

# 测试平台说明文档 (TB.sv)

## 1. 概述

本测试平台用于验证 `da4008_chip_top` 芯片的数字功能。通过 SPI 接口配置芯片内部寄存器，并通过 LVDS 接口输入数据帧，观察芯片输出的并行数据 (`MSB_OUT/LSB_OUT`) 以及经过后处理的波形数据。测试平台包含时钟生成、复位控制、驱动器模型、待测芯片实例化以及数据采集逻辑。

## 2. 主要模块及功能

### 2.1 时钟生成

- `clk`: 周期 1.536 ns, 用于芯片主时钟。
- `clk_40g`: 周期 0.024 ns, 用于高速数据采集 (后处理模块采样时钟)。

两个时钟均由 `clock_tb` 模块实例化产生。

### 2.2 复位信号

- `rst_n`: 低有效复位, 初始为 0, 延迟 20 ns 后置为 1。

### 2.3 配置文件读取

测试平台通过读取临时文件获取配置和数据的路径:

- `CASE_TEMP = "../..case_temp.txt"`: 存储 SPI 配置文件路径。
- `DATA_TEMP = "../..data_temp.txt"`: 存储输出数据文件路径。
- `LVDS_FILE = "../..../..case/lvds/0305/lvds.txt"`: LVDS 输入数据文件路径。

在 `initial` 块中读取这些路径, 并存入 `CONFIG_FILE` 和 `DATA_O_FILE` 字符串变量。

### 2.4 接口及驱动器

- **SPI 接口**: `spi_if` 模块实例化为 `spi_bus`, 包含 `sclk`、`mosi`、`miso`、`csn` 信号。通过 `virtual spi_if vif` 传递。
- **LVDS 接口**: `lvds_if` 模块实例化为 `lvds_bus`, 包含 `data`、`valid`、`clk` 信号。通过 `virtual lvds_if lvds_vif` 传递。
- **SPI 驱动器**: `spi_driver` 类对象 `my_drv`, 负责从 `CONFIG_FILE` 读取配置序列并驱动 SPI 总线, 同时将读取的结果写入 `DATA_O_FILE`。
- **LVDS 驱动器**: `lvds_driver` 类对象 `lvds_drv`, 负责从 `LVDS_FILE` 读取数据帧并驱动 LVDS 总线。先发送训练序列 (`train_count = 100`), 然后发送加扰数据帧。

### 2.5 测试流程

1. 初始化复位和驱动器对象。
2. 释放复位后, LVDS 驱动器发送训练序列和数据帧。
3. 首次调用 `my_drv.do_drive(file_path)` 将 SPI 配置写入芯片 (同时可能回读并保存)。
4. 拉高 `start` 信号两个周期, 启动芯片内部同步。
5. 等待 30,000 ns 后再次调用 `do_drive` 保存数据。

6. 重复步骤 4-5 两次，共三次数据采集。
7. 仿真结束 (`$finish`)。

## 2.6 待测芯片实例化

`da4008_chip_top` 例化为 `u_da4008_chip_top`，连接信号包括：

- SPI 接口： `PI_sclk`, `PI_csn`, `PI_mosi`, `PO_miso`
- 中断： `PO_irq`
- 同步信号： `PI_sync_in` (start), `PO_sync_out`
- 时钟： `clk`
- LVDS 输入： `lvds_data`, `lvds_valid`, `lvds_clk`
- DAC 输出总线： `MSB_OUT[63:0]`, `LSB_OUT[63:0]`, `MSB_DUM[63:0]`, `DEM_VLD`
- 各类配置端口（电流、时钟、P2S 等）

## 2.7 后处理模块

- **DEM\_Reverse\_64CH**：将 `MSB_OUT` 和 `LSB_OUT` 合并并反向排列，输出 64 路 8 位数据 `data_out[63:0]` 及有效信号 `vld_out`。
- 使用 `clk_40g` 将 `data_out` 和 `vld_out` 打一拍得到 `data_out_r` 和 `vld_out_r`。
- 计数器 `cnt_c`：在 `vld_out_r` 有效时递增（0~63 循环），用于轮询选择当前通道数据。
- `cs_wave`：根据 `cnt_c` 选择对应通道的 `data_out_r`，并将最高位取反（可能是为了满足特定输出格式），输出 8 位波形数据。

## 2.8 波形记录

使用 `$fsdbAutoSwitchDumpfile` 和 `$fsdbDumpvars` 记录 FSDB 格式波形，支持 MDA 转储，便于调试。

## 3. 文件依赖

- `../../rtl/define/chip_define.v`：芯片定义文件。
- `../../model/SPI_DRIVER.sv`：SPI 驱动器模型。
- `../../model/LVDS_DRIVER.sv`：LVDS 驱动器模型。
- `clock_tb` 时钟发生模块。
- `spi_if`、`lvds_if` SPI 接口文件。
- `DEM_Reverse_64CH` DEM 解码模块。

## 4. 仿真控制

通过修改 `case_temp.txt` 和 `data_temp.txt` 可指定不同的配置文件和输出文件。LVDS 数据文件路径固定为 `LVDS_FILE`，可根据需要修改。

## 5. 注意事项

- 仿真时间单位/精度为 1ns/1ps。
- 驱动器的具体实现未在此文件中给出，需确保相关模型正确。

- 多次调用 `my_drv.do_drive` 可能用于在不同时刻捕获芯片内部状态。
  - `cs_wave` 的生成方式暗示了数据采集的时序要求，需保证 `clk_40g` 与芯片输出时钟的同步关系。
- 

以上为该测试平台的结构与功能说明，可作为仿真环境的使用参考。