

補足：仮想記憶と TLB

最近の OS は、複数のジョブを並行して走らせることが要求される。複数のジョブをきちんと管理するためには、それぞれのジョブの動作するアドレス空間が独立している必要がある。また、複数のジョブの利用するアドレス空間が実際の物理メモリを越えても実行できるようにしておく必要がある。

仮想記憶 (Virtual Memory) は、このために導入された機構で、論理アドレス (Logical Address: または仮想アドレス: Virtual Address) と、物理アドレス (Physical Address) を分離する方式である。すべてのプログラムは、独立な論理アドレス上で動作するように見えるが、実際に動作する物理的なアドレスは OS の管理下で別に割り当ててある。この機構を実現するためには、CPU でアドレスを出した直後に論理アドレスから物理アドレスへ変換する必要がある。キャッシュは通常物理アドレスで参照するため (論理アドレスで参照する Virtual Cache という方法もあるが、同一論理アドレスが複数の物理アドレスに割り当てられている場合の対処が難しい等の問題がある)、この作業は、キャッシュの前に行う必要があり、高速性が重要となるため、TLB (Translation Lookaside Buffer) と呼ばれるハードウェアの変換テーブルを装備する。

論理アドレスと物理アドレスの変換の単位はページといってキャッシュブロックよりは、はるかに大きな単位 (4KB-16KB) で行う。論理アドレスと物理アドレスの対応テーブルは、OS によってハッシュなどの圧縮手法を使って管理されており、TLB にはこのうちのごく一部、現在アクセスされているページのみが一種のキャッシュとして格納されている。

TLB は図 1 に示すように、フルアソシアティブキャッシュと同様の方法で、論理アドレスを物理アドレスに変換する。ページはキャッシュブロックよりも大きいので、局所性の原則から、さほどに容量が大きくなっても TLB はミスヒットしないが、競合が起きるとミスヒットが増えることから小規模のフルアソシアティブ方式が用いられることが多く、32 エントリ程度を装備する (ここでは 8 しか描いてない)。場合によっては連想メモリ (Content Addressable Memory) を用いる場合もある。

TLB は高速であるものの、変換してから、そのアドレスをキャッシュに送るのでは、1 クロックでデータを返すことは難しい。そこで、小規模なプロセッサの多くでは、ページ内アドレスに相当する部分をキャッシュのインデックスとして用いる。ページ内アドレスは TLB の変換の対象にならないことから、物理アドレスへの変換と同時にキャッシュをアクセスすることができる。残念なことに、この方法では、インデックスの長さがページサイズによって制限される。今、ページサイズを 4Kbyte とし、ラインサイズを 32byte とすると、インデックスは、7bit しか取れないため、キャッシュサイズはダイレクトマップならば 4Kbyte、2-way セットアソシアティブならば 8Kbyte、4-way セットアソシアティブならば 16Kbyte ということになる。これは小規模だが、L1 キャッシュとしては十分な量なので、多くのシステムは、この程度の大きさの L1 キャッシュを使って、その代わりに大規模な L2 キャッシュを用いて、容量の点をカバーしている。

TLB がミスした場合は、Page Fault 割り込みが発生し、OS が呼び出される。OS は、メインメモリ上の変換テーブルをチェックし、ページが物理メモリ上に存在しない場合は、これをディスクから読み出し、管理テーブルおよび TLB の内容を書き換える。この操作をスワップイン (Swap-in) と呼び、逆にディスクにページを書き込む操作をスワップアウト (Swap-out) と呼ぶ。

キャッシュとは違って、OS が介入するため、スワップには時間がかかるが割り込みの発生頻度は極めて小さいので性能への影響は通常はわずかである (プログラムの利用するメモリ量が大きく、アドレスが分散している場合は、この割り込みが頻発し、スラッシングと呼ばれる性能低下の原因となる)。ちなみに最近のプロセッサはアドレス空間が大きいので、論理アドレスと物理アドレスの変換テーブル自体が非常に大きくなってしまふ。このため、テーブルを階層化して、一部のみを主記憶に置き、あまり利用されないテーブル自体をディスク上にスワップアウトしてしまう。

TLB には、物理アドレスの他にもいくつかの状態ビットが装備されている。一つは Dirty bit で、そのページに書き込みが起きたかどうかを保持する。書き込みについてのページの扱いはライトバック方式のキャッシュと同じである。ページに対する最初の書き込みで割り込みが掛かり、Dirty ビットがセットされる。このページがスワップアウトの対象となった際、Dirty ビットがセットされていればディスクに対して書き戻しが行われ、セットされていなければ単に領域が書き潰される。追い出しポリシーもキャッシュと同じで LRU (Least Recently Used) である。この辺、ページの管理機構は OS のソフトウェアで管理されることを除いてキャッシュと似ている。

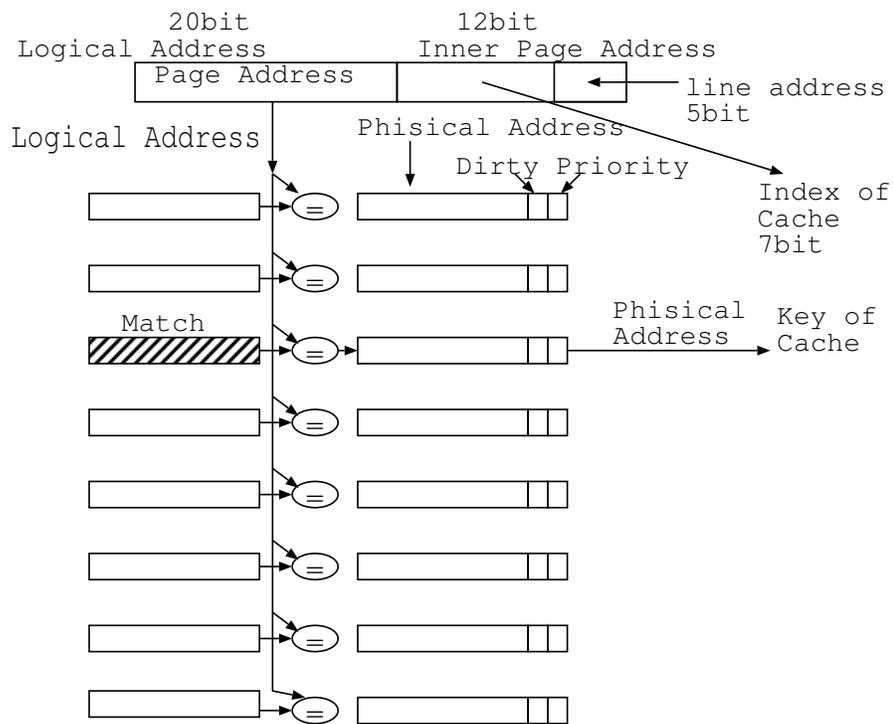


図 1: TLB の構成

ページ変換テーブルには Dirty bit の他にもページ保護用のビットが存在し、これらを用いて、ページ単位のメモリ保護を実現する。OS が管理するページをユーザがアクセスしようとする時、このビットによって割り込みがかかる。仮想記憶は、物理メモリを越える量のメモリを扱うだけでなく、多数のジョブ管理や保護機構についても重要であり、最近では組み込み用のプロセッサでも OS を走らせるために、この機構を装備している。