作出相应的反应。DSP也一样，它需要通过输入/输出引脚来跟外围交换信息。F28335提供了88个多功能引脚，为了节省资源，这些引脚是复用的，既可以作为片内外设的输入/输出引脚，也可以作为通用的数字I/O口。本章将详细介绍由这些引脚所组成的通用输入/输出多路复用端口GPIO的工作原理及相关的寄存器。寄存器具体的定义见C2000助手软件。



F28335为用户提供了88个多功能复用引脚GPIO0 ~ GPIO87。复用的意思是每个引脚都可以配置成通用的数字I/O工作模式，也可以配置成外设I/O工作模式，例如外设SCI、ePWM、eQEP的引脚，每个引脚除了通用数字I/O功能外，通常还有三个外设功能。GPIO引脚究竟用作哪种功能，可以通过寄存器来设置。

F28335的通用输入/输出端口GPIO就是用I/O引脚的管理机构，它将88个引脚分成A、B、C三组或者叫三个端口来进行管理，其中端口A包括GPIO0 ~ GPIO31，端口B包括GPIO32 ~ GPIO63，端口C包括GPIO64 ~ GPIO87。



GPIO寄存器

对于F28335输入/输出引脚的操作，都是通过对寄存器的设置来实现的。例如，选择某个引脚功能是做外设引脚还是做通用数字I/O口，当引脚作为通用数字I/O口时，是做输入还是做输出，如何使其输出高电平或者低电平，如何使其引脚电平翻转，如何知道引脚上的电平是高或者是低，这些都是通过对GPIO寄存器的操作来实现的，每个I/O引脚都可以通过寄存器相应的位得到设置。GPIO的寄存器分成了三大类，第一类是控制寄存器，主要由功能选择控制寄存器GPxMUXn、方向控制寄存器GPxDIR、输入限定控制寄存器GPxQSELn等组成，x代表A、B、C，n代表1或者2，见表6-1；第二类是数据寄存器，主要由数据寄存器GPxDAT、置位寄存器GPxSET、清除寄存器GPxCLEAR和状态翻转寄存器GPxTOGGLE等组成，见表6-2；第三类是外部中断源及低功耗模式唤醒选择寄存器，比如外部中断源XINT1输入端口选择寄存器GPIOXINT1SEL，见表6-3。需要注意的是第一类和第三类寄存器是受EALLOW保护的，也就是说对这些寄存器进行设置时，在操作前需要写EALLOW，在操作完成后要写EDIS，否则操作是无效的。第二类寄存器没有这个限制。



GPIO寄存器

名称	地址	大小 (×16bit)	寄存器说明
GPACTRL	0x6F80	2	GPIOA控制寄存器 (GPIO0 ~ GPIO31)
GPAQSEL1	0x6F82	2	GPIOA输入限定选择寄存器1 (GPIO0 ~ GPIO15)
GPAQSEL2	0x6F84	2	GPIOA输入限定选择寄存器2 (GPIO16 ~ GPIO31)
GPAMUX1	0x6F86	2	GPIOA功能选择控制寄存器1 (GPIO0 ~ GPIO15)
GPAMUX2	0x6F88	2	GPIOA功能选择控制寄存器2 (GPIO16 ~ GPIO31)
GPADIR	0x6F8A	2	GPIOA方向控制寄存器 (GPIO0 ~ GPIO31)
GPAPUD	0x6F8C	2	GPIOA上拉控制寄存器 (GPIO0 ~ GPIO31)
GPBCTRL	0x6F90	2	GPIOB控制寄存器 (GPIO32 ~ GPIO63)
GPBQSEL1	0x6F92	2	GPIOB输入限定选择寄存器1 (GPIO32 ~ GPIO47)
GPBQSEL2	0x6F94	2	GPIOB输入限定选择寄存器2 (GPIO48 ~ GPIO63)
GPBMUX1	0x6F96	2	GPIOB功能选择控制寄存器1 (GPIO32 ~ GPIO47)
GPBMUX2	0x6F98	2	GPIOB功能选择控制寄存器2 (GPIO48 ~ GPIO63)
GPBDIR	0x6F9A	2	GPIOB方向控制寄存器 (GPIO32 ~ GPIO63)
GPBPUD	0x6F9C	2	GPIOB上拉控制寄存器 (GPIO32 ~ GPIO63)
GPCMUX1	0x6FA6	2	GPIOC功能选择控制寄存器1 (GPIO64 ~ GPIO79)
GPCMUX2	0x6FA8	2	GPIOC功能选择控制寄存器2 (GPIO80 ~ GPIO87)
GPCDIR	0x6FAA	2	GPIOC方向控制寄存器 (GPIO64 ~ GPIO87)
GPCPUD	0x6FAC	2	GPIOC上拉控制寄存器 (GPIO64 ~ GPIO87)

表6-1 GPIO控制寄存器



GPIO寄存器

名称	地址	大小 (×16bit)	寄存器说明
GPADAT	0x6FC0	2	GPIOA数据寄存器 (GPIO0 ~ GPIO31)
GPASET	0x6FC2	2	GPIOA置位寄存器 (GPIO0 ~ GPIO31)
GPACLEAR	0x6FC4	2	GPIOA清零寄存器 (GPIO0 ~ GPIO31)
GPATOGGLE	0x6FC6	2	GPIOA状态翻转寄存器 (GPIO0 ~ GPIO31)
GPBDAT	0x6FC8	2	GPIOB数据寄存器 (GPIO32 ~ GPIO63)
GPBSET	0x6FCA	2	GPIOB置位寄存器 (GPIO32 ~ GPIO63)
GPBCLEAR	0x6FCC	2	GPIOB清零寄存器 (GPIO32 ~ GPIO63)
GPBTOGGLE	0x6FCE	2	GPIOB状态翻转寄存器 (GPIO32 ~ GPIO63)
GPCDAT	0x6FD0	2	GPIOC数据寄存器 (GPIO64 ~ GPIO87)
GPCSET	0x6FD2	2	GPIOC置位寄存器 (GPIO64 ~ GPIO87)
GPCCLEAR	0x6FD4	2	GPIOC清零寄存器 (GPIO64 ~ GPIO87)
GPCTOGGLE	0x6FD6	2	GPIOC状态翻转寄存器 (GPIO64 ~ GPIO87)

表6-2 GPIO数据寄存器



GPIO寄存器

名称	地址	大小 (×16bit)	寄存器说明
GPIOXINT1SEL	0x6FE0	1	外部中断源XINT1输入端口选择寄存器 (GPIO0 ~ GPIO31)
GPIOXINT2SEL	0x6FE1	1	外部中断源XINT2输入端口选择寄存器 (GPIO0 ~ GPIO31)
GPIOXNMISEL	0x6FE2	1	外部中断源XNMI输入端口选择寄存器 (GPIO0 ~ GPIO31)
GPIOXINT3SEL	0x6FE3	1	外部中断源XINT3输入端口选择寄存器 (GPIO32 ~ GPIO63)
GPIOXINT4SEL	0x6FE4	1	外部中断源XINT4输入端口选择寄存器 (GPIO32 ~ GPIO63)
GPIOXINT5SEL	0x6FE5	1	外部中断源XINT5输入端口选择寄存器 (GPIO32 ~ GPIO63)
GPIOXINT6SEL	0x6FE6	1	外部中断源XINT6输入端口选择寄存器 (GPIO32 ~ GPIO63)
GPIOXINT7SEL	0x6FE7	1	外部中断源XINT7输入端口选择寄存器 (GPIO32 ~ GPIO63)
GPIOLPMSSEL	0x6FE8	1	低功耗模式唤醒输入端口选择寄存器 (GPIO0 ~ GPIO31)

表6-3 GPIO外部中断源及低功耗模式唤醒选择寄存器



1. GPIO功能选择控制寄存器

前面已经讲到，F28335的I/O引脚都是功能复用的，既可以选择用来做通用的数字I/O口，也可以选择用来做外设的功能引脚，但是在同一时刻，只能选择一种功能来用。实际选择使用哪种功能，是通过功能选择寄存器来设置的，F28335的功能选择寄存器有6个，即GPAMUX1、GPAMUX2、GPBMUX1、GPBMUX2、GPCMUX1和GPCMUX2，每个寄存器的详细信息见表6-4 ~ 6-9。表中I/O表示既可以做输入也可以做输出，I表示仅做输入，O表示仅做输出。



GPIO寄存器

寄存器位	复位时默认功能	外设选择1	外设选择2	外设选择3
GPAMUX1	00	01	10	11
1~0	GPIO0 (I/O)	EPWM1A (O)	保留	保留
3~2	GPIO1 (I/O)	EPWM1B (O)	ECAP6 (I/O)	MFSRB (I/O)
5~4	GPIO2 (I/O)	EPWM2A (O)	保留	保留
7~6	GPIO3 (I/O)	EPWM2B (O)	ECAP5 (I/O)	MCLKRB (I/O)
9~8	GPIO4 (I/O)	EPWM3A (O)	保留	保留
11~10	GPIO5 (I/O)	EPWM3B (O)	MFSRA (I/O)	ECAP1 (I/O)
13~12	GPIO6 (I/O)	EPWM4A (O)	EPWMSYNCI (I)	EPWMSYNCO (O)
15~14	GPIO7 (I/O)	EPWM4B (O)	MCLKRA (I/O)	ECAP2 (I/O)
17~16	GPIO8 (I/O)	EPWM5A (O)	CANTXB (O)	$\overline{\text{ADCSOCA0}}$ (O)
19~18	GPIO9 (I/O)	EPWM5B (O)	SCIRXDB (I)	ECAP3 (I/O)
21~20	GPIO10 (I/O)	EPWM6A (O)	CANRXB (I)	$\overline{\text{ADCSOCB0}}$ (O)
23~22	GPIO11 (I/O)	EPWM6B (O)	SCIRXDB (I)	ECAP4 (I/O)
25~24	GPIO12 (I/O)	$\overline{\text{TZ1}}$ (I)	CANTXB (O)	MDXB (O)
27~26	GPIO13 (I/O)	$\overline{\text{TZ2}}$ (I)	CANRXB (I)	MDRB (I)
29~28	GPIO14 (I/O)	$\overline{\text{TZ3/XHOLD}}$ (I)	SCITXDB (I)	MCLKXB (I/O)
31~30	GPIO15 (I/O)	$\overline{\text{TZ4/XHOLDA}}$ (O)	SCIRXDB (O)	MFSXB (I/O)

表6-4 GPIO功能选择控制寄存器GPAMUX1



GPIO寄存器

寄存器位	复位时默认功能	外设选择1	外设选择2	外设选择3
GPAMUX2	00	01	10	11
1~0	GPIO16 (I/O)	SPISIMOA (I/O)	CANTXB (O)	$\overline{TZ5}$ (I)
3~2	GPIO17 (I/O)	SPISOMIA (I/O)	CANRXB (I)	$\overline{TZ6}$ (I)
5~4	GPIO18 (I/O)	SPICLKA (I/O)	SCITXDB (O)	CANRXA (I)
7~6	GPIO19 (I/O)	$\overline{SPISTEA}$ (I/O)	SCIRXDB (I)	CANTXA (O)
9~8	GPIO20 (I/O)	EQEP1A (I)	MDXA (O)	CANTXB (O)
11~10	GPIO21 (I/O)	EQEP1B (I)	MDRA (I)	CANRXB (I)
13~12	GPIO22 (I/O)	EQEP1S (I/O)	MCLKXA (I/O)	SCITXDB (O)
15~14	GPIO23 (I/O)	EQEP1I (I/O)	MFSXA (I/O)	SCIRXDB (I)
17~16	GPIO24 (I/O)	ECAP1 (I/O)	EQEP2A (I)	MDXB (O)
19~18	GPIO25 (I/O)	ECAP2 (I/O)	EQEP2B (I)	MDRB (I)
21~20	GPIO26 (I/O)	ECAP3 (I/O)	EQEP2S (I/O)	MCLKXB (I/O)
23~22	GPIO27 (I/O)	ECAP4 (I/O)	EQEP2I (I/O)	MFSXB (I/O)
25~24	GPIO28 (I/O)	SCIRXDA (I)	$\overline{XZCS6}$ (O)	$\overline{XZCS6}$ (O)
27~26	GPIO29 (I/O)	SCITXDA (O)	XA19 (O)	XA19 (O)
29~28	GPIO30 (I/O)	CANRXA (I)	XA18 (O)	XA18 (O)
31~30	GPIO31 (I/O)	CANTXB (O)	XA17 (O)	XA17 (O)

表6-5 GPIO功能选择控制寄存器GPAMUX2



GPIO寄存器

寄存器位	复位时默认功能	外设选择1	外设选择2	外设选择3
GPBMUX1	00	01	10	11
1~0	GPIO32 (I/O)	SDAA (I/O)	EPWMSYNCI (I)	$\overline{\text{ADCSOCA0}}$ (O)
3~2	GPIO33 (I/O)	SCLA (I/O)	EPWMSYNCO (O)	$\overline{\text{ADCSOCB0}}$ (O)
5~4	GPIO34 (I/O)	ECAP1 (I/O)	XREADY (I)	XREADY (I)
7~6	GPIO35 (I/O)	SCITXDA (O)	XR/ $\overline{\text{W}}$ (O)	XR/ $\overline{\text{W}}$ (O)
9~8	GPIO36 (I/O)	SCIRXDA (I)	$\overline{\text{XZCS0}}$ (O)	$\overline{\text{XZCS0}}$ (O)
11~10	GPIO37 (I/O)	ECAP2 (I/O)	$\overline{\text{XZCS7}}$ (O)	$\overline{\text{XZCS7}}$ (O)
13~12	GPIO38 (I/O)	保留	$\overline{\text{XWE0}}$ (O)	$\overline{\text{XWE0}}$ (O)
15~14	GPIO39 (I/O)	保留	XA16 (O)	XA16 (O)
17~16	GPIO40 (I/O)	保留	XA0/ $\overline{\text{XWE1}}$ (O)	XA0/ $\overline{\text{XWE1}}$ (O)
19~18	GPIO41 (I/O)	保留	XA1 (O)	XA1 (O)
21~20	GPIO42 (I/O)	保留	XA2 (O)	XA2 (O)
23~22	GPIO43 (I/O)	保留	XA3 (O)	XA3 (O)
25~24	GPIO44 (I/O)	保留	XA4 (O)	XA4 (O)
27~26	GPIO45 (I/O)	保留	XA5 (O)	XA5 (O)
29~28	GPIO46 (I/O)	保留	XA6 (O)	XA6 (O)
31~30	GPIO47 (I/O)	保留	XA7 (O)	XA7 (O)

表6-6 GPIO功能选择控制寄存器GPBMUX1



GPIO寄存器

寄存器位	复位时默认功能	外设选择1	外设选择2	外设选择3
GPBMUX2	00	01	10	11
1~0	GPIO48 (I/O)	ECAP5 (I/O)	XD31 (I/O)	XD31 (I/O)
3~2	GPIO49 (I/O)	ECAP6 (I/O)	XD30 (I/O)	XD30 (I/O)
5~4	GPIO50 (I/O)	EQEP1A (I)	XD29 (I/O)	XD29 (I/O)
7~6	GPIO51 (I/O)	EQEP1B (I)	XD28 (I/O)	XD28 (I/O)
9~8	GPIO52 (I/O)	EQEP1S (I/O)	XD27 (I/O)	XD27 (I/O)
11~10	GPIO53 (I/O)	EQEP1I (I/O)	XD26 (I/O)	XD26 (I/O)
13~12	GPIO54 (I/O)	SPISIMOA (I/O)	XD25 (I/O)	XD25 (I/O)
15~14	GPIO55 (I/O)	SPISOMIA (I/O)	XD24 (I/O)	XD24 (I/O)
17~16	GPIO56 (I/O)	SPICLKA (I/O)	XD23 (I/O)	XD23 (I/O)
19~18	GPIO57 (I/O)	$\overline{\text{SPISTEA}}$ (I/O)	XD22 (I/O)	XD22 (I/O)
21~20	GPIO58 (I/O)	MCLKRA (I/O)	XD21 (I/O)	XD21 (I/O)
23~22	GPIO59 (I/O)	MFSRA (I/O)	XD20 (I/O)	XD20 (I/O)
25~24	GPIO60 (I/O)	MCLKRB (I/O)	XD19 (I/O)	XD19 (I/O)
27~26	GPIO61 (I/O)	MFSRB (I/O)	XD18 (I/O)	XD18 (I/O)
29~28	GPIO62 (I/O)	SCIRXDC (I)	XD17 (I/O)	XD17 (I/O)
31~30	GPIO63 (I/O)	SCITXDC (O)	XD16 (I/O)	XD16 (I/O)

表6-7 GPIO功能选择控制寄存器GPBMUX2



GPIO寄存器

寄存器位	复位时默认功能	外设选择1	外设选择2	外设选择3
GPCMUX1	00	01	10	11
1~0	GPIO64 (I/O)	GPIO64 (I/O)	XD15 (I/O)	XD15 (I/O)
3~2	GPIO65 (I/O)	GPIO65 (I/O)	XD14 (I/O)	XD14 (I/O)
5~4	GPIO66 (I/O)	GPIO66 (I/O)	XD13 (I/O)	XD13 (I/O)
7~6	GPIO67 (I/O)	GPIO67 (I/O)	XD12 (I/O)	XD12 (I/O)
9~8	GPIO68 (I/O)	GPIO68 (I/O)	XD11 (I/O)	XD11 (I/O)
11~10	GPIO69 (I/O)	GPIO69 (I/O)	XD10 (I/O)	XD10 (I/O)
13~12	GPIO70 (I/O)	GPIO70 (I/O)	XD9 (I/O)	XD9 (I/O)
15~14	GPIO71 (I/O)	GPIO71 (I/O)	XD8 (I/O)	XD8 (I/O)
17~16	GPIO72 (I/O)	GPIO72 (I/O)	XD7 (I/O)	XD7 (I/O)
19~18	GPIO73 (I/O)	GPIO73 (I/O)	XD6 (I/O)	XD6 (I/O)
21~20	GPIO74 (I/O)	GPIO74 (I/O)	XD5 (I/O)	XD5 (I/O)
23~22	GPIO75 (I/O)	GPIO75 (I/O)	XD4 (I/O)	XD4 (I/O)
25~24	GPIO76 (I/O)	GPIO76 (I/O)	XD3 (I/O)	XD3 (I/O)
27~26	GPIO77 (I/O)	GPIO77 (I/O)	XD2 (I/O)	XD2 (I/O)
29~28	GPIO78 (I/O)	GPIO78 (I/O)	XD1 (I/O)	XD1 (I/O)
31~30	GPIO79 (I/O)	GPIO79 (I/O)	XD0 (I/O)	XD0 (I/O)

表6-8 GPIO功能选择控制寄存器GPCMUX1



GPIO寄存器

寄存器位	复位时默认功能	外设选择1	外设选择2	外设选择3
GPCMUX2	00	01	10	11
1~0	GPIO80 (I/O)	GPIO80 (I/O)	XA8 (O)	XA8 (O)
3~2	GPIO81 (I/O)	GPIO81 (I/O)	XA9 (O)	XA9 (O)
5~4	GPIO82 (I/O)	GPIO82 (I/O)	XA10 (O)	XA10 (O)
7~6	GPIO83 (I/O)	GPIO83 (I/O)	XA11 (O)	XA11 (O)
9~8	GPIO84 (I/O)	GPIO84 (I/O)	XA12 (O)	XA12 (O)
11~10	GPIO85 (I/O)	GPIO85 (I/O)	XA13 (O)	XA13 (O)
13~12	GPIO86 (I/O)	GPIO86 (I/O)	XA14 (O)	XA14 (O)
15~14	GPIO87 (I/O)	GPIO87 (I/O)	XA15 (O)	XA15 (O)
31~16	保留	保留	保留	保留

表6-9 GPIO功能选择控制寄存器GPCMUX2



GPIO寄存器

从表6-4 ~ 6-9可以看到，每个GPIO引脚都有多个功能，但是在同一时刻，通过功能选择控制寄存器只能选择一种功能。同时还可以看到，作为同一个外设功能，也可以选择配置不同的GPIO引脚，比如SCIA的发送引脚SCITXDA，既可以选择GPIO29，也可以选择GPIO35，但只能从这两个引脚里选一个作为SCITXDA，如果选了GPIO29，那么GPIO35可以用来配置其他的功能，反过来也一样。



GPIO寄存器

只要对GPxMUXn寄存器的位赋值，就可以给相应的引脚选择相应的功能，上电复位时默认选择通用I/O数字功能。比如需要配置GPIO0为通用数字I/O口，GPIO1为EPWM1B，GPIO18为SCITXDB，GPIO19为CANTXA，语句如下所示：

```
EALLOW;  
GpioCtrlRegs.GPAMUX1.bit.GPIO0=0; //GPIO0为通用数字I/O口  
GpioCtrlRegs.GPAMUX1.bit.GPIO1=1; //GPIO1为EPWM1B  
GpioCtrlRegs.GPAMUX2.bit.GPIO18=2; //GPIO18为SCITXDB  
GpioCtrlRegs.GPAMUX2.bit.GPIO19=3; //GPIO19为CANTXA  
EDIS;
```



GPIO寄存器

2.GPIO方向控制寄存器

假设已经将GPIO0设置为通用数字I/O口，那这个I/O口是作为输入引脚还是输出引脚呢？如果想作为输入引脚，则需要将GPADIR的GPIO0位设置为0，如下所示：

```
EALLOW;  
GpioCtrlRegs.GPADIR.bit.GPIO0=0; //GPIO0为输入引脚  
EDIS;
```

如果想作为输出引脚，则需要将GPADIR的GPIO0位设置为1，如下所示：

```
EALLOW;  
GpioCtrlRegs.GPADIR.bit.GPIO0=1; //GPIO0为输出引脚  
EDIS;
```



3.GPIO数据寄存器

GPIO数据寄存器有GPADAT、GPBDAT和GPCDAT，寄存器中的每一个位都对应一个I/O口。不管相应的GPIO被配置成数字I/O口，还是外设功能引脚，数据寄存器中的每一位都反映了引脚的当前状态。读GPxDAT寄存器中的数据，若结果为0，则说明引脚为低电平；若结果为1，则说明引脚为高电平。写GPxDAT寄存器，可以对相应的输出锁存器清零或置位，此时如果引脚被设置为数字I/O口，则相应的引脚就会被驱动为低电平或高电平。



GPIO寄存器

需要注意的是，当使用写GPxDAT操作来改变一个输出引脚的状态时，可能会对同一端口的其他引脚产生不确定的影响。例如，当使用“读-校正-写”模式对GPADAT的最低位GPIO0写0时，如果此时A端口中的任何一个引脚的电平发生了改变，便会出现不可预知的错误，而通过GPxDAT寄存器读引脚的当前状态则不会出现类似的错误。所以，不建议通过写GPxDAT寄存器来改变通用I/O口的电平，但可以通过读GPxDAT寄存器来获得I/O口的状态。如果要获取通用数字I/O引脚GPIO0当前的电平状态，如下所示：

```
int x;  
x=GpioDataRegs.GPADAT.bit.GPIO0; //读取GPIO0的状态。  
若x=0，则为低电平；若x=1，则为高电平。
```



4.GPIO置位寄存器

如果想要让通用数字I/O引脚输出高电平，又不影响其他引脚，则只需要对GPxSET寄存器的相应位写1，对其写0无效。比如，要让GPIO0输出高电平，需如下所示：

```
GpioDataRegs.GPADAT.bit.GPIO0=1; //GPIO0输出高电平  
前提是GPIO0配置成I/O引脚，方向输出
```



5.GPIO清零寄存器

如果想要让通用数字I/O引脚输出低电平，又不影响其他引脚，则只需要对GPxCLEAR寄存器的相应位写1，对其写0无效。比如，要让GPIO0输出低电平，需如下所示：

```
GpioDataRegs.GPACLEAR.bit.GPIO0=1; //GPIO0输出低电平，前提是GPIO0配置成I/O引脚，方向输出
```



6.GPIO电平翻转寄存器

电平翻转寄存器用来在不影响其他引脚状态的情况下将相应的引脚状态进行翻转。如果将相应的引脚配置成输出状态，那么向电平翻转寄存器中相应的位写1会将引脚的当前电平翻转，写0无效。比如，如果引脚当前为低电平，则向相应位写1后，引脚会翻转到高电平；如果引脚当前为高电平，则向相应位写1后，引脚会翻转到低电平。若要让GPIO0翻转当前的电平，需如下所示：

```
GpioDataRegs.GPATOGGLE.bit.GPIO0=1; //GPIO0电平翻转，前提是GPIO0配置成I/O引脚，方向输出
```



7.GPIO上拉控制寄存器

引脚接上拉电阻可以使该引脚从一个不确定的状态到高电平状态，也可以提高它的驱动能力。通过GPIO上拉控制寄存器GPxPUD可以为各个引脚选择是否需要接内部上拉电阻，这个功能不仅适用于通用数字I/O引脚，也适用于外设功能引脚。当DSP复位时，可以配置为ePWM引脚的GPIO0 ~ GPIO11的上拉功能都会被禁止，而其他引脚会被使能，也就是说GPIO0 ~ GPIO11内部上拉的默认状态是禁止的，而其他引脚内部上拉的默认状态就是使能的。复位结束后，引脚的上拉配置会一直保持默认状态，直到用户在软件里通过设置该寄存器来改变其状态。比如需要禁止GPIO0的内部上拉功能，禁止GPIO15的内部上拉功能，可以通过下面的语句来实现。

```
EALLOW;  
GpioCtrlRegs.GPAPUD.bit.GPIO0=0; //使能GPIO0的内部上拉  
GpioCtrlRegs.GPAPUD.bit.GPIO15=1; //使能GPIO15的内部上拉  
EDIS;
```



GPIO输入限定功能

在理论上，GPIO引脚上的电平不是高电平就是低电平，信号质量是非常好的，但是在实际应用时，通常会存在干扰，为了提高引脚信号的质量，提高系统运行的可靠性，F28335的GPIO具有输入限定的功能，从而可以方便的消除引脚输入信号中的噪声信号。只有端口GPIOA和GPIOB具有输入限定功能，GPIOC没有这个功能。输入限定的类型有三种：仅与SYSCLKOUT同步、通过采样窗限定和不同步或限定。可以通过设置寄存器GPAQSEL1、GPAQSEL2、GPBQSEL1和GPBQSEL2来选择采用哪种限定方式。



1. 仅与SYSCLKOUT同步

在这种模式下，输入信号被限定到与系统时钟SYSCLKOUT同步，由于引脚输入的信号通常是异步的，所以在于系统时钟同步过程中，会产生一个SYSCLKOUT周期的延时。如果没有对引脚的限定方式做配置，或者引脚在复位时，都会默认采用这种限定方式。



GPIO输入限定功能

2.通过采样窗设定

在这种模式下，外部引脚的输入信号首先与系统时钟SYSCLKOUT同步，然后对输入信号进行采样，只有当输入信号在一个采样窗内保持不变，才能确定引脚电平，最后输入给DSP，从而实现了滤出噪声信号的目的，其原理如图6-1所示。

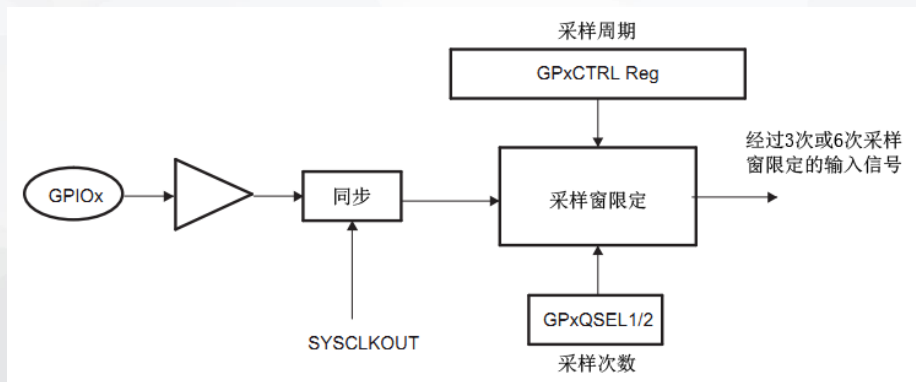


图6-1 采样窗限定输入信号的原理



GPIO输入限定功能

从6-1可以看到，通过采样窗的方式来对输入信号限定的话，有两个参数需要设置：采样周期和采样次数，这两个参数决定了采样窗的长度。先来看采样周期，其意思就是进行一次采样需要花多长时间，它是以系统时钟SYSCLKOUT为单位的，也就是说，采样周期是设定了进行一次采样需要多少个SYSCLKOUT，它可以通过寄存器GPACTRL和GPBCTRL进行设置。以GPACTRL为例，它具有4个位域，QUALPRD0、QUALPRD1、QUALPRD2和QUALPRD3。每个QUALPRDn的取值范围是0~510。每8个GPIO引脚作为1组，例如GPIO0~GPIO7是由GPACTRL[QUALPRD0]来设置，GPIO8~GPIO15是由GPACTRL[QUALPRD1]来设置。采样周期和GPxCTRL[QUALPRDn]位之间的关系如表6-10所示。表中 $T_{\text{SYSCLKOUT}}$ 表示系统时钟SYSCLKOUT的周期。



GPIO输入限定功能

寄存器的设置	采样周期
GPxCTRL[QUALPRDn]=0	$1 \times T_{\text{SYSCLKOUT}}$
GPxCTRL[QUALPRDn]≠0	$2 \times \text{GPxCTRL}[\text{QUALPRDn}] \times T_{\text{SYSCLKOUT}}$

表6-10 采样周期与GPxCTRL[QUALPRDn]的关系

采样周期设置好后，还需要通过寄存器GPxQSELn来设置采样次数，只有两个选项可以选择，采样3次或者采样6次。当选择采样3次，则采样窗口的长度是2个采样周期；当选择采样6次，则采样窗口的长度5个采样周期。图6-2是采样次数是6次时，引脚输入信号经采样窗限定的原理。



GPIO输入限定功能

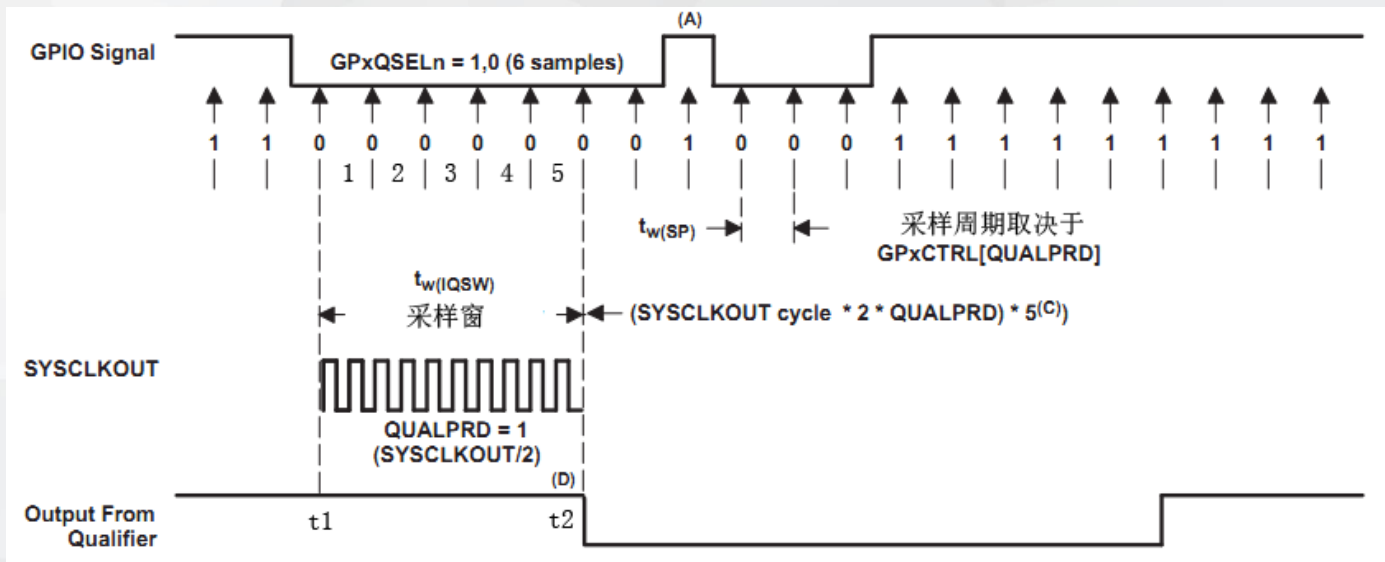


表6-10 采样周期与GPxCTRL[QUALPRDn]的关系



GPIO输入限定功能

从图6-2可以看到，在 t_1 时刻，DSP开始采集到低电压，但此时DSP还不能马上认定现在输入的信号就是低电平信号，必须连续的对其进行6次采样，如果这6次采样结果均为0，DSP才会确定输入信号已经变为低电平，中间只要有1次采样结果不是0，就必须重新开始采样限定。图6-2中，到 t_2 时刻，DSP已经连续6次采集到低电压，所以就认定输入信号为低电平，很容易看到， t_1 与 t_2 间有6次采样，历经5个采样周期。通过采样窗限定输入信号，就可以把后面的这个只有1个采样周期宽度的高电平干扰过滤掉，从而提高了信号的质量。



6.4GPIO配置步骤

下面总结下，如何正确使用一个GPIO引脚，可以参照如下顺序进行配置：

- 1.选择GPIO工作模式：首先需要搞清每个GPIO引脚所具有的功能，并通过GPxMUXn寄存器选择让其工作在通用数字I/O模式，还是工作在某种外设功能模式。默认情况下，DSP上电时，GPIO被配置成通用数字I/O模式，且为输入引脚。

- 2.使能或禁止内部上拉电阻。具有ePWM输出功能的GPIO0 ~ GPIO11的内部上拉功能默认是禁止的，其他引脚的内部上拉功能默认是使能的。



6.4GPIO配置步骤

3.如果引脚是作输入用的，那需要为引脚配置输入限定模式。默认情况下，所有输入引脚的限定模式是与SYSCLKOUT同步，当然也可以通过相关的寄存器选择其他的输入限定模式，比如采样窗限定。

4.如果引脚被配置为通用数字I/O口，那么还需要通过GPxDIR寄存器来设定该引脚是做输入还是做输出，也就是要选择引脚方向。

5.如果引脚被配置为通用数字I/O口，并且方向是输出的话，可以通过GPxSET或者GPxCLEAR寄存器来设定引脚的初始电平。

通过上面五个步骤，基本完成了GPIO引脚的初始化工作。