

新16ビットマイコン78K0Rシリーズ

溝口 誠・石川 潔・中野 正隆
熊谷 敏幸・磯貝 英夫・田中 健太郎

要 旨

高性能化、低消費電力、開発期間短縮など、組み込み市場のニーズに対応するため、「16ビットマイコンの性能を8ビットマイコンの消費電力で」「8ビットマイコン（78K0）に対する上位互換性確保」「快適な開発環境の提供」をコンセプトに、新16ビットマイコン78K0Rシリーズを開発しました。消費電力を削減するため、先端フラッシュ混載CMOSプロセスの採用、CPUコアを含めた回路的、システムの工夫を行っています。78K0Rは、様々な分野のお客様に高性能と低消費電力を同時に提供していきます。

キーワード

- マイクロコンピュータ ●16ビットマイコン ●フラッシュメモリ ●フラッシュマイコン
- 高性能 ●低消費電力 ●組み込み機器 ●開発環境

1. はじめに

近年、組み込み機器の裾野は大きく広がり、システムを制御するために、マイクロコンピュータ（以下マイコン）があらゆる機器に搭載されるようになりました。また、この拡大傾向は今後も続きます。

このような組み込み機器では、お客様は常に高機能化を求める傾向にあり、システムを制御するためのマイコンにも、機能・性能の向上が求められています。同時に、エコロジーへの関心の高まりから、電力消費量削減が重要なポイントとなり、マイコンにも強く求められています。

このような市場要求に対し、新16ビットマイコンである78K0Rは、16ビットの性能を8ビットクラスの消費電力で実現し、消費電力の増加を抑えつつ機器の性能向上を実現したいというお客様のニーズを満たし、お客様の競争力向上に貢献します。

2. 新16ビットマイコンのコンセプト

従来、NECエレクトロニクスでは、78K0シリーズという8ビットマイコンを開発し、様々な製品を展開してまいりました。

しかしながら、近年の機器の高機能化に伴い、従来8ビットマイコンを使用していたお客様において、「8ビットマイコン

では性能に余裕が無い」、「今後不足する」というケースが増加していました。このようなお客様は、16ビットマイコンへの移行を検討されます。しかし、現存する弊社製および他社製16ビットマイコンでは、「消費電流が増加してしまう」、「ソフト資産の流用ができない」などのニーズとのギャップがありました。また、マイコンを使用するには、ソフトを開発するための開発環境が必要です。開発環境には、コンパイラ、デバッカなどのソフトウェア環境と、マイコンの動作をエミュレーションするハードウェア環境の2種類が存在します。

この中で、まずソフトウェア環境についてですが、コンパイラの性能（コード効率）の向上、ドライバレベルのソフトウェアを簡単に開発できるツールの提供など、使いやすい環境に対するニーズがありました。また、ハードウェア環境は、マイコンそのものを使用するため、マイコン自身ができてからしか提供できず、お客様のセット開発期間短縮に対するボトルネックとなる場合があり、早期に提供してほしいというニーズや、8ビットマイコンである78K0と同一の環境を使用できるようにして欲しいというニーズが存在しました。

そこで、これらのニーズを満たすことをめざし、「16ビットマイコンの性能を8ビットマイコンの消費電力で」「8ビットマイコン（78K0）に対する上位互換性確保」「快適な開発環境の提供」をコンセプトとする新16ビットシリーズの開発を行いました。

3. 16ビットの性能を8ビットの消費電力で

第2章の通り、市場は高性能と同時に消費電力の少ないマイコンを求めており、本製品の開発にあたっては、単に処理性能の向上だけではなく、従来の8ビットマイコンや16ビットマイコンと比して電力性能比が小さくなる低消費電力マイコンをめざしました。処理性能向上のために動作周波数を上げることは消費電力の増加を招くことから、CPUをパイプライン化し動作周波数を上げずに性能を向上させました。また、従来製品の解析により電力削減効果が期待されていたクロック供給制御を消費電力削減のための課題として取り組みました。

1) CPUのパイプライン化

弊社の8ビットマイコンである78K0は、1つの命令を実行するのに平均4クロック程度を要していました。動作周波数を上げれば性能は上がりますが、それに比例して消費電力が増加します。よって、本製品では、1クロックで1命令の実行ができるパイプライン方式を採用しました。パイプライン方式とは、CPUの動作を複数のステージに分割し、各ステージを並行して動作させることにより全体の処理速度を向上させる方式であり、動作周波数を変えずに処理性能を向上させることができます。パイプライン方式の採用決定とともに、今回ターゲットとする消費電流と性能を達成するには、どのようなパイプライン構成が最適かを検討しました。その結果、3段パイプラインが、消費電流が最小かつ目標性能を達成できる構成であることを確認でき、採用しました。このようにして、周波数を上げることなく性能を向上させることができました（図1）。

2) 周辺機能のクロック制御

従来製品は、周辺機能が未使用の状態であってもクロック

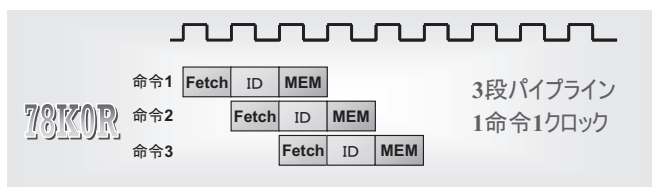


図1 パイプライン

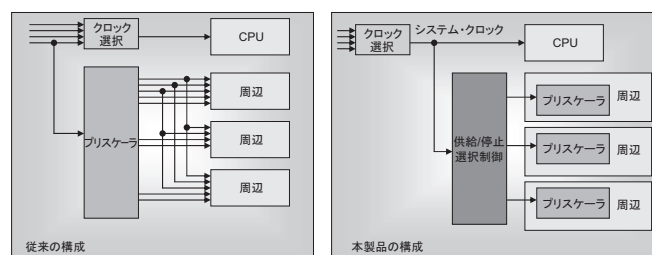


図2 クロック供給経路

供給回路が動作し続け、スタンバイ時に動作電流の約20%を無駄に消費するケースがあることが判明していました。図2に示すように、プリスケアラ回路で数種のクロックを生成し、それらを個々に周辺機能へ供給していたため、周辺回路の使用いかんにかかわらず、クロック生成に伴う電流や周辺回路への配線経路自体の負荷による電流を無駄に消費していました。この対策として本製品は、周辺機能ごとにクロックを自由に供給停止できる仕組みを搭載しました。図2のように、周辺機能へ供給するクロック信号を1本とし、プリスケアラは各周辺機能内に配置すると同時に、周辺機能停止時にはクロック信号の供給をクロック生成回路部で停止できるように制御できる機構を搭載しました。この結果、停止している周辺機能で消費していた無駄な電力を“0”にすることができました。

また、スタンバイ時の消費電力削減も重要なポイントです。通常動作時には使用するが、CPU停止中には動作不要となる回路を詳細に調査し、それらの回路へ供給するクロックをクロック生成回路部内で停止する制御を追加しました。前述の周辺機能のクロック供給制御と合わせ、全周辺停止時のスタンバイ電流を、従来通りのクロック制御を搭載した場合の予測に対して約40%削減することができました。

このような取り組みを行うことにより、図3に示すように、16ビットマイコンで業界最高の電力性能比（1MIPS当たり1.8mW）を実現し、従来の弊社16ビットマイコンに対して、電力性能比を1/3以下にすることができました。

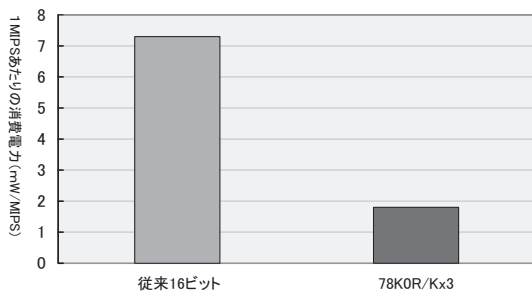


図3 電力性能比の比較

4. 8bitマイコンに対する上位互換性確保

弊社の8ビットマイコンである78K0のソフトウェア資産を活かしつつ、性能の向上を達成するため、命令セットとアセンブラを最適化し、さらにメモリ空間の拡張を実現しました。

(1) 命令互換性確保と性能の向上を同時に実現

78K0Rは、アセンブラでの自動命令変換なども取り入れ、アSEMBL言語のレベルで78K0に対して、命令互換を達成しました。さらに、拡張したアドレス空間の指定に関しては、16ビットのオペラントで20ビットのアドレス空間を指定できるように、上位4ビットのアドレスを独立して指定するCSレジスタを実装することで、78K0に存在するレジスタ・ペア分岐や、16ビット絶対アドレス分岐命令をそのまま使用できるようしました。

次に、性能の向上です。まず、16ビットの配列データを取り扱う際などに、そのアドレス計算に多用される乗算命令(MULU)の代わりに、2倍、4倍や1/2、1/4の乗除算処理を高速に処理できるシフト命令(SHR、SHLなど)を追加しました。さらに、パイプライン化により演算命令などが高速化した一方で、分岐命令による遅延対策として、条件判断結果次第では実行されない後続の命令を無効化する、スキップ命令(SKZ、SKCなど)を実装しました。これにより、分岐命令では1分岐処理あたり2クロックかかるはずの処理を1クロックで行うことができるようになりました。これらに加えて、16ビット演算命令などを追加することによって、Dhrystone1.1において13MIPSの高い性能を実現しました。このようにして、図4に示すように78K0と互換性を保ちつつ、性能を強化することができました。

(2) メモリ空間の拡張

メモリ空間はリニア・アドレスで最大1MBとしました。

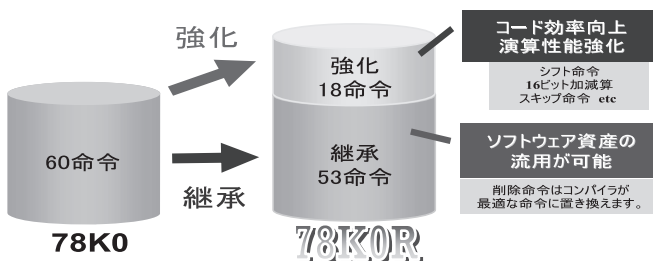


図4 78K0Rでの命令追加

16bit市場で要求されるROMメモリサイズ、および内蔵高速RAMやSFR (special function register) 空間を考慮して設定しました。空間の拡張は、BANK 切り替え方式 (一次元配列のメモリ空間を多次元配列化し、アドレスバスのビット数を増やすことなくメモリ空間を増やす方法) ではなく、より使いやすいリニア・アドレス方式 (一次元配列のメモリ空間) を選択しました。

メモリ空間の拡張に伴い、いくつかの工夫を行いました。その中の1つが、プログラムエリアに配置するデータ参照を行うアドレス指定方法の新設です。1MBのメモリ空間のアドレス指定 (オペラント) には、3バイト必要です。必然的にオペコードは1バイトでまかなわなければならなくなります。しかし、オペコードが1バイトの命令は、性能、コード効率上重要となる命令を優先的に命令マップ上に配置する必要があり、通常64kバイトのアドレスしか指定できない2バイトのオペラントで、1MBメモリ空間のアドレスを指定する仕組みが必要でした。そこで、上位アドレスの4ビットを独立して指定するESレジスタを設け、PREFIX命令を準備しました。図5に示すように、PREFIX命令を実行した直後の命令のオペラントで示されるアドレスの上位4ビット

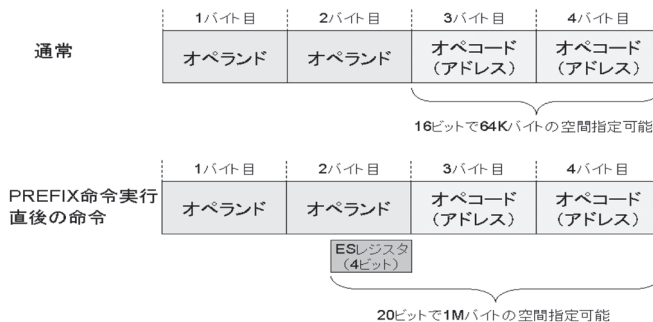


図5 ESレジスタとPREFIX命令

に、ESレジスタの値を付加する仕組みを入れました。これにより、20bit分のアドレスを指定できるようになり、2バイトのオペランドで1MBメモリ空間のアドレス指定を実現しました。

5. 快適な開発環境の提供

78K0R用の開発環境として、ソフトウェアの開発を快適にし、かつ8ビットマイコン78K0と共通で使えるツールなどを準備しています。具体的には、デバイス・ドライバ・コンフィギュレータ(マイコン周辺の制御プログラムの作成支援ツール)であるApplilet2やソフトウェア・シミュレータであるSM+, 8ビットマイコン78K0と共通で使えるオンチップ・デバック・エミュレータであるMINICUBE2や早期提供可能な高機能インサーキット・エミュレータ(以下、フルICE)であるIECUBEを開発しました。また、16ビットマイコンではC言語での開発が増えることから、コード効率を強化したコンパイラCC78K0Rを開発しました。これらのなかでも、特にコンパイラであるCC78K0RとフルICEであるIECUBEについては、コンセプトの1つになっている「快適な開発環境の提供」をより強固なものとするために新たな取り組みを行いました。

まずはコンパイラについてです。CC78K0Rは、16ビットCPUコアである78K0R用であり、16ビットマイコンを採用されるお客様の多くがC言語でプログラミングされることから、そのコード効率を上げる(つまりコードサイズを小さくする)必要がありました。そこで、8ビットCPUである78K0で採用されていた技術に加え、最新の最適化技術を採用しました。この段階で他社ベンチマークを行ったところ、残念ながらこのままでは他社を上回ることができないことが判明しました。そこで、さらなるコード効率の改善の手法を検討しました。その結果、不特定の類似コードを検出、自動的にサブルーチン化し、1つにまとめること(これを自動サブルーチン化技術と呼びます)により、コード効率の大幅な改善が見込めることが判明しました。ただし、本技術の採用により、デバック情報を出しにくくなり、デバックし難くなるのではないかと懸念がありました。この点については、独自の自動検索アルゴリズムを開発し、デバック効率を落とすことなく最適化を行うことができました。結果として、図6に掲載するように、従来比約20%のコード効率改善を実現し、他社に

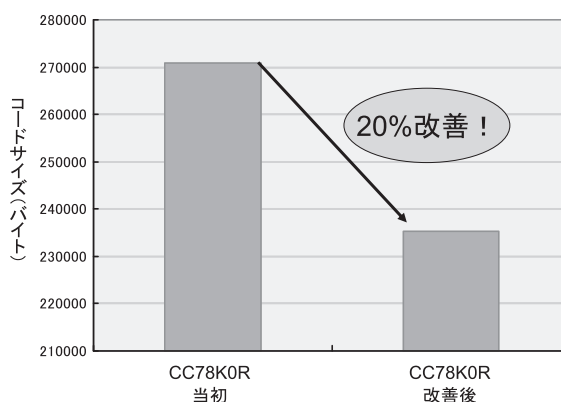


図6 コード効率の改善 (コードサイズ縮小)

勝るコンパイラを開発することができました。

次に、フルICEです。従来のフルICEは、マイコンとの機能等価性確保のため、エミュレーションチップ部分をマイコン自体で実現していました。このため、マイコン自体の試作が完了した後でしか提供できませんでした。しかし、78K0R用のフルICEはロジック機能の実現にFPGAを活用し、かつFPGAでは実現できないアナログ部分は他のデバイスとの組み合わせにより等価な機能を実現することで、図7に示すようにフルICEをマイコンの回路設計が完了した直後に提供できるようになりました。このように、早期に開発環境を提供することで、お客様はソフトウェアの開発に早く着手することができ、セットの早期開発に大きなメリットとなっています。

このようなFPGAを使ったフルICEは、これまででも限定的な顧客対応のためには用いられていた手法でした。しかし、FPGAを使用する手法の場合、大きな課題がありました。それは、マイコン回路に含まれる非同期回路をFPGAに入れること

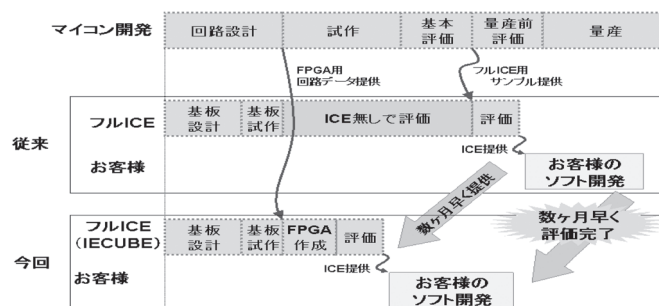


図7 フルICEの提供と顧客開発完了時期の差

ができず、同期回路に設計し直さなければならないという点でした。その結果、フルICEとして重要であるマイコンとの機能等価性という点で、完全等価とすることができない場合があります。さらに、その「非同期回路の同期化」に設計工数がかかるだけでなく、検証パターンなどもICE独自のものを用意する必要があるなど、マイコン回路設計が完了してから相応のフルICE開発期間が必要となり、早期提供を実現できませんでした。そこで、78K0Rではマイコン開発時から極力非同期回路を排除するなど、FPGAを使用することを考慮した回路設計を行いました。その結果、マイコンとの機能等価性が保たれたフルICEをマイコン回路設計が完了した直後に提供することが可能となりました。また、FPGAによるフルICEの実現は開発開始時のメリットにとどまらず、開発環境バージョンアップ時の負担も大幅に低減しました。従来は、バージョンアップ時に修理工場へ持ち込み、必要な部品の交換が必要でした。これに対して、78K0RのフルICEでは、FPGAデータの書き換えによりバージョンアップが可能となりました。この結果、お客様の手元にフルICEを置いたまま、お客様の開発に負担をかけずにタイムリーにバージョンアップ版を提供できるようになりました。

このように、ソフトウェア環境、ハードウェア環境ともに新たな取り組みを行うことで、他社を上回る快適な開発環境を実現し、市場のアプリケーションやセットの高機能化が進むなか、セット開発効率の向上に大きく貢献しています。

6. むすび

このように、新16ビットマイコン78K0Rは、「16ビットマイコンの性能を8ビットマイコンの消費電力で」、「8ビットマイコン（78K0）に対する上位互換性確保」、「開発環境の早期提供」のコンセプトを実現し、ニーズに対するギャップを埋めるとともに、お客様のシステム付加価値の向上に貢献することができます。

今回紹介した新16ビットマイコン78K0Rシリーズとして、汎用機能を備えた基本製品群である78K0R/Kx3をすでに市場へリリース済みです。しかしながら、市場には汎用品のみでは対応できない周辺機能を必要とするアプリケーションも多く存在します。今後は、図8に示すような「LCD C/D内蔵版」、「3相モータ制御機能内蔵版」など、アプリケーションに対応する機能を備えた製品群を拡充していきます。

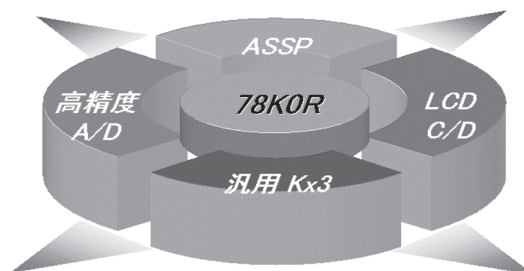


図8 78K0Rの今後の展開

執筆者プロフィール

溝口 誠
NECエレクトロニクス
マイクロコンピュータ事業本部
汎用マイコンシステム事業部
チームマネージャ

中野 正隆
NECエレクトロニクス
マイクロコンピュータ事業本部
汎用マイコンシステム事業部
主任

磯貝 英夫
NECエレクトロニクス
第二マイコン開発事業部
主任

石川 潔
NECエレクトロニクス
マイクロコンピュータ事業本部
汎用マイコンシステム事業部
主任

熊谷 敏幸
NECエレクトロニクス
第二マイコン開発事業部
チームマネージャ

田中 健太郎
NECエレクトロニクス
第二マイコン開発事業部
主任