



Gowin PicoRV32 软件编程参考手册

IPUG911-1.0,2020-01-16

版权所有©2020 广东高云半导体科技股份有限公司

未经本公司书面许可，任何单位和个人不得擅自摘抄、复制、翻译本档内容的部分或全部，并不得以任何形式传播。

免责声明

本档并未授予任何知识产权的许可，并未以明示或暗示，或以禁止发言或其它方式授予任何知识产权许可。除高云半导体在其产品的销售条款和条件中声明的责任之外，高云半导体概不承担任何法律或非法律责任。高云半导体对高云半导体产品的销售和 / 或使用不作任何明示或暗示的担保，包括对产品的特定用途适用性、适销性或对任何专利权、版权或其它知识产权的侵权责任等，均不作担保。高云半导体对档中包含的文字、图片及其它内容的准确性和完整性不承担任何法律或非法律责任，高云半导体保留修改档中任何内容的权利，恕不另行通知。高云半导体不承诺对这些档进行适时的更新。

版本信息

日期	版本	说明
2020/01/16	1.0	初始版本。

目录

目录	i
表目录	ii
1 软件编程库	1
2 存储系统	2
3 中断处理	3
3.1 中断特征	3
3.2 中断控制	3
3.3 中断编号	4
4 Simple UART	5
4.1 特征	5
4.2 寄存器定义	5
4.3 驱动程序使用方法	5
5 Wishbone I2C Master	6
5.1 特征	6
5.2 寄存器定义	6
5.3 驱动程序使用方法	7
6 Wishbone SPI Master/Slave	8
6.1 特征	8
6.2 寄存器定义	8
6.3 驱动程序使用方法	9
7 Wishbone UART	10
7.1 特征	10
7.2 寄存器定义	10
7.3 驱动程序使用方法	10

表目录

表 1-1 Gowin_PicoRV32 软件编程库	1
表 2-1 Gowin_PicoRV32 标准外设内存映射	2
表 3-1 Gowin_PicoRV32 中断控制函数.....	3
表 3-2 Gowin_PicoRV32 中断编号	4
表 4-1 UART 寄存器定义.....	5
表 4-2 Simple UART 驱动程序使用方法.....	5
表 5-1 Wishbone I2C Master 寄存器定义.....	6
表 5-2 Wishbone I2C Master 驱动程序使用方法.....	7
表 6-1 Wishbone SPI 寄存器定义.....	8
表 6-2 Wishbone SPI 驱动程序使用方法.....	9
表 7-1 Wishbone UART 寄存器定义.....	10
表 7-2 Wishbone UART 驱动程序使用方法.....	10

1 软件编程库

Gowin_PicoRV32 提供软件编程库 Gowin_PicoRV32\src\c_lib, 如表 1-1 所示。

表 1-1 Gowin_PicoRV32 软件编程库

文件	描述
start.S	Gowin_PicoRV32启动引导程序
custom_ops.S	自定义中断处理指令宏定义
picorv32.h	寄存器、地址映射、时钟信号、系统配置等定义
irq.c	中断处理函数
sections.lds	Flash链接器脚本
firmware.c	性能统计、延时等定义
printf.c	Simple UART/WB UART打印输出
simpleuart.c	Simple UART的驱动函数定义
wbi2c.c	Wishbone I2C Master的驱动函数定义
wbspi.c	Wishbone SPI Master/Slave的驱动函数定义
wbuart.c	Wishbone UART的驱动函数定义

2 存储系统

Gowin_PicoRV32 标准外部设备内存映射地址，如表 2-1 所示。

表 2-1 Gowin_PicoRV32 标准外设内存映射

标准外设	类型	地址映射	描述
DLM	-	0x00000000	1KB, 2KB, 4KB, 8KB, 16KB, 32KB, 64KB, 128KB, 256KB
ILM	-	0x01000000	1KB, 2KB, 4KB, 8KB, 16KB, 32KB, 64KB, 128KB, 256KB
SIMPLEUART	SIMPLEUART_RegDef	0x02000000	Simple UART
WBSPI_MASTER	WBSPI_RegDef	0x10000000	Wishbone SPI Master
WBSPI_SLAVE	WBSPI_RegDef	0x10001000	Wishbone SPI Slave
WBUART	WBUART_RegDef	0x10002000	Wishbone UART
WBI2CM	WBI2CMMASTER_RegDef	0x10003000	Wishbone I2C Master
OPEN_WB_INTERFACE	-	0x20000000	Open Wishbone Interface for Customized Peripherals

3 中断处理

3.1 中断特征

中断控制器包括以下特征：

- 轻量化的中断控制器和中断控制指令
- 中断由上升沿触发
- 提供最多 32 个用户可用的中断处理信号
- 软件可控制的中断优先级
- 不支持中断嵌套

3.2 中断控制

Gowin_PicoRV32 中断控制函数，如表 3-1 所示。

表 3-1 Gowin_PicoRV32 中断控制函数

中断控制函数	参数	描述
mask_irq	32 位屏蔽码	中断屏蔽函数： - 屏蔽码中为 0 的位对应的中断为开放状态 - 屏蔽码中为 1 的位对应的中断为屏蔽状态
irq_mask_one_bit	32 位中断编号	单个中断屏蔽函数： 该编号对应的中断位被屏蔽。
irq_unmask_one_bit	32 位中断编号	单个中断解除屏蔽函数： 该编号对应的中断位解除屏蔽。
set_timer	32 位的计时长 度 timer_val	定时器中断设置函数： 自动开始计时，当计时达到 timer_val 个时钟周 期时，触发定时器中断。
irq	32 位中断寄存 器 32 位中断编号	中断处理函数： 当触发中断后，自动进入该函数，根据触发中断 的中断位编号，调用相应的中断处理函数。

3.3 中断编号

Gowin_PicoRV32 预置的中断编号，如表 3-2 所示。

表 3-2 Gowin_PicoRV32 中断编号

编号	描述
0	32位定时器
1	EBREAK/ECALL指令中断
2	内存访问非对齐错误
3	Wishbone I2C Master中断
4	Wishbone SPI Master中断
5	Wishbone SPI Slave中断
20 ~ 31	预留给Wishbone扩展总线接口，供用户扩展外部设备使用

4 Simple UART

4.1 特征

Gowin_PicoRV32 包含一个轻量级的通用异步收发器 Simple UART:

- 无奇偶校验位
- 8 位数据位
- 1 位停止位
- 不支持中断

用户在使能 Simple UART 时, 需设置波特率分频寄存器, 例如, 如果系统时钟频率运行在 12MHz, 需要波特率为 9600, 则可以设置波特率分频寄存器为 $12000000/9600=1250$ 。

4.2 寄存器定义

Simple UART 寄存器定义如表 4-1 所示。

表 4-1 UART 寄存器定义

寄存器名称	地址偏移	类型	宽度	初始值	描述
CLKDIV	0x004	RW	32	0x1	分频系数寄存器, 用于配置波特率
DATA	0x008	RW	32	0x-	输入/输出数据寄存器

4.3 驱动程序使用方法

Simple UART 驱动程序使用方法如表 4-2 所示。

表 4-2 Simple UART 驱动程序使用方法

名称	描述
uart_init	对 Simple UART 进行初始化, 配置波特率
outbyte	输出一个字符, 当输出换行符时自动返回行首
uart_putchar	输出一个字符
getchar_prompt	返回 UART RX 接收到的字符
uart_getchar	返回 UART RX 接收到的字符

5 Wishbone I2C Master

5.1 特征

Gowin_PicoRV32 包含一个通过 Wishbone 总线访问的内部集成电路 I2C Master 模块：

- Wishbone 总线接口
- 符合业界标准的 I2C 总线协议
- 总线仲裁及仲裁丢失检测
- 总线忙状态检测
- 产生中断标志
- 产生起始、终止、重复起始和应答信息
- 支持起始、终止和重复起始检测
- 支持 7 位寻址模式

5.2 寄存器定义

Wishbone I2C Master 寄存器定义如表 5-1 所示。

表 5-1 Wishbone I2C Master 寄存器定义

寄存器名称	地址偏移	类型	宽度	初始值	描述
PRER	0x00	RW	32	0x0000 FFFF	Clock prescale register [31:15] Reserved [15:0] Prescale value = $\text{sys_clk}/(5*\text{SCL})-1$
CTR	0x04	RW	32	0x0000 0000	[31:8] Reserved [7] Enable I2C function [6] Enable I2C interrupt [5:0] Reserved
TXR	0x08	WO	32	0x0000 0000	[31:8] Reserved [7:1] Next transmission data [0] Data direction
RXR	0x08	RO	32	0x0000 0000	[31:8] Reserved [7:0] Last received data
CR	0x0c	WO	32	0x0000 0000	[31:8] Reserved

寄存器名称	地址偏移	类型	宽度	初始值	描述
					[7] STA, Start transmission status [6] STO, Over transmission status [5] RD, Read enable, read data from slave [4] WR, Write enable, write data to slave [3] Acknowledge [2:1] Reserved [0] Interrupt acknowledge
SR	0x0c	RO	32	0x0000 0000	[31:8] Reserved [7] Receive acknowledge signal from slave [6] I2C busy status [5] Arbitration loss [4:2] Reserved [1] Data transmission status flag [0] Interrupt flag

5.3 驱动程序使用方法

Wishbone I2C Master 驱动程序使用方法如表 5-2 所示。

表 5-2 Wishbone I2C Master 驱动程序使用方法

名称	描述
I2C_Init	I2C Initialization
I2C_SendByte	Send a byte to I2C bus
I2C_SendBytes	Send multiple bytes to I2C bus
I2C_SendData	Send multiple bytes to I2C bus once time
I2C_ReceiveByte	Read a byte from I2C bus
I2C_ReadBytes	Read multiple bytes from I2C bus
I2C_ReceiveData	Read multiple bytes from I2C bus once time
I2C_Rate_Set	Set I2C traffic rate
I2C_Enable	Enable I2C bus
I2C_Disable	Disable I2C bus
I2C_InterruptOpen	Open I2C interrupt
I2C_InterruptClose	Close I2C interrupt

6 Wishbone SPI Master/Slave

6.1 特征

Gowin_PicoRV32 包含一个通过 Wishbone 总线访问的串行外设 Master 接口 SPI Master 和一个通过 Wishbone 总线访问的串行外设 Slave 接口 SPI Slave:

- Wishbone 总线接口
- 全双工同步串行数据传输
- 可配置时钟极性和相位
- SPI 产生的串行时钟频率可配置
- 可配置的数据接收寄存器和数据发送寄存器位宽

6.2 寄存器定义

Wishbone SPI 寄存器定义如表 6-1 所示。

表 6-1 Wishbone SPI 寄存器定义

寄存器名称	地址偏移	类型	宽度	初始值	描述
RDATA	0x00	RO	32	0x0000 0000	Read data register [31:8] Reserved [7:0] Read data
WDATA	0x04	WO	32	0x0000 0000	Write data register [31:8] Reserved [7:0] Write data
STATUS	0x08	RW	32	0x0000 0000	[31:8] Reserved [7] Overflow error status [6] Receive ready status [5] Transmit ready status [4] Be transmitting [3] Transmit overrun error status [2] Receive overrun error status [1:0] Reserved
CTRL	0x0C	RW	32	0x0000 0000	[31:5] Reserved [4:3] Clock selected, CLK_I / 2/4/6/8

寄存器名称	地址偏移	类型	宽度	初始值	描述
					[2] Clock polarity [1] Clock phase [0] Direction, 1 is MSB first
SSMASK	0x10	RW	32	0x0000 0000	[31:1] Reserved [0] Select and enable slave

6.3 驱动程序使用方法

Wishbone SPI 驱动程序使用方法如表 6-2 所示。

表 6-2 Wishbone SPI 驱动程序使用方法

名称	描述
wbspi_master_select_slave	WBSPI Master选择一个从设备进行通信
wbspi_enable_interrupt	开启WBSPI的中断
wbspi_disable_interrupt	关闭WBSPI的中断
wbspi_master_txdata	WBSPI Master向从设备发送数据
wbspi_master_rxdata	WBSPI Master读取从设备的数据
wbspi_slave_prepare_txdata	WBSPI Slave预备数据，等待主设备读取
wbspi_slave_read_data	WBSPI Slave读取主设备发送过来的数据

7 Wishbone UART

7.1 特征

Gowin_PicoRV32 包含一个通过 Wishbone 总线访问的 UART:

- Wishbone 总线接口
- 无奇偶校验位
- 8 位数据位
- 1 位停止位

7.2 寄存器定义

Wishbone UART 寄存器定义如表 6-1 所示。

表 7-1 Wishbone UART 寄存器定义

寄存器名称	地址偏移	类型	宽度	初始值	描述
SETUP	0x00	RW	32	0x00000000	UART参数/设置寄存器
FIFO	0x04	RO	32	0x00000000	输入FIFO和输出FIFO的状态寄存器
RXREG	0x08	RO	32	0x00000000	UART接受数据寄存器
TXREG	0x0C	RW	32	0x00000000	UART发送数据寄存器

7.3 驱动程序使用方法

Wishbone UART 驱动程序使用方法如表 7-2 所示。

表 7-2 Wishbone UART 驱动程序使用方法

名称	描述
wbuart_init	Wishbone UART初始化，配置波特率
wbuart_putc	Wishbone UART发送一个字节
wbuart_getc	Wishbone UART接受一个字节
wbuart_outbyte	Wishbone UART发送一个字节，并在输出换行时自动返回行首

