

組込み技術を用いたファンコントローラ（試作報告）

— μ ITRON と TCP/IP を用いた組込み技術の習得 —

青木 邦知* 若生 進一* 中嶋 勝也*

1. はじめに

1.1 組込み技術の現状

マイコンが登場した当時から、組込み技術は様々な機器の制御に用いられてきた。近年、その技術は高度化が進んでおり、例えば、高級車には100個近くものマイコンが車載機器として搭載されている。さらに今後は、マイコンが単独で制御するのではなく、CAN, LIN, FlexRay 等の車載向けに規格化されたプロトコルによるネットワークを構築することで相互に情報を交換し、協調して制御する方向に進んでいる¹⁾。

また、電子制御化によって増大している制御要求の管理に対しては、各制御のタスク化、割込処理の簡素化を行うことを目的として、組込み用 OS を採用するケースも増えてきている。

このようなことから、今後の組込み機器においてネットワークによる分散制御、組込み用 OS は重要であり、組込み分野において中核をなす技術である。

1.2 センターとしての取り組み

組込み技術は企業における製品の高性能化、差別化を図る上でも非常に有効な技術である。これらの技術を習得、普及することで中小企業への技術支援の一助とする。

今回、組込み用 OS 及びネットワークの実装技術習得のために試作を行なったのでここに報告する。

2. 試作概要

2.1 全体の構成

全体の構成を図1に示す。

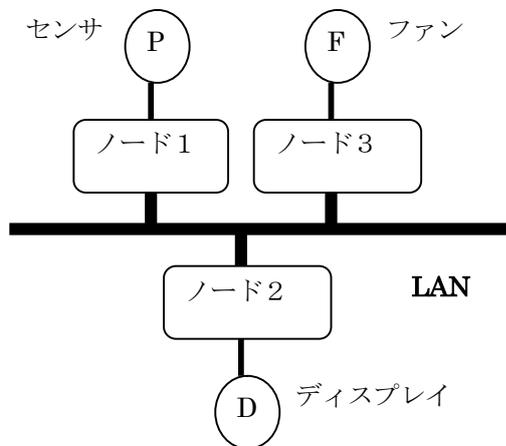


図1 機器構成

センサからの入力によってファンの回転数を制御し、入力値と回転数を表示する機能を3つのノードに分け、ネットワーク上で制御する構成とした。

2.2 動作要件

システムとしての動作要件を表1にまとめる。

表1 システム動作要件

ノード1 (センサ)	<ul style="list-style-type: none"> ・入力値のセンシング (0.3 秒毎) ・通信切替スイッチの監視 (0.3 秒毎) ・LEDの点滅制御 ・データ送信 (0.5 秒毎)
ノード2 (ディスプレイ)	<ul style="list-style-type: none"> ・入力値・回転数データ受信 ・入力値・回転数データ表示 (0.5 秒毎)
ノード3 (ファン)	<ul style="list-style-type: none"> ・入力値データ受信 ・PWM信号デューティ変更 (0.3 秒毎) ・PWM信号出力 ・回転パルスカウンタ出力 (1 秒毎) ・入力値・回転数データ表示 (0.3 秒毎) ・回転数データ送信 (0.5 秒毎)
共通	<ul style="list-style-type: none"> ・ネットワークの即時復帰

共通にあるネットワークの即時復帰とは、物理的な切断があった場合などでも、その状態を認識し、再度接続が復旧した際に遅滞無くネットワークが復帰するための動作要件である。

2.3 ソフトウェア仕様

今回の試作に採用したソフトウェアの仕様を表2に示す。

表2 ソフトウェア仕様

組込み用 OS	JSP-1.4.1
プロトコルスタック	TINET-1.2.6
開発環境	GNU 開発環境 binutils-2.15 gcc-3.4.1 newlib-1.13.0
エミュレータ	Cygwin-1.5.18

組込み用 OS には μ ITRON、組込み Linux などいくつかの種類があるが、仕様が公開されていること、ドキュメント類が豊富であること、オープンソースで手に入ることを重視して、今回の試作には μ ITRON 仕様²⁾ に準拠した TOPPERS/JSP (Just Standard Profile) を採用した。TOPPERS³⁾ (Toyohashi OPen Platform for Embedded Real-time Systems) とは、ITRON 仕様の技術開発成果をオープンソースソフトウェアとして公開することで組込みシステム技術と産業の振興を図ることを目的としたプロジェクトの名称であり、現在は NPO 法人として普及活動等を行っている組織である。

*先端材料部門

ネットワークにはパソコン(以下PC)のLAN等で一般的に用いられている Ethernet とし, プロトコルスタックとして TCP/IP 準拠の TINET を採用した。TINET は TOPPERS プロジェクトにおいて仕様が策定されたプロトコルスタックであり, TOPPERS/JSP への実装が容易であることから, 今回の試作に用いることとした。

また, 組込みソフトウェアの開発にはマイコンと PC とのクロス開発環境が必要となる。商用の開発環境等もあるが, 今回はオープンソースの GNU 開発環境⁴⁾を採用した。GNU 開発環境は Linux ベースの開発環境であり, Windows 環境での開発が行なえないため, Windows 上で Linux 環境を仮想構築するためのエミュレーションソフト(Cygwin⁵⁾)を利用して, C 言語によるソフトウェア開発環境を整えた。

2.4 マイコン仕様

今回の試作に用いたマイコンボードの仕様を表3に示す。

表3 マイコンボード仕様

マイコン	H8/3069
クロック周波数	20MHz
インタフェース	Ethernet (RTL8019AS) RS232C
表示部	LED×2 (赤, 緑) 液晶 (SC1602BSLB)
入力部	DIPSW×4

マイコンボードは, TOPPERS プロジェクトにおいて実装実績のある, 秋月通商製のボードを採用した。

3. 試作及び結果

完成したファンコントローラを写真1に示す。

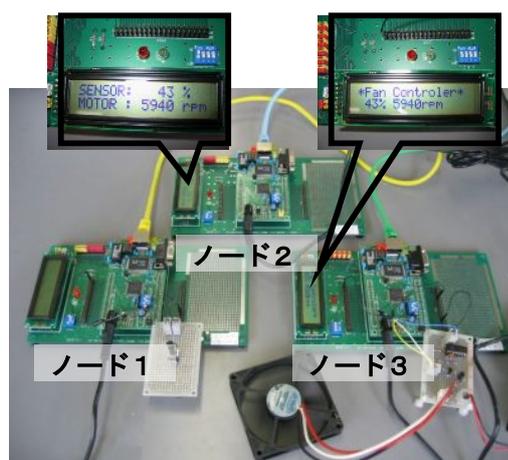


写真1 ファンコントローラ

試作にあたって, 組込み用 OS の有効性を確認するために各ノードの開発に1人ずつ充て, 各ノード単独による動作確認, ネットワークの実装まで完了した後にネットワーク上での動作確認を行った。

結果として, 想定していた動作要件を満たすことが出来た。

今回, 組込み用 OS を使用した利点を挙げると, プログラムをタスク毎に区切り, 優先度を設定できるため, タイムスケジュールの管理が非常に容易で, タスクの追加や変更が簡単にできた点が挙げられる。また, 仕様が明確であるために, 複数の人員が関わる開発においてもプログラムソースの読解や再利用が容易になり, 有効であることが確認できた。

今回使用したプロトコル(TINET)については Windows 等によく用いられる socket 通信に比べて手順が単純で, 各ノードのネットワーク対応も容易だった。また, ドキュメントも豊富で TOPPERS プロジェクトのメーリングリストや, 宮城県産業技術総合センター⁶⁾のホームページも非常に参考になった。

一方, オープンソースのソフトウェアを用いる場合, バージョンアップが頻繁にあるので, バージョンの競合やバグの修正履歴に注意する必要がある。また, オープンソースの組込み用 OS をカスタマイズするのはかなりの知識を必要とするため, 融通の利かない面もある。

4. まとめ

今回の試作を通して以下のような知見が得られた。

- (1) TOPPERS/JSP, TINET の実装技術を習得することができた。
- (2) マイコンに対する組込み用 OS の有効性が確認できた。
- (3) ネットワークによる分散制御を確認することができた。

今後は, 組込み Linux 等, μ ITRON 以外の組込み OS についても調査し, 実装技術等の習得を図る。また, ネットワーク技術についても, Ethernet だけでなく, その他のインタフェースの調査及び技術習得を進め, 組込み機器への応用技術として展開させていく予定である。

5. 参考文献

- 1) 服部博行, 大西秀一. LIN 通信ドライバによる TOPPERS/OSEK での制御の実例. Interface. 31, 4(334), 2005, 78-87
- 2) 坂村健. " μ ITRON4.0 標準ガイドブック ". 初版 東京, パーソナルメディア株式会社, 2004
- 3) TOPPERS プロジェクト
<http://toppers.jp/>
- 4) GNU プロジェクト
<http://www.gnu.org/home.ja.html>
- 5) 小川淳一. "Cygwin 徹底入門". 初版. 東京, 株式会社ソーテック社, 2003
- 6) 宮城県産業技術総合センター
<http://www.mit.pref.miyagi.jp/embedded/>