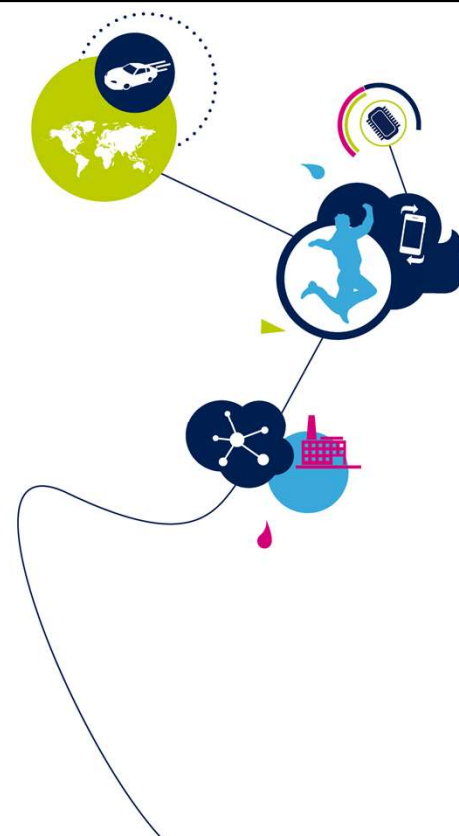
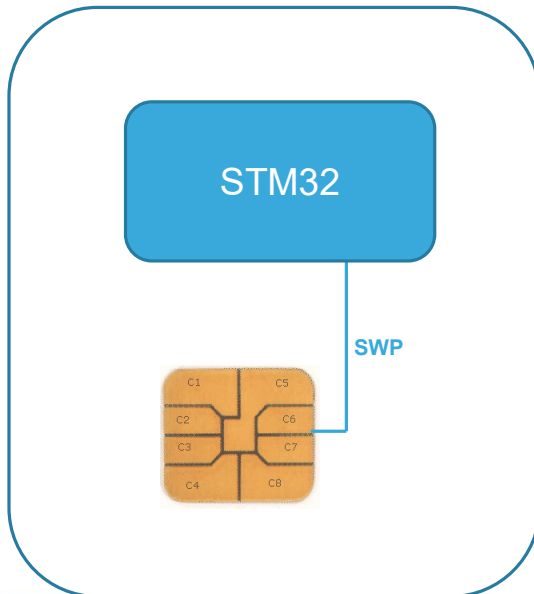


STM32H7 - SWPMI

単線(シングルワイヤ)プロトコル・マスタ・インタフェース
1.0版



こんにちは、単線プロトコル・マスタ・インタフェース(SWPMI)のプレゼンテーションへようこそ。これは、マイクロコントローラにスマートカードを接続するために使用されるこのインタフェースの主な機能について説明します。



- マスタ・モードで、ETSI TS 102 613標準で定義されたシングルワイヤ・プロトコルに従って、全二重シングルワイヤ通信インタフェースを実装

アプリケーション側の利点

- SWPトランシーバはSTM32に内蔵
- スマートカードのC6コンタクトでの単線全二重通信は、100キロビット/秒から2メガビット/秒まで

SWPMIIはSTM32の製品の内部に統合され、マスタモードで、ETSI TS102613標準で定義された単線プロトコルに準拠して、全二重シングルワイヤ通信インタフェースを実装しています。STM32は、SWPトランシーバを実装しています。アプリケーションは、2メガビット/秒までの全二重通信のためのスマートカードへの容易な単ピン接続を提供します。

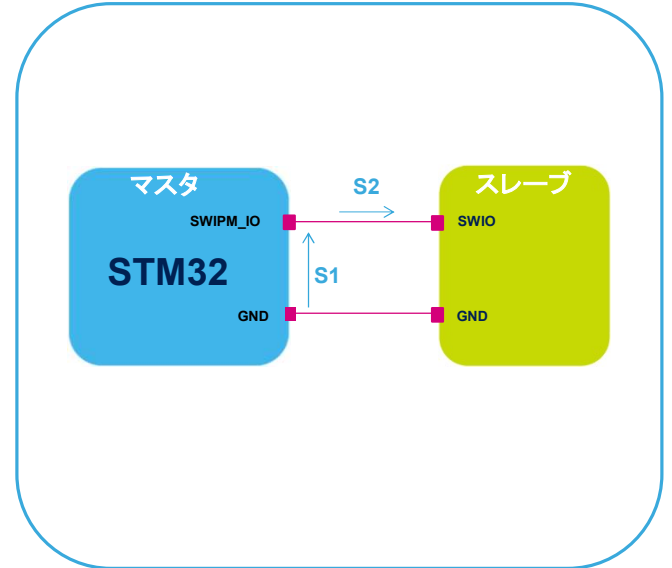
- 3つの送受信モード
 - No Software Buffer (no DMA)
 - 割込み、またはポーリング・モード
 - Single Software Buffer (DMA)
 - フレームの送信または受信中にソフトウェアの介入なく、最後のフレームのみ必要
 - Multi Software Buffer (サーキュラ・モードでのDMA)
 - いくつかのフレームは、ソフトウェアの介入なしに処理することが可能
- Class B (3V)とClass C(1.8V)の電圧を供給



STM32製品内蔵SWPMIは、DMAありなしなど後に詳細を説明する、3つの動作モードを提供しています。STM32は、クラスBおよびクラスCの両方の動作電圧の供給をサポートしています。

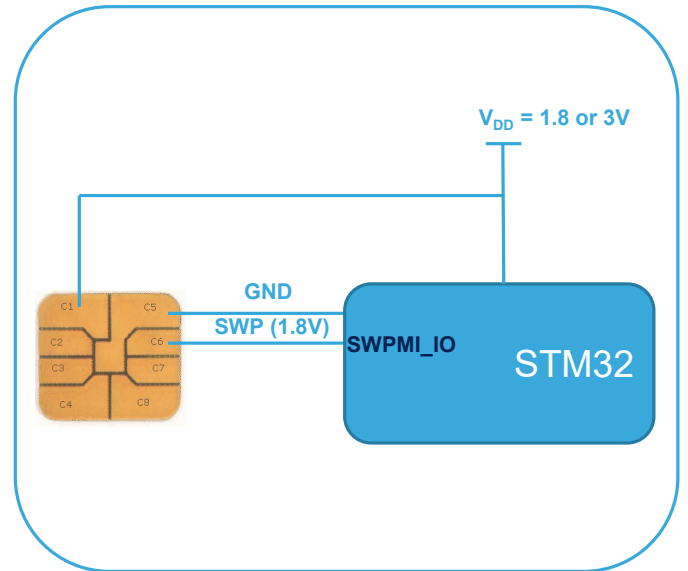
単線全二重通信

- S1信号は、マスタからスレーブに、電圧ドメインでデジタル変調(LまたはH)で送信(0/1.8V)
- S2信号はスレーブからマスターに、デジタル変調(LまたはH)された電流ドメインで送信(0/800 μ A)



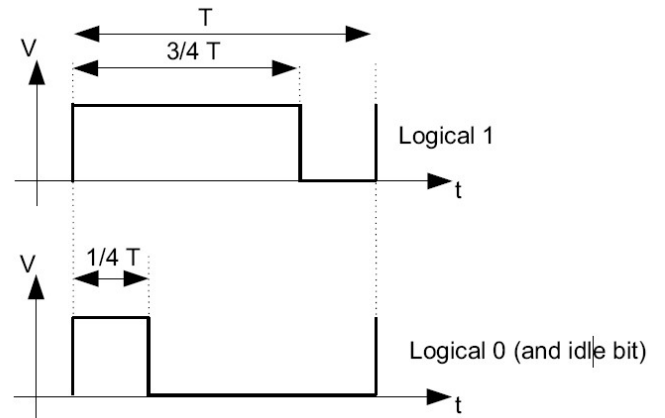
SWPは以下の原理による単線の全二重です。S1信号は、マスタからスレーブへの電圧ドメインで送信されます。S2信号は、スレーブからマスタへ電流ドメインで送信されます。

- STM32で以下のどちらかを選択する必要がある:
 - Class B: VDD = 3V
 - Class C: VDD = 1.8V
- SWPピンの電圧はSTM32によって常に1.8Vに設定
 - Class Bが選択されている場合、SWPMI_IOレジスタ内部レギュレータは、1.8Vに電圧を調整



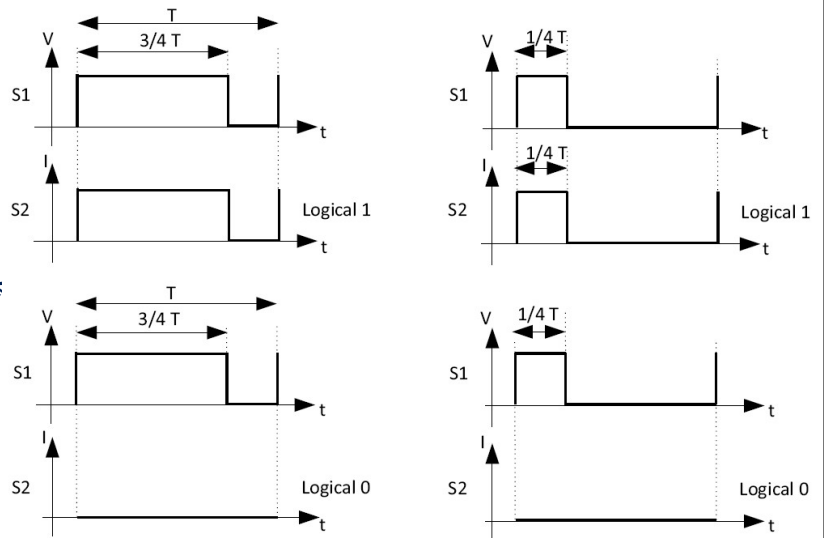
電源電圧やクラスは、ソフトウェアの初期化中にSTM32で選択する必要があります。VDDが3Vである場合、STM32 SWPMI_IOレジスタ内部専用の1.8Vレギュレータは、SWPの電圧を調整するために使用されます。

- S1信号: マスタ(STM32)からスレーブ(smartcard)へ
- S1はPWMでビット・コーディング
 - 送信クロック
 - データ

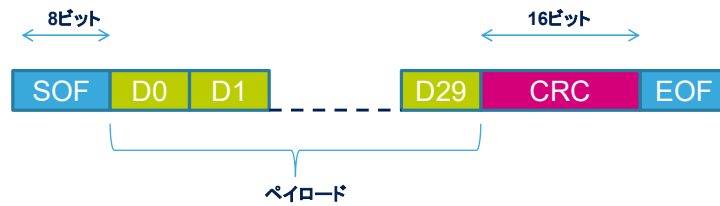


S1信号は、STM32マスタからスマートカードスレーブに、送信されます。S1コード論理0(アイドルビット)に25%のデューティサイクル、S1コード論理1に75%のデューティサイクルを使う。S1信号の周波数は、送信クロックを決定します。

- S2信号:スレーブから(smartcard) マスターへ(STM32)
- マスタがHigh状態としてS1を送信すると、スレーブは電流を引く(High)、または何もしない(Low)のいずれかでS2信号を送信



S2信号はスレーブのスマートカードによって、マスタのSTM32に送信されます。スレーブはS1がHighの間、電流を引っ張ることによって、論理1を送る。もし、S1がHighの間、スレーブが電流を引っ張らなければ、それは論理0です。



- Start Of Frame (SOF) = 7Eh = 01111110b
- End Of Frame (EOF) = 7Fh = 01111111b
- ペイロードは1から30のデータバイトを含む
- ビットスタッフィング
 - 論理値1を持つビットが5つ続いた場合は、値0を持つ追加のスタッフィング・ビットを挿入する
 - ビット・スタッフィングでSOFとEOFを区別することができる
- 16ビットのCRC ($X^{16}+X^{12}+X^5+1$)をEOFの前に挿入する



SWPフレームは16進形式で7Eバイトで符号化されたフレームの開始フィールドで始まり、16進形式で7Fバイトで符号化されたフレームの終端フィールドで終わります。ペイロードは、データの1から30バイトが含まれています。プロトコルは、ビットスタッフィングを実装しています。余分なビットは、1が5つ連続した場合に挿入されます。これがスタートとエンドフレームがペイロードバイトと区別されることを保証します。データの整合性は、16ビットの巡回冗長検査(CRC)によって保証されています。

SWPMIによるSWPフレーム処理

9

ソフトウェアはペイロードフィールドのみ管理



- SWPMIは以下を自動的に行う:
 - Start of Frame (SOF)挿入/削除
 - End of Frame (EOF)挿入/削除
 - スタッフィングビット挿入/削除
 - CRC-16の計算、生成、確認



SWPMIが自動的に、スタートフレーム、ストップフレーム、スタッフィングビット、CRCを処理します。この方法では、ソフトウェアはペイロードデータだけを管理する必要があります。

- トランジションのアクティブ化



- トランジションの非アクティブ化



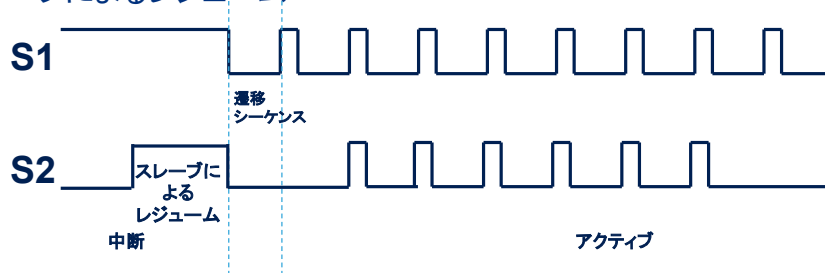
いくつかのバス状態は、SWPバスのために定義されています。非アクティブ状態では、S1信号がローレベルです。すべての通信を開始する前に、マスタはサスペンド状態でSWPを設定するためにハイレベルにS1信号を上げる必要があります。通信が不要になったら、SWPは、マスタによって非アクティブにすることができます。

• レジューム遷移

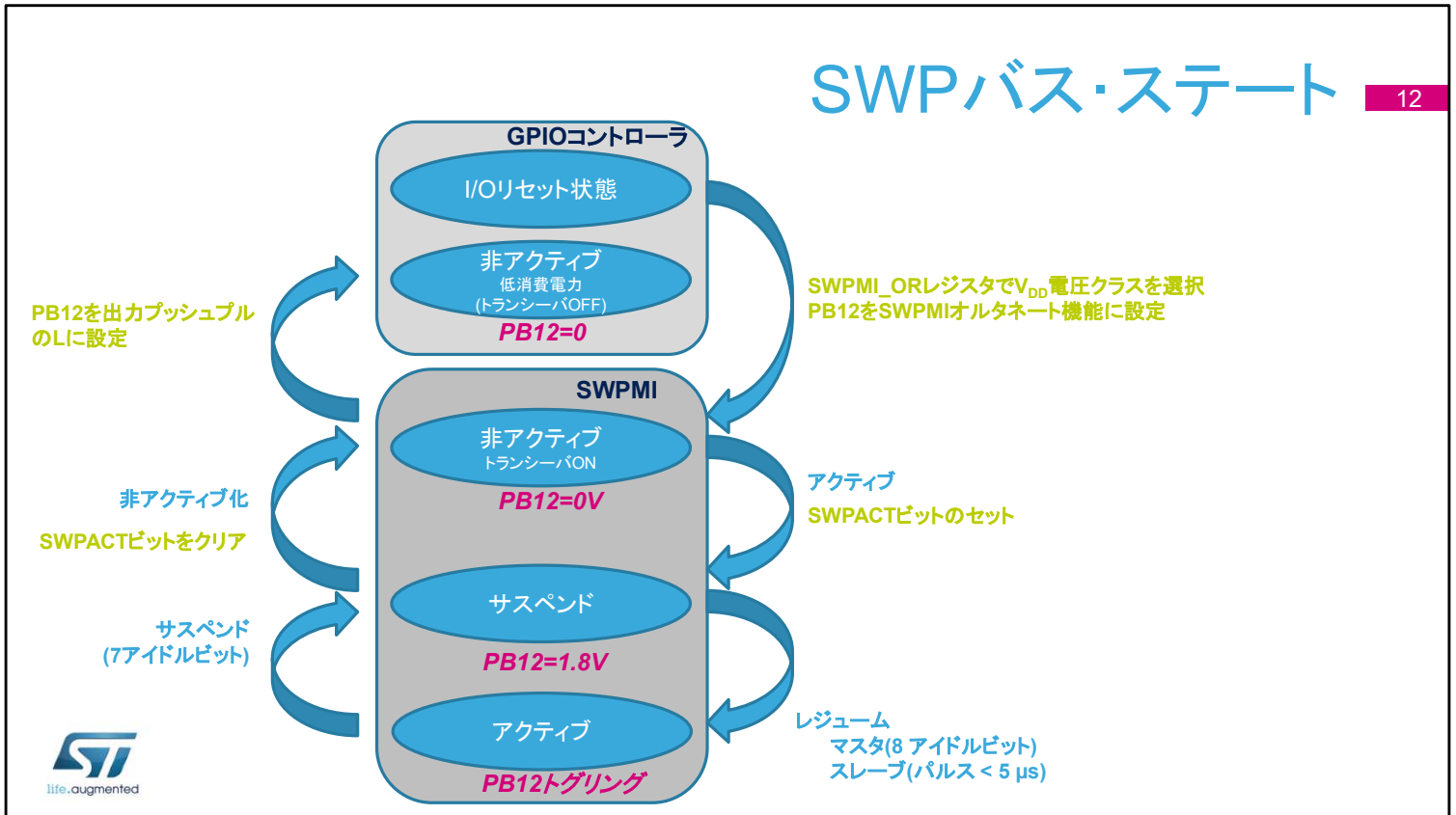
• マスタによるレジューム



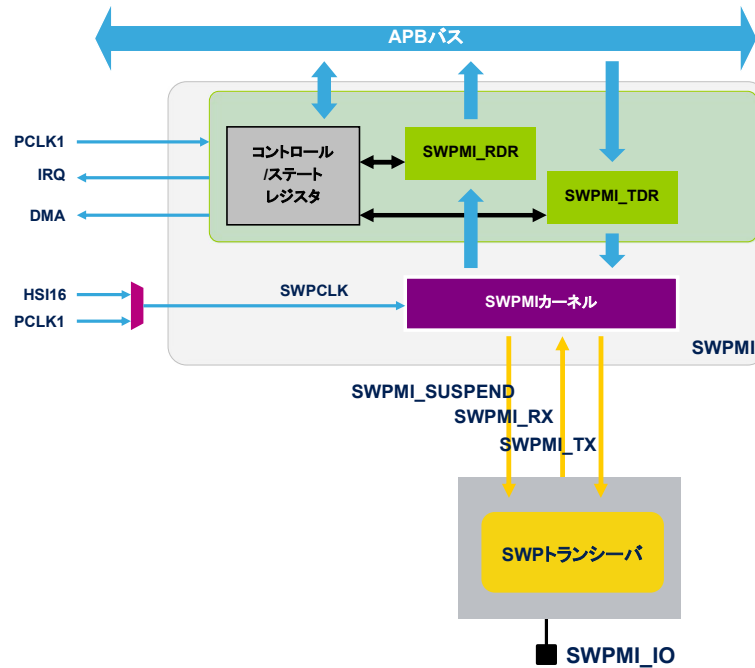
• スレーブによるレジューム



マスタまたはスレーブのどちらかがレジュームシーケンスを送信することによって通信を開始することができます。マスタによるレジュームシーケンスは遷移シーケンスと8つのアイドルビットで構成されますが、スレーブによるレジュームシグナルはマスタが検出するまで電流を引きこむことにより、結果として、S1信号がスレーブがデータ送信を開始するためトグルを開始します。

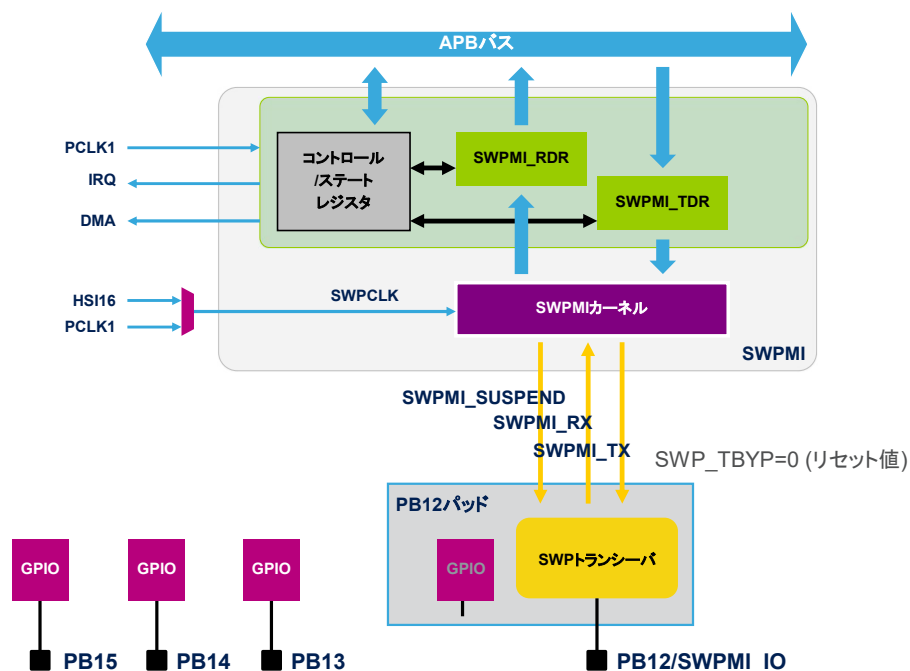


これは、SWPバス状態の、STM32によって管理されている方法の概要です。初期化と活性化の手順の詳細についてはリファレンスマニュアルをご参照ください。



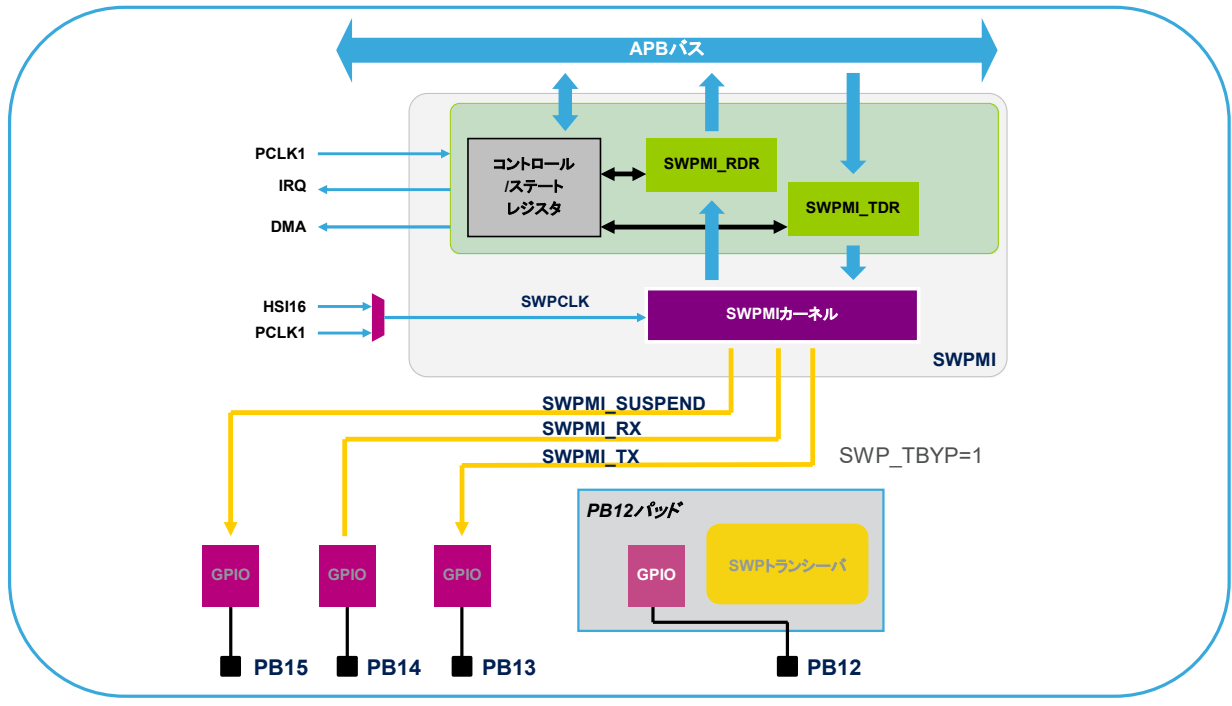
これはSWPMIペリフェラルのブロック図です。カーネル部分は、内部RC発振器であるHSI、またはAPBバスクロックであるPCLK1の、いずれかによってクロック供給されています。APBバスとのインターフェースは、CPUからSWPMIレジスタへのアクセスが可能です。NVICとDMAへの接続もあります。SWPMI_IO信号を介して外部ピンとのインターフェースとなる、SWPTランシーバはSTM32に内蔵されています。

内部ランシーバを備えたブロック図



内蔵のランシーバを使用したデフォルトの設定を次に示します。
SWPMI_IO信号はPB12ピンで利用できます。

外部ランシーバを備えたブロック図



SWPMIレジスタの構成ビットを使用して、外部ランシーバを接続することも可能です。この場合、サスペンド、受信および送信信号がピンPB15、PB14とPB13で利用できます。ピンPB12を標準GPIOとして使用することができます。

ソフトウェア・バッファ無しモード No Software Buffer Mode(NSB)

16

- 割込みまたはポーリング・モード
- ソフトウェアの介入は、4バイトのペイロード・データの送受信 (SWPMI 32ビット SWPMI_TDR/ SWPMI_RDRデータ・レジスタの書込み/ 読込み)に必要



ソフトウェアバッファ無しモード(NSB)で始まる、異なる動作モードについて示します。このモードでは、ポーリングまたは割込みモード、またはSWPMIフラグをチェックして、データを送受信します。受信データレジスタが一杯になったり、データ送信レジスタが空になったりするたびに、ソフトウェアの介入が必要になります。つまり、ペイロード内の4つのデータバイトごととなります。

ソフトウェア・シングルバッファ・モード Software Single Buffer mode(SSB)

17

- フレームの送信/受信中に、最後を除いて、ソフトウェアの介入は必要ない
- 送信用のRAM上の32バイトのフレームバッファ、および受信のために別のものを割り当てる必要がある
- DMAが32ビットSWPMI_TDR/SWPMI_RDRLレジスタに対して読み書きする
- フレーム長(ペイロードのバイト数)は、RAM上のソフトウェア・バッファの最初のバイトに書き込む

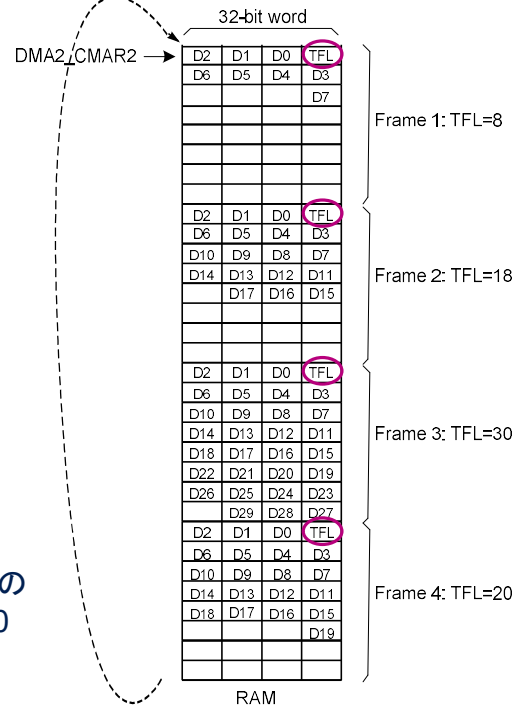


life.augmented

ソフトウェアシングルバッファモード(SSB)は、送信またはソフトウェアの介在なしに全体SWPフレームを受信するために使用されます。フレーム伝送のための32バイトのソフトバッファがRAMに定義され、SWPMIは自動的にフレームの終わりが受信されるまでは、DMAを通じてSWPMI_TDRレジスタに書き込みます。受信のために、32バイトのソフトバッファは、フレーム受信のためにRAM内に定義され、SWPMI_RDRLレジスタのコンテンツは、DMAによってRAMに転送されます。RAMバッファの最初のバイトは、フレームペイロード内のバイト数です。

ソフトウェア・マルチバッファ・モード(SMB)/送信時

- ソフトウェアの介入なしに、いくつかのフレームを送信
- 例:RAM上に4つのソフトウェア・フレームバッファがある場合
 - サーキュラモードで設定されたDMAを転送ワード数を32にする
 - バッファの最初のバイトはペイロード数をセット:送信フレーム長Transmit Frame Length (TFL)
 - ソフトウェアは、必要に応じて、DMAカウンタと更新フレームバッファを読み出す
 - DMAサーキュラモードを無効にすることによって、転送は停止されます。場合によっては、RAM内の1つまたは複数のバッファのTFLに0を書き込むことにより停止される(TFL=0はバッファが空であることを意味し、送信がトリガされない)



最後のモードは、ソフトウェアのマルチバッファモード(SMB)です。このモードは、DMAを使用し、複数のSWPフレームは、ソフトウェアの介入なしに扱うことができます。伝送のRAM上の4つのフレームバッファの例を示します。このモードでは、32バイトはペイロードサイズに関係なく、常に各フレームのために予約されています。

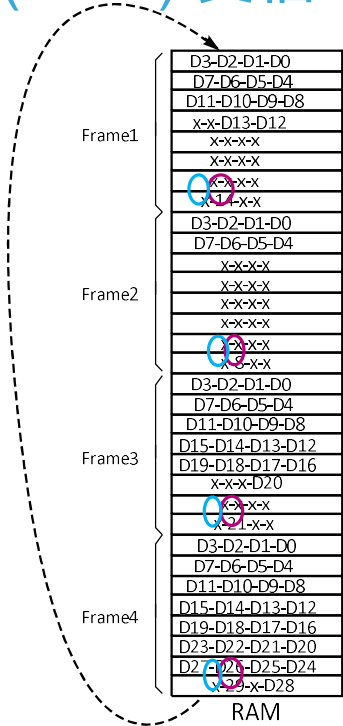
DMAはサーキュラモードで設定する必要があり、転送するワード数は32に設定する必要があります。

SSBモードと同様に、各バッファの最初のバイトは、フレームの長さ(これはTFLフィールドと呼ぶ)をセットするために使用されます。ソフトウェアは、DMAカウンタを読み取り、その情報に基づいて各フレームバッファを更新することができます。この例では、3つのフレームは、ソフトウェアの介入なしに送信することができます。送信は、DMAサーキュラモードを無効にすることによって停止されます。

DMAのカウンタの終了前に送信を停止する必要がある場合は、TFLフィールドに0をセットする必要があります。この場合、SWPMIIは、DMAリクエストを発行しません。

ソフトウェア・マルチバッファ・モード(SMB)/受信時

- ソフトウェアの介入なしに、いくつかのフレームを受信
- 例: RAM上に4つのソフトウェア・フレームバッファがある場合
 - サーキュラモードで設定されたDMAを転送ワード数を32にする
 - フレーム長はRAMバッファ内のデータの終わり(31番目、インデックス30)に追加される
 - その他の以下のフラグはフレーム長の次のバイト(32番目、インデックス31)にセーブされる:
 - RXBERF(エラー)
 - RXOVRF(オーバーラン・エラー)
 - RXBFF(バッファレディ)



SMBモードでは、いくつかのフレームは、ソフトウェアの介入なしに受信することができます。RAM上の4つのフレームバッファを受信する例を示します。このモードでは、DMAはサーキュラモードで設定され、転送されるワード数は、32に設定されている必要があります。フレーム長は各ソフトウェアバッファの最後の31番目のバイトにあります。各ソフトウェアバッファに格納されたフレームの状態は32番目のバイトにあります。これは、エラー、オーバーランエラー、バッファのレディフラグが含まれています。この場合、ソフトウェアは、バッファレディフラグをチェックし、バッファを読んで、32番目のバイトをクリアすることができます。

割込みイベント	説明
受信バッファフル	受信中のフレームの最後のワードがSWPMI_RDRLレジスタにある
送信バッファempty	送信フレームを送るのにSWPMI_TDRのアップデート必要なし
受信CRCエラー	受信フレームでCRCエラーがあった
受信オーバーラン	ペイロード受信中にオーバーラン・エラーが発生 SWPMI_RDRLレジスタは、ソフトウェアまたはDMAによって時間内に読み取れない
送信アンダーラン	ペイロード送信中にアンダーランエラーが発生 SWPMI_TDRレジスタは、ソフトウェアまたはDMAによって時間内に書き込まれていない
受信レジスタnot empty	SWPMI_RDRLに受信データあり
送信レジスタempty	SWPMI_TDRレジスタに書き込まれたデータが送信され、SWPMI_TDRレジスタに書き込める
スレープ・レジューム	SWPバスがサスペンドのとき、スレープがレジュームを要求



ここでNVICコントローラ内の割込みをトリガすることができるイベントの概要を示します。

送信および受信バッファ、
送信および受信レジスタ、
エラー（CRC、オーバーランやアンダーラン）、
スレープによってのレジューム 等があります。

割込みイベント	説明
送信リクエスト	“送信レジスタempty”イベント時、DMAリクエスト発生
受信リクエスト	“受信レジスタnot empty”イベント時、DMAリクエスト発生

- DMAリクエストは、SSBまたはSMB動作モードで動作させるために機能を有効にする必要がある

DMAリクエストは、送受信のためにSWPMIIによって生成されます。SSBおよびSMBモードを使う場合、これらを有効にする必要があります。

モード	説明
RUN	アクティブ
SLEEP	アクティブ SWPMIの割込みはデバイスをSleepモードを終了
STOP	停止 ペリフェラルレジスタの内容は保持 HSI16がSWPMIソースクロックとして選択されている場合、スレーブによる再開受信によってデバイスはSTOPモードを終了
STANDBY	パワーダウン STANDBYモードを抜けるときにペリフェラルの再初期化が必要



ペリフェラルは、RUNモードとSLEEPモードでアクティブです。すべてのSWPMI割込みは、SLEEPモードからデバイスを起動できます。デバイスがSTOPモードに設定されている場合、スレーブによる再開イベントのみがデバイスを起動できます。STANDBYモードでは、ペリフェラルはパワーダウンになり、STANDBYモードを終了した後で再初期化する必要があります。

- このペリフェラルにリンクされているこれらのペリフェラルのトレーニングを参照ください
 - リセットとクロックコントロール(RCC)
 - SWPMIクロックを有効
 - SLEEPモードでSWPMIクロックの設定
 - SWPMIリセットの制御
 - ネステッドベクトル割込みコントローラ(NVIC)
 - SWPMI割込みの設定
 - 汎用I/O(GPIO)
 - 内部SWPMI_IOTランシーバを有効にするために、ピンPB12にオルタネート機能を設定



これは、シングルワイヤプロトコル・マスタ・インタフェースに関連するペリフェラルの一覧です。必要に応じて、詳細についてこれらのペリフェラルトレーニングをご参照ください。