

システム・シミュレータ SM+がもつ シミュレーション機能のいろいろ

ここまで、システム・シミュレータ SM+のシミュレーション機能として、入出力パネル機能だけを使ってきましたが、SM+のもつシミュレーション機能はこれだけではありません。

ロジック・アナライザやウェーブ・フォーム・ジェネレータのような波形レベルで入出力を確認する機能、自在に入力信号波形を編集できる信号データエディタ機能、printf、scanf関数などを使用可能にする標準入出力機能など、多彩な機能をサポートしています。

ここでは、第9章で作成したBLINKサンプルをベースにして、SM+のいろいろなシミュレーション機能を試してみることにします。なお、シリアル・ポートの入出力のシミュレーション機能については、第12章で解説します。

10-1 端子の波形を観測できる「タイミングチャート」機能

「タイミングチャート」機能とは、ロジック・アナライザ/デジタル・オシロスコープのように、端子の波形を観測する機能です。プログラムの実行トレースだけではわかりにくい、実際の信号の状態を観測することで、プログラムが予定どおりの出力を行っているかや処理時間のかかりぐあいなどをチェックすることができます。

それでは、実際にタイミングチャート機能を使って、波形を観測してみることにしましょう。

10-1-1 Applilet で方形波の周期を変更

タイミングチャート機能の波形拡大/縮小倍率選択は実物のオシロスコープほど広くなく、BLINKのように500ms周期という遅い波形を見ると、スクロールしないと方形波になっているかどうかすらわかりません。これではいささか見えていて(画面キャプチャ上も)おもしろくないので、方形波の周期を2msに縮めて波形を見やすくしてみます。

第9章で作成したBLINKプロジェクトはたいせつに取っておいて、ディレクトリ(フォルダ)を丸ごとコピーして、BLINK_Tmchartというディレクトリ名に変更しました。

AppliletでBLINK_TmchartディレクトリのBLINKプロジェクトを開いて、10-1のように方形波の幅(周期)を2msに設定します。再びコード生成を行うと10-2のように、すでにint_user.cコードに手を入れているため、Appliletが生成するコードと置き換えるかどうか聞いてきます。ここでは既存のもの(BLINKで作成したもの)を利用するので、「ファイルをそのまま残す」を選びます。

ほかのファイルについても聞いてくるので、同様にしてください。面倒なときはこのダイアログの左下