

GWT429PC104B 用户手册

一. 产品介绍

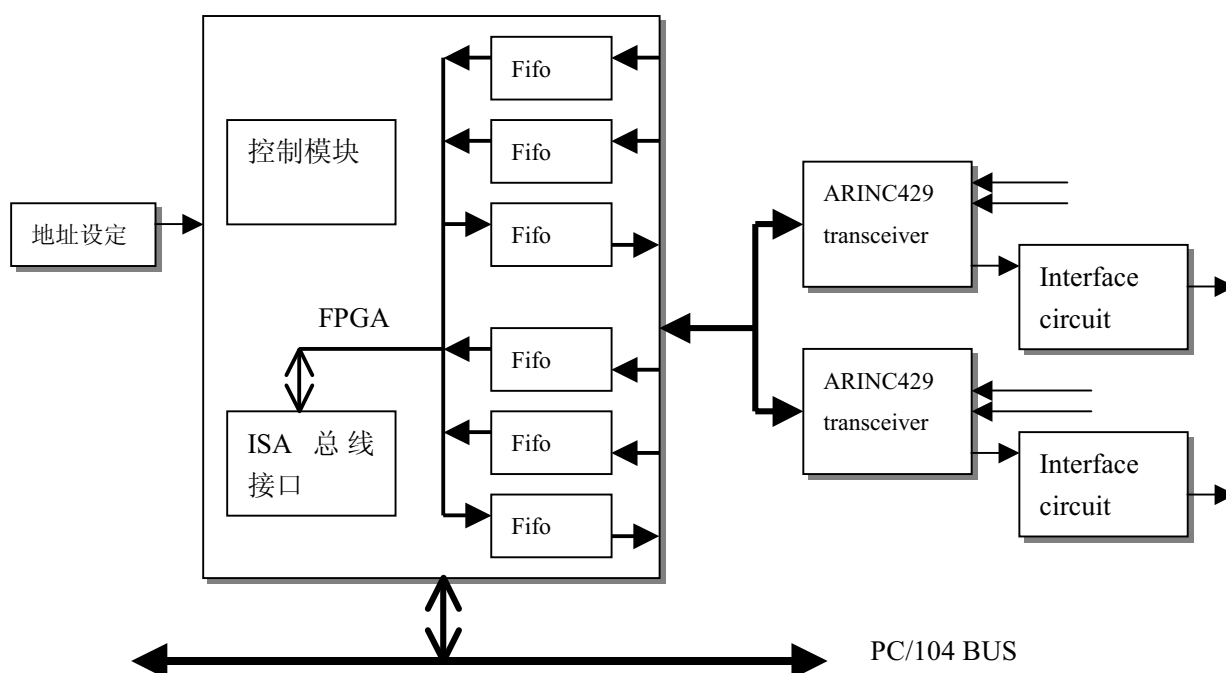
1. 主要性能

- 四路接收通道
- 二路发送通道
- 波特率及数据格式软件可编程
- 3K FIFO
- FIFO 触发深度软件可编程
- 占用 16 个 8 位 IO 空间地址
- IO 地址拨码开关设定
- 中断和查询工作方式
- 16 位 PC/104 总线
- 工作温度-40---+85 摄氏度

2. 功能描述

本产品为实现 ARINC 429 总线接口功能的 PC/104 板卡。由两组 ARINC429 专用芯片共同实现四路接收通道，二路发送通道。每组芯片可独立编程,设定其波特率（100K bps 或 12.5K bps）和数据格式（32 bit 或 25 bit）。每个接收和发送通道配有独立的 256 X 2 BYTE 的 FIFO，以减轻主机的负担，防止主机来不及响应而丢数。每个接收通道还可以编程设定触发深度，当接收 FIFO 中的数据达到触发深度后，置触发状态，和或中断。板卡地址用拨码开关设置。

3. 逻辑框图



二. 引脚定义

1. 地址拨码开关 SW1

SW1 用于设定板卡的基地址和中断请求号。

IRQ11	IRQ10	SA4	SA5	SA6	SA7	SA8	SA9
-------	-------	-----	-----	-----	-----	-----	-----

8

1

拨码开关往下的位置为：ON，往上拨的位置为：OFF。

对于板卡基地址来说，响应位置的 ON 对应 0，OFF 对应 1。

设定中断号时，ON 表示板卡中断状态与对应 PC/104 总线中断请求线相连。OFF 表示板卡中断状态与对应 PC/104 总线中断请求线断开。

如要将板卡基地址设定为 340H，中断请求号设定为 IRQ10，SW1 的状态应如下所示：

OFF	ON	ON	ON	OFF	ON	OFF	OFF
-----	----	----	----	-----	----	-----	-----

8

1

2. ARINC429 信号连接器 J5

2

20

Out0B	GND	GND	GND	GND	GND	GND	GND	GND	Out1B
Out0A	In1B	In1A	In0B	In0A	In3B	In3A	In2B	In2A	Out1A

1

19

Out0A,Out0B 为第 0 路 发送通道信号线

Out1A,Out1B 为第 1 路 发送通道信号线

In0A,In0B 为第 0 路 接收通道信号线

In1A,In1B 为第 1 路 接收通道信号线

In2A,In2B 为第 2 路 接收通道信号线

In3A,In3B 为第 3 路 接收通道信号线

三. 地址空间映象

1. 地址空间

主机通过 16 位 PC/104 总线进行访问板卡，板卡共占用 16 个 8 位的 IO 地址空间，在接口板上有一地址设置拨码开关 (SW1)，它的前 6 个开关对应板卡地址的高六位，用来设置板卡的基地址 (BaseAddr)。这样板卡占用的 IO 地址空间为：BaseAddr+0H ~ BaseAddr+f H，分别对应板上的众多 16 位功能寄存器。

2. 功能寄存器概况

寄存器名	描述	读/写	地址
RCVR0	第一路接收 FIFO	只读	BaseAddr+0
RCVR1	第二路接收 FIFO	只读	BaseAddr+2
RCVR2	第三路接收 FIFO	只读	BaseAddr+4
RCVR3	第四路接收 FIFO	只读	BaseAddr+6
FSR0	FIFO 状态寄存器 0	只读	BaseAddr+8
FSR1	FIFO 状态寄存器 1	只读	BaseAddr+a
ISR	中断状态寄存器	只读	BaseAddr+c
SETR	板卡设置寄存器	只读	BaseAddr+e
TRANSR0	第一路发送 FIFO	只写	BaseAddr+0
TRANSR1	第二路发送 FIFO	只写	BaseAddr+2
CS0	429 数据格式、波特率设置寄存器 0	只写	BaseAddr+4
CS1	429 数据格式、波特率设置寄存器 1	只写	BaseAddr+6
RFTLR0	接收 FIFO 触发深度寄存器 0	只写	BaseAddr+8
RFTLR1	接收 FIFO 触发深度寄存器 1	只写	BaseAddr+a
TFTLR	发送 FIFO 触发深度寄存器	只写	BaseAddr+c
SETR	板卡设置寄存器	只写	BaseAddr+e

3. 功能寄存器位定义

(1) FSR0 (FIFO 状态寄存器 0)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RER3	RF3	RE3	RT3	RER2	RF2	RE2	RT2	RER1	RF1	RE1	RT1	RER0	RF0	RE0	RT0

RTx: 第 x 路接收 FIFO 已经达到设定的触发深度。

1: 第 x 路接收 FIFO 达到设定的触发深度，0: 第 x 路接收 FIFO 没有达到设定的触发深度。接收 FIFO 达到设定的触发深度，表示 FIFO 中至少有触发深度规定数量的 32 位 ARINC 429 数据等待主机读取。

REx: 第 x 路接收 FIFO 空标志。

1: 第 x 路接收 FIFO 空 , 0: 第 x 路接收 FIFO 不空。接收 FIFO 不空, 表示 FIFO 中至少有一组 32 位 ARINC 429 数据等待主机读取。

RFx: 第 x 路接收 FIFO 满标志。

1: 第 x 路接收 FIFO 满 , 0: 第 x 路接收 FIFO 不满。接收 FIFO 满, 表示 FIFO 中有 127 (FF) 组 32 位 ARINC 429 数据等待主机读取。此时, 所有新到 ARINC429 数据被自动丢弃。

RERx: 第 x 路接收通道, 由于接收 FIFO 满, 有数据溢出。复位命令可以将此状态位清零。

(2) FSR1 (FIFO 状态寄存器 1)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	TW1	TW0	RW3	RW2	RW1	RW0	TER1	TF1	TE1	TT1	TER0	TF0	TE0	TT0

TTx: 第 x 路发送 FIFO 已经达到设定的触发深度

1: 第 x 路发送 FIFO 达到设定的触发深度 , 0: 第 x 路发送 FIFO 没有达到设定的触发深度。发送 FIFO 达到设定的触发深度, 表示发送 FIFO 中, 所剩下的 32 位 ARINC 429 数据少于触发深度规定的数量。

TEx: 第 x 路发送 FIFO 空标志。

1: 第 x 路发送 FIFO 空 , 0: 第 x 路发送 FIFO 不空。发送 FIFO 不空, 表示发送 FIFO 中至少有一组 32 位 ARINC 429 数据没有发送 (等待发送)。

TFx: 第 x 路发送 FIFO 满标志。

1: 第 x 路发送 FIFO 满 , 0: 第 x 路发送 FIFO 不满。发送 FIFO 满, 表示 FIFO 中有 127 (FF) 组 32 位 ARINC 429 数据等待发送。此时, 写发送 FIFO, 所写数据会被自动丢弃。发送 FIFO 不满, 主机可以向其中写入待发送数据。

TERx: 第 x 路发送 FIFO 已满, 但由于主机没有判断 FIFO 状态, 继续写入, 导致所写数据被丢弃。复位命令可以将此状态位清零。

RWx: 第 x 路接收 FIFO 同步状态。

0: 接收 FIFO 当前数据为低位 16Bit ARINC429 数据。即 (D0 -----D15)

1: 接收 FIFO 当前数据为高位 16Bit ARINC429 数据。即 (D16 -----D32)

TWx: 第 x 路发送 FIFO 同步状态。

0: 发送 FIFO 将要写入数据为低位 16Bit ARINC429 数据。即 (D0 -----D15)

1: 发送 FIFO 将要写入数据为高位 16Bit ARINC429 数据。即 (D16 -----D32)

(3) ISR (中断状态寄存器)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	INT	0	TTI1	TTI0	RTI3	RTI2	RTI1	RTI0

INT: 板卡中断请求位。

1: 中断条件满足, 有中断请求产生。 0: 不满足中断条件, 没有中断请求产生。

TTIx: 发送 FIFO 触发中断申请

1: 第 x 路发送 FIFO 中, 所剩数据小于或等于相应触发深度, 产生了中断申请。

0: 第 x 路发送 FIFO 没有产生中断申请。

RTIx: 接收 FIFO 触发中断申请

1: 第 x 路接收 FIFO 中, 所存数据大于相应触发深度, 产生了中断申请。

0: 第 x 路接收 FIFO 没有产生中断申请。

(4) CS0 (429 数据格式、波特率设置寄存器 0)

设置第一, 二路接收通道和第一路发送通道的数据格式和波特率。

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
WLSEL	RCVSEL	TXSEL	PARCK	Y2	X2	SDENB2	Y1	X1	SDENB1	SLFTST	PAREN	X	X	X	X

PAREN: 发送通道校验使能。

1: bit32 为校验位。 0: bit32 为数据位。

SLFTST: 自检使能。

1: 正常工作状态。 0: 自检状态。

在自检状态下, 第一路发送通道的输出, 在内部被连接到第一和第二接收通道。其中, 数据在进入第二接收通道前被反相, 以便与第一接收通道区分。自检状态下, 发送通道仍然正常输出。

SDEN1: 第一路接收通道 S/D 码检测使能。

1: 使能第一路接收通道, 源/目标译码器。 0: 关闭第一路接收通道, 源/目标译码器。

X1, Y1: 第一路接收通道 S/D 码。

如果 SDEN1=1, 第一路接收通道将收到 ARINC429 数据的 S/D 与 X1,Y1 进行比较, 相同则接纳, 不相同则丢弃。其中, X1 与串行数据 bit9 比较, Y1 与串行数据 bit10 比较。

SDEN2: 第二路接收通道 S/D 码检测使能。

1: 使能第二路接收通道, 源/目标译码器。 0: 关闭第二路接收通道, 源/目标译码器。

X2, Y2: 第二路接收通道 S/D 码。

如果 SDEN2=1, 第二路接收通道将收到 ARINC429 数据的 S/D 与 X2,Y2 进行比较, 相同则接纳, 不相同则丢弃。其中, X2 与串行数据 bit9 比较, Y2 与串行数据 bit10 比较。

PARCK: 校验检测使能。

1: 将发送通道校验位取反 (使用偶校验), 以测试校验电路。 0: 使用正常的奇校验。

TXSEL: 第一路发送通道波特率选择。

1: 低速 (12.5 K bps) 0: 高速 (100 K bps)。

RCVSEL: 第一, 二路接收通道波特率选择。

1: 低速 (12.5 K bps) 0: 高速 (100 K bps)。

WLSEL: 字长度选择

1: 25 位。 0: 32 位

(5) CS1 (429 数据格式、波特率设置寄存器 1)

设置第三, 四路接收通道和二路发送通道的数据格式和波特率。

定义与 CS0 相对应。如需要将接收通道, 发送通道, 都设置为: 100K bps, 32bit, 奇校验。不使用 S/D 功能, 相应的设定值为 0020h。

(6) RFTLR0 (接收 FIFO 触发深度寄存器 0)

D6 ~ D0: 第一路接收 FIFO 触发深度设置

D14 ~ D8: 第二路接收 FIFO 触发深度设置

D7、D15: 没有用到

单位为 32 bit, 即 2 个 16 bit WORD

(7) RFTLR1 (接收 FIFO 触发深度寄存器 1)

D6 ~ D0: 第三路接收 FIFO 触发深度设置

D14 ~ D8: 第四路接收 FIFO 触发深度设置

D7、D15: 没有用到

单位为 32 bit, 即 2 个 16 bit WORD

(8) TFTLR (发送 FIFO 触发深度寄存器)

D6 ~ D0: 第一路发送 FIFO 触发深度设置

D14 ~ D8: 第二路发送 FIFO 触发深度设置

D7、D15: 没有用到

单位为 32 bit, 即 2 个 16 bit WORD

(9) SETR (板卡设置寄存器)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
IE	ETT1	ETT0	ERT3	ERT2	ERT1	ERT0	X	X	X	X	X	RST3	RST2	RST1	RST0

RSTx: 板卡复位命令

1: 板卡内部状态复位。 0: 正常工作状态。

在开始工作前, 需将 RSTx 置 1, 然后再清 0。(x 对板卡用户没有实际意义)。

IE: 板卡中断允许

1: 允许本卡产生中断。 0: 禁止本卡产生中断

ETT_x: 第 x 路发送 FIFO 触发深度中断允许

1: TTx=1 时, 允许产生中断 0: 不允许 TTx 产生中断

ERT_x: 第 x 路接收 FIFO 触发深度中断允许

1: RTx=1 时, 允许产生中断 0: 不允许 RTx 产生中断

X: 没有用到

四. 使用说明

1. 硬件设置

使用板卡前, 首先须通过 SW1 设置好其基地址, 如果要使用中断, 还要将板卡的中断请求通过 SW1 与对应的 PC/104 总线的中断请求线相连。

将 ARINC429 总线的信号线和地线通过专用电缆, 与板卡的 J5 连接器对应信号相连。

将板卡插入主机的 PC/104 总线。

2. 软件编程方法

(1) 板卡初使化

在操作板卡前, 首先必须要对板卡进行初始化。初使化分四步。

第一步复位板卡, 举例如下:

```
outport(BaseAddr+e, 0x000f); //对 RSTx 置 1
```

```
outport(BaseAddr+e, 0x0000); //对 RSTx 清 0
```

第二步设置 ARINC429 通道参数, 举例如下:

```
outport(BaseAddr+4, 0x0020);
```

```
outport(BaseAddr+6, 0x0020);
```

第三步设置各通道 FIFO 触发深度，举例如下：

```
outport(BaseAddr+8, 0x0505);
//第一，第二路接收通道 FIFO 的触发深度都为 5，
//当 FIFO 中有 6 组或 6 组以上的 429 数据时，
//相应 RT0 或 RT1 位为 1
outport(BaseAddr+a, 0x0606);
//第三，第四路接收通道 FIFO 的触发深度都为 6，
//当 FIFO 中有 7 组或 7 组以上的 429 数据时，
//相应 RT2 或 RT3 位为 1
outport(BaseAddr+c, 0x0808);
//第一，第二路发送通道 FIFO 的触发深度都为 8，
//当 FIFO 中只剩下 8 组或少于 8 组的 429 数据时，
//相应 TT0 或 TT1 位为 1
```

第四步设置中断屏蔽字，举例如下：

```
outport(BaseAddr+e, 0x9e00);
//对应位置 1，相应的状态为 1 时，可触发中断，为 0 则屏蔽该路中
//断，IE 为 0 则屏蔽所有中断。本例使用四路接收通道触发中断。
```

(2) 接收 ARINC429 数据

首先判断接收 FIFO 中是否有数，有数则直接从相应 FIFO 中读取，举例如下：

```
unsigned int  flag, DataL, DataH;
flag=0x0002;
while((flag&0x0002))//第一路接收 FIFO 为空，等待
{
flag=inport(BaseAddr+8);
}
DataL= inport(BaseAddr+0); //低 16 位 429 字
DataH= inport(BaseAddr+0); //高 16 位 429 字
```

(3) 发送 ARINC429 数据

首先判断发送 FIFO 是否满，不满则直接向相应 FIFO 中写入要发送的 429 字，低 16 字在前，高 16 位字在后，举例如下：

```
unsigned int  flag, DataL, DataH;
DataL=0x1234;
DataH=0x4567;
flag=0x0004;
while((flag&0x0004))//第一路发送 FIFO 满，等待
{
flag=inport(BaseAddr+8);
}
```



```
outport(BaseAddr+0,DataL); //低 16 位 429 字  
outport(BaseAddr+0, DataH); //高 16 位 429 字
```

(4) 使用中断

本板卡可以通过 SW1 选用 IRQ10, 或 IRQ11。在初始化中, 要将相应状态的屏蔽位置为 1, 并且, IE 也要置 1。在中断服务程序中通过读取本卡的 ISR 寄存器, 判断中断源。

五. 定货信息

GWT429PC104B

GLE: 公司标志

429: 功能

PC/104: 总线类型

B: 配置