

リアルタイム制御によるシステム LSI 電源電圧スケジューリングの低電力化

佐藤賢和 渡辺重佳
湘南工科大学 情報工学科

1. はじめに

今日、携帯機器の発展により、搭載されるシステム LSI の消費電力の増大化が問題となっている。システム LSI の設計の段階において、従来の回路等のハードウェアのみにおいて低電力化の設計手法を取り入れてきたが、近年では、ソフトウェアを用いた低電力化の設計が考えられている。今回、リアルタイム OS を用いたスケジューリング手法を用いて、多電源方式[1]・ダイナミック電源方式[1]、FV 制御方式[2]の電源方式を用いてどの程度の低消費効果が得られるのかについて検討をした。

2. 各種電源方式

今回、検証にあたり各種の電源方式を用いた。まず、従来の電源方式については、電圧値は一定であり変化することはないものとし、多電源方式については、高い電圧と低い電圧の一定の 2 値をもつ 2 電源方式を用いる。そして、ダイナミック電源方式は、処理量に応じて動的に電圧値が変化するものであり今回の研究では、高い電圧と低い電圧の 2 つのみを動的に変化させるものとする。FV 制御方式は処理量に応じて動的に、電圧値のみ変化するものと、動作周波数と電圧値の両方が変化するものを併用して低電力効果を得るものである。

3. リアルタイムシステム

リアルタイムシステムとは、図 1 に示すように、イベント発生ごとに対応するタスクをある定められた時間内に処理を完了していなければいけないシステムである。[3]

図 1 での **release** は、イベント発生から対応するタスクが実行可能になり、**start** から処理が開始し、**final** で処理が完了する。この **final** が **deadline** を超えないように制御をするのがリアルタイムシステムの特徴である。図 1 の WCET は処理の最悪実行時間

Real time supply voltage scheduling architecture for low-power system LSI

Yoshikazu Sato, Shigeyoshi Watanabe

Department of Information science, Shonan Institute of Technology

であり、最長でもその時間に処理が終了する見込みである。本論文では実処理を本来の実行時間とせず、図中の予測最長実行時間を実行時間、WCET までを実行時間と置き換えて今後述べる。

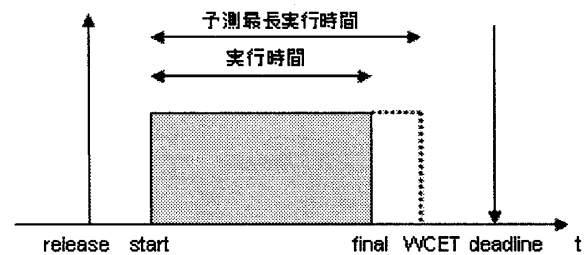


図 1. リアルタイムシステム

4. RM スケジューリング

リアルタイムシステムには、複数タスクをどのように処理をするのかを決める仕組みがある。この RM スケジューリングでは、タスクの周期を短い順番に優先度を高く付けプリエンティブなスケジューリングを行う。下図がタスク 3 つでの例である。なお、それぞれタスクは、 $\tau_i = \{T_i, C_i, D_i\}$ で表し、 T を周期、 C を WCET、 D をデッドラインとする。図 2 の場合、 $\tau_1 = \{9, 4, 9\}$ 、 $\tau_2 = \{18, 3, 18\}$ 、 $\tau_3 = \{36, 2, 16\}$ である。なお、周期の短い順に優先度が高くなるのでこの例の場合、タスクの優先度の関係は $[\tau_1 > \tau_2 > \tau_3]$ が成り立つ。

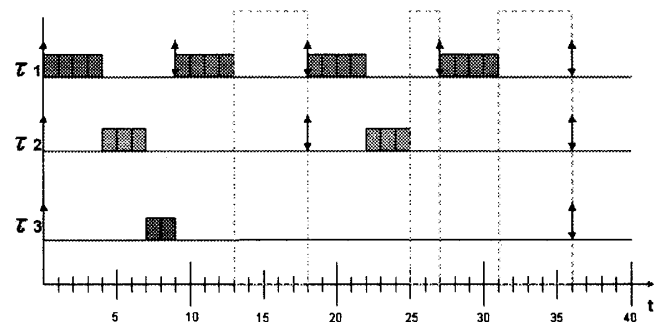


図 2. RM スケジューリング

図 2 で、点線で囲まれた部分はプロセッサアイドル区間で、この間プロセッサはどのタスクも実行していない。

5. 低電力化手法と考察

各電源方式と、RM スケジューリングで、低電力化を行うには、プロセッサアイドル区間を利用し、電源電圧、又は動作周波数、あるいは両方を低くすることで、処理をデッドラインを超えないように引き伸ばしすることで、低電力化が可能となる。この場合、デッドラインを超えないようにするには、いくつかの高低の組み合わせが存在する。これから、低電力化を行うために図2を対象とし、まず、図3のように極端に組み合わせの選択を狭めて低電力化を考えてみる。

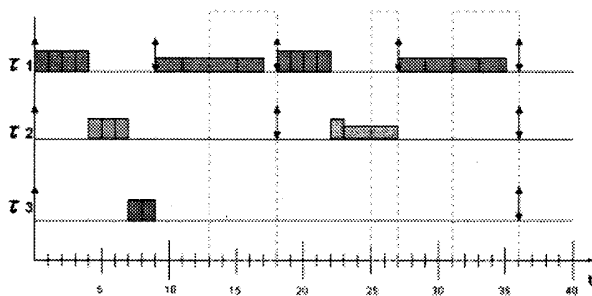


図3. 図2を低電力化を行った例 (1)

図3では、プロセッサアイドル区間の前の処理、つまり変化の組み合わせは単純なものとなり、低電力化の手法としては簡単なものであり、その効果は元の電力の33%減少させられる。

次に、処理内容全体に対してデッドラインを超えないように処理を引き伸ばす、つまり高低の組み合わせの選択を広めて低電力化を考えると、図4のようになる。

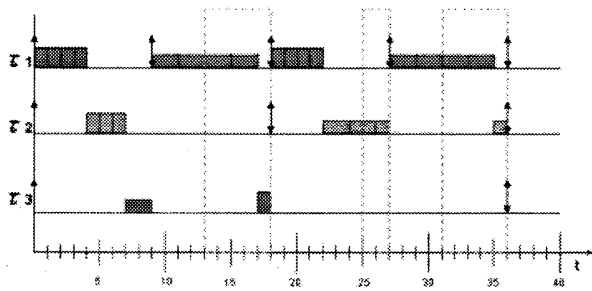


図4. 図2を低電力化を行った例 (2)

図4は、図3と比べて引き伸ばす個数が明らかに増えているのがわかる。この時の低電力効果としては、元の電力の39%も減少し、図3よりも低電力効果を得ていることがわかる。

以上が、多電源方式・ダイナミック電源方式での低電力効果についてであるが、FV制御については、同じように、引き伸ばし処理を行うと、図5のよう

な現象が起きる。このとき、FV制御は動作周波数と電圧値の両方を半分に下げたものとする。

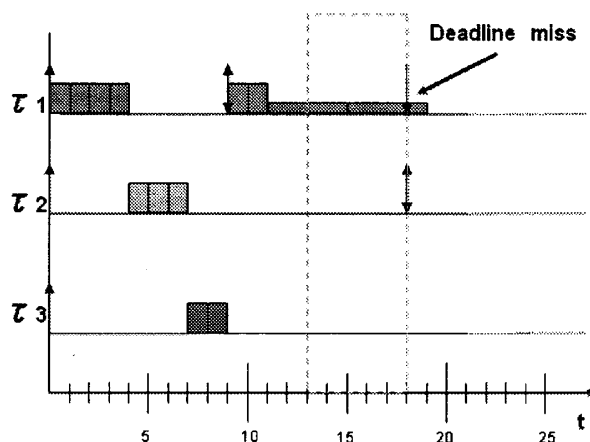


図5. 図2をFV制御で処理

図5のように、デッドラインミスが発生し、システムとしてなりたたない。これを回避するには、電圧値のみの変化と動作周波数と電圧値の両方を併用して、低電力効果を得る必要がある。なお、デッドラインミスをしないように上記方法をとって得た効果は、元の電力の33%減という結果であった。

6. まとめ

以上のように、いずれの方法も低電力効果を得ることができることが証明できた。今回は、多電源方式・ダイナミック電源方式がFV制御方式よりも低電力効果を得ることができたが、タスクの内容しだいでは、FV制御の方が低電力効果を得られる場合もあることが他の検証で得ることができる。今回、多電源方式とダイナミック電源方式は2段階のみの電圧値の変化で検証したが、これが3つ4つと増えると、同じ効果が得られてもチップ面積が大きくなり好ましくないため、そのような場合にはダイナミック電源方式を用いる方法が良いと考えられる。

参考文献

- [1] 花見智、“システムLSIにおける多電源方式とダイナミック電源方式の電力比較” 湘南工科大学情報工学科 2006年度卒業論文
- [2] 川口博、辛英洙、桜井貴康、“75%電力節減可能な離散FV制御機構を有する低電力リアルタイムOS: μ ITRON-LP” 東京大学生産技術研究所、IBM T.J. Watson Research Center
- [3] 白川洋充、竹垣盛一、“リアルタイムシステムとその応用” システム制御情報学会編、朝倉書店