

第1章

プロセッサの概念からARMアーキテクチャ/ファミリを解説

ARMアーキテクチャ詳解

五月女 哲夫 / 小林 達也

ARM プロセッサはRISCアーキテクチャをベースに設計されている。はじめに、RISCプロセッサの概念を解説し、次にARMプロセッサの特徴とアーキテクチャを解説する。さらに、性能や用途に応じるためのARMファミリの特徴(インプリメンテーション)を説明する。

1 RISC プロセッサとは

1.1 プロセッサの概念

本書はARMプロセッサの解説が目的ですが、プロセッサのメカニズムを知っていないと以後の説明がわかりにくいものになってしまいます。ここではARMプロセッサの解説の前に、そのプロセッサのメカニズムの概念がわかるように、工場を例にとって動作を説明します。

● 工場の作業

この工場は、指示に従って何らかの部品を生産する

工場だとします。図1を見ながら、この工場の生産の流れを見てみましょう。

- (1) 敷地外の指示書倉庫(プログラム・メモリ)から指示書(命令)を持ってきて(フェッチ・ステージ)、それを生産管理棟(デコーダ)に渡す
- (2) 生産管理棟(デコーダ)は命令を解釈して、作業指示を作成し、作業指示を作業棟(エグゼキュート・ステージ)に送る。同時に作業棚(レジスタ)から作業に必要な製品(データ)を出庫して作業棟(エグゼキュート・ステージ)に送る
- (3) 作業棟(エグゼキュート・ステージ)では、作業指示に従って製品(データ)を加工し、加工した製品(データ)と作業指示を入出荷棟(メモリ・ステージ)

Column 1 ARMの歴史

ARMは、1990年11月、米国Apple Computer、英国Acorn Computer Group、米国VLSI Technologyの3社による英国拠点の合弁事業「Advanced RISC Machines Ltd.」として設立されました。AppleとVLSIの両社は資金を提供し、Acornは技術とARMの設立に貢献した12人のエンジニアを提供しました。世界初の商業用シングルチップRISCプロセッサを開発したAcornと、自社システムにRISC技術の導入を進めようとしていたAppleは、ARMによって新しいマイクロプロセッサ標準を作り上げようとしていました。ARMは、デスクトップ・コンピューティングに理想的な初の低価格RISCアーキテクチャを作ることによって、市場での差別化に成功しました。逆に、性能の向上に焦点を当てることの多かった競合アーキテクチャは、まずハイエンド・ワークステーションに採用されました。

1991年、初の組み込み型RISCコアであるARM6を発表すると、ARMは最初のライセンスとしてVLSIと契約しました。1年後、シャープと米国GEC Plesseyが

ライセンスに加わり、1993年には米国Texas Instrumentsと米国Cirrus Logicが続きました。1995年にはARMの特徴である、16ビットのシステム・コストで32ビットRISCのパフォーマンスの提供を可能にするThumbアーキテクチャ拡張を発表しました。1998年にはARM7TDMIのシンセサイザブル・バージョンを、1999年には信号処理の拡張されたシンセサイザブルなARM9Eプロセッサを、2000年にはJavaアプリケーション向けに拡張されたアーキテクチャJazelleテクノロジーを発表しました。

このようにARMは、数年の間に、アーキテクチャや製品ファミリを拡充し、ライセンスを増やすだけでなく、Bluetoothなどの先進技術へもいち早く対応してきました。さらにARMは、サード・パーティとも密接に連携して開発環境を整え、パートナーシップを拡大し、より多くのエンジニアに、ARM技術へアクセスするための選択肢を多数提供しています。

ARM命令セットの詳細

小林 達也

ARM7 マイコンにはARMアーキテクチャ標準のARM命令のほかに、コード・サイズを抑えられるThumb命令が使えるものが多い。本章ではARMプロセッサの命令セットについて解説する。(編集部)

ARMプロセッサはシンプル、かつ省電力であることに重点をおいて設計されています。第1章で述べたように、ARMプロセッサはソフトウェアとハードウェアのバランスを最適化するために大胆な取捨選択をしました。その結果、ソフトウェアの視点から見たときに、ほかのプロセッサにはない独特の特徴があります。これらの特徴とともに、ARMプロセッサの命令セットについて解説します。

1 ARM命令セットの特徴

すでに第1章で説明したように、ARMプロセッサはRISCアーキテクチャをベースに設計されたものです。このため、ARMプロセッサはほかのRISCプロセッサと同じように、いくつかのRISCとしての特徴をもっています。その一方で、その命令セットの構造にはほかのRISCプロセッサには見られないユニークな特徴があります。

● 演算と算術/論理シフトの1命令同時実行

ほとんどの演算命令で基本演算と同時に一方のオペランドに対する算術、または論理シフト演算を実行できます。わかりやすくいえば、算術/論理シフト命令と演算命令を同時に実行できるということです。たとえば、ARMプロセッサではレジスタR0にレジスタR1を8倍(3ビット左シフト)して加算し、結果をR0へ代入する動作を、

```
ADD R0, R0, R1, LSL #3
```

と表記して1命令で行うことができます。これを応用すると、レジスタの3倍算や5倍算といった演算も1命令で実行できます。

● 多彩なアドレッシング・モード

ARMプロセッサは命令セットに応じて複数のアド

レッシング・モードを持ちます。特にメモリ転送関係の命令では、前述のシフト命令と演算命令の同時実行を活用した非常に多彩なアドレス指定が可能です。これらのアドレッシング・モードは一見複雑ですが、ARMプロセッサ独特のハードウェア構成を無駄なく活用して効率の良い実行を可能にしています。

● 演算命令のフラグ・セット選択

多くの演算命令において、演算結果をフラグに反映する/しないを選択できます。

たとえば減算命令であるSUB命令で、R0とR1の各レジスタに同じ値が入っており、

```
SUB R0, R1, R0
```

としたとき、演算結果が0であることを示すZフラグは変化しませんが、フラグ・セットを意味する記述子‘S’をSUB命令に付けて、

```
SUBS R0, R1, R0
```

とした場合はフラグ・レジスタが更新され、Zフラグがセットされます。これを、次に説明する条件実行と組み合わせれば有効に活用できます。

● 命令の条件実行

ほとんどの命令で、フラグに応じて命令の実行と非実行(NOP)を決定できます。

たとえばレジスタR0が0以上であれば‘1’、-1以下のときは‘0’をレジスタR1に代入するプログラムを考えます。一般的なプロセッサでは比較命令と分岐命令を使用しなければならないため、リスト1(a)のようなプログラムになります。これに対してARMプロセッサでは、命令の条件実行の使用によりリスト1(b)のように3命令で、しかも分岐命令を使用せずに記述できます。このとき分岐によるパイプラインのフラッシュも発生しないため、全体的に見て高効率の命令実行が可能になります。

Appendix 1

A社製ARMマイコンとB社製ARMマイコンには互換性がない？！

ARMな世界のちょっと不思議な常識

熊谷 あき

●組み込みマイコンのいろいろ

世の中には、いろいろなベンダから組み込み向けCPUが出荷されています。またある一つの会社だけを見ても、8ビットの小さなマイコンから32ビットの高性能マイコンまで、いくつかのファミリーを持っているのが普通です。そして一般的に、これらの間には互換性はありません(図A)。

●ARMアーキテクチャとは

しかし、ARMはちょっと特殊です。

もともと、ARMアーキテクチャは英国Acorn社が自社のコンピュータ用に開発したCPUアーキテクチャです。同社からCPU設計部隊が独立して英国ARM社が設立されました。ARMアーキテクチャは、現在でもARM社で設計・開発されているCPUアーキテクチャです。

しかし、ARM社からは半導体デバイスとしてのARM CPUは市販されていません。ARM社はARMアーキテクチャの設計データを、半導体ベンダなどにライセンス供与しているのです。そしてARMアーキテクチャのライセンスを取得した半導体ベンダが、自社の型番を付けて実際のデバイスを販売しています。

たとえば、ARMコアを搭載したA社のAyというCPUシリーズと、B社のBxというCPUシリーズがあります。双方とも、ARMアーキテクチャ用に作られたソフトウェアが動作します(図B)。ARMコア搭

載CPUシリーズは、CPUベンダも型番も異なるCPUでありながら、互換性があるのです。

●周辺機能までは互換性がない

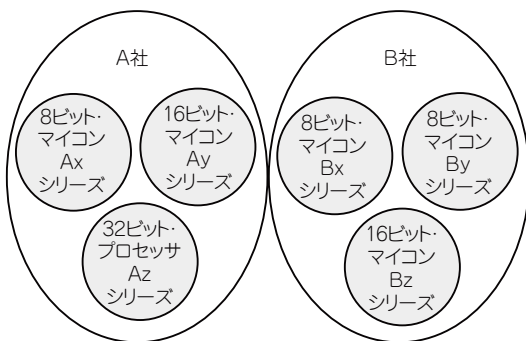
組み込み向けCPUは、CPUコア以外に、割り込みコントローラやタイマ・コントローラ、シリアル・コントローラなどのさまざまな周辺機能を内蔵しています。各社のARMコア搭載CPUシリーズは、CPUそのものには互換性がありますが、これらの周辺機能には互換性はありません(図C)。

また、組み込み向けマイコンの場合、周辺機能も非常に重要です。A社のARMマイコン用に作成したプログラムに、周辺機能まで含んだ互換性がなければ、他社のマイコンにそのまま持っていても動作しません。

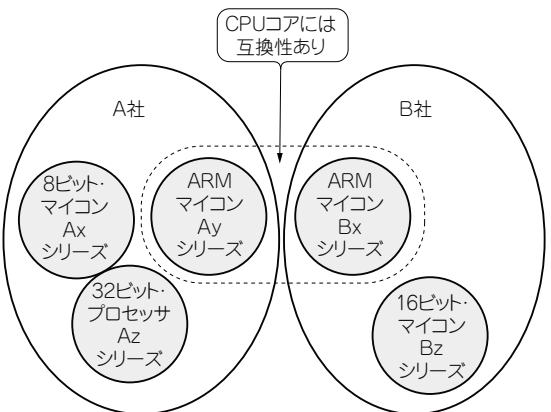
●A社ARMマイコン対応OSはB社ARMマイコンでは動かない

周辺機能にまで互換性がないときによく問題になる点として、OSの対応が挙げられます。

一般的に、あるOSが「○○CPU対応」といった場合、同じ○○シリーズのCPUであればすぐに動作させることができます。しかしARM対応OSがあり、それがA社のARMマイコンで動くからといって、B社のARMマイコンでも動くかということ、そのまま



図A 組み込みマイコンのいろいろ
同じ会社のマイコンでも、ファミリーが異なると一般的には互換性はない。



図B ARMコア搭載マイコンのいろいろ
会社が異なってもCPUコアには互換性がある。

ADuC7000 シリーズの概要

藤森 弘己 / 松添 信宏

本章では、米国 Analog Devices 社のマイクロコントローラ「MicroConverter」のうち、「ADuC7000 シリーズ」について解説する。ADuC7000 シリーズは、ARM7TDMI コアのほか、メモリ、A-D/D-A コンバータ、PWM 出力、プログラマブル・ロジック・アレイ (PLA) などを搭載している。それぞれの機能ブロックの特徴や使い方を説明する。
(編集部)

ARM コアは、システム LSI 用途において最も代表的な 32 ビット RISC プロセッサ・コアです。フラッシュ・メモリや RAM を集積した汎用の 1 チップ・マイコンのコアとして使われ始めました。

ここで紹介する ADuC7000 シリーズは、ARM コアのほか、クロック発振器やリセット回路などの CPU 周辺回路、およびアナログ I/O を集積しています。また、小実装面積・低価格が求められるアプリケーションに対応できるように、ローエンド・モデルも用意されています(写真1)。たとえば ADuC7022BCPZ32 の場合、外形寸法が 6mm × 6mm の 40 ピン・パッケージで、1,000 個購入時の単価は 4.03 ドル)。これにより 1 チップ・マイコンとしての ARM アーキテクチャがより身近になることを、筆者らは期待しています。

1 ADuC7000 シリーズとは

米国 Analog Devices 社の「MicroConverter」は、高精度の A-D コンバータ、D-A コンバータにマイクロコントローラを集積するというコンセプトで開発されました。ADuC7000 シリーズは、32 ビット RISC プロセッサ・コア「ARM7TDMI」を搭載した第3世代の製品です。そして、8 ビット・マイクロコントローラ 8052 コアをベースにした ADuC800 シリーズに追加された新ファミリです。ARM7TDMI コアの採用により、既存の 8 ビット・コアに加えて、32 ビット・コアの選択も可能となりました(図1)。ADuC7000 シリーズでは新たにクロック発振器やコンパレータ、プログラマブル・ロジック・アレイなどが追加され、周辺回路が強化されています。

表1に ADuC7000 シリーズの仕様を示します。

● ADuC7000 シリーズの特徴

ADuC7000 シリーズはアナログ・センサなどを使用するアプリケーションに向けた製品で、アナログ信号の監視やシステム制御に必要な周辺回路を集積しています。内蔵する主な機能ブロックは以下のとおりです。

- 12 ビット A-D コンバータ
(1M サンプル/s, 最大 16 チャンネル)
- 12 ビット D-A コンバータ (4 チャンネル)
- コンパレータ
- 2.5V バンド・ギャップ基準電圧源
- 32kHz 発振器 + PLL (phase-locked loop)
- プログラマブル・ロジック・アレイ (PLA)
- 3 相 PWM 発生器
- リセット回路
- 62K バイトのフラッシュ ROM
- 8K バイトの SRAM

なかでもアナログ周辺回路の性能に重点がおかれており、A-D コンバータは 12 ビットの絶対精度を -40℃ ~ +125℃ の温度範囲で得ることができます。また、基準電圧源を内蔵しているため、外付け部品を減らせ

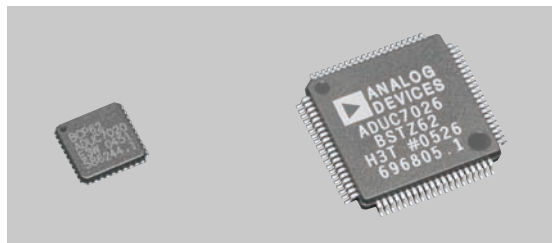


写真1 MicroConverter シリーズの外観
40 ピンの ADuC7020 (写真左) と 80 ピンの ADuC7026 (写真右)。

Appendix 2

CPLD/FPGA のように自由にロジック回路を組み込める

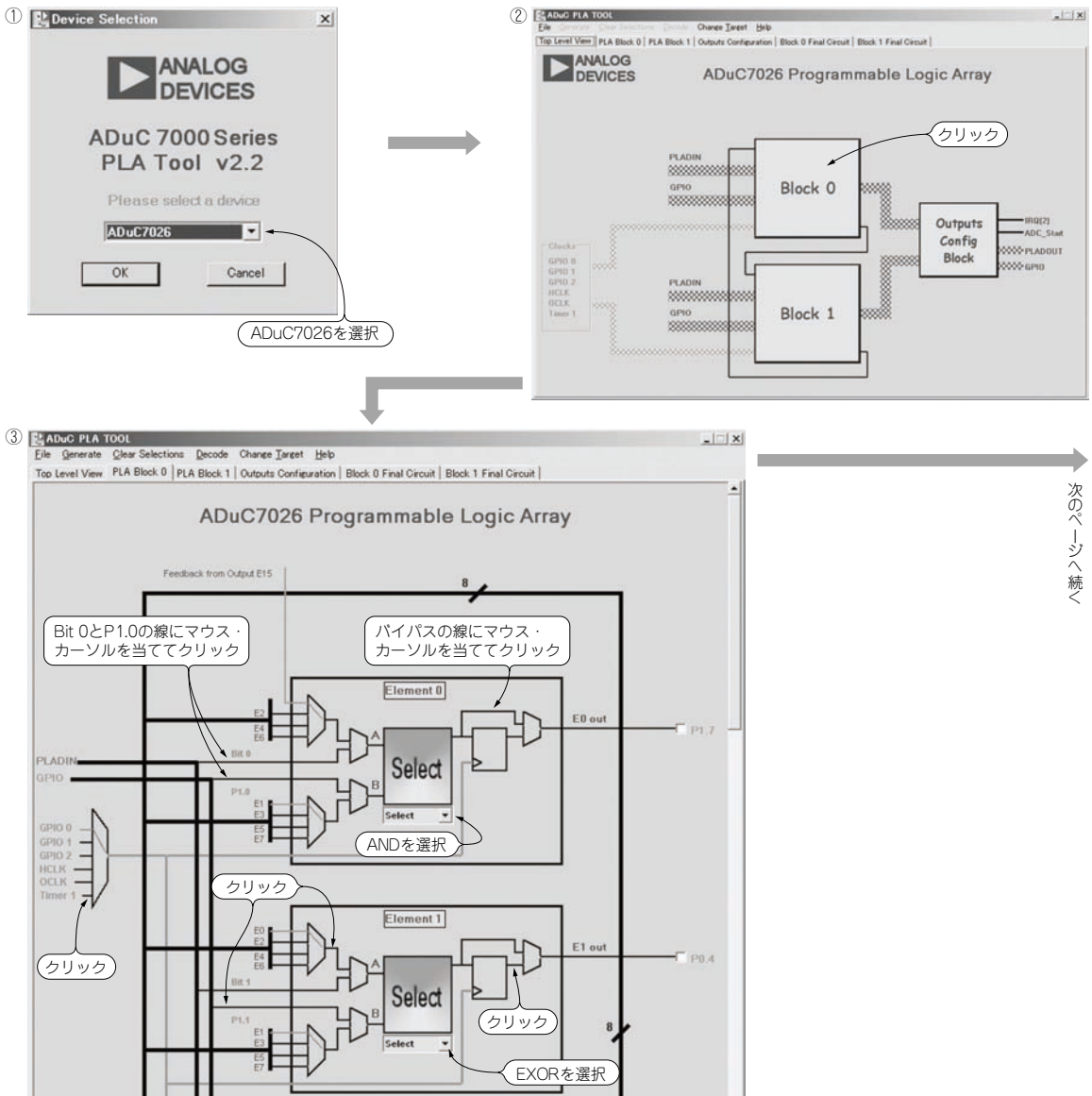
プログラマブル・ロジック・アレイ PLA の使い方

松添 信宏 / 佐々木 淑恵

1 PLA 設計ツールの使い方

ADuC7000 シリーズは 8 エLEMENT (Element), 2

ブロック (Block) 構成のプログラマブル・ロジック・アレイ (PLA) を内蔵しており、合計 16 エLEMENT を使えます。PLA の設計には専用の PLA Tool を使用します。ここでは、PLA Tool の使い方について説明



次のページへ続く

図 A PLA Tool の使い方

第4章

スタンダードな ARM7TDMI コアと周辺機能を内蔵したマイコン

**ADuC7026 搭載 CPU カード
& 拡張ベースボードの設計**

江崎 雅康 / 山武 一朗

ここで取り上げる ADuC7026 は 64 ピン QFP パッケージと小型ながら、各種シリアルやタイマなどの一般的なコントローラから、A-D/D-A コンバータやオンチップ基準電源/温度センサなどの各種アナログ周辺機能まで内蔵している。さらに 32.768kHz から CPU コア・クロックの約 40MHz を生成できるなど、小型コントローラに適したマイコンである。ここでは、組み込みベースボード CQBB に対応した小型 CPU カードの設計事例について解説する。

(編集部)

**1 小型 CPU カード
CQ-AD7026 の設計**● ADuC7026 を搭載した小型 CPU カード
CQ-AD7026

写真 1 に示す、ADuC7026 を搭載した小型 CPU カード CQ-AD7026 を設計しました。表 1 に基板の仕様を、図 1 に回路図を示します。

ADuC7026 は UART を 1 チャンネル内蔵しています。CPU 内蔵フラッシュ ROM は、このシリアル経由で書き換えられます。しかし、昨今のパソコンには COM ポートを備えたものが少なくなってきていま

す。そこで、この CPU カードでは、USB-シリアル変換 IC として CP2102 (米国 Silicon Laboratories 社) を搭載し、パソコンから USB 仮想 COM ポート経由で

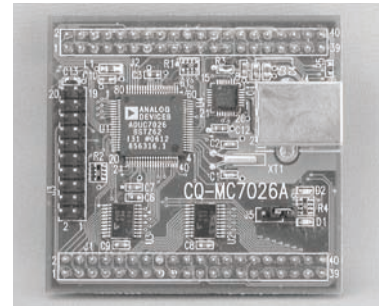


写真 1
ADuC7026 搭載
CPU カード CQ-
AD7026

表 1 ADuC7026 搭載 CPU カード CQ-AD7026 の仕様

CPU	32 ビット ATRM7TDMI プロセッサ ADuC7026 (最大 41.78MHz)
メモリ	内蔵フラッシュ ROM 62K バイト (1 万回書き換え可能, 20 年間)
	内蔵 SRAM 8K バイト
	外部メモリ・インターフェース 最大 512K バイト
フラッシュ ROM プログラミング	USB 仮想 COM ポート経由のイン・サーキット・ダウンロード JTAG デバッガによるイン・サーキット・プログラミング
JTAG インターフェース	JTAG インターフェース・コネクタ (20 ピン BOX) 装備
アナログ・ インターフェース機能	最大 16 チャンネル 12 ビット A-D コンバータ (完全差動モード, シングルエンド・モード) 入力電圧: 0 ~ 2.5V, サンプリグ・レート: 1M サンプル/秒
	最大 4 チャンネル 12 ビット D-A コンバータ, 電圧出力
	電圧コンパレータ
	オンチップ温度センサ (精度 ± 3 °C) オンチップ基準電圧 (2.5V ± 5mV, ± 40ppm/°C)
USB インターフェース	USB コネクタ 標準 B (USB-シリアル変換 IC CP2102)
外部拡張バス	組み込みベースボード CQBB 対応 (40 ピン × 2)
電源	電圧: 4.0 ~ 5.25V (UBS バス・パワーから供給も可能)
	電流: 11 mA (クロック周波数 5.22MHz 動作時), 45 mA (クロック周波数 41.78MHz 動作時)
プリント基板	4 層ガラス・エポキシ基板
外形寸法	53mm × 56mm

内蔵 A-D/D-A コンバータを活用して自動制御を実現する

ADuC7026 を使った PID 制御の実験

相田 泰志

ここでは、ADuC7026 基板を使って PID 制御の実験を行う。本章の前半で PID 制御について簡単に解説を行った後、実際に回路を動作させる。抵抗とコンデンサのみを追加する簡単な回路である。制御の状態は USB-シリアル経由で接続したパソコンを用いて観測するので、比較的簡単に実験を行える。(編集部)

米国 Analog Devices 社のマイクロコンバータ「ADuC702x」は、ARM マイコンを核とし、高精度 A-D コンバータと D-A コンバータを装備した、アナログ処理を得意とするマイクロプロセッサです。この ADuC7026 を使用して、簡単な制御の実験を行いました。

今回は、電源とわずかな部品を外付けするだけで制御の実験ができるコントローラを作成してみました。ADuC7026 は、電源リファレンスや A-D コンバータ、D-A コンバータが内蔵されているので、複雑な外付け回路を用意しなくても実験できます。

1 自動制御について

現在では、いろいろな場面で自動制御が適用されています。温度制御や速度制御のような「連続制御」と、工作機械などのようにあるパターンに従って動作を行う「シーケンス制御」と呼ばれるものがあります。

● 制御のための動作

温度制御などに使われる連続制御の基本は、

- ① センサなどにより物理量を測定し(たとえば温度)、
- ② その測定値に対して判断を行い(温度調節計)、
- ③ それに応じて物理量を変化させた出力を出す(ヒータ、ファンなど)。
- ④ その出力結果により物理量に変化し、また、センサで測定する。

といった一連の動作の連続的な繰り返しです。この一連の動作を制御ループと呼び、制御動作の基本になります。制御ループにより、いろいろな物理量の制御を行うわけです。

目標値(設定値)に対していかに物理量を制御できるかということが制御の目的であり、設定値と実際の値を一致させられることが、制御性がよいということになります。

● 目標値に一致させるのは難しい

目標値に実際の値を合わせるだけならば、一見簡単そうに思えます。しかし、実際のシステムで制御性の向上を実現することは簡単ではありません。

図1のように、ヒータにより部屋の温度を一定にするような制御システムを考えてみます。温度が低ければヒータを ON して部屋の温度を上げていきます。部屋が設定温度になったときにヒータを OFF すれば、一見よさそうに思えます。しかし、ヒータに余熱があると、それによって室温が上昇してしまいます。少し早めに OFF すればよさそうですが、早すぎると今度

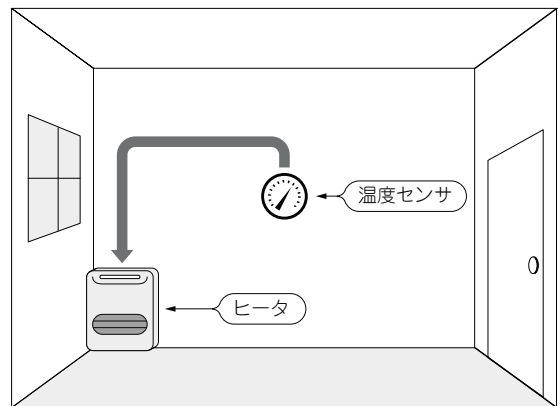


図1 部屋の温度を制御するシステム

温度センサにより部屋の温度を測定して、設定温度より低い場合にはヒータで部屋を暖める。人の出入りや換気のためにドアや窓が開けられると、部屋の温度は急激に変化する。より短い時間で部屋の温度を設定温度に調整し、その温度を保つためには、工夫が必要になる。

USB Mass Storage Class に対応した簡易計測器

ADuC7026 を使った アナログ・データ・キャプチャの製作

河野 崇

ここでは、USB を介してパソコンと接続できる簡易計測器の設計事例を紹介する。ADuC7026 の A-D コンバータからデータを取り込む。USB Mass Storage Class に対応しており、パソコン側のドライバ・ソフトウェアを開発しなくても使用できる。
(編集部)

最近の組み込み機器に要求される機能の一つとして、パソコンとの接続機能があります。

一昔前はシリアル (EIA-232) 接続が主流でした。この規格は、もともとモデムとパソコンを接続するために策定されたものであり、リモート・ホストとのキャラクタ・ベースのデータ通信に最適化されています。汎用性が高く直感的に理解しやすいため、簡単に制御ソフトウェアを書くことができます。多くの場合、組み込み機器をパソコンから制御するという用途ではこれで十分です。しかし、転送速度が低いため、高い応答性が必要だったり、大量のデータを高速にやりとりするという用途には向きません。

これに対して、最近のパソコンでよく使われている USB は、制御にも高速データ転送にも適した規格です。しかし、一部の規格化された機能以外については、Operating System (OS) に依存した専用のドライバ・ソフトウェアを用意する必要があります。開発に負担がかかるという問題があります。

本章では、USB 固有の情報を知らなくても、オープンな規格に準拠した方法で USB を活用する手法を提案します。USB Mass Storage Class へ対応することで、計測器やアクチュエータなどの制御機能とデータ転送機能を持つ機器を USB 経由で接続します。

Mass Storage Class は USB 経由でハード・ディスクやフロッピーディスクといった外部記憶装置を接続するための規格で、多くの OS で標準的にサポートされており、特別なドライバがなくても利用できます。OS は、コマンド/ステータス用およびデータ転送用のパーティションをもつ外部記憶装置としてデバイスを認識します。

制御用ソフトウェアは、POSIX 規格で定義されて

いる入出力システム・コール `open()`、`read()`、`write()`、`close()` を用います。デバイス・ファイルにアクセスすることにより、デバイスの制御やデータの入出力を行います。デバイスは仮想的に、あるいは物理的に制御用やデータ用の記憶領域 (パーティション) を用意し、それらに対してパソコンが USB 経由で送ってくる Mass Storage Class のコマンドを解釈・実行します。制御用パーティションに対する書き込みはコマンド発行、読み出しはステータス要求として解釈します。もちろん、データ用パーティションに対する読み書きはデータの読み書きと解釈します。

実装例として、ADuC7026 内蔵の A-D コンバータ (2チャンネル) からアナログ・データを取り込んでパソコンに転送する簡易計測器を作成しました。

OS は、ストレージ機器に対してキャッシュ機構を持っています。`open()` システム・コールでオープンする限り、オプション・フラグ (`O_DIRECT` と `O_SYNC`) を設定すればキャッシュを回避することができます。しかしここでは、アクセス方法の選択肢を広げ、あらゆる環境下で OS のキャッシュ機構を避けることを意図して、メディア・ロック機構のない USB フロッピーディスクをエミュレートしています。こうすることで、デレイは避けられないものの、ライト・キャッシュを自動的に回避できます。

1 簡易計測器のシステム構成

設計した簡易計測器のブロック・ダイアグラムを図 1 に示します。

USB インターフェースとデバイス固有の動作 (今回は A-D コンバータからのデータ取り込み) の独立性を

Appendix 3

汎用 I/O ポートと A-D コンバータを使って座標を判定

タッチ・パネル・システムの制御事例

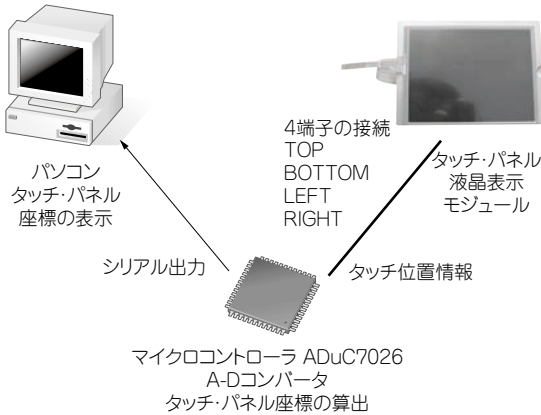
江崎 雅康 / 森川 聡久

1 タッチ・パネル付き液晶表示システムの構成

今回開発したシステムの概要を図Aと図Bに示します。タッチ・パネル液晶表示モジュールからは、TOP, BOTTOM, LEFT, RIGHTの4本の端子が出ています。このうちのTOPとLEFTの電圧を取得すると、タッチしている場所を知ることができます。電圧の値はADuC7026のA-Dコンバータの機能を利用して、デジタル・データに変換して取得します。

ADuC7026側では、

- (1) A-Dコンバータからデータの取得
- (2) 座標データに変換



図A タッチ・パネル付き液晶表示システムの構成図
タッチ・パネルの押されたXY座標をマイクロプロセッサADuC7026で読み取ってシリアル出力する。

(3) 座標データをシリアル出力を定期的に行います。

2 タッチ・パネルの原理とタッチ位置の取得方法

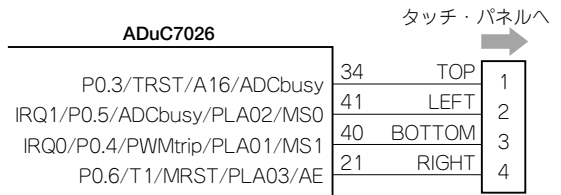
タッチ・パネルのタッチ位置を検出する原理には、

- 抵抗膜方式
- 静電容量方式
- 超音波表面弾性波方式
- 赤外線遮光方式
- 電磁誘導方式
- 画像認識方式

など多くの方式があります。

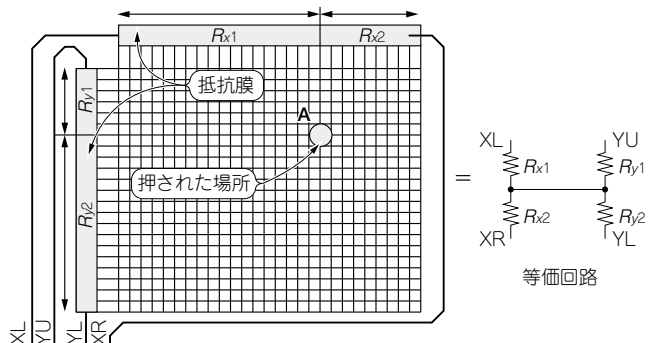
今回使用した液晶パネルTCG057VGLAD-G00(京セラ)のタッチ・パネルは抵抗膜方式です。図Cは抵抗膜方式タッチ・パネルの動作原理です。横方向に抵抗膜 R_x 、縦方向に R_y が配置されています。

抵抗膜 R_x および R_y から縦横に無数の導電スリット・パターンが配置された構造になっています。図の



図B ADuC7026とタッチ・パネルの接続

図C 抵抗膜方式タッチ・パネルの動作原理
押された場所のXY座標が抵抗膜の分圧値として検出できる。



シリーズ共通の周辺機能とその制御方法を解説

AT91SAM7 シリーズの概要とシステム・コントローラの使い方

大野 俊治

第3部では米国 Atmel 社製 ARM7 マイコンを取り上げる。同社は8ビット・マイコンの代表である AVR シリーズを出しており、上位マイコンとして ARM コアを搭載した AT91SAM シリーズがラインナップされている。ここでは、この中の ARM7TDMI コアを搭載した AT91SAM7 シリーズについて紹介する。(編集部)

1 AT91SAM7 シリーズの特徴

● AT91SAM7 シリーズの概要

米国 Atmel 社製 ARM7 マイコンには、型名の頭に“AT91SAM7”と付きます。同シリーズは、ARM7 TDMI コアを搭載したフラッシュ ROM 内蔵 ARM7 マイコンです。同シリーズは、豊富な周辺機能のサポートはもちろんのこと、その多くで PDC (Peripheral DMA Controller) と呼ばれる DMA 機能を使い、I/O 処理に伴う CPU 負荷を低減できます。

またほとんどの製品で USB デバイス・ポートをサポートしています。内蔵 ROM に格納された SAM-BA (SAM Boot Assist) と呼ばれるブート・プログラムを用いれば、USB 経由でのフラッシュ ROM の書き込みが可能です。これによりフィールドでのフラッシュ ROM の書き換えにも容易に対応できます。

CPU コアは 1.8V で動作します。しかし、チップ内に 1.8V のレギュレータを内蔵しているため、3.3V の単一電源で動作させられます。GPIO ポートは 5V トレラントになっているので、5V 系のデバイスをつなぐこともでき、とても使いやすいマイコンに仕上がっています。

AT91SAM7 のシリーズの一覧を表 1 にまとめました。それぞれのシリーズでフラッシュ ROM と RAM 容量の異なるモデルがいくつか存在し、製品群を構成しています。AT91SAM7A3 は例外ですが、ほかの製品では型番の最後の数字がフラッシュ ROM の容量を示しています。

このように SAM7 シリーズでは、外部メモリが必要な SAM7SE を、Ethernet が欲しければ SAM7X

を選択します。USB デバイス機能は AT91SAM7L を除く各シリーズのほとんどのデバイスでサポートされていますが、第9章で説明するように、シリーズによってサポートされているエンドポイント数やパケット・サイズが異なります。

Atmel 社では ARM マイコンとして、AT91SAM7 シリーズ以外にも ARM9 ベースの AT91SAM9 シリーズと Cortex-M3 ベースの ATSAM3 シリーズを製造/出荷しています。これらのシリーズでも、AT91SAM7 シリーズとほぼ同等の周辺機能が利用できるため、AT91SAM7 シリーズで開発したドライバやソフトウェアは、比較的簡単にほかのシリーズへ移植できます。

● AT91SAM7S シリーズ

AT91SAM7 シリーズのうち、最も基本的な構成を持つのが AT91SAM7S シリーズです。64ピンまたは 48ピンと最もピン数が少ないため、小型の USB デバイスを開発したり、8/16ビット・マイコンからの移

表1 AT91SAM7 シリーズの一覧

シリーズ名	周辺機能	デバイス名
AT91SAM7S	PIT, WDT, RTT, PIO, UDP, SSC, USART, SPI, TC, PWMC, TWL, AD	AT91SAM7S512/ 256/64/321/32/ 161/16
AT91SAM7X	AT91SAM7S の周辺機能に加えて Ethernet と CAN を追加	AT91SAM7X512/ 256/128
AT91SAM7SE	AT91SAM7S の周辺機能に加えて EBI を追加	AT91SAM7SE512/ 256/32
AT91SAM7A	AT91SAM7S の周辺機能に加えて MCI を追加	AT91SAM7A3
AT91SAM7L	AT91SAM7S の周辺機能から SSC と UDP を削除	AT91SAM7L128/64

AT91SAM7 シリーズ内蔵周辺機能の使い方

大野 俊治

第7章では SAM7 シリーズの概要と、ほぼすべての SAM7 シリーズで共通のシステム・コントローラに含まれる機能を中心に説明した。本章ではそれ以外の、汎用 I/O やシリアル、A-D 変換などの周辺機能について説明する。
(編集部)

1 周辺機能の選択

● PIO と周辺機能の選択

多くのマイコンがそうであるように、AT91SAM7 シリーズにおいても、複数の周辺機能を割り当てられているピンが数多くあります。そのため、使用する機能を選択するためには、ピンの機能選択設定が必要です。AT91SAM7 シリーズでは、選択が必要となるピンはすべて PIO ポートのピン上に割り当てられているので、機能選択は PIO に関連する設定の一環として行えるようになっていきます。

表1に AT91SAM7S における周辺機能割り当てのようすを示します。PIOA としてグルーピングされている PA0～PA31 の 32 本の PIO 端子に周辺機能が割

り当てられています。ほとんどの端子に Peripheral A と Peripheral B の二つの周辺機能が割り当てられているので、各端子は、

- (1) PIO (入出力ポート)
- (2) Peripheral A
- (3) Peripheral B

の三つの役割のいずれかを選択して使用できます。リセット直後の状態ではすべての端子は PIO として動作する状態になっています。

● 制御レジスタの構成

図1に示したように、PIO の機能制御に関連して数多くのレジスタが用意されており、とても複雑に見えますが、個々の機能は単純なので容易に理解して使いこなせるでしょう。基本的には次の手順でレジスタを設定して機能を選択します。

表1 AT91SAM7S における周辺機能割り当て

I/O	Peripheral A	Peripheral B		I/O	Peripheral A	Peripheral B	
PA0	PWM0	TIOA0		PA16	TK	TIOB1	
PA1	PWM1	TIOB0		PA17	TD	PCK1	AD0
PA2	PWM2	SCK0		PA18	RD	PCK2	AD1
PA3	TWD	NPCS3		PA19	RK	FIQ	AD2
PA4	TWCK	TCLK0		PA20	RF	IRQ0	AD3
PA5	RXD0	NPCS3		PA21	RXD1	PCK1	
PA6	TXD0	PCK0		PA22	TXD1	NPCS3	
PA7	RTS0	PWM3		PA23	SCK1	PWM0	
PA8	CTS0	ADTRG		PA24	RTS1	PWM1	
PA9	DRXD	NPCS1		PA25	CTS1	PWM2	
PA10	DTXD	NPCS2		PA26	DCD1	TIOA2	
PA11	NPCS0	PWM0		PA27	DTR1	TIOB2	
PA12	MISO	PWM1		PA28	DSR1	TCLK1	
PA13	MOSI	PWM2		PA29	RI1	TCLK2	
PA14	SPCK	PWM3		PA30	IRQ1	NPCS2	
PA15	TF	TIOA1		PA31	NPCS1	PCK2	

CPU内蔵USBターゲット・コントローラでシリアル変換器を実現

コミュニケーション・クラスを使った 仮想シリアル・コンバータの作成

大野 俊治

ここではUSBコントローラをCPUに内蔵したAtmel社製ARM7マイコンについて取り上げる。内蔵USBコントローラの制御方法について詳しく解説しているの、仮想シリアル・ポートに限らず、同社のマイコンを使ったUSB制御事例としても参考になるだろう。

(編集部)

1 USBマイコンの概要

米国Atmel社のUSBマイコンの特長について説明します。

● AT91SAM7シリーズ

AT91SAM7は、英国ARM社のARM7TDMIコアを搭載したフラッシュROM内蔵マイコンです。周辺機能としてDMA (Direct Memory Access) 機能^{注1}を持ち、I/O処理に伴うMCU (Micro Control Unit) の負荷を低減します。また、USB 2.0フル・スピード対応のUDP (USB Device Port) も持ち、内蔵ROMに格納されたブート・プログラム [SAM-BA (SAM-Boot Assist) と呼ばれる] を用いてUSB経由でフラッシュROMを書き換えられます^{注2}。

注1：Atmel社のマイコンでは、PDC (Peripheral DMA Controller) と呼ばれる。

注2：SAM-BAに関しては、本書第7章 (pp.107-124) を参照。

ARMコアの動作電圧は1.8Vですが、レギュレータ内蔵により3.3Vの単一電源で動作でき、GPIOポートは5Vトレラントのため、5V系デバイスも接続できます。

● 必要な機能に応じてマイコンを選ぶ

本シリーズの中で、UDPを持つマイコンを表1にまとめました。USB機能に加えて、外部メモリが必要であれば末尾がSEのAT91SAM7SEを、Ethernetが必要であれば末尾がXのAT91SAM7Xを選択します。また表2に示すように、シリーズによって、USBポートでサポートされているエンドポイント数や最大パケット・サイズが異なります。USB規格上でフル・スピードのアイソクロナス転送では1,023バイトのデータ転送をサポートしますが、AT91SAM7では最大512バイトしか転送できないことに注意してください。

● ARM9コアやCortex-M3コア搭載のシリーズ

表1にはありませんが、ARM9コア搭載のAT91SAM9やCortex-M3コアを搭載したAT91SAM3の

表1 USBコントローラ機能を持つAtmel社のARM7マイコン一覧

フラッシュROMとRAMの容量が異なるモデルを用意している。製品型番の数字はフラッシュROMの容量を示す (AT91SAM7A3は例外)。

シリーズ名	周辺機能	型番
AT91SAM7S	インターバル・タイマ、ウォッチドッグ・タイマ、リアルタイム・タイマ、PIO、USBデバイス・ポート、同期シリアル・コントローラ、USART、SPI、16ビット・タイマ、PWMコントローラ、I ² C、A-Dコンバータ	AT91SAM7S512/256/64/321/161
AT91SAM7X AT91SAM7XC	SAM7Sの機能に加えて、Ethernet、CAN	AT91SAM7X512/256/128 AT91SAM7XC512/256/128
AT91SAM7SE	SAM7Sの機能に加えて、外部バス・インターフェース	AT91SAM7SE512/256/32
AT91SAM7A	SAM7Sの機能に加えて、マルチメディア・カード・インターフェース	AT91SAM7A3

Appendix 4

PINGの要求&応答やテスト・パケットを送受信できる

AT91SAM7シリーズを使ったネットワーク・テスト・プログラムの作成

大野 俊治

1 AT91SAM7XシリーズのEthernet機能

● 論理層を内蔵

AT91SAM7Xでは、10/100Mbps対応のEthernet MAC(論理層)機能がサポートされており、EMACと呼ばれています。EMACは、次のような機能を持っています。

- 最大四つまでのMACアドレスを設定可能
- ポーズ・フレーム対応
- VLAN対応
- ジャンボ・フレーム対応(10,240バイトまで)

図Aに示すように、EMACはPDCではなく専用のDMA機能を有しています。DMA用の特別なメモリ空間は用意されておらず、通常のSRAM空間を使用するようになっています。EMACの内部には送受信それぞれ28バイトのFIFOが用意されていますが、このFIFOはメモリとEMACの転送速度の違いを吸収するために設けられているものであり、ドライバ・ソフトウェアが意識する必要はありません。

● 送受信バッファとバッファ・ディスクリプタ

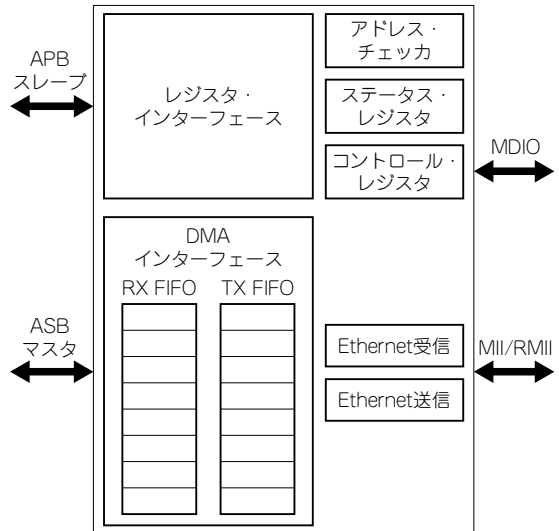
EMACでは、送受信のバッファ管理を行うためにバッファ・ディスクリプタという2ワード(8バイト)の構造体の配列を用います。バッファ・ディスクリプタの構造は送信と受信の場合で異なっています。まず

受信バッファ・ディスクリプタの概要を示します。

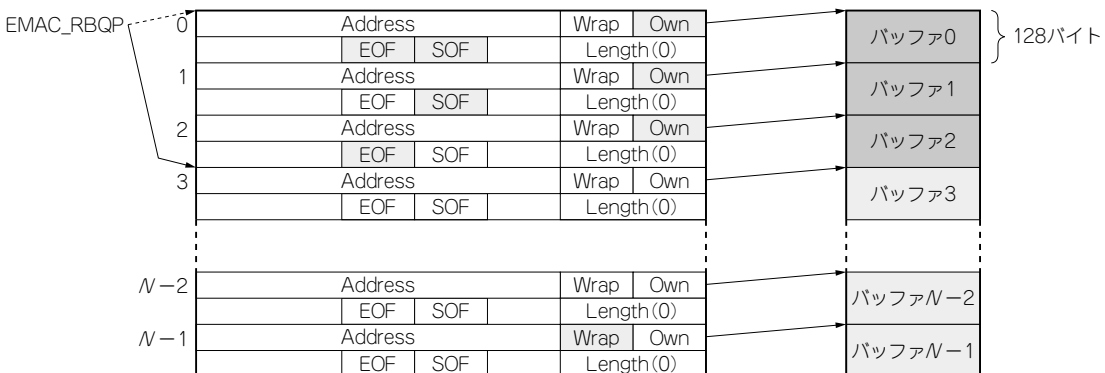
● 受信バッファ・ディスクリプタ

図Bに受信バッファ・ディスクリプタの構造を示します。

受信バッファ・ディスクリプタの各エントリは、対応する受信バッファのアドレスと、その受信ステータスを保持します。最初のワードには、受信バッファのアドレスを指定します。各受信バッファのサイズは128バイトで固定で、ワード境界に配置することに



図A EMACの構成



図B 受信バッファ・ディスクリプタ

第10章

NXP Semiconductors 社製 ARM マイコンのシリーズと特徴

LPC シリーズと LPC2388 の概要

田中 法和

ここでは NXP Semiconductors 社製 LPC2388 について解説する。まず同社の ARM マイコンについて全体を説明した後に、LPC2388 の特徴や各機能について解説する。最後に LPC2388 についてのさまざまなドキュメントやアプリケーション・ノートなどの情報の入手先についても紹介する。 (編集部)

1 ARM マイコン LPC シリーズの概要

● NXP Semiconductors 社のマイコン

オランダ NXP Semiconductors 社 (旧 Philips Electronics 社の半導体部門, 以下 NXP 社) は, かつては 8 ビットの 80C51 シリーズを中心にマイコンを展開してきました。現在は ARM コアを中心としたマイコン製品をシリーズ展開し, 6 コア, 80 種類以上の製品にさらに Cortex-M0 コアを追加, 今後は Cortex-M4 コアを加え 150 製品を超える予定です (図 1)。NXP 社の ARM マイコンの特徴の一つは, ピン互換性を保ったままの同一パッケージで, 内蔵メモリのサイズや周辺を組み合わせた異なる品種が豊富にそろっていることです。また, 用途に合わせてコアを選べます。代表的なコアを挙げると次のようになります。

- LPC1000 … Cortex-M3 (Rev.2), -M0 コア
- LPC2000 … ARM7TDMI-S コア
- LPC2900 … ARM968E-S コア

● LPC3000 … ARM926EJ-S コア

これ以外に, LCD コントローラを搭載した Blue Streak (ARM7, ARM9) や LPC2800 のシリーズもあります。今回紹介するマイコンは NXP 社の中でもより汎用性の高いマイコンとして製品化されたものです。

意外に知られていないことですが, NXP 社はかつての Philips Semiconductors の時代に, ARM 設立メンバーの一つである米国 VLSI Technology 社を継承しており, あらゆる部門において ARM コアのデバイスを従来から展開しています。NXP 社が汎用の ARM マイコン製品を製造すること自体は, かなり以前から行ってきたことで, 最近になってから始めた事業ではありません。ちなみに, 米国 ARM アーキテクチャの歴史についてもっと詳しく知りたい方は, ARM 社の Web サイトにある「ARM アーキテクチャ 20 周年記念」を見てください。

<http://www.jp.arm.com/aboutarm/>

[index.html](http://www.jp.arm.com/aboutarm/index.html)

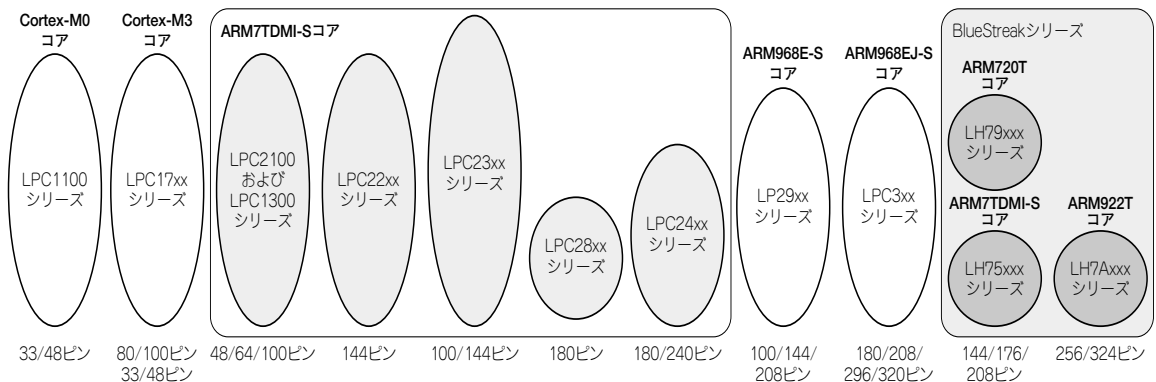


図1 NXP 社製 ARM マイコンのシリーズ一覧

Appendix 5

LPC シリーズ内蔵フラッシュ ROM を Windows 環境から書き換える

内蔵フラッシュ ROM 書き換えツール FlashMagic の使い方

田中 法和

● FlashMagic とは

LPC シリーズはフラッシュ ROM を内蔵しているため、ISP (In-System Program) 機能によりユーザ・プログラムを書き込むことができます。そのためのプログラミング・ツールが FlashMagic です。

FlashMagic は次の URL からダウンロードできます (図 A)。

<http://www.flashmagictool.com/>
 もともと LPC2000 シリーズにはオランダ NXP Semiconductors 社の Web サイトからダウンロードできる

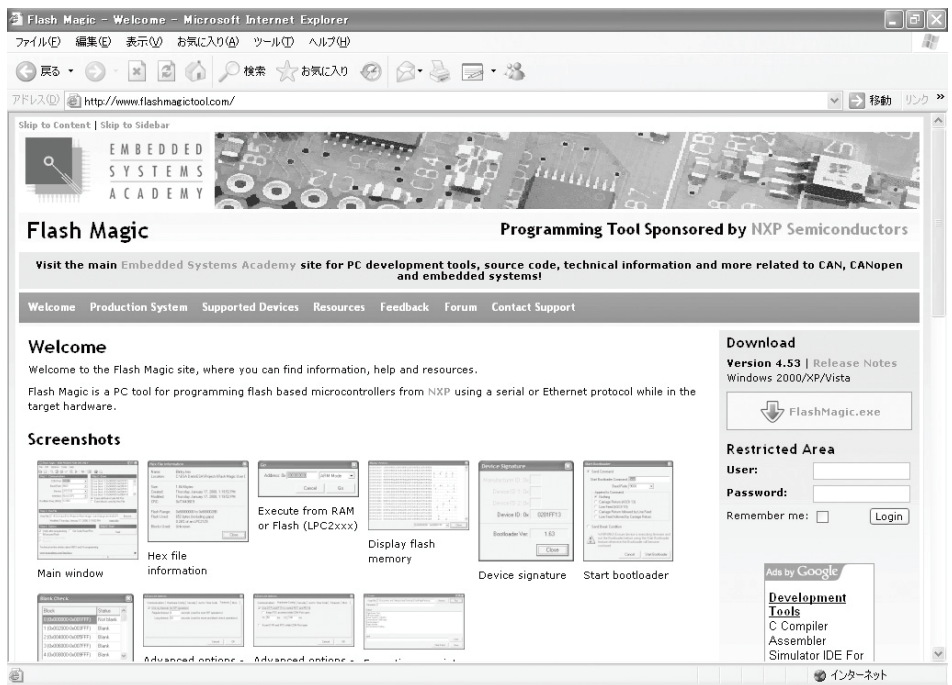


図 A
FlashMagic ダウンロード・ページ
(<http://www.flashmagictool.com/>)



図 B FlashMagic のインストール開始

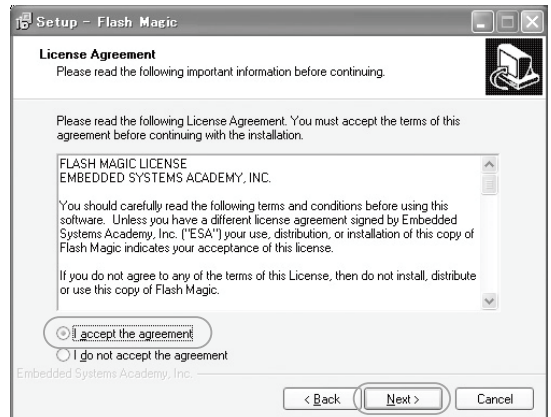


図 C ライセンス承諾

第 11 章

4ビット・ネイティブ転送対応の高速アクセス

LPC2388 内蔵 MMC カード・コントローラの使い方

赤松 武史

近年の組み込み機器ではメモリ・カードを用いてデータを外部とやりとりすることが多い。そのような需要から、LPC2388 には MMC/SD メモリ・カードを接続するためのインターフェースが内蔵されている。ここでは、LPC2388 のカード・コントローラである MCI モジュールについて解説する。 (編集部)

LPC2388 には内蔵 I/O モジュールの一つとして、マルチメディア・カード (以下 MMC) や SD メモリ・カード (以下 SD カード) を制御する MCI (Multimedia Card Interface) モジュールが搭載されています。本章では MCI の機能と具体的な制御方法の解説をし、第 12 章で実際にメモリ・カードを読み書きするドライバを作成します。

1 MCI モジュールの機能

MCI には次のような機能があります。

- MMC および SD カードのホスト・コントローラ機能
- SD カードの 4 ビット・バスをサポート
- DMA によるデータ転送機能

もともと MMC は 1 ビットのデータ・バス (Path : 伝送路) による転送を行う規格でしたが、これを拡張した SD カードでは 4 ビット・バスによる高速転送が可能になっています。LPC2388 はこの 4 ビット・バスをサポートしています。

図 1 に MCI の機能ブロック図を示します。コマン

ド・バスは、メモリ・カードへのコマンドの送信とカードからのレスポンスの受信を行うブロックです。またデータ・バスはメモリ・カードへのデータの送信とカードからのデータ・ブロックの受信を行うブロックです。これらのブロックは互いに独立し、同時並行の動作が可能です。

データ・バスと内部バスとの間にはバッファとして 64 バイト長の FIFO が設けられており、高速なデータ転送におけるデータの取りこぼしを防止しています。

2 MCI の周辺回路

図 2 に MCI とメモリ・カードの結線を示します。MMC は複数のカードをバス接続できます。SD カードは共通バス接続をサポートしていないので、MCI では同時に 1 枚しか接続できません。

● MCI の信号

● MCICLK

転送クロック出力です。メモリ・カードの CLK 端子に接続します。

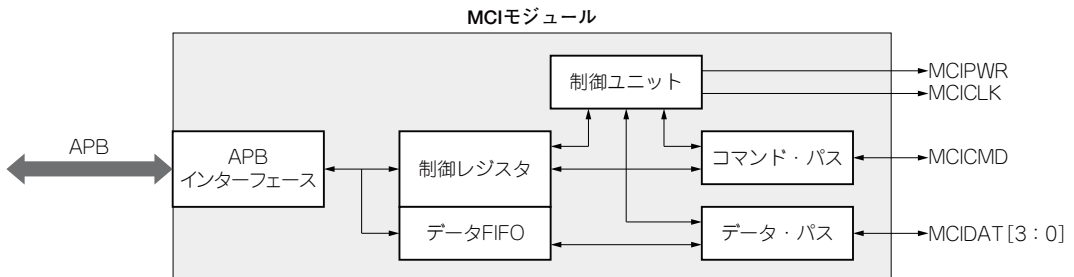


図 1 MCI モジュールの機能ブロック図

第12章

オリジナルなモバイル情報端末が実現できる！

FAT ファイル・システムの構築と有機ELディスプレイの接続

赤松 武史

第11章の解説で LPC2388 基板へ MMC/SD カードをハードウェア的に接続できるようになった。しかし MMC/SD カードを実用的に使うためには、物理的な MMC/SD カードを FAT ファイル・システムとして解釈できるようなソフトウェアが必要になる。ここでは、オープン・ソースのファイル・システムとして筆者が公開している FatFs を移植し、ファイルの読み書きを行うシステムを作成する。 (編集部)

メモリ・カードが使えるようになると、単独で大容量のデータを容易に扱え、またパソコンとの連携がしやすくなるなど、マイコン・システムの応用の幅が広がります (図1)。

第11章では LPC2388 内蔵の MCI (Multimedia Card Interface) の詳細について解説しました。ここでは LPC2388 マイコン基板 (以降、LPC 基板) へメモリ・カード・ソケットを取り付け、MCI を使って実際にメモリ・カードを読み書きしてみます。さらに、メモリ・カードの応用例として、カードに記録された画像ファイルを読み出し、それを OLED (Organic Light-Emitting Diode ; 有機発光ダイオード) ディスプレイに表示するところまでを実験します。写真1に

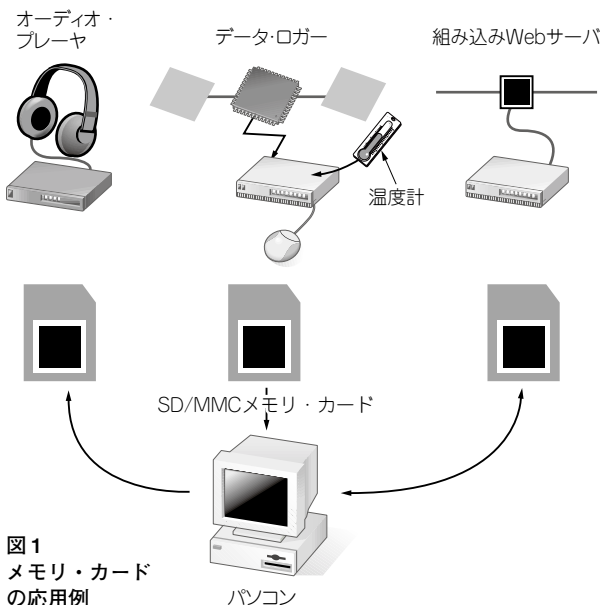


図1
メモリ・カードの応用例

筆者が試作したオリジナル・モバイル端末を示します。

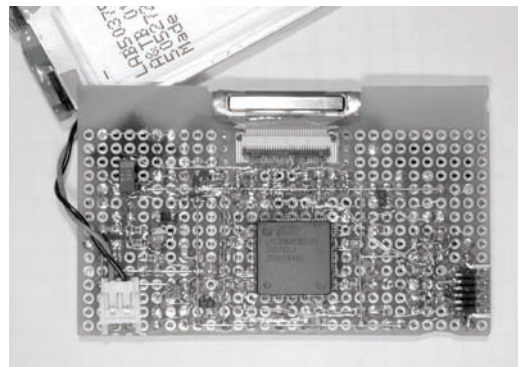
1 メモリ・カードの接続

● ソケット基板の回路

LPC 基板では MCI の信号がヘッダ (CN3) に接続さ



(a) 表面



(b) 裏面

写真1 製作したオリジナル・モバイル端末 (LPC2368 を採用)

第13章

PINGの要求&応答やテスト・パケットを送受信できる

LPC2388 内蔵 LAN コントローラを使った ネットワーク・テスト・プログラムの作成

サントシュ・パワル

LPC2388 には、Ethernet コントローラの論理層が内蔵されている。これを活かすために Interface 2009 年 5 月号付属 ARM マイコン基板には、物理層チップを搭載した LAN 拡張子基板が用意されている。ここでは LPC2388 基板と LAN 拡張子基板を使って、ARM マイコン基板をネットワークに接続するために必要な、Ethernet ドライバの作成事例を解説する。
(編集部)

オランダ NXP Semiconductors 社の LPC2388 は、ARM7TDMI コアをベースに、UART、USB ターゲット&ホスト、CAN、SPI、SSP、I²C、I²S、そして Ethernet コントローラなど多数のシリアル通信コントローラを組み込んだワンチップ・マイコンです。

ここでは LPC2388 マイコンの Ethernet コントローラの使い方を解説します。具体的には、LPC2388 マイコン内蔵 Ethernet コントローラのドライバを作成し、PING パケットの要求や応答に対応したネットワーク・テスト・プログラムを完成させます。

1 LPC2388 内蔵 Ethernet コントローラの紹介

● LPC2388 は論理層までを内蔵

図1に LPC2388 内蔵 Ethernet コントローラのブロック図を示します。Ethernet コントローラは大きく論理層 (MAC : Media Access Control) と物理層

(PHY : Physical Layer)に分けられます。LPC2388 には論理層までが内蔵されています。内蔵 Ethernet コントローラの特徴を以下に簡単に示します。

- 10Mbps (半2重/全2重), 100Mbps (半2重/全2重) 通信モードをサポート
- アクセスを制限するための VLAN フレーム定義可能
- Ethernet 専用内蔵 SRAM (16K バイト) を使った高速データ送受信
- 送信バッファ管理と受信バッファ管理を別々に行える
- パケット・データ転送専用 DMA バス・コントローラで CPU バスへの負担が少ない
- 受信パケット・フィルタリングが可能
- ハッシュ CRC 自動計算でハッシュ・テーブル・フィルタリングが可能

Ethernet 専用の内蔵 SRAM 以外のメモリを、送受

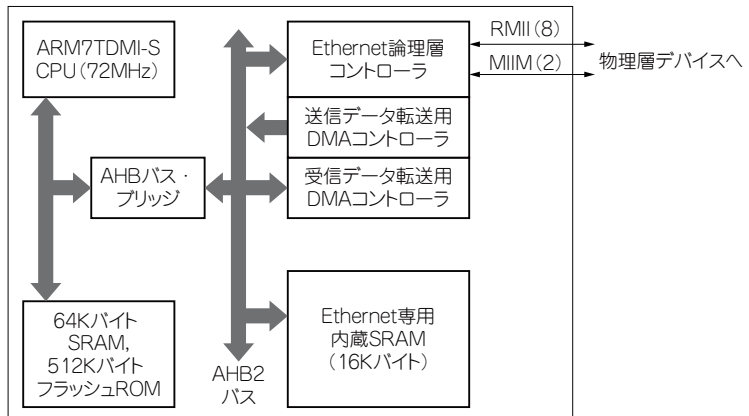


図1
LPC2388 内蔵 Ethernet コントローラのブロック図

第14章

安価な USB 接続 JTAG ツールを使った ARM7TDMI のソフトウェア開発

LPC2214/2138 の JTAG 機能と SPI 機能の活用事例

富田 寛美

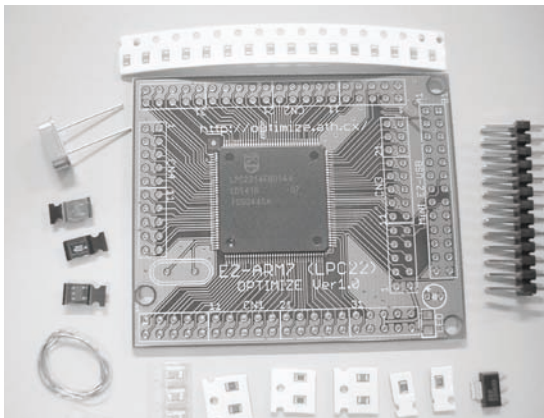
本章では、LPC シリーズの LPC2214 と LPC2138 を取り上げる。安価な評価ボードや、JTAG 経由でデバッグ可能なソフトウェア開発環境についても紹介する。ARM7TDMI の JTAG 制御事例について具体的なプログラムを示すので参考にしてほしい。
(編集部)

オランダ NXP Semiconductors 社 (以下 NXP 社) の LPC シリーズはラインナップも豊富です。第13章までは LPC2388 について詳しく解説しました。本章では同じシリーズの LPC2214 と LPC2138 について取り上げます。

1 お手軽に開発できる CPU ボード

● EZ-ARM7/mini EZ-ARM7 とは

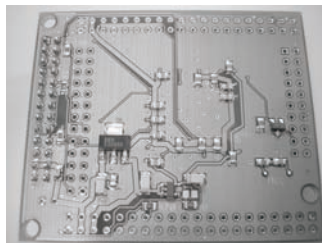
EZ-ARM7 (写真1) は、高性能な組み込み用 ARM7



(a) キットの状態



(b) 完成状態 (表)



(c) 完成状態 (裏)

写真1 組み込み用 ARM7 ボード EZ-ARM7 の外観

ボードを「お手軽 (EZ) に開発できる」ことを目的に開発しました。お手軽に開発するためには、CPU ボードの価格も重要ですが、開発環境が容易に入手できる必要もあります。

英国 ARM 社は自社では CPU の製造は行わず、設計のみに徹してアーキテクチャをライセンスすることをビジネスにしています。このためか、ARM CPU に関する情報は、個人でも簡単に同社の Web サイトから入手できます。

国内の CPU メーカーなどを含めていえることですが、通常は JTAG デバッグに必要な情報などは公開されません。そのため簡単に JTAG デバッガを作ることはできないのですが、ARM の場合はこの情報が公開されているのです。

そこで米国 Cypress Semiconductor (以下 Cypress) 社の EZ-USB を使用した mini EZ-USB (写真2) を JTAG プロブとして使用する、GDB 用のリモート・スタブと、JTAG 経由でフラッシュ ROM に書き

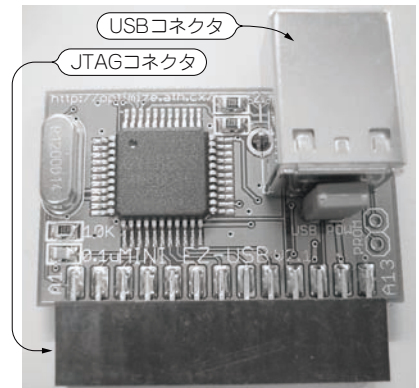


写真2 JTAG プロブとして使用する mini EZ-USB の外観

Appendix 6

ほかにもARM7マイコンを出荷している会社を紹介

まだまだある ARM7 マイコンのいろいろ

熊谷 あき

1 Atmel社製 AT91SAM7シリーズ以外のARM7マイコン

本書の第3部では米国Atmel社製ARM7マイコンとしてAT91SAM7シリーズを紹介しています。同社ではAT91SAMシリーズの前にもARM7マイコンのAT91x40シリーズを出荷していました。表Aに同シリーズを示します。現在の主力はSAM7シリーズになっているので、新規設計のシステムへの採用は控えめの方がよいでしょう。

とはいえ、秋月電子通商(写真A)などではAT91R40807を搭載したARM7キットが販売されています。たまたまひょんなことから入手した、またはちょっと

実験的に購入して使ってみる…ということもあるでしょう。

そのような場合は、Interface 2006年11月号の特集にある『第2章 Atmel AR91R40807の内蔵メモリの活用法』を参考にしてください。秋月電子通商から販売されているキットの使い方が紹介されています。

2 Cirrus Logic社製 ARM7マイコン

米国Cirrus Logic社もARMコアを搭載した各種コントローラを出荷しているベンダです。表Bに同社製ARM7マイコンの一覧を示します。

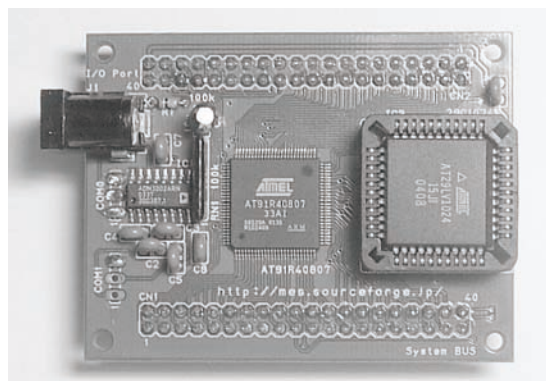
<http://www.cirrus.com/jp/products/>

[pro/techs/T7.html](http://www.cirrus.com/jp/products/pro/techs/T7.html)

また、世代としてはすでに古いCPUですが、CS89712というARM720Tコアがありました。これを搭載したARM7マイコンを搭載した評価ボードとして、Armadillo(写真B、アットマークテクノ)があります。紹介記事としてInterface 2002年7月号『Armadilloの概要と使用方法』があります。こちらも参考にしてください。

3 OKIセミコンダクタ社製 ARMマイコン

日本のほとんどの半導体ベンダはARMのライセン



写真A AT91R40807搭載ARM7キット(秋月電子通商)

表A
Atmel社AT91x40
シリーズ

型番	プライマリSRAMバンク	セカンダリSRAMバンク	ROM
AT91M40800	8Kバイト	—	—
AT91R40807	8Kバイト	128Kバイト	—
AT91M40807	8Kバイト	—	128Kバイト
AT91R40800	256Kバイト	—	—

表B Cirrus Logic社製ARM7マイコン一覧

型番	CPUコア	クロック(MHz)	キャッシュ	MMU	オンチップRAM(バイト)	対応DRAM	パッケージ
EP7309	ARM720T	74	8Kバイト・ユニファイド・キャッシュ	あり	48K	—	208ピンLQFP 256ピンPBGA
EP7311		90~74				SDRAM	
EP7312							