Application Note

SPC11x8/SPD11x8 SIO_QEP 使用指南

Revision 2.1.1 – January 2020

目录

表格列表

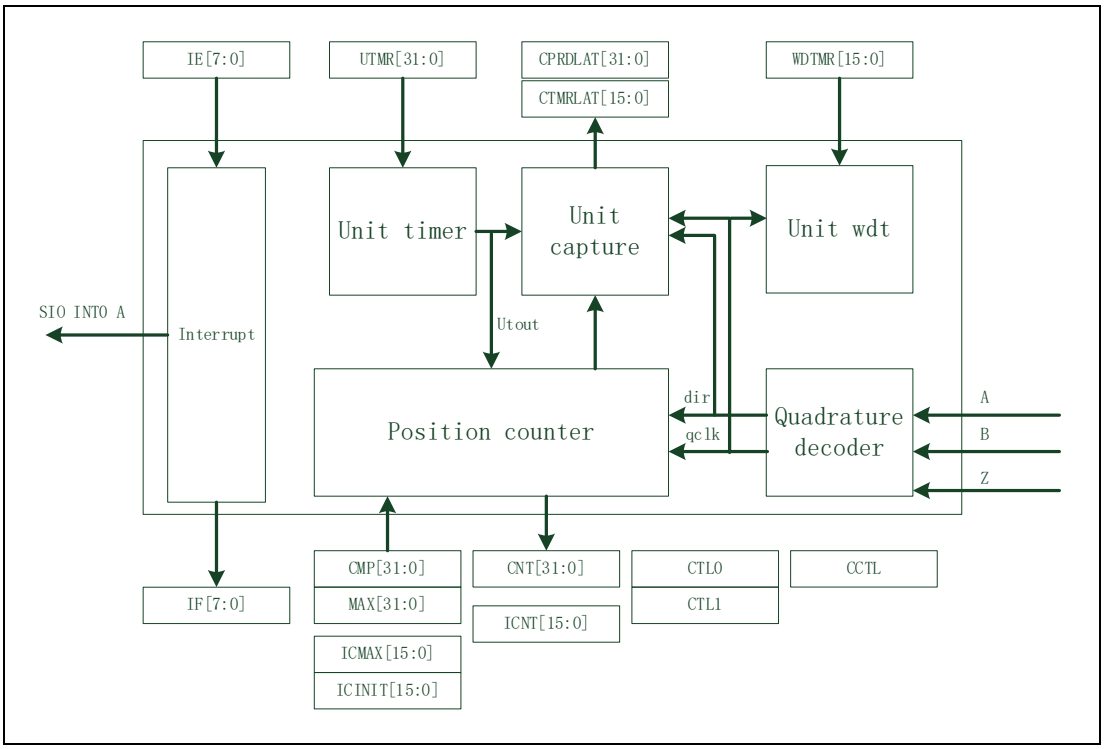
图片列表

1 基于 SIO 的正交编码器（QEP）模块

1.1 系统概述

如图 1-1 所示，正交编码器（QEP）模块由正交解码，位置计数器，计时器，捕获模块和看门狗模块组成。

图 1-1: 基于 SIO 的 QEP 系统架构



A/B/Z 为编码器输入信号。

编码器输入正交信号 A 和 B，正交解码模块将其解码为方向 QDIR 和 QCLK 作为位置计数器的输入。位置计数器从 0 开始计数，计数方向根据 QDIR 改变：当 QDIR 为 1，则向上计数，计数到 MAX 值，则重新从 0 开始计数，并且置位向上溢出(PCO)标志位；当 QDIR 为 0，则向下计数，计数到 0，则从 MAX 值重新计数，并且置位向下溢出(PCU)标志位。计数值等于 CMP 时，则置位位置匹配(PCM)标志位。CNT 寄存器为计数器计数值。

计时器模块被用于测频法测量速度，设置 UTMR 寄存器确定计时周期。QEP 模块启动则开始计时。当计时器超时，触发捕获模块锁存当前计数值，并将该值与计时器开始计时时刻的计数值的差锁存进 CPRDLAT，然后计时器重新计时，重复该过程。

当使用测周法测量速度，需要设置 QCLK 的预分频系数（CCTL.QPS），产生 QEVENT。QEVENT 触发捕获模块，并且将 QEVENT 之间的计时器值的差锁存进 CTMRLAT。

看门狗模块根据周期 WDTMR 计时。在有效时间内没有检测到 QCLK，则会置位看门狗超时(WTO)标志位。

ICNT/ICMAX/ICINIT 为位置信号 Z 相关的寄存器。当 QDIR 为 1，向上计数，检测到位置信号 Z 上升沿，则复位 ICNT 为 0；当 QDIR 为 0，向下计数，检测到位置信号 Z 下降沿，则复位 ICNT 为

ICMAX；ICINIT 用于设置 ICNT 初始值。CTL0.EN 为 0 时候，CTL0.ICNTINIT 置位，然后使能 CTL0.EN 运行 SIO_QEP 模块，ICNT 从 ICINIT 加载初始值。

注意：

QEP 中的标志位都必须使能相应的中断使能位，否则标志位始终为 0。

注意：

不同芯片中，SIO 模块数量不同，本文中统一使用 SIO_QEP，实际代码中为 SIO_QEPx （x 为数字）

图 1-2: 基于 SIO 的 QEP 时序，预分频为 4

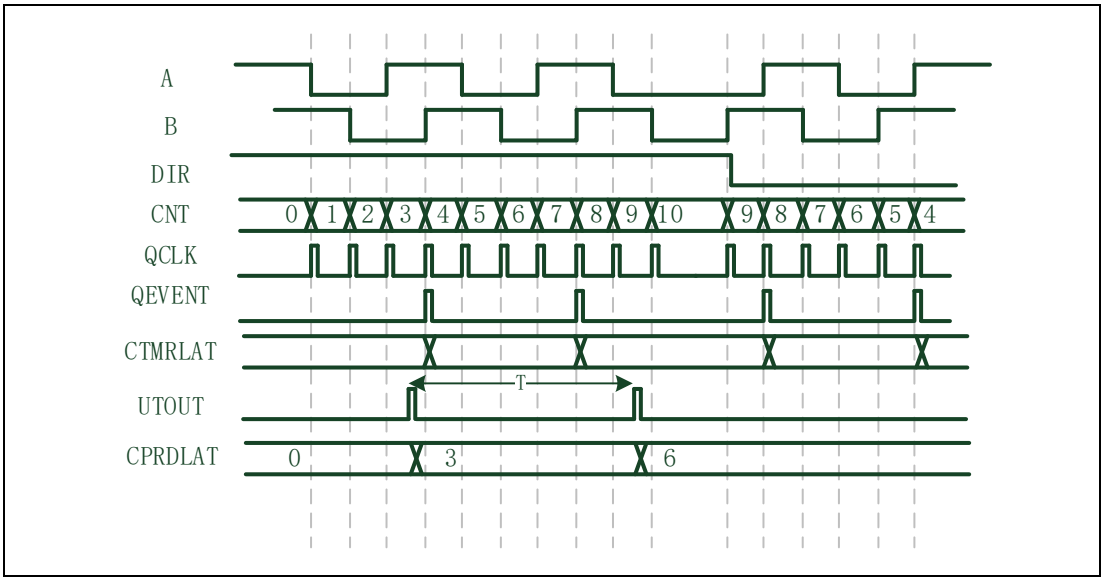
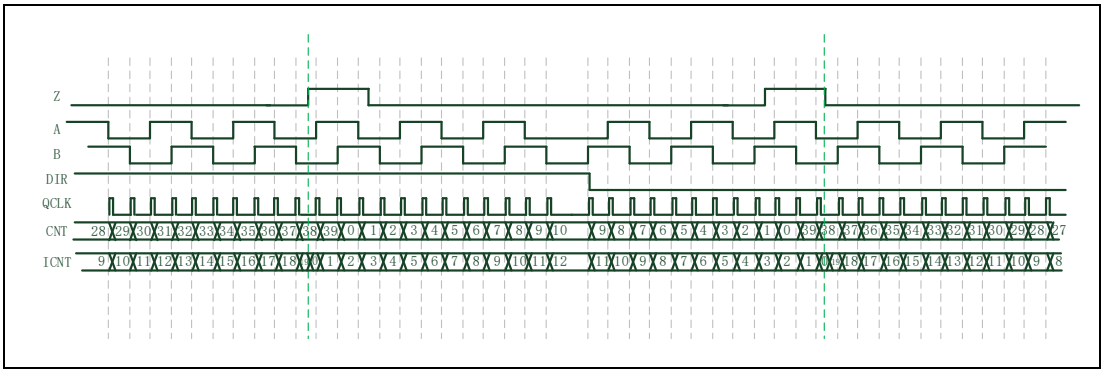


图 1-3: CNT 和 ICNT 计数 （MAX 为 39， ICMAX 为 19）



1.2 特性

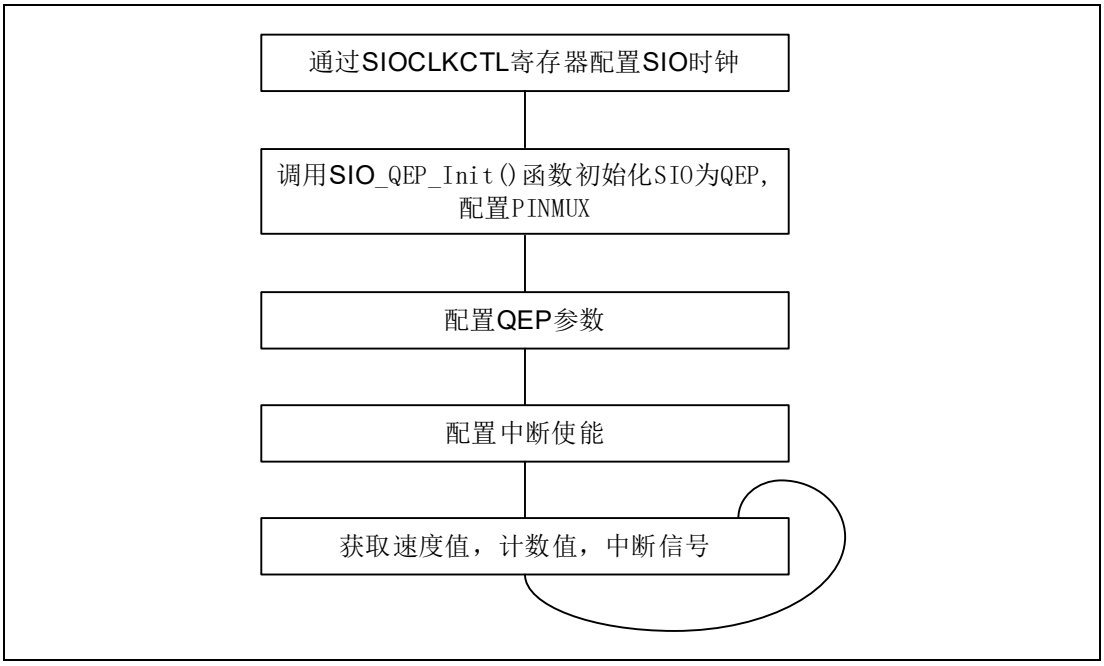
基于 SIO 所实现的 QEP 系统具备如下特性：

- 支持正交编码器 AB 相输入；
- 支持位置信号 Z 输入；
- 支持测速功能（测周法或者测频法）；
- 支持看门狗检测速度太慢；
- 支持相位错误检测；
- 支持系统中断。

2 操作方式

图 2-1 给出了基于 SIO 的 QEP 系统在使用时的具体操作流程。Spintrol 提供了相应的软件库来简化该系统的使用。

图 2-1: 基于 SIO 的 QEP 系统操作流程



2.1 配置 SIO 时钟

用户可以通过 SIOCLKCTL 寄存器来配置 SIO 时钟，包括时钟的使能和分频比。具体可参见《SPC11X8/SPD11X8 Technical Reference Manual》的第 3 章。当 SIO 被配置用作 QEP 时，所允许的 SIO 模块时钟最高频率可达 100MHz（对于部分型号芯片，最高频率会达不到 100MHz，具体参考 TRM）。

2.2 初始化 SIO 为 QEP 协议引擎，配置 PINMUX

SPC11X8/SPD11X8 SDK 中提供了 SIO_QEP_Init()函数，下载固件到 SIO 模块，将其初始化成 QEP，同时配置 PINMUX，将引脚切换至 SIO 通道。用户只需要在代码中直接调用该函数即可。用户想使用不同的引脚配置，请参考《SIO 配置工具使用指南.docx》重新生成 SIO 库文件。

2.3 QEP 配置

首先根据编码器的线数设置 QEP 最大位置寄存器(MAXH,MAXL)以及位置比较寄存器(CMPH,CMPL)。再根据所测量的速度范围合理使用测周法和测频法，设置 QCLK 预分频 CCTL.QPS，计时器周期(UTMRH,UTMRL)和看门狗计时周期(WDTMR)。
根据需要，配置中断使能（IE）。
最后，使能 QEP(CTL0.EN)。

2.4 QEP 获取位置计数，方向

QEP 使能后，调用 SIO_QEP_GetCounterValue()和 SIO_QEP_GetDirection()来获取当前计数值和计数方向。

2.5 QEP 获取位置速度

若采用测周法测量，可以使用 SIO_QEP_GetSpeedBySurveyCycle()获取速度，其原理如下：

V = X / dt

V : speed

X : distance between QEVENT, equal to QCLK prescaler

dt: capture timer latch value

若采用测频法测量，可以使用 SIO_QEP_GetSpeedBySurveyFrequency()获取速度，其原理如下：

V = dx / T

V : speed

X : distance during unit timer timeout

dt: unit timer period value

2.6 QEP 中断处理

当 QEP 使能中断，并且相应中断时间发生，则会触发中断，用户可以使用中断回调函数处理相应中断事件。

3 API 函数

表 3-1: API 函数列表

函数名称	说明
void SIO_PinInit(void)	SIO PLD 初始化。
void SIO_Init(void)	SIO 初始化，调用 PLD 初始化，引脚初始化。
void SIO_QEP_Init(void)	SIO 初始化为 QEP 功能。
SIO_QEP_Run()	QEP 启动。
SIO_QEP_Stop()	QEP 停止。
SIO_QEP_EnableInitICounter()	使能位置信号 Z 计数器初始化。
SIO_QEP_DisableInitICounter()	禁用位置信号 Z 计数器初始化。
SIO_QEP_GetDirection()	获取方向，返回 1bit 数据。
SIO_QEP_GetFlag()	获取标志位，返回 8bits 数据。
SIO_QEP_IsFlagSet(eFlagSel)	判断标志是否置位，返回 1bit 数据。 eFlagSel：SIO_QEP_FlagTypeEnum 枚举类型，可以多个枚举值或运算作为参数。
SIO_QEP_ClrFlag()	清除所有标志位。
SIO_QEP_SetComparePosition(u32Value)	设置位置比较寄存器。 u32Value：32bits 整数。
SIO_QEP_GetComparePosition()	获取位置比较寄存器值，返回 32bits 数据。
SIO_QEP_SetMaxPosition(u32Value)	设置最大位置寄存器。 u32Value：32bits 整数。
SIO_QEP_GetMaxPosition()	获取最大位置寄存器值，返回 32bits 数据。
SIO_QEP_SetICMaxPosition(u16Value)	设置位置信号 Z 计数器最大位置。 u16Value：16bits 整数。
SIO_QEP_GetICMaxPosition()	获取位置信号 Z 计数器最大位置，返回 16bits 数据。
SIO_QEP_SetICInitPosition(u16Value)	设置位置信号 Z 计数器初始位置。 u16Value：16bits 整数。
SIO_QEP_GetICInitPosition()	获取位置信号 Z 计数器初始位置，返回 16bits 数据。
SIO_QEP_GetCounterValue()	获取 QCLK 计数值，返回 32bits 数据。
SIO_QEP_GetICounterValue()	获取位置信号 Z 计数器计数值，返回 16bits 数据。
SIO_QEP_GetUTimerPeriod()	获取计时器计时周期，返回 32bits 数据。
SIO_QEP_SetUWDTPeriod(u16Value)	设置看门狗计时周期。 u16Value：16bits 整数。
SIO_QEP_GetUWDTPeriod()	获取看门狗计时周期，返回 16bits 数据。
SIO_QEP_GetCaptureTimerLatch()	获取捕获的时间值，返回 16bits 数据。

	QEVENT 触发捕获。
SIO_QEP_SetQClockPrescaler(u8Value)	设置 QCLK 预分频系数，生成 QEVENT。 u8Value：8bits 整数。
SIO_QEP_GetQClockPrescaler()	获取 QCLK 预分频系数，返回 8bits 数据。
SIO_QEP_GetCapturePeriodLatch()	获取捕获的计数值，返回 32bits 数据。计时器超时触发捕获。
void SIO_QEP_EnableInt(uint32_t eIntSel) ;	使能中断。 eIntSel：SIO_QEP_IntTypeEnum 枚举类型，可以多个枚举值或运算作为参数。
void SIO_QEP_DisableInt(uint32_t eIntSel) ;	关闭中断。 eIntSel：SIO_QEP_IntTypeEnum 枚举类型，可以多个枚举值或运算作为参数。
void SIO_QEP_SetUTimerPeriod(uint32_t u32Value)	设置计时器计时周期。 u32Value：32bits 整数。
double SIO_QEP_GetSpeedBySurveyCycle(void)	根据测周法获得速度值，返回 double。
double SIO_QEP_GetSpeedBySurveyFrequency(void)	根据测频法获得速度值，返回 double。
void SIO_QEP_IRQHandler(void)	QEP 中断服务程序。
void SIO_QEP_PositionCounterUnderflowCallback(void)	位置计数器向上溢出中断服务程序回调函数。
void SIO_QEP_WatchdogTimeoutCallback(void)	看门狗中断服务程序回调函数。
void SIO_QEP_QEVENTCallback(void)	QEVENT 中断服务程序回调函数。
void SIO_QEP_UtimerTimeoutCallback(void)	计时器超时中断服务程序回调函数。
void SIO_QEP_PositionMatchCallback(void)	位置比较中断服务程序回调函数。
void SIO_QEP_DirectionChangedCallback(void)	方向改变中断服务程序回调函数。
void SIO_QEP_QuadraturePhaseErrorCallback(void)	正交编码信号相位错误中断服务程序回调函数。
void SIO_QEP_PositionCounterOverflowCallback(void)	位置计数器向下溢出中断服务程序回调函数。
void SIO_QEP_IndexEventCallback(void)	位置信号 Z 事件中断服务程序回调函数。

4 代码示例

以 SIO_QEP0 为例子。

4.1 初始化 SIO 并配置 QEP，获取速度值

示例代码 4-1：初始化 SIO_QEP，获取速度值

```
uint32_t gu32EncodeLine ;
uint32_t gu32IntMask ;
uint32_t gu32IntFlagMask ;

int main(void)
{
    double f64Clock;

    uint32_t u32MaxPositionValue  ;
    uint32_t u32CmpPositionValue  ;
    uint32_t u32UTimerPeriodValue ;
    uint32_t u32PrescalerValue    ;
    uint32_t u32UWDTPeriodValue   ;

    /* Config Flash Timing for 200 MHz */
    FLASH_WALLOW();
    FLASH_SetTiming(200000000);
    FLASH_WDIS();

    /* Clock Init */
    CLOCK_InitWithRCO(CLOCK_HCLK_200MHZ);

    /* Delay Init */
    Delay_Init();

    /* UART Init */
    PINMUX->GPIO34.bit.MUXSEL = GPIO34_BIT_MUXSEL_UART_TXD;
    PINMUX->GPIO35.bit.MUXSEL = GPIO35_BIT_MUXSEL_UART_RXD;

    UART_Init(UART, 256000);

    /* Configure SIO Clock */
    CLOCK_SetModuleDiv(SIO0_MODULE, 2);
    CLOCK_EnableModule(SIO0_MODULE);

    f64Clock = CLOCK_GetModuleClock(SIO0_MODULE);
```

```
    gu32EncodeLine       = 360          ;
    u32UTimerPeriodValue  = 200000 * 1000 ; // 200000 us
    u32UWDTPeriodValue    = 300 * 1000    ; // 300 us

    printf("Encoder line is %d\n", gu32EncodeLine      );
    printf("Ut is %d\n"          , u32UTimerPeriodValue );
    printf("Wdt is %d\n"         , u32UWDTPeriodValue   );

    u32MaxPositionValue  = gu32EncodeLine * 4 - 1     ;
    u32CmpPositionValue  = gu32EncodeLine * 4 / 2 - 1 ;
    u32UTimerPeriodValue = u32UTimerPeriodValue / ( 1.0e9 /
f64Clock ) - 1 ;
    u32PrescalerValue    = 4;
    u32UWDTPeriodValue   = u32UWDTPeriodValue / ( 1.0e9 / f64Clock )
- 1 ;

    /* SIO_QEP Init */
    SIO_QEP0_Init();

    SIO_QEP0_SetMaxPosition( u32MaxPositionValue ) ;
    printf("SIO_QEP0_SetMaxPosition: %d\n",
SIO_QEP0_GetMaxPosition());

    // Set compare position, gu32EncodeLine * 4 / 2
    SIO_QEP0_SetComparePosition( u32CmpPositionValue ) ;
    printf("SIO_QEP0_SetComparePosition: %d\n",
SIO_QEP0_GetComparePosition());

    SIO_QEP0_SetUTimerPeriod( u32UTimerPeriodValue ) ;
    printf("SIO_QEP0_GetUTimerPeriod: %d\n",
SIO_QEP0_GetUTimerPeriod());

    SIO_QEP0_SetQClockPrescaler( u32PrescalerValue );
    printf("SIO_QEP0_GetQClockPrescaler: %d\n",
SIO_QEP0_GetQClockPrescaler());

    SIO_QEP0_SetUWDTPeriod( u32UWDTPeriodValue ) ;
    printf("SIO_QEP0_GetUWDTPeriod: %d\n",
SIO_QEP0_GetUWDTPeriod());


    gu32IntMask = INT_MASK_WDTOUT | INT_MASK_DIR_CHANGED |
INT_MASK_PHASE_ERROR ;
```

```
    gu32IntFlagMask = FLAG_MASK_WDTOUT | FLAG_MASK_DIR_CHANGED |
FLAG_MASK_PHASE_ERROR ;
    SIO_QEP0_EnableInt( gu32IntMask );

    NVIC_EnableIRQ(SIO0A_IRQn);

    SIO_QEP0_Run();

    while(1)
    {
        printf("%s, %d, %f, %f\n",
               SIO_QEP0_GetDirection()?"Forward":"Backward",
                                SIO_QEP0_GetCounterValue(),
                                SIO_QEP0_GetSpeedBySurveyCycle(),
                                SIO_QEP0_GetSpeedBySurveyFrequency() );

        printf("==================\n\n");
    }
}


void SIO_QEP0_WatchdogTimeoutCallback(void)
{
    printf("===>>> %s\n",__FUNCTION__);
}

void SIO_QEP0_DirectionChangedCallback(void)
{
    printf("===>>> %s\n",__FUNCTION__);
}

void SIO_QEP0_QuadraturePhaseErrorCallback(void)
{
    printf("===>>> %s\n",__FUNCTION__);
}
```

5 寄存器

5.1 SIO_QEP 寄存器表

表 5-1: SIO_QEP 模块基地址

外设模块	基地址
SIO_QEP	0x4000B000

表 5-2: SIO_QEP 模块寄存器表

寄存器	对应 SIO 寄存器	偏移地址	说明	默认值
CTL0	SIOM[0]	0x00	Control Register 0	0x00000000
IE	SIOM[0]	0x00	Interrupt Enable Register	0x00000000
CMPL	SIOM[1]	0x04	QEP Position-compare Low	0x00000000
CMPH	SIOM[2]	0x08	QEP Position-compare High	0x00000000
MAXL	SIOM[3]	0x0C	QEP Maximum Position Count Low	0x00000000
MAXH	SIOM[4]	0x10	QEP Maximum Position Count High	0x00000000
CNTL	SIOM[5]	0x14	QEP Position Counter Low	0x00000000
CNTH	SIOM[6]	0x18	QEP Position Counter High	0x00000000
UTMRL	SIOM[7]	0x1C	QEP Unit Timer Period Low	0x00000000
UTMRH	SIOM[8]	0x20	QEP Unit Timer Period High	0x00000000
WDTMR	SIOM[9]	0x24	QEP Watchdog Timer Period	0x00000000
CTMRLAT	SIOM[10]	0x28	QEP Capture Timer Latch	0x00000000
CCTL	SIOM[11]	0x2C	QEP Capture Control Register	0x00000000
CPRDLATL	SIOM[12]	0x30	QEP Capture Period Latch Low	0x00000000
CPRDLATH	SIOM[13]	0x34	QEP Capture Period Latch High.	0x00000000
IF	SIOM[14]	0x38	QEP Interrupt Flag Register	0x00000000
IC	SIOM[15]	0x3C	QEP Interrupt Flag Clear Register	0x00000000
ICMAX	SIOM[27]	0x6C	Index counter maximum value	0x00000000
ICNT	SIOM[28]	0x70	Index counter	0x00000000
ICINIT	SIOM[29]	0x74	Index counter init value	0x00000000
CTL1	SIOM[31]	0x7C	Control Register 1	0x00000000

5.2 SIO_QEP 寄存器

表 5-3 到表 5-44 列举了 SIO_QEP 寄存器的所有细节。

表 5-3: Control Register 0 (CTL0) Layout

CTL0 (Control Register 0) Offset: 0x0 Default: 0x00000000 Access: SIO_QEP -> CTL0.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
RESERVED_15_9							ICNTINIT
7	6	5	4	3	2	1	0
DIR	RESERVED_6_1						EN

表 5-4: Control Register 0 (CTL0) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:9	RESERVED_15_9	RW	0x0	Reserved.
8	ICNTINIT	RW	0x0	Index counter init enable 0: Disable index counter init 1: Enable index counter init
7	DIR	RO	0x0	Quadrature direction flag 0: Counter-clockwise rotation (or reverse movement) 1: Clockwise rotation (or forward movement)
6:1	RESERVED_6_1	RW	0x0	Reserved.
0	EN	RW	0x0	Quadrature position counter enable/software reset 0: QEP position counter is disabled. 1: QEP position counter is enabled.

表 5-5: **Interrupt Enable Register (IE) Layout**

IE (Interrupt Enable Register) Offset: 0x0 Default: 0x00000000 Access: SIO_QEP -> IE.all							
31	30	29	28	27	26	25	24
RESERVED_31_10							
23	22	21	20	19	18	17	16
RESERVED_31_10							
15	14	13	12	11	10	9	8
RESERVED_31_10						IEE	RESERVED_8
7	6	5	4	3	2	1	0
RESERVED_7	QPE	QDC	PCM	UTO	QE	WTO	RESERVED_0

表 5-6: **Interrupt Enable Register (IE) Field Description**

Bits	Field Name	Type	Reset	Description
31:10	RESERVED_31_10	RO	0x0	Reserved.
9	IEE	RW	0x0	Index event interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled
8	RESERVED_8	RW	0x0	Reserved.
7	RESERVED_7	RW	0x0	Reserved.
6	QPE	RO	0x0	Quadrature phase error interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled
5	QDC	RW	0x0	Quadrature direction change interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled
4	PCM	RW	0x0	Position compare match interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled
3	UTO	RW	0x0	Unit time out interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled
2	QE	RW	0x0	Quadrature event interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled
1	WTO	RW	0x0	Watchdog time out interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled
0	RESERVED_0	RW	0x0	Reserved.

表 5-7: QEP Position Compare Low Register (CMPL) Layout

CMPL (QEP Position-compare Low) Offset: 0x4 Default: 0x00000000 Access: SIO_QEP -> CMPL.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-8: QEP Position Compare Low Register (CMPL) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-9: QEP Position Compare High Register (CMPH) Layout

CMPH (QEP Position-compare High) Offset: 0x8 Default: 0x00000000 Access: SIO_QEP -> CMPH.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-10: QEP Position Compare High Register (CMPH) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-11: QEP Maximum Position Count Low Register (MAXL) Layout

MAXL (QEP Maximum Position Count Low) Offset: 0xC Default: 0x00000000 Access: SIO_QEP -> MAXL.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-12: QEP Maximum Position Count Low Register (MAXL) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-13: QEP Maximum Position Count High Register (MAXH) Layout

MAXH (QEP Maximum Position Count High) Offset: 0x10 Default: 0x00000000 Access: SIO_QEP -> MAXH.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-14: QEP Maximum Position Count High Register (MAXH) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-15: QEP Position Counter Low Register (CNTL) Layout

CNTL (QEP Position Counter Low) Offset: 0x14 Default: 0x00000000 Access: SIO_QEP -> CNTL.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-16: QEP Position Counter Low Register (CNTL) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-17: QEP Position Counter High Register (CNTH) Layout

CNTH (QEP Position Counter High) Offset: 0x18 Default: 0x00000000 Access: SIO_QEP -> CNTH.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-18: QEP Position Counter High Register (CNTH) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-19: QEP Unit Timer Period Low (UTMRL) Layout

UTMRL (QEP Unit Timer Period Low) Offset: 0x1C Default: 0x00000000 Access: SIO_QEP -> UTMRL.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-20: QEP Unit Timer Period Low (UTMRL) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-21: QEP Unit Timer Period High (UTMRH) Layout

UTMRH (QEP Unit Timer Period High) Offset: 0x20 Default: 0x00000000 Access: SIO_QEP -> UTMRH.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-22: QEP Unit Timer Period High (UTMRH) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-23: QEP Watchdog Timer Period (WDTMR) Layout

WDTMR (QEP Watchdog Timer Period) Offset: 0x24 Default: 0x00000000 Access: SIO_QEP -> WDTMR.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-24: QEP Watchdog Timer Period (WDTMR) Filed Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-25: QEP Capture Timer Latch (CTMRLAT) Layout

CTMRLAT (QEP Capture Timer Latch) Offset: 0x28 Default: 0x00000000 Access: SIO_QEP -> CTMRLAT.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-26: QEP Capture Timer Latch (CTMRLAT) Filed Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-27: QEP Capture Control (CCTL) Layout

CCTL (QEP Capture Control Register) Offset: 0x2C Default: 0x00000000 Access: SIO_QEP -> CCTL.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
RESERVED_15_8							
7	6	5	4	3	2	1	0
QPS							

表 5-28: QEP Capture Control (CCTL) Filed Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:8	RESERVED_15_8	RW	0x0	Reserved.
7:0	QPS	RW	0x0	qevent prescaler

表 5-29: QEP Capture Period Latch Low Register (CPRDLATL) Layout

CPRDLATL (QEP Capture Period Latch Low) Offset: 0x30 Default: 0x00000000 Access: SIO_QEP -> CPRDLATL.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-30: QEP Capture Period Latch Low Register (CPRDLATL) Filed Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-31: QEP Capture Period Latch High Register (CPRDLATH) Layout

CPRDLATH (QEP Capture Period Latch High) Offset: 0x34 Default: 0x00000000 Access: SIO_QEP -> CPRDLATH.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-32: QEP Capture Period Latch High Register (CPRDLATH) Filed Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-33: QEP Interrupt Flag Register (IF) Layout

IF (QEP Interrupt Flag Register) Offset: 0x38 Default: 0x00000000 Access: SIO_QEP -> IF.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
RESERVED_15_9							IEE
7	6	5	4	3	2	1	0
PCO	QPE	QDC	PCM	UTO	QE	WTO	PCU

表 5-34: QEP Interrupt Flag Register (IF) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:9	RESERVED_15_9	RW	0x0	Reserved.
8	IEE	RO	0x0	Index event interrupt flag 0: No interrupt generated 1: Interrupt was generated
7	PCO	RO	0x0	Position counter overflow interrupt flag 0: No interrupt generated 1: Interrupt was generated
6	QPE	RO	0x0	Quadrature phase error interrupt flag 0: No interrupt generated 1: Interrupt was generated
5	QDC	RO	0x0	Quadrature direction change interrupt flag 0: No interrupt generated 1: Interrupt was generated
4	PCM	RO	0x0	Position compare match interrupt flag 0: No interrupt generated 1: Interrupt was generated
3	UTO	RO	0x0	Unit time out interrupt flag 0: No interrupt generated 1: Interrupt was generated
2	QE	RO	0x0	Quadrature event interrupt flag 0: No interrupt generated 1: Interrupt was generated
1	WTO	RO	0x0	Watchdog time out interrupt flag 0: No interrupt generated 1: Interrupt was generated
0	PCU	RO	0x0	Position counter underflow interrupt flag 0: No interrupt generated 1: Interrupt was generated

表 5-35: QEP Interrupt Flag Clear Register (IC) Layout

IC (QEP Interrupt Flag Clear Register) Offset: 0x3C Default: 0x00000001 Access: SIO_QEP -> IC.all							
31	30	29	28	27	26	25	24
RESERVED_31_4							
23	22	21	20	19	18	17	16
RESERVED_31_4							
15	14	13	12	11	10	9	8
RESERVED_31_4							
7	6	5	4	3	2	1	0
RESERVED_31_4				ICLR	RESERVED_2_0		

表 5-36: QEP Interrupt Flag Clear Register (IC) Field Description

Bits	Field Name	Type	Reset	Description
31:4	RESERVED_31_4	RO	0x0	Reserved.
3	ICLR	RW	0x0	QEP Interrupt Clear Register 0: No effect 1: Clear all interrupt flag
2:0	RESERVED_2_0	RW	0x0	Reserved.

表 5-37: Index Counter Maximum Value (ICMAX) Layout

ICMAX (Index counter maximum value) Offset: 0x6C Default: 0x00000000 Access: SIO_QEP -> ICMAX.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-38: Index Counter Maximum Value (ICMAX) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-39: Index Counter (ICNT) Layout

ICNT (Index counter) Offset: 0x70 Default: 0x00000000 Access: SIO_QEP -> ICNT.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-40: Index Counter (ICNT) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-41: Index Counter Init Value (ICINIT) Layout

ICINIT (Index counter init value) Offset: 0x74 Default: 0x00000000 Access: SIO_QEP -> ICINIT.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
VAL							
7	6	5	4	3	2	1	0
VAL							

表 5-42: Index Counter Init Value (ICINIT) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:0	VAL	RW	0x0	

表 5-43: Control Register 1 (CTL1) Layout

CTL1 (Control Register 1) Offset: 0x7C Default: 0x00000101 Access: SIO_QEP -> CTL1.all							
31	30	29	28	27	26	25	24
RESERVED_31_16							
23	22	21	20	19	18	17	16
RESERVED_31_16							
15	14	13	12	11	10	9	8
RESERVED_15_9							RESERVED_8
7	6	5	4	3	2	1	0
PCOIE	PCUIE	RESERVED_5_1					RESERVED_0

表 5-44: Control Register 1 (CTL1) Field Description

Bits	Field Name	Type	Reset	Description
31:16	RESERVED_31_16	RO	0x0	Reserved.
15:9	RESERVED_15_9	RW	0x0	Reserved.
8	RESERVED_8	RW	0x1	Do not modify this value.
7	PCOIE	RW	0x0	Position counter overflow interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled
6	PCUIE	RW	0x0	Position counter underflow interrupt enable 0: Interrupt is disabled 1: Interrupt is enabled
5:1	RESERVED_5_1	RW	0x0	Reserved.
0	RESERVED_0	RW	0x1	Do not modify this value.

6 修订记录

表 6-1: 文档修订记录

日期	版本	修改内容
2019-08-21	1	初始版本
2020-01-17	2	增加 Z 信号
2020-04-28	2	添加寄存器对应 SIO 寄存器位置，修复文档中描述错误。
2020-04-30	2.1	添加中断标志清除寄存器 IC.ICLR，删除 CTL1.ICLR。
2021-01-04	2.1.1	修正 SIO 模块最高始终速率为 100MHz