

ahb_slave 模块设计

Tips 需要的工具：电脑、vivado 等仿真软件、CG 平台

一、 ahb_slave 模块

1.1 模块功能与连接关系

模块功能:

ahb_slave 模块实现了一个简单的 AHB-Lite 总线协议的从设备 (Slave) 接口。它的主要功能是:

侦听 AHB 总线事务: 监视来自 AHB 主设备 (如 CPU 或 APU) 的控制信号 (hsel, htrans, hwrite, haddr 等)。

响应选择信号: 当 hsel 信号有效时, 表示该从设备被总线主设备选中。

解析传输类型: 根据 htrans 和 hwrite 信号, 判断当前是读操作还是写操作 (主要关注 NONSEQ 非序列传输)。

地址解码 (部分): 接收 32 位的 AHB 地址 haddr, 并将其低位部分 (由参数 T_ADDR_WID 决定) 传递给内部逻辑。

数据传输:

写操作: 接收来自主设备的写数据 hwdata。

读操作: 将从内部目标读取的数据 t_rdata 输出到主设备的读数据总线 hrdata。

生成内部控制信号: 将 AHB 总线事务转换为更简单的读/写控制信号 (t_wren, t_rden) 和地址 (t_waddr, t_raddr), 用于驱动内部连接的模块。

生成 AHB 响应: 提供 hready (在本模块中始终为 1, 表示无等待状态) 和 hresp (在本模块中始终为 00, 表示 OKAY) 信号给 AHB 主设备。

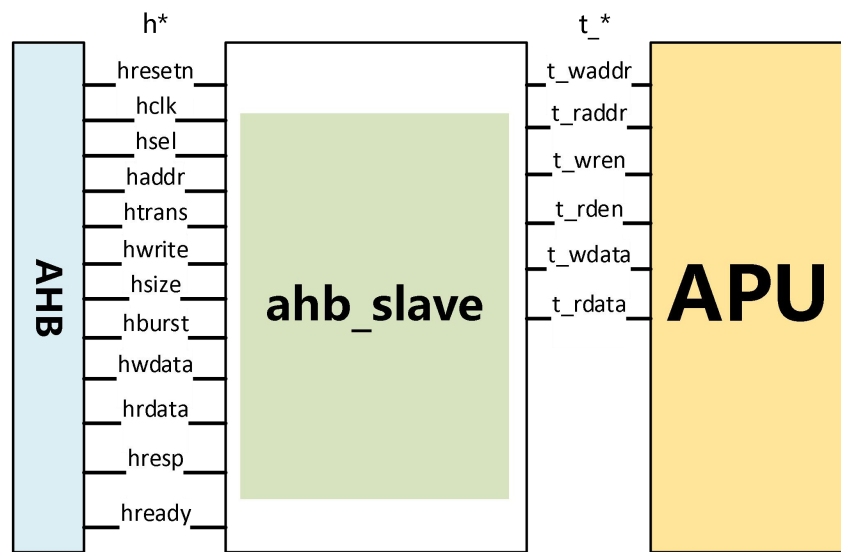
连接关系:

- AHB 总线侧: 模块的 h* 接口 (如 hclk, hresetn, hsel, haddr, htrans, hwrite, hwdata, hrdata, hready, hresp 等) 连接到系统中的 AHB 总线。
- 内部目标侧 (APU): 模块的 t_* 接口 (如 t_waddr, t_raddr, t_wren, t_rden, t_wdata, t_rdata) 连接到 APU 内部。ahb_slave 模块负责将来自 AHB 总线的读写请求, 转换成对 APU 地址空间的读写操作。

1.2 参数与接口

参数名称		默认值	参数说明
T_ADDR_WID		14	定义了连接到内部目标 (APU) 的地址总线 (t_waddr, t_raddr) 的位宽
接口名称	方向	位宽	接口说明
hresetn	input	1	AHB 总线复位信号, 低电平有效。
hclk	input	1	AHB 总线时钟信号, 上升沿触发。
hsel	input	1	AHB 从设备选择信号。高电平有效时, 表示此从设备被选中。

haddr	input	32	AHB 地址总线。提供当前访问的地址。
htrans	input	2	AHB 传输类型。2'b00=IDLE, 2'b01=BUSY, 2'b10=NONSEQ, 2'b11=SEQ。本模块主要响应 NONSEQ。
hwrite	input	1	AHB 传输方向。1'b1=写操作, 1'b0=读操作。
hsize	input	3	AHB 传输大小。指示传输的数据位宽（如 8/16/32 位）。此简单模块假定总是 32 位传输 (3'b010)。
hburst	input	3	AHB 突发传输类型。指示传输是单个传输还是突发传输。此模块只支持 SINGLE (3'b000)。
hwdata	input	32	AHB 写数据总线。主设备通过此总线写入数据。
hrdata	output	32	AHB 读数据总线。从设备通过此总线向主设备提供读取的数据。数据来源于 t_rdata。
hresp	output	2	AHB 传输响应。2'b00=OKAY, 2'b01=ERROR。此模块总是为 2'b00 (OKAY)。
hready	output	1	AHB 就绪信号。1'b1=传输可以完成, 1'b0=插入等待状态。此模块总是为 1'b1 (无等待)。
t_waddr	output	T_ADDR_WID	写地址总线。传递给内部 APU 的写操作目标地址（来自 haddr 的低位）。
t_raddr	output	T_ADDR_WID	读地址总线。传递给内部 APU 的读操作目标地址（来自 haddr 的低位）。
t_wren	output	1	写使能信号。高电平有效时，指示内部 APU 执行写操作。
t_rden	output	1	读使能信号。高电平有效时，指示内部 APU 执行读操作，并将数据放到 t_rdata 上。
t_wdata	output	32	写数据总线。将从 hwdata 接收到的数据传递给内部 APU。
t_rdata	input	32	读数据总线。从内部 APU 读取的数据，将被驱动到 hrdata 上。



1.3 模块功能

1.3.1 控制使能信号并透传

```
assign t_rden = hsel & htrans[1] & (!hwrite);  
assign t_wdata = hwddata;  
assign hrdata = t_rdata;  
assign t_raddr = XXXXXXXX? haddr[T_ADDR_WID-1:0]: 'h0; （自行添加）
```

1.3.2 向 AHB 写地址

如果在时钟上升沿，**hresetn** 为高，但 **hsel** 为低（即从设备未被选中），则 **t_waddr** 的值不会改变，保持其上一个时钟周期的值。

如果 **hsel** 为高，表示当前 AHB 主设备正在访问本从设备。此时，模块会捕获（锁存）AHB 地址总线 **haddr** 的低 **T_ADDR_WID** 位，并将其存储到 **t_waddr** 寄存器中。

1.3.3 控制 **t_wren** (内部写使能)注意阻塞赋值和非阻塞赋值的区别

检查 **wr_en** 信号的值。**wr_en** 是一个组合逻辑信号，其值为 1 当且仅当：

- **hsel** 为高（从设备被选中）。
- **htrans[1]** 为高（传输类型是 NONSEQ 或 SEQ，表示有效的传输正在进行）。
- **hwrite** 为高（当前是写操作）。

如果上述所有条件都满足，表示这是一个有效的、针对本从设备的 AHB 写传输的数据阶段的开始（因为假设 **hready** 总是为 1，地址和数据阶段可以认为是连续的），此时 **t_wren** 在下一个时钟沿被置为高电平（1'b1）。

else: 如果 **wr_en** 为低（例如，不是写操作、从设备未被选中、或者是 IDLE/BUSY 传输），则 **t_wren** 在下一个时钟沿被置为低电平（1'b0）。

其他部分 assign 语句：

```
assign hresp = 2'b00; //always ok  
assign hready = 1'b1; //no wait
```

1.4 仿真与提交

根据 **ahb_slave** 模块的工作原理，利用 Verilog HDL 语句编写 **ahb_slave.v** 模块，测试后上传 CG 平台。

二、 ahb_slave_top 模块

2.1 模块功能与连接关系

该模块相比 **ahb_slave 模块**，额外提供对更多种内部存储器（RAM）和控制/状态寄存器的访问接口。当 AHB 主设备选中 (hsel 有效) 该从设备并发起一次传输 (htrans 为 NONSEQ 或 SEQ) 时，ahb_slave_top 会根据地址 (haddr) 和写信号 (hwrite) 执行相应的操作：

写入操作 (hwrite = 1): 将主设备提供的数据 (hwddata) 写入到由地址 haddr 映射的内部 RAM 或寄存器中。

读取操作 (hwrite = 0): 从由地址 haddr 映射的内部 RAM 或寄存器中读取数据，并通过 hrdata 返回给主设备。

注意：以下模块的大部分代码已给出，请结合代码注释，在空缺处（一般标有 TODO）补全代码，并将 **function.v** 内部的两个模块和 **ahb_slave** 一起粘贴进 **ahb_slave_top.v**，再将顶层模块提交到评测机。

评测机可能要跑 3 分钟左右

ahb_slave_top 模块由三个主要的子模块实例化并连接而成，它们协同工作以实现完整的功能：

1. **ahb_slave 模块**: 直接处理 AHB 总线协议的接口层。
2. **addr_map 模块**: 地址解码器和寄存器文件。
3. **ram_mux 模块**: RAM 多路选择器和数据通路适配器。

2.2 模块代码一见云盘链接function.v与ahb_slave_top.v

2.3 仿真与提交

根据 ahb_slave_top 模块的工作原理，利用 Verilog HDL 语句补完 ahb_slave_top.v 模块，测试后上传 CG 平台。