基本概念：

静态分支预测：基于编译器的编译信息对分支指令进行预测，预测信息不再改变

动态分支预测：依据程序运行的实时信息，不断对预测信息进行更新，具有较高的预测准确率

A：基于局部历史的分支预测机制：该预测机制利用局部历史寄存器作为索引信息，利用分支指令前几次执行的结果与此次跳转方向的相关性

B：全局预测分支机制：利用前几条分支指令的跳转信息来索引一个2位的分支预测表

C：组合分支预测机制：该机制将不同的分支指令信息组合起来，决定分支指令的预测结果

BHR（Branch History Register）:用来存储最近k个分支指令的预测结果

PHT（Pattern History Table）：

动态分支预测是近来的处理器已经尝试采用的的技术。最简单的动态分支预测策略是分支预测缓冲区（Branch Prediction Buff)或分支历史表(branch history table)。

BHT——Branch History Table，顾名思义，这是记录分支历史信息的表格，用于判定一条分支指令是否token；这儿记录的是跳转信息，简单点的，可以用1bit位记录，例如1表示跳转，0表示不跳转，而这个表格的索引是指令PC值；考虑在32位系统中，如果要记录完整32位的branch history，则需要4Gbit的存储器，这超出了系统提供的硬件支持能力；所以一般就用指令的后12位作为BHT表格的索引，这样用4Kbit的一个表格，就可以记录branch history了。当然，通过大伙的不懈努力和分析，发现在BHT中用1bit位记录分支是否跳转还不够准确，用2bit位记录就非常好了，而用3bit或者更多位记录，效果与2bit类似。所以在BHT中，一般就用2bit位记录分支是否跳转：例如11和10表示这条分支会跳转；01和00表示分支不会跳转。这个2bit计数器大伙叫做饱和计数器。

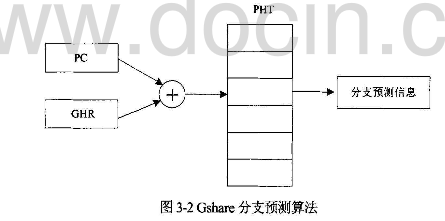
BTB——用于记录一条分支指令的跳转地址，由于这儿存储的是指令地址，例如32位地址，因此，这个表格就不能做到存储BHT那样多的内容了，如果也支持4K条指令，则需要128Kbit的存储空间，这几乎可以赶上一个L1Cache的容量了，所以BTB一般很小，就32项或者64项。由于这个BTB容量小，并且其用于是记录分支指令的跳转地址，因此，如果这条指令不跳转，即其下一条指令就是PC+4，则不会在BTB中记录的。

基于BTB和BHT的分支预测就很简单了：

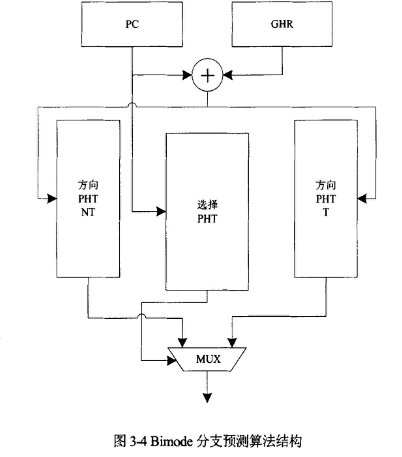
1）在取指阶段利用PC寻址BTB，如果命中，则说明这是一条跳转指令，利用从BTB中获取到的地址去取icache；

2）由于BTB中保存的内容不够多，因此BHT的准确率更高，这个时候索引BHT表格，如果发现BHT也跳转，则说明这条指令预测是跳转的；如果BHT不跳转，则说明不跳转，这个时候就取消BTB中的指令地址，重新PC+4去取icache；

预测分支指令主要依据三个方面：分支指令的存储地址，与分支指令相关的存储结果（存储在全局历史寄存器中GHR），当前分支指令前几次执行的结果（局部历史寄存器）



将PC（程序计数器）和GHR（全局历史寄存器）进行异或，PC和GHR采用的数据位数决定了历史模式表的存储容量，若PC选取N位信息用于索引，同样GHR也采用N位信息，并且PHT每个表项的数据为2位，那么PHT的存储容量为2\*2的N次方bit

Bi-mode 分支预测

将PHT分成两层，第一层为选择PHT，第二层为方向PHT。选择PHT中的数据用于对两个方向PHT进行选择，方向PHT中的信息用于预测分支指令的跳转方向。选择PHT只有PC的值进行索引，而方向PHT的有PC和GHR的数据信息进行索引。

当预取到分支指令时，先获取分支指令的地址信息和全局分支指令执行的历史信息，将这两者信息进行异或操作，得到的操作结果共同索引两个方向PHT，一个是跳转（taken T）PHT，一个是不跳转（not taken NT）PHT.然后访问两个方向PHT，将得到分支指令的两个跳转预测信息。最后，该条分支指令的跳转预测信息由选择PHT进行仲裁。利用分支指令的地址对选择PHT进行索引，选择PHT的表项与方向PHT相同，表项中的数据信息代表分支指令最近使用哪个方向PHT对分支指令进行预测。

当对分支指令预测错误时，需要对Bi\_mode分支预测算法中的信息进行更新。选择PHT需要根据分支执行的结果进行实时更新。如果分支指令执行结果为跳转时，对选择PHT中的数据向跳转状态转变，保证选择PHT中的信息能够选择T PHT。如果分支指令执行结果为不跳转时，对选择PHT中的数据向不跳转状态转变，保证选择PHT中的信息进行选择NT PHT。当选择PHT选择的方向与分支指令执行的结果相反，但方向PHT对分支指令预测正确时，对选择PHT不进行更新。当方向PHT对分支指令预测错误时，要无条件的对其更新。