

福祉機器搭載型ネットワーク構築技術の開発

中島 康博, 橋場 参生, 高橋 裕之

Development of Network System for Assistive Mobiles

Yasuhiro NAKAJIMA, Mitsuo HASHIBA, Hiroyuki TAKAHASHI

抄 録

重度障がい者は身体能力に限られるため、電動車いすやベッド等、自身の近辺にいつでも使用できるように携帯電話や会話補助装置、リモコンなど様々な補助デバイスを設置するケースが多い。しかし、これらのデバイスはたいてい各々独立にボタンやジョイスティックなどの入力装置を持っているため、身体能力に限られるユーザでは操作しきれず、使用を断念するという矛盾を生じる場合がある。また個々の装置についても、ユーザの身体能力に合わせて入力デバイスを特殊仕様にする場合があり、各メーカーで特殊仕様の入力デバイスの開発や工房レベルでデバイス改造等を行う等の対応をしているが、メーカー間の互換性はないのでPCのマウスのように好きなメーカーのものを選択するといったことは不可能である。このような問題を解決し、入出力装置の互換性を確保しつつインターフェースを統合すれば、障がい者の機器使用の可能性を広げると共に利便性の向上を図れ、かつメーカー間のアセンブリが実現することで開発コスト低減にも結びつく。

本研究では、福祉機器用通信プロトコルとして開発されたM3S(Multiple Master Multiple Slave intelligent interface for the rehabilitation environment)に着目し、同規格に準拠したM3Sネットワークボードと同ボード用APIを開発した。これらを使用してM3S入力デバイス・システム制御デバイス・車いす制御デバイス・PC制御デバイスを開発した。開発したデバイスを使用してM3Sバスに接続しシステム動作試験を行った結果、各デバイスが協調してM3Sシステムとして動作することを確認した。

キーワード：電動車いす, 通信プロトコル, CAN, M3S

Abstract

There are many cases which users of electric wheelchairs install various support devices in their wheelchairs, such as portable telephone and conversation auxiliary equipment, remote control, etc. But there is a case, which they abandon the use, since it respectively have input unit such as button and joystick, without manipulating in the users who the body ability is limited. And there are another cases, which input devices having special specification are also made for the users by device manufacturers. But it is impossible the fact that the thing of the manufacturer of which it likes the compatibility between manufacturers like a mouse of PC. If the compatibility of the input-output unit is ensured, it is convenient for the users to choose the better one. And it is also convenient for the manufacturers, because it will reduce the cost of device development.

In this study, we pay attention to M3S (Multiple Master Multiple Slave intelligent interface for the rehabilitation environment), developed as communication protocol for the welfare equipment. We developed M3S network board and the API for the board which follow M3S. And we also developed using them: M3S input device, system control device, wheelchair control device and PC control device. It was confirmed that it operated as a M3S system by each device cooperating as the result that it is connected with the M3S bus using developed device and carried out the system performance test.

KEY-WORDS : electric wheelchairs, controller area network, communication protocol, M3S

1. はじめに

重度障がい者は身体能力が限られるため、電動車いすやベッド等、自身の近辺にいつでも使用できるように携帯電話や会話補助装置、リモコンなど様々な補助デバイスを設置するケースが多い。しかし、これらのデバイスはたいてい各々独立にボタンやジョイスティックなどの入力装置を持っているため、身体能力の限られるユーザでは操作しきれず、使用を断念するという矛盾を生じる場合がある。また個々の装置についても、ユーザの身体能力に合わせて入力デバイスを特殊仕様にする場合があり、各メーカーで特殊仕様の入力デバイスの開発や工房レベルでデバイス改造等を行う等の対応をしているが、メーカー間の互換性はないのでPCのマウスのように好きなメーカーのものを選択するといったことは不可能である。このような問題を解決し、入出力装置の互換性を確保しつつインターフェースを統合すれば、障がい者の機器使用の可能性を広げると共に利便性の向上を図れ、かつメーカー間のアセンブリが実現することで開発コスト低減にも結びつく。

本研究では、上記問題を解決する手段として福祉機器用通信プロトコルとして開発されたM3S(Multiple Master Multiple Slave intelligent interface for the rehabilitation environment)に着目し、同規格を実装したM3Sネットワークボードを試作開発した。また、このボードをベースとした入力デバイスを数点開発し、電動車いすに搭載して操作テストを行ったので報告する。

2. M3Sプロトコルの概要

電動車いすに限ったことではないが、初期の機器や各種制御装置は入力装置と出力装置が直結されたもので、操作デバイスとモータとは一体不可分のものであった(図1上)。システムの構造はシンプルだが、制御や設定の自由度は非常に制限されたものであり、ユーザの身体状況に合わせて様々な条件を設定しなければならない電動車いすには制約が大きいものであった。その後マイクロコンピュータの登場により、入力装置の信号は一つのチップのI/Oを介して取り込まれ、様々な制御演算後に駆動装置に出力できるようになった。この技術革新により入力と出力は分離され、細かい設定や制御条件をプログラムできるなど自由度は大幅に拡大した(図1中)。

しかし、このシステムもいくつかの問題を抱えている。基本的にマイクロコンピュータのピン一つに入出力一つが対応するため、表示器など制御する装置が増えるに従い、制御装置本体から出る配線は放射状に増えていく。また、当然ながらコンピュータのI/O数には制限があり、これらの要因によりオプション装置の拡張性は制限される。また、これらの装置はそれぞれメーカー固有の方式で接続されるため、互換性がないことも問題である。

M3Sは、このような問題点を解決するために開発されたプロトコルで、省配線化とシステムの柔軟性・拡張性の確保、互換性の確保を求め、一つのCPUによる中央集権的なシステムから、複数のモジュールと省配線型のネットワークによる処理形態を指向したシステムとなっている(図1下)。

M3SはEC(現EU)のTIDE-Projectで開発された電動車いす等福祉機器用通信プロトコルで、ver.1.0は1992年にリリースされ、現在はver.2.0(1997年)である。つい先年までISO化の動きもあったが最終的にはリジェクトされた経緯がある。ISOでは現在、通信プロトコルの標準化のためのWGを再度立ち上げる動きがあるが、対象はM3Sをベースとした規格となると言われている。

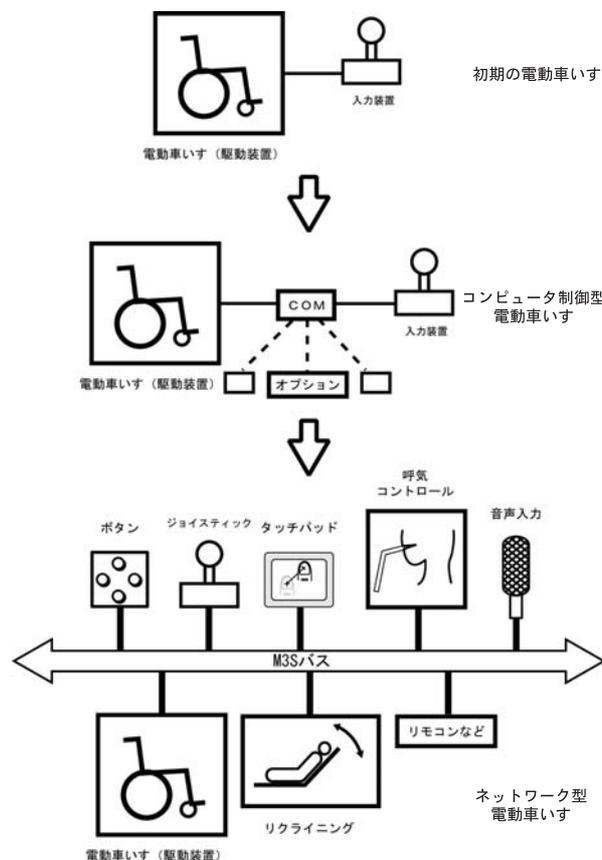


図1 電動車いすデバイスの接続形態の変遷

2.1 CANをベースとするM3Sバス

ネットワーク化したシステムの最大の弱点は通信部分にある。ケーブルは断線、接触不良、ノイズなどの危険にさらされており、ここにエラーが発生して誤動作が起きれば、ユーザにとって致命的な事故が起きる。このような事態が起きないように通信システムには最大級の安全性を確保する必要があるが、現在普及している通信規格で最もそれを満たすものが標準車載LANとして知られているCAN(Controller Area Network)である。M3Sは、このCANをベースとしていて開発された。

CANは通信の安全性を最大限考慮したバスとして開発された。CANは、5種類のエラー検出機能を持ち、検出後の動作が明確化されているなど頑強で安全性が高く自動車の制御系や生産ラインのネットワークにも採用されている。また、CANはハードウェア上でエラー処理を行うため制御コンピュータにかかる通信負荷が非常に小さいことが特徴である。

M3Sバスは、上記のCANにシステム起動用のKeyライン、安全装置用のDMS(Dead Man Switch)ライン、+24 V電源ラインを付加した構成となっている。このバスで駆動装置、入力装置、制御装置等のM3Sバスを持つ装置(以下、M3Sデバイスと呼ぶ)を接続し、バス上での通信によりシステム全体を制御する。

システム制御データは、CANを通じて各モジュールが送受信する。CANの送受信メッセージにはアイデンティファイヤと呼ばれる優先順付きの11bitデータ識別符号が付加されている。M3Sプロトコルはこれらアイデンティファイヤに独自の規定を作成しており、通信仕様の共通化を図っている。

2.2 デバイスの機能表現

M3S デバイスは、各々のデバイスが持つ機能をParameter, DOF, Event, Restriction, Function, Exceptionの6つのオブジェクトで表現する。これらオブジェクトのデータをM3Sバス通信でやりとりすることで、システムは各デバイスを制御する。上記オブジェクトは、デバイス開発時にその固有の機能に合わせて任意に作成できる。

Parameterオブジェクトは、システムに公開されたデバイスの設定値である。例えば、入力装置の感度やモータの最高速度などがある。また、物理スイッチと外部デバイスのFunctionオブジェクト(後述)との対応関係などもここで設定する。デバイスは、Parameterのデフォルト値を保持するのみで、変更値をそのデバイスの不揮発性メモリには保持しない。Parameterの変更値はCCM(Control and Configuration Module)と呼ばれるシステム管理機能を持つデバイスが保持し、必要に応じてParameterオブジェクトにアクセスして値を変更する。

DOFオブジェクトは、ジョイスティックの座標値やインジケータなど、デバイス内で随時可変な値をやりとりする一種のポートとして機能する。基本的には、ジョイスティック、キーボード、インジケータなどの値を出力したり、モータなどとリンクして速度指令値を受信するといったことに使用するが、データ転送用のポートとしても使用される。各入出力DOFはCCMにより仮想的に接続され(通信用に特定のCANアイデンティファイヤを割り振ることで実現する)、その後はCCMを介さずデバイス間でダイレクトに通信する。

Eventオブジェクトは、デバイスで発生したイベントをCCMに通知する機能を持つ。イベントは、主に安全性に関

するもので、例えば「速度超過」や「マニピュレータ未収納」などがある。CCMはこれを受けてRestrictionオブジェクト(後述)を通じて各デバイスの制限状態を変更できる。

Restrictionオブジェクトは、CCMからの指令により各デバイスの制限状態を変更する機能を持つ。例えば、マニピュレータ未収納イベントを受け取った場合に最大速度を1/2に制限する等がある。

Exceptionオブジェクトは、デバイスが発生する例外(バッテリー電圧低下など)を通知する機能を持つ。Eventとは異なり、システムがデバイスの状態を変更することはないが、発生した情報はFeedbackデバイス(ユーザに車いすの状況を音・画面などで知らせる機能を持つデバイス)を通じてユーザに通知できる。

Functionオブジェクトは、他のデバイスからアクセス可能なデバイスの機能である。DOFとは異なり、設定可能な値はONかOFFのみである。例えば、ランプやウィンカーの点灯、リフトの上下スイッチなどがある。

デバイスはこれら6つの機能表現オブジェクトを持つが、Modeオブジェクトというデバイスの振る舞いを記述するオブジェクトがこれを統括する。Modeオブジェクトは、デバイスの機能表現オブジェクトから必要なものをピックアップし、外部から見たデバイスの振る舞いを記述する。また、デバイスはModeオブジェクトを複数持つことができる。つまり、一つのデバイスが、複数の振る舞いを持つことが可能である(図2)。例えば、あるデバイスが赤外線通信機能を持つ場合に、Mode 0は環境制御装置へのアクセスモード、Mode 1はPCへのアクセスモード、といった記述が可能である。デバイス間の通信は、CCMがこれらModeオブジェクトを選択してデバイスの振る舞いを決め、その中の機能表現オブジェクトにアクセスしてデバイスを制御する。

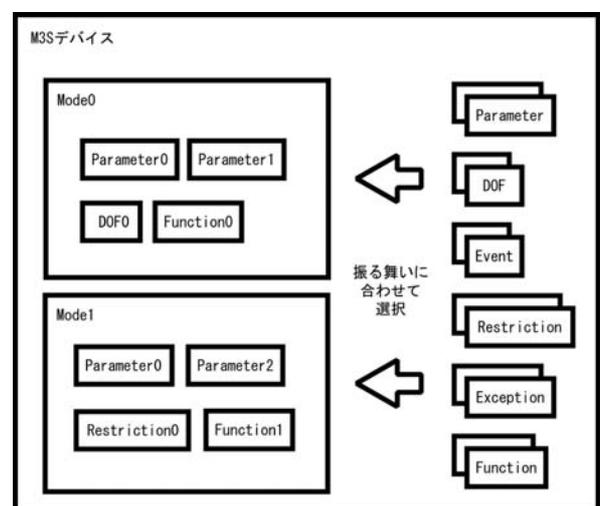


図2 M3Sデバイスの機能表現

2.3 システムの表現

M3Sシステムでは、CCMが実際のシステム管理を行う。CCMは、Taskと呼ばれるシステム動作を記述するオブジェクトを持っており、その中に各デバイスのMode選択、パラメータ設定、DOF接続情報、Event発生時の動作などが記述されている(図3)。また、Taskは(許可されていれば)ユーザの状況に合わせてユーザ自身が生成できる。CCMは複数のTaskを持つことが可能であり、ユーザが任意のTaskを選択してシステムを制御できる。例えば、Task0ではジョイスティックから電動車いすを操作し、Task1では同じくジョイスティックからPCをリモコン操作するといった具合である。また、Taskごとにパラメータ設定を保持し、高速動作モードや低速動作モードを切り替えるといったことにも使用できる。

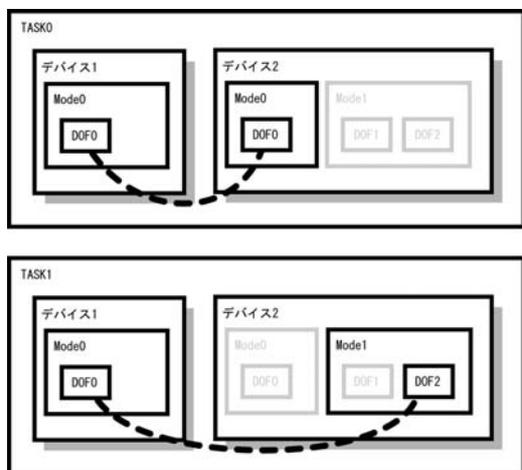


図3 CCM上でのタスク記述

2.4 デバイスのアービトレーションとPnP機能

TCP/IPでは事前にIPを割り当て、それを基に通信を行う必要があるが、M3SではSNAP(Serial Number Arbitration Protocol)という機能により、デバイス番号を動的に割り当てることができる。この機能により、デバイスのPnP(プラグアンドプレイ)も、一定の条件下で実現可能である。同様に、デバイスの自動アップデート(新しいデバイスへの換装)も可能である。

3. M3Sプロトコルの実装

以上に示したM3Sプロトコルに準拠してM3Sネットワークボードと同ボード用のAPIを開発し、同プロトコルを実装した。これらを利用して、M3Sデバイスを数点开発した。

3.1 M3Sネットワークボードの開発

M3Sデバイスを開発するため、M3Sネットワークボードを開発した(図4)。同ボードのベースには、オンチップで

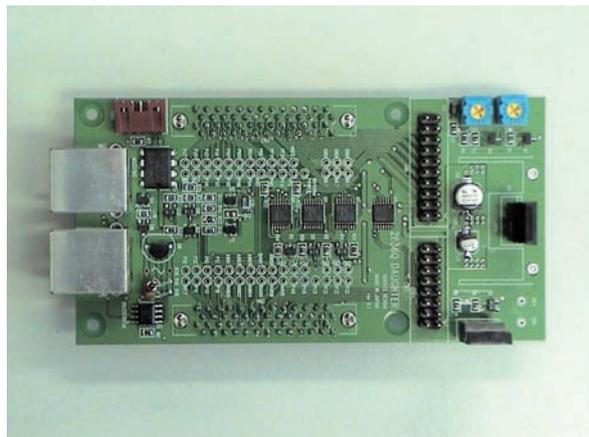


図4 M3Sネットワークボード

CANプロトコルドライバを有するルネサスH8S/2638Fを搭載したCAN評価用ボードHSB8S2638Q(北斗電子)を使用し、これにM3S仕様に準拠したドータボードを搭載する形で実装した。ドータボードには、M3SプロトコルのISO/OSIモデルの物理レイヤ、データリンクレイヤに相当する部分を実装した。また、同ボードには各パラメータの格納用にEEPROM32kBと、SG12232グラフィック液晶表示用ポートを搭載した。仕様を表1に示す。

表1 M3Sネットワークボードの仕様

ベース	HSB8S2638Q (北斗電子)
MPU	H8S/2638F (ルネサス)
クロック	20MHz
ROM	256kB
RAM	16kB
EEPROM	32kB
M3S	1ポート (CAN, Key, Power)
A/D	8ch
D/A	2ch
その他	液晶表示器用ポート スイッチ接続用ポート

3.2 ネットワークボード用APIの開発

M3Sプロトコルのアプリケーションレイヤについて、同ボード用のM3Sデバイス用APIを開発した。M3Sのハード非依存コンポーネントは無償提供されているが、これを同ボード用に最適化した。また、液晶ドライバや車いす駆動用ドライバなど、同ボード用のハード非依存拡張コンポーネント、ハード依存コンポーネント、ハード依存拡張コンポーネントを新たに開発した(図5)。

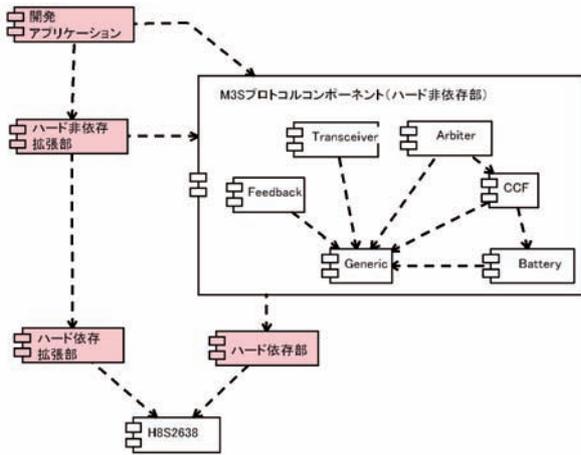


図5 M3SAPIのコンポーネントと依存関係 (着色部分が新規開発)

3.3 M3Sデバイスの開発

同ボード・APIを使用して、入力デバイス・CCMデバイス・車いす制御デバイス・PC制御デバイスを開発した。

車いす制御デバイスは、ベースとなる車いす(ワコー技研Emu)にM3Sボードを接続して仮想M3S化する「M3Sコンバータ」として開発した。同デバイスは電動車いすの入力制御ポートと電源ポートに接続するとともに物理レベルの差異(入出力および電源の電圧レベルやスイッチ制御など)を吸収し、車いす全体を一つのM3Sデバイスと見なしてバスに接続・制御できるようにした(図6)。

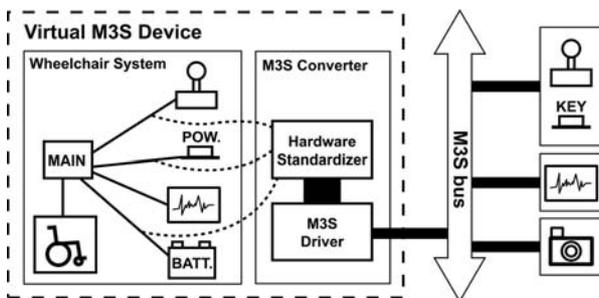


図6 既存電動車いすの仮想M3S化

PC制御デバイスは、IrDA赤外線通信を介してPCのPS/2マウスポートにアクセスする設計とした。IrDAドライバにはMCP2150(Microchip)を使用し、PIC16F873(Microchip)で制御した。IrDA間のマウス制御のプロトコルは独自のものを搭載し、PIC上でプロトコル変換してマウスカーソルを制御した。車いすからは、ジョイスティック及びボタンに類するDOFを持つ入力デバイスにより上下左右の操作とクリックできる構成とした。また、同デバイスは受信側に電源スイッチ用ポートを搭載し、車いす側から電源操作及びハードウェアリセットが可能な設計とした。

4. M3Sネットワークの動作試験

開発したデバイスを使用して、M3Sデバイス及びシステムの動作試験を行った(図7, 8)。ベースの電動車いすには、ワコー技研Emuを使用し、入力デバイス、CCM, 車いす駆動デバイスをM3Sバスで接続した。CCMにはタスクを記述せず、PnPによる動的な接続による動作を試みた。

試験の結果、これらのデバイスはM3Sバスを介して協調して制御を行い、一つのM3Sシステムとして動作することを確認した。

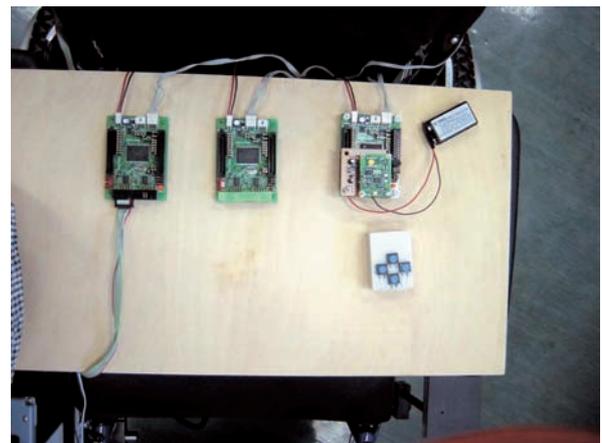


図7 M3Sネットワークボードの接続



図8 M3Sシステムの動作試験

5. まとめ

M3S仕様に基づき、M3Sネットワークボードと同ボード用APIを開発した。これらを使用してM3S入力デバイス・システム制御デバイス・車いす制御デバイス・PC制御デバイスを開発した。開発したデバイスを使用してM3Sバスに接続しシステム動作試験を行った結果、各デバイスが協調してM3Sシステムとして動作することを確認した。

なお、M3S Reference Manual について、M3S dissemination officeのサイトが閉じており、現在入手困難

である。この文献についての情報を知りたい方は、下記宛に連絡を請う。

E-mail : nakajima@hokkaido-iri.go.jp(担当 : 中島)

謝 辞

M3Sネットワークボードの開発には、(株)北斗電子にご尽力いただいた。ここに記して感謝の意を表す。

引用文献

- 1) TIDE-FOCUS M3S specification working-group :
M3S Reference Manual, M3S dissemination office,
(2003)
- 2) CAN in Automation : CAN Specification 2.0, CAN
in Automation, (2002)
- 3) 宇野重雄 : CANの基礎知識, トランジスタ技術, 40(3),
pp.181-192, CQ出版社, (2003)