

7 J-2

HITAC M-880における
ストレージ上のデータ競合高速化方式

釜田 栄樹*, 新谷 洋一*, 庄内 亨*, 井上 潔*, 長井 清治**
*(株)日立製作所 中央研究所, **(株)日立製作所 神奈川工場

1. はじめに

近年、汎用大型計算機の処理性能の向上に対するニーズは、ますます高まりを見せている。このニーズに応えるために、ハードウェア技術の向上および論理方式技術の改善によって、処理性能を大幅に向上した汎用大型計算機M-880を開発した。M-880の命令プロセッサは、マシンサイクル時間を短縮し、命令の演算実行に要するマシンサイクル数を減少するために演算器の構成方式を工夫することによって、演算処理の高速化が行われている。このように、1つの命令の演算処理時間が短縮されているので、先行する命令の演算結果を後続の命令が演算データとして読み出すというデータの競合のために必要となる余計な待ち時間の短縮が重要である。

2. 演算処理におけるデータ競合

M-880の命令プロセッサの演算器は、図1に示すように、8バイト長の二進十進加減算器/シフターから構成されており、その入力および出力にはワークレジスタが設けられている。この構成において、演算処理は、
(1)演算データを汎用レジスタまたはバッファストレージから読み出して、入力ワークレジスタに設定する。
(2)命令の種類に応じて指示される演算を二進十進加減算器/シフターを用いて実行する。
(3)演算結果を出力ワークレジスタに設定して、汎用レジスタまたはバッファストレ

ージに書き込む。

という一連の過程を経て実行される。この演算処理の過程において、先行する命令の演算結果を後続の命令が演算データとして読み出すときにデータの競合が発生する。このデータの競合には、汎用レジスタ上で発生する競合とストレージ上で発生する競合がある。

演算処理の過程においてデータの競合が生じている場合、一般的には、先行命令が処理(3)を完了して演算結果を汎用レジスタまたはバッファストレージに書き込むまで、後続命令は処理(1)を開始することができない。その結果として、演算処理に待ち時間が発生し、処理性能の低下をもたらすという問題がある。この問題を解決する方式として、後続命令が待ち時間なしで処理(1)を実行できるように、先行命令の演算結果を後続命令の演算データとして直接入力ワークレジスタへバイパスして設定するデータバイパス方式が考えられている。

3. データバイパス方式適用上の問題点

図1において、8バイト長のワークレジスタ内の演算データや演算結果は、そのデータ形式によって、二進十進加減算器/シフターが演算処理を行うために都合がよいバイト位置に設定される。例えば、十進データは右詰め、固定小数点データなどは左詰めでワークレジスタに設定される。一方、ストレージ上に置かれているデータは、1バイト単位でアドレスが付けられている。したがって、スト

レジ上でデータの競合が発生した場合に、先行する命令の演算結果と後続の命令の演算データの間で、ワークレジスタ上においてバイト位置のずれが生じている可能性がある。

従来、バイト位置のずれが生じている場合には、データバイパス方式を適用できなかった。その結果、データバイパス方式を適用できる範囲が限定されているという問題点があった。例えば、常にデータが右詰めでワークレジスタに設定されるために、バイト位置のずれが生じることがない十進演算命令の組合せのみに適用範囲が限定されていた。

4. データバイパス方式適用範囲の拡大

M-880の命令プロセッサにおいては、ストレージ上で発生するデータの競合に対して、データバイパス方式を適用できる範囲の拡大を行った。図2は適用できる命令の組合せの例を示している。

この適用範囲の拡大のため、図1に示すように、複数の8バイト長のストアレジスタとその前後にアライナーを設置した。ストレージに対して演算結果のストアが行われると、その演算結果はアライナーによってストレージ上の8バイト境界に位置の整合が行われてストアレジスタにも設定される。ストレージから演算データの読み出しを行うときにデータの競合が発生すると、ストアレジスタから対応する演算データが読み出され、アライナーによって所望のバイト位置に整合が行われてワークレジスタに設定される。このときに用いられるデータパスを図1において鎖線で示す。この場合、ストアレジスタを経由して演算データの設定が行われるので、処理性能の低下要因である待ち時間を完全に除去することはできない。このため、データの位置の整合を必要としない場合には、図1において破線で示されているデータパスを用いて演算データの設定を行い、可能な限り処理性能の低下を抑えている。

5. おわりに

M-880の命令プロセッサにおいて、ストレージ上でデータの競合が発生した場合の高速化方式について述べた。1マシンサイクル時間の短縮とともに、演算器とストレージの間の相対的な距離が増加傾向にある。このため、ストレージ上で発生するデータの競合による処理性能の低下の問題を解決することが一層重要となると考えられる。ここで述べたデータバイパス方式の適用範囲をさらに拡大していくことが今後の課題である。

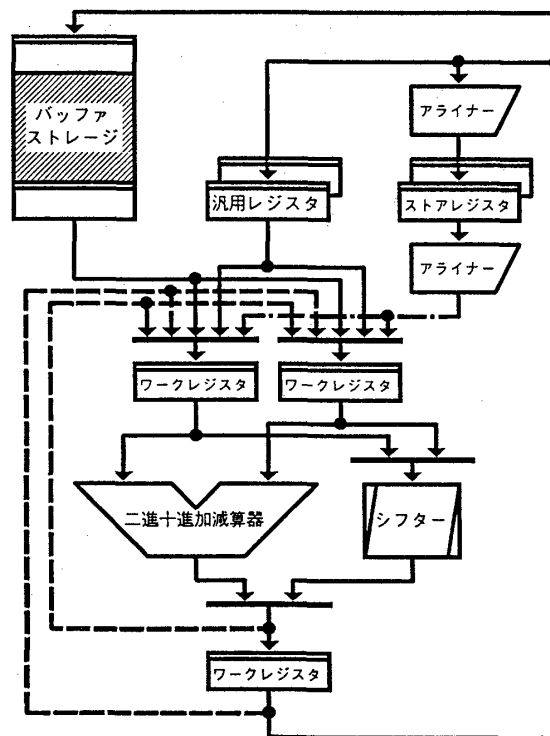


図1 演算器の構成

先行命令 — 後続命令	適用範囲	
	従来	M-880
PACK — AP	○	○
AP — UNPK	○	○
ZAP — AP AP — ZAP	○	○
MVC — AP AP — MVC	—	○
AP — NI	—	○
MVC — NI	—	○

図2 データバイパスの適用範囲