

第2章でも軽くふれたように, Analog Devices はDSP プログラムの開発環境として, VisualDSP++ を中心とするシステムを供給しています. VisualDSP++ はプロジェクト管理,エディタ,コンパイラ, アセンブラ,リンカ,ROM 化ツール,シミュレータ,コンパイル済みシミュレータ,性能測定ツー ルを1パッケージにまとめた強力な統合開発環境で,プロジェクト管理統合開発環境プラグインとし てICEを接続することもできます.また,ICEをもっていなくてもUSB ケープル越しにPC と Analog Devices 製の評価基板を接続して簡単な評価を行うこともできます.

この章では、VisualDSP++と評価基板であるEZ-KIT Lite BF533を組み合わせて使う場合に最低 限知っておくべきことを説明します.一般にツールの使い方を細かく書くと、バージョン・アップと 同時にその情報の価値が落ちてしまいます.そのため、本書ではマニュアル本的な情報は極力少なく したいと考えていますが、この章と第10章、第11章だけは例外としてツールの使い方を細かく説明し ます.

5-1 EZ-KIT Lite版VisualDSP++のインストール

三つのステップに分かれる VisualDSP++のインストール

EZ-KIT Lite版 VisualDSP++のインストールは,三つのステップに分かれています.すなわち, VisualDSP++のインストール

EZ-KIT Lite 用ドライバのインストール

ライセンスのインストール

の3ステップです.これらのうち最初の2ステップは,順番どおりに行えば難しいことはありません. 最初に VisualDSP++をCD-ROM からインストールします.インストールが終了したら,念のため に PCを再起動してください.次に EZ-KIT Liteに電源を入れ,USB ケーブルで PCと接続します. 環境に問題がなければこれでデバイス・ドライバがインストールされます.EZ-KIT Lite BF533の USBポート横の USB MONITOR LED が点灯すればインストールは成功しています(写真5-1).インス



写真5-1 EZ-KIT Lite BF533のUSB MONITOR LED

トール後に念のためUSBケーブルをはずして再びPCを再起動し, ログインして再度USBケーブルを 挿してください.

やや面倒なライセンスのインストール

インストールが終わったら,最後にライセンスを設定します.ライセンスの設定はやや面倒で,以下のステップをふみます.

1.シリアル番号を入力してライセンスを仮設定状態にする.

2. Analog Devices に登録してバリデーション・コードを取得する.

3. バリデーション・コードを入力してライセンスを恒久設定状態にする.

VisualDSP++にシリアル番号を入力するとライセンスが仮設定状態となり,10日間利用可能になり ます.この間に Analog DevicesのWebサイトからユーザ登録を済ませ,バリデーション・コードを 取得します.バリデーション・コードをVisualDSP++に入力すると,恒久的に利用できるようになり ます.

Column...5-A インストール・ディレクトリ

VisualDSP++のインストール・ディレクトリは, メジャー・バージョンごとに違います.たとえば 筆者のシステムでは次のようになっています.

```
C:¥Program Files¥Analog
Devices¥VisualDSP 3.5 16-Bit
C:¥Program Files¥Analog
```

```
Devices¥VisualDSP 4.0
```

```
いうまでもなく,上が VisualDSP++ 3.5で,下
```

が 4.0です . VisualDSP++ 4.0では 16/32ビット 製品群の開発ツールが統合されたため,以前あっ た" 16-Bit "という言葉が取れています.

本書で「インストール・ディレクトリ」と呼ぶと きには上記のような Visual DSP++ 関連ディレクト リの基点となる場所を指しますので,覚えておい てください.また,ファイル・パスはとくに断ら ないかぎりインストール・ディレクトリを基点と したものとします. ライセンス・インストール手順の実際

実際の手順を追ってみましょう.EZ-KIT Lite に電源を入れ,PCに接続します.USB MONITOR LED が点灯したのを確認して Visual DSP++を立ち上げます.最初の起動であれば,ライセンスの登録が必要であると知らせるダイアログが表示されます(図5-1).そこで Licenses ...ボタンを押すと,ライセンス管理ダイアログが現れます(図5-2).ライセンス管理ダイアログが現れない場合は,メニュー・バーから Help About Visual DSP++...項目を選び,Licenses タブをクリックしてライセンス管理ダイアログを出してください.ダイアログの New ボタンをクリックして EZ-KIT Lite に同梱されている CD-ROM ジャケット裏のシリアル番号を入力します(図5-3).シリアル番号は,KIT-で始まる文字列です.

ユーザ登録は先のシリアル番号の入力から10日以内に済ませます.10日を越えると一時的に VisualDSP++を使えなくなるので注意が必要です.また,登録からバリデーション・コードの発行ま で,3日ほど見ておくほうがよいでしょう.ユーザ登録は,Analog DevicesのWebページから行い

Column...5-B ライセンスについて

VisualDSP++には,ライセンス形態に応じて三 つの亜種があります.

- 正規製品版
- EZ-KIT lite版
- ・お試し版(Test Drive)

これらの三つの亜種は, すべて同じバイナリか らなっています.つまり, インストールされるプ ログラムはまったく同じなのです.それぞれの版 の違いは,登録するシリアル番号とバリデーション・コードからなります.

ライセンスごとの違いは, VisualDSP++4.0にお いては表5-Aのようになっています.

バイナリがすべての版で共通であるため,どの ライセンスでもアップデートを適用することがで きます.

ライセンス	シリアル番号			バリデーション・コード
形態	形式	入手法	入力後の制限	入力後の制限
正規製品版	ADI-###	ADI-### シリアル番 号は製品 CD のジャケッ トに貼ってある.	機能制限なし . バリデー ション・コードを入力 しないと 30 日間で使用 不能になる .	制限なし
EZ-KIT Lite版	KIT-###	KIT-### シリアル番 号は製品 CD-ROM のジ ャケットに貼ってある.	機能制限なし . バリデー ション・コードを入力 しないと 10 日間で使用 不能になる .	インストールから90日 間は機能制限なし.その 後,シミュレータとエミ ュレータが使用不能にな る.USBケーブルによる EZ-KIT LiteとPCの接 続は制限されない.また, ビルドできるコード・サ イズに制限がかかる.
お試し版 (Test Drive)	TST-###	TST-### シリアル番 号は Analog Devicesの ウェブサイトから取得	機能制限なし.90日間 で使用不能になる.	入力できない.

New Session	New Section
Debug target Processor:	Det Manage Licenses
No valid targets found	No
Platform:	Plat Serial Number Family Status
· · · · · · · · · · · · · · · · · · ·	
Session name:	Ses
Show all targets and platforms	
OK Cancel Licenses Configurator	
No valid targets could be found.	No Machine ID
This may be because a license has not yet been installed or your current license has expired. Click the Licenses button to view and/or modify the currently installed licenses	The Register Validate 76ABUSE2
citer are been bes baken to view anayor mouny are carrently instance incenses.	OK ++>>セル ヘルク
図5-1 初回立ち上げ時のダイアログ	図5-2 ライセンス管理ダイアログ
	🔍 Analog Devices: Embedded Processing & DSP: Visual DSP++ Regist 📃 🔲 🔀
	ファイル(E) 編集(E) 表示(V) 移動(G) ブックマーク(E) ツール(I) ヘルプ(H) 〇
	← ・ → ・ 台 ② ① http://forms.analog.co 図 移動 G.
	Analog Devices: Embedded P
	ANALOG DEVICES
	Search: Embedded Processing & DSP
	Home > Embedded Processing & DSP
	Processors VisualDSP++ Registration
New Session	Communities
M Install New License	Platforms Thank you for purchasing Analog Devices' VisualDSP++™.
Deb	Tools and Resources dstools.registration@analog.com. If you have multiple computers, you need to
You are about to install a software license to run VisualDSP++.	Technical Library your serial number and computer machine ID. Your validation code will be e-ma therefore, please verify you have entered a valid e-mail address in the registrati
Plan Please select one of the following license types:	Design Tools Microsoft Internet Explorer 5.0 or Netscape 6.0 or higher.
Single user license or Test Drive license	Beginner's Guide to DSP This form is for VisualDSP++ development environments with the serial numb KIT Register the Test Drive Version here.
Client license	Third Party Developers Dance 1 of 2
Description	Benchmarks Fields with a red asterisk (*) are required
S This license type allows you to use the VisualDSP++ tools on a single	Please provide your Serial # exactly as it appears on the CD-ROM case (inclu Technical Support
registration card, CD sleeve or web site.	Training and Events
	News Which product are you registering? *
No Serial number: KIT-1072040447	
T2	All Product Categories Machine ID: * To locate your Machine ID: install VisualDSP++, go to the VisualDSP \System s Decime Contex b
Cl Cancel	Buy Online >
	Enter up to eight hex digit Machine ID exactly as it appears in your reg_info.txt file) (e.g., 1s2b304d, may be less than eight if there are leading zeros in the Volume Serial Nun V
	4

図5-3 シリアル番号の入力

図5-4 ライセンスの入力

ます.このページには図5-2のライセンス管理ダイアログのRegisterボタンをクリックしてアクセス できます.図5-2ではRegister...ボタンは無効になっていますが,シリアル番号を入力してリストの 中で選択するとボタンが使用可能になります.Webページが開いたら,シリアル番号と図5-2の右下 に表示されているマシンIDのほか,ユーザ情報を登録します(図5-4).登録が完了すると数日でバリ デーション・コードが送られてきます.

バリデーション・コードは,シリアル番号同様にライセンス管理ダイアログから登録できます.バ リデーション・コードを登録するとライセンスがPCに固定され,恒久的にVisualDSP++を利用でき るようになります(**図**5-5). 最新アップデートの適用

ライセンスのインストールが成功したら,最新のアップデートを適用します.アップデート・ファ イルは Visual DSP++の最新版とともに Analog Devices の Web サイトからダウンロードできます^{注1}.こ のファイルは本書の執筆時点では不定期に更新されています.バグが取れていきますのでなるべくア ップデート・ファイルを適用することをおすすめします.

アップデート・ファイルを入手するには,Windowsのコントロール・パネルから,「プログラムの 追加と削除」を開きます.一覧の中からVisualDSP++を探して「変更と削除」ボタンをクリックします. するとダイアログが現れる(図5-6)ので, "Go to Analog Devices web site"を選んでからNextボタンを クリックします.VisualDSP++のアップデート・ページが開くので,最新のアップデート・ファイル をダウンロードしてください.ファイルの拡張子は.vduです.

アップデート・ファイルをダウンロードしたら Web ページを閉じ,再びコントロール・パネルのリ ストから VisualDSP++を探して「変更と削除」をクリックします.ダイアログが開いたら,今度は "Apply a downloaded update"を選んでNext ボタンをクリックします.あとは,ウィザードの指示 にしたがってアップデートを進めます.

5-2 統合環境でプログラムを走らせる

ライセンスの設定が終わったら,いよいよ VisualDSP++を使えるようになります.

最初に EZ-KIT Lite版の VisualDSP++を使うときの注意を書いておきます.正規版の VisualDSP++を買わずに EZ-KIT Lite版のみを買った場合には, EZ-KIT Liteなしで VisualDSP++ を使うことはできません. VisualDSP++を立ち上げる前に必ず EZ-KIT Liteの電源を入れ, PC に接続してから MONITOR_LED が点灯するのを待ちます.点灯を確認(写真5-1)してから VisualDSP++を起動します.



注1:http://www.analog.com/processors/processors/Blackfin/crosscore/toolsUpgrades/

さっそくプログラムを作ってみる

VisualDSP++が立ち上がったら、さっそくプログラムを作ってみましょう.VisualDSP++では、プ ログラムは「プロジェクト」という単位で管理されています.プロジェクトとは、ソース・コードやラ イブラリからなるツリー構造で、VisualDSP++上でグラフィカルに管理することができます.新しい プロジェクトを作るには、メニュー・バーのFile New Project...項目を選びます.そうすると、 プロジェクト・ウィザードが現れてプロジェクトの初期設定が始まります(図5-7).ここではプロジ ェクトの保存場所とプロジェクトに付ける名前を指定します.保存場所とプロジェクト名には日本語 を含まないように気をつけてください.英語圏で作られたソフトウェア全体にいえることですが、う

Project Wizard Project Information Choose the name, location,	and type of project that you would like to create.	Project Wizard Project Information Choose the project's target	processor and output type.	
Project Posternari Post	Project : General Name: Name: Directory O'VDocuments and SettingsYshorieYMy DocumentsYVsuaIDSP Project types: Ibstandard application Iuther/threaded application using VDK Inter/threaded application using LwIP and VDK.	Project Pag General Pg Output Type Other Type W Finish	Project : Output Processor types: Processor Apple-DeF32 Apple-DeF33 A	ut Type Description Blackfin Processor (800 MMAOS, 84K by: Blackfin Processor (800 MMAOS, 84K b; Blackfin Processor (700 MMAOS, 84K b; Blackfin Processor (700 MMAOS, 84K b; Blackfin Processor (104K bytes orn-chip, Blackfin Processor (144K byt
	Kext > Finish キャンセル	Processor types Select the processor for which you w	ant to build your project.) Next > 「Finish キャンセル

図5-7 プロジェクト・ウィザード

図5-8 プロジェクトの出力設定

Column 5-C 日本語ディレクトリについて

VisualDSP++は,日本語をはじめとする2バイ ト・コードからなるディレクトリをうまく扱えま せん.この結果,次のような場合には動作が非常 に怪しくなります.

一つは、プロジェクト・ディレクトリを日本語 ディレクトリの下に置いた場合です。典型例がデ スクトップで、ここにプロジェクト・ディレクト リを作ると、ビルドで失敗してしまいます。プロ ジェクト・ディレクトリには漢字やひらがなをは じめとして2バイト・コードを使用しないでくだ さい、Windows 2000やXPの場合、マイドキュ メント・フォルダの下にプロジェクトを作っても 正しく動くため,日本語ディレクトリが許されて いるように錯覚してしまいます.しかし,マイド キュメント・フォルダのパス名はMy Documents であるため,実際には2バイト・コードを含んで いません.

問題が発生するもう一つの場合は, VisualDSP++を日本語ディレクトリにインストー ルしてしまった場合です.中にはデスクトップの 下にプログラムをインストールして「動かない」と おっしゃる方もあります.VisualDSP++はなるべ くデフォルト・ディレクトリにインストールして ください. っかり日本語のファイルやディレクトリを使うと理解に苦しむエラーを起こすことがあるので注意が 必要です.

NEXTボタンを押すと、ウィザードが次の設定に進みます(図5-8).ここでは、プロジェクトのタ ーゲット・プロセッサなどコードの出力に関する設定を行います.EZ-KIT BF533の場合、 Processor TypeはADSP-BF533にしてください.Silicon Revisionは、使用しているチップの版で す.これは写真5-2のように、パッケージ上に印字されているものを入力してください.Project Type は、Executable Fileを選びます.設定が終わったら、Finishボタンを押してください.まだ設定で きる項目もありますが、今は無視してかまいません.

以上の設定を終えると,図5-9のように空プロジェクトによる VisualDSP++の画面が現れます.左側にはプロジェクト・ウィンドウ,右側にはディスアセンブル・ウィンドウ,下側に出力ウィンドウが配置された構成です.インストール直後と似ていますが,左側のプロジェクト・ウィンドウに新しいプロジェクトが表示されている点が異なります.



写真5-2 シリコン・リビジョン

Column...5-D アノーマリはチェックしてください

Analog Devices はチップの不具合のうち設計 に起因するものを「アノーマリ」として公開してお り,製品ページからダウンロードして読むことが できます.

アノーマリの中には軽度のものもあれば,かな りきついものもあり,さまざまです.たいていの ものには回避策がありますが,回避が困難なものの場合は,次のリビジョンで変更されます.

公表されているアノーマリのリストには,それ ぞれの回避策が提示されています.アノーマリは 定期的にアップデートされるので,ときどき目を 通すようにしてください.



図5-9 新しいプロジェクト

ソース・ファイル作成

さっそく何かプログラムを作ってみましょう.まずはソース・ファイルを作ります.ソース・ファ イルは、メニュー・バーから File New... File 項目を選ぶと作られます.ソース・ファイルを作っ たら、名前をつけて保存します.ここで注意すべきこととして拡張子の指定があります.ワープロソ フトなどは拡張子を勝手に付けてくれるので,拡張子なしでファイルを保存することに慣れている人 がいるかもしれません.しかし、VisualDSP++は各種のファイルを作る都合上,デフォルトの拡張子 は「なし」になっています.そのため、名前をつけてファイルを保存する場合,必ずユーザが拡張子を つけて保存しなければなりません.とりあえず今はmain.cppという名前でファイルを保存してくだ さい.

プロジェクトへのファイル追加

ファイルを保存したら,今度はプロジェクトに追加します.プロジェクトにファイルを追加すると,



図5-10 プロジェクトへのファイル追加

ビルド時に統合環境が自動的に適切なコード・ジェネレータを呼び出してくれます.プロジェクトへの追加は, Project メニューから Add to Project サブメニューの Files...項目を選んでください(図5-10).ファイル選択ダイアログが現れるので,先ほどのmain.cppを選択します.すると,プロジェクト・ツリーに追加したファイルが現れます.

ファイルの追加が終わったら,空のままのソース・ファイルにプログラムを書き込みましょう.何でもよいのですが,定番のHello,World!にしておきましょう.LEDの制御などの組み込みらしいプログラムは後の節で紹介します.

```
#include <iostream.h>
int main(void)
{
    cout << "Hello, World!" << endl;
}</pre>
```

入力したら,保存してビルドをかけます.ビルドはProject メニューから Rebuild All 項目を選んで ください.プログラムのビルドが進み,エラーがなければ実行ファイルの EZ-KIT Lite へのダウンロ ードが始まります.ダウンロードには少し時間がかかりますが,やがてmain 関数の頭でプレークがか かって停止します.

この状態でEZ-KIT Liteは停止しており, VisualDSP++はユーザの操作待ちです.そこでDebug メニューのRun項目を選択すると実行が始まり, VisualDSP++の出力ウィンドウにメッセージが表示 され,実行が終了します. VisualDSP++によるプログラミングの例に関しては第6章や第7章にもう少し詳しく説明していま すので,そちらも参照してください.

5-3 C/C++言語のコンパイラ,アセンブラ,リンカ

高級言語とアセンブリ言語の混用を考えた設計

VisualDSP++には,C/C++言語のコンパイラ,アセンブラ,リンカが一式付属してきます.これ らのツールはC++言語のマクロ・プロセッサを共用しており,アセンブラとコンパイラがプロセッサ

Column ... 5-E エディタの日本語対応化

VisualDSP++のエディタは,日本語対応化する ことができます.日本語対応によって日本語のコ メントを入力,編集,表示できるようになります. ただし,この機能は正式には未サポートとなって いるので注意が必要です.

日本語への対応は,エディタのフォント変更に よって行います.フォントの設定はSettingsメニ ューのPreferences項目を選ぶとダイアログが現 れるので,ツリーの中からEditorをクリックする ことで設定画面を呼び出せます.設定画面のType コンボボックスを選んで,どのファイルのフォン トを設定するのか選びます.普通はC/C++のソー ス・ファイルと,アセンブリ言語のソース・ファ イルだけで十分でしょう.ファイル・タイプを選 んだら,font...ボタンをクリックしてフォントを 選択します.

フォントは日本語を表示できるものでなければ なりませんが,筆者は普通Fixedsysを使用してい ます.このフォントは等幅であるため,プログラ ムの作成に適しています.それ以外のフォントで はうまく日本語が表示されない場合もあるので気 をつけてください.

なお,以上の変更を行っても,アセンブリ言語 のソースの場合は,コメントが正しく表示されま せん.これはシンタックス・カラーリング機能の せいです.コメントの扱いに問題があるらしく, 1文字の予約語と同じキャラクタが2バイト・コー ドの中に存在すると,日本語文字が崩れてしまい ます.仕方がないので,シンタックス・カラーリ ングの機能を変更することでこの問題を解決しま す.具体的には,カラーリング対象文字列から1 文字のものをすべて取り除きます.カラーリング 文字列の宣言は,インストール・ディレクトリの 下のsys¥blackfinasm.iniファイルで行われ ているので,このファイルをテキスト・エディタ で開きます.そして次のように,1文字予約語の 宣言をすべてコメントアウトします.

JUMP=Mnemonic	
;L=Mnemonic	;予約語"L"の宣言をコメントアウト
LE=Mnemonic	

コメントアウトば;;"(セミコロン)で行います. 1文字予約語をすべてコメントアウトすれば,文 字化けはなくなります.

以上の作業で,日本語のコメントを入力・表示 できるようになります.なお,エディタ自身は日 本語に対応しているわけではなく,無理やり日本 語を表示させているだけです.そのため,2バイ ト文字の削除は削除キーを2回押さなければなら ないという制限があります.

繰り返しになりますが,これはAnalog Devices が公式にすすめている方法ではないので,対応は 自己の責任の元で行ってください.

	Project Options for hello			<u>?×</u>
	B D Project	Project Target		
	- 29 General (1) - 29 General (2) - 29 Parm cascor	Processor: Type:	ADSP-BF533 V Revision: 0.2 V Executable file V	
		Name:	hello	
Analog Devices VisualDSP++ - [Target: Simulator BF533] - [Project: hello]	- 팽 Workarounds = - - 팽 Assemble	Tool Chain	C/C++ Compiler for Blackfin	
File Edit Session View Project Register Memory Debug Settings Tools Windov	- Sa General - Sa LDF Preproce - Sa Elimination	Assembler: Linker:	Blackfin Family Assembler	
● R B B B 参 曲 当 hello V Debug V B B	- 평 Processor 은 팀 Load - 명 Options	Loader:	Blackfin Family Loader	
註 卧 鄧 本 也 像 阿 颐 ひ ひ か 10 66 尋 南 氣 <mark>Release</mark> Project: hello.dpj √ x	-Bogi Kernel Bogi Splitter Bogi Pre−build States	Splitter:	Debug	
Project Group (1 project) Define Define Define Define Define Define		Jettings for	OK ++/>	セル

図5-11 コンフィグレーション・コンボ・ボックス

図5-12 プロジェクト・オプション

のレジスタ・アドレスを定義したマクロを共用できるなど,はじめから高級言語とアセンブリ言語の 混用を考えた設計になっています.

コンパイラは, とくに16ビット演算の最適化に力点を置いて設計されています.これは, Blackfin には32ビットRISCプロセッサとしての「顔」のほかに, 16ビットDSPの「顔」もあるからです.一方で C/C++言語は,本来は信号処理アルゴリズムのような極度に性能に敏感な分野での応用を考えて作ら れたものではありません.そのため VisualDSP++には,コンパイラに最適化のヒントを与えるための pragma や,特殊な命令を活用するための組み込み関数が用意されています.とくに組み込み関数は, ライブラリ関数と異なり,動作をコンパイラが理解しているためかなり本格的な最適化を行うことが できます.

コンパイラとアセンブラに関しては、よほど特殊なことをしないかぎり神経質にオプションをいじ くり回す必要はありません.きつい最適化をかける必要がないならば、最適化やデバッグ・オプショ ンの設定はツールバーの Configuration コンボ・ボックスを切り替えるだけで間に合います(図5-11). Configuration コンボ・ボックスはデバッグ・ビルドとリリース・ビルドを簡単に切り替えるためのも ので、このコンボ・ボックスに連動してコンパイラやアセンプラの設定済みオプションが変わります. それぞれのビルドに対応する設定は、プロジェクト・ダイアログから変更可能です.また、デバッグ とリリース以外のビルドを追加することもできます.きつい最適化をかける必要がないならば、最適 化やデバッグ・オプションの設定はデフォルトのままでよいでしょう.

細かな設定はオプション・ダイアログから行う

細かい設定を行いたいときは,コンパイラとアセンブラのオプションの主なものはプロジェクト・ オプション・ダイアログから設定可能です(図5-12).このダイアログはメニュー・バー Project Options.項目を選択すると現れます.プロジェクト・ダイアログを使って行うよりも踏み込んだ最適 化設定に関しては,第12章で説明します.

コンパイラを使うかアセンブラを使うかの判断は,統合開発環境が自動的に行います.つまりソース・ファイルが.cあるいは.CPPという拡張子をもっていればビルド時にコンパイラが呼び出され,ソ

ース・ファイルが.ASMという拡張子をもっていれば,ビルド時にアセンブラが呼び出されます.した がって,ユーザがファイルとツールを明示的に結びつける必要はありません.アセンブリ言語を使う 場合の文法などの説明は,第9章で行います.

リンカの呼び出しも,自動的に行われます.リンカに関してはソース・コードだけではなくライブ ラリの選択や配置の最適化,メモリ領域の調整といった組み込みならではのことが絡んでくるため, 製品化の前にはかなりの調整を行わなければならないのが普通です.しかし手始めとしては,デフォ ルトの設定も十分使えます.リンカについては第10章で説明します.

5-4 ペリフェラルにアクセスする

MMRにアクセスする

さて, ADSP-BF533の内蔵ペリフェラルはすべてメモリ空間中の MMR(Memory Mapped Register)領域に割り当てられています.このレジスタにアクセスしてみましょう.

レジスタへC/C++言語からアクセスするときには、ポインタからアクセスします.このためのポインタ宣言はVisualDSP++に同梱されているインクルード・ファイルで行われています.cdefBF53x.hというファイルがそれで、#include文で読み込むとポインタ・マクロの宣言が取り込まれます.このマクロ宣言はすべて型付きポインタとなっているので、C/C++言語から比較的安全に使うことができます.一例をあげます.

```
#include <cdefBF53x.h>
volatile unsigned short sw;
int main(void)
{
 *pFIO_DIR = 0x0000; // FIOを入力に
 *pFIO_INEN = 0x0100; // PF8をイネーブルに
 sw = *pFIO_FLAG_D & 0x0100; // PF8の状態を読み込む
}
```

このプログラムはプログラマブル・フラグ(FIO)の8番ピン(PF8)を入力に設定して状態を読み込み ます.PF8はEZ-KIT LiteのSW4に接続されているので,プログラムの実行によってSW4の状態を 読み込むことができます.SW4を押すと変数swには0x0100が,離すと0x0000が代入されます.

マクロを使ったために, FIO_DIRやFIO_INEN レジスタもポインタから指し示される変数としてア クセスできます.そのためプログラムの見通しが非常によくなります.マクロ定義されているポイン タは定数なので,ポインタに値を代入して壊してしまう心配はありません.また,ポインタは volatile unsigned shortへのポインタなので,コンパイラの最適化機構がアクセスを消し去ってしまう心配もあ りません.