

# CAN 通信による自動車内データ収集システム

—モータ回転数の測定と通信の調停およびフィルタ機能の確認—

Design and Trial Manufacture of a CAN-based prototype Vehicular Data Acquisition System

小澤 倫也\*

金原 正幸\*

袴田 吉朗\*\*

Michinari OZAWA

Masayuki KINPARA

Yoshiro HAKAMATA

**Abstract:** The paper describes the design and manufacture of a CAN-based prototype vehicular data acquisition system. It consists of PIC micro computers, CAN controllers, CAN transceivers and transmission lines, resulting in several transmitting and receiving nodes. Rotation numbers of a DC motor were measured and transmitted. Arbitration and filter functions were confirmed to work well. The design and manufactured results of the system is given precisely.

## 1. はじめに

電気電子工学科ではH19年度から学科の象徴物として電気自動車を取り上げ、電気自動車に関するプロジェクトを立ち上げて検討を行ってきた。本論文はその電気自動車プロジェクトの一環として、4年生の卒業研究におけるテーマとして検討してきたCAN通信による自動車内データ収集システムに関する検討結果をまとめたものである。一昨年度までは市販のCAN通信ボードを用いて検討を行ってきた。またCAN通信に関する理解をさらに深めることを目指して、昨年度にはPICマイコン、CANコントローラ、CANトランシーバ、LCD（液晶ディスプレイ）などを用いてCAN通信ノードを自作し、これらを用いたポイント・ツー・ポイントによるCAN通信システムの検討を行った[1]。その結果送信ノードにおいて測定した電圧を、CAN通信を用いて正しく送受信できることを確認した。さらにモータの回転数については、モータから生成される回転数測定用パルスをPICマイコンのキャプチャ機能を用いて測定し、CAN通信を用いて受信ノードに伝送できることを確認した。しかしその測定精度に検討の余地のあることが分かった。

そこで今年度は、昨年度に製作したノードに加えて新たに複数のCAN通信ノードを製作し、これらを用いて複数ノードからなるCAN通信システムを構成して検討を進めることにした。検討した内容を以下に示す。

- ・ モータ回転数をより精度高く測定する方法とCAN通信によるデータ伝送方法の検討
- ・ 受信データを振り分けるフィルタ・マスク機能の実装方法の確認
- ・ 複数ノードよりなるCAN通信システムの構築とCAN通信において発生する調停（アービトレーション）の実証確認の実施
- ・ 以上を実現するためのPICプログラムの作成と動作確認

本論文では、以上の検討結果についてその詳細を述べる。

## 2. CAN 通信の概要

### 2.1 伝送路フォーマット

CANバスの信号は、CAN-HおよびCAN-Lと呼ばれる2本の信号線により伝送される。伝送されるデータは、両信号間に電位差があるか、ないかにより決まり

- 電位差のある状態はドミナント (Dominant)・・・データの0
- 電位差のない状態はリセッシブ (Recessive)・・・データの1と呼ばれる。

CANのデータ・フレーム（標準ID）を図2.1.1に示す。またフレームの概要を以下に示す。

- (1) SOF (Start of Frame), 1ビット, ドミナント
- (2) アービトレーション・フィールド (12ビット)  
このフィールドは11ビットのメッセージIDと最後尾1ビットのRTRフィールドからなる。ドミナントのRTRがデータ・フレームを表す。
- (3) 制御フィールド (6ビット)  
標準データフォーマットは最初のIDEビット (1ビット) がドミナントであることにより識別される。DLCフィールド (4ビット) は0~8バイトのデータ長を表す。
- (4) データ・フィールド (最大8バイト)
- (5) CRCフィールド (16ビット)  
CRCフィールドは、15ビットのCRCシーケンス (生成多項式  $x^{15}+x^{14}+x^{10}+x^8+x^7+x^4+x^3+1$ ) と、1ビットのCRCデリミタからなる。
- (6) アクノリッジ・フィールド (2ビット)  
送信ノードは最初のACKスロットにリセッシブを送信する。エラーのないフレームを受信したノードは、このACKスロットにドミナントを入れて返送する。これにより送信ノードは、フレームが正しく受信されたことを認識する。後続するACKデリミタはリセッシブである。
- (7) EOF (End of Frame), 7ビット, リセッシブ
- (8) IFS (Interframe Space), 3ビット, リセッシブ  
他にリモート・フレーム (RTRがリセッシブ), エラー・フレームおよびオーバーロード・フレームがある。  
なお、伝送路符号のフォーマットはNRZ (Non Return to Zero)

2012年3月9日受理

\*理工学部 電気電子工学科4年生

\*\*理工学部 電気電子工学科

フォーマットである。このため受信部におけるPLLがタイミング消失しないようにするために、同一信号レベルが5ビット以上連続すると、その5ビット後にレベルを反転したスタフビット(SB)を強制的に1ビット挿入し、レベルの変化を起こさせるビット・スタフingが行われる。したがって実際のフレームにおけるビットの並びは、送信されるデータの内容に依存して変化するので以上に述べたフレーム構成とは微妙に異なってくる。図2.1.1においてSBと示してあるビットがスタフビットである。

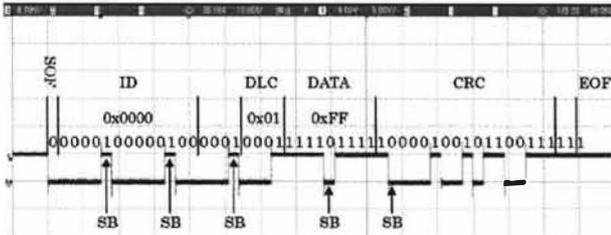


図 2.1.1 MCP2515 出力の測定例 [1]

## 2.2 CANにおける伝送路アクセス制御

CANは分散システムでありCSMA/CA方式によって伝送路にアクセスする。具体的には、CSMA方式+非破壊のビット単位アービトラション(NBA方式: Non-Destructive Bitwise Arbitration)によって信号衝突を防いでいる。

以下のような手順で伝送路にアクセスする。

- ① バスのキャリアをセンスし、バスが空いていれば、バスにつながっている全てのノードが送信を開始することができる。
- ② 複数ノードが同時に送信を開始した場合の信号衝突を回避するために、各ノードはアービトラション・フィールドを送出する期間、バスのレベルをモニタする。
- ③ ドミナントは文字通りリセツpを上書きするので、自ノードが送信した信号と検出したバスのレベルが異なる場合には調停負けと判断し、受信動作に移るようになっている。
- ④ したがってID値が小さい方が高優先度になる。
- ⑤ CANでは、送信される各メッセージが、メッセージIDによって一意に識別される。その結果、メッセージは基本的にどのノードにおいても受信でき、メッセージを受け付けるかどうかはコントローラのみによって決定される(受信側選択システム)。例えば電気自動車の場合には、モータの回転数をデータ収集ノードにおいて収集し、一方で同時に同じデータをメータに表示するような使い方ができることになる。

## 3. 試作した自動車内データ収集システムの概要

試作したシステムの構成を図3.1に示す。4個の送信ノードと1ないし2個の受信ノードおよびCAN伝送路によって構成している机上モデルである。送信ノードでは、PICのAD変換機能を使用して測定したモータの制御電圧に、必要な演算を行ってモータの回転数に換算したデータをCANバスに送信している。

受信部では、CANバスに伝送されたデータをフィルタ機能により振り分けて特定のノードから受信したデータのみをLCD

に表示させている。またPIC内のEEPROMにデータを書き込んで、送受信データを数値として確認できるようにした。

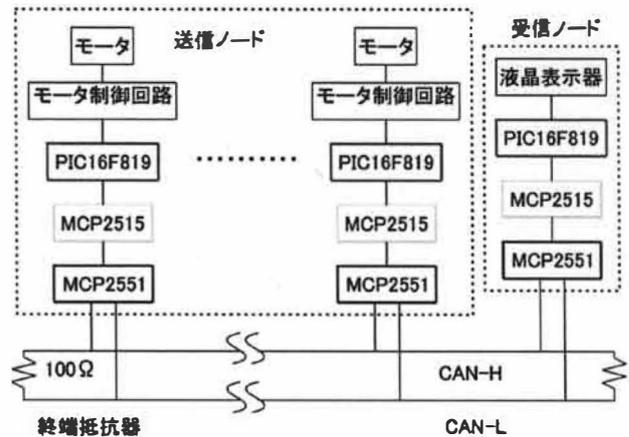


図3.1 試作したシステムのブロック構成

システムにおける構成要素の概要を以下に示す。

- ・ PIC16F819・・・18ピンのPICであるが、AD変換機能、SPIインタフェースを有するためこのPICマイコンを採用した。測定データをMCP2515の送信レジスタに書き込む、受信データの読み取りを行う、LCDに表示する、などの各ノードの制御を行う。
- ・ MCP2515・・・CANコントローラ。送信データをCANのメッセージ形式に変換してMCP2551に送る。また、ビット・スタフingを行って受信部におけるPLLの動作の確実性を高める。受信部では、受信データをMCP2515自身が持つレジスタに格納する、エラーチェックを行うなどの処理を司る。
- ・ MCP2551・・・CANトランシーバ。MCP2515から送信されるデータを差動信号に変換してCAN伝送路に送出する。これにより高いノイズ耐性を得る。受信部では、CANバスの信号をCANのメッセージ形式に戻し、MCP2515に転送する。
- ・ CAN伝送路・・・約20cm長の2本のビニール線であり、両端が100Ωの抵抗により終端されている。

## 4. 試作システムのハードウェア構成

### 4.1 PICマイコンにおけるポートの設定

PICマイコンには、前述したようにSPIインタフェースおよびAD変換機能を有する18ピンのPIC16F819を使用している。

送信ノードにおけるPIC16F819では電圧入力(RA0)、モータ関連の信号(RA2, RA3, RA4)、SPI関連の信号(RB1, RB2, RB4, RB5)を入出力している。受信ノードではLCD関連の信号(RA0~RA3, RB0, RB3, RB6)、SPI関連の信号(RB1, RB2, RB4, RB5)を入出力している。

### 4.2 PICマイコンにおけるAD変換の設定

PIC16F819のAD変換機能を使用するために、レジスタADCON0およびADCON1を以下のように設定した。

ADCON0 = 0xC0

ADCON1 = 0x8E

内容は以下の通りである。

- ・ AD 変換のクロック・・・内部 RC 発振による
- ・ AD 変換の結果を右詰め  
AD 変換の結果は 10 ビット長で出力され、レジス ADRESH および ADRESL の各 1 バイトのレジスタに格納される。格納方法を右詰めとし、10 ビットを結果として使用している。10 ビット使用の場合、変換結果は 0~1023 となる。
- ・アナログ入力ポートを RA0 (AN0) に設定、電圧の基準  $V_{REF+}$  を電源電圧  $V_{DD}$  (5V) に設定  
なお、ADCON0 レジスタにおけるビット 0 は、AD 変換の開始指令であり、AD 変換を開始する度に 1 に設定する。

### 4.3 MCP2515 のピン配置

MCP2515 のピン配置を図 4.3.1 に示す。ピン 8 およびピン 7 間には内部で CMOS インバータと高抵抗が接続されている。したがってここに水晶振動子とコンデンサを図のように接続し、3 点接続型のコルピッツ形発振器を構成してクロック源とした。

また、RX0BF、RX1BF にそれぞれ LED を接続し、送信完了時とエラー発生時に発光するように設定した。高輝度 LED を使用したので電流制限抵抗 5.1kΩ と大きくした。

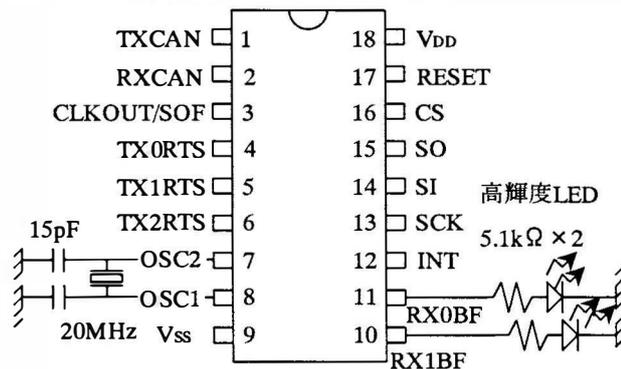


図 4.3.1 MCP2515 のピン配置 (top view)

### 4.4 CAN コントローラ MCP2515 のレジスタ

本検討で用いた CAN コントローラ MCP2515 のレジスタについて説明する。

- (1) コンフィギュレーション・レジスタ (CNF1, CNF2, CNF3)  
表 4.1~表 4.3 に示す。CAN ビット・タイムの設定、CAN フレームのボーレートの設定を行う。
- (2) コントロール・レジスタ (CANCTRL)  
MCP2515 のオペレーション・モードの設定、クロック出力ピンの設定などを行う (表 4.4)。
- (3) ステータス・レジスタ (CANSTAT)  
各種レジスタのステータスを示す。
- (4) 割り込み関係レジスタ (CANINTE, CANINTF)  
割り込み許可および割り込み要因を示すレジスタである。
- (5) 送信バッファ  
TXB0, TXB1 および TXB2 の 3 つの送信バッファがあり、

各々 14 バイトの領域を有する。送信バッファごとにコントロール・レジスタ (TXBnCTRL), メッセージの格納エリア (TXBnSIDx, TXBnEIDx), データ長レジスタ (TXBnDLC) 及びデータ本体 (TXBnD0~TXBnD7) からなる。

レジスタにおける各ビットの内容を表 4.6~表 4.9 に示す。

- (6) 受信バッファ  
RXB0 および RXB1 の 2 つの受信バッファがあり、各々 14 バイトの領域を有する。受信バッファごとにコントロール・レジスタ (RXBnCTRL), メッセージの格納エリア (RXBnSIDx, RXBnEIDx), データ長レジスタ (RXBnDLC) およびデータ本体 (RXBnD0~RXBnD7) からなる。バッファの数は一つ少ないが、送信バッファと対応している。表 4.10~表 4.12 にビットの詳細を示す。
- (7) マスク、フィルタ・レジスタ  
受信バッファに関連するフィルタとマスクのレジスタがある。フィルタ・レジスタはメッセージとフィルタ値が完全に一致するメッセージだけを処理する機能、マスク・レジスタは設定することにより、フィルタ値の範囲を限定させる機能の値を設定するレジスタである。フィルタ・レジスタの内容を表 4.17, マスク・レジスタを表 4.18 に示す。
- (8) エラーフラグ  
発生したエラーの内容が判断できるレジスタであり、読み出し専用である。レジスタの内容を表 4.19 に示す。

### 4.5 MCP2515 の初期設定

MCP2515 はパワーアップ時、リセット時には自動的にコンフィギュレーション・モードに設定される。また各種コンフィギュレーション関係のレジスタを設定する際にも、このコンフィギュレーション・モードにする必要がある。したがって以下に示す設定はコンフィギュレーション・モードにおいて行った。

- (1) ボーレートの設定  
ボーレートを選択するために、以下のレジスタを設定した。  
CNF1 = 0x04  
CNF2 = 0xB1  
CNF3 = 0x05

これより BRP=4, TQ=10/Fosc, PHSEG1=6, PHSEG2=5, PRSEG=1 である。したがって TQ 構成数=16 になる。Fosc=20MHz としたのでボーレートは 125 kbaud である。なお 1 ビット中におけるサンプリングは 1 回だけである。レジスタの内容を表 4.1~4.3 に示す。

- (2) フィルタの設定  
図 4.5.1 にフィルタのレジスタ構成を示す。  
ID 値とフィルタ値の比較はフィルタ 0 から順番に行われ、フィルタ 0~1 で一致した場合は受信バッファ 0 にデータが入り、フィルタ 2~5 で一致した場合は受信バッファ 1 にデータが入る。ID 値がどのフィルタ値とも一致しない場合には、そのデータは処理されない。  
また、フィルタの機能を無効にした場合、すべてのデータを

受信することになる。そのため、フィルタ機能の設定を行う場合には以下を考慮する必要がある。レジスタの内容を表4.17に示す。

① 受信バッファ0 (RXB0) にヒットさせる場合

(U,Vは任意の値)

RXF0~1 フィルタ0~1 標準ID 上位・下位フィルタレジスタ  
 [アドレス:(0x00,0x01)(0x04,0x05)]の設定

RXF0SIDH = 0xUU	} フィルタ値	} ID 値が完全一致するデータを 受信バッファ0へ
RXF0SIDL = 0xU0		
RXF1SIDH = 0xVV		
RXF1SIDL = 0xV0		
	0xVVV0	0xVVV0

② 受信バッファ1 (RXB1) にヒットさせる場合

(W,X,Y,Zは任意の値)

RXF2~5 フィルタ2~5 標準ID 上位・下位フィルタレジスタ  
 [アドレス:(0x08,0x09)(0x10,0x11)(0x14,0x15)(0x18,0x19)]の設定

RXF2SIDH = 0xWW	} フィルタ値	} ID 値が完全一致するデータを 受信バッファ1へ
RXF2SIDL = 0xW0		
RXF3SIDH = 0xXX		
RXF3SIDL = 0xX0		
RXF4SIDH = 0xYY		
RXF4SIDL = 0xY0	0xWWW0	
RXF5SIDH = 0xZZ	0xXXX0	
RXF5SIDL = 0xZ0	0xYYY0	
	0xZZZ0	

なおこれだけではフィルタ機能は動作しない。(6)に示す受信バッファ0, 1のコントロール・レジスタの設定が必要である。

またマスク機能を表4.18に示す。今回、マスク機能は使用していないので説明は省略する。

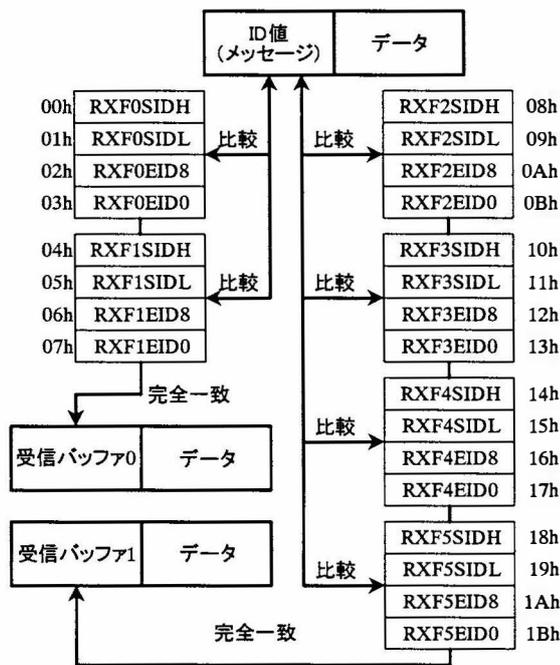


図4.5.1 フィルタのレジスタ構成

(3) LED を点灯させるための設定

表4.16に示すレジスタを以下のように設定した。

BFPCTRL = 0x0C

(4) モードの設定

表4.4 に示すレジスタの内容を以下のように設定した。

CANCTRL = 0x0E

これはノーマル・モード, 1回だけ送信するワンショット・モード, CLKOUTピンを使用する設定である。

これ以降は送信側と受信側で設定が分かれる。

(5) 送信側ノードの設定

各レジスタを表4.6~4.9 に示す。

- TXB0CTRL ... 送信バッファの優先度の設定 (送信ノードの優先度ではない)

- TXBnSIDH, TXBnSIDL (送信ノードn標準ID)
  - ... 送信ノードの優先度の設定 (小さいほど高優先度)
 この値と受信側のフィルタ値を一致させることにより処理を振り分けることが可能になる。

- TXB0DLC ... データ長の設定 (0~8バイト)

(6) 受信側ノードの設定

各レジスタを表4.10~4.12 に示す。

- 標準IDのフィルタとマスクを適用するために

RXB0CTRL=0x00, RXB1CTRL=0x00

に設定した。

- RXB0DLC ... 受け取るデータのデータ長の設定 (0~8バイト)

4.6 LCD への表示

測定したデータを受信部において表示させるために16文字×2行のLCD (SCI602B) を4ビットモードにより用いた。このLCDの制御にはMicrochip社のライブラリLCD\_BUSY.asmを使用した。使用方法を以下に示す。

- ① LCD\_BUSY.asmをインクルード
- ② CALL LCD\_INI ; 初期化コマンド
- ③ CALL LCD\_DATA ; データを表示するコマンド
- ④ CALL LCD\_CMD ; コマンドを送信する

4.7 モータにおける回転数の測定結果

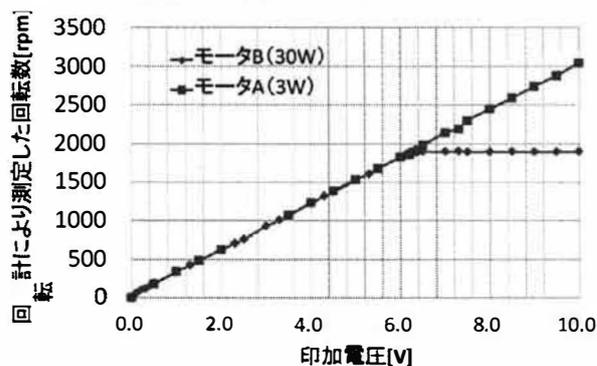


図 4.7.1 モータの印加電圧と回転数の測定結果

送信ノードにおいて使用したDCモータの印加電圧と回転計を用いて測定した回転数の測定結果の関係を図4.7.1に示す。モータA (3W) およびモータB (30W) の2個を使用した。モータAでは印加電圧の増加に比例して回転数が増加する関係が確認できた。しかし、モータBはある一定の印加電圧までは回転数が印加電圧に比例するが、その電圧を超えると回転数が一定になる特性を示していた。

5. 回転数の送受信に用いた処理プログラムの構成

5.1 概略フローチャート

送信部における処理プログラムの概略フローチャートを図5.1(a)に示す。送信部では、まずSPIの設定、MCP2515の設定およびAD変換機能の設定を行う。その後AD変換によるモータの印加電圧の測定およびその電圧を回転数に変換する演算を行う。またSPIコマンドを用いてデータを送信バッファに書き込み、送信要求を発行してCANバスからデータを送信する。なおエラーの発生を検出し、エラー発生時には再送処理を行うようにプログラムを作成している。

受信部における概略フローチャートを図5.1(b)に示す。受信部では、SPIの設定、MCP2515の設定およびLCDの設定を行った後に、SPIの受信ステータス・リード・コマンド(表4.14)を用いて受信データの有無を確認する。受信データがあるとそのデータがどちらの受信バッファにヒットしたかをビット6およびビット7を見て判断し、LCDに表示するようにしている。

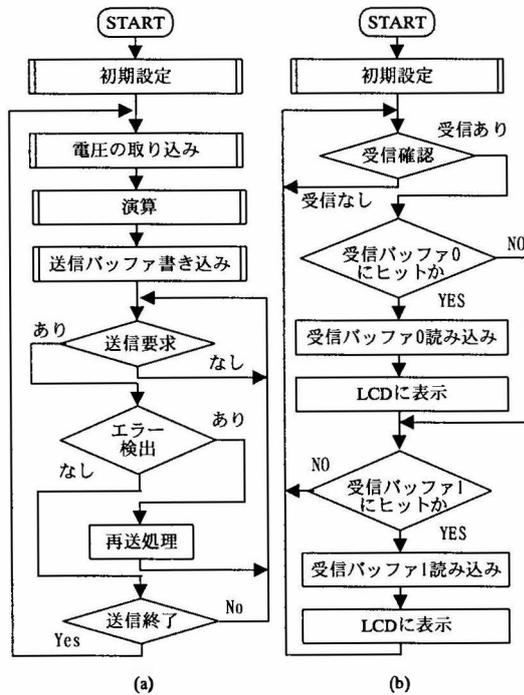


図 5.1 概略フローチャート

5.2 送信部におけるAD変換出力の回転数への変換

図4.7.1に示したように使用したモータには、ある一定の印加電圧以下において印加電圧と回転数に比例関係があった。その

比例定数は図4.7.1より 300rpm/Vである。これより送信ノードにおいてモータの印加電圧を回転数に変換するために、図5.2.1に示すプロセスを用いた。このAD変換のプロセスは2バイトの処理となる。プログラムを図5.2.2に示す。このサブプログラムを実行すると上位バイトがUPDATAに、下位バイトがLOWDATAに得られる。この値を3倍するために、これらの変数を左に2ビットシフトし、元の値との和を求めている。

送信部 AD変換の流れ

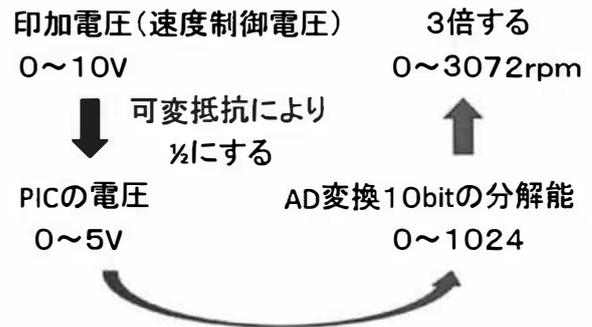


図 5.2.1 送信部における電圧から回転数への変換

```

VOLGET
    MOVLW    0x81      ;設定 CH0 A/D ON
    MOVWF   ADCON0    ;select CH0
    CALL    TIME15    ;wait settling
    BSF     ADCON0,GO ;start A/D

WAIT
    BTFSC   ADCON0,GO
    GOTO    WAIT
    MOVF   ADRESH,W  ;get A/D data
    MOVWF  UPDATA
    BSF     STATUS,RP0 ;バンク1へ切り替え
    MOVF   ADRESL,W
    BCF     STATUS,RP0 ;バンク0へ切り替え
    MOVWF  LOWDATA
    BCF     ADCON0,0  ;A/D off
    
```

図 5.2.2 送信部のPIC16F819におけるAD変換プログラム

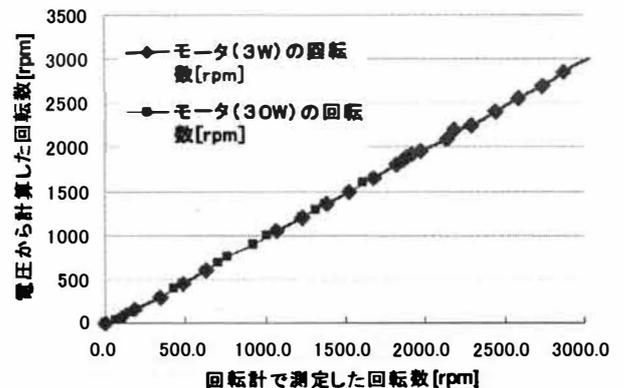


図 5.5.1 送信部で測定した回転数と受信部で確認した回転数

5.4 受信部における電圧から回転数への変換

AD変換結果が2バイトのバイナリデータとして受信部に送信されてくる。これを5桁の2進化10進数に変換してLCDに表示する。そのためにMicrochip社のライブラリを使用した。使

用方法を以下に示す。

- ① ライブラリ HEX16DEC.ASM をインクルード
- ② CALL B16BCD を実行

この結果5桁の2進法10進数がb4,b3,b2,b1,b0に得られる。

5.5 受信部で確認した回転数の測定結果

図 5.51 に回転数の比較を示す。送信部の回転計で測定した回転数と、CAN を通じて受信部に伝送された値から求めた回転数とは良好に一致している。誤差率は2%程度であった。

6. フィルタ機能の確認

送信ノード4台、受信ノード2台を用いてCAN通信を行った。最優先度のデータ [ID 値:0xUUUU] を受信ノード1の受信バッファ0で、次に優先度の高いデータ [ID 値:0xWWWW] を同じく受信ノード1の受信バッファ1で受信するためにそのフィルタ値を RXF0SID=0xUUUU, RXF2SID=0xWWWW に設定した。その他の送信ノードのデータは受信しないようにした。

また受信バッファ0のデータを LCD の CH1 に、受信バッファ1のデータを CH2 に表示した結果を図 6.1 に示す。



図 6.1 フィルタ機能による LCD への表示

7. 調停 (アービトレーション) の確認

7.1 調停確認用回路の構成

調停が発生する条件として以下が挙げられる。

- ・ 送信ノードが2つ以上存在する。
- ・ 送信タイミングが同時である。

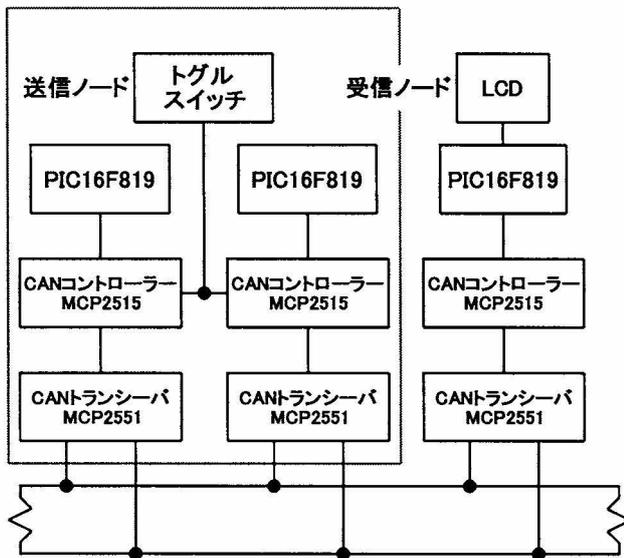


図 7.1 調停確認用回路

この条件を満たす状況を作り出すために、図 7.1 に示すように送信ノード2台、受信ノード1台からなる調停確認用回路を1枚のボード上に製作した。また送信タイミングを同時にするために、トグルスイッチを図 7.1 に示した MCP2515 の TX0RTS 端子に接続して送信ノードに送信要求を出すようにした。なお、チャタリングの影響を除くためにシュミットトリガ付のインバータ2段を介して接続している。

処理プログラムのフローチャートは図 5.1 と全く同じである。ただ送信要求が外部スイッチによって行われる点が異なる。

7.2 外部入力により送信要求を発行する方法

プログラムでデータをバッファに書き込むまでの作業を行って送信要求を待っているの、トグルスイッチを一回押すごとに送信が行われる。しかし、トグルスイッチを離すとバスエラーが発生したので、送信波形が観測できるまで押し続けておくようにして波形を測定した。また、送信と同時に LED が発光するように設定し、調停が発生したタイミングを計る材料とした。

MCP2515 を外部入力により送信要求を発行して動作させるためには表 4.15 に示す TXRTSCTRL レジスタ (アドレス 0x0D) に適切な値 TXRTSCTRL=0x09 を設定する必要がある。

ビットで表すと以下ようになる。

- ① B0RTS=1・・・スイッチオフ時に1 (High) に設定
- ② B0RTSM=1・・・外部入力による送信要求発行に設定

7.3 調停の確認波形

2台の送信部において優先度の高いメッセージ値を 0x0000、優先度の低いメッセージ値を 0x0F00 として送信を行った。結果を図 7.3.1 に示す。また、調停が発生したときのオシロスコープによる観測波形を図 7.3.2 に、図 7.3.1 の左側の□で囲った部分をビット対応で書き表したものを図 7.3.3 に示す。優先度の高いメッセージは即時に受信部に受信されることが分かる。

一方優先度の低いメッセージは、送信したデータと受信したバスのレベルが同一である時点までは送信されているが、それ以後は送信が中断されている。これは送信したデータと受信したバスのレベルが異なる時点で調停負けが判断されたことによる。これより 2.2 の伝送路アクセス制御の方法に示した調停の動作を実験的にも確認することができた。

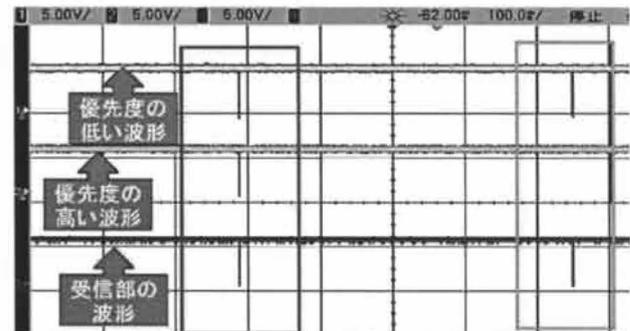


図 7.3.1 調停発生時における送受信の全体像

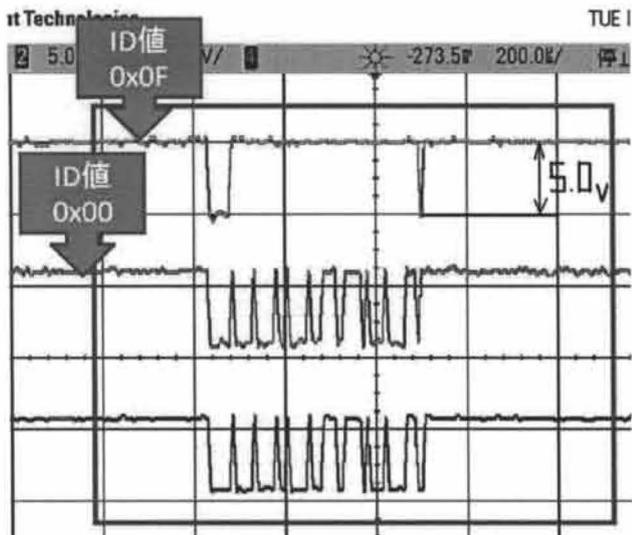


図 7.3.2 調停が発生したときの時間軸の拡大波形

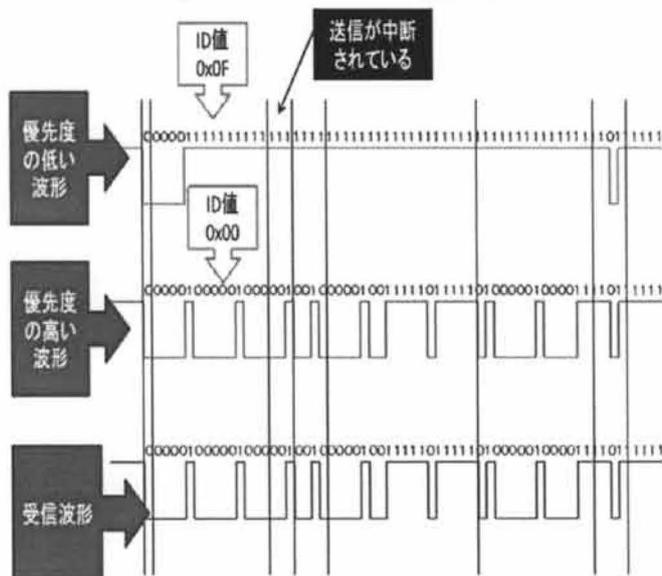


図 7.3.3 調停が発生したときの時間軸の拡大図 (その2)



図 7.3.4 再送された波形 (上:送信波形, 下:受信波形)

図 7.3.4 は調停負けした優先度の低いメッセージの再送処理された送信波形と受信ノードにおける受信波形である。CSMA/CA 方式を用いている CAN のアクセス制御では、CAN バスに早くアクセスしたメッセージが勝ち残る。そのため、負

けたメッセージは CAN バスが空くまで待機してから送信する必要がある。本検討では調停が発生した場合にも、負けたメッセージを失わないように再送処理を行うようにしている。

### 8. むすび

本論文では、4 台の送信ノードと 2 台の受信ノードおよび CAN 伝送路からなる CAN 通信システムの机上モデルを試作し、その設計法を示した。また雑音や干渉のない環境下においてシステムの伝送特性を測定し評価した。

これにより以下の結果が得られた。

- (1) 送信部において測定した DC モータの回転数を AD 変換し、CAN 通信により受信部に伝送し、その受信データを LCD に表示することにより実用的に問題のない結果を得ることができた。誤差率は 2% 程度であった。
- (2) 複数のノードから送信されるメッセージを、CAN コントローラ MCP2515 のフィルタ機能を用いて選択的に受信する方法を示し、実際にシステムの評価に適用した。
- (3) 2 つ以上のノードが同時に送信を試みた場合に行われる調停の確認を行うための回路を製作し、その波形をオシロスコープにより観測した。優先度は ID 値により決定し、調停が発生しても高優先度のメッセージは即時に受信されることを観測結果から確認した。また、調停負けしたメッセージの再送処理が期待通りに動作することを確認した。

今回製作したシステムは机上モデルである。実際に電気自動車に CAN 通信システムを搭載して実験を行っていくことが今後の検討課題である。

### 【謝辞】

電気電子工学科服部 敬和 氏には、モータの回転計を借用させて頂いた。深く感謝いたします。

### 【参考文献】

- (1) 袴田吉朗, 小栗宏和, "CAN 通信による自動車内データ収集システム—机上モデルの設計と試作", 静岡理科大学紀要, Vol.19, pp.27-36 (2011)
- (2) MCP2515 SPI スタンドアロン CAN コントローラ, Microchip
- (3) 小川晃, "PICCAN テクニカルガイド", Micro Application Laboratory
- (4) 中尾司, "動かして学ぶ CAN 通信", CQ 出版社
- (5) PIC16F818/819 Data Sheet 18/20-Pin Enhanced Flash Micro controllers with nano Watt Technology, Microchip
- (6) 後関哲也, "改訂版 電子工作のための PIC16F 活用ガイドブック", 技術評論社
- (7) "ELECTRONICS SHELVES (趣味の電気工作)", <http://orange.zero.jp/electronics/>

表 4.1 CNF1 (アドレス: 0x2A)

b <sub>7</sub>	SJW1	同期ジャンプ長	11= 4×TQ, 10= 3×TQ 01= 2×TQ, 00= 1×TQ
b <sub>6</sub>	SJW0		
b <sub>5</sub>	BRP5	ポーレート・プリスケアラ	ポーレート・プリスケアラ値設定 $TQ=(BRP+1)/F_{osc}$
b <sub>4</sub>	BRP4		
b <sub>3</sub>	BRP3		
b <sub>2</sub>	BRP2		
b <sub>1</sub>	BRP1		
b <sub>0</sub>	BRP0		

表 4.2 CNF2 (アドレス: 0x29)

b <sub>7</sub>	BTLMODE	PS2時間長ビット	1=PS2の長さはPHSEG22-PHSEG20で定義 0=PS2の長さはPS1とIPT(2TQ)より大きい
b <sub>6</sub>	SAM		1=3回サンプリング 0=1回だけサンプリング
b <sub>5</sub>	PHSEG12	PH1ビット長	PH1のビット長 = (PHSEG1+1)×TQ
b <sub>4</sub>	PHSEG11		
b <sub>3</sub>	PHSEG10		
b <sub>2</sub>	PHSEG2	PropSegビット長	プロパゲーション・セグメントのビット長 = (PRSEG+1)×TQ
b <sub>1</sub>	PHSEG1		
b <sub>0</sub>	PHSEG0		

表 4.3 CNF3 (アドレス: 0x28)

b <sub>7</sub>	SOF	SOFピン設定	1=CLKOUTピンをSOFピンに設定 0=CLKOUTピンをクロック出力に設定
b <sub>6</sub>	WAKFIL	ウェイクアップ・フィルタ	1=フィルタを有効 0=フィルタを無効
b <sub>5</sub> ~ b <sub>3</sub>		未実装	
b <sub>2</sub>	PHSEG22	PH2ビット長	PH2のビット長を設定 (最小設定値2) (PRSEG2+1)×TQ
b <sub>1</sub>	PHSEG21		
b <sub>0</sub>	PHSEG20		

表 4.5 CANINTF (アドレス: 0x2C)

		割込要因	
b <sub>7</sub>	MERR	メッセージ・エラー	1=割込発生中(ソフトでクリア必要)
b <sub>6</sub>	WAKIF	ウェイクアップ	
b <sub>5</sub>	ERRIF	エラー	1=割込未発生
b <sub>4</sub>	TX2IF	送信バッファ2空	
b <sub>3</sub>	TX1IF	送信バッファ1空	
b <sub>2</sub>	TX0IF	送信バッファ0空	
b <sub>1</sub>	RX1IF	送信バッファ1フル	
b <sub>0</sub>	RX0IF	送信バッファ0フル	

表 4.4 CANCTRLレジスタ(アドレス: 0x×F)

b <sub>7</sub>	REQOP2	オペレーション・モード設定要求	000= ノーマル・モード
b <sub>6</sub>	REQOP1		001= スリープ・モード
b <sub>5</sub>	REQOP0		010= ループアップ 011= リスン・オンリ 100= コンフィグ・モード
b <sub>4</sub>	ABAT	全送信中断	1= すべて送信を中断 0=送信中断要求を終了
b <sub>3</sub>	OSM	ワンショット・モード	1= ワンショット・モード 0= 非ワンショット・モード
b <sub>2</sub>	CLKEN	CLKOUTピン許可	1=CLKOUTピン使用する 0=使用しない(Hi-Z)
b <sub>1</sub>	CLKPRE1	CLKOUTピンプリスケアラ	クロックの分周比を設定 00=F <sub>osc</sub> /1, 01=F <sub>osc</sub> /2 10=F <sub>osc</sub> /4, 11=F <sub>osc</sub> /8
b <sub>0</sub>	CLKPRE0		

表 4.6 TXBnCTRLレジスタ (アドレス: 0x30, 0x40, 0x50)

b <sub>7</sub>	-	未実装	
b <sub>6</sub>	ABTF	メッセージ・アポート	1= メッセージ送信は中断 0= メッセージ送信成功
b <sub>5</sub>	MLOA	メッセージ喪失	1= 調停負け, メッセージ喪失 0= 正常に送信
b <sub>4</sub>	TXERR	転送エラー検出	1= 転送エラー発生, メッセージ保留 0= 保留中の送信なし
b <sub>3</sub>	TXREQ	メッセージ送信要求	1= メッセージ送信要求 0= 保留中の送信要求なし
b <sub>2</sub>	-	未実装	
b <sub>1</sub>	TXP1	転送バッファ優先順位	11= 優先度(最高)~ 00= 優先度(最低)
b <sub>0</sub>	TXP0		

表 4.7 TXBnSIDHレジスタ(n=0,1,2)(アドレス: 0x31,0x41,0x51)

b <sub>7</sub>	SID10	標準IDビット10
b <sub>6</sub>	SID9	標準IDビット9
b <sub>5</sub>	SID8	標準IDビット8
b <sub>4</sub>	SID7	標準IDビット7
b <sub>3</sub>	SID6	標準IDビット6
b <sub>2</sub>	SID5	標準IDビット5
b <sub>1</sub>	SID4	標準IDビット4
b <sub>0</sub>	SID3	標準IDビット3

表 4.8 TXBnSIDLレジスタ(n=0,1,2)(アドレス: 0x32,0x42,0x52)

b <sub>7</sub>	SID2	標準IDビット2
b <sub>6</sub>	SID1	標準IDビット1
b <sub>5</sub>	SID0	標準IDビット0
b <sub>4</sub>		
b <sub>3</sub>	EXIDE	拡張ID許可ビット 1=拡張ID(29ビット) 0=標準ID(11ビット)
b <sub>2</sub>		
b <sub>1</sub>	EID4	拡張IDビット17
b <sub>0</sub>	EID16	拡張IDビット16

表 4.9 TXBnDLCレジスタ (アドレス : n=0x35, 0x45, 0x55)

b7	-	未使用	
b6	RTR	リモート送信 要求	1= リモート・フレーム 0= データ・フレーム
b5	-	未実装	
b4	-	未実装	
b3	DLC3	データ長 コード(DLC)	データ・フィールドの データ長(0-8)
b2	DLC2		
b1	DLC1		
b0	DLC0		

表 4.10 RXB0CTRLレジスタ (アドレス : 0x60)

b7	-	未実装	
b6	RXM1	オペレーショ ン・モード	11= マスク, フィルタ未使用 10= 拡張ID, マスク, フィルタ を適用 01= 標準ID, マスク, フィルタ を適用 00=マスク, フィルタを適用
b5	RXM0		
b4	-	未実装	
b3	RXRTR	リモート送信 要求	1=送信要求を受けた 0=送信要求は受けていない
b2	BUKT	ロールオーバー・オ プション	1= ロール・オーバーを許可 0= ロール・オーバー許可せず
b1	BUKT1	リード・ オンリ・コピー	MCP2515 が内部で使用するフラ グ
b0	FILHIT0	フィルタ・ ヒット	1= フィルタ1(RXF1)にヒット 0= フィルタ0(RXF0)にヒット

表 4.11 RXB1CTRLレジスタ (アドレス : 0x70)

b7	-	未実装	
b6	RXM1	オペレー ション・ モード	11= マスク, フィルタ未使用 10= 拡張ID, マスク, フィルタ を適用 01= 標準ID, マスク, フィルタ を適用 00=マスク, フィルタを適用
b5	RXM0		
b4	-	未実装	
b3	RXRTR	フィルタ・ ヒット	101= フィルタ5にヒット
b2	FILHIT2		100= フィルタ4)にヒット
b1	FILHIT1		011= フィルタ3にヒット
b0	FILHIT0		010= フィルタ2にヒット
			001= フィルタ1にヒット
		000= フィルタ0にヒット	

表 4.12 RXBnDLCレジスタ (アドレス : n=0...0x65)

b7	-	未実装	
b6	RTR	リモート送信要求	1=リモート・フレーム 0=データ・フレーム
b5	RB1	リザーブ1	
b4	RB0	リザーブ2	
b3	DLC3	データ長コード (DLC)	データ・フィールドの データ長(0-8)
b2	DLC2		
b1	DLC1		
b0	DLC0		

表 4.13 ステータス・リード・コマンドにおけるビット定義

bit	要因	意味
b7	TXB0CNTRL TXREQ	送信バッファ0 送信要求
b6	TXB1CNTRL TXREQ	送信バッファ1 送信要求
b5	TXB2CNTRL TXREQ	送信バッファ2 送信要求
b4	CANINTF. TX2IF	送信バッファ2 「空」
b3	CANINTF. TX1IF	送信バッファ1 「空」
b2	CANINTF. TX0IF	送信バッファ0 「空」
b1	CANINTF. RX1IF	受信バッファ1 「フル」
b0	CANINTF. RX0IF	受信バッファ0 「フル」

表4.14 受信ステータス・リード・コマンド(RX STATUS)

bit	要因	意味
b7	受信メッセ ジ	00= 受信なし 01= RXB0 にマッチ 10= RXB1 にマッチ 11= 両バッファに受信あり
b6		
b5		未実装
b4	メッセージ・ タイプ	00= スタンダード・データ・フレーム 01= スタンダード・リモート・フレーム 10= 拡張データ・フレーム 11= 拡張リモート・フレーム
b3		
b2 ~ b0	フィルタ・ マッチ	000= RXF0 にマッチ 001= RXF1 にマッチ 010= RXF2 にマッチ 011= RXF3 にマッチ 100= RXF4 にマッチ 101= RXF5 にマッチ 110= RXF0 にマッチ(RXB1 ヘロール・オーバー) 111= RXF1 にマッチ(RXB1 ヘロール・オーバー)

表4.15 TXRISCTRL:TXnRTSピン制御レジスタ(アドレス 0x0D)

b7		なし
b6		なし
b5	B2RTS	*TX2RTS ピンステータス 0 ポート端子=0 1 ポート端子=1
b4	B1RTS	*TX1RTS ピンステータス 0 ポート端子=0 1 ポート端子=1
b3	B0RTS	*TX0RTS ピンステータス 0 ポート端子=0 1 ポート端子=1
b2	B2RTSM	*TX2RTS ピンモード指定 0 デジタル入力ポート 1 TXB2 送信スタート
b1	B1RTSM	*TX1RTS ピンモード指定 0 デジタル入力ポート 1 TXB1 送信スタート
b0	B0RTSM	*TX0RTS ピンモード指定 0 デジタル入力ポート 1 TXB0 送信スタート

表4.16 BFPCTRL:RXnBFピン制御レジスタ(アドレス:0x0C)

b7		なし
b6		なし
b5	B1BFS	RX1BF ピンステータス ポート端子=0 ポート端子=1
b4	B0BFS	RX0BF ピンステータス ポート端子=0 ポート端子=1
b3	B1BFE	RX1BF ファンクション デジタル出力ポート 1 有効
b2	B0BFE	RX0BF ファンクション デジタル出力ポート 1 有効
b1	B1BFM	RX1BF ピンモード指定 デジタル出力ポート RXB1 メッセージのロード
b0	B0BFM	RX0BF ピンモード指定 デジタル出力ポート RXB0 メッセージのロード

表4.17 受信バッファのフィルタ・レジスタ

(a) RXFnSIDH フィルタ n 標準 ID 上位(n=0,1,2,3,4,5)

(アドレス: 0x00,0x04,0x08,0x10,0x14,0x18)

b7	b6	b5	b4	b3	b2	b1	b0
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3

SID10~SID3…標準 ID ビット 10~3

(b) RXFnSIDL フィルタ n 標準 ID 下位(n=0,1,2,3,4,5)

(アドレス: 0x01,0x05,0x09,0x11,0x15,0x19)

b7	b6	b5	b4	b3	b2	b1	b0
SID2	SID1	SID0	-	EXIDE	-	EID17	EID16

SID2~SID0…標準 ID ビット 2~0

EXIDE…拡張 ID 許可ビット

'1'=拡張 ID(29 ビット)のみに適用

'0'=拡張 ID(11 ビット)のみに適用

EID17,EID16…拡張 ID ビット 17,16

表4.18 受信バッファのマスク・レジスタ

(a) RXMnSIDH マスク n 標準 ID 上位(n=0,1)

(アドレス: 0x20,0x24)

b7	b6	b5	b4	b3	b2	b1	b0
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3

SID10~SID3…標準 ID ビット 10~3

(b) RXMnSIDL マスク n 標準 ID 下位(n=0,1)

(アドレス: 0x21,0x25)

b7	b6	b5	b4	b3	b2	b1	b0
SID2	SID1	SID0	-	-	-	EID17	EID16

SID2~SID0…標準 ID ビット 2~0

EID17,EID16…拡張 ID ビット 17,16

表4.19 エラーフラグ(アドレス:0x2D)

b7	MERRF	メッセージ・エラー 割り込み	1= 割り込み発生中* 0= 割り込み未発生
b6	WAKIF	ウェイクアップ 割り込み	1= 割り込み発生中* 0= 割り込み未発生
b5	ERRIF	エラー 割り込み	1= 割り込み発生中* 0= 割り込み未発生
b4	TX2IF	送信バッファ2 空割り込み	1= 割り込み発生中* 0= 割り込み未発生
b3	TX1IF	送信バッファ1 空割り込み	1= 割り込み発生中* 0= 割り込み未発生
b2	TX0IF	送信バッファ0 空割り込み	1= 割り込み発生中* 0= 割り込み未発生
b1	RX1IF	受信バッファ1 フル割り込み	1= 割り込み発生中* 0= 割り込み未発生
b0	RX0IF	受信バッファ0 フル割り込み	1= 割り込み発生中* 0= 割り込み未発生

\*:ソフトウェアでクリア必要)