

家電製品向けソフトウェア構造モデルの開発 (1)

5K-5

- スケジューリング方式 -

関口 卓也 内山 亘 岩本 恵子 伊藤 謙次 縄田 毅史 安武 剛一
松下電器産業株式会社

1 はじめに

現在、大部分の家電製品には制御用マイコンが組み込まれている。このマイコンのためのソフトウェア（以下、家電ソフトウェアと呼ぶ）のサイズが製品の高機能化に伴い急速に増大している。また機種展開などで多数の類似ソフトウェアを開発する必要があり、生産性の向上が急務となっている [1]。

そこで我々は、生産性向上のために、家電ソフトウェア設計支援システム Visual CASE [2] を開発している。Visual CASEでは、機能の入れ換えによって再利用を実現するために、家電ソフトウェア設計モデルとして機能関係モデルを提案している [3]。

今回我々は、機能関係モデルに従って、マイコンプログラムの枠組として家電製品向けソフトウェア構造モデル（以下、家電ソフトウェア構造モデルと呼ぶ）を提案する。本稿では、そのスケジューリング方式について報告する。

2 課題

機能関係モデルに従って家電ソフトウェアを開発する場合、入れ換えの単位となる機能を、処理行程の一連の流れ（以下、制御行程と呼ぶ）と考える。制御行程は、処理時間の制限の有無によって、(a) ポートの入出力などのハードウェア制御（以下、回路制御と呼ぶ）と、(b) データ設定などハードウェア以外の制御（以下、シーケンス制御と呼ぶ）に大別できる。(a) では、制御するハードウェアごとに決められた時間間隔で起動し、限られた時間内に処理を終了しなければならない。一方 (b) では、起動の決められた時間間隔や、処理時間の制限がなく、順次に処理を進行させればよい。

すなわち、家電ソフトウェアでは、以下の要件が満たされる必要がある。

1. 決められた時間間隔で回路制御が起動され、制限時間内に処理を終了すること
2. 入れ換えの単位となる制御行程が、モジュールなどの一つの単位で記述できること

回路制御の起動のタイミングは、割り込みを利用して、処理の制限時間は、次の割り込みが発生するまでの間である。この時間制限は、機器を正常に動作させるために、厳守しなければならないため、回路制御は、シーケンス制御よりも優先的に実行されなければならない。

A Software Model for Home Appliances(1) - Scheduling -
Takuya SEKIGUCHI, Wataru UCHIYAMA, Keiko IWAMOTO,
Kenji ITO, Takeshi NAWATA, Kouichi YASUTAKE
Matsushita Electric Industrial Co., Ltd.

このため、シーケンス制御は所定の回路制御が終了した後、次の割り込みが発生するまでの時間内に実行可能な処理単位にシーケンス制御を分断せざるを得ない（図1参照）。図1の例では、シーケンス制御が4つに分断されている。分断される単位は、入れ換えの単位と一致するとは限らないため、シーケンス制御を一つの単位として記述することが困難である。

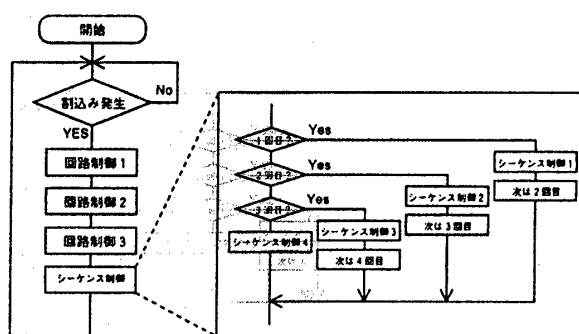


図1: 制御行程が分断されるソフトウェア構造

3 並列処理による家電ソフトウェア構造

前節で述べた2つの要件を実現するために、以下の3つから構成される家電ソフトウェア構造モデルを提案する（図2参照）。

- シーケンス制御部
- 回路制御部
- スケジューラ

スケジューラは割り込みを監視し、シーケンス制御部と回路制御部の処理を切替える。これにより、回路制御が定期的かつ制限時間内に実行される。つまり、スケジューラは、シーケンス制御部が実行中であっても、割り込みの発生を検知すると回路制御部を起動する。そして、回路制御部における所定の処理を終了した後、再びシーケンス制御部を継続して実行する。

上記のソフトウェア構造を枠組として、家電ソフトウェアの開発を行なうと、前節の要件は以下のように実現できる。

1. スケジューラが割り込みを監視し、シーケンス制御部と回路制御部の処理を切り換えることにより、回路制御を定期的に行うことができる。

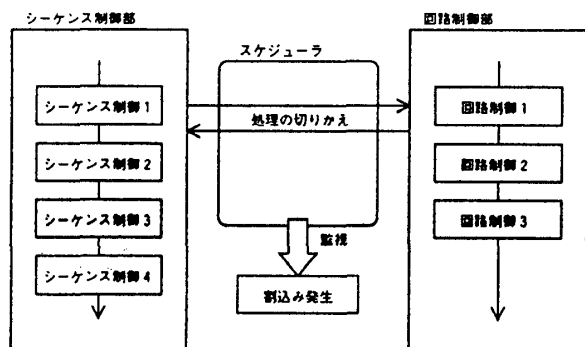


図 2: 並列処理によるソフトウェア構造

2. シーケンス制御部と回路制御部を分離したことにより、シーケンス制御部を分断することなく記述できる。

これにより、回路制御部の処理を保証し、かつシーケンス制御の入れ換えが可能になる。

4 スケジューラ

前節で提案した家電ソフトウェア構造の構成要素の一つであるスケジューラの概要を述べる。

一般に、家電製品に搭載されるマイコンは、以下の特徴を持つ。

- 実行速度の遅いものが多い
- ROM および RAM サイズなどのメモリ資源の少ないものが多い

このような特徴を持つマイコンのプログラムにスケジューラを導入する場合の要件として、以下の2点が挙げられる。

- スケジューリングのオーバーヘッドが小さいこと
スケジューリングのオーバーヘッドが大きくなると、回路制御部における処理が、限られた時間内で終了しない可能性がある。よって、スケジューラのオーバーヘッドはできるだけ小さい方が望ましい。
- 占有するメモリ資源を少なくすること
メモリ資源が少ないため、スケジューラのコードサイズや RAM などではできるだけ少ない方が望ましい。

そこで、一つの回路制御を実行している時に、他の回路制御およびシーケンス制御が実行されないことに着目し、以下の特徴を持つスケジューラを設計した。

- 2つのタスクだけを対象にする
家電ソフトウェアの場合、ハードウェア回路の性能上、回路制御の順序が自ずと決定される。そこで、その順序を一連の処理と考え、これを1つのタスクとした。また、シーケンス制御部は、これを1つのタス

クとした。スケジューラは、これら2つのタスクを対象にスケジューリングを行なう。

- 2つのタスクで使用するスタックを共有する
回路制御部からシーケンス制御部へタスクを切り換えるとき、回路制御部のタスクは必ず終了している。つまり、シーケンス制御部を使用するスタックが変化するとき、回路制御部はスタックを全く使用していない。よって、使用するスタックを共有できる。

これにより、以下の効果が挙げられる。

- 回路制御部からシーケンス制御部へタスクを切り換える際、回路制御部のタスクは必ず終了していることから、コンテキスト情報の保存が不要となり、コンテキスト切替時間を節約できる
- 各タスクごとにスタックを持つ必要がないため、スタックに使用するメモリ資源を節約できる

5 おわりに

機能関係モデルに従って、マイコンプログラムの枠組として家電ソフトウェア構造モデルを提案し、さらに、その構成要素の1つであるスケジューラの概要について述べた。これにより、家電ソフトウェアのシーケンス制御の再利用性の向上を図った。

今後の課題としては、入れ換えの単位となる制御行程の設計指針についての検討などが挙げられる。

参考文献

- [1] 坂村健ほか. ITRON 標準ガイドブック'92-'93. 社団法人 トロン協会, 1992.
- [2] 角谷和俊, 眞田紀男, 春名修介, 今井良彦. 家電ソフトウェア設計支援システム Visual CASE の開発. 情報処理学会ソフトウェア工学研究会, pp. 9-16. 情報処理学会, March 1993.
- [3] 角谷和俊, 安武剛一, 向井雅樹, 眞田紀男, 田中裕彦, 春名修介, 今井良彦. 家電ソフトウェア設計支援システム-Visual CASE-. 電気関係学会関西支部連合大会講演論文集, p. S47. 電気関係学会他, November 1992.