

SDRAM 時代の ROM 化

セクションとその役割

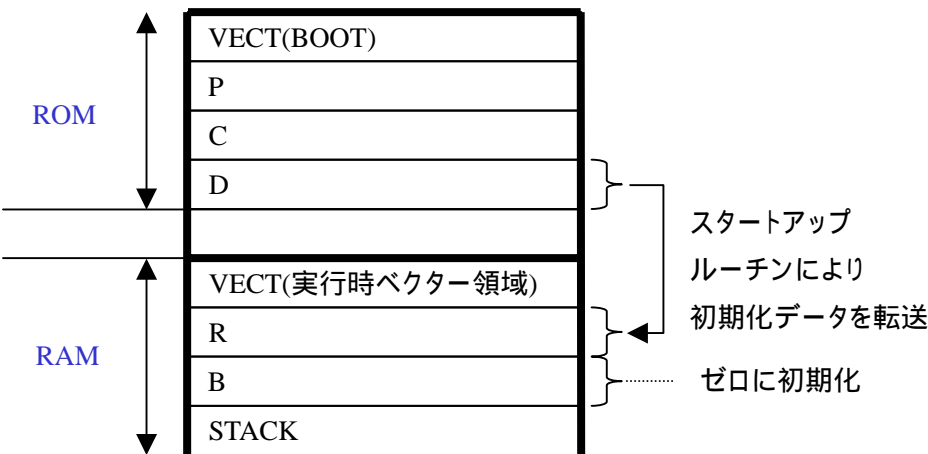
プログラムをコンパイル/リンクすると、セクションと呼ばれる目的別のエリアに分かれてメモリー上に配置されます。以下にルネサス SH C/C++ のセクション名と内容を示します。

内容	セクション名
ベクター(BOOT プログラム)	VECT
(BOOT プログラム)	(PON)
プログラム	P
コンスタントデータ	C
R セクションの ROM 保存用コピー	D
初期化が必要なデータ	R
ゼロに初期化するデータ	B
スタック	STACK

従来の ROM 化

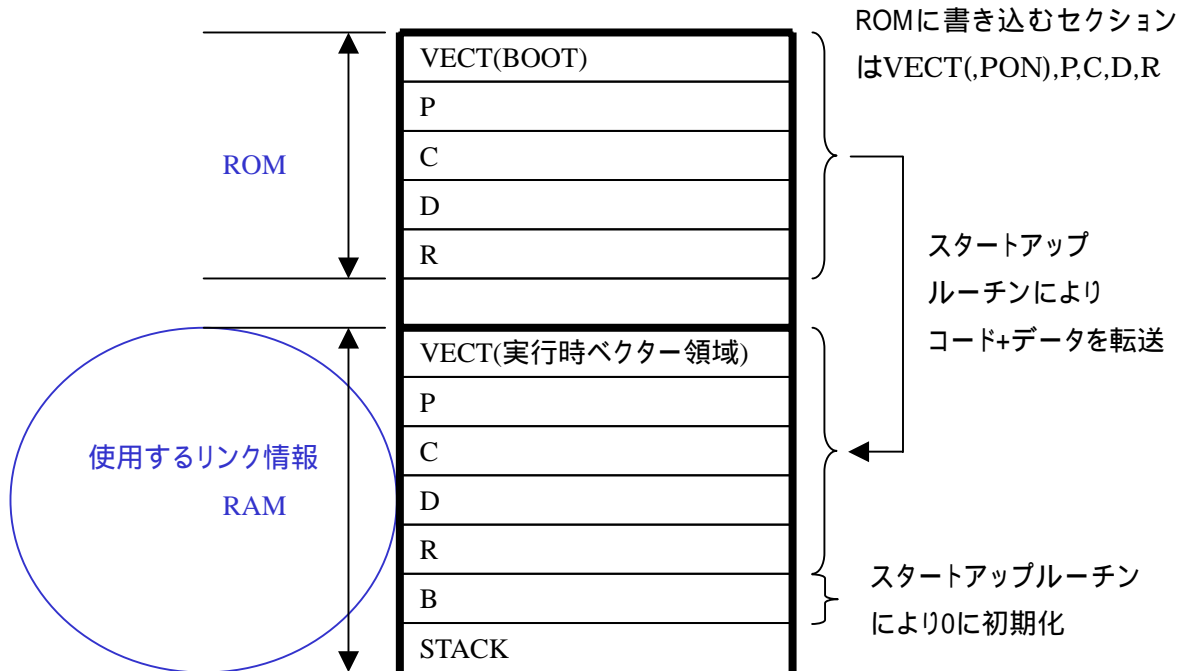
従来の ROM 化では以下の図のようにスタートアップルーチンにより初期化データを ROM から RAM にコピーし B セクションをゼロに初期化します。プログラムコード(P セクション)及びコンスタントデータ(C セクション)は ROM エリアのままです。

SH 等の SDRAM が使用可能な CPU では FLASH/ROM 化時にデータセクションだけを SDRAM 上に配置しても本来のスピードが出せません。



FLASH/ROM でコードを実行してもシステムとして十分なパフォーマンスが得られる場合は、コード領域を FLASH/ROM に割り当てるこの従来方法を使用することも出来ます。

SDRAM 時代の ROM 化



NORTi の組込み BOOT コードではリンクされた時のアドレスと現在実行中のアドレスに差異がある場合(オフセットを付けて書き込まれた FLASH のコードを実行している場合)、元のアドレス (SDRAM) にプログラムのコード/データをコピーし SDRAM 上でプログラムを実行します。

組込み BOOT コードは移動(オフセット)がない場合は何もしません。

実際の ROM 化と実行実際の FLASH 化のステップは以下ようになります。

1. SDRAM 上で実行するプログラムを作成・デバッグする。

(このときこのコピーコード(copy_s 部分)は何もしません)

2. プログラムを SDRAM にロードしたイメージをそのまま

オフセットを付けて FLASH に書き込みます。

オフセットを付けて書き込んだプログラムが動き始めると。

1. SDRAM の初期化をする。
2. コピーコード(copy_s 部)は移動(オフセット書き込み)を確認し元のアドレス(SDRAM)にプログラムをコピーし元々のアドレス(SDRAM)にジャンプしあたかも最初から SDRAM で動いていたかのように動き続けます。

FLASH->SDRAM コピー機能を使用するときの制限事項

プログラムの開始から copy_1 までのコードはすべて「プログラムカウンタ相対命令を使用する」か、「オフセットを配慮したコード」にする必要があります。copy_1 の後の「mov.l INIT,r3 jmp @r3」で元のアドレスに絶対アドレスジャンプしています(FLASH から SDRAM ヘジャンプ)。

リンカーコマンド “ROM D=R” は削除して下さい。R セクションはダミーセクションとして残して下さい。

例:SH7616EV+の設定と PALMiCE での書き込み

SH7616EVABOD+に富士通 MBM29LV004TC を実装した時の例です。

・SH7616EV+の設定

SRAM を CS1 へ接続

JP9 1-2

FLASH を CS0 へ接続

JP10 1-2

CS0 空間 8 ビットバス幅に

JP11...2-3, JP12...2-3, S1 - 4...ON, S1 - 5...ON

・ PALMiCE(CSIDE)の設定

CSIDE を立ち上げ[設定(S)]->[ターゲット・システムの設定(G)]を選択します。[メモリ・マッピング・リスト]タグで“No.”の列の“1”を右クリックし[変更(M)...]を選択しフラッシュの種類を選択します。

終了アドレス(N) ... 07FFFF

メーカー名 ...富士通

バス幅(U) ... 8 ビット

品名 ... MBM29LV004TC

メモリ種別(M) ... フラッシュ・メモリ

ファイルのダウンロードする時に“ロード・アドレス(&R): “に 06000000 を設定して CTX をロードすると FLASH に書き込みます。