

マイコン・ツールキット入門

2.6

レベルアップ! PSoC 入門

あるいは一種の動的再構成プロセッサ

■秋田純一 (金沢大学)

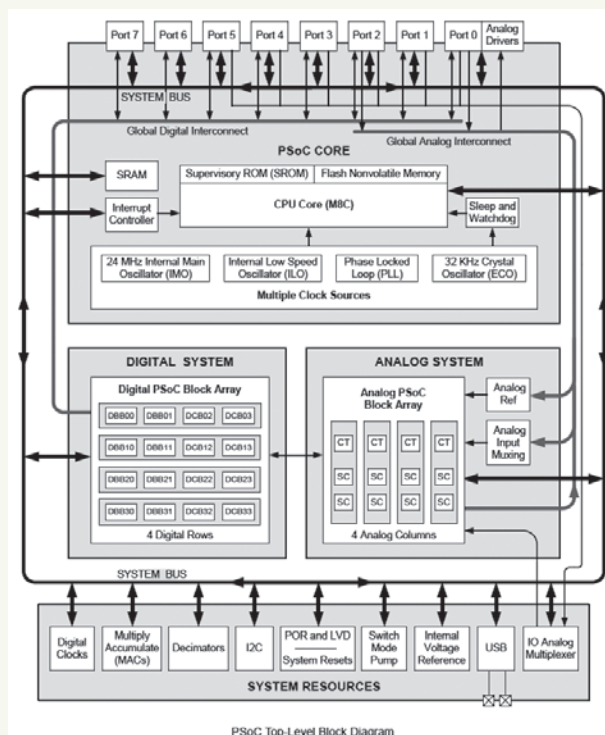
はじめに

計算機の性能向上とソフトウェア開発環境の充実によって、潤沢な計算機資源を使ってシステムを構築することができるようになってきて、システム構築にあたってCPU速度やメモリ容量などを心配する機会は減ってきています。携帯電話などのモバイル機器上でも同様の傾向は見られますが、まだやはりPCよりは計算機資源が制限される場合が多く、システム開発者は、その制限の中で、アルゴリズムの工夫などに力を注ぎます。フィジカルコンピューティングで使うマイコンでは、その制約がさらに厳しく、さらに内蔵されている通信ユニットやA/Dコンバータ、I/Oピンの配置なども考慮しなければなりません。これらの制限の中で最大限の性能を発揮できるためのアルゴリズムやハードウェア資源割り当ては、逆にマイコンを使ってシステムを構築する「醍醐味」と言うこともできるかもしれません。本稿では、既存のマイコンとは少し違った趣向の「マイコン」の醍醐味をご紹介します。

PSoC (Programmable System on Chip)

■ 概要

ここでご紹介するCypress社のPSoC (Programmable System on Chip) は、プログラムから外部とのやりとりを制御するという通常の処理手法とは異なり、外部とのやりとりを行う周辺回路ができるだけ主体的に処理を行う、という思想から生まれたものです。PSoC そのものの詳細については、詳しく

図-1 PSoCの内部構成⁴⁾

分かりやすく実践的な解説書籍¹⁾に譲りますが、内部は図-1のように8ビットマイコン(M8C)と機能可変な周辺回路からなり、マイコン部の性能は最大6MIPSと、お世辞にも高いとは言えません。これはマイコン部はあくまでも周辺回路の機能設定と最低限の演算処理のためのもの、という設計思想といえます。しかし周辺回路は、FPGAほどの柔軟性はないものの、タイマやPWM、各種シリアル通信、A/D変換器などの機能やそれらの接続をレジスタから設定することができるため、同じPSoCを使ってもタイマやPWMの数などを、ハードウェア資源の

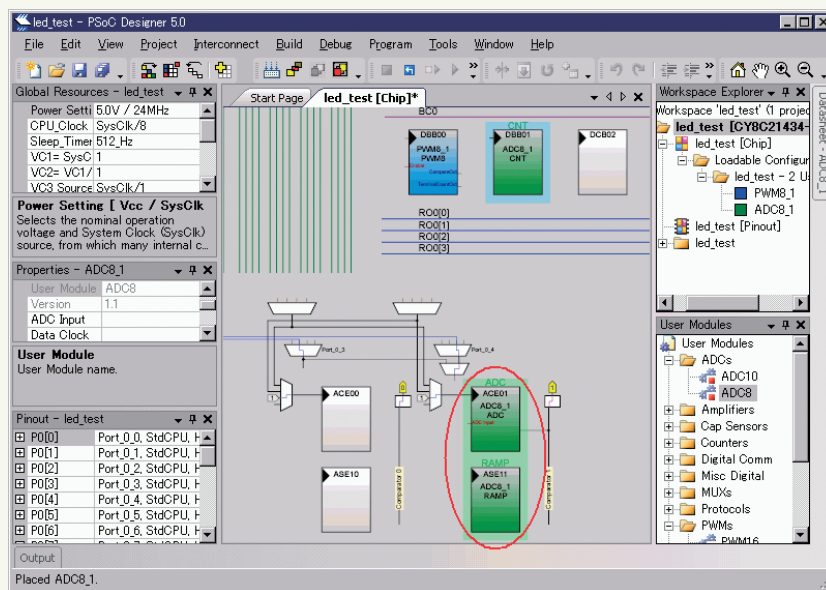


図-2 ハードウェア資源の周辺回路への割り当て

範囲内で自由に割り当てることができるだけでなく、プログラムによって動的に機能を変えることができるため、従来のマイコンではできないような芸当もできます。

PC側の開発環境であるPSoC Designerは、大きく分けて(1) System Level Designと(2) Architecture Level Designの2種類の設計方法が選べます。(1)は機能ブロックをブロック線図でつないで処理の流れをつくる方法で、それに合わせたハードウェア資源の割り当てなどが内部で自動的に行われます。(2)はC言語でプログラムを書く方法で、皆さんはこちらのほうがなじみが深いかもしれません。ただしPSoCではタイマ等の周辺回路を自由に使えますので、ハードウェア資源をどのような周辺回路に割り当てるかを選ぶ段階が必要になりますが、図-2のようにGUI操作でハードウェア資源の割り当てを行うことができます。

PSoCの入門キットは多数ありますが、その中でもCypress社直販や秋月電子他から入手できるPSoC First Touchは、PCにUSB接続で書き込みや動作ができる入門キットで、PC側の開発ソフトウェアもCypress社のWebページ²⁾から無償で配布されています。チュートリアル³⁾などを参考に、ぜひいろいろ遊んで慣れてみてください。

■ PSoCの内部ハードウェア資源

PSoCを「使いこなす」ためには、PSoCの内部ハードウェア資源の構成をよく理解する必要があります。詳細は文献4)に譲りますが、主なポイントをまとめておきます。

- デジタル機能ブロックの入出力は、行配線 (Row Digital Interconnect; RDI) とグローバル配線 (Global Digital Interconnect; GDI) の2つの配線を通して、I/Oピンに接続されます。また行配線などを使ってデジタルブロックどうしをつなぐことでもできます。
- RDIとGDI、およびGDIとI/Oピンは互いに自由に接続できるわけではなく、接続できる相手が決まっています。
- デジタル機能ブロックには、SPIやUARTなどの通信機能を設定できるDCBブロックと、できないDBBブロックの2種類が同数あります。
- アナログ機能ブロックの入力は、2つのマルチプレクサを通して指定されたI/Oピン(主にPort0)へ接続されます。また出力は、指定されたI/Oピンへ出力できます。
- アナログ機能ブロックには、オペアンプを含むアンプを構成できるCT (Continuous Time) ブロックと、スイッチト・キャパシタからなるアンプを

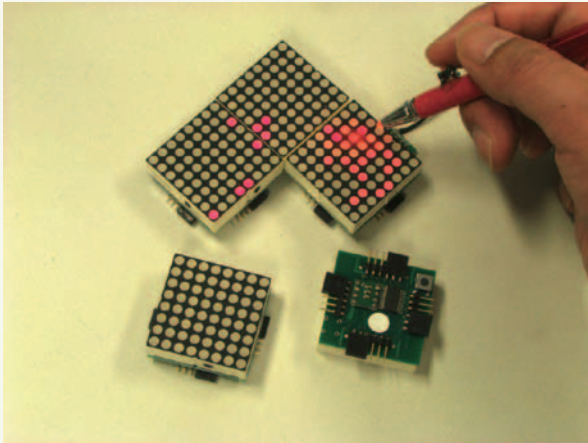


図-3 LED Tile⁵⁾

構成できない SC (Switched Capacitor) ブロックがあります。

PSoC を動的に再構成する

PSoC の周辺回路の構成・ハードウェア資源の割り当ては、ソフトウェア的にはマイコンから見える設定レジスタの値⁴⁾によって決められます。つまりソフトウェア的に、周辺回路の機能・構成を動的に再構成することが可能であるため、マイコンのコア部分の機能は固定ですが、一種の動的再構成プロセッサとして使うことができます。各機能を設定するレジスタの詳細は文献 4) をご参照ください。

■ I/O ピンの機能割り当て

PSoC の I/O ピンの機能は、(1) 1本ごとのソフトウェアから制御可能なデジタル入出力と (2) 内部回路の出力または内部回路への入力、から選ぶことができます。これらは、各 I/O ピンごとに、以下のレジスタによって設定することができます。

- ピンの駆動モード：出力（トータムポールかオープンドレイン、など）か入力か……PRTxDM2, PRTxDM1, PRTxDM0
 - ピンの機能：ソフトウェアから読み書きするか、内部回路に接続するか……PRTxGS
 - ピンの値：各ピンの "1" か "0" の値……PRTxDR
- なお各 I/O ピンの電気的特性も、通常の 1/0 出力のほかにプルアップ、オープンドレインなどが

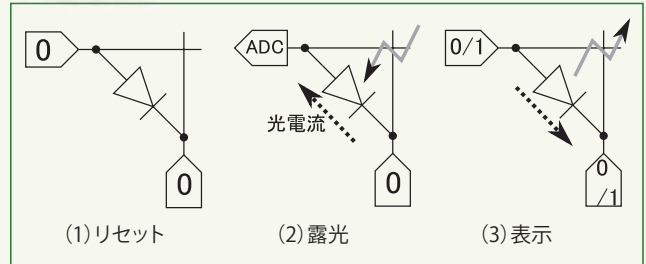


図-4 LED Tile のセンシング・表示の動作モード

ピンごとに設定できます。たとえば PRT0DM2/1/0 の 2 ビット目をそれぞれ "0"/"0"/"1" とすると、ポート 0 の 2 番ピン (P0²⁾) がトータムポールの出力ピン ("H" = 電源電圧, "L" = 0V) となり, "0"/"1"/"0" とすると、デジタル入力ピンとなります。これと合わせて PRT0GS の 2 ビット目を "0" とする (PRT0GS&=~0x04) と、P0²⁾ は PRT0DR のビット 2 を通して読み書きすることができ、PRT0GS の 2 ビット目を "1" とすると P0²⁾ は GDI を通して内部回路に接続され、このピンの駆動モードによって内部回路(たとえば PWM)から出力、または内部回路(たとえばシリアル受信)への入力とできます。

たとえば図-3 の LED Tile⁵⁾ では、LED の光センサとしての入力動作のための入力ピン設定と LED 点灯のための出力ピン設定を図-4 のように切り替えています。

■ デジタル部の構成

デジタル機能ブロックの機能や外部との接続は、各機能ブロック、行配線 (RDI)、グローバル配線 (GDI) に対して、以下のレジスタによって設定することができます。

- 機能設定：PWM, タイマ, シリアル通信 (UART・SPI) を設定する……DxBxxFN
- 入出力設定 (1)：各ブロックの入出力をどの RDI に接続するか……DxBxxOU, DxBxxIN
- 入出力設定 (2)：RDI をどの GDI に接続するか……RDixRI, RDixROx
- 出力設定 (3)：出力側の RDI から GDI へそのまま接続するか他のバスとの論理演算を行って接続するか……RDixLTx

なお GDI から各 I/O ピンへの接続は、「I/O ピンの機能割り当て」で述べた PRTxGS で設定されます。

たとえば一般的なマイコンシステムでは、非同期シリアル通信 (UART) などのシリアル通信をよく使いますが、PSoC ではシリアル通信機能を設定できる DCB ブロックの数が足りなくなることがよくあります。そこでこれらの設定を駆使すると、普段は受信機として設定しておき、必要なときだけ送信機に切り替えてデータを送信する、という使い方⁵⁾もすることができます。なお I/O ピンには割り込み設定をすることもできるため、受信端子にデータが到着したことを I/O ピン割り込みで検出し、割り込み処理ルーチンで即座に受信機を構成して受信するという芸当も、前述のように PSoC のマイコン部の演算速度がそれほど高くないため切替に 5 μ s 程度かかるものの、データ速度によっては十分可能です。

■ アナログ部の構成

PSoC には大きく分けてオペアンプとスイッチト・キャパシタのアナログ機能ブロックがあり、これらを組み合わせて可変倍率アンプや A/D コンバータなどを構成することができます。これらの構成は PSoC Designer から行うことができますが、これらの入力端子は、アナログ・マルチプレクサによって切り替えることができ、これは AMUX_IN レジスタによって動的に設定することができます。つまり複数入力切り替え A/D コンバータなどを構成することができます。たとえば LED Tile⁵⁾ では、「I/O ピンの機能割り当て」で述べた I/O ピンの切り替え機能と合わせて、2 個の A/D 変換器の入力を各 4 回切り替えての 8 個の LED の受光検出と表示を動的に行っています。

また CT ブロックをコンパレータ (電圧比較器) として使うこともでき、そのコンパレータの値はコンパレータバス (CMP) を通してデジタルブロックへ接続したりレジスタを通して読み出したりできます。

一般にアナログ回路では消費電力と精度がトレードオフの関係にあります。PSoC のアナログ機能

ブロックは消費電力を 3 段階に設定でき、用途に応じて (あるいは場面に応じて動的に) 切り替えることができます。もちろんアナログ機能ブロックへの電力供給をまとめて OFF とすることも (必要であれば動的に) できます。

またアナログ回路で重要となる安定な基準電圧源も、BGR (Band Gap Reference) を内蔵しており、電源電圧に依存しない安定な基準電圧を用いることができます。A/D コンバータなどの電圧範囲を決める基準電圧は、この BGR や電源電圧、外部から与える電圧から選択することができ、用途に合わせた範囲の電圧を扱うことができます。

おわりに

本稿では、通常のマイコンとは異なった設計思想から、少し変わった構成を持つ PSoC について紹介しました。ぜひ動的再構成の魅力とその可能性を楽しんでください。

また PSoC シリーズの後継として、PSoC3/5 シリーズも発表されています。これは、CPU コアがそれぞれ 8051 と ARM Cortex-M3 に強化されていて、また周辺回路部も、デジタル部に自由に論理回路を組める CPLD を搭載し、またアナログ部にも DAC や単体オペアンプが追加されています。もちろん、それらの構成はレジスタから設定できますので、動的再構成も可能です。特にデジタル部のハードウェア自由度が高くなった分、使いこなすのも大変そうですが、興味のある方は、ぜひチャレンジしてみてください。

参考文献

- 1) 桑野雅彦：はじめての PSoC マイコン，CQ 出版，東京 (2010)。
- 2) Cypress Inc. : <http://www.cypress.com/>
- 3) マイコンブ : <http://combu.merl.jp/>
- 4) PSoC Technical Reference Manual, Cypress.
- 5) 秋田純一：パターン入力と連結拡張が可能なインタラクティブマトリクス LED ユニット，情報処理学会論文誌，Vol.52, No.2, pp.733-736 (Feb. 2010).

(2011 年 4 月 6 日受付)

■ 秋田純一 (正会員) akita@is.t.kanazawa-u.ac.jp

1998 年東京大学大学院工学系研究科電子情報工学専攻修了 (博士 (工学))。金沢大学助手，公立はこだて未来大学講師，金沢大学講師を経て，2007 年より金沢大学理工学域電子情報学類准教授。専門は集積回路とその応用システム。