

FM3ベースのキーボードコントローラ

FM3-based Keyboard Controller

● 小池信之

あらまし

富士通は、ノートブックPC向けに、富士通セミコンダクター FM3ファミリのワンチップマイクロコンピュータを使ったキーボードコントローラ兼パワーマネジメントコントローラを開発した。また、これに合わせてRTOS(Realtime Operating System)を採用した制御ファームウェアを開発し、多岐にわたる機能を持ちながら低消費電力とレスポンスの良さを両立させた。

Abstract

We at Fujitsu have developed a keyboard controller-cum-power management controller for notebook PCs, based on Fujitsu Semiconductors' FM3 microcontroller series (ARM® Cortex™-M3 processor core). We have also developed control firmware that adopts a realtime operating system (RTOS) to give the controller a variety of features with minimal power consumption, and yet make it highly responsive.

まえがき

コンピュータは何にでも使える技術である。その性能と価格が折り合えば、どんな場所に使っても良い。今日では、マイクロコントローラ（以下、マイコン）を内蔵した家電製品が当たり前になっている。面白いことにパソコンにも、マイコンが組み込まれている。ハードディスクの制御や、無線LANの制御にそれぞれマイコンが使われているのはごく普通のことである。

この状況は最初のパソコンが作られた頃には既にあって、キーボードの制御にマイコンを使うのは普通のことであった。このマイコンはキーが押されているかどうかを調べ、そのキーに従ったキーコードを作り、メインCPUがこれを読み出すまで保持するという機能を担っている。これを「キーボードコントローラ」と呼んでいる。ノートブックPCは電池で動作し、この電池の残量監視や充電制御などについてもマイコンに処理させることが多い。これを「パワーマネジメントコントローラ」と呼ぶこともあるが、あまり一般に使われることのない名称である。

今ではこれら二つの機能を統合して一つのマイコンで処理するのが一般的であり、そしてほかにも欲しい機能が議論に上がってくれば、自然とこのマイコンに統合することになる。単にキーボードコントローラと呼ばれることが多いが、多くの機能が統合された結果キーボード以外の処理の方が多いほどである。

富士通は、ノートブックPC向けにこれらの多くの機能を低消費電力と高レスポンスで実現するために富士通セミコンダクター FM3^(注1) ファミリのワンチップマイクロコンピュータを使用したキーボードコントローラを開発した。

キーボードコントローラの機能

富士通のノートブックPCでこのようなマイコンを使うのは、今回が初めてというわけではない。ごく初期には4ビットマイコンを使っていたことも

あるが、世代を重ねるごとに順当に進化してきており、ここ数年は16ビットマイコンを使っていた。しかし要求される機能が増えるに従って処理量が増え、そろそろ既存のマイコンではこれ以上の機能追加も難しくなっていた。

幸いなことに、実績のあるRTOS (Realtime Operating System)^(注2) や、進歩した半導体技術を低コストで利用できる時代であり、むしろ旧世代の半導体デバイスを使い続けようとする製造設備の維持などの問題も出てくる。良い機会と考えて今回は32ビットマイコンを使って上位互換のコントローラを開発し、これをEastChipと名付けた。PCのブロック図を描くときに、メインCPUを上(北)に、各種機能モジュールを下(南)に描くことが多く、「東の一面を貫こう」という意気込みである。

ところでマイコンという言葉は超小型コンピュータという意味であるが、これを英語圏ではEmbedded Computer, もしくはEmbedded Controller, そしてその略でECという呼び方をする。「組み込んで使うコンピュータ/コントローラ」という意味である。このためキーボードコントローラは、より一般的な名称としてECと呼ばれることがあり、これを連想させる名称となっている。

ベースとなったのは、富士通セミコンダクター FM3ファミリのマイコンである。384 KバイトのFLASHメモリと、48 KバイトのRAMを持ち、32ビットのCPUコアを16 MHzで動作させている。調べてみると初期のPC/AT (Personal Computer/Advanced Technology) は、6 MHzの16ビットマイコン、メインメモリ256 Kバイトという構成で出荷されたとのことで、これを考えるとこのマイコンにはこのようなPCが丸ごと一つ入っているような状態である。これだけの性能が要求されるほど、このマイコンの担う役割は多岐にわたる (図-1)。

基本になっているのは、キーボードのスキャン、メインCPUからの問合せへの応答、タッチパッドなどのポインティングデバイスとの通信、電池の残量測定、充電制御、交換時の切替え制御などである。

(注1) ARM® Cortex™-M3搭載、汎用32ビットマイコンの製品群。性能を重視して144 MHz動作ができるもの、省電力を重視して1.65 Vで動作できるものなど、多数の製品がある。⁽¹⁾

(注2) 割込みに対する応答時間の最悪値を規定・保証したOS。OSの提供する各種サービスには、割込み処理を止める必要があるものがあるが、このようなサービスも最悪値を規定・保証することが求められる。悪くても10 ms程度である。

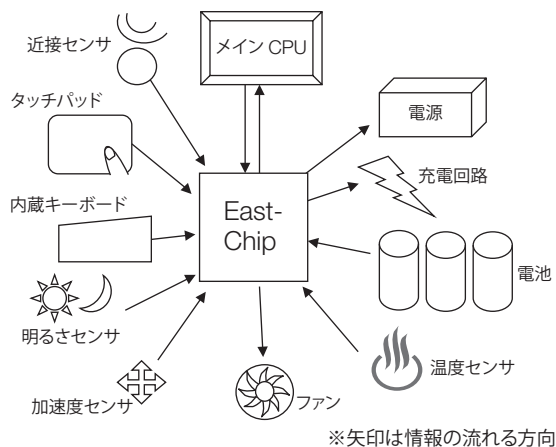


図-1 ノートブックPCにおけるマイコンの役割

ファンの制御は、メインCPUの温度に応じた回転数でファンを回し、もし過熱状態になればその旨を伝え、一定時間以内に状況が変わらなければ電源を切って熱破壊からシステムを守る、というものである。このため温度センサの情報読出しとファンの回転数測定も行っている。

電池・充電回路制御は、電池の電圧・電流・温度を監視し、ACアダプタが接続されていれば電池を充電するというものである。電圧や電流を測定し、異常があればシステムから切り離して警告も出す。

各種センサ類はI²Cなどのシリアルバスでつながっているため、これらの通信エラー検出、リカバリも行う。加速度センサを読んで、不安定な置き方をしていることを検出した場合はハードディスクのヘッド待避を指示するという機能も実現している。これら全てをマルチタスクでこなすことが求められる。

要求される性能とその実現

従来の製品で使っていた16ビットマイコンは、8 MHzで動作させ、アセンブラで組んだ独自のマルチタスクカーネルを使って処理していた。今回は全面的に見直してTRON系のRTOSを使い、C言語による近代的な開発ができるようにした。これにより将来の仕様変更や機能拡張に対応できるものと考えている。

今回採用したマイコンは最大動作周波数が32 MHzのものである。最近の半導体技術を考えてとそれほど高速なものではないが、消費電力抑制

を重視した品種であり、今回はこれをその半分の16 MHzで動作させることで電力を切り詰めている。この結果、消費電力は最大でも数10 mWに過ぎない。またPCとして稼働していないときは省電力モードになっているので数mW、あるいはそれ以下となる。つまりマイコンのコアやバスの動作クロック周波数を動的に切り替えることで更に消費電力を抑えている。

このように頻繁に動作クロック周波数を切り替えるシステムでは、スケジューリング(時間の管理)が難しくなる。一般にOSのスケジューリングはおろそかにできないが、特に今回のような電池を扱うシステムの場合は、充電不足や過充電、ひいては電池容量の低下につながる。よって腕時計に使われるような高精度の32 kHzのクロック発振回路を常時動作させ、これを使ったタイマ群を用意した。これらは低消費電力でありながら1か月以上カウント動作を続けてもオーバーフローしないようになっており、動作クロックの切替え処理を簡単にしている。またこの32 kHzから、内蔵するPLL周波数乗倍回路を使ってマイコンの動作に必要な全ての周波数を作っている(図-2)。

FM3ファミリのマイコンは、既存の富士通セミコンダクター製8ビット・16ビットマイコンで実績のある周辺モジュールを流用できるように整備されているが、今回は実績よりも「なるべくRTOSに合った」周辺機能を選び、処理の効率を上げている。大まかに言って「RTOSに合うか合わないか」は、割込み発生頻度で言い換えることができる。常識に反してRTOSは、微小時間の管理が苦手である。例えば100 μsないし1 ms程度の幅を持ったパルス精度良く出力しようとする、その間はほかの仕事ができなくなってしまう。この範囲の時間制御を必要とするところはできるだけハードウェアで処理してしまう方が良い。例えばハードウェアのパルスジェネレータ機能を利用できれば、RTOSとは独立して精度の良いパルスを作ることができる。パルスジェネレータは既存のモジュールを流用している一方で、同様な時間精度でパルス幅を測定するモジュールは新規に起こしている。新しく起こしたモジュールは、仕様策定からハード設計までをファームウェア開発者が行ったものであり、なるべくRTOSに負担をかけないように、「ソ

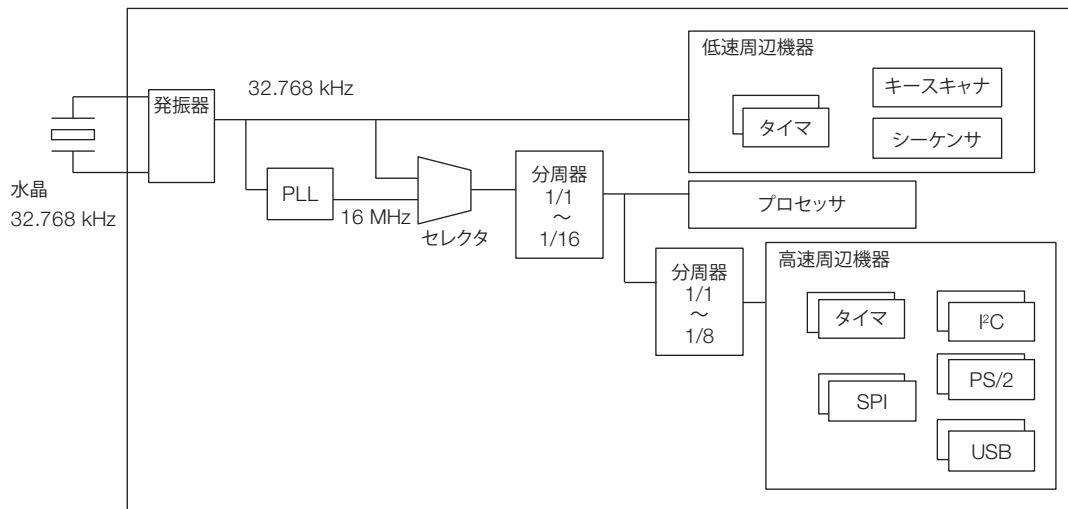


図-2 クロック分配ブロック図

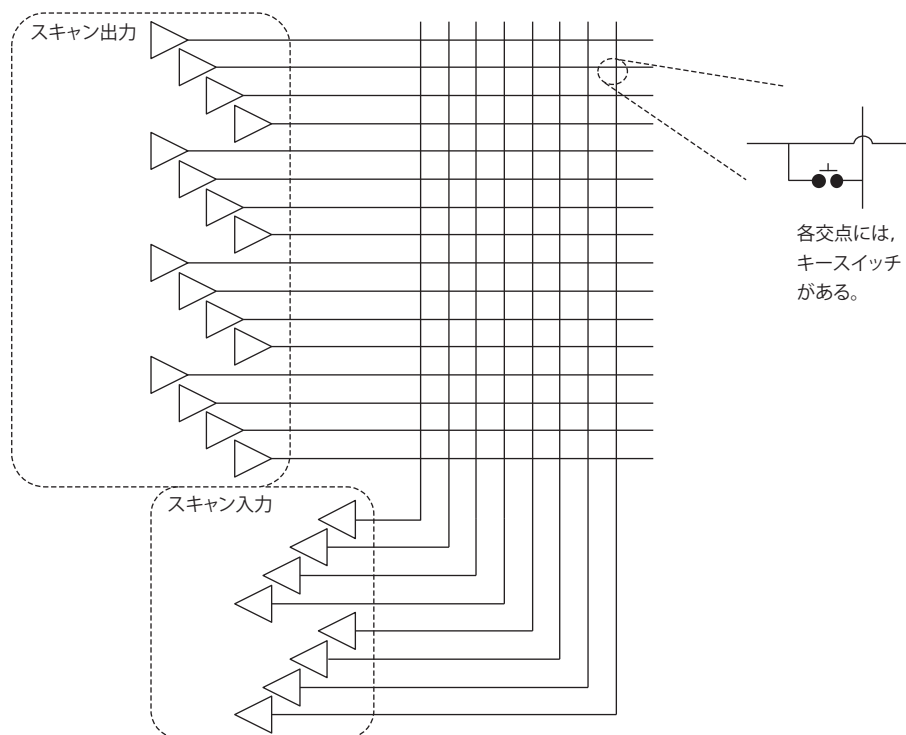


図-3 キースキャン回路

ソフトウェアは初期設定をしたら割り込みを待っていれば良い」という方針で設計されている。例えばシリアル通信モジュールには大きなバッファを持たせて、パケットを丸ごとバッファできるようにしている。

外部割り込み入力として使える端子は70本を超え、これら全てがエッジ検出機能を持つ。内部のもの

を合わせると割り込み発生源は100本近いので、専用の割り込みコントローラを追加して対応している。無駄とも思える規模ではあるが、おかげで割り込みハンドラの開発やメンテナンスの負荷が下がっている。キーボードコントローラとして使うことを基本にしているので、キースキャン回路は充実した仕様となっている (図-3)。PCで使われるキー

ボードは、キースイッチを8行16列程度のマトリクス状に配線したものである。この場合16本のストロブ線を順次ドライブしていき、その都度8本の入力線の状態を読み取ることになる。

このように簡単な回路・処理ではあるが、実際に使おうとすると難しいものである。まず、ストロブ線をドライブしてから入力信号が安定するまでに数10 μs かかるので、その分の待ちが必要である。次に、スイッチには、押し離し動作時の機械的な振動で、約20 msの不安定な期間がある。この間はスイッチが入ったり切れたりして不安定になるので、このまま読み取るとキーを連打しているように見えてしまう。よってこの時間も待つ必要がある。

このようにキースキャンは時間のかかる処理であり、しかもRTOSにとっては苦手な時間領域の処理である。ソフトウェアで処理しようとする「ほとんど待っているだけなのに処理負荷が大きい」ということになってしまう。今回はこれをハードウェアで実装することにより、「ほかの処理をしている間にキースキャンができていく」という状況にしている（図-4）。しかも常時供給する32 kHzを

使って自動スキャンするようになっており、マイコンのコアがスリープ状態であっても動作できるようにになっている。

ファームウェアの開発

要求される機能はハードウェアでほとんど準備されているので、制御するファームウェアがこれらを扱うことについて問題はない。RTOSの骨組みの中に要求仕様に合わせて組み込んでいけば、できあがるはずである。唯一特別な対応をしたのが、先に述べたスケジューリングである。ここはRTOSの根幹の部分でありながら、ハードウェアに依存する部分であり、標準的な方法が確立されていない。そのためRTOS自体を拡張する形で実現した。全体としてはごく標準的な使い方をしている。

しかし歴史のあるシステムによくある話で、仕様の細部がいちいち設計の妨げになった。以前より高性能なマイコンを使っているのにひどく処理時間がかかる部分があったり、大幅に増やしたのにメモリが不足したり、更には今まで気づかなかった仕様の不備が見つかったりと、様々な問題に苦しんだ。

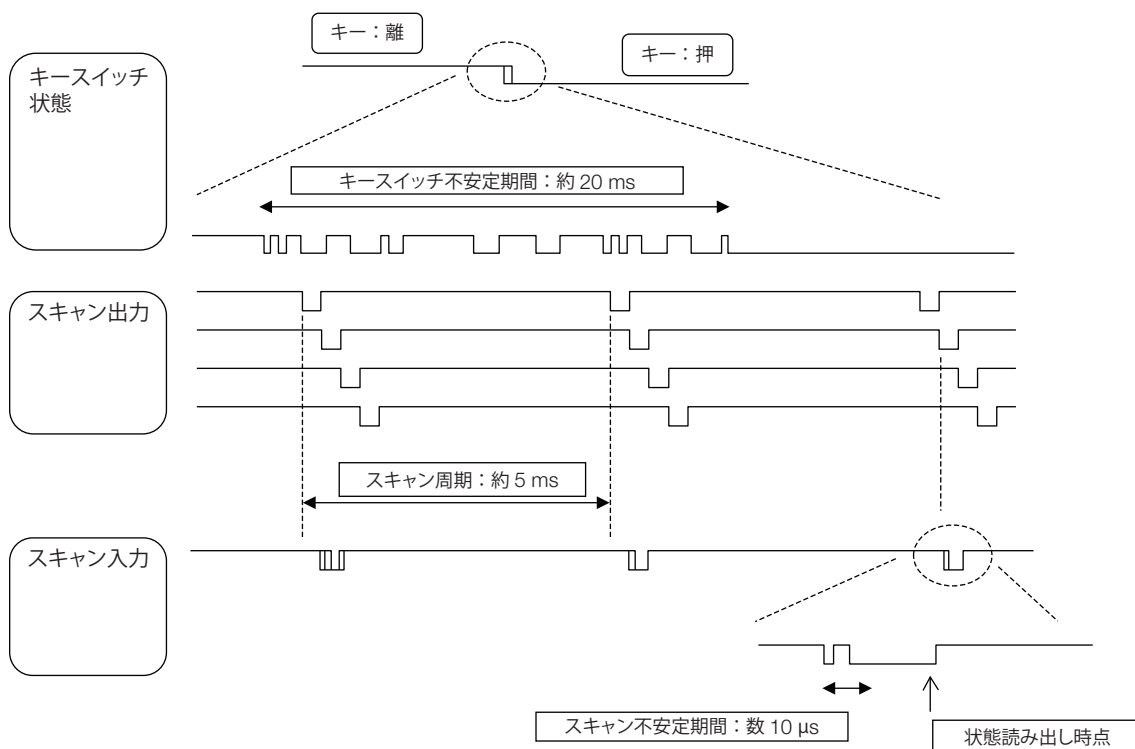


図-4 キースキャンタイムチャート

とは言え動作テストや評価をする側の歴史も負けてはいない。富士通が経験してきた問題はデータベース化されており、またその要点はチェックリストにまとまっている。これを追うだけでもほとんどの問題が見つかる。更にユーザの操作を想定したシステム全体の動作テストも繰り返し行われ、問題が見つければ報告される。

著者らは見つかる限りの問題を見つけ出し、その全てを解決した。新しいチェック項目が生まれ、仕様の不備も修正された。

む す び

コンピュータの価値は処理性能とともに、「何にでも使える」ことで決まる。とは言え実際の使い方はある程度決まった仕事ばかりである。何にでも使えるようにそれぞれに特化したハードウェアを集積していくと無駄にコストが膨らむので、メインCPUに何でも処理させて専用部分を最小限に抑えるという手法が使われることが多い。確かに近年ではメインCPUが処理能力を持て余しているので、キーボードコントローラの制御を全て肩代わりすることも可能である。しかしこうすると少なくとも数Wの消費電力増加を招くことになり、電池動作をするノートブックPCにそぐわない。電

池寿命を延ばせるなら専用ハードウェアの採用も検討すべきである。

同じことがマイコンの使い方にも言える。なんでもかんでもソフトウェアで処理させると無駄な電力を消費することになる。例えばこれだけの性能を持つマイコンであれば、シリアル通信を全てソフトウェアで行うこともできるが、その分の電力増加は無視できないものとなる。通信制御をハードウェアで実現すれば、回路は増えるが消費電力は下がる。

ノートブックPCには「高性能」と「低消費電力」という矛盾する特性が求められ、またこれらに折り合いをつけることで価値が高まる。キーボードコントローラは、その矛盾を受け止める部分である。

本稿では、今日ノートブックPCのキーボードコントローラに求められる機能とその実現について述べた。これらは既に基本機能であり、今後も更なる機能拡張と追加を求められると考えるが、できる限り応えていきたい。

参考文献

- (1) 富士通: FM3ファミリ Fujitsu Cortex-M3マイコン.
<http://jp.fujitsu.com/microelectronics/products/micom/roadmap/industrial/fm3/>

著者紹介



小池信之 (こいけ のぶゆき)

パーソナルビジネス本部第一PC事業部
所属
現在、キーボードマイコンのファーム
ウェア開発に従事。