

ハードを隠ぺいしたお任せ OS を組み込みで使う可能性を探る

実験リサーチ!

カーネル内部
とことん可視化計画

Linux 応答時間の実力

第9回

ソフトだけじゃできない! データ出力周期12ns!
専用回路を使った高速データ転送DMA

海老原 祐太郎

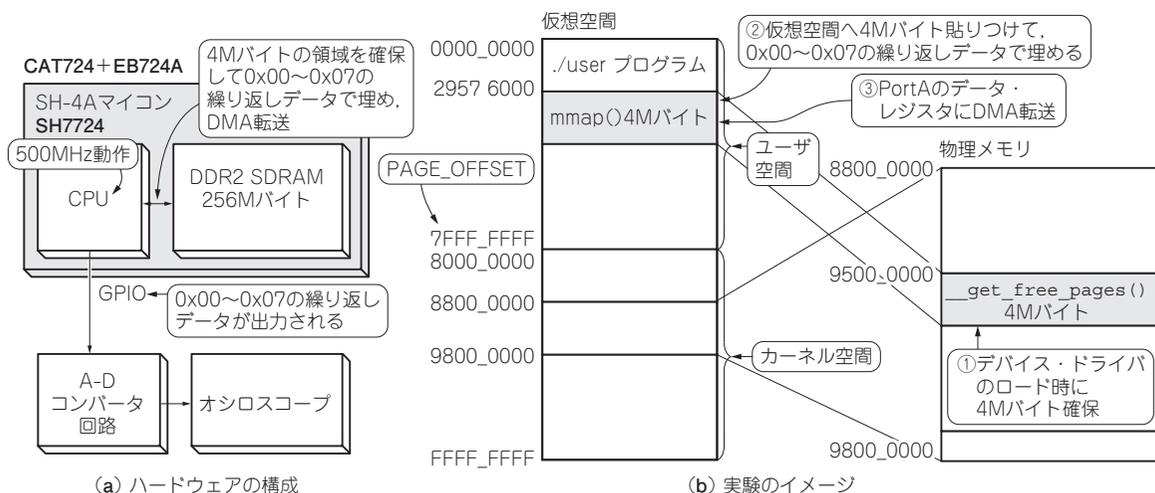


図1 4Mバイトの仮想メモリを確保してGPIOにひたすらDMA転送してみる

本当は12ns周期で高速データ転送を行えるが、確認しやすくするために256分周した約3 μ s (12ns \times 256)周期で実験を行った

割り込みを用いて周辺デバイスとデータ転送を行う方法はとても一般的ですが、割り込み応答速度によってリアルタイム処理性能は頭打ちとなります。

Linuxでは割り込み禁止期間が長いこともあり、ワースト・ケースにおいては1ms以下(1kHz以上)の割り込みは間に合わないことがあります。1ms以下の時間の制御はリアルタイムOSに比べると劣るといえます。しかし、ハードウェアDMA (direct memory access) 転送を使うことによりLinuxが苦手としている高速なデータ・サンプリングやデータ転送が実現できます。

DMAはCPUを介さずメモリと周辺デバイス間でデータ転送を実現するしくみです。

今回は、Linuxの苦手とする高速なデータ転送を、専用DMAコントローラ回路を使って実現してみます。

こんな実験

図1に今回の実験を示します。4Mバイト分のデータをDMAでGPIOポートに出力してみます。

- (1) 物理メモリを4Mバイト確保
- (2) ユーザ・プロセスの仮想空間へ4Mバイトのメモリ(0x00~0x07の繰り返しデータ)を貼り付ける
- (3) DMA転送でGPIOポートAへ出力
- (4) GPIOポートAに3ビットのD-A変換回路を接続しオシロスコープで波形を観察する

● ハードウェア

実験に使用するボードは500MHz動作のSH-4AマイコンSH7724(ルネサス エレクトロニクス)を搭載したCPUボードCAT724+ベース・ボードEB724(シリコンリナックス)です。

本稿ではSH-4Aを題材としますが他のアーキテクチャでもほぼ同様です。カーネルのバージョンは3.0.4です。本稿で紹介する全プログラムやビルド方法、ロード方法などは筆者のサイト(<http://www.sil-linux.co.jp/>)に掲載いたします。

▶ GPIOポートに接続するD-A変換回路

ポートAに接続した3ビット簡易D-A変換回路を図2に示します。R-2Rラダー回路で構成しています。