HPL算法模拟主要流程：

注1：由于HPL引入了MPI并行操作，CPU多个核心之间通信过程复杂，通信时间暂时没有考虑，**因此本流程仅考虑单CPU，单核，单GPU的HPL执行过程。**

注2：通过对比CPU版本的HPL与NV版本的HPL源码发现NV版本实现CPU与GPU计算负载均衡的方式是重新编译了HPL\_dgemm和HPL\_dtrsm函数，主要由cuda\_dgemm来实现自动负载均衡，因此在模拟HPL计算时，我们认为执行HPL\_dgemm的时候，我们考虑该函数由cpu\_dgemm和gpu\_dgemm两个函数构成，并将会根据CPU和GPU计算力自动进行负载均衡分别调用cpu\_dgemm和gpu\_dgemm。

注3：系统主要耗时主要由dtrsm、dgemm以及CPU和GPU之间的通信开销构成，本模型中尚未考虑其他因素造成的耗时。

1.读取参数

2.根据参数初始化增广矩阵A|b

3.计时开始

4.循环分解并求解增广阵A|b

循环i开始

4.1.HPL\_pdpanel\_pdfact() //分解A|b耗时t1i，对于分解耗时实际没有太多，且计算起来较为复杂，是否可以暂时考虑不计

4.2HPL\_pdupdate() //求解更新A12,A21,A22

4.2.1.HPL\_dtrsm() //求解A12，A21，耗时max(tcdti, tgdti) + t2i + t3i

4.2.1.1.根据CPU与GPU运算能力进行对需要进行乘加运算的矩阵进行split

4.2.1.2.将split后需要传输到GPU显存上的部分矩阵传输到GPU显存上 //耗时t2i

4.2.1.3.CPU执行cpu\_ dtrsm()耗时tcdti，GPU执行gpu\_ dtrsm()耗时tgdti

4.2.1.4.GPU将运算完成的结果传输回内存耗时t3i

4.2.2.HPL\_dgemm() //对A22执行矩阵乘加运算，耗时max(tcdgi, tgdgi) + t4i + t5i，此部分为主要耗时的地方，占整个流程约94%的时间

4.2.2.1.根据CPU与GPU运算能力进行对需要进行乘加运算的矩阵进行split

4.2.2.2.将split后需要传输到GPU显存上的部分矩阵传输到GPU显存上 //耗时t4i

4.2.2.3.CPU执行cpu\_dgemm()耗时tcdgi，GPU执行gpu\_dgemm()耗时tgdgi

4.2.2.4.GPU将运算完成的结果传输回内存耗时t5i

5.循环结束，HPL主要耗时部分计算完毕，剩余一些其他操作比如对内存空间的释放等暂时不考虑耗时

6.计时结束，即

7.根据总计算时间计算出系统的浮点运算速率

以上是粗略估计HPL总体计算时间的流程图

