

状態遷移図ベースのコード生成とログ可視化機能を備えたセンサネットワーク実機検証基盤の開発

辻村篤志[†] 小林侑生[†] 不破泰[†] アサノデービッド[†]

Atsushi TSUJIMURA[†], Yu KOBAYASHI[†], Yasushi FUWA[†], and David ASANO[†]

あらまし センサネットワークの開発は幅広い専門知識と多大な工数を要する。先行研究では、センサネットワークの動作を表す状態遷移図からコードを生成し、汎用ハード上で動作する環境が構築されてきた。本研究ではその環境を拡張し、その上で実用的な無線通信プロトコルを実装・検証した。その過程で汎用ハード上でのプロトコルの動作が確認しづらいことを課題として認識し、デバッグ用ログ出力とそれを可視化・ステップ実行できる環境を開発しデバッグ効率の向上を図った。

キーワード 無線センサーネットワーク、無線通信プロトコル、状態遷移図、コード生成、ログ可視化

1. はじめに

1.1 背景

無線センサーネットワークは、環境モニタリングやスマートシティなど、さまざまな分野での応用が期待されている。しかし、これらのシステムの開発には、複雑な通信プロトコルやデータ処理アルゴリズムの実装が必要であり、専門的な知識と多大な工数を要する。先行研究では、センサネットワークの動作を表す状態遷移図からコードを生成し、汎用ハード上で動作する環境が構築されてきた。

1.2 従来研究

従来研究では、無線通信プロトコルの状態遷移図から自動生成したプログラムを汎用ハード上で動作させ、CSMA および TDMA ベースの MAC プロトコルを実機で実装・検証できる環境が構築されていた(旭ら)。このシステムにより基本的な通信プロトコルの動作検証が可能となったが、デバッグ方法はシリアルコンソールへのログ出力に依存しており、通信処理が高速に進むため逐次的な状態遷移を追跡することが難しかった。その結果、異常動作の原因を把握しづらく、開発効率や教育的利用の観点では十分でない点が課題として残されていた。

1.3 目的

本研究の目的は、従来研究で課題となっていたデバッグ効率の低さを改善し、実機でのプロトコル検証をより効果的に行える環境を実現することである。そのために、状態遷移図から生成したコードの動作をログとして出力し、それを可視化・ステップ実行できる仕組みを開発した。これにより、プロトコルの動作を逐次的に把握でき、異常動作の原因究明を容易にするとともに、教育や応用実証に適した支援環境を提供することを目指す。

2. 提案システム

2.1 システム構成

本研究で開発したシステムの全体構成を図1に示す。本システムは、状態遷移図を入力として C++ コードを自動生成し、汎用ハードウェア上で実行することでセンサネットワークのプロトコルを検証可能とする。さらに、通信動作の状態遷移の過程をログとして記録し、ブラウザ上で状態遷移の可視化およびステップ実行を行える環境を備えている。これにより、プロトコル設計から実機検証、デバッグまでを一貫して支援することが可能となる。

2.2 ファームウェア生成部

- 無線通信プロトコルの動作を状態遷移図 (Astar) で設計し、xml 形式で出力- 変換プログラム (Python) を用いて、出力された xml を C++ プログラムに変換-

[†] 信州大学, 長野県
Shinshu University, 4-17-1 Wakasato, Nagano-shi, Nagano 380-8553 Japan
DOI:10.14923/transfunj.??????????

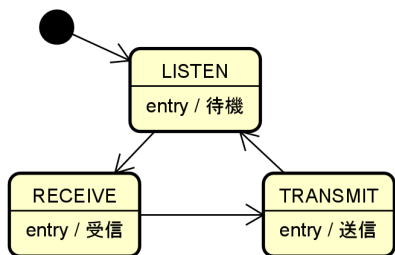


図1 状態遷移図の例
Fig. 1 State transition diagram.



図2 どこでモデム(左)とGPS モジュール(右)
Fig. 2 Block diagram of the proposed system.

- 2.3 実行環境 (ハードウェア)
- 2.4 デバッグシステム部
- 3. プロトコル実装と評価
- 4. 考 察
- 5. まとめ、今後の展望

謝辞

文 献

[1]

付 録

1.

(xxxx 年 xx 月 xx 日受付)



PlatformIO 環境でビルドし、ファームウェアを生成

まず、ファームウェア生成部では、無線通信プロトコルの動作を状態遷移図 (Astah) 上で設計し、XML 形式で出力する。次に、変換プログラム (Python) を用いて XML を C++ プログラムへ変換し、PlatformIO 環境でビルドすることでファームウェアを生成する。

```
typedef enum { LISTEN, RECEIVE, TRANSMIT } NodeState;
```

```
void protocol_main() {  
    switch (state) {  
        case LISTEN:    updateState(RECEIVE);    break;  
        case RECEIVE:   updateState(TRANSMIT);   break;  
        case TRANSMIT:  updateState(LISTEN);     break;  
    }  
}
```

Abstract The development of sensor networks requires extensive expertise and significant effort. Previous research has generated code from state transition diagrams representing the behavior of sensor networks, establishing an environment that operates on general-purpose hardware. This study expands that environment and implements and verifies practical wireless communication protocols. During this process, we recognized the difficulty in confirming the operation of protocols on general-purpose hardware as a challenge and developed an environment for debugging log output and visualizing and step-executing it to improve debugging efficiency.

Key words wireless sensor networks, wireless communication protocols, state transition diagrams, code generation, log visualization