



APB-GPIO

2024.08.23

一 修订

版本	日期	编辑人	内容
1.00	2024.08.23	陈家耀	创建了第一个正式版本

二 简介和特性

APB-GPIO 是一个带有 APB 从接口的 GPIO 控制器,可直接挂载在 APB 总线上充当 GPIO 外设。本 IP 简单易用、资源消耗少,具有以下特性:

- 1~32 通道 GPIO 输入/输出,三态门方向运行时可配置
- 可启用 GPIO 输入中断(捕获上升/下降沿)
- 支持 GPIO 写电平掩码

APB-GPIO 的组成如图 2-1 所示,它由 **APB 寄存器接口**、**GPIO 控制**、**中断控制**三部分组成。**GPIO 三态门**的输入来自输出寄存器,方向控制来自方向寄存器,输出经过两级同步器送入 APB 寄存器接口和 GPIO 输入中断检测逻辑。**GPIO 写电平掩码**可用于改变指定通道的 GPIO 输出电平而不影响其他位,与传统的先读后写的 GPIO 输出电平的配置方式相比,提高了写电平效率和灵活性。

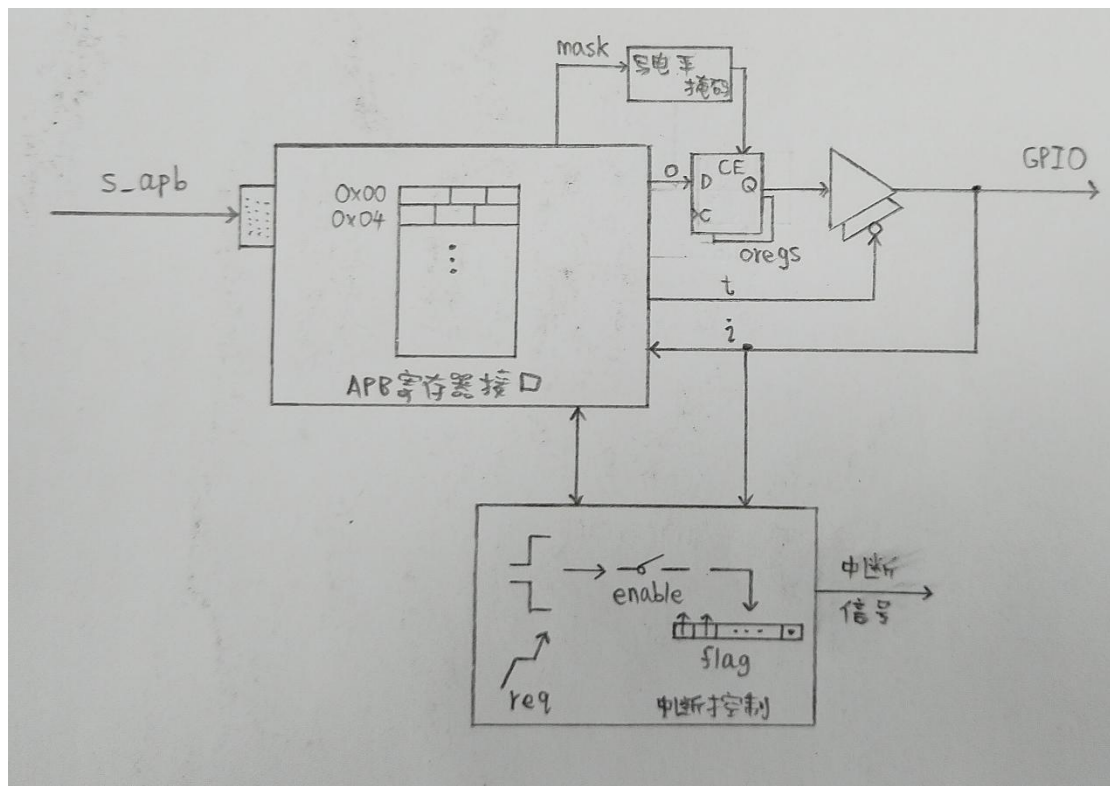


图 2-1 APB-GPIO 组成框图

三 IP 功能

APB-GPIO 是通用的 GPIO 外设，能够控制 GPIO 三态门，可启用 GPIO 输入中断。其功能描述如下：

- 1、**获取 GPIO 输入电平**。从 GPIO 三态门的输出端获取电平值，并传输到 APB 寄存器接口。
- 2、**设置 GPIO 输出电平**。根据**写电平掩码**，改变 GPIO 输出寄存器上指定位的电平值，进而影响 GPIO 三态门的输入。
- 3、**设置 GPIO 三态门的方向**。GPIO 三态门的方向可通过 APB 寄存器接口进行配置。需要注意的是，0 表示输出，1 表示输入。
- 4、**可启用的 GPIO 输入中断**。配置中断使能向量以启用对应通道的 GPIO 输入中断，读取中断标志向量以查询各个通道的中断状态，GPIO 输入中断所捕获的极性（上升沿或下降沿）可通过 IP 可配置参数进行设置。

四 IO 描述

表 4-1 APB-GPIO IO 表

端口名	方向	位宽	含义
时钟和复位			
clk	input	1	时钟
resetn	input	1	复位，低有效
APB 从接口			
paddr	input	32	APB 从机地址
psel	input	1	APB 从机片选
penable	input	1	APB 从机传输使能
pwrite	input	1	APB 从机读写类型
pwdata	input	32	APB 从机写数据
pready_out	output	1	APB 从机传输完成，固定为 1
prdata_out	output	32	APB 从机读数据
pslverr_out	output	1	APB 从机传输错误，固定为 0
GPIO 接口			
gpio_o	output	gpio_width	GPIO 输出
gpio_t	output	gpio_width	GPIO 方向，0 表示输出，1 表示输入
gpio_i	input	gpio_width	GPIO 输入
中断信号			
gpio_itr	output	1	GPIO 外设中断请求

五 可配置参数描述

表 5-1 APB-GPIO 可配置参数表

配置参数名	含义	可取值
gpio_width	GPIO 通道数	1~32
gpio_dire	GPIO 方向	"inout" "input" "output"
default_output_value	GPIO 默认输出电平，复位时的输出电平	32 位二进制向量
default_tri_value	GPIO 默认方向，复位时的三态门方向，仅在 inout 模式下可用，0 表示输出，1 表示输入	32 位二进制向量
en_itr	是否使能 GPIO 中断	"true" "false"
itr_edge	GPIO 输入中断所捕获的极性	"pos" "neg"
simulation_delay	仿真延时，可用于仿真时模拟 D 到 Q 延迟	0.1f~100.0f

六 应用指南

6.1 RTL 设计指南

APB-GPIO 是标准的 APB 外设，请将 APB-GPIO 挂载在 APB 总线上使用，典型情况是挂载在 AXI-APB 桥或 AHB-APB 桥上作为一个 APB 从机，如图 6-1-1 所示。关于 AXI-APB 桥或 AHB-APB 桥，请参见 UG200。

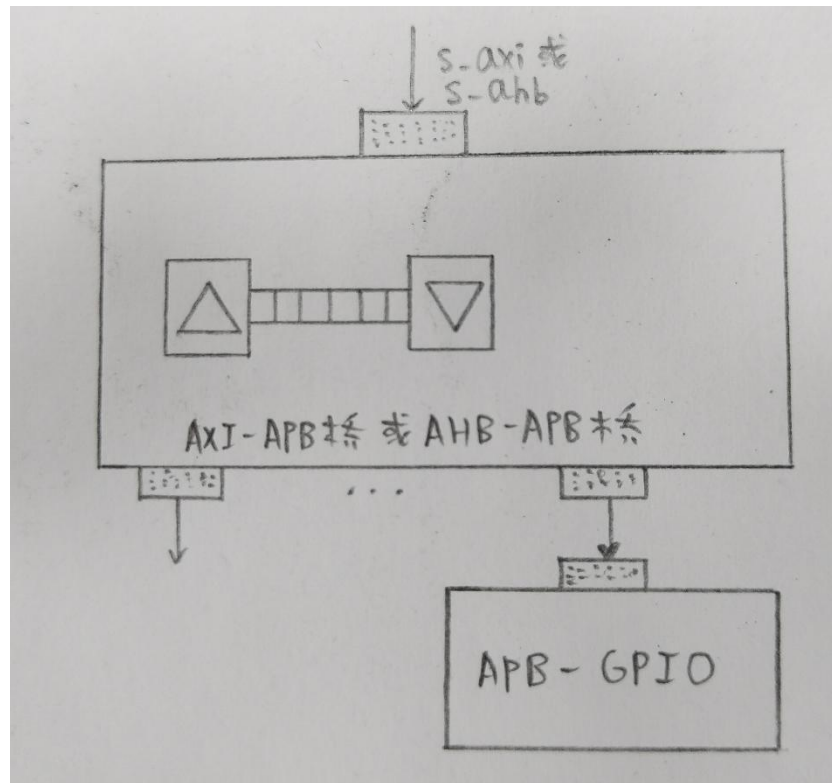


图 6-1-1 APB-GPIO 应用图

6.2 软件编程指南

6.2.1 软件驱动 API

1、类型定义

- ApbGPIO 结构体（APB-GPIO 外设结构体）

ApbGPIOHd* hardware: APB-GPIO 寄存器接口结构体指针，映射到 GPIO 外设的寄存器接口

uint32_t now_dire: 当前的 GPIO 方向，0 表示输出，1 表示输入

- ApbGPIOHd 结构体（APB-GPIO 寄存器接口结构体）

uint32_t out_v: APB-GPIO 寄存器接口偏移地址 0x00，GPIO 输出

uint32_t out_mask: APB-GPIO 寄存器接口偏移地址 0x04，GPIO 写电平掩码

uint32_t dire: APB-GPIO 寄存器接口偏移地址 0x08，GPIO 方向，0 表示输出，1 表示输入

uint32_t in_v: APB-GPIO 寄存器接口偏移地址 0x0C, GPIO 输入
uint32_t itr_global: APB-GPIO 寄存器接口偏移地址 0x10, 全局中断使能, 全局中断标志
uint32_t itr_status: APB-GPIO 寄存器接口偏移地址 0x14, 中断状态
uint32_t itr_en: APB-GPIO 寄存器接口偏移地址 0x18, 中断使能

2、函数

- **void apb_gpio_init(ApbGPIO* apb_gpio, uint32_t base_addr);**
简介: 初始化 APB-GPIO
参数: **apb_gpio** APB-GPIO (结构体指针)
base_addr APB-GPIO 外设基地址
返回值: 无
- **void apb_gpio_write_pin(ApbGPIO* apb_gpio, uint32_t mask, uint32_t value);**
简介: APB-GPIO 写电平, 根据掩码向指定的 GPIO 通道写电平
参数: **apb_gpio** APB-GPIO (结构体指针)
mask 写电平掩码
value 待写电平值
返回值: 无
- **uint32_t apb_gpio_read_pin(ApbGPIO* apb_gpio);**
简介: APB-GPIO 读电平
参数: **apb_gpio** APB-GPIO (结构体指针)
返回值: 读取到的电平值
- **void apb_gpio_set_direction(ApbGPIO* apb_gpio, uint32_t dire);**
简介: APB-GPIO 设置三态门方向
参数: **apb_gpio** APB-GPIO (结构体指针)
dire 方向向量 (0 表示输出, 1 表示输入)
返回值: 无
- **void apb_gpio_enable_itr(ApbGPIO* apb_gpio, uint32_t itr_en);**
简介: APB-GPIO 使能中断
参数: **apb_gpio** APB-GPIO (结构体指针)
itr_en 中断使能向量
返回值: 无
- **void apb_gpio_disable_itr(ApbGPIO* apb_gpio);**
简介: APB-GPIO 除能中断
参数: **apb_gpio** APB-GPIO (结构体指针)
返回值: 无
- **uint32_t apb_gpio_get_itr_status(ApbGPIO* apb_gpio);**
简介: APB-GPIO 获取中断状态

参数: **apb_gpio** APB-GPIO (结构体指针)

返回值: 中断状态向量

• **void apb_gpio_clear_itr_flag(ApbGPIO* apb_gpio);**

简介: APB-GPIO 清除中断标志

参数: **apb_gpio** APB-GPIO (结构体指针)

返回值: 无

说明: 请在 GPIO 输入中断服务函数中调用本函数清除中断标志

6.2.2 软件编程示例

1、GPIO 输出

本示例基于 APB-GPIO 实现了流水灯。

```
1.  /**
2.  * APB-GPIO 示例代码
3.  * @brief 基于 APB-GPIO 实现流水灯
4.  * 流水灯绑定在 GPIO 外设的 0~5 号通道
5.  * @attention 请根据硬件平台更换与延迟(delay)相关的 API
6.  * @date 2024/08/23
7.  * @author 陈家耀
8.  * @idt 2024/08/23 1.00 创建了第一个正式版本
9.  */
10.
11. #include "../apb_gpio.h"
12. #include "delay.h"
13.
14. //////////////////////////////////////
15.
16. #define APB_GPIO_BASEADDR 0x40000000 // APB-GPIO 外设基地址
17.
18. //////////////////////////////////////
19.
20. const static uint8_t flow_led_out_value[7] = {0x20, 0x10, 0x08, 0x04, 0x02, 0x01, 0x00}; // 流水灯样
    式
21.
22. static ApbGPIO gpio; // APB-GPIO 外设结构体
23. static uint8_t flow_led_stage_cnt = 0; // 流水灯阶段计数器
24.
25. //////////////////////////////////////
26.
27. void apb_gpio_flow_led_example(void){
28.     apb_gpio_init(&gpio, APB_GPIO_BASEADDR); // 初始化 APB-GPIO
```

```

29.
30.     apb_gpio_set_direction(&gpio, 0xFFFFFC0); // 设置 0~5 号通道方向为输出
31.     apb_gpio_write_pin(&gpio, 0x0000003F, 0x00000000); // 设置 0~5 号通道输出低电平
32.
33.     while(1){
34.         apb_gpio_write_pin(&gpio, 0x0000003F, (uint32_t)flow_led_out_value[flow_led_stage_cnt]); // 输出
流水灯电平
35.
36.         // 更新流水灯阶段计数器
37.         if(flow_led_stage_cnt == 6){
38.             flow_led_stage_cnt = 0;
39.         }else{
40.             flow_led_stage_cnt++;
41.         }
42.
43.         delay_ms(500); // 延迟 0.5s
44.     }
45. }
46.

```

2、GPIO 输入中断

本示例启用了 GPIO 输入中断，当检测到 0 号通道的上升/下降沿（取决于 IP 参数配置）时，翻转 1 号通道的输出电平。

```

1.     /*****
*****
2.     APB-GPIO 示例代码
3.     @brief GPIO 输入中断示例
4.     @attention 请根据硬件平台更换与全局中断控制器相关的 API
5.     @date 2024/08/23
6.     @author 陈家耀
7.     @eidt 2024/08/23 1.00 创建了第一个正式版本
8.     *****/
*****/
9.
10.    #include "../apb_gpio.h"
11.
12.    #include "CMSDK_CM0.h"
13.
14.    //////////////////////////////////////
15.
16.    #define APB_GPIO_BASEADDR 0x40000000 // APB-GPIO 外设基地址
17.
18.    //////////////////////////////////////

```

```

19.
20. static ApbGPIO gpio; // APB-GPIO 外设结构体
21.
22. static uint8_t led_pin_value = 0x00; // 1 号通道当前的输出电平值
23.
24. ///////////////////////////////////////////////////
25.
26. /*****
27. @itr_handler
28. @private
29. @brief APB-GPIO 输入中断服务程序(示例)
30. @param none
31. @return none
32. *****/
33. void USER_GPIO_Handler(void){
34.     // 翻转 1 号通道的输出电平
35.     led_pin_value = !led_pin_value;
36.     apb_gpio_write_pin(&gpio, 0x00000002, led_pin_value ? 0xFFFFFFFF:0x00000000);
37.
38.     apb_gpio_clear_itr_flag(&gpio); // 清除中断标志
39. }
40.
41. ///////////////////////////////////////////////////
42.
43. void apb_gpio_in_itr_toggle_example(void){
44.     apb_gpio_init(&gpio, APB_GPIO_BASEADDR); // 初始化 APB-GPIO
45.
46.     apb_gpio_set_direction(&gpio, 0xFFFFFDD); // 设置 0 号通道方向为输入, 1 号通道方向为输出
47.     apb_gpio_write_pin(&gpio, 0x00000002, 0x00000000); // 设置 1 号通道输出低电平
48.
49.     NVIC_SetPriority((IRQn_Type)3, 0x03); // NVIC 设置 3 号中断优先级
50.     NVIC_EnableIRQ((IRQn_Type)3); // NVIC 使能 3 号中断
51.
52.     apb_gpio_enable_itr(&gpio, 0x00000001); // APB-GPIO 使能 0 号通道的输入中断
53.
54.     while(1);
55. }
56.

```