

Gowin_EMPU_M1 **硬件设计参考手册**

IPUG531-1.2,2019-08-18

版权所有©2019 广东高云半导体科技股份有限公司

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

免责声明

本文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。高云半导体对文档中包含的文字、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文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文档进行适时的更新。

版本信息

| 日期 | 版本 | 说明 |
|------------|-----|--|
| 2019/02/19 | 1.0 | 初始版本。 |
| 2019/07/18 | 1.1 | MCU硬件设计支持扩展外部设备CAN、Ethernet、SPI-Flash、 RTC、DualTimer、TRNG、I2C、SPI、SD-Card。 |
| 2019/08/18 | 1.2 | MCU硬件设计与软件编程设计支持扩展外部设备 DDR3 Memory; 修复已知 ITCM、DTCM Size 和 IDE 问题。 |

目录

| 目 | 录i |
|---|---------------------------------|
| 图 | 目录iii |
| 表 | ·目录 |
| 1 | 硬件架构 |
| | 1.1 系统架构1 |
| | 1.2 系统特征 |
| | 1.2.1 Cortex-M1 子系统 |
| | 1.2.2 AHB-Lite Extension 子系统3 |
| | 1.3 系统端口 |
| | 1.3.1 Cortex-M1 端口 |
| | 1.3.2 AHB-Lite Extension 端口4 |
| 2 | 硬件设计流程7 |
| | 2.1 硬件环境7 |
| | 2.2 软件环境7 |
| | 2.3 软核生成器7 |
| | 2.4 下载软件7 |
| | 2.5 设计流程 |
| 3 | 工程模板9 |
| | 3.1 工程创建9 |
| | 3.1.1 新建工程9 |
| | 3.1.2 设定工程名称和路径10 |
| | 3.1.3 选择器件10 |
| | 3.1.4 完成工程创建 11 |
| | 3.2 硬件设计11 |
| | 3.2.1 Cortex-M1 硬件设计12 |
| | 3.2.2 AHB-Lite Extension 硬件设计17 |
| | 3.3 用户设计 |
| | 3.4 约束 |

| 4 | 参考设计 | 38 |
|---|---------------------------|----|
| | 3.7 布局布线 | 37 |
| | 3.6 综合 | 36 |
| | 3.5.3 Dual-Purpose Pin 配置 | 35 |
| | 3.5.2 Post-Place File 配置 | 35 |
| | 3.5.1 顶层模块配置 | 34 |
| | 3.5 配置 | 34 |

图目录

| 图 1-1 Gowin_EMPU_M1 系统架构 | 1 |
|---------------------------|----|
| 图 3-1 新建 FPGA Design 工程 | 9 |
| 图 3-2 设定工程名称和路径 | 10 |
| 图 3-3 选择器件 | 10 |
| 图 3-4 完成工程创建 | 11 |
| 图 3-5 选择 Gowin_EMPU_M1 | 11 |
| 图 3-6 Gowin_EMPU_M1 系统架构 | 12 |
| 图 3-7 Cortex-M1 配置页面 | 13 |
| 图 3-8 Cortex-M1 通用配置 | 14 |
| 图 3-9 Cortex-M1 调试配置 | 15 |
| 图 3-10 Cortex-M1 存储配置 | 16 |
| 图 3-11 GPIO 配置 | 18 |
| 图 3-12 CAN 配置 | 19 |
| 图 3-13 Ethernet 配置 | 20 |
| 图 3-14 DDR3 配置 | 21 |
| 图 3-15 SPI-Flash 配置 | 22 |
| 图 3-16 AHB2 Extension 配置 | 23 |
| 图 3-17 UART 配置 | 24 |
| 图 3-18 Timer 配置 | 25 |
| 图 3-19 WatchDog 配置 | |
| 图 3-20 RTC 配置 | 27 |
| 图 3-21 DualTimer 配置 | |
| 图 3-22 TRNG 配置 | 29 |
| 图 3-23 I2C 配置 | 30 |
| 图 3-24 SPI 配置 | 31 |
| 图 3-25 SD-Card 配置 | 32 |
| 图 3-26 APB2 Extension 配置 | 33 |
| 图 3-27 顶层模块配置 | |
| 图 3-28 Post-Place File 配置 | 35 |

| 图 3-29 Dual-Purpose Pin 配置 | |
|----------------------------|----|
| 图 3-30 综合 | 36 |
| 图 3-31 Place & Route | 37 |

表目录

| 表 1-1 Cortex-M1 端口 | 3 |
|-------------------------------|----|
| 表 1-2 AHB-Lite Extension 端口 | 4 |
| 表 3-1 Cortex-M1 配置选项 | 12 |
| 表 3-2 AHB-Lite Extension 配置选项 | 17 |

】硬件架构

1.1 系统架构

Gowin_EMPU_M1 包括三级结构,如图 1-1 所示。





- 第一级, Cortex-M1 及 ITCM、DTCM
- 第二级,AHB 总线及 GPIO、CAN、Ethernet、DDR3 Memory、SPI-Flash、 AHB2 Extension Bus
- 第三级, APB 总线及 UART0、UART1、Timer0、Timer1、Watch Dog、 RTC、TRNG、DualTimer、I2C、SPI、SD-Card、APB2 Extension Bus

1.2 系统特征

Gowin_EMPU_M1 包括两个子系统:

- Cortex-M1 内核子系统
- AHB-Lite 扩展 AHB 总线和 APB 总线,及外设接口子系统

1.2.1 Cortex-M1 子系统

处理器内核

- ARM architecture v6-M Thumb 指令集架构,支持 16-bit Thumb 和 32-bit Thumb2 指令集
- 可配置扩展操作系统
- 系统异常处理
- 中断异常处理和正常线程模式
- 栈指针,正常是一个栈指针,扩展操作系统时两个栈指针
- 大小端格式
 - 可配置数据大小端格式
 - 指令和系统控制寄存器小端格式
 - 调试系统小端格式

NVIC

- 可配置外部中断数量,1、8、16、32
- 4个优先级等级
- 进入中断处理时自动保存处理器状态,中断处理结束时自动恢复状态

调试系统

- 不支持调试系统
- 支持调试系统
 - 可配置完整(full)模式和简化(reduced)模式
 - 完整模式: 4个 BreakPoint Unit 和 2个 Data Watchpoint
 - 简化模式: 2个 BreakPoint Unit 和 1 个 Data Watchpoint
 - 可配置 DAP 端口
 - JTAG/SW
 - JTAG
 - SW

Memory

- ITCM: 指令存储器,可配置大小, Upper 或 Lower alias 方式和读入初 始值
- DTCM: 数据存储器,可配置大小

32 位硬件乘法器

- Normal 模式
- Small 模式

1.2.2 AHB-Lite Extension 子系统

- AHB 总线,及 GPIO、CAN、Ethernet、DDR3 Memory、SPI-Flash、 AHB2 Extension
- APB 总线,及UART0、UART1、Timer0、Timer1、Watch Dog、RTC、 DualTimer、TRNG、I2C、SPI、SD-Card、APB2 Extension

1.3 系统端口

1.3.1 Cortex-M1 端口

Cortex-M1 端口如表 1-1 所示。

表 1-1 Cortex-M1 端口

| 名称 | I/O | 位宽 | 描述 |
|--------------|-----|--------|---------------------------------|
| HCLK | in | 1 | System clock |
| SYSRESETn | in | 1 | System reset |
| DBGRESETn | in | 1 | Power on debug reset |
| LOCKUP | out | 1 | Core is in lockup state |
| HALTED | out | 1 | Core is in Halt debug state |
| nTRST | in | 1 | JTAG reset |
| SWCLKTCK | in | 1 | Serial wire and JTAG clock |
| SWDITMS | in | 1 | SW Data / JTAG Test Mode Select |
| TDI | in | 1 | JTAG data input |
| JTAGNSW | out | 1 | JTAG = 1, serial wire = 0 |
| JTAGTOP | out | 1 | state controller indicator |
| TDO | out | 1 | JTAG data output |
| nTDOEN | out | 1 | JTAG data out enable |
| SWDO | out | 1 | Serial wire data out |
| SWDOEN | out | 1 | Serial data output enable |
| IRQ | in | [31:0] | External interrupts |
| NMI | in | 1 | Non-maskable interrupt |
| SYSRESETREQ | out | 1 | System reset require |
| EDBGRQ | in | 1 | External debug request |
| DBGRESTART | in | 1 | Restart from halt request |
| DBGRESTARTED | out | 1 | Restart from halt acknowledge |
| HREADY | in | 1 | Slave ready signal |
| HRESP | in | 1 | Slave response signal |
| HRDATA | in | [31:0] | Data from slave to master |
| HTRANS | out | [1:0] | Transfer type |
| HBURST | out | [2:0] | Burst type |
| HPROT | out | [3:0] | Transfer protection bits |
| HSIZE | out | [2:0] | Transfer size |
| HWRITE | out | 1 | Transfer direction |
| HMASTLOCK | out | 1 | Transfer is a locked transfer |
| HADDR | out | [31:0] | Transfer address |
| HWDATA | out | [31:0] | Data from master to slave |

1.3.2 AHB-Lite Extension 端口

AHB-Lite Extension 端口如表 1-2 所示。

表 1-2 AHB-Lite Extension 端口

| 名称 | I/O | 位宽 | 描述 | 所属模块 |
|--------------|-------|--------|------------------------|-----------|
| SYSRESETn | in | 1 | 系统复位 | |
| PORESETn | in | 1 | 上电复位 | |
| IRQ | out | [31:0] | 外部中断 | |
| NMI | out | 1 | Non-Maskable 中断 | |
| GPIO | inout | [15:0] | 通用输入输出端口 | GPIO |
| UARTORXD | in | 1 | UARTO 接收端口 | |
| UARTOTXD | out | 1 | UART0 发送端口 | UARTU |
| UART1RXD | int | 1 | UART1 接收端口 | |
| UART1TXD | out | 1 | UART1 发送端口 | UARTI |
| TIMER0EXTIN | in | 1 | Timer0 外部中断 | Timer0 |
| TIMER1EXTIN | in | 1 | Timer1 外部中断 | Timer1 |
| WDOGRESREQ | out | 1 | Watch Dog 复位请求 | Watch Dog |
| RTCSRCCLK | in | 1 | RTC 时 钟 源 32.768KHz | RTC |
| SCL | inout | 1 | 串行时钟 | 100 |
| SDA | inout | 1 | 串行数据 | 120 |
| MOSI | out | 1 | 主设备输出/从设备输 | |
| MISO | in | 1 | 入 主设备输入/从设备输 | SPI |
| | | • | | 511 |
| SCLK | out | 1 | 时钟信号 | |
| NSS | out | 1 | 从设备选择信号 | |
| SD_CLK | out | 1 | 时钟信号 | |
| SD_CS | out | 1 | 片选信号 | |
| SD_DATAIN | out | 1 | 数据输入 | |
| SD_DATAOUT | in | 1 | 数据输出 | SD-Card |
| SD_CARD_INIT | out | 1 | 初始化"0" | |
| SD_CHECKIN | in | 1 | 输入检查 | |
| SD_CHECKOUT | out | 1 | 输出检查 | |
| CAN_RX | in | 1 | 数据输入 | CAN |
| CAN_TX | out | 1 | 数据输出 | |
| RGMII_TXC | out | 1 | RGMII 发送时钟 | |
| RGMII_TX_CTL | out | 1 | RGMII 发送控制 | |
| RGMII_TXD | out | [3:0] | RGMII 发送数据 | Ethernet |
| RGMII_RXC | in | 1 | RGMII 接收时钟 | RGMII |
| RGMII_RX_CTL | in | 10 | RGMII 接收控制 | Interface |
| RGMII_RXD | in | [3:0] | RGMII 接收数据 | |
| GTX_CLK | in | 1 | RGMII 125MHz 时钟输 入 | |
| GMII_RX_CLK | in | 1 | GMII 接收时钟 | |
| GMII_RX_DV | in | 1 | GMII 接收使能 | Ethernet |
| GMII_RXD | in | [7:0] | GMII 接收数据 | GIVIII |
| GMII_RX_ER | in | 1 | GMII 接收错误 | menace |

| 名称 | I/O | 位宽 | 描述 | 所属模块 |
|---------------------|-------|--------|-------------------------|---------------|
| GTX_CLK | in | 1 | GMII 125MHz 时钟输 λ | |
| GMII GTX CLK | out | 1 | GMII 发送时钟 | |
| GMII TXD | out | [7:0] | GMII 发送数据 | |
| GMII TX EN | out | 1 | GMII 发送使能 | |
| GMII TX ER | out | 1 | GMII 发送错误 | |
| MII_RX_CLK | in | 1 | MII 接收时钟 | |
| MII_RXD | in | [3:0] | MII 接收数据 | |
| MII_RX_DV | in | 1 | MII 接收使能 | |
| MII_RX_ER | in | 1 | MII 接收错误 | |
| MII_TX_CLK | in | 1 | MII 发送时钟 | Ethernet |
| MII_TXD | out | [3:0] | MII 发送数据 | MII Interface |
| MII_TX_EN | out | 1 | MII 发送使能 | |
| MII_TX_ER | out | 1 | MII 发送错误 | |
| MII_COL | in | 1 | MII 冲突信号 | |
| MII_CRS | in | 1 | MII 载波信号 | |
| MDC | out | 1 | 管理通道时钟 | Ethernet |
| MDIO | inout | 1 | 管理通道数据 | Luiemer |
| DDR_CLK_I | in | 1 | 50MHz 时钟输入 | |
| DDR_INIT_COMPLETE_O | out | 1 | 初始化完成信号 | |
| DDR_ADDR_O | out | [15:0] | Row 地址、Column 地 址 | |
| DDR_BA_O | out | [2:0] | Bank 地址 | |
| DDR_CS_N_O | out | 1 | 片选信号 | |
| DDR_RAS_N_O | out | 1 | Row 地址选通信号 | |
| DDR_CAS_N_O | out | 1 | Column 地址选通信号 | |
| DDR_WE_N_O | out | 1 | Row 写使能 | |
| DDR_CLK_O | out | 1 | 提供给 DDR3 SDRAM 的时钟信号 | |
| DDR_CLK_N_O | out | 1 | 与 DDR_CLK_O 组成 差分信号 | DDR3 |
| DDR_CKE_O | out | 1 | DDR3 SDRAM 时钟使 能信号 | |
| | out | 1 | 此百 5 内存信号端接由阳控制 | |
| | out | • | DDR3 SDRAM 复位信 | |
| DDR_RESET_N_O | out | 1 | 号 | |
| DDR_DQM_O | out | [1:0] | DDR3 SDRAM 数据屏 蔽信号 | |
| DDR_DQ_IO | inout | [15:0] | DDR3 SDRAM 数据 | |
| DDR_DQS_IO | inout | [1:0] | DDR3 SDRAM 数据选 通信号 | |
| DDR_DQS_N_IO | inout | [1:0] | 与 DDR_DQS_IO 组成 差分信号 | |
| FLASH_SPI_HOLDN | inout | 1 | NC | |
| FLASH_SPI_CSN | inout | 1 | 从设备选择信号 | |
| FLASH_SPI_MISO | inout | 1 | 主设备输入/从设备输 出 | SPI-Flash |
| FLASH_SPI_MOSI | inout | 1 | 主设备输出/从设备输 | |

| 名称 | I/O | 位宽 | 描述 | 所属模块 |
|---------------|-------|--------|----------------|-----------|
| | | | λ | |
| FLASH_SPI_WPN | inout | 1 | NC | |
| FLASH_SPI_CLK | inout | 1 | 时钟信号 | |
| APB2PSTRB | out | [3:0] | APB2 PSTRB | |
| APB2PPROT | out | [2:0] | APB2 PPROT | |
| APB2PSEL | out | 1 | APB2 PSEL | |
| APB2PENABLE | out | 1 | APB2 PENABLE | |
| APB2PADDR | out | [31:0] | APB2 PADDR | |
| APB2PWRITE | out | 1 | APB2 PWRITE | APB2 |
| APB2PWDATA | out | [31:0] | APB2 PWDATA | Extension |
| APB2PRDATA | in | [31:0] | APB2 PRDATA | |
| APB2PREADY | in | 1 | APB2 PREADY | |
| APB2PSLVERR | in | 1 | APB2 PSLVERR | |
| APB2PCLK | out | 1 | APB2 PCLK | |
| APB2PRESET | out | 1 | APB2 RESET | |
| AHB2HSEL | out | 1 | AHB2 HSEL | |
| AHB2HADDR | out | [31:0] | AHB2 HADDR | |
| AHB2HTRANS | out | [1:0] | AHB2 HTRANS | |
| AHB2HWRITE | out | 1 | AHB2 HWRITE | |
| AHB2HSIZE | out | [2:0] | AHB2 HSIZE | |
| AHB2HBURST | out | [2:0] | AHB2 HBURST | |
| AHB2HPROT | out | [3:0] | AHB2 HPROT | |
| AHB2HWDATA | out | [31:0] | AHB2 HWDATA | Extension |
| AHB2HMASTLOCK | out | 1 | AHB2 HMASTLOCK | LAGHSION |
| AHB2HRDATA | in | [31:0] | AHB2 HRDATA | |
| AHB2HREADYOUT | in | 1 | AHB2 HREDAYOUT | |
| AHB2HRESP | in | [1:0] | AHB2 HRESP | |
| AHB2HMASTER | out | [3:0] | AHB2 MASTER |] |
| AHB2HCLK | out | 1 | AHB2 HCLK | |
| AHB2HRESET | out | 1 | AHB2 RESET | |



2.1 硬件环境

- DK-START-GW2A18 V2.0 GW2A-LV18PG256C8/I7
- DK-START-GW1N9 V1.1 GW1N-LV9LQ144C6/I5
- DK-START-GW2A55 V1.1 GW2A-LV55PG484C8/I7

2.2 软件环境

GOWIN FPGA Designer version 1.9.x Beta

2.3 软核生成器

GOWIN FPGA Designer 提供软核生成器 IP Core Generator,用于配置和产生 Cortex-M1 和 AHB-Lite Extension 硬件设计。

IP Core Generator 软件使用方法请参考 SUG284, <u>Gowin IP Core</u> Generator 用户指南。

2.4 下载软件

Gowin_EMPU_M1 支持 Programmer 软件下载码流文件。

Programmer 软件使用方法请参考 SUG502, <u>Gowin Programmer 用户</u> <u>指南</u>。

2.5 设计流程

Gowin_EMPU_M1 硬件设计流程:

- 1. IP Core Generator 软核生成器配置 Cortex-M1、APB Bus Peripherals 和 AHB Bus Peripherals,产生 Gowin_EMPU_M1 硬件设计,导入工程;
- 2. 实例化 Gowin_EMPU_M1,导入用户设计、物理约束、时序约束,连接 端口;
- 3. Synplify_Pro 或 GowinSynthesis 综合, Place & Route 布局布线, 产生 码流文件;
- 4. Programmer 下载。



3.1 工程创建

3.1.1 新建工程

双击 GOWIN FPGA Designer,选择菜单栏 File 中的 New...,选择 FPGA Design Project,如图 3-1 所示。

图 3-1 新建 FPGA Design 工程



3.1.2 设定工程名称和路径

输入工程名称,选择工程路径,如图 3-2 所示。

| 图 3-2 | 设定工程名称和路径 |
|-------|-----------|
|-------|-----------|

| ** | Project Wizard | × |
|--|---|------|
| Project Name Select Device Summary | Project Name Enter a name for your project, and specify a directory where the project will be stored. The directory w be created if it doesn't exist. | rill |
| | Name: gowin_empu_m1 Create in: /home/liukai Use as default peoject location | |

3.1.3 选择器件

选择器件、封装和速度等,如图 3-3 所示。

图 3-3 选择器件

| ** | | Project Wiz | ard | | | | | × |
|--|---|------------------------------|--|-------------------------|---------------------|------------------|-----------------------|----------------------|
| Project Name Select Device Summary | Select Device | r proiect | Device: Package Speed: | GW2 :: PBGA C8/17 | A-18 A256 | | | ÷ |
| | Part Number GW2A-LV18PG256C8/I7 GW2A-UV18PG256C8/I7 | Device GW2A-18 GW2A-18 | Package PBGA256 PBGA256 | Speed C8/I7 C8/I7 | Voltage LV UV | IO 207 207 | LUT 20736 20736 | FF 15552 15552 |
| | | m | | | | | | |
| | | | | | | < <u>B</u> a | ck | Next > |

3.1.4 完成工程创建

如图 3-4 所示,完成新建工程。

图 3-4 完成工程创建

| ** | Project Wizard | : |
|--------------------------|---|-------------------------------|
| Project Name | Summary | |
| Select Device Summary | <pre>Project Name: gowin_empu_m1 Directory: /home/liukai/gowin_empu_m1/src Implementation Directory: /home/liukai/gowin_empu_m1/impl Device Part Number: GW2A-LV18PG256C8/I7 Series: GW2A Device: GW2A-18 Package: PBGA256 Speed: C8/I7</pre> | |
| | | < <u>B</u> ack <u>F</u> inish |

3.2 硬件设计

使用 IP Core Generator 产生 Gowin_EMPU_M1 硬件设计。

选择菜单栏 Tools 中的 IP Core Generator,打开 IP Core Generator 后, 选择 Soft IP Core 列表下 Microprocessor System\Soft-Core-MCU\Gowin_EMPU_M1,如图 3-5 所示。

图 3-5 选择 Gowin_EMPU_M1

| GOWIN FPGA Designer - [IP Core Generator] | | | | |
|---|---|---|--|--|
| 🚴 File Edit Tools Window Help | | | | |
| 🗋 📂 🗄 🖷 🖶 🗠 / | a 🍾 🖻 🛍 🖪 is 👶 | | | |
| Design 🕑 🗵 | Target Device: GW2A-LV18PG256C8/I7 | | | |
| Govin_empu_m1 - (/home/luk Govin_empu_m1 - (/home/luk Govin_empu_m1 - (/home/luk) | Name Hand Module CLOCK CLOCK CLOCK Soft Soft Soft Soft Soft Soft Soft Soft | <section-header><section-header><section-header><section-header><section-header><text><text><text><text></text></text></text></text></section-header></section-header></section-header></section-header></section-header> | | |
| | | | | |
| Output | | | | |

打开 Gowin_EMPU_M1, Gowin_EMPU_M1 系统架构如图 3-6 所示,

包括 Cortex-M1 和 AHB Lite Extension(APB Bus Peripherals 和 AHB Bus Peripherals)选择配置。

| le arget Device: GWZA-LV18L0144C8/7 treate In: /nome/fliukai/gowin_empu_m1 Cortex-M1 INO(31:0) Corte | | Gowin EMPU M1 |
|--|--------------|---|
| arget Device: GW2A-LV18LQ144C8/7 Language: Verilog • treate In: /home/liukai/gowin_empu_m1 · · · · Cortex-M1 | le | |
| reate In: //ome/liukai/gowin_empu_m1 | arget Device | e: GW2A-LV18LQ144C8/I7 Language: Verilog |
| Cortex-M1 IRO(31.0) TCM Cortex-M1 DTCM AHB-Lite AHB-Lite AHB-Lite GPIO CAN Ethernel DDR3 SPI-Fiash AHB2AFB AHB2AFB AHB2AFB AHB2AFB TIMET TIMETO TIM | reate In: | /home/liukai/gowin_empu_m1 |
| Cortex-M1 IRQ(31:0) ITCM Cortex-M1 DTCM AHB-Lite AHB Bus GPID CAN Ethernet DDR3 SPI-Flash Extension AHB2 LTC SPI SPI-Flash Extension CAN Etherson Etherson CAN Ether | | |
| Cotex-M1 ITCM Cotex-M1 AHB-Lite AHB Bus GPIO CAN Ethernet DDR3 SPI-Flash AHB2 GPIO CAN Ethernet DDR3 SPI-Flash Categories AHB2APB UARTO UARTI TIMEO TIMEI D09 RTC TING Dual 2C SPI SD-Card APB20 Extension ITCM Categories Cotex-M1 DDR3 SPI-Flash Categories Cotex-M1 DDR3 SPI-Flash Categories Cotex-M1 DDR3 SPI-Flash Categories Cotex-M1 Cotex-M1 DDR3 SPI-Flash Categories Cotex-M1 Cotex-M1 DDR3 SPI-Flash Categories Cotex-M1 Cotex-M1 DDR3 SPI-Flash Categories Cotex-M1 Cotex-M1 DDR3 SPI-Flash Categories Cotex-M1 Cotex-M1 Cotex-M1 DDR3 SPI-Flash Categories Cotex-M1 Cotex-M1 Cotex-M1 Cotex-M1 DDR3 SPI-Flash Categories Cotex-M1 Cote | | |
| Cortex-M1 IRQ[31:0] ITCM Cortex-M1 DEBUG TTCM Cortex-M1 DTCM AHB-Lite AHB-Lite AHB-Lite AHB-Lite CRIO CAN Ethernet DDR3 SPI-Flash AHB2 Extension AHB2APB CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CO Extension Ethernet DDR3 SPI-Flash Ethernet CO SPI SD-Card Extension Ethernet CO CAN Ethernet DDR3 SPI-Flash Ethernet CO SPI SD-Card Ethernet CO Extension Ethernet CO SPI SD-Card Ethernet CO SD-Card Ethernet CO CAN Ethernet CO SPI SD-Card Ethernet CO SD-Card CO SD-Card Ethernet CO SD-Card Ethernet CO SD-Card Ethernet CO SD-Card Ethernet CO SD-Card Ethernet CO SD-Card Ethernet CO SD-Card CO SD-Card Ethernet CO SD-Card CO SD-CAR CO SD-CAR CO SD-CAR | | |
| AHB Bus AHB Bus AHB Bus AHB CAN Ethernet DDB3 SPI-Flash AHB2 AHB ZAPB AHB ZAPB AHBZAPA AHBZAPA | Cortex-M1 | |
| AHB BUS AHB BUS AHB BUS AHB CONCENT GPIO CAN Ethernet DDR3 SPI-Flash AHB2 Extension AHB2APB AHB2APB AHB2APB CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet DDR3 SPI-Flash Extension CAN Ethernet CAN Eth | | |
| AHB BUS AHB BUS AHB BUS APB BUS APB BUS APB BUS AHB2APB AHBA AHBA APB AHBA AH | | |
| AHB BUS AHB BUS GPIO CAN Ethernet DDR3 SPI-Flash AHB2 Extension APB BUS AHBZAPB AHBZAPB AHBZAPB AHBZAPB AHBZAPB CAN UARTI Timer0 Timer1 Watch RTC TRNG Dual I2C SPI SD-Card APB2 Extension Concert APB2 Extension AHBZAPB AHBZAPA AHBZAPB AHBZAPA AHBZ | | ITCM Cortex-M1 DTCM |
| AHB Bus GPIO CAN Ethernet DDR3 SPI-Flash AHB2 AHBZAPB AHBZAPB AHBZAPB UARTO UARTI TIMERO TIMERI Watch RTC TRNG Dual IZC SPI SD-Card Extension Watch Correct Market Correct Corre | | AHB-Lite |
| AHB Bus GPIO CAN Ethernet DDR3 SPI-Flash AHB2 Extension APB Bus AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB CARTO UARTI Timer0 Timer1 Watch RTC TRNG Dual I2C SPI SD-Card APB2 Extension Correct APB2 Extension Correct APB2 Extension Correct APB2 Correct APB2 | | × |
| APB Bus APB Bus APB Bus AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB AHB2APB AHB2 APB Compared APB2 Extension Compared APB2 Extension Compared APB2 Compared A | AHB Bus | |
| APB Bus APB Bus APB Card APB Card AP | | |
| APB Bus AHB2APB AHB2APB UARTO UART1 Timer0 Timer1 Watch RTC TRNG Dual I2C SPI SD-Card APB2 Extension Extension Extension | | GPIO CAN Ethernet DDR3 SPI-Flash AHB2 Extension |
| APB Bus AHB2APB AHB2APB AHB2APB UARTO UARTI Timer0 Timer1 Watch RTC TRNG Dual I2C SPI SD-Card APB2 Extension Extension Extension | | |
| AHB2APB UARTO UARTI Timer0 Timer1 Watch RTC TRNG Dual I2C SPI SD-Card APB2 Extension | APB Bus | |
| UARTO UARTI Timero Timeri Watch RTC TRNG Dual I2C SPI SD-Card APB2 Extension | | АНВЗАРВ |
| UARTO UARTI Timero Timeri Dog RTC TRNG Timer IZC SPI SD-Card Extension | 1 | |
| | | UARTI Timero Timero Watch RTC TRNG Dual Izc Spi Sp. card APB2 |
| | OARTO | Dog Net Timer Timer Solution Extension |
| | | |
| | | |
| | | |
| | Holp | |

图 3-6 Gowin_EMPU_M1 系统架构

3.2.1 Cortex-M1 硬件设计

Cortex-M1 硬件设计配置选项如表 3-1。

表 3-1 Cortex-M1 配置选项

| 配置选项 | 描述 |
|----------------------|--|
| Number of | 配置 Cortex-M1 外部中断数量,可选择 1 或 8 或 16 或 32, 默 |
| interrupts | 认为 32 |
| OS Extension | 配置 Cortex-M1 可以支持操作系统 |
| Small Multiplier | 配置 Cortex-M1 small 模式硬件乘法器 |
| Big Endian | 配置 Cortex-M1 数据大端格式 |
| Enable Debug | 使能 Cortex-M1 Debug 功能 |
| Debug Port | 配置调试器接口,可以选择 JTAG 或 Serial Wire 或 JTAG and |
| Select | Serial Wire |
| Small Debug | 配置 Small 模式调试器 |
| ITCM Alias Select | 配置 ITCM Alias,可以选择 Upper 或 Lower |

| 配置选项 | 描述 |
|-----------------------------|--|
| | 配置ITCM的大小,可以选择1KB或2KB或4KB或8KB或16KB |
| | 或 32KB 或 64KB 或 128KB 或 256KB |
| | ● GW1N-9 系列最大选择 32KB |
| ITCM Size | ● GW1NR-9 系列最大选择 32KB |
| | ● GW2A-18 系列最大选择 64KB |
| | ● GW2AR-18 系列最大选择 64KB |
| | ● GW2A-55 系列最大选择 256KB |
| Initialize ITCM | 使能 ITCM 初始化 |
| ITCM Initialization Path | ITCM 初始值文件路径 |
| | 配置 DTCM 的大小,可以选择 1KB 或 2KB 或 4KB 或 8KB 或 |
| | 16KB 或 32KB 或 64KB 或 128KB 或 256KB |
| | ● GW1N-9 系列最大选择 32KB |
| DTCM Size | ● GW1NR-9 系列最大选择 32KB |
| | ● GW2A-18 系列最大选择 64KB |
| | ● GW2AR-18 系列最大选择 64KB |
| | ● GW2A-55 系列最大选择 256KB |

双击 Cortex-M1, 打开 Cortex-M1 的配置页面, 如图 3-7 所示, 包括通用配置、调试配置和存储配置。

图 3-7 Cortex-M1 配置页面



通用配置

选择通用配置选项卡,如图 3-8 所示,可以配置中断数量、操作系统扩

| | | Cortex-M1 |
|--|--|---|
| ortex-M | 1 | 6 |
| HCLK SYSRESETn DBGRESETn nTRST TDI SWCLKTCK | LOCKUP HALTED JTAGTOP nTDOEN SWDO SWDOEN TDO | Gowin Cortex-M1 is intended for deeply embedded applications that is integrated into Gowin FPGA. Gowin Cortex-M1 supports debug and non-debug core, operating system, nested vectored interrupt controller, ITCM and DTCM, and AHB-Lite interface. User can configure number of interrupts, size of ITCM and DTCM, OS extension, small o normal multiplier, big or little endian, full or small debug, debug port. |
| ► SWDITMS | SYSRESETREQ | Common Debug Memory |
| RQ[31:0] | | Number of interrupts |
| NMI | HBURST[2:0] | 01 |
| EDBGRQ | HPROT[3:0] | 08 |
| DBGRESTART | HSIZE[2:0] | 0.16 |
| HREADY | HWRITE | © 10 © 32 |
| HRESP | HMASTLOCK | |
| | HADDR[31:0] | ✓ OS Extension |
| HRDAIA[51:0] | HWDATA[31:0] | Small Multiplier |
| Gowin C | Cortex-M1 | 🗌 Big Endian |
| | | |

● 中断数量配置

可以选择 1 或 8 或 16 或 32,可以配置 1 个或 8 个或 16 个或 32 个外部 中断。

● 操作系统扩展配置

如果选择,则 Cortex-M1 扩展支持操作系统。

● 乘法器模式配置

如果选择,则 Cortex-M1 支持 Small 乘法器,否则支持 Normal 乘法器。

● 数据存储格式配置

如果选择,则 Cortex-M1 支持数据大端格式,否则支持数据小端格式。 调试配置

选择调试配置选项卡,如图 3-9 所示,可以配置使能调试、调试接口和调试器模式。

| \$ | | Cortex-M1 × |
|---|---|---|
| Cortex-M1 | L | 🐣 |
| HCLK SYSRESETN DBGRESETN NTRST TDI SWCLKTCK SWDITMS IRQ(31:0) NMI EDBGRQ DBGRESTART HREADY HREADY HRDATA[31:0] Gowin Cu | LOCKUP HALTED JTAGTOP nTDOEN SWDO SWDOEN TDO SYSRESETREQ DBGRESTARTED HTRANS[1:0] HBURST[2:0] HSIZE[2:0] HSIZE[2:0] HWRITE HMASTLOCK HADDR[31:0] HWDATA[31:0] HWDATA[31:0] HWDATA[31:0] | Gowin Cortex-M1 is intended for deeply embedded applications that is integrated into Gowin Cortex-M1 supports debug and non-debug core, operating system, nested vectored interrupt controller, ITCM and DTCM, and AHB-Lite interface. User can configure number of interrupts, size of ITCM and DTCM, OS extension, small or normal multiplier, big or little endian, full or small debug, debug port. Common Debug Memory ✓ Enable Debug Debug Port Select JTAG Serial Wire Small Debug |
| | | |

图 3-9 Cortex-M1 调试配置

• Enable Debug

如果选择 Enable Debug, Cortex-M1 支持调试功能, 否则 Cortex-M1 不支持调试功能。

● 调试接口配置

可以选择 JTAG 或 Serial Wire 或 JTAG and Serial Wire。

● 调试器模式配置

如果选择,则 Cortex-M1 支持 Small 模式调试器,否则支持 Full 模式调试器。

存储配置

选择存储配置选项卡,如图 3-10 所示,可以配置 ITCM 和 DTCM。

| | - | Gowin Cortex-M1 is intended for deeply embedded applications that is integrated into Gowin FPGA. Gowin Cortex-M1 supports debug and non-debug core, operating system, nested |
|--------------|--------------|---|
| HCLK | LOCKUP | vectored interrupt controller, ITCM and DTCM, and AHB-Lite interface. User can configure number of interrupts, size of ITCM and DTCM, OS extension, small normal multiplier, big or little endian, full or small debug, debug port. |
| SYSRESETn | HALTED | |
| DBGBESETn | JTAGTOP | |
| TRET | nTDOEN | |
| | swdo 🔶 | |
| TDI | SWDOEN | |
| SWCLKTCK | | |
| SWDITMS | | Common Debug Memory |
| IRQ[31:0] | | TITCM Alias Select |
| NMI | HBURST[2:0] | O Enable Upper Alias |
| EDBGRQ | HPROT[3:0] | Enable Lower Alias |
| DBGRESTART | HSIZE[2:0] | |
| | HWRITE | |
| UDEED | HMASTLOCK | ✓ Initialize ITCM |
| | HADDR[31:0] | ITCM Initialization Path: |
| HRDAIA[31:0] | HWDATA[31:0] | DTCM |
| Gowin C | Cortex-M1 | DTCM Size 32K |
| | | |

图 3-10 Cortex-M1 存储配置

- ITCM 配置
 - ITCM Alias 配置可以选择 Enable Upper 或 Lower Alias。
 - ITCM Size 配置
 - 可以选择 1KB 或 2KB 或 4KB 或 8KB 或 16KB 或 32KB 或 64KB 或 128KB 或 256KB。
 - GW1N-9 系列 ITCM Size 最大选择为 32KB。
 - GW1NR-9 系列 ITCM Size 最大选择为 32KB。
 - GW2A-18 系列 ITCM Size 最大选择为 64KB。
 - GW2AR-18 系列 ITCM Size 最大选择为 64KB。
 - GW2A-55 系列 ITCM Size 最大选择为 256KB。
 - ITCM Initialization 配置如果选择 Initialize ITCM,则支持 ITCM 初始 化,可以在 ITCM Initialization Path 导入 ITCM 初始值文件路径。
 - 如果选择使用片外 SPI-Flash 下载启动方式, ITCM 初始值导入 bootload 文件路径。
- DTCM 配置
 - DTCM Size 配置可以选择 1KB 或 2KB 或 4KB 或 8KB 或 16KB 或 32KB 或 64KB 或 128KB 或 256KB。
 - GW1N-9 系列 DTCM Size 最大选择为 32KB。
 - GW1NR-9 系列 DTCM Size 最大选择为 32KB。
 - GW2A-18 系列 DTCM Size 最大选择为 64KB。
 - GW2AR-18 系列 DTCM Size 最大选择为 64KB。

- GW2A-55 系列 DTCM Size 最大选择为 256KB。

3.2.2 AHB-Lite Extension 硬件设计

AHB-Lite Extension 配置选项如表 3-2 所示。

表 3-2 AHB-Lite Extension 配置选项

| 配置选项 | 描述 |
|-----------------------|--|
| Enable GPIO | 使能 GPIO |
| Enable CAN | 使能 CAN |
| Buffer Depth | CAN 选择 Buffer Depth |
| Enable Ethernet | 使能 Ethernet |
| Interface | Ethernet 选择 Interface (RGMII/GMII/MII) |
| RGMII Input Delay | RGMII input delay |
| MIIM Clock Divider | MIIM clock divider |
| Enable DDR3 | 使能 DDR3 Memory |
| Enable SPI-Flash | 使能 SPI-Flash |
| Enable AHB2 Extension | 使能 AHB2 扩展接口 |
| Enable UART0 | 使能串口 0 |
| Enable UART1 | 使能串口 1 |
| Enable Timer0 | 使能定时器 0 |
| Enable Timer1 | 使能定时器 1 |
| Enable WatchDog | 使能看门狗 |
| Enable RTC | 使能 RTC |
| Enable TRNG | 使能 TRNG |
| Enable DualTimer | 使能 DualTimer |
| Enable I2C | 使能 I2C |
| Enable SPI | 使能 SPI |
| Enable SD-Card | 使能 SD-Card |
| Enable APB2 Extension | 使能 APB2 扩展接口 |

GPIO 配置

双击打开 GPIO,可以选择配置 GPIO,如图 3-11 所示。 如果选择 Enable GPIO,则 Gowin_EMPU_M1 支持 GPIO。

```
图 3-11 GPIO 配置
```



CAN 配置

- 双击打开 CAN,可以选择配置 CAN,如图 3-12 所示。
- 如果选择 Enable CAN,则 Gowin_EMPU_M1 支持 CAN。
- 如果已经选择 Enable CAN,则可以配置 Buffer Depth。
- 选择 Buffer Depth, 配置 Buffer Depth。

图 3-12 CAN 配置



Ethernet 配置

- 双击打开 Ethernet,可以选择配置 Ethernet,如图 3-13 所示。
- 如果选择 Enable Ethernet,则 Gowin_EMPU_M1 支持 Ethernet。
- 如果已经选择 Enable Ethernet,则可以配置 Interface、RGMII Input Delay、MIIM Clock Divider。
 - 选择 Interface,可以选择配置 RGMII、GMII、MII 或 GMII/MII。
 - 如果选择 Interface 为 RGMII,则可以选择配置 RGMII Input Delay。
 - 选择 MIIM Clock Divider,可以配置 MIIM Clock Divider。
- 如果 Interface 选择 RGMII 或 GMII,则端口 GTX_CLK 必须接入 125MHz 时钟输入。

图 3-13 Ethernet 配置

| â | Ethernet × |
|--|--|
| Ethernet | |
| → SYSRESETN IRQ[31:0] → PORESRTN MDC → RGMII_RXC | The Cortex-M1 core of Gowin_EMPU_M1 interacts with Ethernet block through AHB bus.The Ethernet block is a fully verified block, which can implement Ethernet MAC layer function and provide RGMII/GMII/MII interface to connect to PHY IC. The block implements MAC layer protocol. |
| | Options |
| RGMII_RXD[3:0] | Configuration Cnfiguration Cnfiguration |
| | Interface: RGMII \$ |
| Gowin AHB Bus | MIIM Clock Divider: 20 |
| | |
| | Scancel |

DDR3 配置

- 双击打开 DDR3,可以选择配置 DDR3,如图 3-13 所示。
- 如果选择 Enable DDR3,则 Gowin_EMPU_M1 支持 DDR3。
- DDR3 端口 DDR_CLK_I 必须接入 50MHz 时钟输入。

图 3-14 DDR3 配置

| ♣ | | DDR3 × |
|--------------------------------|--|--|
| DDR3 | | |
| SYSRESETn PORESRTn DDR_CLK_1 | IRQ(31:0) NMI DDR_INIT_COMPLETE_O DDR_ADDR_O(15:0) DDR_BA_O(2:0) DDR_CS_N_O DDR_CAS_N_O DDR_CAS_N_O DDR_CAS_N_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_CLK_O DDR_ODT_O DDR_RESET_N_O DDR_DQS_IO(1:0) DDR_DQS_IO(1:0) CDR_DQS_N_IO(1:0) CDR_ | The Cortex-M1 Core of Gowin_EMPU_M1 interacts with DDR3 block through AHB bus. The DDR3 block provides a complete solution for customers to use DDR3 memory. This block located between the DDR3 memory and the user logic include with Gowin Memory Controller and Physical interface, reduces the usera'rs effort to deal with the DDR3 memory and interface by providing a simple generic system interface to the user. Options Configuration ✓ Enable DDR3 Enable DDR3 |
| | Gowin And Bus | |
| 2 | | Cancel QK |

SPI-Flash 配置

- 双击打开 SPI-Flash,可以选择配置 SPI-Flash,如图 3-15 所示。
- 如果选择 Enable SPI-Flash,则 Gowin_EMPU_M1 支持 SPI-Flash。
- 如果 Gowin_EMPU_M1 使用片外 SPI-Flash 下载启动方式,则必须选择 Enable SPI-Flash。

| 冬 | 3-15 | SPI-Flash | 配置 |
|---|------|-----------|----|
|---|------|-----------|----|

| â | SPI-Flash × |
|--|---|
| SPI-Flash | en e |
| IRQ(31:0) NMI SYSRESETN FLASH_SPI_HOLDN FLASH_SPI_CSN | The Cortex-M1 core of Gowin_EMPU_M1 interacts with SPI-Flash block through AHB bus.The SPI-Flash block is a fully verified block which can be as Cortex-M1 Coreâ'rs chip-off running memory.The SPI-Flash block is SPI interface. |
| FLASH_SPI_MISO | Options |
| -> PORESRTN FLASH_SPI_MOSI | Configuration |
| FLASH_SPI_WPN | ✓ Enable SPI-Flash |
| FLASH_SPI_CLK | |
| Gowin AHB Bus | |
| | S Cancel |

AHB2 Extension 配置

- 双击打开 AHB2 Extension,可以选择配置 AHB2 Extension,如图 3-16 所示。
- 如果选择 Enable AHB2 Extension,则 Gowin_EMPU_M1 支持 AHB2 Extension,用户可以在此接口自行连接扩展的 AHB 外部设备。

图 3-16 AHB2 Extension 配置

| * | AHB2 Extension × |
|--|--|
| AHB2 Extension | |
| SYSRESETN NMI AHB2HSEL | The Cortex-M1 core of Gowin_EMPU_M1 interacts with AHB2 Extension block through AHB bus. The AHB2 Extension block is a fully verified block which can extend AHB bus to user to design more AHB peripherals. |
| PORESRTn AHB2HADDR[31:0] AHB2HRDATA[31:0] AHB2HRDATA[31:0] | |
| AHB2HRDAIA(31.0) AHB2HSIZE[2:0] | Options |
| AHB2HREADYOUT AHB2HBURST[2:0] AHB2HPROT[3:0] | Configuration |
| AHB2HRESP[1:0] AHB2HWDATA[31:0] AHB2HMASTLOCK | |
| Gowin AHB Bus | |
| | |

UART 配置

- 双击打开 UARTO 或 UART1,可以选择配置 UART0 或 UART1,如图 3-17 所示。
- 如果选择 Enable UART0,则 Gowin_EMPU_M1 支持 UART0。
- 如果选择 Enable UART1,则 Gowin_EMPU_M1 支持 UART1。

图 3-17 UART 配置



Timer 配置

- 双击打开 Timer0 或 Timer1,可以选择配置 Timer0 或 Timer1,如图 3-18 所示。
- 如果选择 Enable Timer0,则 Gowin_EMPU_M1 支持 Timer0。
- 如果选择 Enable Timer1,则 Gowin_EMPU_M1 支持 Timer1。

图 3-18 Timer 配置



WatchDog 配置

- 双击打开 WatchDog,可以选择配置 WatchDog,如图 3-19 所示。
- 如果选择 Enable WatchDog,则 Gowin_EMPU_M1 支持 WatchDog。

图 3-19 WatchDog 配置

| ♣ | WatchDog x |
|---|--|
| WatchDog | |
| PORESRTN WDOGRESREQ Gowin APB Bus | The Cortex-M1 core of Gowin_EMPU_M1 interacts with WatchDog block through APB bus.The WatchDog block is a fully verified block which can implement system reset when system is running over. Options Configuration ✓ Enable WatchDog |
| | Cancel QK |

RTC 配置

- 双击打开 RTC,可以选择配置 RTC,如图 3-20 所示。
- 如果选择 Enable RTC,则 Gowin_EMPU_M1 支持 RTC。

图 3-20 RTC 配置



DualTimer 配置

- 双击打开 DualTimer,可以选择配置 DualTimer,如图 3-21 所示。
- 如果选择 Enable DualTimer,则 Gowin_EMPU_M1 支持 DualTimer。

图 3-21 DualTimer 配置



TRNG 配置

- 双击打开 TRNG,可以选择配置 TRNG,如图 3-22 所示。
- 如果选择 Enable TRNG,则 Gowin_EMPU_M1 支持 TRNG。

图 3-22 TRNG 配置



I2C 配置

- 双击打开 I2C,可以选择配置 I2C,如图 3-23 所示。
- 如果选择 Enable I2C,则 Gowin_EMPU_M1 支持 I2C。

图 3-23 I2C 配置



SPI 配置

- 双击打开 SPI,可以选择配置 SPI,如图 3-24 所示。
- 如果选择 Enable SPI,则 Gowin_EMPU_M1 支持 SPI。

图 3-24 SPI 配置



SD-Card 配置

- 双击打开 SD-Card,可以选择配置 SD-Card,如图 3-25 所示。
- 如果选择 Enable SD-Card,则 Gowin_EMPU_M1 支持 SD-Card。
- 此版本 SD-Card 属性为只读 SD-Card。

图 3-25 SD-Card 配置

| Å | | SD-Card × |
|----------|---|---|
| SD-Card | | |
| | IRQ[31:0] → NMI → SD_CLK → SD_CS → SD_DATAIN → SD_CARD_INIT → SD_CHECKOUT → | The Cortex-M1 core of Gowin_EMPU_M1 interacts with SD-Card block through APB bus. The SD-Card block is a fully verified block which can implement SD-Card memory to save data. Options Configuration ✓ Enable SD-Card |
| Gowin A | AHB Bus | |
| | | |

APB2 Extension 配置

- 双击打开 APB2 Extension,可以选择配置 APB2 Extension,如图 3-26 所示。
- 如果选择 Enable APB2 Extension,则 Gowin_EMPU_M1 支持 APB2 Extension,用户可以在此接口自行连接扩展的 APB 外部设备。

图 3-26 APB2 Extension 配置

| 4 | | APB2 Extension × |
|--|---|---|
| APB2 Exte | ension | 🐣 |
| SYSRESETN PORESRTN APB2PRDATA[31:0] APB2PREDADY APB2PSLVERR Gowin | IRQ[31:0] NMI APB2PSTRB[3:0] APB2PPROT[2:0] APB2PSEL APB2PENABLE APB2PADDR[31:0] APB2PWRITE APB2PWDATA[31:0] APB Bus | The Cortex-M1 core of Gowin_EMPU_M1 interacts with APB2 Extension block through APB bus. The APB2 Extension block is a fully verified block which can extend APB bus to user to design more APB peripherals. Options Configuration ✓ Enable APB2 Extension |
| | | Sancel Cancel |

3.3 用户设计

- 完成 Gowin_EMPU_M1 配置后,产生 Gowin_EMPU_M1 硬件设计
- 实例化 Gowin_EMPU_M1
- 导入用户设计,连接 Gowin_EMPU_M1,形成完整的 RTL 设计

3.4 约束

完成用户 RTL 设计后,根据使用的开发板和需要输出的 IO,产生物理 约束文件。

根据时序要求,产生时序约束文件。

物理约束的产生方法请参考 SUG101, Gowin 设计约束指南。

3.5 配置

3.5.1 顶层模块配置

综合配置中,根据工程设计中的实际顶层模块名称配置此处顶层模块名称,如图 **3-27** 所示。

图 3-27 顶层模块配置

| ** | Configurations | × |
|---|--|---|
| | Synthesize | |
| General | General Svnthesis Tool: Svnplify Pro O GowinSvnth | esis |
| Place & Route General Dual-Purpose Pin BitStream | Top Module/Entity: Gowin_empu_M1 Include Path: Synplify Pro Category: All | Reset all to default |
| | Label Frequency Use Clock Period for Unconstrainted IO Fanout Guide Disable I/O Insertion Update Compile Point Timing Data Read Write Check on RAM | Value Auto False 10000 False False True |
| | V A | pply 🛛 🔁 Cancel 🖉 OK |

3.5.2 Post-Place File 配置

如果使用 Gowin_EMPU_M1 软件编程设计和硬件设计自动化合并的下载方法,则配置 Place & Route 产生 Post-Place File,如图 3-28 所示,否则不需要配置输出 Post-Place File。

图 3-28 Post-Place File 配置

| 👯 Configurations | | | |
|--------------------------|---|----------------------|---|
| | Place & Route | | |
| ⊡- Synthesize General | Category: All | Reset all to default |) |
| 🖻 Place & Route | Label 🔦 | Value | |
| General | Generate SDF File | False | |
| Dual-Purpose Pin | Generate Constraint File of Ports | False | |
| BitStream | Generate IBIS File | False | |
| | Generate Post-Place File | True | |
| | Generate Post-PNR Simulation Model File | False | |
| | Initialize Primitives | False | |
| | Show All Warnings | False | |
| | Generate Plain Text Timing Report | False | |
| | Run Timing Driven | True | |
| | Use SCF | False | |
| | Not place register to IOB | False 💌 | |
| | Generate Post-Place File. Default: *.posp | | |
| | | pply 🔽 Cancel 🖉 OK | |

3.5.3 Dual-Purpose Pin 配置

如果 Gowin_EMPU_M1 使用片外 SPI-Flash 下载启动方式,复用 SSPI 和 MSPI 端口为通用端口,如图 3-29 所示,否则不需要配置端口复用。

| ** | Configurations × |
|------------------------------------|--|
| | Dual-Purpose Pin |
| Synthesize General General General | Use JTAG as regular IO ✓ Use SSPI as regular IO ✓ Use MSPI as regular IO |
| Dual-Purpose Pin BitStream | Use READY as regular IO Use DONE as regular IO Use RECONFIG_N as regular IO Use MODE as regular IO |
| | Apply Scancel |

图 3-29 Dual-Purpose Pin 配置

3.6 综合

运行 GOWIN FPGA Designer 的综合工具 Synplify_Pro 或 GowinSynthesis,完成 RTL 设计的综合,如图 3-30 图所示。

图 3-30 综合



综合工具的使用方法请参考 SUG100, Gowin 云源软件用户指南。

3.7 布局布线

运行 GOWIN FPGA Designer 的布局布线工具 Place & Route,完成布局布线和生成码流文件,如图 3-31 所示。

图 3-31 Place & Route

| 🗱GOWIN FPGA Designer - [/home/liukai/m1/debug/gowin_empu_m1/impl/pnr/gowin_empu_m1.rpt.html] | | | | | |
|---|--|-------------------------------|--|---|--|
| File Edit Tools Window Help | , , | | | 8 | |
| 🗋 📂 🗄 🧌 🔒 🛌 | a 🗏 🗅 🛍 📓 🗲 👶 | | | | |
| Process Ø 🗵 | | | | | |
| – 📔 Design Summary | | | | | |
| 🕀 📝 User Constraints | PnR Messages | | PnR Messages | | |
| - 🛃 FloorPlanner | PnR Details | | | | |
| Timing Constraints Editor | Placer | Report Title | Gowin PnR Report | | |
| Synthesize | Resource Usage Summary | Design File | /home/liukai/m1/debug/gowin_empu_m1/impl/synplify/rev_1/gowin_empu_m1.vm | | |
| Synthesis Report | I/O Bank Usage Summary | Physical Constraints File | /home/liukai/m1/debug/gowin_empu_m1/src/gowin_empu_m1.cst | | |
| Place & Route | Router | Timing Constraints File | | | |
| Place & Route Report | Global Clock Usage Summary | GOWIN Version | v1.9.0Beta | _ | |
| - 📄 Timing Analysis Report | Global Clock Signals | Part Number | GW2A-LV18PG256C8//7 | - | |
| - 📄 Ports & Pins Report | Pinout by Port Name | Created Time | Fri Feh 1 14:33:12 2019 | - | |
| Power Analysis Report | All Package Pins | Legal Announcement | Convright (C)2014-2019 Gowin Semiconductor Compration All rights reserved | - | |
| | Memory Usage | Lugar Annoarcement | oop night to zo zie zo zo oomin annihontaaton oo portugen. He nights reas voor | | |
| | Henory usage | | | | |
| | | | PnR Details | | |
| | | Placer: | | | |
| | | Hacen | | | |
| | | Starting Placer: | | | |
| | | Placement Phase 0 REAL tin | ne: 9.71 secs | | |
| | | Placement Phase 1 REAL tin | ne: 6.84 secs | | |
| | | Placement Phase 2 REAL tin | ne: 152.55 secs | | |
| Design Process | 💡 Start Page 💥 | Design Summary | 💥 🔽 gowin_empu_m1.rpt.html 💢 | | |
| Output | | | | | |
| Info (FS0001) : Bitstream g | eneration in progress | | | | |
| Info (FS002) : Bistream generation completed. | | | | | |
| Info (CM0008) : Generate '/ | into (FW001); Fower analysis completed. Info (CM008): Generate '/home'lukal/ml/debug/gowin.empu.ml/impl/pnr/gowin.empu.ml.power.html' file completed. | | | | |
| Info (CM0008) : Generate '/ | home/liukai/ml/debug/gowin_empu_ml/impl/pn | /gowin_empu_ml.pin.html' file | completed. | | |
| Info (CM008) : Generate '/home/liukai/mi/debug/gowin_empu_mi/impl/pur/gowin_empu_mi.rpt.html' file completed. | | | | | |
| Info (CM0008) : Generate '/ | home/liukai/m1/debug/gowin_empu_m1/impl/pn | /gowin_empu_ml.tr.html' file | completed. | | |
| Info (CM0001) : Fri Feb 1 | 14:33:14 2019 | | | | |
| | | | | | |

布局布线工具使用方法请参考 SUG100, Gowin 云源软件用户指南。



Gowin_EMPU_M1 提供 Debug 和 No Debug 参考设计:

- Gowin_EMPU_M1\ref_design\FPGA_RefDesign\Debug_RefDesign
- Gowin_EMPU_M1\ref_design\FPGA_RefDesign\NoDebug_RefDesign

