

前回(本誌 2007 年 7 月号, pp. 144-153)は,本誌 2007 年 5 月号付属の V850 マイコン基板を用いた,カラー表示ビデオ・ゲーム機の製作方法を解説した.後編の今回は,本ゲーム機のソフトウェアの仕組みを解説し,NTSC ビデオ信号によるテレビへのカラー画像表示方法やゲーム操作に用いた 3 軸加速度センサの利用方法を解説する.なお,ここで紹介するファイルは,本誌の Web サイト(http://www.cqpub.co.jp /interface/)からダウンロードできる.解凍してできるフォルダ(VideoGame)はC:¥CQ に置くものとする. (編集部)

1. ソフトウェア開発環境の構築

開発ツールを設定してコンパイルを行う 本ゲーム機上でプログラムを作成・変更するための開発 ツールの設定と使い方を簡単に説明します.まず, V850 マイコン基板の開発ツールPM+のプロジェクト・ファイ ルC:¥CQ¥VideoGame¥project¥CQ_V850E.prwをオープ ンしてください.このプロジェクト・ファイルでは,既に 本ゲーム機のプログラムをコンパイルするための各種設定 がなされています.自分で設定する場合は,図1と図2を 参考にしてください.ほかの設定項目はそのままでOKで す.なお本ゲーム機のプログラムではC++型のコメント



図1 開発ツール PM+の設定(1)

メニューの「ツール」「コンパイラ共通オプション」を選択する.

コンパイラオブションの設定	コンパイラオブションの設定	コンパイラオブションの設定
	ー 穀 入力ファイル グリブロセッサ C言語と漢字 最適化の詳細設定 外部変数レジスタ 出力ファイル 出力コード	最適化の詳細設定 外部変数レジスタ 出力マール 出力コード メッセージ アセンブラ 一般 入力ファイル アリブロセッサ C言語と漢字 最適化とディバ
「 平通の最適化オブションに設定(1) 最適化レベル(0):	sdata/sbssセクションに配置する データ長の上限(-G)(D): - sonsttzゲノョンのデータ配置	インカルードファイルのパス1-100. 187 Encream Files NEO: Electronics Tools VeASSOVES 11 a Vinces 0
「高度な最適化と-O」 ▼ アセンブルリスト ▼ アセンブルリスト ● ▼ アセンブルリスト	「 配置する[-Xsconst]©) 「 データ長の上限① 「 「 「 「 「 「 「	定義マクロ[-D](地):
		未定義マクロ[-1](山):
○美口通信の表示「*/12/ 定義マクロ[-D](<u>他</u>):	ブロローグエピローグランタイム(W): 「ディフォールト[使用] 「 アナライザ情報の	マクロ酸の上級(-Xm)(2) 「 フリブロセッサのコメントの保存(-O)(2) マクロ酸の上級(-Xm)(2) 「 フリブロセッサのコメントの保存(-O)(2) マサーコントの使用(-1)(2) マサーコントの使用(-1)(2)
	switchの出力コード⑤: 「ディフォールト[自動選択]	
コマンドラインオプション	コマンドラインオブション:	コマンドラインオナション

図2 開発ツール PM+ の設定(2)

メニューの「ツール」「コンパイラオプション」を選択する.



Close

(((()

Refresh



図3 PM+上でデバッガを設定

メニューの「ツール」「ディバッガの設定」

とで , プログラムがデバッガ用モニタと一緒にマイコンのフラッシュ ROM にダウン ロードされる . ダウンロード完了後 , 自動的にデバッガが立ち上がる .

▶▶▶ ▲▶ 씨스 않은 해양 제에 가지에 가지에 가지만 하는 것이다.

> Watc

//
Init_Port();
Init_Video();
InitADC();
InitDAC();

// Main Loop

10(1)

UI08 state = BREAKOU

// // Initialize Hardware

Disp_Logo_CO(); Disp_Logo_Interface();

șvitch (state)

図4 デバッガ ID850QB を起動

Search ((

(//)を使っています.

まずはコンパイルします.デバッグ情報を残すため, 「Debug Build」を選択してからメニューの「ビルド」「リビ ルド」を選択してください.エラーやワーニングを出力せ ず,無事にコンパイルが終了することを確認してください.

デバッガでプログラムの動作を確認

次にデバッガ ID850QB を起動します.まず, PM+上で 図3のようにデバッガの設定を行ってください.ここでデ バッグ対象となるオブジェクト・ファイルを指定します. また, V850 マイコン基板のジャンパは, J2とJ3をショー トします.メニューの「ビルド」「ディバグ」を選択する と,オブジェクト・ファイルが V850 マイコンにダウンロー ドされます.

ダウンロードが終了すると図4のようなデバッガID 850QBが立ち上がります.メニューの「実行」「リスター ト」で本ゲーム機のプログラムがスタートします.デバッ グが終わったら前編で説明したように,FPL(フラッシュ ROM 書き込みツール)でオブジェクト・ファイルを V850 にダウンロードします.PM+上で「Debug Build」により コンパイルした romp.hex でも問題なく動作しますが,念 のため最終的には「Release Build」で生成したオブジェク ト・ファイルを使います.

デバッガを使用したマイコンは量産品には使えない デバッガを使用する場合でも,プログラムは実際にマイ コン内蔵フラッシュ ROM へ書き込まれるので,リアルタ イム・デバッグが可能です.FPLよりもデバッガによるダ ウンロードの方が高速ですが,ダウンロード時間を短縮す るために正規の書き込み方式をとりません.フラッシュ ROM のリテンション特性(データ保持特性)を無視して, 浅めの書き込み深さにとどめているのだと思います.なお, 書き換え回数が100回に達していなくても,1回でもデバッ ガを使用したマイコンは量産品には使えませんので注意し てください.

PM+上でプログラムのビルドを完了したあと、「ビルド」「ディバグ」を選択するこ

2. ソフトウェア設計

シンプルなメイン・プログラム構造

メイン・プログラムの構造を図5に示します.パワー ON リセット後,アセンブリ言語で書かれたスタートアッ プ・ルーチンが実行されます.この中で,スタック初期 化,sbss/bssセクションのクリア,ハードウェアの設定が 行われます.本ゲーム機のプログラムはローカル変数が多 めなので,スタックのサイズをデフォルトの512バイトか ら1,024 バイトに増やしてあります.プログラムが消費す るスタック・サイズは,STK850というツールで確認でき ます.

スタートアップ・ルーチンの実行後,メイン・プログラ ム main()に入ります.ここではアプリケーションに関連 する各種初期化の後,ロゴ画面を表示して,無限ループに