

「DSP」という略語には、「Digital Signal Processor」と「Digital Signal Processing」の意味があります。

一般的に「DSP」という場合は、前者のDigital Signal Processorを指しますが、ユーザー側から見た利用技術の面から後者のDigital Signal Processing機能が重要視される場合が多く、同意語的に使われるようになってきました。

DSP (Digital Signal Processor) : デジタル演算処理装置でCPUの一種

デジタル信号処理を行うための演算処理装置でCPUの一種です。過去においては、信号処理で非常に多く使われる積和演算を高速処理する為、乗算 + 加算機能を別のハードウェアで構成して高速化を実現していましたが、CPUモジュールの高速化にともない積和演算をCPU自体で実行する方向に変化してきました。

DSPは、一般的に入出力段に高速のAD変換器やDA変換器などを搭載し、信号処理系や制御系などの中核として使われます。また、HDDやROMなどからプログラムを書き換えるだけで、多種多様な機能を実現することが可能となります。使用目的の変更や機能を追加したい場合など、ソフトウェアの変更・追加で目的の演算処理が実現できるよう柔軟性をもってます。

DSP (Digital Signal Processing) : デジタル信号を処理する

「信号をデジタルで処理をする」という意味があります。従来、計測や制御などに使われる信号処理のほとんどはアナログ技術でした。アナログ回路では、調整が難しく時間がかかり、複雑な処理では回路が大規模になるという難点があります。また、温度や時間経過などの環境変化により回路部品が影響を受け、回路全体の特性が変化する弱点がありました。

デジタル信号処理は、必要な回路をプログラム表現できるものであれば基本的にどんな構成でも実現が可能となり、経年変化や温度など環境変化にも影響を受け難く信頼性が大幅に向上し、回路調整の自動化も可能となり、さらにアナログ回路では難しい対象によって変化する適応フィルタや複雑な非線形制御なども容易に実現ができるようになりました。

参考 Analog と Digital の関係

例えば、気温は時間と共に連続的に変化しています。このように連続的に変化する情報を扱うのがアナログ(Analog)です。アルコールを用いた温度計では、アルコールの体積の変化という形を便宜的に作られた目盛りから読み取ります。このような表示法がアナログ表示です。気温を数値として直読できるタイプのものは、温度センサーが検出した信号を一定時間ごとに数値に変換する仕組みが組み込まれています。このように一定時間ごとの離散的な(間隔を持った)情報を扱うのがデジタル(Digital)です。この場合の気温はデジタル量(数値)として表され、気温が直読できる表示形式をデジタル表示といいます。

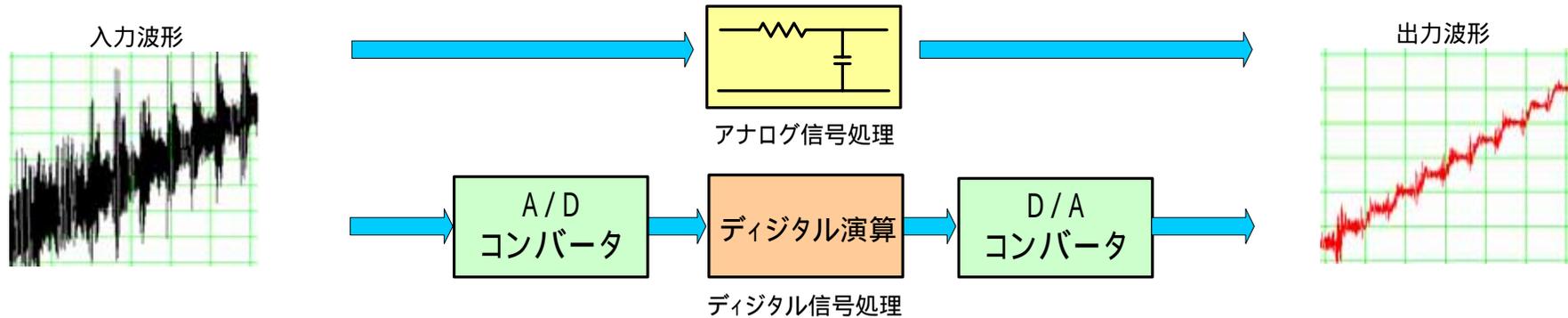
アナログ信号処理とデジタル信号処理の違い

一般的にアナログ回路に比較して、デジタル回路では経年変化が少ない、調整が簡単、高精度、高信頼性などが挙げられます。

	アナログ信号処理	デジタル信号処理
信号情報の形	電圧や電流といった絶対値を保証した連続する信号	“0”と“1”で表現する2つの信号レベル(電圧がある値より高いか低いかといった絶対値を必要としない表現)の信号
信号処理精度	0.5 ~ 5%程度 (回路部品、回路構成などによる)	0.001 ~ 0.1%程度 (デジタル処理部分のみを考えるとCPUの演算誤差。計測・制御の全体から考えると入出力にA/DやD/Aで構成する為、一部にアナログ回路系が存在し精度の低下要因)
環境に対する影響	電圧や電流といった絶対値で信号を扱うため、回路部品の温度変化や経年変化などに影響を受けやすい。	“0”と“1”のレベル信号を扱う為、回路部品の温度変化や経年変化などがあっても信号レベル判定の許容幅が広く影響を受け難い。
使われる部品	増幅器(アンプ)を主体としたアナログ回路部品 (回路の一部にデジタル回路を組入れる場合もある)	“0”か“1”のレベル信号を伝えるためのデジタル回路部品と処理手順を記述したプログラム(ソフトウェア)。
回路設計・変更など	目的に合わせて回路部品で構成する。変更に伴う回路の追加や構成変えを物理的な回路部品で組換えが必要	CPUとソフトウェア(プログラム)で構成する。変更に伴う回路の追加や構成変えもソフトウェアの部分変更で可能。
非線形・適応処理	回路が複雑になり、構成や精度保証が難しい。	CPUでプログラム表現できる処理は、基本的にどんな構成でも実現が可能。非線形処理や適応(アクティブ)処理なども実現できる。
回路の調整	回路部品には一定の範囲で性能にバラつきがある為、設計値に合わせる為の調整が必要。回路の規模によって使用する部品も多くなり、調整箇所が増える。	回路調整の殆んどをソフトウェアで行なうことができ、定数などの変更も演算プログラムの数値を変えることで調整できるため、調整の処理ソフトを組み込むことで自動調整も可能になる。
信頼性	低い (回路部品の種類や個数が多く、調整箇所が多い。温度や経年変化を受けやすいため、性能維持が難しい)	高い (ハードウェアとソフトウェアに分離して考えることができる。ハードウェアは、CPUやA/D・D/Aなどの構成ユニットを量産する製造手法により単体の信頼性が高い。ソフトウェアは、イレギュラー処理(想定できない現象に対応する処理)能力の完成度を除いて通常の信号処理プログラムが完成すると、それ以後に変化する要因が皆無であるため信頼性が高い。

フィルター回路を例にした信号処理

アナログ信号処理とデジタル信号処理のシステム概念を示します。

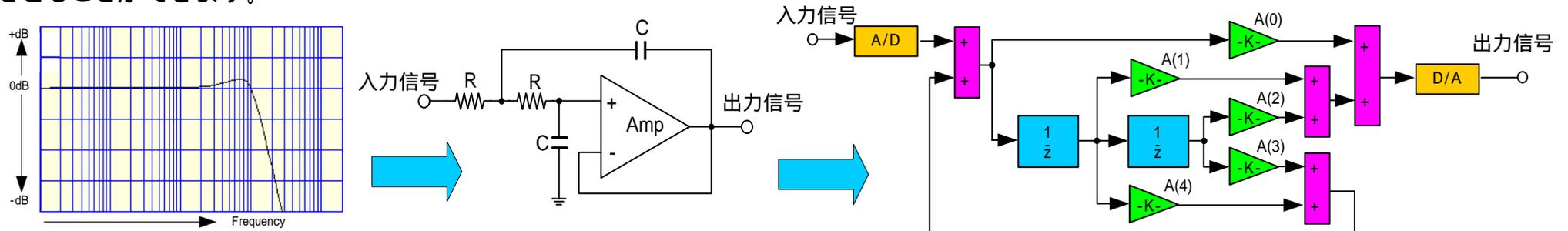


一般的に、回路は伝達関数(特性を数式で表したもの)で表せます。その伝達関数をR,C,Lなどの部品を使って実現したものがアナログ信号処理であり、CPUなどによって演算処理することで実現したものがデジタル信号処理です。

例えば、PID制御など微・積分回路で実現できる場合は、アナログ信号処理で十分な場合がありますが、ファジィ制御、AI制御、多項要素制御などといった場合には、アナログ信号処理では実現が困難になります。デジタル信号処理ならそのアルゴリズムをプログラム実装することで実現が可能になります。したがって、複雑な信号処理を行う時ほどDSPの威力が発揮されます。

音・振動解析などで多く用いられるフィルター回路の事例

ローパス・フィルター回路の設計事例です。下図のアナログ信号処理とデジタル信号処理(DSPによる)は、ほぼ等価な特性をもたせることができます。



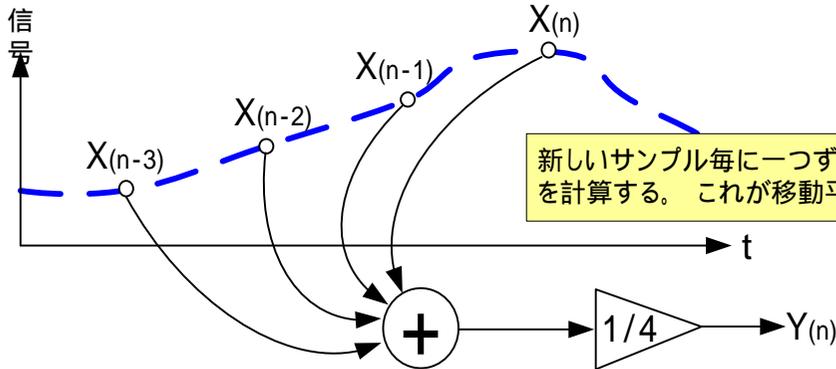
使いたいフィルターの特性(計画)

アナログ回路での実現 :
RやCの値、Ampのゲインなど調整

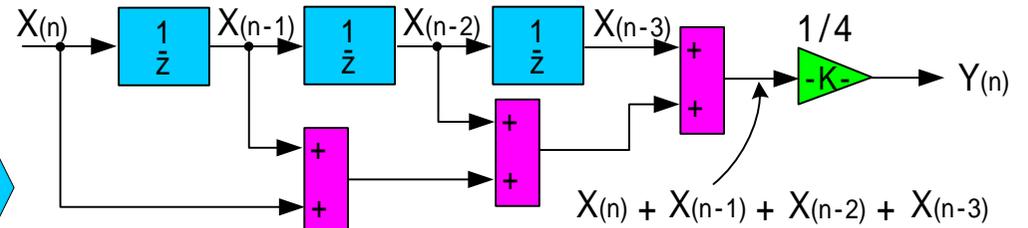
DSPを使用して実現 :
ブロック図表現した機能をソフトウェア的に演算処理する。アナログ回路のRやCなどに相当する値を演算の定数として与えることで自由度の高いデジタル信号処理ができます。

移動平均処理の手法を例にして

実際に使われる移動平均処理 (ローパス・フィルタの一種) をデジタル信号処理の例で説明します。



現在を含めた4サンプル移動平均処理の例

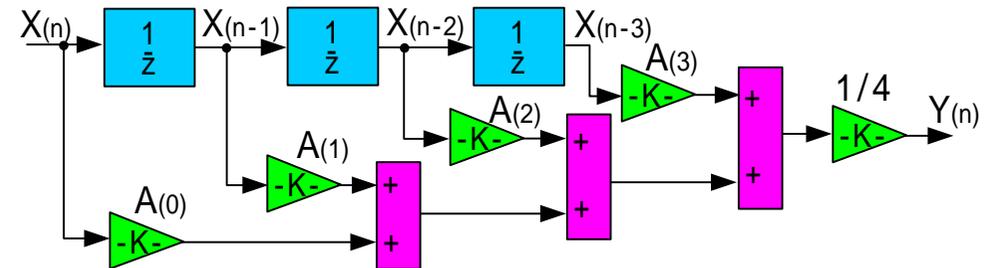


(Simulinkによる記述例)
移動平均処理ブロック図の例

この処理を1サンプル取り込むごとに実行するプログラムを実装することで、4サンプルによる移動平均処理 (ローパス・フィルタ) が実現できます。

応用例

また、右図のように $A(0) \cdot A(1) \cdot A(2) \cdot A(3)$ の機能ブロックを組み込み、定数の演算や遅延処理機能を追加することによって、様々なフィルタ (ローパス・ハイパス・バンドパスなど) を実現することもできます。定数や係数を自由に変更することも可能になります。



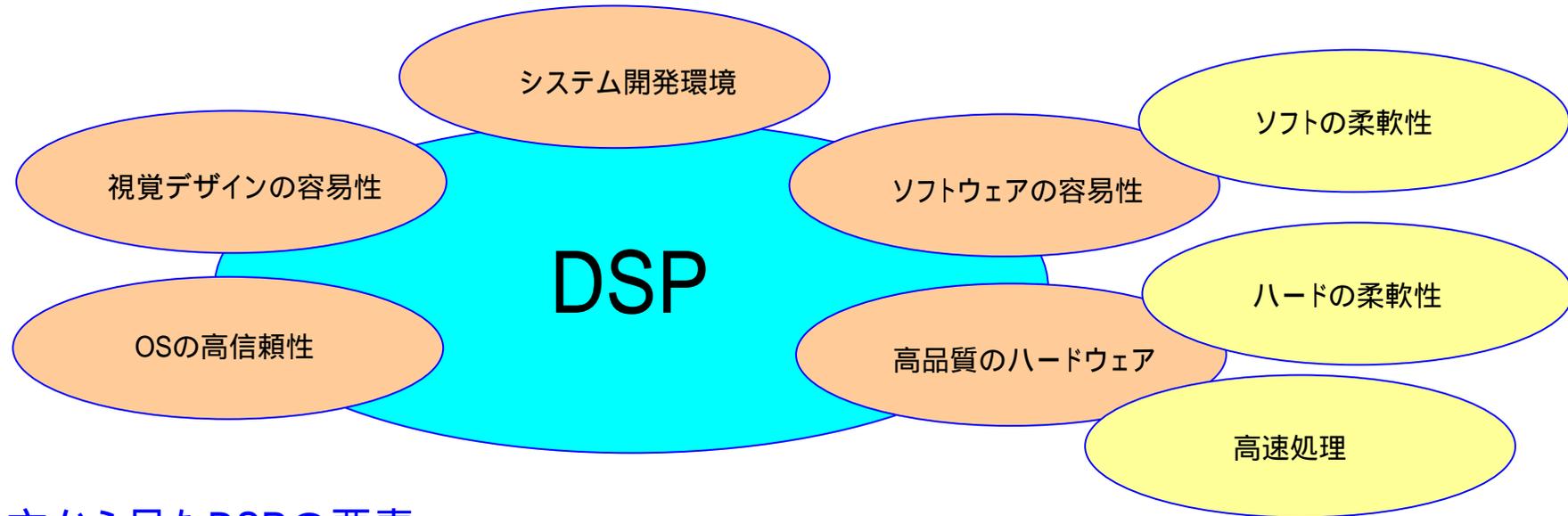
(Simulinkによる記述例)
多機能フィルタ・ブロック図の例

DSPが活躍する

DSPは、上図のブロック図で表された部分の数値演算を精度よく高速に演算処理する演算器として使います。実際のDSPには、それぞれのブロックに相当する機能ブロック (プログラム化した) がライブラリーとして用意されています。

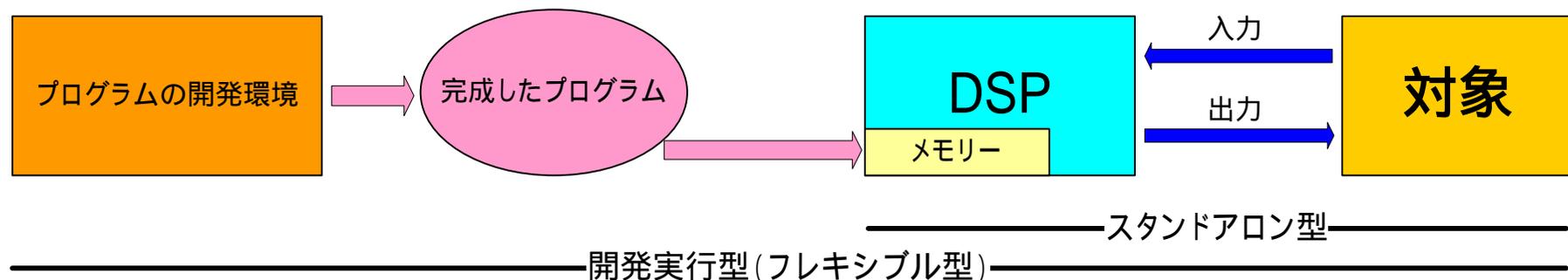
「DSP」に求められる代表的な要素

DSPの需要が高まるにつれて、ユーザーから様々な要求が出されています。下図は、その代表的な要素であり、A&Dは要求に応えられるDSPの提供と、より優れたDSPの開発に取り組んでいます。



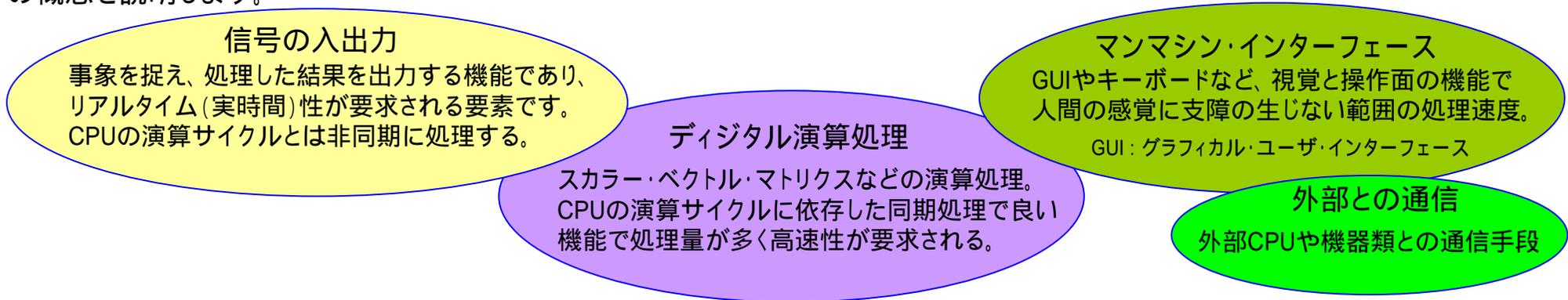
使われ方から見たDSPの要素

DSPの使われ方には、単体で使用するスタンドアロン型とプログラムを開発しながらフレキシブルに使用する開発実行型で大別されます。スタンドアロン型でもメーカー側から単品商品として提供する場合と、ユーザーがシステムの開発環境を持ち確定したプログラムでDSPを単品使用場合があります。このようにDSPの使い方自体のフレキシブル性も要求されます。



ハードウェアから見た要素

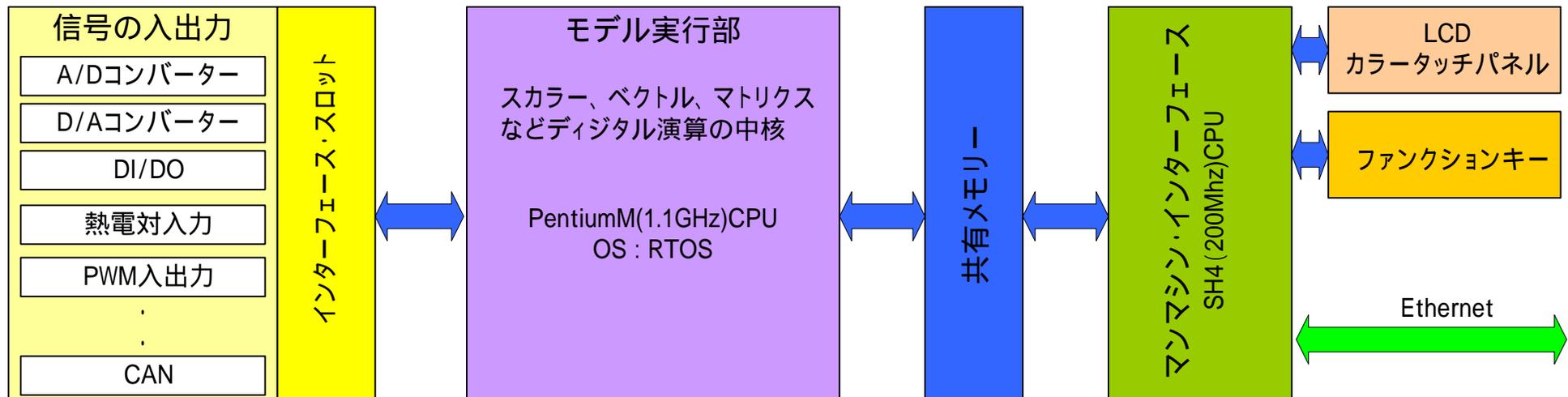
DSPの根幹であるハードウェアについて、デジタル信号処理の機能面から要求される内容とそれに対応するハードウェア構成の概念を説明します。



スタンドアロン型DSPの構成例(A&DのAD5435システム)

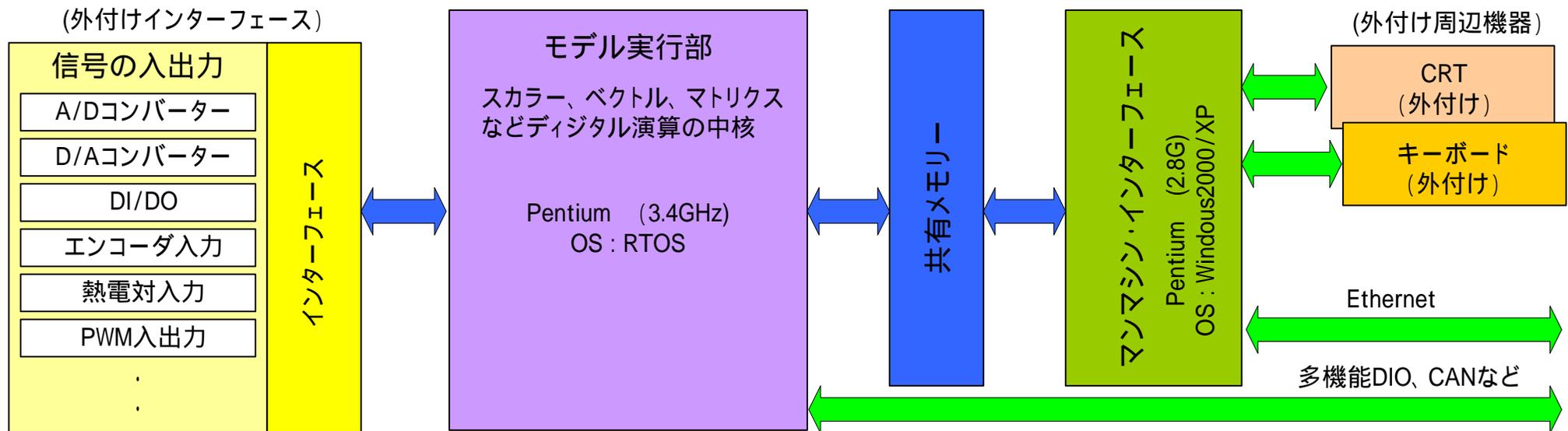
AD5435モデルは、デジタル演算の中核にPentiumM-CPUを採用し高速処理、信号入出力インターフェースにはFPGAを採用して非同期処理のリアルタイム性を確保、マンマシン・インターフェース部にはSH4-CPUを採用するDual-CPU方式によって高性能DSPを実現しました。AD5435は、スタンドアロン型の構成ですが外部にWindows2000/XPベースのパソコンを接続することで開発実行型(フレキシブル型)としても使用することができます。

注) FPGA : Field Programmable Gate Array



開発実行型DSPの構成例(A&DのAD5410システム)

AD5410モデルは、デジタル演算の中核にPentium -CPUを採用して高速処理を実現し、信号入出力インターフェースを外付けユニットにすることで大規模システム構成を可能にしています。非同期処理のリアルタイム性を確保にはFPGAを採用した回路構成になっています。マンマシン・インターフェース部にはPentium を採用するDual-CPU方式によって、Windows2000/XP環境(CRT・キーボードなどを外付けする)でVirtualConsoleやSimlinkの操作などが行え、開発・生産システムなどの構築ができる高性能DSPを実現しています。



- 注) RTOS : Real-Time Operating System
処理をリアルタイムに実行することを重視し、そのための機能を実装したOSのこと。
- 注) FPGA : Field Programmable Gate Array
プログラミングすることができるLSIのことです。マイクロプロセッサやASICの設計図を送りこんでシミュレーションすることができます。専用LSIより動作が遅く高価ですが、ソフトウェアで回路のシミュレーションを行なうよりは高速です。

ソフトウェアから見た要素

DSPのソフトウェアについて、ソフトウェア作成の容易性・柔軟性やGUIにおける視覚デザインの容易性、全体を管理するOSの信頼性などが要求されます。



ソフトウェアの容易性・柔軟性 : MATLAB/Simulinkによるオブジェクト感覚で

A&DのDSPは、デジタル信号処理に必要な機能ライブラリーが最も充実したMathWorks社の「MATLAB/Simulink」を採用し、その能力を最大限に発揮できるハードウェアとソフトウェア開発に特化しています。MATLAB(演算処理ライブラリー)とSimulink(ライブラリーを配置結線するソフト)の採用により、コンピュータの言語体系を意識しないオブジェクト指向(ブロック図感覚でソフトウェアを捕らえられる)でソフトウェア作成ができます。これにより、必要な部分のみブロック(ライブラリー)の追加変更をすることで目的の機能にカスタマイズできる柔軟な対応力を持ったソフトウェア体系になっています。

視覚デザインの容易性 : VirtualConsoleによる簡単操作

A&Dは、独自に開発したGUI(グラフィカル・ユーザー・インタフェース)ソフトウェア「VirtualConsole」を提供しています。このソフトウェアには、GUIの設計機能とSimulinkブロック図のパラメータ設定・変更・信号モニタ機能などがあり、見やすい操作・設定・監視画面が簡単に作成できます。VirtualConsoleの操作については、多数用意したライブラリーをドラック&ドロップ形式でコントロール画面に貼り付けるだけで実行できます。

OSの高信頼性 : リアルタイム処理に優れたRTOS採用

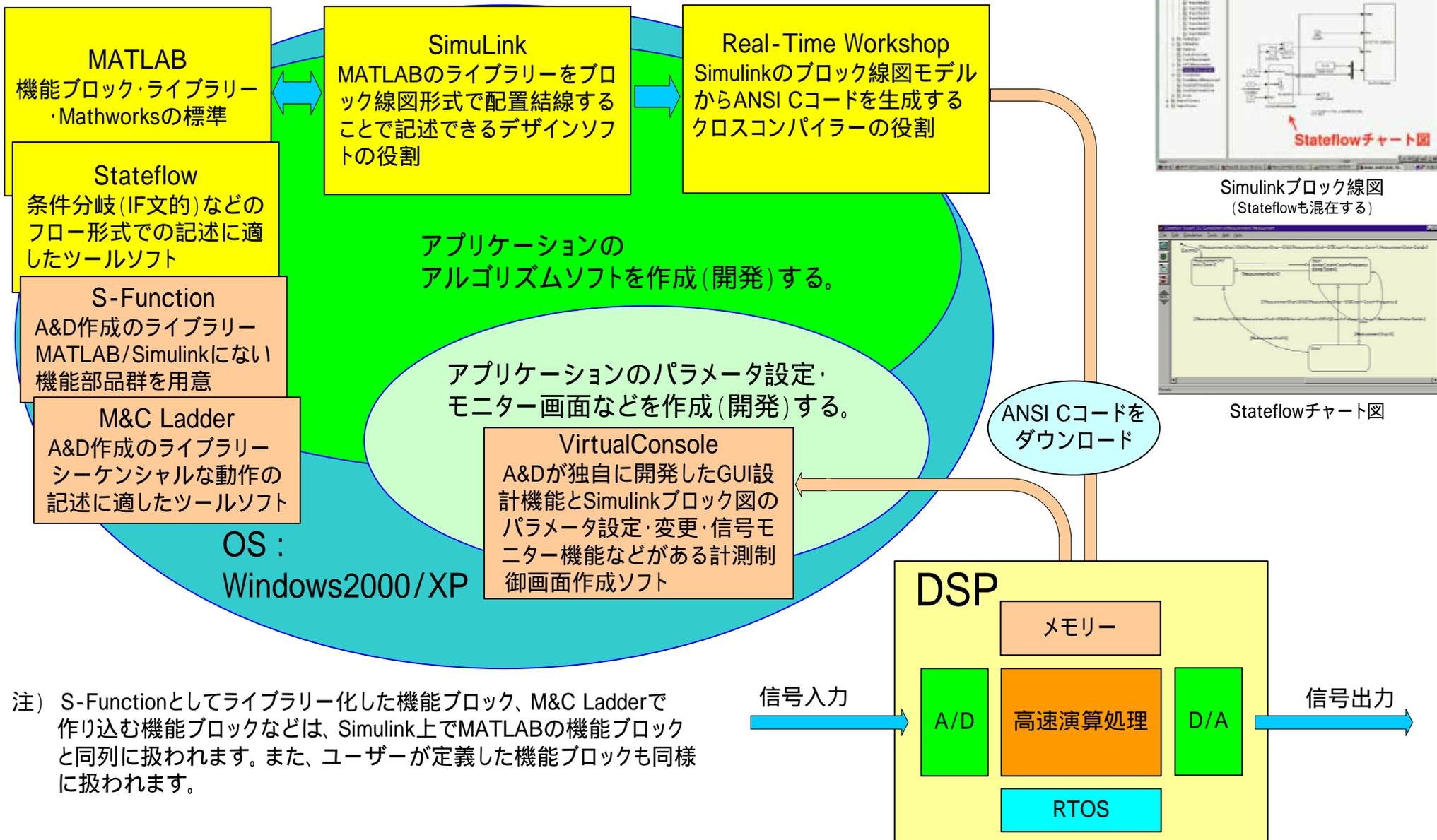
A&DのDSPは、リアルタイム処理を目的として開発された信頼性に最も優れたRTOSを採用しています。

注) OS (Operating System) : キーボード入力や画面出力といった入出力機能、ディスクやメモリの管理など、多くのアプリケーションソフトから共通して利用される基本的な機能を提供し、コンピュータシステム全体を管理するソフトウェアで「基本ソフトウェア」とも呼ばれます。

注) RTOS (Real-Time Operating System) : コンピュータ処理をリアルタイムに実行することを重視し、そのための機能を実装したOSのことです。計測や制御をするコンピュータシステムでは、応答時間が一定の範囲内にあることが要求されることが多く、OSにもリアルタイム性を実現するための様々な機構が必要とされます。そのため、リアルタイムOSには、必要な処理時間の予測を行なう機能や、複数の処理要求が同時に発生した場合でも目的の時間内に完了させるための機構を備えています。

それぞれのソフトウェアの役割

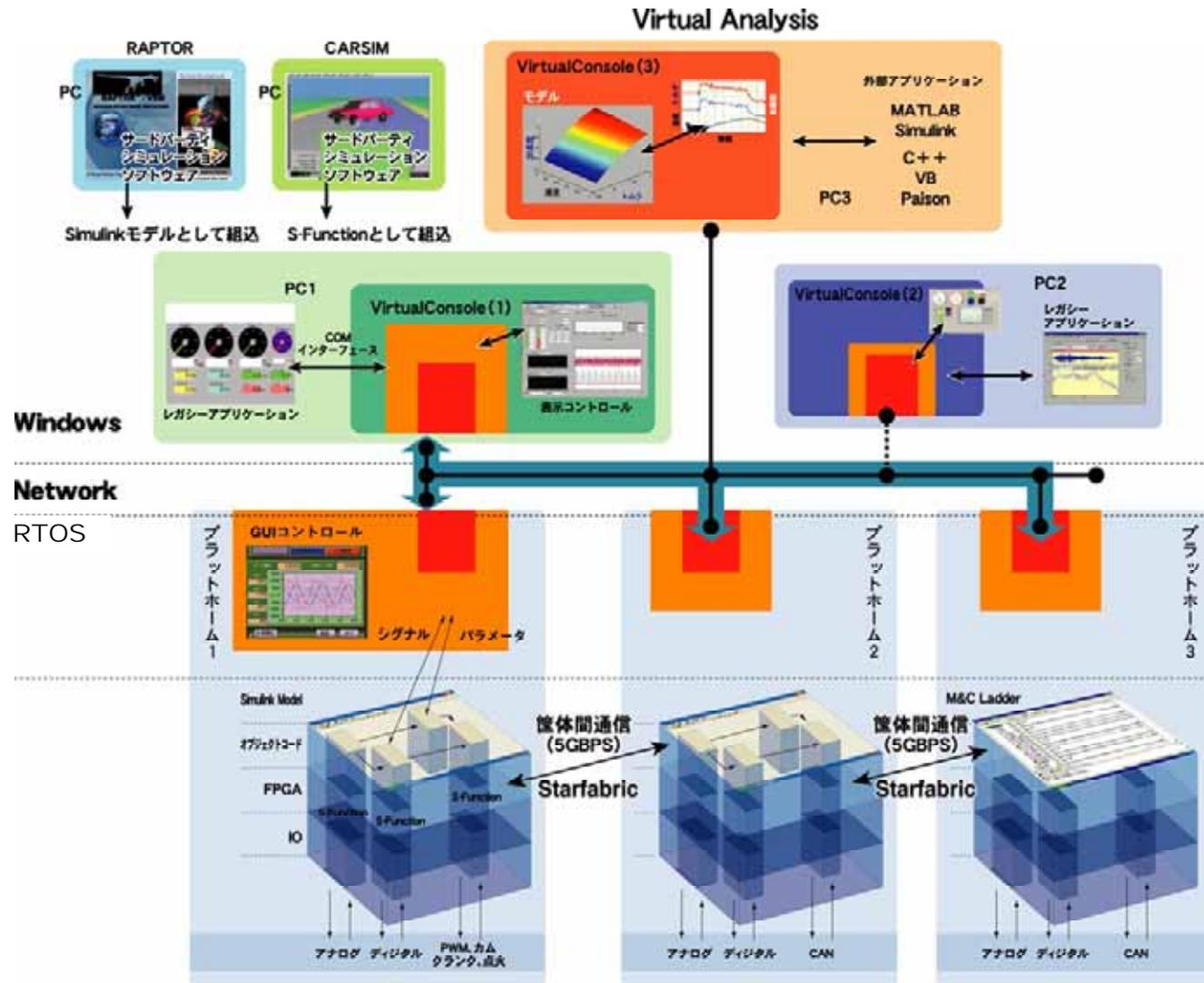
A&DのDSPに採用しているプラットフォームの主要なソフトウェアとソフトウェアの役割を説明します。



注) S-Functionとしてライブラリー化した機能ブロック、M&C Ladderで作成した機能ブロックなどは、Simulink上でMATLABの機能ブロックと同様に扱われます。また、ユーザーが定義した機能ブロックも同様に扱われます。

DSPシステムおよびソフトウェアの階層構造

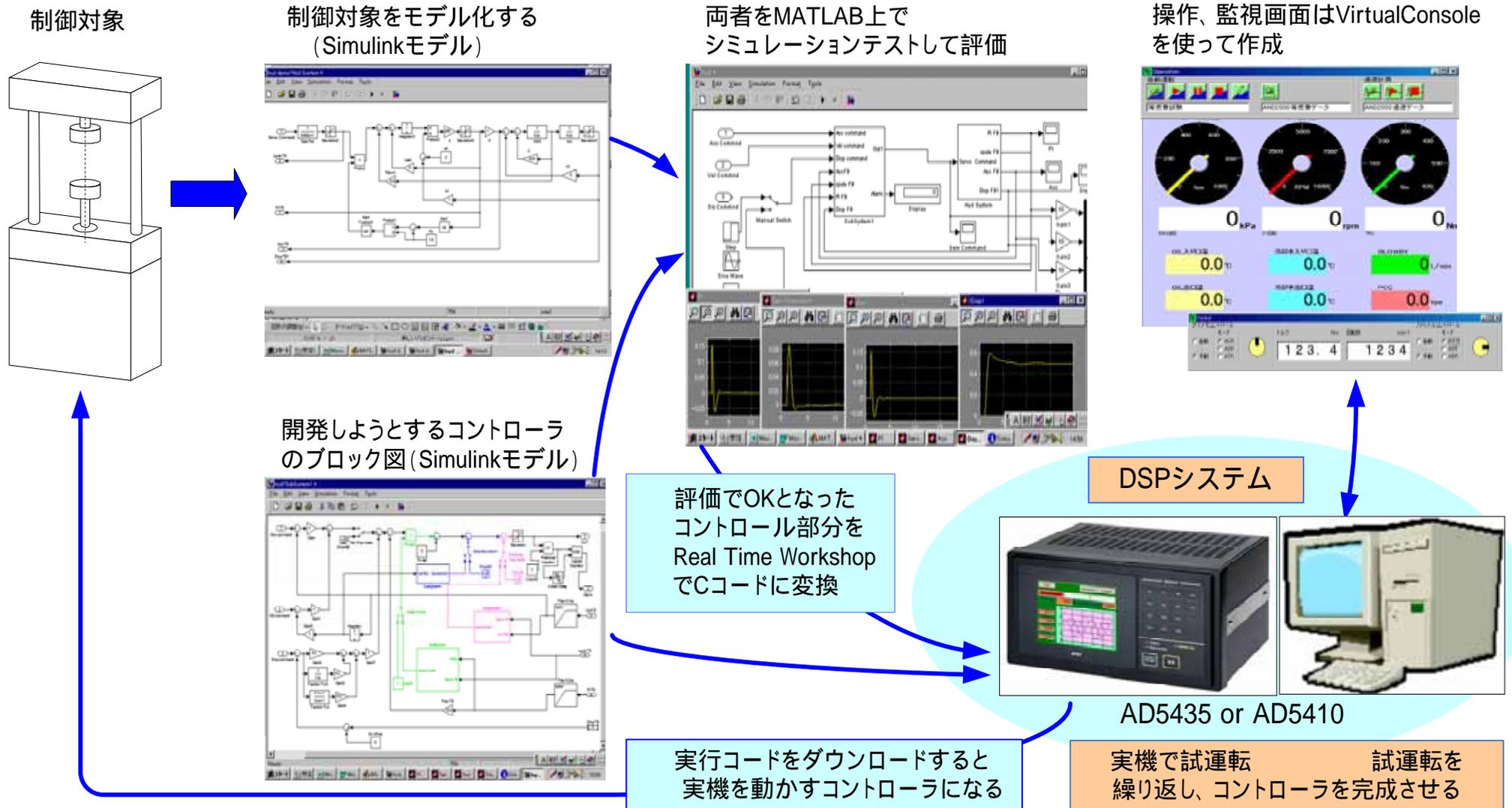
A&Dの提供するDSPシステムは、ハードウェア/ソフトウェアの両面において構築するシステム規模に対応できるように階層構造になっています。ハードウェア的には、入出力規模によってI/Oインターフェースの拡張や複数台のDSPを同時に使用できる筐体間通信機能を設け、規模に対する柔軟性を確保しています。また、ソフトウェア面でユーザーのオペレーションに近い部分は、汎用性の高いWindows2000/XP環境でNetwork結合できる構造にしています。



ソフトウェア作成の手法(1)

制御対象からDSPコントローラを作成する手順例

MATLAB/SimulinkとVirtualConsoleをベースにすることで、下図で示すようにソフトウェア作成の過程は殆どブロック線図を作る手法(オブジェクト指向)で完成までもって行くことができます。これは、ソフトウェア言語の専門知識をもたないユーザーにもソフトの作成ができるということです。また、記述言語で作られたソフトウェアに見られる「作った本人しかわからない」というような弱点が少なく、多くのユーザーが理解できる表現でのソフトウェア作成が可能という特徴があります。

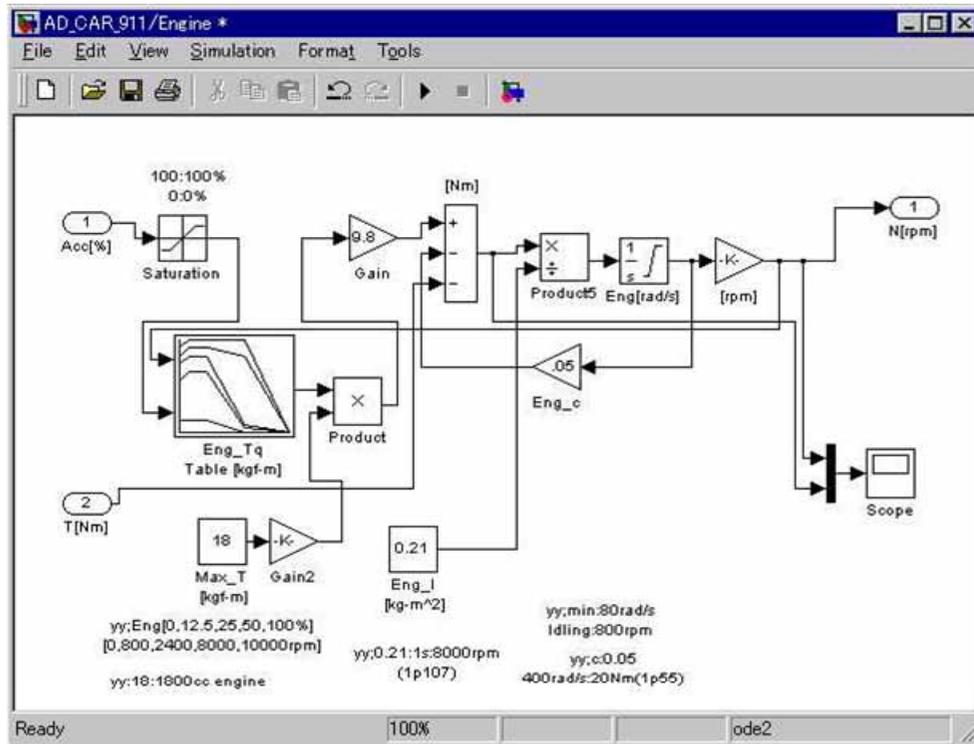


ソフトウェア作成の手法(2)

MATLAB/SimulinkとVirtualConsoleをベースにすることで

プログラムレス化を実現

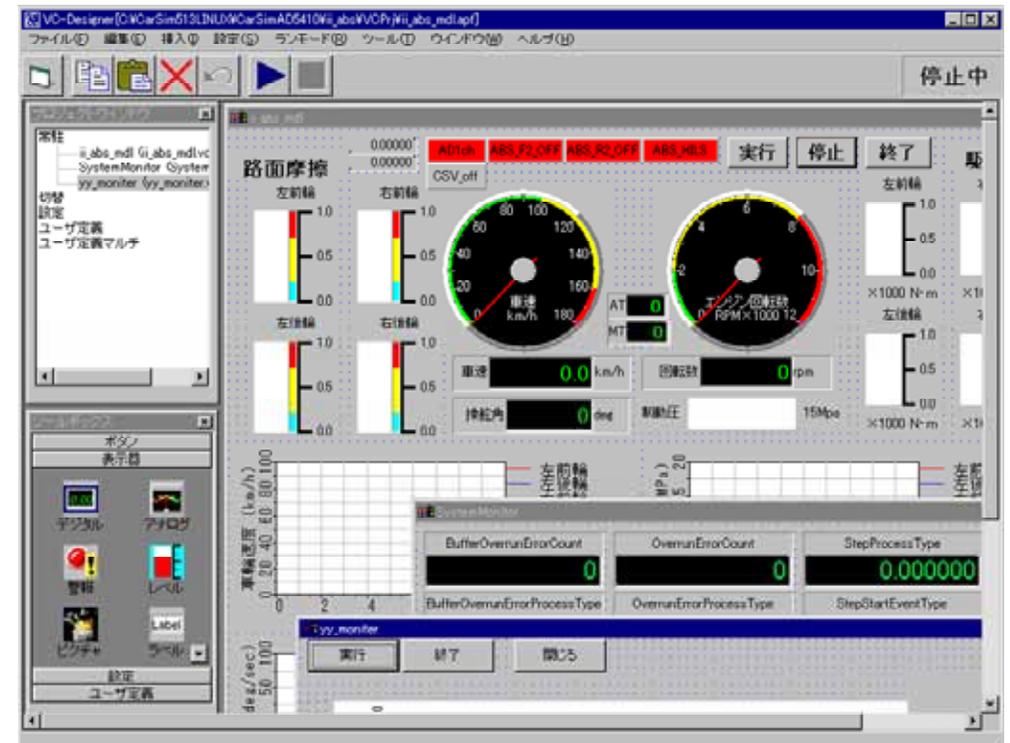
ブロック図を描くことでプログラム実行コードの生成が可能になり、複雑なプログラムを記述することなく、ユーザーによる可視性の高いフレキシブルなシステム作りが可能になります。



エンジンモデル・ブロック図の例

快適な操作環境

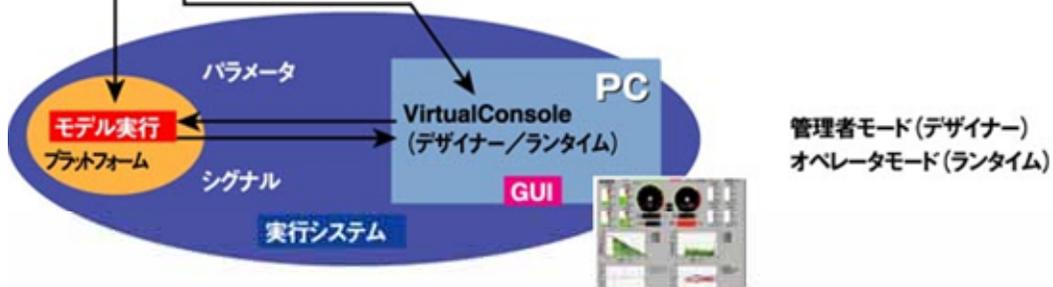
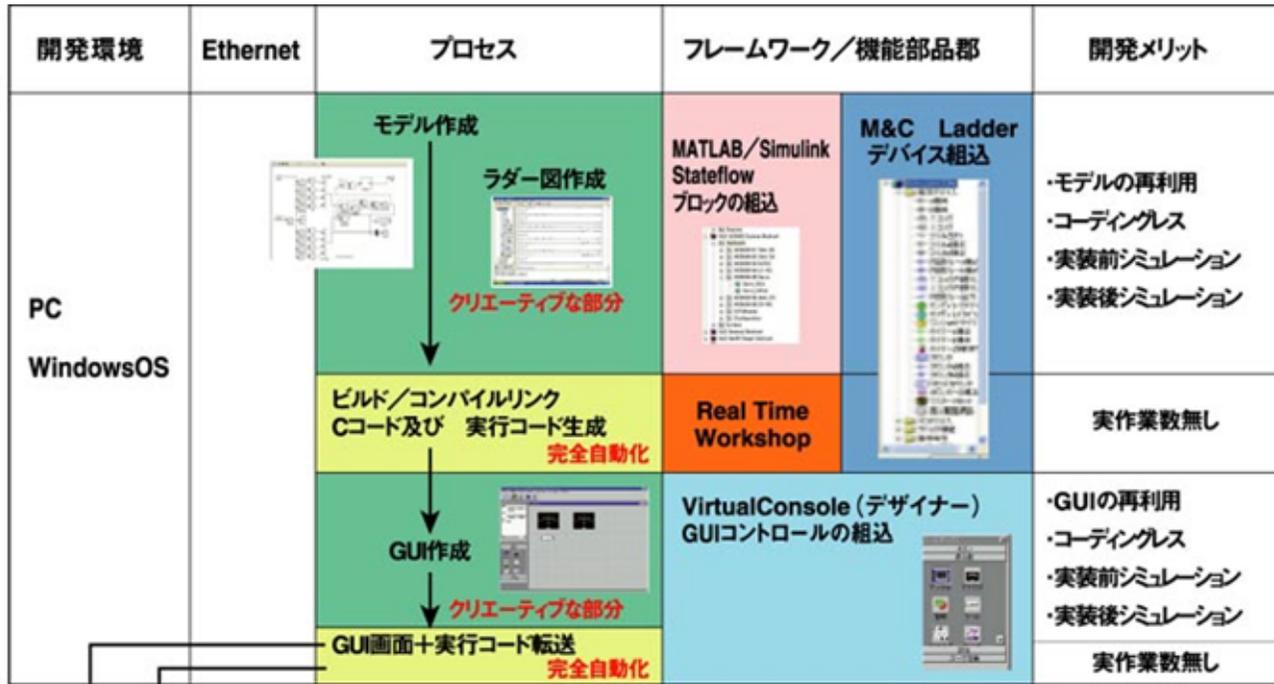
計測制御の条件設定や状況を監視する画面は、A&D独自開発のVirtualConsoleによって見やすくカスタマイズも容易です。



VirtualConsole画面の例

ソフトウェア作成の手法(3)

ソフトウェア開発のプロセス



機能別に見た開発プロセス一覧

機能	フレームワーク	機能部品群	カスタマイズ方法
ユーザインターフェース	VirtualConsole	GUIコントロール群	コントロールをGUIパネルに配置
後処理解析機能	MATLAB/Simulink/Stateflow	S-Function ブロックセット	モデル作成
リアルタイム処理機能	MATLAB/Simulink/Stateflow M&C Ladder	S-Function ブロックセット ラダー回路図デバイス群	モデル作成 ラダー図作成
超高速リアルタイム処理機能	MATLAB/Simulink/Stateflow VHDL開発環境	VHDL部品群	モデル作成

DSPのソフトウェア開発環境

DSPのフレームワークは、MATLAB/Simulink/Stateflowを中核にし、サブフレームワークとしてVirtualConsole・M&C Ladderを使用して機能実現します。これらのフレームワークを使用して作成したモデル図(ブロック図)、ラダー図は、Real Time WorkshopやM&C Ladderで自動的にCコード変換されターゲット(DSP)に実装できます。

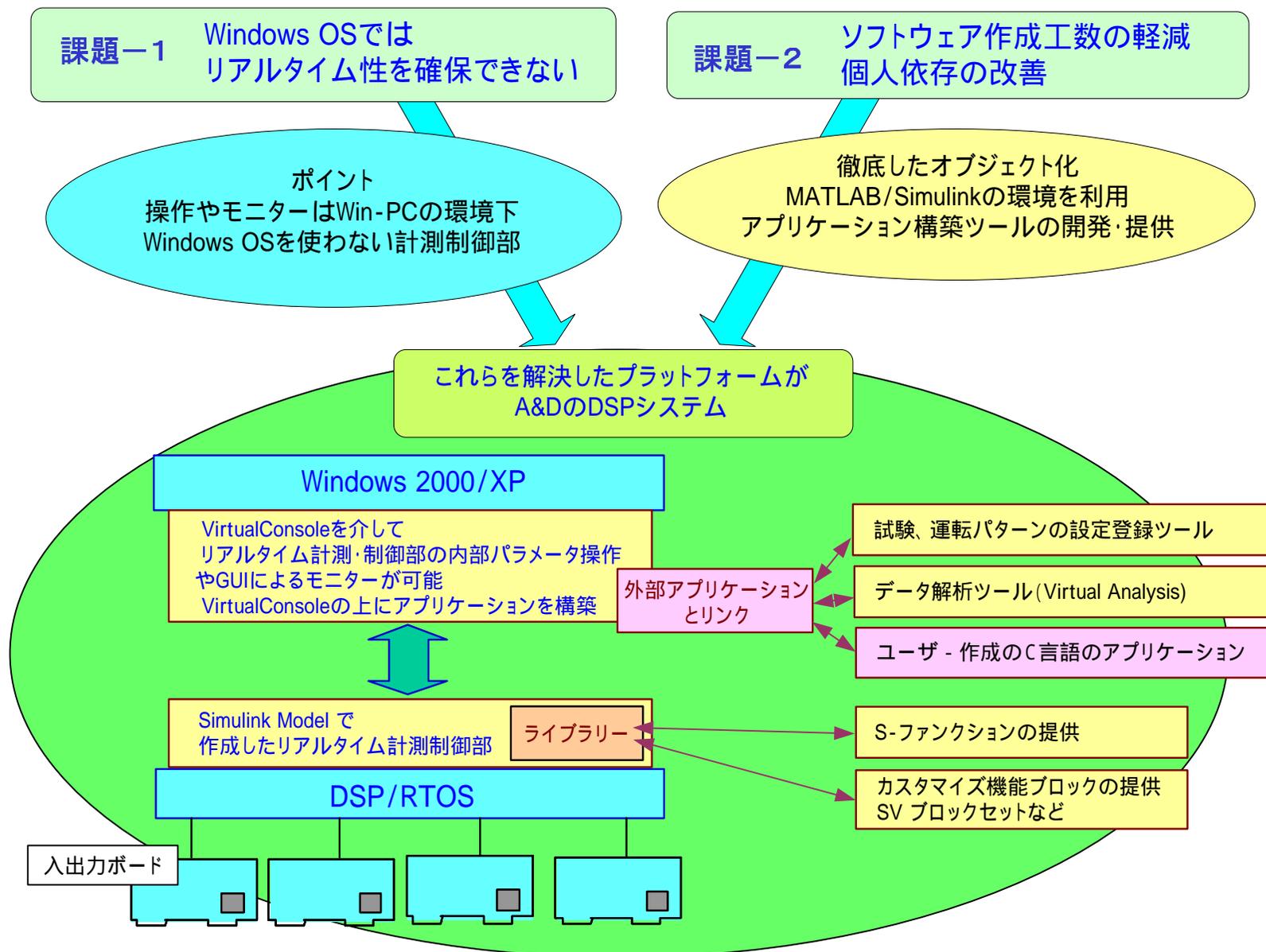
ユーザー・インターフェースもVirtualConsoleにより、コントロール部品を画面に配置する直感的な方法でレイアウト作成ができます。また、レガシーアプリケーション・ソフトウェアもCOMインターフェースにより、リンクでき過去の知的財産の活用も可能にしています。

A&Dが提供する内部オブジェクトモデル

オブジェクトモデル

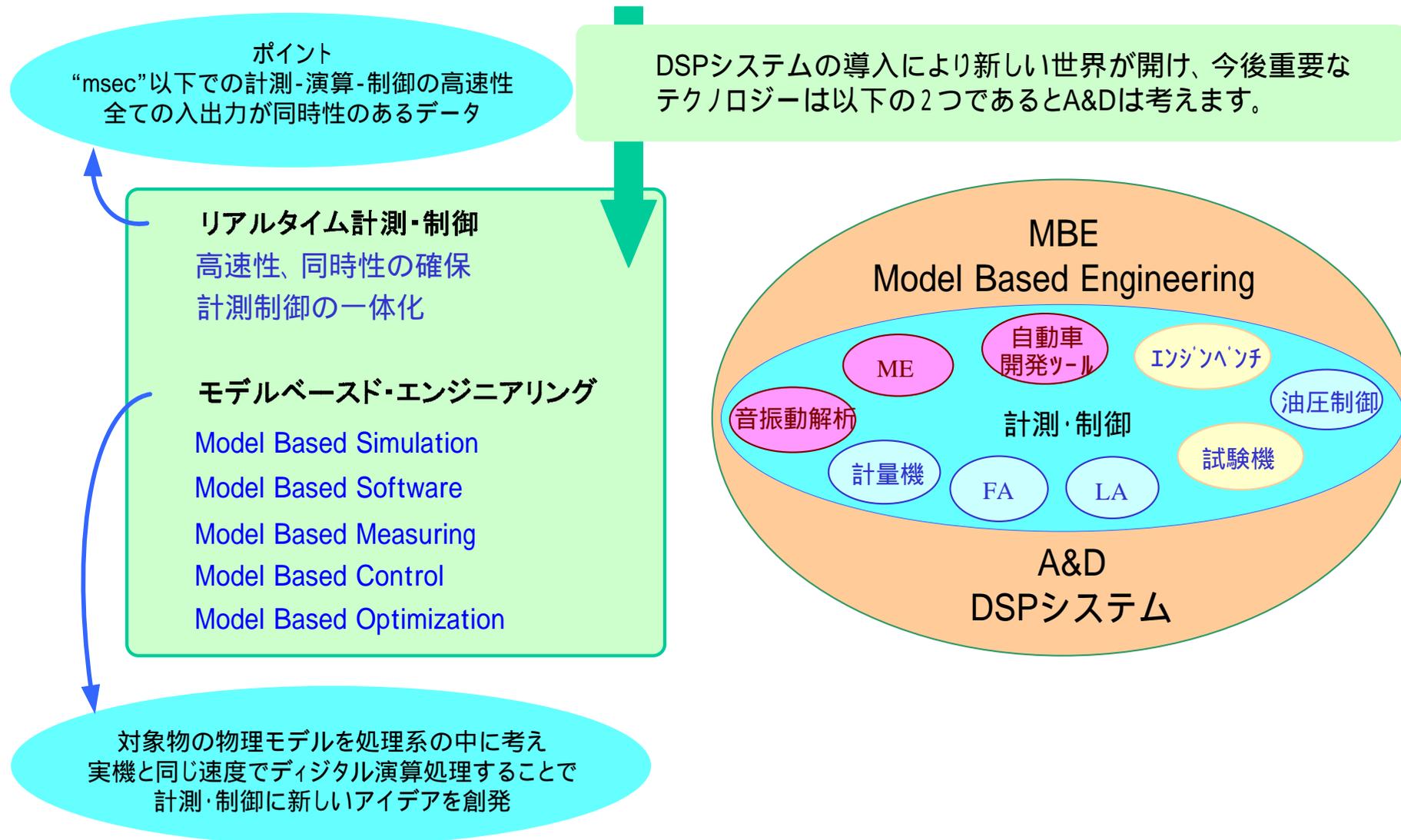
- Implementationブロックセット** AD5400シリーズのすべてのフロントエンドのIOを実装する機能部品群
- ASP ブロックセット** Advanced Signal Proseccing FFTブロック演算などで特殊な演算機能部品群
- GUI ブロックセット** 表示機能付きフロントエンド画面に配置されるコントロールと表示部品群
- FPGA ブロックセット** VHDLを出力する為の機能部品群
- Ladder ブロックセット** デバイス機能部品群
- VirtualConsole** DSP (MAT/Sim)のGUIとハブとなるフレームワーク
- M&C Ladder** PLCライクな開発環境を提供するフレームワーク
- GUI コントロール** VirtualConsoleに組み込まれるコントロールと表示部品群
- Virtual Analysis** 高度な解析機能群 Modeling Tool など

課題解決からのアプローチ



劇的変化の起きている計測制御の世界で、DSPの今後は

A&Dは、MBE (Model Based Engineering)をキーワードに計測・制御の分野で求められるDSPの開発をしています。



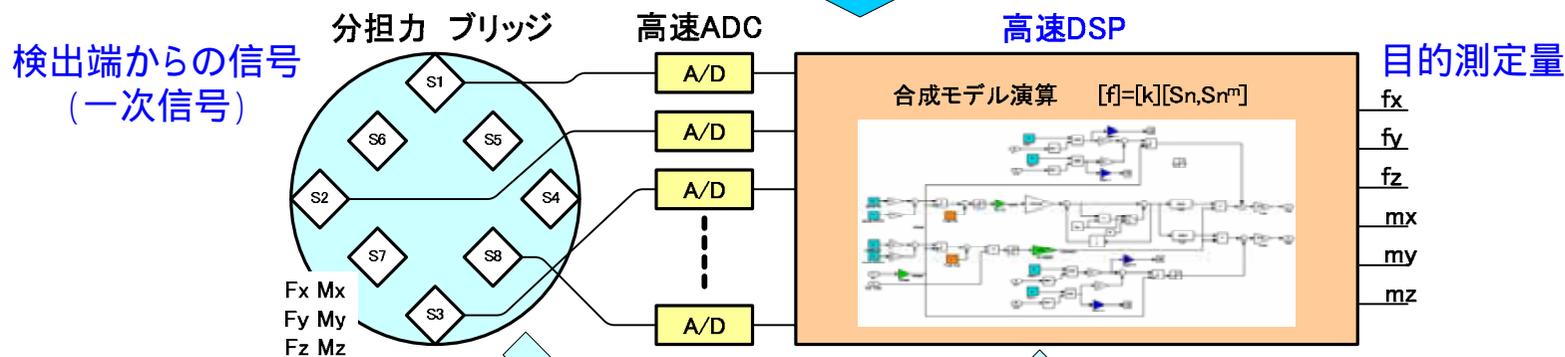
モデルベースの計測例 (RTS トルク計)

一次信号を高速DSPにより、合成モデル演算処理させることで高性能なRTSトルク計が実現します。

A&Dの計測技術は、電子天秤・血圧計にみられるように検出器のアナログ信号を直ちにデジタル信号に変換して、精度や機能UPはデジタル演算で行うという考えがベースにあります。

これに Model Based Engineering の考えを導入

再現性のある信号であれば、一次信号(原始的な信号)をデジタル化して取り込む
その信号が、どんな関係で目的量になるか論理的・実験的に追求し測定器モデルを導入する



この手法により、従来比10倍の高分解能・10倍速の高性能RTS軸トルク計の開発に成功



RTS軸トルクセンサーのベンチ実装例



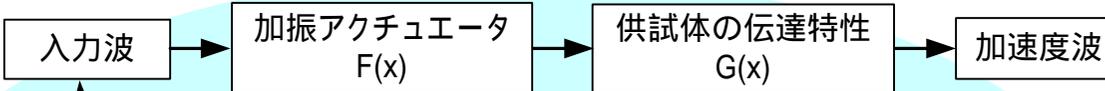
RTS軸トルクセンサーと高性能DSPによるRTS軸トルク計

モデルベースの制御例 (イタレーション技術)

制御モデルを高速DSPにて演算処理することで、忠実な再現が可能になります。



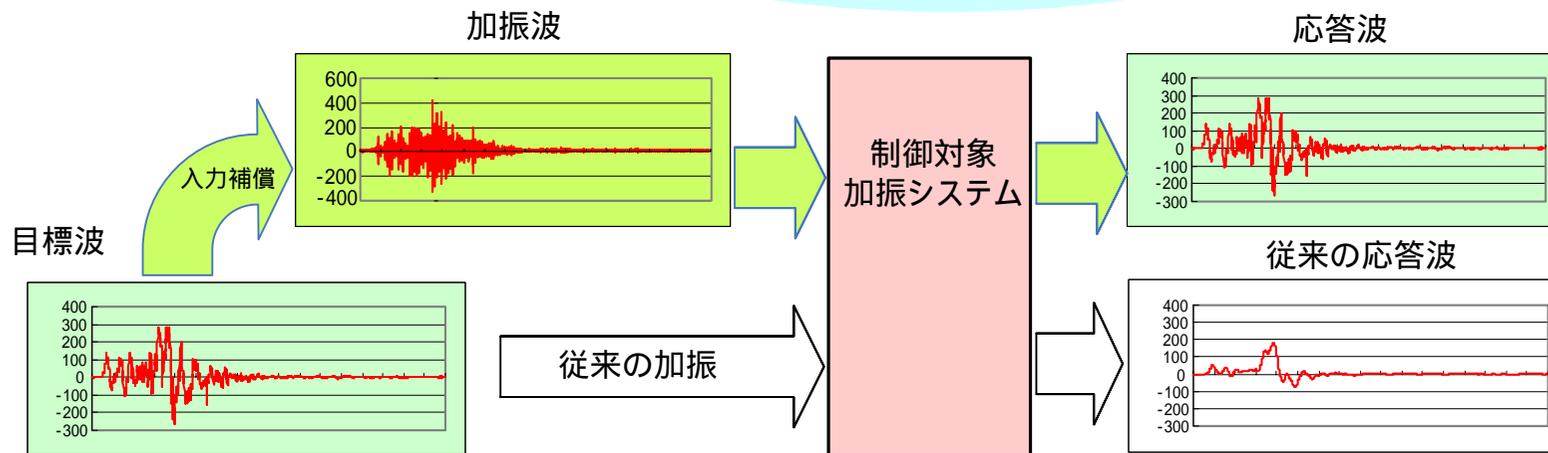
供試体に加える加速度を実路走行の加速度波形になるよう再現して、繰り返し加振したい



このような概念で入力波補償生成する手法をイタレーションと言い、一種のModel Based Controlです。



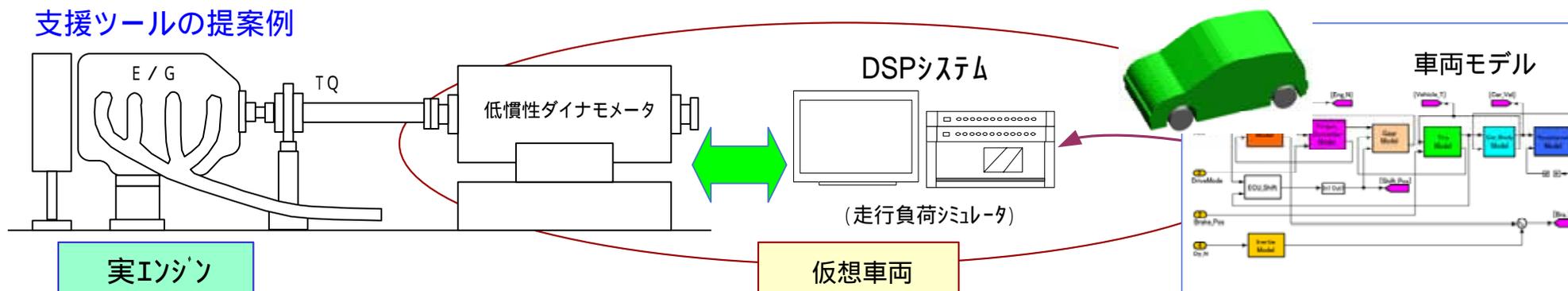
高速DSPで演算処理する



自動車産業の研究開発試験では評価関数が複雑増大化 評価ポイント検証に膨大な時間が必要

A&Dは、高性能DSPを中核としたMBE (Model Based Engineering) により、多様化するユーザー・ニーズに対応して行きます。

支援ツールの提案例



この技術に必要なテクノロジーは

実機と同じ速度で動くプラットフォーム = DSP

+

仮想モデルの作成 = MATLAB/Simulinkによるモデル

この概念は試験方法だけではなく、エンジニアリング全般に応用が効く考え方であると、A&Dは考えます

それは	試験方法では	Model Based Simulation
	ソフト開発手法では	Model Based Software
	計測方法では	Model Based Measuring
	制御方法では	Model Based Control
	適合・チューニングには	Model Based Optimization

デジタル信号処理・解析技術の変遷

