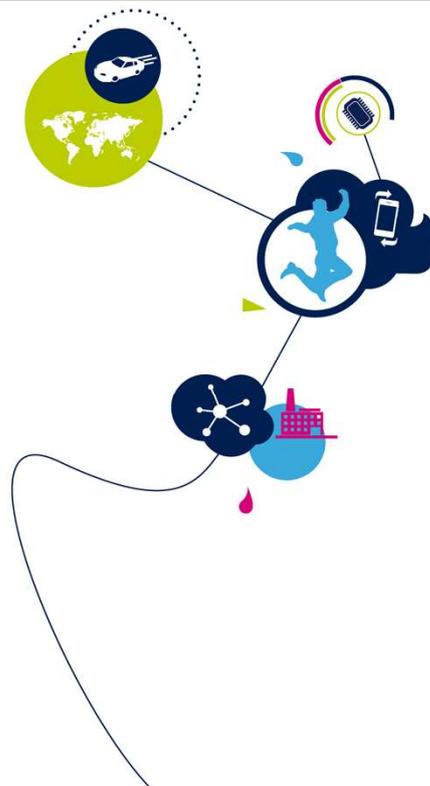


STM32G4 - FD-CAN

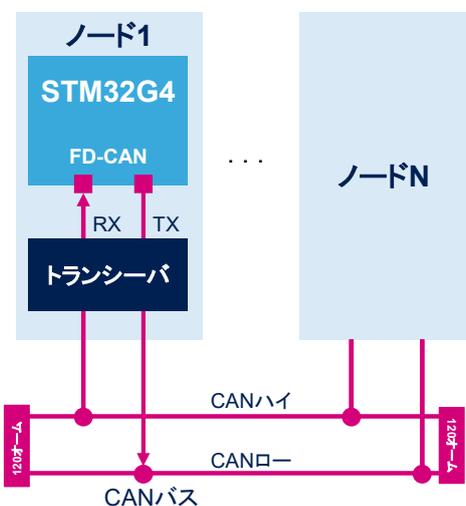
フレキシブルデータレート・コントロールエリアネットワーク・インタフェース

1.0版



こんにちは、フレキシブルデータレートコントロールエリアネットワークインタフェース (FD-CAN) のプレゼンテーションへようこそ。

これは、広くCANネットワークにマイクロコントローラを接続するために使用されているこのインタフェースの主な機能をカバーしています。



- FD-CANコントローラは、2つのピンを介して外部CANトランシーバとの通信インターフェースを提供
- 3つのSTM32G4のFD-CANコントローラ

アプリケーション側の利点

- マルチマスタ・コンセプト
- オブジェクト指向通信
- リアルタイム機能
- 遅延が短いメッセージ転送
- システム全体のメッセージの整合性

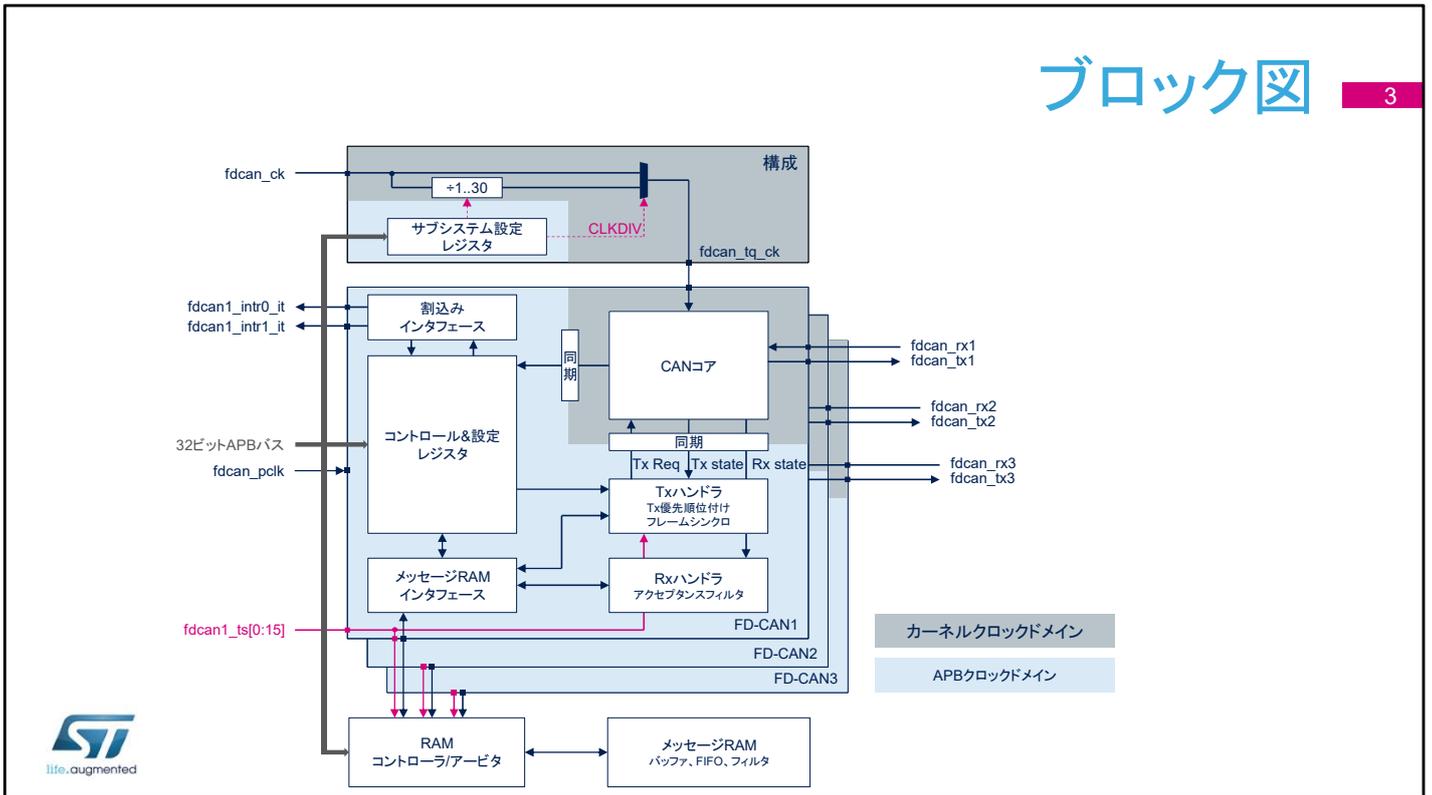


フレキシブルデータレートコントロールエリアネットワークは、マイクロコントローラが同じネットワークバスに接続された外部デバイスと通信する標準シリアル差動バスブロードキャストインターフェースです。

FD-CANインターフェースは高度に設定可能で、ノードは2本のワイヤだけで簡単に接続できます。

アプリケーションは、メッセージ優先度、オブジェクト指向通信(ノード・アドレッシングは不要だが、コンテンツ識別)、メッセージ転送遅延が少なく、システム全体のメッセージ整合性(エラー検出と管理メカニズム)を備えたマルチマスタ概念の恩恵を受けることができます。

STM32G4マイクロコントローラは3つのFD-CANコントローラを内蔵します。



CANサブシステムは、FD-CAN1、FD-CAN2、およびFD-CAN3という名前の3つのFD-CANコントローラをサポートします。

これらの3つのコントローラは共有されているクロックユニットとRAMを除いて独立しており、同じ機能を備えています。

これらのコントローラは、最大ビットレート1メガビット/秒の基本拡張CANプロトコルバージョン2.0AおよびBと、最大64データバイト、最大8メガビット/秒のデータビットレートのCAN FDプロトコルバージョン1.0の両方をサポートしています。

CANコアには、プロトコルコントローラと送受信シフトレジスタが含まれています。

すべてのISO 11898-1: 2015プロトコル機能を処理し、11ビットと29ビットの両方の識別子をサポートしています。

TxハンドラはメッセージRAMからCANコアへのメッセージ転送を制御し、RxハンドラはCANコアから外部メッセージRAMへの受信メッセージの転送を制御します。

APBバスインタフェースとCANコアカーネルクロックの2つのクロックドメインが実装されているため、これら2つのドメイン間で同期ブロックが必要です。共有0.8KバイトメッセージRAMメモリが使用可能です。

このRAMは、フィルタ、バッファ、およびFIFOを格納するために使用されます。

名前	タイプ	説明	
fdcan_ck	デジタル入力	CANサブシステム・カーネル・クロック入力	
fdcan_pclk	デジタル入力	CANサブシステムAPBインタフェース・クロック入力	
fdan_intr0_it	デジタル出力	FD-CAN割込み0	
fdan_intr1_it	デジタル出力	FD-CAN割込み1	
fdcan_ts[0:15]	-	外部タイムスタンプ・ベクタ ・ このタイムスタンプは、FD-CANブロックに含まれるタイマによって提供される	
FDCAN_RX	デジタル入力	FD-CAN受信ピン	オルタネート・ファンクションGPIO
FDCAN_TX	デジタル出力	FD-CAN送信ピン	
APBインタフェース	デジタル入出力	コンフィギュレーション、制御、RAMアクセス用の複数のPSELを備えたシングルAPBスレーブ・インタフェース	



CANサブシステムのI/O信号とピンについて、この表で詳しく説明しています。

2つのクロックがFD-CANユニットに提供されます。

- fdcan_ck、ビットレートを取得するために使用されるカーネルクロック
- fdcan_apb (メモリマップレジスタおよびメッセージRAMにアクセスするために使用されるAPBクロック)

2つの割込み出力により、FD-CANユニットはCortex-M4プロセッサにイベントを報告します。

FD-CANユニットでは、外部16ビットのタイムスタンプ入力ポートを使用して、メッセージの送信または受信のタイムスタンプを設定することができます。

このタイムスタンプは、FD-CANブロックに含まれるタイマによって提供されます。

FDCAN_RXおよびFDCAN_TXは、トランシーバに接続されている必要があります。

最後に、APBスレーブインタフェースは内部的に3つの部分に分割されており、それぞれが専用のチップセレクト:コンフィギュレーション、コントロール、RAMアクセスを有しています。

- CANプロトコル・バージョン2.0A/B、およびCAN FDプロトコル・バージョン1.0
 - ISO 11898-1:2015に準拠
 - 最大64データ・バイトをサポートするCAN FD
 - ビットレート:
 - アービトレーション・ビットレートは最大1メガビット/秒
 - データ・ビットレートは最大8メガビット/秒
- サポート
 - コントローラごとに2つのマスク可能な割り込みを準備
 - パワーダウン・サポート
 - CANエラー・ログ
 - AUTOSARおよびJ1939サポート
 - 優先度の高いメッセージを受信した場合には、別の信号を送信



FD-CANコントローラは、CANプロトコルバージョン2.0パートA、BおよびISO 11898-1:2015に準拠しており、最大64データバイトのCAN FDプロトコルをサポートしています。

FDモードの最大ビットレートは8メガビット/秒です。

各コントローラは、24個の完全に設定可能な割り込みフラグを持つ、2つの独立した、マスク可能な割り込みもサポートしています。

コントローラにはパワーダウンモードがあります。

エラーロギング、AUTOSAR、J1939、および優先度の高いメッセージの受信時の個別のシグナリングをサポートしています。

- メッセージRAMには、次のものが含まれる
 - 3つのペイロードの2つの受信FIFO
 - 設定可能な送信FIFO/3つのペイロードのキュー (ペイロードごとに最大64バイトまで)
 - 設定可能な送信イベントFIFO

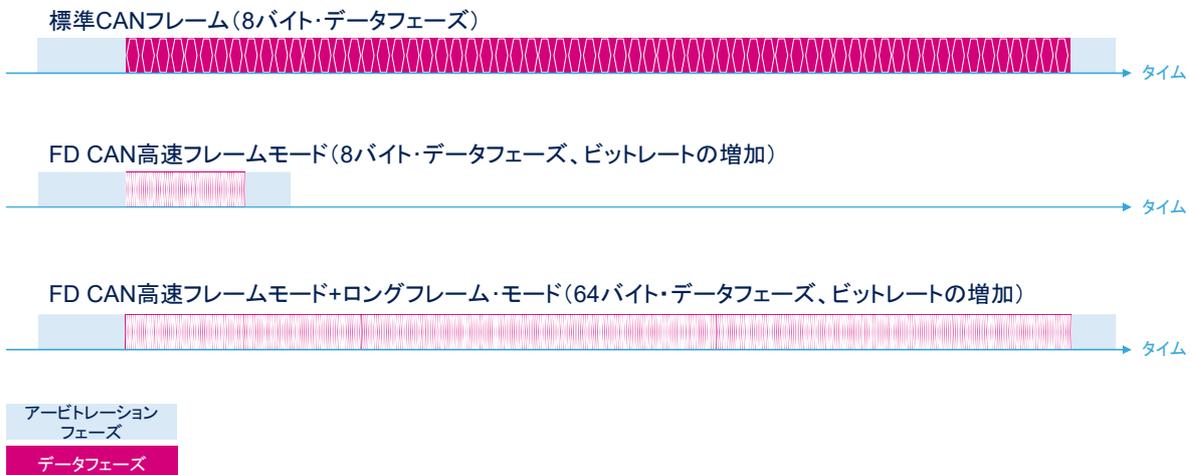


受信したメッセージは、2つのRx FIFOそれぞれに最大3個まで保存できます。受入フィルタは、使用するFIFOを選択します。送信する最大3つのメッセージを、Tx FIFOまたは3つの独立したTxバッファとして構成されたメッセージRAMの一部として保存できます。

Rx FIFOとTx FIFOまたはTxバッファの各エントリは、最大メッセージサイズである64バイトのペイロードをサポートしています。TxイベントFIFOには、対応するメッセージIDと一緒にTxタイムスタンプが保存されます。

フレキシブル・データレート(FD)CAN拡張機能

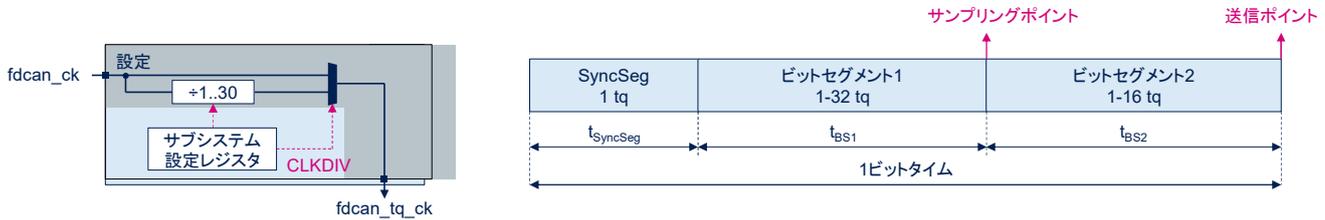
7



FD-CANプロトコルには、次の2つのバリエーションがあります。

1. ロングフレームモード(LFM)は、CANフレームのデータフィールドが8バイト、最大64バイトの長さになる可能性があります。
2. 高速フレームモード(FFM)は、CANフレームの制御フィールド、データフィールド、CRCフィールドが、フレームの先頭と末尾に比べて高いビットレートで送信されるモードです。この高いデータレートは、通常、毎秒8メガビットです。

高速フレームモードは、ロングフレームモードと組み合わせて使用することができます。



- ビットタイミング・ロジックは、シリアル・バス・ラインをモニタし、サンプルポイントのサンプリングと調整を行う
 - $\text{Time Quanta (tq)} = (\text{FDCAN_NBTP}[\text{NBRP}] + 1) * t_{\text{fdcan_tq_clk}}$
 - $\text{ビットタイム} = t_{\text{SyncSeg}} + t_{\text{BS1}} + t_{\text{BS2}}$
 - tBS1とtBS2の長さはプログラム可能



ビットタイミングロジックはシリアルバスラインを監視し、スタートビットエッジで同期をとり、次のエッジで再同期をとることでサンプルポイントのサンプリングと調整を行います。

タイム量子は、コンフィグレーションユニットから取得した基本タイミング単位であり、FDCAN_NBTPレジスタにプログラムされた1から512までの比率を乗じたTFDCAN_TQ_CKに等しくなっています。

ビットタイムは、同期セグメント、ビットセグメント1、ビットセグメント2の3つのセグメントに分割されます。

これらのセグメントの各々は、タイム量子の整数倍である。BS1とBS2の持続時間は、公称ビット時間とデータビット時間のために独立してプログラム可能です。

データ・ビットタイムは、FDモードで動作するときに適用され、データは高データ・レートで送信されます。

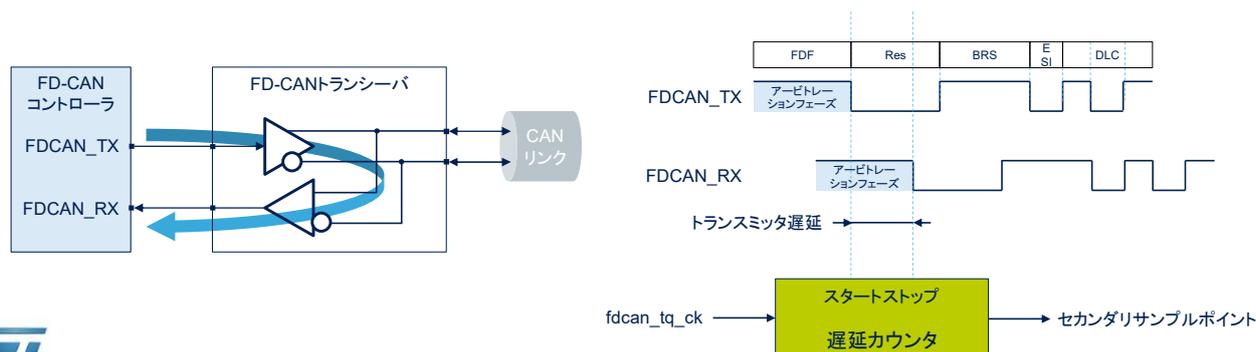
オンチップバスクロックを調整するために、CANコントローラはビットの長さを整数クアンタ分だけ短くしたり長くしたりすることができます。

これらのビット時間調整の最大値は、同期ジャンプ幅(SJW)と呼ばれ、1から4までのTime Quantaでプログラム可能です。

トランスミッタ遅延補償

9

- FD-CANコントローラは、CANトランシーバのループ遅延を補償するための遅延補償メカニズムを実装しており、それにより特定のCANトランシーバの遅延に依存しないFD-CANデータフェーズの間、より高いビットレートでの伝送を可能にする



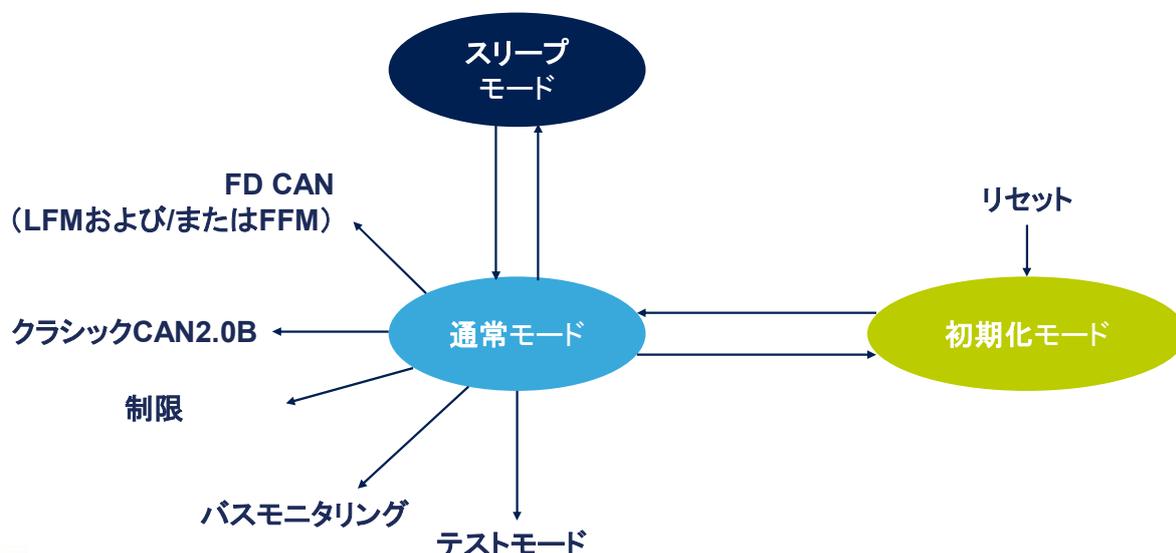
トランスミッタ遅延補償は、データビットの時間を送信遅延よりも短くする構成を可能にします。

DBTPレジスタのビットTDCを設定することで有効になります。受信ビットはセカンダリサンプルポイントで送信ビットと比較されます。

SSP位置は、トランシーバを経由してFDCAN送信出力ピンFDCAN_TXから受信入力ピンFDCAN_RXまでの測定された遅延の合計に送信遅延を加えたものとして定義されます。

補正オフセットを使用します。

トランスミッタ遅延補償オフセットは、受信ビット内のSSPの位置を調整するために使用され、例えば、データフェーズのビット時間の半分の時間が使用されます。



FD-CANには、初期化、通常、スリープの3つの主要な動作モードがあります。

ハードウェアリセット後、FD-CANはソフトウェアを介して初期化モードに入ります。

このモードでは、次の手順を実行します。

- ペリフェラルの設定（ビットタイミングとRAMの割り当て）を行う必要があります。
‘ビットタイミング’の設定では、レートが設定され、実際のシリアルバスラインに合わせてサンプリングポイントが調整されます。
- CANコントローラは、11連続の劣性ビット（Consecutive recessive bit）を待つことによってCANバスと同期します。

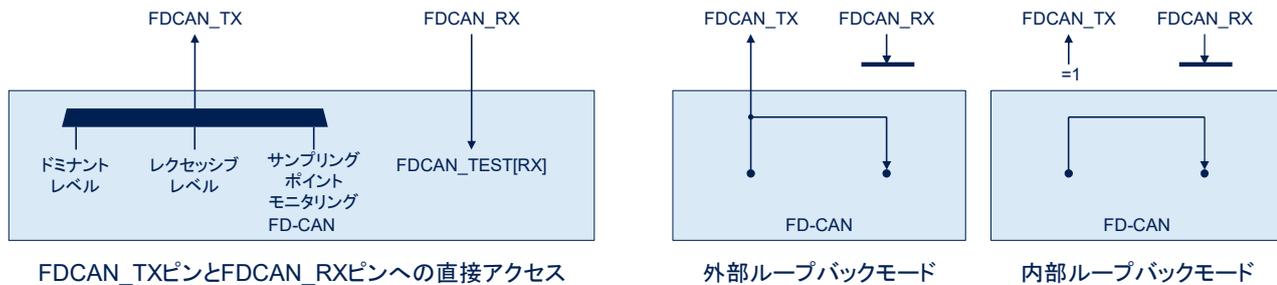
CANが通常モードの場合、ユーザは異なる特定のサブモードを選択できます。

- CAN仕様2.0Bと互換性のあるクラシックCANモード
- FD CANモード：ロングフレームモードとファストフレームモードがあり、それぞれLFMとFFMと名付けられています。
- 制限モード：コントローラはデータフレームを受信して確認することはできませんが、フレームは送信しません。これは、異なるCANビットレートに適応するアプリケーションで使用することができます。
- バスモニタリングモード：コントローラはデータフレームを受信することができます（ただし、データフレームを確認することはできません）。これは、支配的なビットの送信によってそれに影響を与えることなく、CANバス上のトラフィックを分析するために使用することができます。
- テストモードの詳細は次のスライドを参照してください。

CPUからの要求により、バスアイドル状態を検出した場合は、低消費電力で動作するスリープモードになります。

FD-CANテスト・モード

5



FDCAN_TESTレジスタへの書き込みアクセスを有効にするには、CCCRレジスタのTESTビットを1に設定する必要があります。

テストモードでは、ソフトウェアはFDCAN_TXピンの状態を制御し、FDCAN_RXの状態を読み出すことができます。

FDCAN_TESTレジスタを介して、ソフトウェアはFDCAN_TXの出力を制御することができます:ドミナントレベル、レクセッシブレベル、サンプルポイントを監視します。

FDCAN_RXピンの実際の値は、FDCAN_TESTレジスタのRXビットから読み出すことができます。

両方の関数は、CANバスの物理層をチェックするために使用することができます。

これらのテストモードは、生産テストまたは自己テストのみに使用してください。

さらに、FD-CANコントローラは、FDCAN_TESTレジスタとFDCAN_CCCRレジスタの制御ビットを介して入力される2つのループバックモードをサポートしています。

外部ループバックモードでは、FDCANは、自身の送信メッセージを受信メッセージとして扱い、(それらがアクセプタンスフィルタリングに合格した場合)それらをRx FIFOに格納します。このモードは、ハードウェアのセルフテスト用に提供されます。

外部から独立しているため、FDCANは、アクノレッジ・エラー(のアクノレッジ・スロットでサンプリングされたレクセッシブビット)を無視します。

データ/リモートフレーム)をループバックモードで使用することができます。

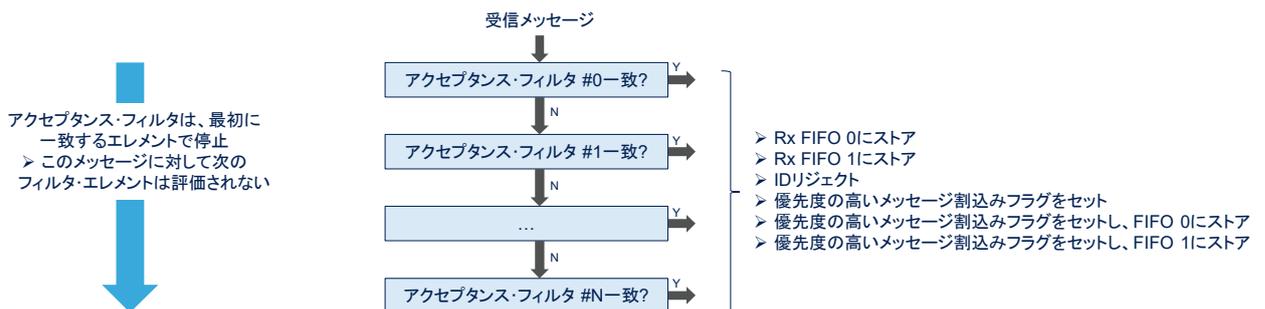
内部ループバックモードは、セルフテストに使用することができます。

つまり、FDCAN_TXとFDCAN_RXピンに接続された実行中のCANシステムに影響を与えることなく、FDCANをテストすることができます。このモードでは、FDCAN_RXピンはFDCANから切り離され、FDCAN_TXピンはレクセッシブレベルに保持されます。

アクセプタンス・フィルタ

12

- FD-CANコントローラは、メッセージRAMに配置された標準識別子用と拡張識別子用の2つのフィルタを設定することが可能
 - 各セットには28個のエントリがある:28x11ビット・フィルタ・エントリ+28x29ビット・フィルタ・エントリ
 - これらのフィルタは、Rx FIFO 0またはRx FIFO 1に割り当てることが出来る



アクセプタンス・フィルタは、最初に一致するエレメントで停止
➢ このメッセージに対して次のフィルタ・エレメントは評価されない



FD-CANコントローラは、標準の11ビット識別子用と29ビット拡張識別子用の2つのアクセプタンスフィルタを設定することができます。

各フィルタエレメントは、受入れまたは拒否フィルタリング用に設定可能

- 各フィルタエレメントは個別に有効/無効にできます
- ・フィルタが順番にチェックされ、最初に一致するフィルタエレメントで実行が停止します。

ソフトウェアはアクティブなフィルタインスタンスの数を設定し、最大は28です。

完全な識別子を受け取った後、アクセプタンスフィルタが開始されます。

アクセプタンスフィルタ処理が完了し、一致するRx FIFOが見つかった場合、メッセージハンドラは、受信したメッセージデータを一致するRx FIFOに32ビット部分で書き込み始めます。

- 各フィルタ・エレメントは次のように設定可能
 - レンジ・フィルタ(から～まで)
 - メッセージID∈標準ID用 [SF1ID, SF2ID]
 - メッセージID∈拡張ID用 [EF1ID, EF2ID] EFT=0の場合
 - メッセージID & XIDAM[EIDN]∈拡張ID用 [EF1ID, EF2ID] EFT=11の場合
 - 1つまたは2つの専用IDのフィルタ
 - メッセージID = 標準ID用 SF1ID ||メッセージID = SF2ID
 - メッセージID & XIDAM[EIDN] = 拡張ID用 EF1ID ||メッセージID & XIDAM[EIDN] = EF2ID
 - クラシック・ビット・マスク・フィルタ
 - メッセージID & SF2ID == 標準ID用 SF1ID & SF2ID
 - メッセージID & XIDAM[EIDN] & EF2ID == 拡張ID用 EF1ID & EF2ID



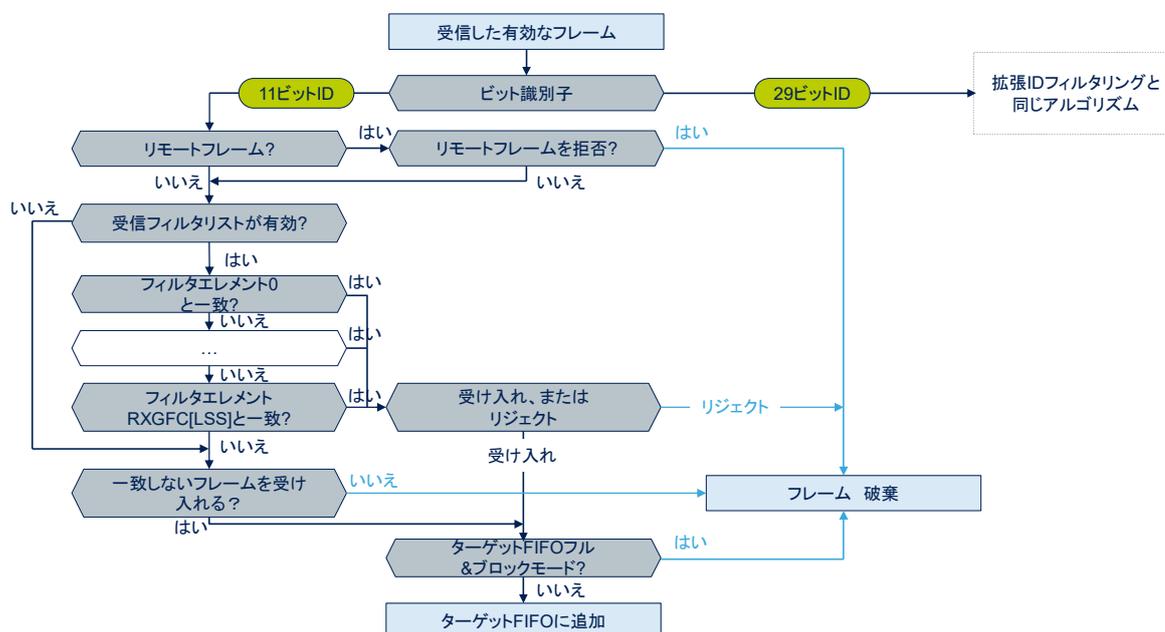
各フィルタエレメントは次のように設定できます。

- レンジフィルタ(から～まで)
- 1つまたは2つの専用IDのフィルタ
- クラシックビットマスクフィルタ

拡張IDについては、フィルタリストを実行する前に、受信した識別子と拡張ID ANDマスク(XIDAM)をANDします。

特定のメッセージIDに対してフィルタリングを行うには、SF1ID=SF2ID、EF1ID=EF2IDでフィルタエレメントを設定する必要があります。

メッセージIDフィルタリング



このアルゴリズムは、標準IDで受信したフレームのフィルタリングシーケンスを記述します。

拡張IDで受信したフレームを処理するために同様のアルゴリズムが使用されますが、これら2つのアルゴリズムの設定は独立して行われます。

最初のステップは、リモートフレームの受け入れまたは拒否することです。

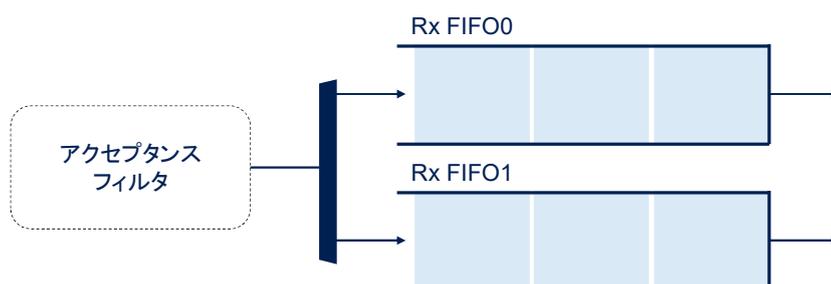
その後、受信側リストが無効になると、フィルタエレメントはバイパスされます。それ以外の場合、最初に一致するエレメントは、フレームが受け入れられるか拒否されるかを決定します。

受信側フィルタが無効になっている場合、またはフィルタエレメントが一致していない場合、フレームは受け入れられるか拒否されます。

最後に、フレームが受け入れられ、対象のRx FIFOがいっぱいでない場合、このフレームはRx FIFOに追加されます。

Rx FIFOがいっぱいでブロッキングモードが選択されると、フレームは破棄されます。

- 2つのRxのFIFOは、ハードウェアとソフトウェアの間で同じ同期メカニズムを実装
 - FD-CANコントローラがRx FIFO*n*に新しいメッセージを書き込むと、読取り専用のPutインデックスフィールドが更新される
 - ソフトウェアは、Getインデックスを更新するために、読み取ったメッセージの数を示す必要がある



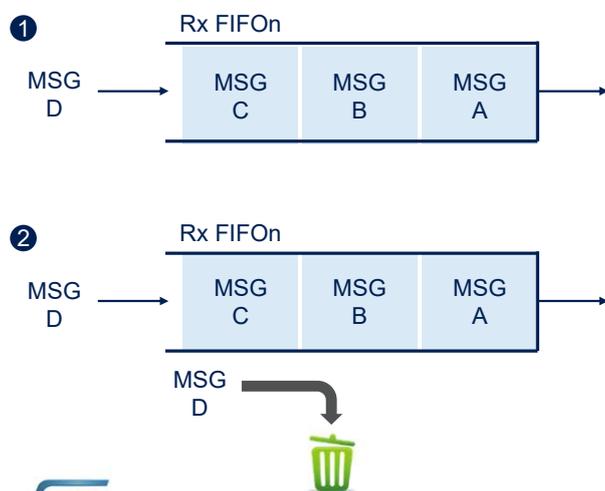
Rx FIFO 0およびRx FIFO 1は、それぞれ最大3つのエレメントを保持できます。

アクセプタンスフィルタリングを通過した受信メッセージは、一致するフィルタエレメントによって構成されたRx FIFOに転送されます。

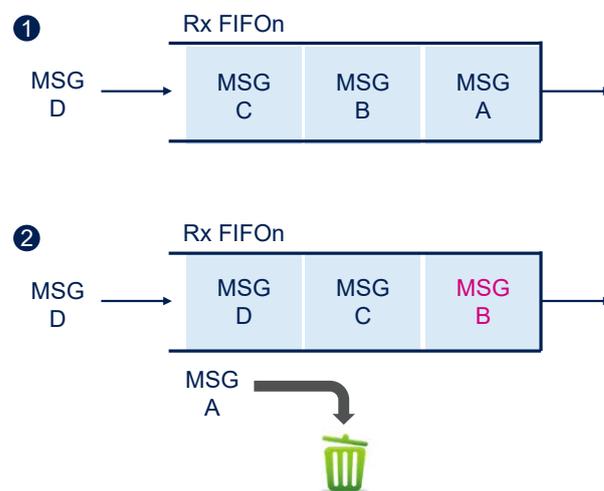
読取り専用レジスタFDCAN_RXF0SおよびFDCAN_RXF1Sは、以下の情報を提供します。

- Putインデックスの位置
- Getインデックスの位置
- 保留中のメッセージの数
- FIFOフルコンディション

Rx FIFOブロッキング・モード



Rx FIFO上書きモード



Rx FIFOブロッキングモードは、Rx FIFOのデフォルトの動作モードです。

Rx FIFOがフルの状態に達すると、少なくとも1つのメッセージが読み出され、Rx FIFOゲットインデックスがインクリメントされるまで、対応するRx FIFOにはそれ以上のメッセージは書き込まれません。

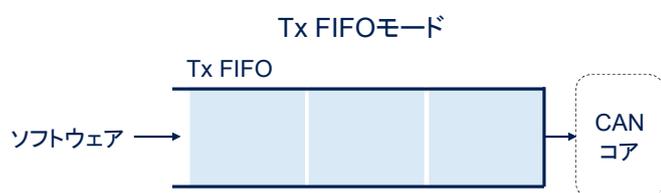
対応するRx FIFOが満杯の状態にメッセージを受信した場合、このメッセージは破棄され、メッセージロスト状態が通知されます。

Rx FIFOオーバーライトモードでは、Rx FIFOのフルコンディションがシグナルされると、右図のように、最も古いメッセージが破棄され、次のメッセージが受け入れられます。

PutとGetインデックスは共に1つずつインクリメントされます。

Tx FIFO/Txバッファ操作

17

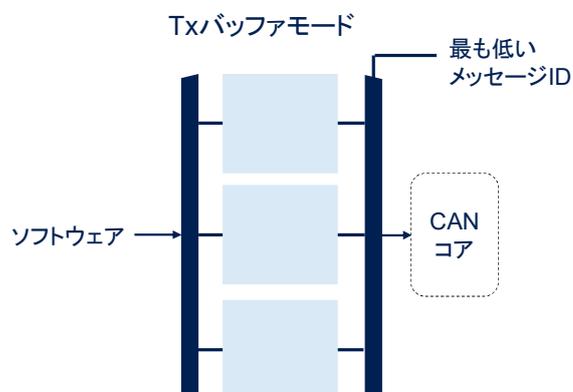


- ソフトウェアは、Putインデックスを更新するために、FIFOに書き込んだメッセージの数を示す必要がある
- FD-CANコントローラがメッセージを読み取ると、Getインデックスが更新される



送信キャンセル

N/A



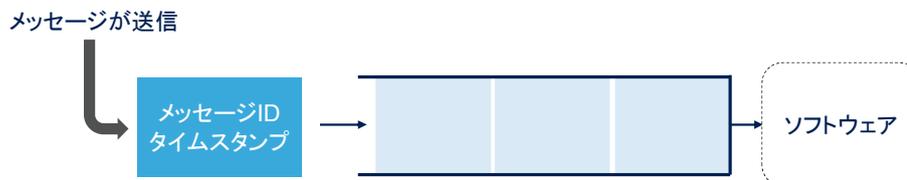
- 3つのバッファは互いに独立して処理される
 - ソフトウェアは、次のメッセージが書き込まれるバッファを選択する
 - FD-CANコントローラは、使用可能なバッファを示す

送信キャンセル

✓

最大3つのTxバッファをメッセージ転送用に設定できます。Tx FIFOモードは、ソフトウェアによって準備された順序ですべてのメッセージが送信されるように選択されます。または、3つのメッセージバッファが互いに独立して処理されるTxキューモードが選択されます。Txキューに格納されたメッセージは、最も優先順位の高いメッセージから送信されます。FD-CANコントローラは、送信キャンセルをサポートします。要求されたTxキューバッファからの送信をキャンセルするには、ソフトウェアはレジスタTXBCRの対応するビット位置に1を書き込む必要があります。送信キャンセルは、Tx FIFO操作を想定していません。

- Txイベント処理をサポートするために、FD-CANはTxイベントFIFOを実装
 - FD-CANコントローラがCANバスでメッセージを送信した後、メッセージIDとタイムスタンプはTxイベントFIFO要素に格納される



Txイベント処理をサポートするために、FD-CANはTxイベントFIFOを実装しました。

TxイベントFIFOの目的は、転送ステータス情報の処理を送信メッセージ処理から切り離す方法です。

Txバッファは送信されるメッセージのみを保持し、送信ステータスはTxイベントFIFOに個別に格納されます。

この機能には、特に動的に管理された送信キューを操作する場合、転送が成功した直後に新しいメッセージにTxバッファを使用できるという利点があります。

Txバッファを上書きする前に、転送ステータス情報をTxバッファから保存する必要はありません。

TxイベントFIFOがいっぱいの際にTxイベントが発生した場合、このイベントは破棄され、割り込みフラグが設定されます。

- コントローラごとに準備された2本の独立した割込みライン
 - 以下の24個の設定可能な割込みフラグとは別にイネーブル/ディセーブルの設定可能

割込みイベント	割込みイベント
予約アドレスへのアクセス	TXイベントFIFOフル
データ・フェーズの Protokol・エラー	TXイベントFIFO新しいエントリ
アービトレーション・フェーズでの Protokol・エラー	TX FIFOエンプティ
ウォッチドッグ・タイムアウト	送信のキャンセルの完了
Bus_Off	トランスミッション完了
警告ステータス	優先度の高いメッセージ
エラー・パッシブ	RX FIFO1メッセージ喪失
エラー・ログのオーバーフロー	RX FIFO1フル割込み
タイムアウトの発生	RX FIFO1新しいメッセージ
メッセージRAMアクセスの失敗	RX FIFO0メッセージ喪失
タイムスタンプ・ラップアラウンド	RX FIFO0フル割込み
TXイベントFIFOエレメントの喪失	RX FIFO0新しいメッセージ



FD-CANコントローラペリフェラルは、2つの独立した割込みラインを提供します。
 このスライドでは、可能な割込みイベントの完全なリストを示しています。

モード	説明
RUN	有効
SLEEP	有効 • ペリフェラル割込みにより、デバイスはSLEEPモードを終了
低電力RUN	有効
低電力SLEEP	有効 • ペリフェラル割込みにより、デバイスは低電力SLEEPモードを終了
STOP0/STOP1	
STANDBY	利用できない
SHUTDOWN	



FD-CANサブシステムの低電力構成モードの概要を次に示します。

デバイスは、STOPまたはSTANDBYモードで通信を実行できません。

ペリフェラルがSTOPモードまたはSTANDBYモードに入る前に、すべてのCANトラフィックが完了していることを確認することが必要です。

- CPUコアがデバッグ・モードの間は、次のようになる
 - FD-CANは通常の機能モードのまま
 - 「読取り時にリセット」または「読取り時に設定」のタイプのレジスタは無効になり、それらを読んでも値には影響しない



CPUコアがデバッグモード(ブレークポイントで停止)の間、

- FD-CANは通常の機能モードを維持しています。特に、受信は通常通り継続しており、これによりFIFOやバッファが一杯になると受信オーバーランエラーが発生することがあります。
- 「読取り時にリセット」または「読取り時に設定」のタイプのレジスタは無効になり、それらを読んでも値には影響しません。

- 次のペリフェラルを参照してください。
 - CANクロック制御と有効化/リセットの詳細については、リセットとクロック・コントローラ(RCC)
 - FD-CANの割り込みのマッピングの詳細については、ネスト化されたベクタ割り込みコントローラ(NVIC)
 - FD-CANの入カピンと出力ピンの詳細については、汎用I/O(GPIO)
 - CPUが停止したときのFD-CANの動作に関する詳細は、デバッグサポート(DBG)



life.augmented

詳細については、FD-CANの動作に影響を与える可能性のあるこれらのペリフェラルのトレーニングを参照してください。

-リセットおよびクロックコントローラ(RCC) : CANクロック制御およびイネーブル/リセットの詳細については、「リセットおよびクロックコントローラ(RCC)」を参照してください。

-FD-CANの割り込みのマッピングの詳細については、「割り込み」を参照してください。

-汎用I/O(GPIO)FD-CANの入カピンと出力ピンの詳細については、「汎用I/O(GPIO)」を参照してください。

-デバッグモードでのFD-CANの動作の詳細については、「デバッグサポート(DBG)」を参照してください。

- 詳細については、以下のリソースを参照してください。
 - アプリケーションノートAN3154: Description of the CAN protocol used in the STM32 boot loader
 - アプリケーションノートAN3364: Migration and compatibility guidelines for STM32 microcontroller applications
 - Web(接続例、使用可能な監視ツールなど)



CANのトピックをカバーするアプリケーションノートは、www.st.comで入手できます。

CANインタフェースの詳細については、CAN通信プロトコルやバス・モニタリング・ツールについて説明しているさまざまなWebページを参照してください。

多くのデジタルオシロスコープは、CANバスを介して送信されたデータの直接読み取りと解析をサポートしています。