

転送命令削減率の算出によるレジスタ割付け方式

2N-8

田中 旭 山本 和司 田中 裕久 湯川 博司

松下電子工業（株）

1. はじめに

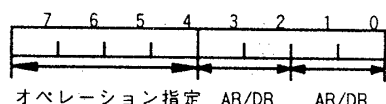
近年、マイコン制御の機器に組込まれるソフトウェアのサイズが大規模化しており、高級言語による開発が要望されている。しかし、高級言語による開発では、オブジェクトコードサイズの増大により、コストアップになることが組込み分野では大きな問題となる。

そこで我々は、マイコンの設計方針として、組込みマイコンに要請されるオブジェクトコードの高効率性と、大規模な組込みソフトウェアを十分に処理可能な高速実行性を両立するために、コンパイラを含めたマイコン設計を行っている。

本稿では、この設計方針に基づいた組込みマイコンMN10200シリーズ用のコンパイラにおけるレジスタ割付け方式について報告する。

2. コンパイラの課題

C言語プログラムのオブジェクトコードサイズを圧縮するために、レジスタをデータ用とアドレス用に機能分離して各4本構成とし、機械語命令のレジスタ指定フィールドを2ビット化することにより、コンパイラが生成する頻度の高い基本命令を1バイト化した。（図1）



【図1】基本命令のビット割付け

しかし、機能分離されたレジスタ4本と本数が少ないために、コンパイラがレジスタに効率よく変数を割付けないと、かえってレジスタ間やレジスタとメモリ間の転送命令が増加してしまい、基本命令1バイト化の効果が削減されてしまう。

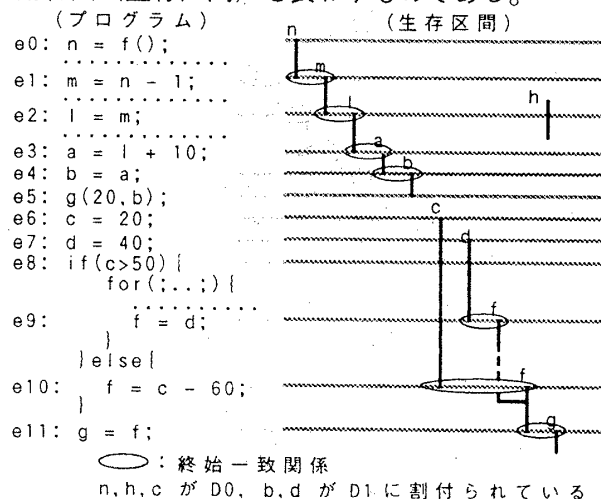
グラフカラーリング方式^[1]として知られている従来のレジスタ割付け方式は、16本以上の比較的本数の多い汎用レジスタに対しては有効な割付け方式であるが、本数が少なくデータ用、アドレス用などのように機能差があるレジスタや、関数の引数や戻り値用のように用途が指定されているレジスタも含めたレジスタ割付け方式としては不十分である。そこで我々は、

- (1) 各変数へレジスタを割り付けたときの、変数の使用場所での詳細な使用コスト算出^[2]と、
- (2) 無駄な転送命令の削減度合を定量化した転送命令削減率の算出により、

使用コストが最小で、転送命令削減率が最大のレジスタを変数に割付けるレジスタ割付け方式を考案し実装した。以下では、(2)についてさらに詳しく述べる。

3. 転送命令削減率の算出

図2はプログラムと変数が保持している値の有効範囲（生存区間）を表わすものである。



【図2】プログラムと生存区間の例

変数aとbのように生存区間の終わりと始まりが一致するものを「終始一致関係」にあるということにする。ここで変数aとbを同じレジスタに割付けると代入式e4に対応する転送命令は不要となる。またコンパイラが生成する機械命令が2オペランド形式であれば、式e3のような演算式においても、変数lとaに同じレジスタを割付ければ、不要な

A Register Assignment Method by Estimating Ratio of Reducing Transfer Instructions

Akira TANAKA, Kazushi YAMAMOTO,

Hirohisa TANAKA, Hiroshi YUKAWA,

Matsushita Electronics Corporation

転送命令は生成されない。

さらに、図2の変数n,m,l,a,bのように、終始一致関係により生存区間が連続しているもの同士は、可能な限り同じレジスタに割付けられれば転送命令を削減できる。そこで、ある変数Xから終始一致関係により生存区間が連続している変数を、変数Xと「区間連鎖関係」にあるということにする。図2では変数aと区間連鎖関係にある変数はn,m,l,bであり、変数fと区間連鎖関係にある変数はc,dである。

以上を踏まえて、本方式では以下の3点に着目した。

(1) 区間連鎖関係にある変数のうち、位置的に近い変数と同じレジスタに割付ける方が有利。

例えば、図2の変数aに割付けるべきレジスタ選択において、変数n,bに予めレジスタD0,D1が割付けられているとする。このとき、aが存在する位置からnが存在する位置までは、bに比べて距離があるため、変数m,lもnと同じD0に割付けられる可能性が低い。よって、aはbと同じD1に割付ける方が確実に転送命令を削減できる。

(2) 区間連鎖関係にある変数と生存区間が重なる変数に割付けられているレジスタと、異なるレジスタに割付ける方が有利。

例えば、図2の変数aに割付けるべきレジスタ選択において、変数hがレジスタD0に割付けられているならば、変数lにはD0は割付け不可能であり、aにD0を割付けると式e3で転送命令が生成されてしまう。

(3) 区間連鎖関係にある変数中のうち、ループ内で終始一致関係が成立している変数と同じレジスタに割付ける方が有利。

例えば、図2において変数fは変数c,dと区間連鎖関係にあり、c,dが異なるレジスタD0,D1に割付けられているとき、fに割付けるべきレジスタ選択において、fとdは式e9で終始一致関係にありかつ式e9はループ中に存在するので、fをdと同じD1に割付けると式e9の転送命令が生成されず、ループ内を高速実行できる。

以上の着目点から、変数Vにレジスタrを割付けたときの転送命令削減率を図3のように定式化した。

$$\text{MovDelRate}(V, r) = \sum_u K(u, r) * W(V, u) / D(V, u)$$

u: rと区間連鎖関係にある変数

K(u, r): uがrに割付けられているとき 1

uがrに割付けられている変数と生存区間が重なるとき -1
そうでないとき 0 ... (1)(2)

W(V, u): 変数Vと変数uにおいて終始一致関係が成立する式のループの重み。内側のループ程大きい値となる。

但し、変数Vと変数uが終始一致関係にないときは1 ... (3)

D(V, u): 変数Vから変数uに至るまでの生存区間の長さの和... (1)(2)

図3 転送命令削減率の式

4. 性能比較

図4は従来のグラフカラーリング方式と本方式である転送命令削減率を使用したレジスタ割付けを行った場合について、コンパイラが生成する命令のステップ数を比較したものである。

評価条件は、ターゲットマシンの命令形式を2アドレス形式とし、レジスタ構成はアドレス・データレジスタ(AD)各4本構成、および汎用レジスタ(GR)8本構成とし、レジスタのうち4本を関数の引数用レジスタおよび戻り値用レジスタとして使用することとした。また、評価対象はスタンフォード・ベンチマークテストである。

図4のように本方式は従来方式に比較して約7%~30%、ステップ数を縮小できた。

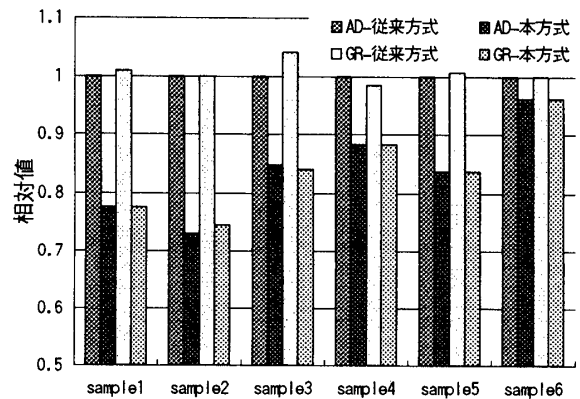


図4 ステップ比較

5. 結び

今後、レジスタ割付け方式とレジスタ特性(本数、機能)との関係を定量的に評価し、よりオブジェクト効率の高いマイコンシステムの構築を行っていく予定である。

【参考文献】

- [1]A.Aho, et al."Compilers Principles, Techniques, and Tools",Addison Wesley,1988
- [2]田中他,"最適化コンパイラにおけるレジスタ割付けの一方式" 情処全国大会 Oct.1991