飛翔ソフトウェア

新人研修

マイコン制御編　CAN通信規格まとめ

|  |  |  |
| --- | --- | --- |
| 日付 | 変更内容 | 変更者 |
| 2021//7/15 | 新規作成 | 庭 |
|  |  |  |

# CAN規格　物理層

## CAN High speedとLow speed

CANプロトコルの規格にはHigh speedとLow speedがあり、違いは物理層にある。

(1)High speed(ISO11898)

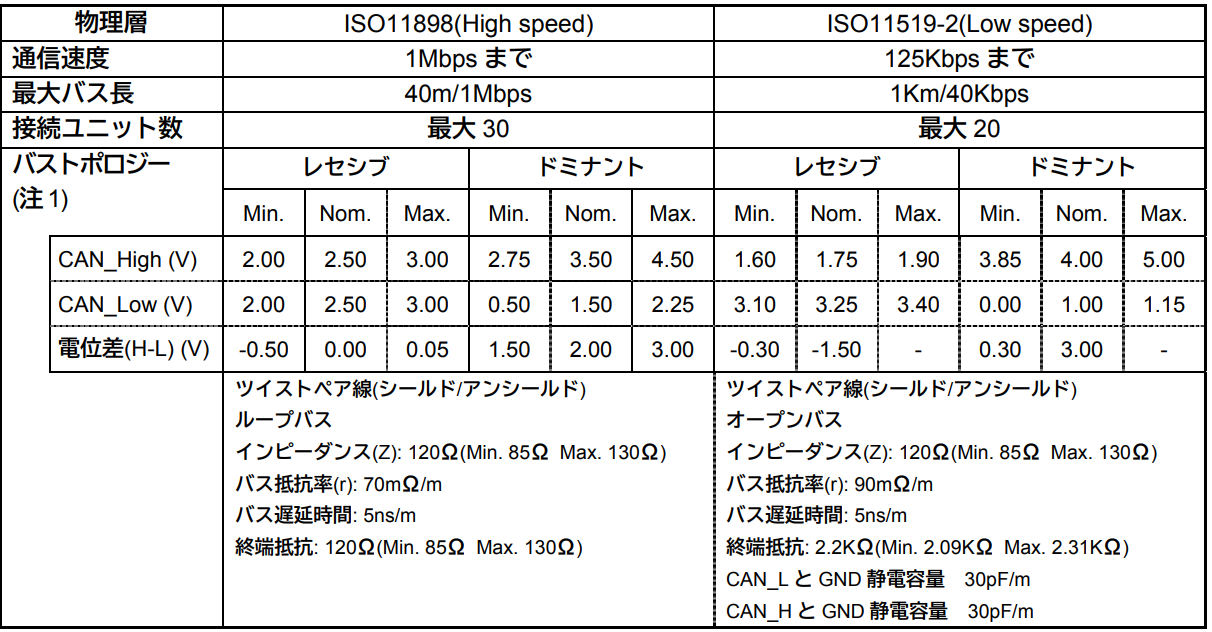
通信速度が125Kbps～1Mbpsのハイスピード通信の規格。主に高速なデータ交換が要求されるパワートレイン系※で使用される。通信は2本線を使用し、電圧差がある状態をドミナント（優性）と呼び、論理’0’、そうでない場合をレセシブ（劣性）と呼び論理’1’を表す。ドミナントとレセシブが同時に送信された場合はドミナントが優先される。電圧差にて判断することにより外来ノイズからの影響を受けにくくしている。

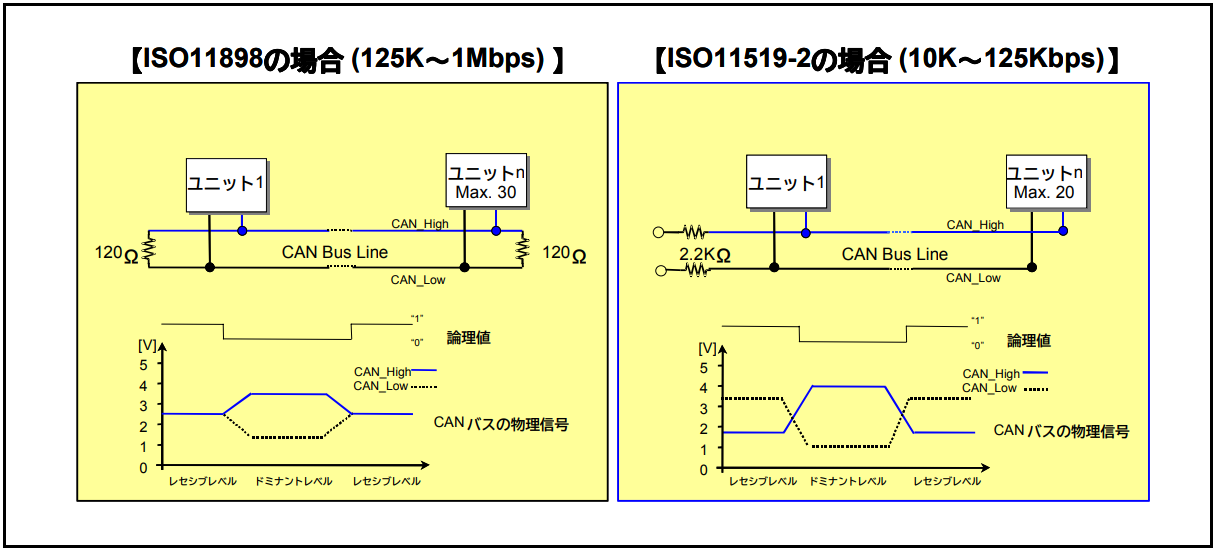
※パワートレイン系…エンジンで発生した回転エネルギーを効率よく駆動輪に伝えるための装置類の総称のこと。具体的なパーツでは、動力を発生させるエンジンや、それを伝達するクラッチ、トランスミッション、ドライブシャフトなど

(2)Low speed(ISO11519)

通信速度が～125Kbpsのロースピード通信の規格。主にそれほど高速なデータ交換が要求されないボディ系、快適装備系で使用される。通常、2本線を使用して、High Speed CANのように電圧差にて通信を行う。ただし、通信線に問題が発生した場合、シングルワイヤーモードに切り替わり単線にて通信を行うことが可能。これは通信線が自動車内でも外側部分に配置される場合があるため、フォールトトレランス（1つが故障しても処理を正常に続行すること）が考慮されている。

RENESAS アプリケーションノート CAN入門書より抜粋





## CAN ビットタイミング

各ビットを確実に認識し、耐障害性を高めるために、CANでは1ビットの転送にかかる時間を細分化している。細分化することで、CANバスの状態に応じて、信号を認識するポイントを操作できる。

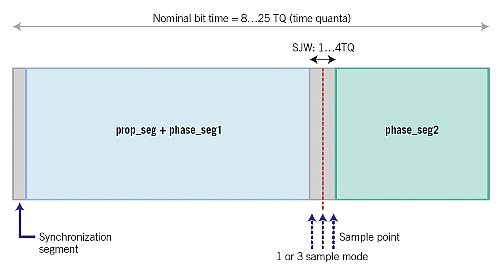
CAN上で1ビットを伝送する時間は、先頭から「Sync\_Seg」、「Prop\_Seg」、「Phase\_Seg1」、「Phase\_Seg2」の4つの期間に分けられる。この4つの期間の長さを操作することで、ビットを検知する時点（サンプル・ポイント）を前後に移動させることができる。

Sync\_Segは、バスとの同期を取るために使う。バスにつながる機器は全てこの期間内で信号の送受信を始めようとする。信号の立ち上がり、立ち下がり（エッジ）がこの点にあることが望ましい。

Prop\_Segは、ネットワークの遅延を吸収するための期間。ここを伸縮させることで、CANバスで発生する遅延や、バスからノードへの配線の遅延の影響を避けることができる。

Phase\_Seg1とPhase\_Seg2は、信号のエッジが早すぎたり、遅すぎたりしたときに、伸縮させる期間。信号を拾うサンプル・ポイントはPhase\_Seg1の直後にあるので、例えば、エッジがSync\_Segよりも遅いタイミングで来たときは、Phase\_Seg1を伸ばし、Phase\_Seg2を縮めることで、サンプル・ポイントをずらし、正確に信号を捉える。

<https://eetimes.itmedia.co.jp/ee/articles/0912/15/news104_2.html>より抜粋



4つの期間を細分化した1つ1つを「Time Quanta（TQ）」と呼ぶ。

Sync\_Seg、Prop\_Seg、Phase\_Seg1、Phase\_Seg2のそれぞれの長さは規格で決まっている。Sync\_Seg：は1TQ、Prop\_Segは1TQ～8TQ、Phase\_Seg1は1TQ～8TQ、Phase\_Seg2は2TQ～8TQとなる。この範囲で、1ビットが合計8TQ～25TQになるように設定する。Phase\_Seg1とPhase\_Seg2の調整幅の合計をreSynchronization Jump Width（SJW）と呼ぶ。

## ビットスタッフィング

ビットスタッフは周期的に再同期を行い受信ノード間のタイミング誤差が累積しないようにするための機能で、通信線上で同じ状態が5回連続した場合、それまで送信されていた状態と反対の状態のビット（“スタッフビット”）を1個挿入する。

# CAN規格　データリンク層

## フレーム

通信は次の4種類のフレームによって行われる。

・データフレーム

・リモートフレーム

・オーバーロードフレーム

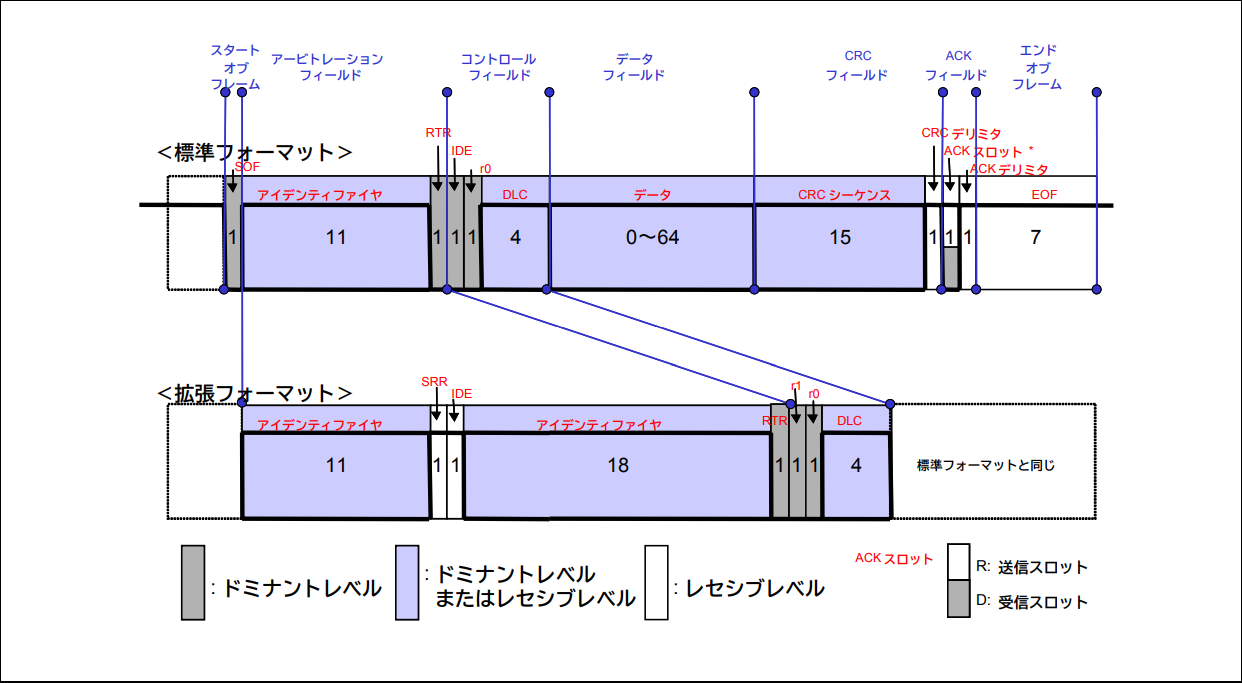
・エラーフレーム

### データフレームとリモートフレーム

データフレームとリモートフレームには標準フォーマット(2.0A)と拡張フォーマット(2.0B)の2つのフレームフォーマットがあり、標準フォーマットは11ビットのIDを、拡張フォーマットは 29ビットのIDを持つ。

データフレームの構成

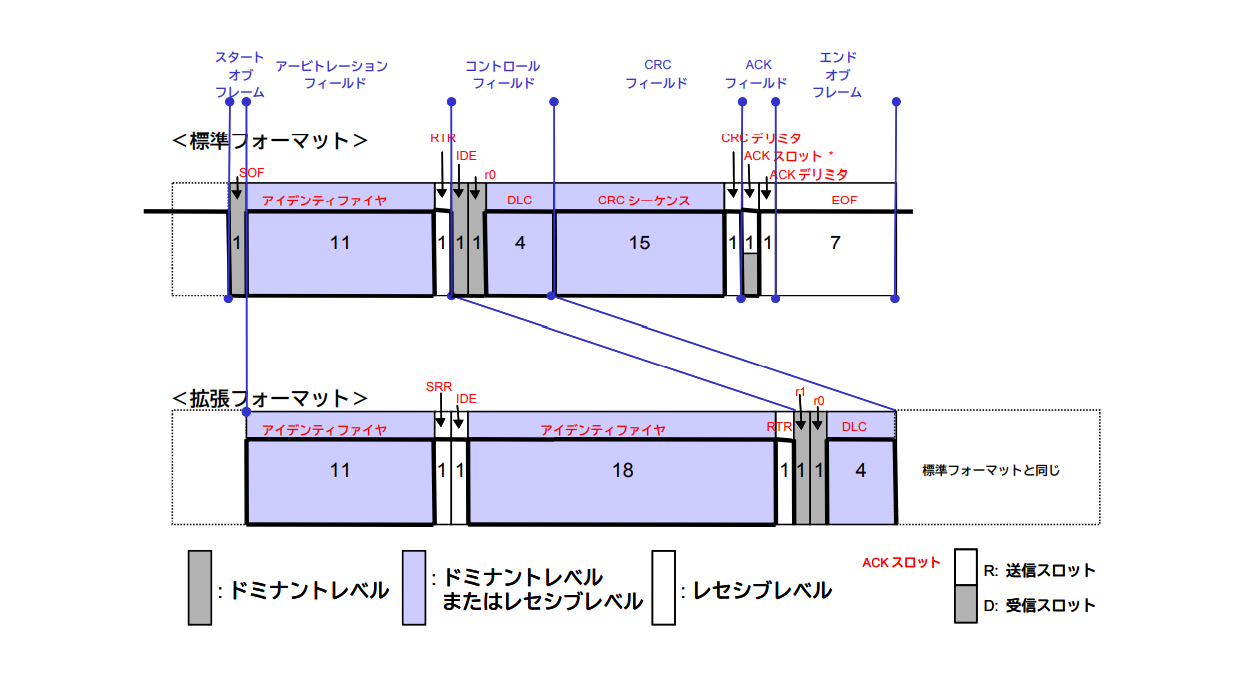
RENESAS アプリケーションノート CAN入門書より抜粋



|  |  |
| --- | --- |
| スタートオブフレーム  (SOF) | データフレームの開始を表すフィールド。  1ビットのドミナントビット |
| アービトレーションフィールド  (調停フィールド) | フレームの優先順位を表すフィールド  ・標準フォーマットのIDは11ビット  ・上位7ビットがすべてレセシブであることは禁止  ・拡張フォーマットのIDは11ビット＋18ビットの合計29ビット  ・バス上の複数のユニットが同一のIDを持つデータフレームを同時に送信することは禁止 |
| コントロールフィールド  (制御フィールド) | 予約ビットとデータのバイト数を表すフィールド  6 ビットの送信するメッセージのデータバイト数を表すフィールド |
| データフィールド | データの中身。0～8 バイト。 |
| CRC フィールド | フレームの伝送誤りをチェックするフィールド |
| ACK フィールド | 正常受信した確認の合図を表すフィールド  ACKスロットとACKデリミタの2ビット  送信ユニットはACKスロットとACKデリミタをレセシブビットで送信。正しいメッセージを受信した受信ユニットは、ACKスロットでドミナントビットを送り正常受信の完了を送信ユニットに知らせる。 |
| エンドオブフレーム | データフレームの終了を表すフィールド。  7ビットのレセシブビット |

リモートフレームの構成

RENESAS アプリケーションノート CAN入門書より抜粋



データフレームとリモートフレームの違い

・リモートフレームはデータフィールドを取り除き、アービトレーションフィールドのRTRビットをレセシブレベルにしたもの。

・データフィールドがないデータフレームとリモートフレームではRTRビットで区別する。

### ITM

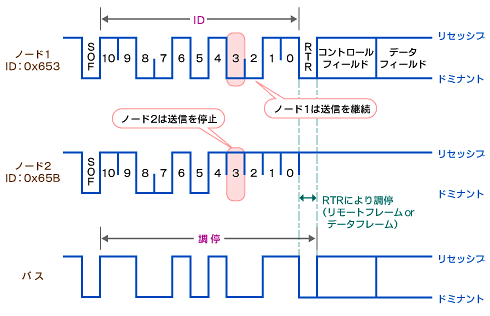
データフレームの範囲はSOF～EOFまでだが、EOF終了後に「ITM（Intermission）」と呼ばれる3ビット長のレセシブ信号が挟まり、その後バスアイドルとなる。

### 調停

CANではバス使用中に他ノードはデータフレームやリモートフレームを送信することができない仕様になっているが、実際には複数ノードから同時に送信されてしまうことを防ぐことができない。そのため、通信調停を行う必要がある。

CANにおいて通信調停に使用されるのはアービトレーションフィールドで、複数のユニットが同時に送信を開始した場合、送信ユニットは、アービトレーションフィールドの1ビット目からアービトレーション(調停)を行い、"ドミナントレベル"を最も長く連続して出力したユニットが送信することができる。アービトレーションに負けたユニットは次のビットから受信動作へ移る。

<https://monoist.atmarkit.co.jp/mn/articles/0807/09/news140_3.html>より抜粋

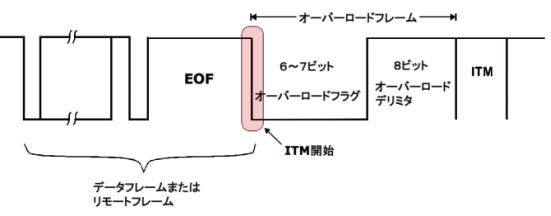


### オーバーロードフレーム

受信ユニットが受信準備未完了を通知するためのフレームで、次のフレームの開始を遅延させるために用いられる。

オーバーロードフレームは「オーバーロードフラグ」と「オーバーロードデリミタ」から構成される。

ベクタージャパン　はじめてのCAN/CAN FDより抜粋



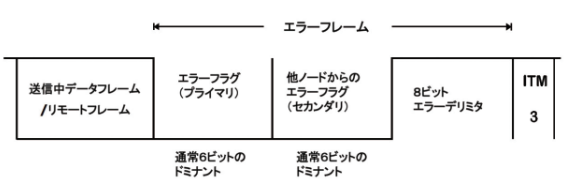
|  |  |
| --- | --- |
| オーバーロードフラグ | 6ビットのドミナントから構成される。インターミッション（ITM）の最初の2ビット以内から送信が開始される。オーバーロードフラグを受信したノードは、即座にオーバーロードフラグを送り返す。これにより、最初のオーバーロードフラグとそれにより送信されるオーバーロードフラグとが重なり、結果としてオーバーロードフラグのビット長は、7ビットとなる。 |
| オーバーロードデリミタ | 8ビットのレセシブから構成され、オーバーロードフレームの区切りを示す。オーバーロードフレームが送信されることにより、バスアイドル開始位置を遅らせることができ、その間に処理が間に合わないノードは処理を行うことができる。 |

### エラーフレーム

送受信中になんらかのエラーを検出したときに、エラーを通知するためのフレーム。CANのハードウェア部分により送信される。

「エラーフラグ」と「エラーデリミタ」から構成される。これらは、ビットスタッフィングルールに違反、または固定フォーム部分を破壊する形で直近の送信を中断させる。

ベクタージャパン　はじめてのCAN/CAN FDより抜粋



|  |  |
| --- | --- |
| エラーフラグ | エラーフラグにはアクティブエラーフラグとパッシブエラーフラグの 2 種類がある。  ・アクティブエラーフラグ: 6 ビットのドミナントレベル  ・パッシブエラーフラグ: 6 ビットのレセシブレベル  6ビット連続で送信することでビットスタッフィングルール違反を発生させる。この違反によって、他ノードもエラーフラグを送信し、結果として、6～12ビットがエラーフラグとしてバスに現れることになる。 |
| エラーデリミタ | エラーデリミタは 8 ビットのレセシブレベル |

### エラー検出時の動作

#### 受信エラー

各ノードのCANコントローラはそれぞれ受信エラーカウンタを持っている。

受信ノードがエラーフラグ（プライマリ）を送信した場合

　→受信エラーカウンタに「8」を加算する。

受信ノードがエラーフラグ（セカンダリ）を送信した場合

　→受信エラーカウンタに「1」を加算する。

受信ノードがエラーなしで、受信を完了した場合

　→受信エラーカウンタから「1」を減算する。

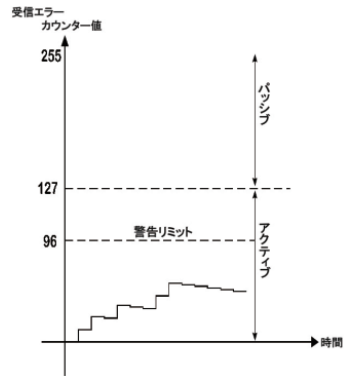
また、各CANコントローラは、それぞれの受信エラーカウンタの値によって状態が定義されている。

0～127：「127」まではノードは『アクティブ』。

96：エラーカウンタ値「96」は、バスに重度の障害があることを示す警告リミットを示す。

127：この値を超えると、ノードは『パッシブ』に移行する。

ベクタージャパン　はじめてのCAN/CAN FDより抜粋



#### 送信エラー

各ノードのCANコントローラはそれぞれ送信エラーカウンタを持っている。

送信ノードがエラーフラグを送信した場合

（ビットエラー、またはアクナレッジエラーの場合）

　→送信エラーカウンタに「8」を加算する。

送信ノードからフレームが正常に送信された場合

　→送信エラーカウンタから「1」を減算する。

例外規定の場合

　→CANコントローラの状態が『パッシブ』となった送信ノードがアクナレッジエラーを検出しても送信エラーカウンタは加算されない。

また、各CANコントローラは、それぞれの送信エラーカウンタの値によって状態が定義されている。

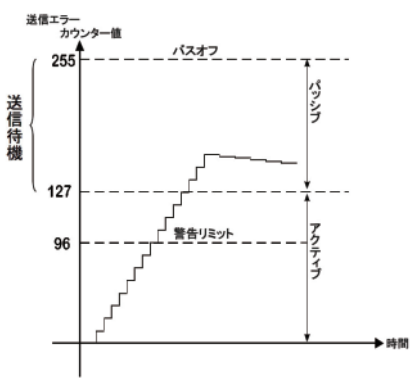
0～96：警告リミット以下の『アクティブ』。エラーカウンタ値が「96」を超えるとCANコントローラは警告（エラーフラグのセット、割り込み）を発する。

97～127：ノードは『アクティブ』。エラーカウンタ値がこのエリアにある場合、バスは重度の障害を持っていることになる。

128～255：ノードは『パッシブ』（ほとんどのCANコントローラはこの領域への変化についてマイコンに報告しない）。また、送信待機を行う。

255：エラーカウンタがこの値への到達後、ノードは『バスオフ』となりバスから切り離される(バスへの送信アクセスが不可となる)。マイコンはこの領域への移行を知ることができる。

ベクタージャパン　はじめてのCAN/CAN FDより抜粋



# 参考資料

CAN関連

・RENESAS　CAN 入門書

・ベクタージャパン　はじめてのCAN/CAN FD

・30分で分かるCAN、設定とデザインのポイント

<https://eetimes.itmedia.co.jp/ee/articles/0912/15/news104.html>

・実装や試験で役立つ物理層から見るCANの仕組み

<https://monoist.atmarkit.co.jp/mn/articles/0809/10/news140.html>