

## 一、算法建模

算法建模基于 matlab 软件进行代码设计，参考 fft.m 文件。

建模过程说明如下：

### 1. FFT 蝶形计算原理

基-2 DIT FFT 算法的核心是蝶形计算单元：

$$X(k) = A + W * B$$

$$X(k+N/2) = A - W * B$$

其中：

- A 和 B 是输入数据点
- W 是旋转因子（复数）
- X(k)和 X(k+N/2)是输出数据点

### 2. 建模关键步骤

#### 1. 位反转重排输入：

- 对输入序列进行位反转排序
- 例如：索引 1(00000001)变为 128(10000000)

#### 2. 分级蝶形计算：

- 对于 8 级计算 ( $\log_2 256=8$ )
- 每级计算  $256/2=128$  个蝶形运算
- 旋转因子角度： $\theta = -2\pi k/2^m$ ，其中 m 为当前级数

#### 3. 定点数处理：

- 使用 Q24.8 格式（32 位，24 位整数+8 位小数）
- 乘法结果保留双精度（64 位），然后截断为 32 位
- 模拟硬件中的量化误差

### 3. 模型验证方法

#### 1. 测试信号生成：

- 50Hz + 120Hz 正弦波复合信号
- 采样率 1000Hz

#### 2. 三种实现比较：

- MATLAB 内置 FFT（参考）
- 浮点模型 FFT
- 定点模型 FFT（Q24.8）

#### 3. 性能指标：

- 频谱幅度对比
- 误差幅度分析

建模对比结果如下：

图 1 为测试信号

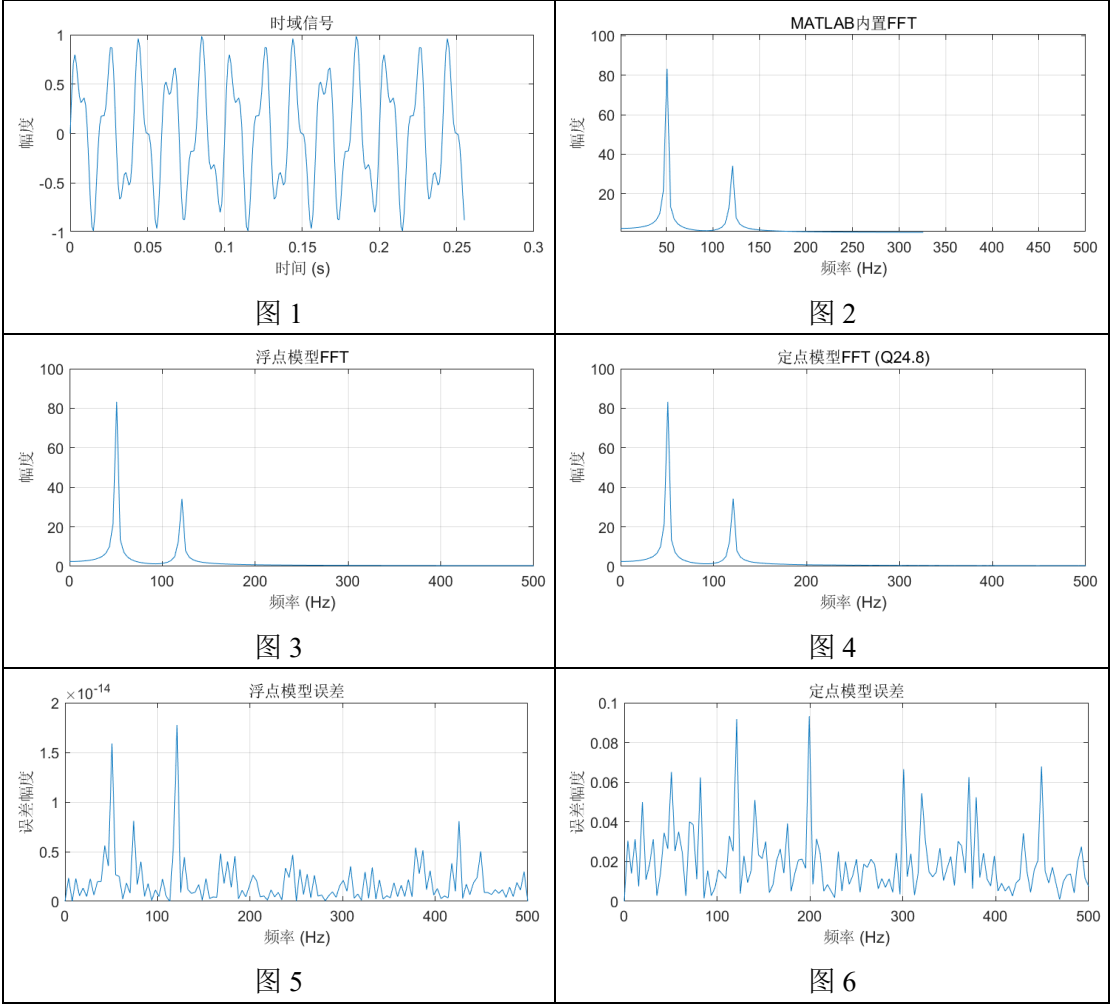
图 2 为使用 matlab 内置 fft 函数的计算结果

图 3 浮点建模计算结果

图 4 定点建模计算结果

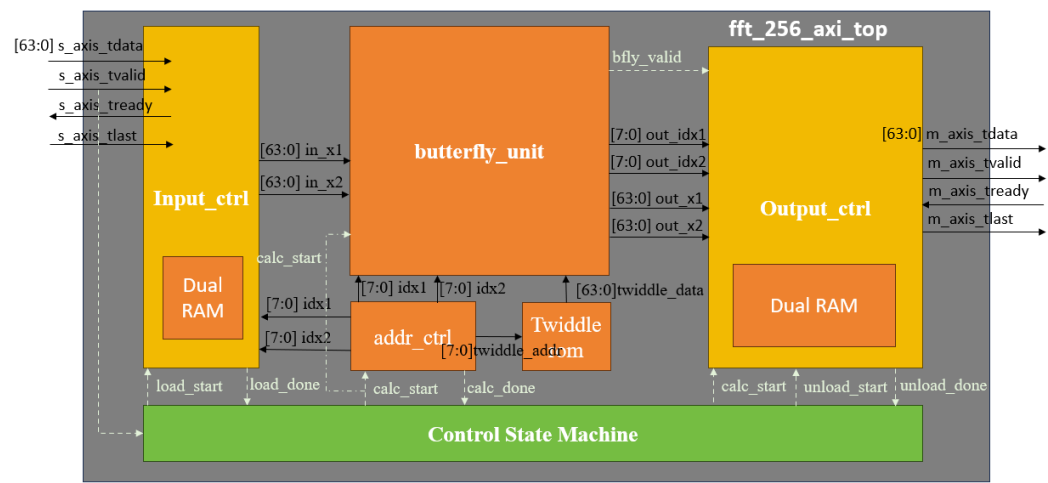
图 5 浮点模型误差

图 6 定点模型误差



二、架构设计

项目架构图

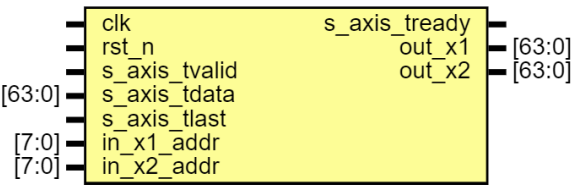


IDEL:初始状态  
LOAD: 输入数据加载中  
CALC: 蝶形计算进行中  
UNLOAD: 输出数据发送中

三、RTL 设计

input\_crtl.v

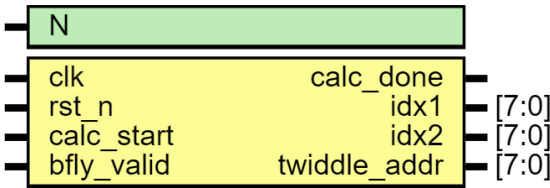
该模块负责接收 AXIS 接口的输入数据，并对其位反转后加载至双口 RAM。



Port name	Direction	Type	Description
clk	input		时钟
rst_n	input		复位
s_axis_tvalid	input		AXIS 的 Valid 信号
s_axis_tdata	input	[63:0]	AXIS 输入数据
s_axis_tlast	input		AXIS 最后一个输入数据标志信号
s_axis_tready	output		AXIS 的 Ready 信号
in_x1_addr	input	[7:0]	第一地址
in_x2_addr	input	[7:0]	第二地址
out_x1	output	[63:0]	第一数据
out_x2	output	[63:0]	第二数据

addr\_ctrl.v

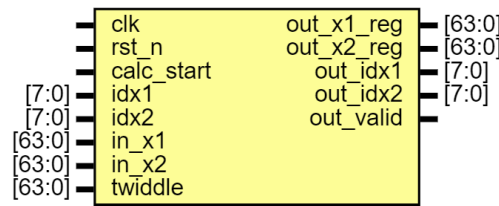
该模块为地址生成器，基于 2 步(Radix-2)索引算法寻址。同时负责 twiddle 的寻址。



Port name	Direction	Type	Description
N	Parameter		FFT 计算点数
clk	input		时钟
rst_n	input		复位
calc_start	input		计算开始标志
bfly_valid	input		计算结果有效
calc_done	output		计算结束标志
idx1	output	[7:0]	第一地址
idx2	output	[7:0]	第二地址
twiddle_addr	output	[7:0]	旋转因子地址

butterfly\_unit.v

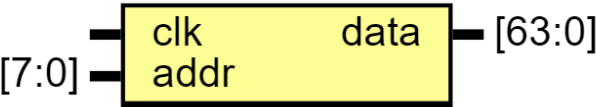
该模块负责进行蝶形计算。



Port name	Direction	Type	Description
clk	input		时钟
rst_n	input		复位
calc_start	input		计算开始标志
idx1	input	[7:0]	第一地址
idx2	input	[7:0]	第二地址
in_x1	input	[63:0]	第一数据
in_x2	input	[63:0]	第二数据
twiddle	input	[63:0]	旋转因子
out_x1_reg	output	[63:0]	计算后的第一数据
out_x2_reg	output	[63:0]	计算后的第二数据
out_idx1	output	[7:0]	输出第一地址
out_idx2	output	[7:0]	输出第二地址
out_valid	output		输出有效标志

twiddle\_rom.v

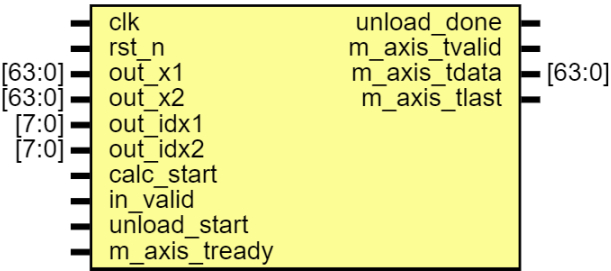
该模块负责存储旋转因子。



Port name	Direction	Type	Description
clk	input		时钟
addr	input	[7:0]	地址
data	output	[63:0]	旋转因子

output\_crtl.v

该模块负责将计算结果缓存后以 AXIS 接口形式发送。



Port name	Direction	Type	Description
clk	input		时钟
rst_n	input		复位
out_x1	input	[63:0]	计算后的第一数据
out_x2	input	[63:0]	计算后的第二数据
out_idx1	input	[7:0]	第一地址
out_idx2	input	[7:0]	第二地址
calc_start	input		计算开始标志
in_valid	input		计算数据有效
unload_start	input		发送开始标志
unload_done	output		发送完成标志
m_axis_tready	input		AXIS 的 Ready 信号
m_axis_tvalid	output		AXIS 的 Valid 信号
m_axis_tdata	output	[63:0]	AXIS 输入数据
m_axis_tlast	output		AXIS 最后一个输入数据标志信号



## 四、UVM 验证

待更新