

CASL II ノウハウ集(1)

- GR1に定数nを入れる

LAD GR1,n

LD GR1,DAT ; DAT番地に DC nが必要

LD GR1,=n ; リテラル=上記をアセンブラが自動化

XOR GR1,GR1 ; (n=0 の場合のみ可)

- GR1の内容を 定数n だけ増減する

LAD GR1,n,GR1 ; FRはセットされない事に注意

ADDA GR1,GR2 ; ; 初期化で LAD GR2,n を行っておく

SUBA GR1,GR2 ; ; 初期化で LAD GR2,n を行っておく

- GR1の最下位ビットだけを残す(上位15ビットを0にする)

SLL GR1,15 → SRL GR1,15

AND GR1,=#0001

- GR1の上位8ビットをGR2の上位8ビットの内容に入れ替える

SLL GR1,8 → SRL GR1,8 → SRL GR2,8 → SLL GR2,8 → OR GR1,GR2

AND GR1,=#00FF → AND GR2,=#FF00 → OR GR1,GR2

- GR1の全ビットを反転する

XOR GR1,=#FFFF

- GR1が奇数か偶数かをテストするためにFRをセットする

SLL GR1,15 ; GR1の最下位が FRのSフラグ に反映される

1

CASL II ノウハウ集(2)

- メモリ上の連続番地をアクセスする(小さい番地からLoadする場合)

インデックス・レジスタ使用の場合

LAD GR2,0 ; インデックス・レジスタの初期化

ループ { LD GR1,DAT,GR2

LAD GR2,1,GR2 ; インデックス・レジスタの増加

アドレスを直接レジスタに入れる場合

LAD GR2,DAT ; レジスタに開始番地をセット

ループ { LD GR1,0,GR2

LAD GR2,1,GR2 ; アドレスの増加

- メモリ上の連続番地をアクセスする(大きい番地からStoreする場合)

インデックス・レジスタ使用の場合

LD GR2,N ; N番地に語数nが入っていると仮定

LD GR2,-1,GR2 ; インデックス・レジスタ GR2=n-1 から始める

ループ { ST GR1,DAT,GR2

LAD GR2,-1,GR2 ; インデックス・レジスタの減少

アドレスを直接レジスタに入れる場合

LD GR3,N ; N番地に語数nが入っていると仮定

LAD GR2,DAT,GR3 ; レジスタに開始番地+nをセット

LD GR2,-1,GR2 ; 開始番地+(n-1)から始める

ループ { ST GR1,0,GR2

LAD GR2,-1,GR2 ; アドレスの減少

2