**ldl\_top.v是算法模块的顶层，包括LDL分解，Buf，Fw，Bw等核心运算模块。**

**ldl\_top.v接口列表说明**

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **方向** | **位宽** | **说明** |
| clk | input | 1 | 工作时钟 |
| rst | input | 1 | 复位 |
| col\_num | input | 11 | 矩阵维数 |
| eps | input | 64 | Eps数据（修正LDL结果） |
| delta | input | 64 | Delta数据（修正LDL结果） |
| init\_l\_col\_cnt\_en | input | 1 | 初始化L矩阵列个数积分使能 |
| init\_l\_col\_cnt\_accu | input | 16 | 初始化L矩阵列个数积分数据 |
| init\_l\_row\_cnt\_en | input | 1 | 初始化L矩阵行个数积分使能 |
| init\_l\_row\_cnt\_accu | input | 16 | 初始化L矩阵行个数积分数据 |
| init\_l\_wr\_en | input | 1 | L矩阵输入使能 |
| init\_l\_wr\_col | input | 11 | L矩阵输入列号（0~col\_num-1） |
| init\_l\_wr\_row | input | 11 | L矩阵输入行号（0~col\_num-1） |
| init\_l\_wr\_dat | input | 64 | L矩阵输入数据 |
| a\_wr\_en | input | 1 | A矩阵输入使能 |
| a\_wr\_col | input | 11 | A矩阵输入列号（0~col\_num-1） |
| a\_wr\_row | input | 11 | A矩阵输入行号（0~col\_num-1） |
| a\_wr\_dat | input | 64 | A矩阵输入数据 |
| sign\_en | input | 1 | Sign输入标志 |
| sign\_dat | input | 32 | Sign输入数据 |
| b\_flag | input | 1 | B向量标志 |
| b0\_wr\_en | input | 1 | B0向量输入使能 |
| b0\_wr\_dat | input | 64 | B0向量输入数据 |
| b1\_wr\_en | input | 1 | B1向量输入使能 |
| b1\_wr\_dat | input | 64 | B1向量输入数据 |
| x0\_vout | output | 1 | X0向量输出有效标志 |
| x0\_dout | output | 64 | X0向量输出数据 |
| x1\_vout | output | 1 | X1向量输出有效标志 |
| x1\_dout | output | 64 | X1向量输出数据 |

**接口时序描述**

1. 配置参数col\_num、eps、delta

|  |  |  |  |
| --- | --- | --- | --- |
| col\_num | input | 11 | 矩阵维数 |
| eps | input | 64 | Eps数据（修正LDL结果） |
| delta | input | 64 | Delta数据（修正LDL结果） |

所有输入开始前，必须先正确配置col\_num的数值，在运算期间一直保持。

1. 输入Ｌ矩阵行列个数积分信息

|  |  |  |  |
| --- | --- | --- | --- |
| init\_l\_col\_cnt\_en | input | 1 | 初始化L矩阵列个数积分使能 |
| init\_l\_col\_cnt\_accu | input | 16 | 初始化L矩阵列个数积分数据 |
| init\_l\_row\_cnt\_en | input | 1 | 初始化L矩阵行个数积分使能 |
| init\_l\_row\_cnt\_accu | input | 16 | 初始化L矩阵行个数积分数据 |

连续输入col\_num+1个周期，第一个积分数据为0，而且行列个数统计时不包含对角线元素。

1. 输入Sign信息

|  |  |  |  |
| --- | --- | --- | --- |
| sign\_en | input | 1 | Sign输入标志 |
| sign\_dat | input | 32 | Sign输入数据 |

连续输入col\_num个周期

1. 输入A矩阵数据

|  |  |  |  |
| --- | --- | --- | --- |
| a\_wr\_en | input | 1 | A矩阵输入使能 |
| a\_wr\_col | input | 11 | A矩阵输入列号（0~col\_num-1） |
| a\_wr\_row | input | 11 | A矩阵输入行号（0~col\_num-1） |
| a\_wr\_dat | input | 64 | A矩阵输入数据 |

按列输入，每列输入之间需要有间隔，列内连续；行列编号采用0~col\_num-1模式，第一行或第一列的编号为0，依次类推。

1. 配置b\_flag,并输入B向量

|  |  |  |  |
| --- | --- | --- | --- |
| b\_flag | input | 1 | B向量标志 |
| b0\_wr\_en | input | 1 | B0向量输入使能 |
| b0\_wr\_dat | input | 64 | B0向量输入数据 |
| b1\_wr\_en | input | 1 | B1向量输入使能 |
| b1\_wr\_dat | input | 64 | B1向量输入数据 |

b\_flag=1表示同时计算2组b向量（b0、b1均有效），b\_flag=0表示仅计算b0向量，b\_flag在运算期间保持；b0、b1向量连续输入col\_num个周期。

1. 输出运算结果X0、X1

|  |  |  |  |
| --- | --- | --- | --- |
| x0\_vout | output | 1 | X0向量输出有效标志 |
| x0\_dout | output | 64 | X0向量输出数据 |
| x1\_vout | output | 1 | X1向量输出有效标志 |
| x1\_dout | output | 64 | X1向量输出数据 |

运算结束后，输出x0及x1（b\_flag=1时x1才会输出），x0、x1向量连续输出col\_num个周期。

1. 下一次迭代

可以迭代返回到5 （再次输入B向量后，开始运算）

可以迭代返回到3/4（再次输入A矩阵及B向量后，开始运算）

可以迭代返回到2 （更新L矩阵积分、A矩阵、B向量后，开始运算）

可以迭代返回到1 （更改矩阵维数、更新L矩阵积分、A矩阵、B向量后，开始运算）

**时间估计**

时钟：240Mhz

数据：给定的757维测试数据

A矩阵输入时间：17.58us

B向量输入时间：3.16us

LDL计算时间（A矩阵输入完成到LDL分解完成）：167.45us

FW计算时间（包含L矩阵输入时间，包含x矩阵输出时间）：58us

BW计算时间（不含L矩阵输入时间、不含b矩阵输入时间，包含x矩阵输出时间）：65us

为方便推算，可以将上述时间分为2组：

1. LDL分解时间：包含A矩阵输入时间+LDL计算时间，约185.03us
2. 线性方程LDLX=B求解时间：包含FW计算时间+BW计算时间，约123us

PL侧RTL这边剩余主要工作（2019/08/26）：

1.LDL\_TOP算法模块需要对分解结果增加Sign修正运算

2.LDL\_TOP算法模块对小于757维数矩阵的仿真验证（需要提供几组数据）

3.FW&BW运行时间优化（考虑能否优化，目前没有想到好方法）

4.PS->PL协议帧到LDL\_Top算法输入接口时序的转换（Intf2Core.v）

5.LDL\_Top算法输出接口时序到PL->PS协议帧的转换（Core2Intf.v）

