

ソフトウェア開発支援システム

Software Development System for DASP

藤 本 昇 治⁽¹⁾ 佐 古 和 也⁽²⁾
Shoji Fujimoto Kazuya Sako

要 旨

ディジタル・オーディオ・シグナル・プロセッサ(DASP/FT8800)のソフトウェア開発支援システムを開発した。

このシステムは、ソフトウェア開発に必要なフィルタ設計プログラムやシミュレーション・プログラム、信号処理プログラム開発のためのクロスアセンブラおよびエミュレータ、DASPの入門解説書や各ツールのマニュアル類まで網羅した総合的な支援システムである。これらを用いることで、DASP用の信号処理アルゴリズム開発や評価が効率良く行える。

本稿では、開発支援システムの全容を紹介すると共に、エミュレータの開発に関して報告する。

Our Software development system for DASP is provided with digital filter design tools, simulators, a cross assembler for DASP, emulators, and manuals for things.

This system makes it easy to develop a DASP products, which is reliable, high quality, and to evaluate them without complicated technique.

The following report describes outline of these systems and gives a special feature of the emulator.

(1), (2) 第一開発部

テムの構成要素を表-1に示す。DASP用開発支援システムは次のように大別することができる。

- ①フィルタ設計プログラムおよびシミュレーション・ツール
- ②信号処理プログラム開発のためのクロスアセンブラおよびエミュレーション・ツール
- ③製品開発のための評価ボードや、プログラム用のROMボード
- ④実際の製品により近い形での、プログラム評価を可能にするICE (In-Circuit Emulator)
- ⑤DASPの入門解説書や各種ツールのマニュアル類

これらのシステムを用いることで、前記した目標が達成できる。

3. エミュレータ

本項では、DASPの信号処理プログラムの開

発や評価に使用するエミュレータについて述べる。

エミュレータは、DASPチップ2個を搭載した本体と、A/D (以下ADC)、D/Aコンバータ (以下DAC) を内蔵したコンバータモジュール、エミュレータを制御するための制御プログラムで構成している (図-2)。エミュレータを用いることで、DASP用信号処理プログラムを実際に動作させ、オーディオ信号を処理し、評価することができる。

3.1 ハードウェア構成

エミュレータはDASPチップと、チップを動作させる上で必要な周辺回路の全てを含んでおり、単体で動作するので、機器設計と並行したソフトウェアの開発が可能である。

また、汎用性を重視して、DASP2個をカス

表-1 開発支援システムの構成要素とその内容

名 称	内 容
入門解説書	DASPの機能や、使用方法の概要を解説したマニュアル。
フィルタ係数算出プログラム	楕円形IIRフィルタ、直線位相FIRフィルタの係数算出プログラム。
フィルタ設計・シミュレーションプログラム	バターワース・チェビシェフ形IIRフィルタの設計およびシミュレーション・プログラム。
FFTプログラム	数値データに対する汎用のFFTプログラム。
クロスアセンブラ	DASP用クロスアセンブラ。
スタンドアロン・エミュレータ	複数のDASPとその周辺回路の全て (拡張メモリ、プログラムメモリ、ADC、DACなど) を含んだシステム。主としてオーディオ・アプリケーションの開発に用いる。
エミュレータ制御プログラム	スタンドアロン・エミュレータを制御し、DASPのプログラム実行、デバッグ、ユーザアプリケーションの開発を支援するプログラム。
ICE機能エミュレータ	DASPエバチップのデバッグ用端子のみを制御し、周辺回路はユーザの用意したものを用いる。プログラム・メモリ・ボード必要。各種アプリケーションに対応可能。
プログラムメモリ・ボード	DASPエバチップのデバッグ用端子に接続し、P-ROM、EP-ROM、RAMを搭載可能。
ICE制御プログラム	ICEを制御する制御プログラム。
ターゲット・ボード	マスクROM、プログラムROMボードを用いたDASP評価用の小型ボード。
プログラムROMボード	P-ROM専用のプログラム・メモリ・ボード。



図-2 エミュレータ
Fig. 2 Emulator.

カード接続した構成をとっており、1チップまたは、2チップ構成のアプリケーション開発ができる。

図-3にブロックダイアグラムを示す。

エミュレータはDASPのマスタークロックを生成するクロック回路、DASP制御回路を初め、外部データメモリや外部プログラムメモリと

その周辺回路、ADC、DAC等を含んでおり、パーソナル・コンピュータを用いて外部から制御する。

3.2 機能

表-2にエミュレータ・ハードウェアの機能一覧を示す。これらの機能により各種オーディオ用アプリケーション開発に対応できる。

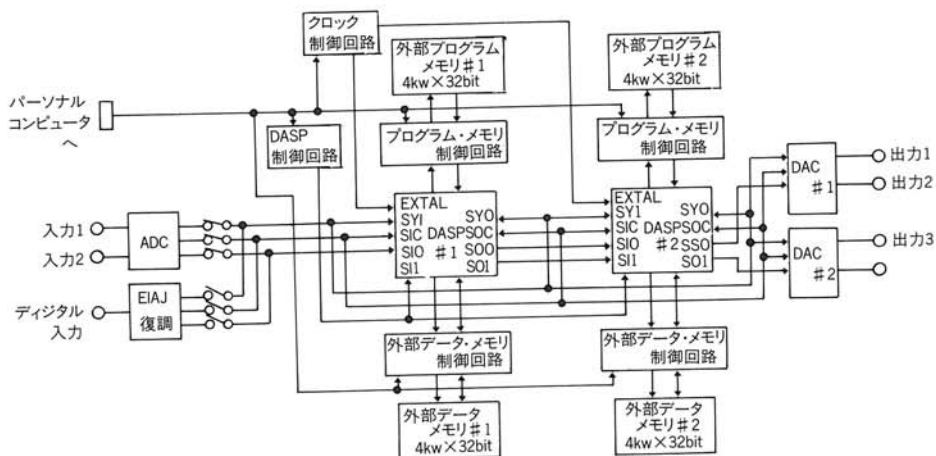


図-3 エミュレータ・ハードウェア・ブロックダイアグラム
Fig. 3 Block diagram of an emulator.

表-2 エミュレータ・ハードウェア機能一覧

機 能	内 容
DASP制御	ホールト、ステップ実行、連続実行割り込み制御、マスタークロック制御
DASP内部レジスタアクセス	演算レジスタ (A, B, D, P, SFT, SP)、アドレス演算レジスタ (B0, B1, X0, X1, PGI, VSM)、特殊用途レジスタ (MOD. ST, MASK)、周辺レジスタ (BLSI, DPSI, AMC, PI) 上記レジスタの読み書き
メモリアクセス	外部プログラムメモリの読み書き、DASP内部メモリの読み書き、DASP外部データメモリの読み書き
入力	アナログ入力 2ch デジタル入力 1系統 (同軸)
出力	アナログ出力 4ch
サンプリング周波数	48 kHz, 44.1 kHz 系列で任意

4. エミュレータ・ソフトウェア

本項では、エミュレータのソフトウェアに関して述べる。

エミュレータ本体を制御し、効率的なソフトウェア開発を可能にするのが、エミュレータ・ソフトウェアである。図-4に今回開発したソフトウェアの機能を示す。

DASPのソフトウェア開発は、設計者と、パソコン上で動作する開発支援システム（制御ソフト）との対話によって行われる。従って、この間のインターフェースがシステムの良さを決定するといっても、決して過言ではない。

今回のシステム開発に当たっては、ソフトウェア設計者が、最も使いやすいシステムを実現するために、このようなヒューマン・インターフェースに重点をおいた。

具体的には、視覚的に理解しやすく、使い勝手の良い、しかも、操作を熟知しなくても使用できるマルチウィンドウや、プルダウンメニューを採用した。

以下、これらのヒューマン・インターフェースについて述べる。

4.1 マルチウィンドウ

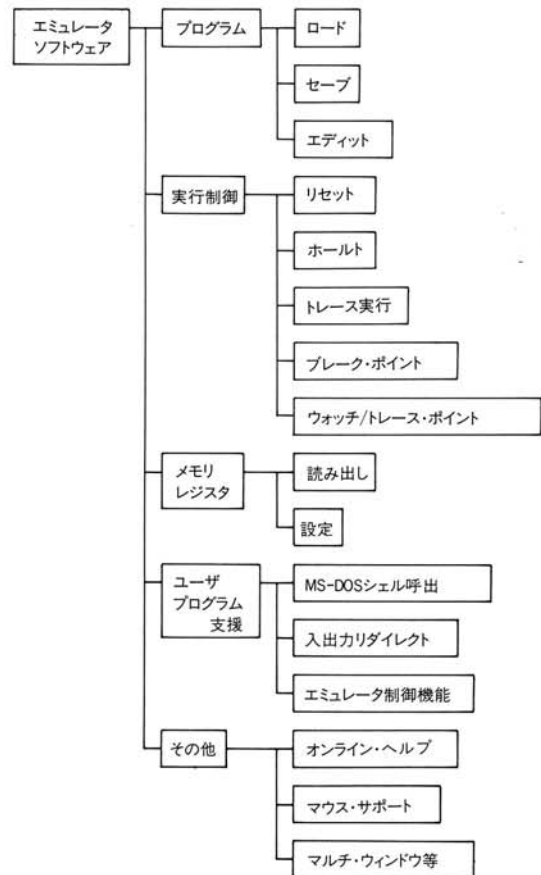


図-4 エミュレータ・ソフトウェア機能概要

Fig. 4 A summary of control program function.

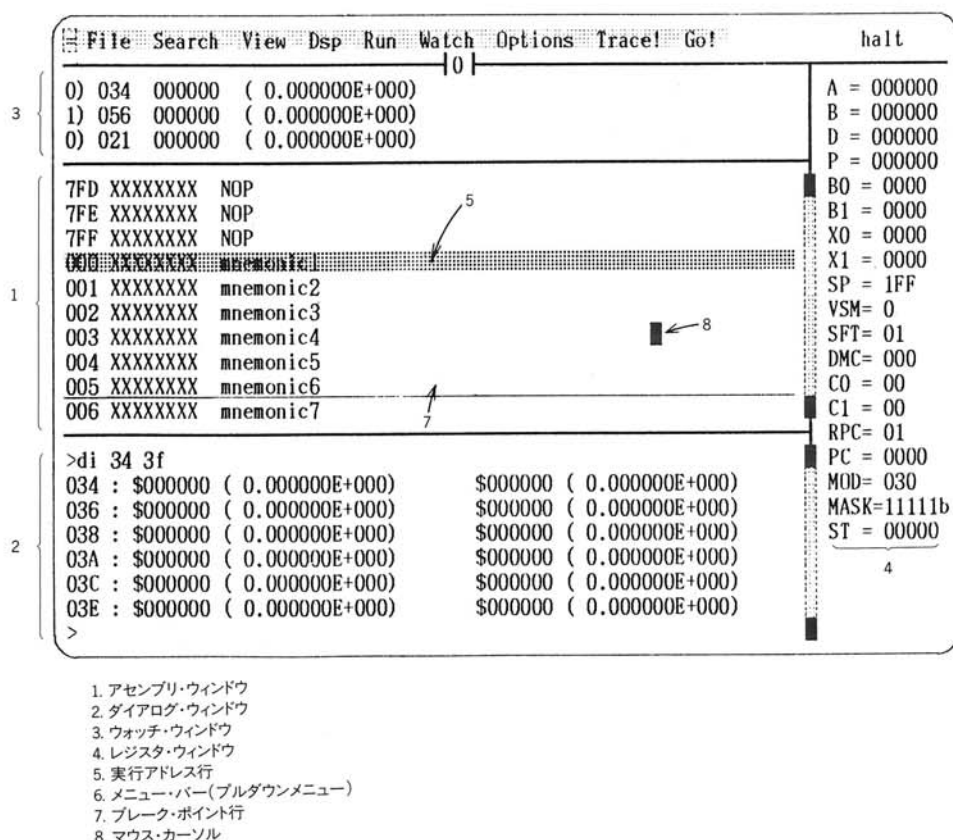


図-5 エミュレータ・ソフトウェア画面例
Fig. 5 A display example of control program.

これまでのソフトウェアは、使用者の入力や、制御ソフトウェア側の表示出力が、画面の上側から下側へと連続的に、しかも一様に表示されていた。また、画面一杯になると、先に表示した情報から順に消失するスクロール画面になっていた。

これに対し、今回開発したソフトウェアでは、一つの画面を、機能ごとに専用化したウィンドウに分割した。(図-5)

さらに、これらのウィンドウは独立、あるいは相互に関連を持ちながら動作・表示を行うようにした。この様な画面構成により、必要な情報が一目で把握でき、処理に対する適切な判断ができる様にした。

1) アセンブリ・ウィンドウ

このウィンドウはデバック中のプログラム (DSPプログラム) の逆アセンブルリストを常時表示すると共に、実行中のアドレスやブレークポイントなども表示色や表示属性 (アンダーラインなど) を変えて分かり易く表示するものである。行中のアドレスを表示しているのも、トレースを行なった場合など、プログラムの動いている様子が手にとるように分かる。(図-6)

また、プログラムが常に表示されているので実行中のプログラム内容が簡単に把握できる。

2) ダイアログ・ウィンドウ

このウィンドウは、入力されたコマンド入力のエコーバック、各種コマンドの実行結果などを表示する。入力されたコマンドや実行結果の履歴

000	0000A050	mnemonic1
001	0005A0D0	mnemonic2
002	000B0B85	mnemonic3
003	000F4BA5	mnemonic4
004	00138A10	mnemonic5
005	0017E010	mnemonic6
▽		
000	0000A050	mnemonic1
001	0005A0D0	mnemonic2
002	000B0B85	mnemonic3
003	000F4BA5	mnemonic4
004	00138A10	mnemonic5
005	0017E010	mnemonic6
▽		
000	0000A050	mnemonic1
001	0005A0D0	mnemonic2
002	000B0B85	mnemonic3
003	000F4BA5	mnemonic4
004	00138A10	mnemonic5
005	0017E010	mnemonic6
▽		
000	0000A050	mnemonic1
001	0005A0D0	mnemonic2
002	000B0B85	mnemonic3
003	000F4BA5	mnemonic4
004	00138A10	mnemonic5
005	0017E010	mnemonic6
▽		
006	000B0B85	mnemonic7
007	00138A10	mnemonic8
008	0000A050	mnemonic9
009	00138A10	mnemonic10
00A	0017E010	mnemonic11
00B	0005A0D0	mnemonic12

図-6 ステップ実行状態

Fig. 6 A state of step execution.

は、ソフトウェア内部のバッファに記録しており表示内容がスクロールして表示範囲を越えても簡単な操作で、さかのぼって確認することが可能である。

従って、コマンドをメモしたり、プリンタに出力するといった、実行状態の記録が不要になり負担が軽減できる。(図-7)

3) ダイアログ/メッセージ・ボックス

これらのボックス(ウィンドウ)は、通常の画面には表示されない、パラメータの入力があった場合や、DASPあるいは、本ソフトウェア上でエラーが発生した場合などに、必要に応じて画面中央に表示し、エラー内容を示したり、パラメータ等の入力を促す。(図-8)

入力やエラー確認が完了すると、画面から消えるので認識し易く、また、間違いも少なくなる。

4) ウォッチ・ウィンドウ/レジスタ・ウィンドウ

ステップ実行時、1ステップ実行(1命令実行)するごとにDASPの内部メモリやレジスタ内容を表示する。

これまでは、ステップ実行するごとに、メモリ内容を表示したり、レジスタの表示コマンドを毎回入力して結果を得ていたのに対し、この方法はDASPの実行に伴って、確認したい内部状態が一目で把握でき、命令実行や動作の検討が即座に行える。

4.2 プルダウンメニュー

ソフトウェアは、その機能が多くなればなるほどコマンドが多岐にわたり、操作が煩雑になる傾向が強い。本ソフトウェアも豊富な機能を有しており、全て、コマンドによる入力が可能だが、いきなり多くのコマンドを使いこなすことは、事実上困難であり、誤操作の原因ともなる。

そこで、使用頻度の高いコマンドに関しては、マウスや簡単なキー操作によりプルダウンメニューと呼ばれるメニューを選択するだけで、実行出来るようにした。

プルダウンメニューは、通常は図-4に示す様に、メニュー・バーに大項目だけが表示されており、これをマウス等で選択すると、小項目のリストが表示される。この小項目の一つを選択すればコマンドが実行できる。

他の作業を行っている時は、メニュー・バーの表示だけなので支障にはならない。

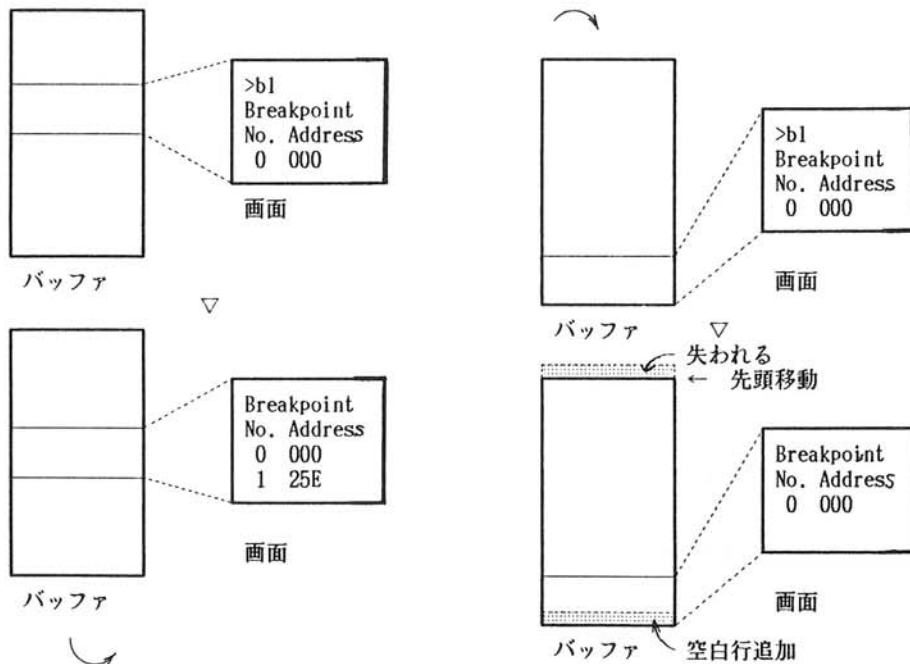


図-7 ダイアログ・ウィンドウ表示方式

Fig. 7 A display example of dialog window.

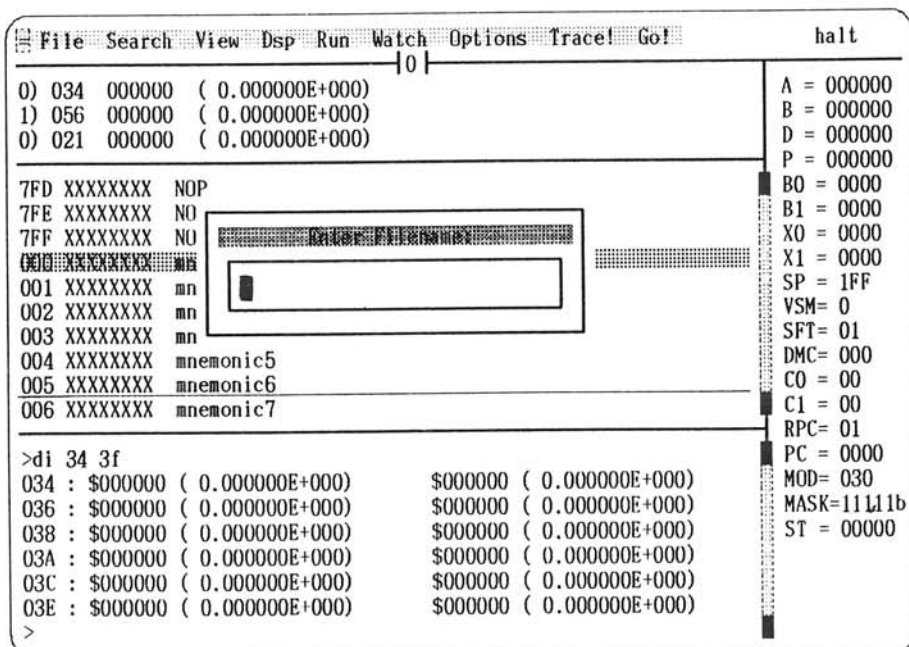


図-8 ダイアログ・ボックス表示例

Fig. 8 A display example of dialog box.

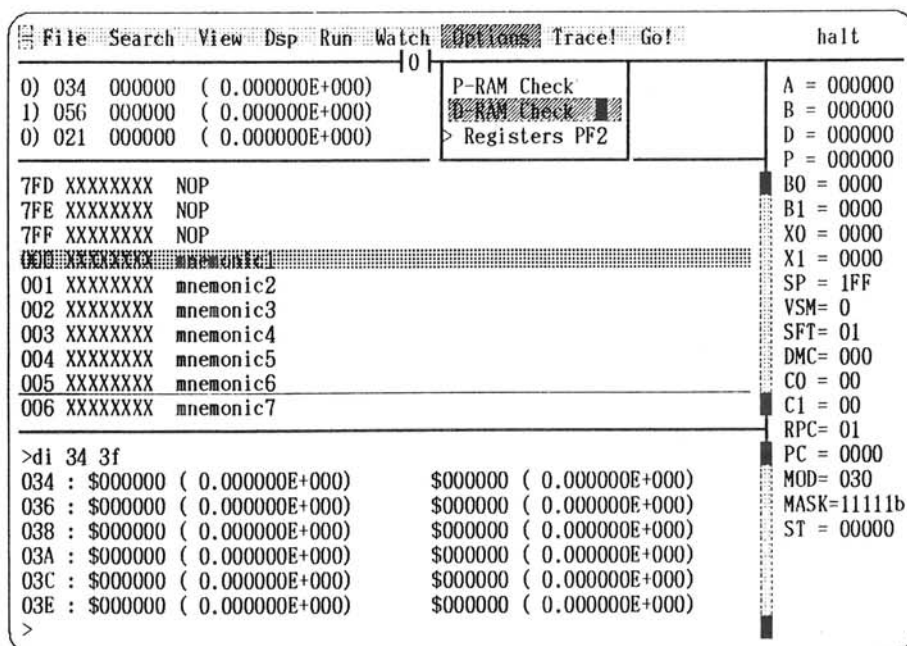


図-9 プルダウンメニュー選択例

Fig. 9 A example of pull-down menu selection.

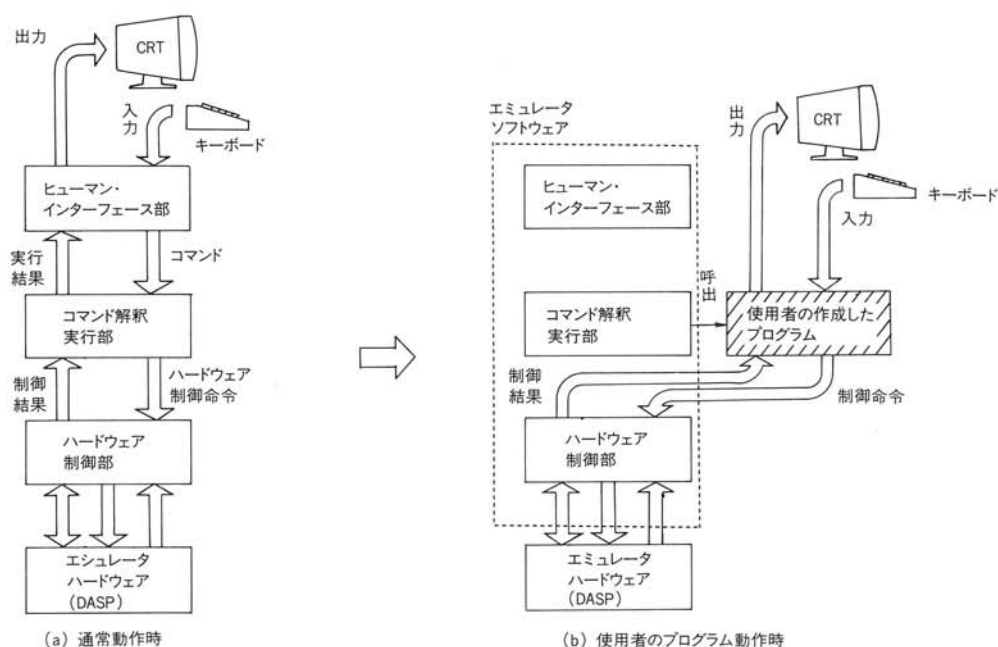


図-10 プログラム支援機能概要

Fig. 10 A state of user program support function.

また、マウスを用いているので素早い選択が可能である。表示および選択の例を図-9に示す。

マウスについては、プルダウン・メニューの選択以外にも、ブレークポイントの設定や、ウィンドウ操作などにも使用できるようにした。

4. 3 その他の機能

本ソフトウェアは、従来には無かった機能として、設計者が作成したDASP制御プログラムとエミュレータ制御プログラムが共存できる機能を設けた。これは、前述した、ヒューマン・インタフェースとは異なり、プログラム間のインタフェースとも言えるべきものである。この機能をプログラム支援機能と呼んでいる。図-10に動作の様子を示す。

これまでの、エミュレータは、専用ハードウェアとソフトウェアを組み合わせる事が前提であったため、設計者の作成したプログラムで、エミュレータ・ハードウェアを制御するのは容易ではなかった。

そこで、エミュレータ・ソフトウェアの内部をヒューマン・インターフェース部、コマンド実行部、ハードウェア制御部の三つに分け、この中のハードウェア制御部を、設計者が作成した制御プ

ログラムから簡単に呼び出せるようにした。

これによって、DASP内部レジスタやメモリの読み書き、プログラムの実行制御が、ハードウェアを意識せずにできるので、DASPプログラムの評価・デバッグがスムーズに行えるようになった。

5. ま と め

以上、DASPソフトウェア開発支援システムの概要と、エミュレータ・ハードウェア、ソフトウェアについて述べた。当社では、これらのツールを用いて、これまでにDASPチップ自体の評価や、音場制御、イコライザ等のオーディオ信号処理プログラム開発を行った。

本システムを用いることで、DASPプログラムの基本設計からシミュレーション、さらに、実際のチップを用いたエミュレーションや、特性評価まで行うことができる。また、強力なヒューマン・インターフェースも備えており、使い勝手の良いシステムであると自負している。

本ツールを用いて効率的に開発されたDASPアプリケーションが多くのオーディオ製品や、車載用電子機器に搭載され、普及する事を切望する。