



# STM32L4 – USART

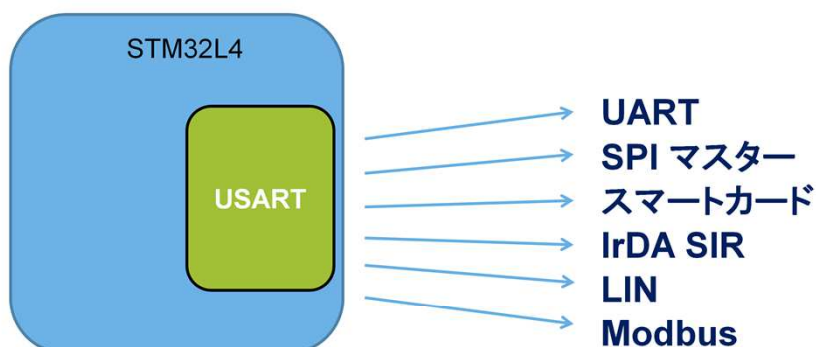
Universal Synchronous/Asynchronous Receiver/Transmitter Interface

Revision 1



Jan - 2016

こんにちは。STM32 Universal Synchronous/Asynchronous Receiver/Transmitter Interfaceのプレゼンテーションへようこそ。  
こちらでは、組み込みシステムでのシリアル通信に幅広く利用されているUSART(ユーサート)インタフェースについてご紹介します。



### Application の利点

- 多目的な通信ペリフェラル
- 簡単なハードウェア、数ピンのみ
- Stop 1低消費電力モードからのWakeup



USARTは非常に柔軟なシリアルモジュールで、以下をサポート

- 非同期UART通信
  - SPI(シリアル・ペリフェラル・インターフェース)マスタモード
  - LIN(ローカル・インターコネクト・ネットワーク)モード
- また、ISO/IEC7816スマートカードとIrDAデバイスとインターフェースすることができる。

また、Modbusの通信を実装する際に有用な機能を提供する。アプリケーションがUSARTを使う利点は、わずか数のピンを必要とするデバイス間で簡単かつ安価な接続ができること。また、USARTペリフェラルは、低電力モードにおいても動作する。

- 完全にプログラマブルなシリアル・インターフェース
  - データは 7、8、9bit
  - Even、odd、パリティ無し
  - 0.5、1、1.5、2 ストップビット
  - プログラマブルなデータオーダー (MSB/LSB first)
  - プログラマブルなボーレート生成器
  - 8倍か16倍かのオーバーサンプリング
- RS-232 と RS-485 ハードウェアフロー制御をサポート
- デュアルクロックドメインをサポート:
  - UART 機能とStop 1モードからのWakeup
  - PCLKの変更と独立したボーレート設定



USART は完全にプログラマブルなシリアルインターフェースで、プログラマブルな機能は:

- データ長
- パリティ
- ストップビットの数
- データオーダー
- ボーレート生成器

および、8倍または16倍のオーバーサンプリング

また、基本的なRS-232フロー制御を使用するCTSとRTSをオプション信号に持っている。

RS-485 DE (ドライバイネーブル) 信号もサポートされている。

USARTはペリフェラルクロック (PCLK) からボーレートプログラミングを独立し、Stop モードからのウェイクアップを可能にするデュアルクロックドメインをサポートしている。

これはまた、ペリフェラルクロックは、通信を中断することなく、コアクロックと一緒に減速できる。

- マルチプロセッサ通信
- 単線半二重通信
- オートボーレート検知
- 受信タイムアウト機能
- さらに
  - LIN モード
  - 同期モード (マスターモード)
  - IrDA SIR エンコーダ/デコーダ
  - スマートカード (T=0, T=1)
  - Modbus/RTU と Modbus/ASCII のための基本機能



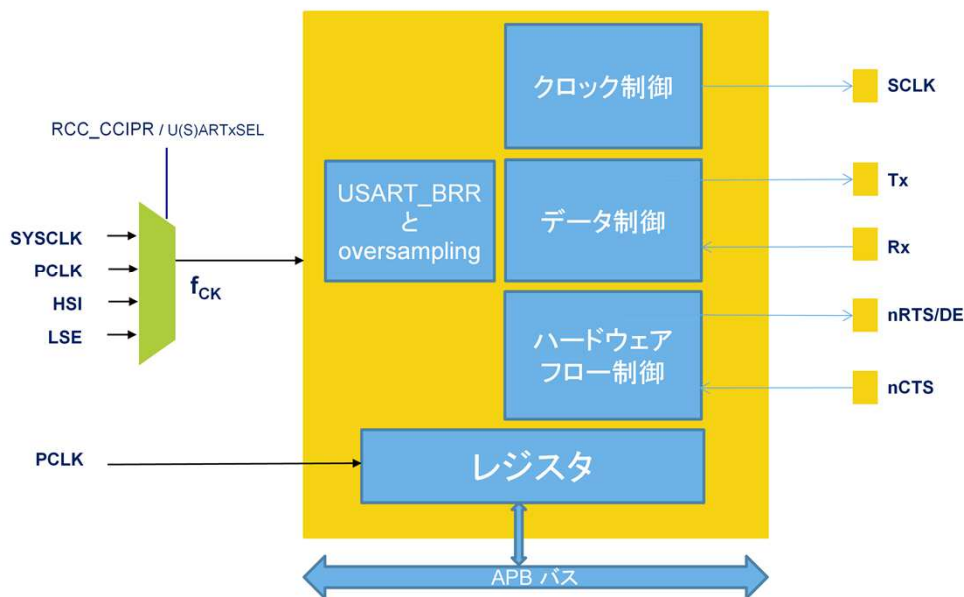
USARTの機能マルチプロセッサモードは、それがアドレスされていない時、USARTはアイドル状態のまますることができます。

全二重通信に加えて、単線半二重モードもサポートされています。

USARTは、その他、自動ボーレート検出、受信タイムアウト機能を含む多くの機能を提供し、プレゼンテーションで後述するいくつかのモードをサポートしている。

## ブロック図

5



これはUSARTのブロック図である。

USARTクロック(FCK)は、いくつかの供給源から選択することができる: システムクロック、ペリフェラルクロック(APBクロック)、HSI 16 MHzクロック、LSE 32.768kHzクロック。

TxとRxは、データの送受信のために使用される。

nCTSとnRTSは、RS-232ハードウェアフロー制御のために使用される。

ドライバインレーブル(DE)はnRTSと同じI/Oピンに存在し、RS-485モードで使用される。

クロック出力(SCLK)は二つの目的がある:

USARTが同期マスターモードで使用する場合、スレーブデバイスに供給されるクロックは、SCLKピンに出力される。

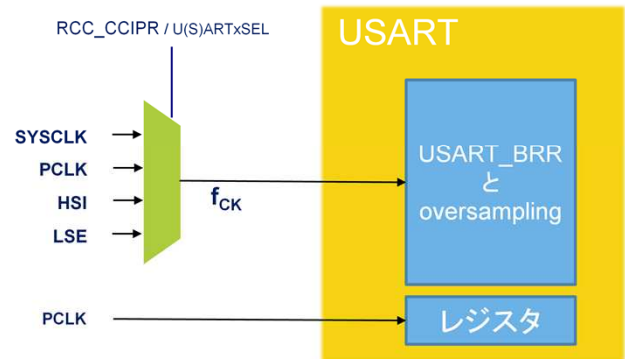
USARTがスマートカードモードで使用する場合、カードに供給されるクロックは、SCLKピンに出力される。

# デュアルクロックドメイン

6

PCLKの再プログラミングとボーレートのプログラミングは独立している

- 選択可能なクロックソースを介した、フレキシブルなクロッキング方式
  - PCLK (デフォルト)
  - HSI クロック
  - LSE クロック
  - システムクロック (SYSCLK)
- レジスタへのアクセスは、常にペリフェラルバス速度



USART はフレキシブルなクロック方法を持つ。  
そのクロックソースは、RCCで選択することができ、いずれかのクロック・ソース、PCLK(ペリフェラルクロック:デフォルト)、HSI、LSE、システムクロックを選択することができる。  
レジスタは、APBバスを介してアクセスされ、カーネルはAPBクロックから独立しているFCKでクロックされる。

# オーバーサンプリング

7

## ユーザー設定可能なオーバーサンプリング技術

- オーバーサンプリングの選択肢は、速度とフレーミングのトレランスに影響を与える:

|    | オーバーサンプリング /8        | オーバーサンプリング /16           |
|----|----------------------|--------------------------|
| 利点 | $f_{ck}/8$ の最大速度がだせる | クロックずれの最大受信耐性が増加する       |
| 欠点 | クロックずれの最大受信耐性が減少する   | 最大速度が $f_{ck}/16$ に制限される |

- 最大ボーレートは、選択されたクロックとオーバーサンプリングに依存: クロックソースが80 MHz、およびオーバーサンプリングが割る8であるとき、それは10メガボー。



USART受信機は有効な着信データとノイズとを判別することにより、データ回復のためのユーザが設定できるオーバーサンプリング技術を実装している。

これは、最大通信速度とノイズ/クロック不正確耐性とのトレードオフが可能になる。

オーバーサンプリング割る8を選択するのは( $f\_CLK/8$ までの)最大高速を達成するためで、 $f\_CLK$ はUSARTクロックソース周波数のこと。この場合、クロックの偏差の最大受信耐性が低下する。

オーバーサンプリング割る16(OVER8=0)を選択によって、クロック偏差に受信機の許容範囲を増加させることができる。この場合、最大速度は $f\_CLK/16$ に制限されている。

クロックソースが80 MHzおよびオーバーサンプリング割る8で設定されているときに到達することができる最大ボーレートは、10 Mボー。

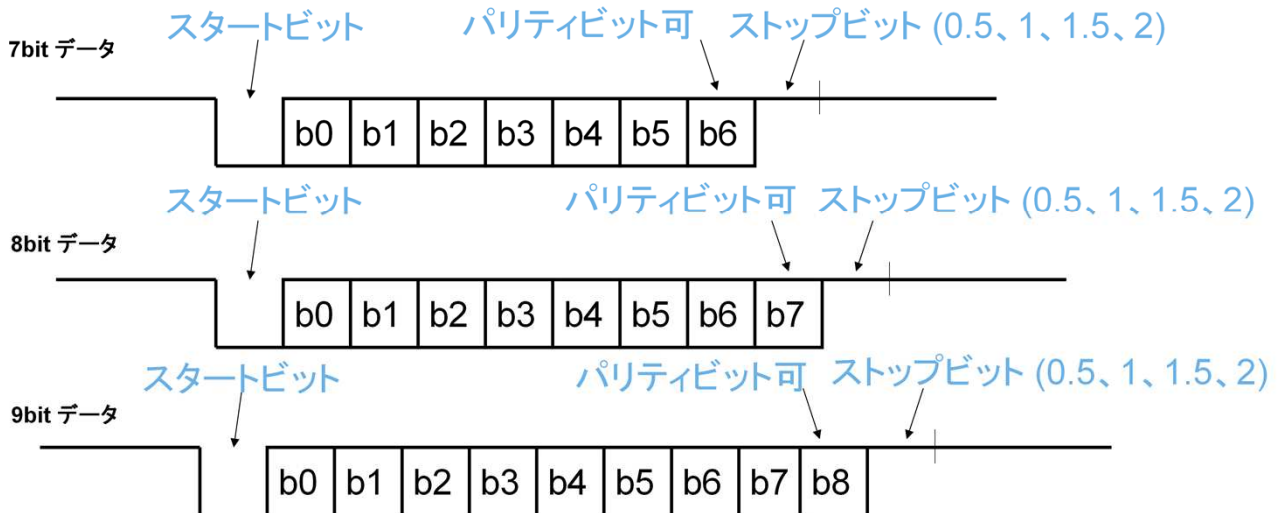
他のクロック源、および/またはより高いオーバーサンプリング比で、最高速度は制限される。



# データフォーマット – 非同期モード

8

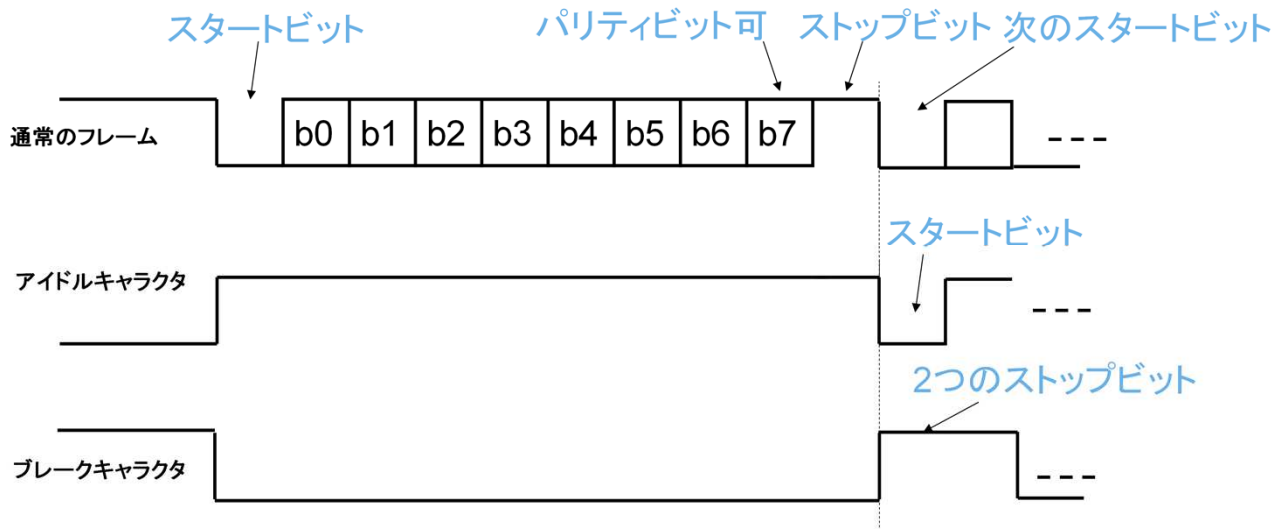
サポートしているデータ長: 7、8、9bit



非同期モードで使用するフレームフォーマットは、エラーチェックのための同期および必要に応じて、パリティビットのためのビットを加えた、データビットのセットからなる。USARTは7、8、9ビットのデータ長をサポートしている。フレームは、ラインが1ビット周期ローに駆動される1つのスタートビットで始まる。これは、フレームの開始を通知し、同期のために使用される。スタートビットのあと、7、8、9のデータビットが続く。パリティ制御が有効になっている場合、パリティビットは、最後のデータビットとして送信され、データ長カウントに含まれている。最後に、ストップビット(0、1、1.5、2)の数、ラインがハイに駆動され、フレームを終了します。



## アイドル/ブ레이크 キャラクタ 9



標準のフレームは、前のスライドで説明済み。  
このスライドは、1ストップビットで構成された8ビットのデータフレームの一例を示す図である。  
アイドルキャラクタはすべて「1」のフレームとして解釈される。（「1」の数は、ストップビットの数を含む）。  
ブ레이크キャラクタは、フレーム期間にすべて「0」を受信すると解釈される。ブ레이크フレームの終了時に、2ストップビットが挿入されている。

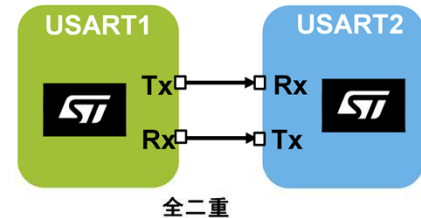
## 全二重／半二重モード

10

全二重: 2つのワイヤ  
半二重: 1つのワイヤ

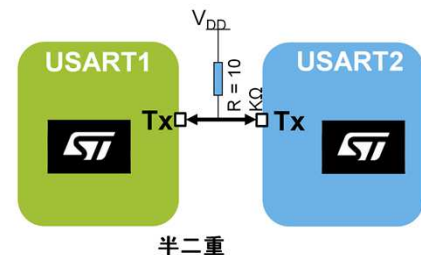
### • USART 全二重通信:

- TxとRxラインはそれぞれ、他のインタフェースのRxおよびTxラインに接続されている



### • USART 半二重通信:

- TxとRxラインは内部で接続されている
- Txピンは、送信と受信の両方のために使用される



USARTは、TxとRxラインがそれぞれ他のインタフェースのRxおよびTxラインに接続されている全二重通信をサポートしている。

USARTは、TxとRx線が内部で接続される、単線半二重プロトコルに設定することができる。

この通信モードでは、Txピンだけが、送信と受信の両方のために使用される。

Tx端子は何もデータが送信されない場合、常に、アイドルまたは受信モードで標準的なI/Oとして機能するようになっている。

これは、Txは、外部プルアップでオルタネート機能オープンドレインとして設定されているように、I/Oを構成しなければならないことを意味する。

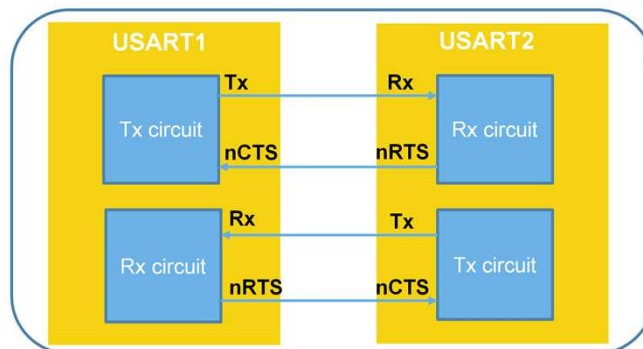
# RS-232 ハードウェアフロー制御

11

データアンダーラン/オーバーランを避けるためのハードウェアハンドシェイク

## • RS-232 ハードウェアフロー制御

- nRTS (送信要求) 出力がアサート（高レベル）のときは、受信機がデータを受け入れる準備ができていることを意味する。
- nCTS (送信クリア) 入力のアサート（高レベル）のときは、送信機が通信を継続できることを意味する。
- 特に半二重システムに有用。



RS-232通信では、nCTS入力とnRTS出力を用いて、2つのデバイス間のシリアルデータのフローを制御することが可能。この2ラインは、受信機と送信機は、それらの状態のお互いに警告することができる。この図は、このモードで、2つのデバイスを接続する方法を示している。半二重通信の場合にドロップバイトまたは衝突を防止するのに便利。両方の信号はアクティブ・ロー。

# RS-485 ハードウェアフロー制御

12

## ハードウェアハンドシェイク

- マスターは、トランシーバ(物理層(PHY))を制御するための方向信号を生成する必要があるが半二重システムで有用。この信号は、送信か受信かを、PHYに通知する。
- 外部のRS-485バスのドライバを活性化するために、DE(ドライバイネーブル)ピンを使用する。
- DEとnRTS信号は同じピンを利用している。



RS-485のようなシリアル半二重通信プロトコルでは、マスターは、トランシーバ(物理層)を制御するための方向信号を生成する必要がある。その信号は、送信か受信かを、物理層に通知する。

RS-485モードでは、Driver Enable(DE)ラインが使用される。DEは、外部トランシーバ制御を活性化するために使用される。DEとnRTSでピンを共有している。

## 複数のデバイス間の通信

- マルチプロセッサ通信では、唯一の意図されたメッセージの受信者は、積極的にメッセージを受信することが望ましい。
- アドレス指定されていないデバイスがミュートモードになる。
- ミュートモードは2つの方法を使用して制御することができる:
  - アイドルライン検知
  - アドレスマーク検知

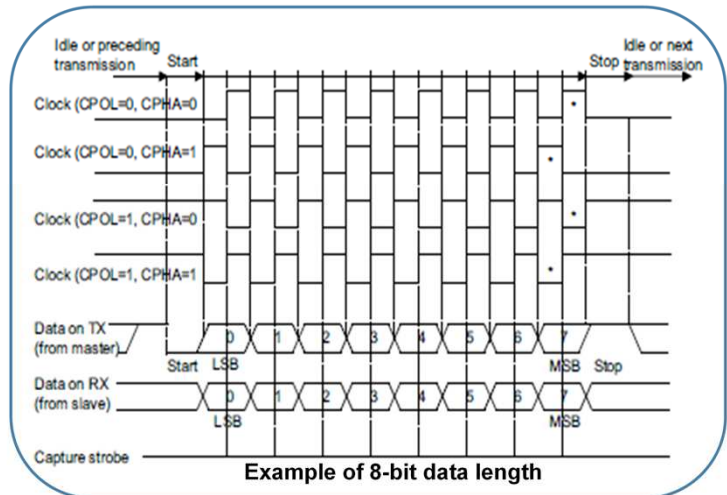
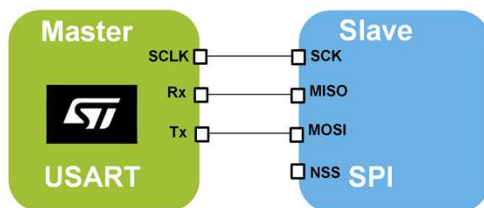


複数のプロセッサ間の通信を簡素化するため、USARTは、マルチプロセッサモードをサポートする。マルチプロセッサ通信では、唯一の意図されたメッセージの受信者は、積極的にメッセージを受信することが望ましい。アドレス指定されていないデバイスがミュートモードになる。USARTは2つの方法のいずれかを使用してミュートモードに入力または出ることができる。

- アイドルライン検知
- アドレスマーク検知

## USART をSPIマスターのように使う

- 全二重同期通信モード:
  - SPI マスターモード
  - プログラマブルなクロック極性 (CPOL) と位相 (CPHA)
  - クロックの出力は SCLK ピン
  - スタートとストップビット中はクロックパルス無し



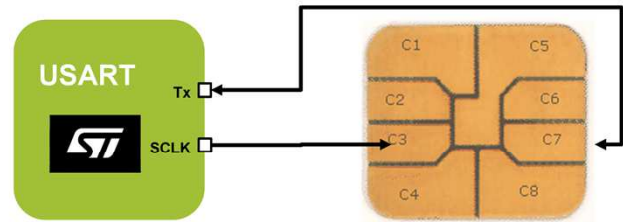
USARTはまた、同期通信を行うことができる。  
 これは、プログラマブルクロック極性 (CPOL) と位相 (CPHA) とマスターモードのSPIとして動作することができる。  
 クロックはSCLK端子に出力される。クロックパルスは、スタートビット時、ストップビットに提供されない。

# ISO/IEC 7816 モード

15

## スマートカードとセキュリティアクセスモジュールのUSARTインタフェース

- 半二重モード
- スマートカードへのクロックはSCLKピンから出力
- クロック入力の広い範囲を保証するための、プログラマブルクロックプリスケアラ
- ISO/IEC 7816 T=0 と T=1 プロトコルをサポート
- ダイレクトとインバースコンベンションの両方が利用可能



USARTは半二重モードに基づくスマートカードモードが使用可能

スマートカードへのクロックはSCLKピンから出力

これはT=0とT=1の両方のプロトコルをサポート

ハードウェアによってダイレクトとインバースコンベンションの両方が利用可能

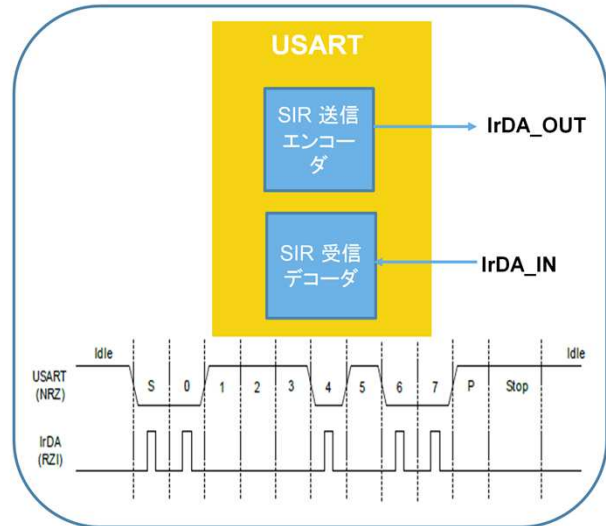


# IrDA SIR エンコーダ/デコーダ

16

## 赤外線リモコン通信用USARTインタフェース

- 半二重通信
- USARTのデータはNRZ (Non Return to Zero) フォーマットで表現されている
- IrDAではRZI (Return to Zero Inverted) フォーマットが必要
- SIR送信エンコーダは、USART出力前に、信号を変調する。同様に、入力信号は、受信SIRデコーダで復調される
- 最大ビットレートは 115.2 Kbits/s
- パルス幅はノーマルモードで 3/16bit期間



USARTは、半二重通信プロトコルであるIrDAの仕様をサポートしている。

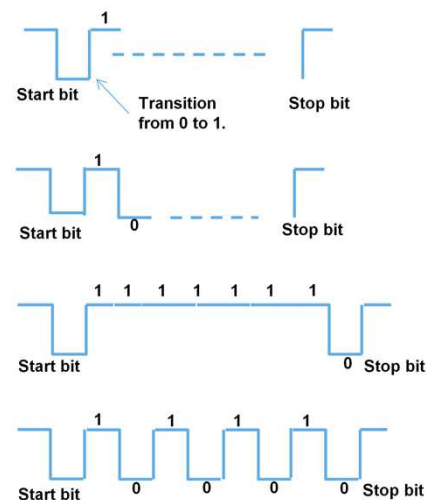
信号値が全体のビット周期を介して同じレベルであると、USARTからのデータは、フォーマットNRZ (Non Return to Zero) で表されます。IrDAのために、必要な形式はRZI (Return to Zero Inverted) で"1"の行は、Lを保持することによって通知され、"0"が短い高パルス信号で通知される。SIR送信エンコーダは、USARTからのNRZビットストリームを変調する。SIR受信デコーダは、赤外線検出器からのRZIを復調し、NRZシリアルビットストリームをUSARTに出力する。

USARTは、SIR ENDECで115.2Kbpsまでのビットレートをサポートしています。

ノーマルモードでは、送信されたパルス幅はビット周期の3/16として指定されます。

## 自動ボーレートの設定 - UART 受信

- USARTは一つの文字の受信に基づいて自動的に通信速度を決定することができる。
- 受信キャラクタは
  - 最初のビットが'1'の任意の文字
  - 最初のビット列が'10xx'の任意の文字
  - 0x7F
  - 0x55



USART受信機は、一つの文字の受信に基づいて通信速度を検出し、自動的に設定することができる。

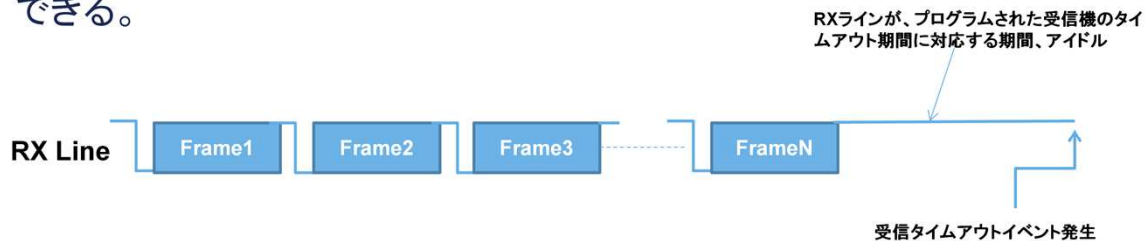
受信した文字は、

1のビットで始まる文字。この場合、USARTは立ち上がりエッジ、立ち下がりエッジでスタートビットの持続時間を測定する。10XXパターンで始まる任意の文字。この場合には、USARTを開始し、第1データビットの継続時間を計測する。期間は、立ち下がりエッジに立ち下がりエッジから測定する。低速信号スロープの場合は、より良い精度を確保できる。

0x7F。この場合、ボーレートはビット6の最後からスタートビットの最後で更新されます。

0x55。この場合、ボーレートは、スタートビットの終わりに更新され、最終的に6ビットの終わりに更新される。平行して、別のチェックは、RXラインの各中間遷移を行います。

- USART受信機は、プログラムされた時間の間、新たなデータを受信しない場合、受信タイムアウト・イベントを介してアプリケーションに通知することができる。



- USART受信機のタイムアウト・カウンタがカウントを開始:
  - 1または1.5ストップビット構成の場合の最初のストップビットの終わりから
  - 2ストップビット構成の場合の第2のストップビットの終わりから
  - 0.5ストップビット構成の場合のストップビットの初めから



USARTは受信タイムアウト機能をサポートする。  
USARTがプログラムされた時間に新しいデータを受信しないとき、受信タイムアウトイベントが生成され有効なときに割込みが発生する。

USART受信タイムアウトカウンタは以下のコンディションで開始

- 1または1.5ストップビット構成の場合の最初のストップビットの終わりから
- 2ストップビット構成の場合の第2のストップビットの終わりから
- 0.5ストップビット構成の場合のストップビットの初めから

# StopモードからのWake up

19

- USARTは、クロックソースが以下の場合、Stopモードからマイコンをwake up可能
  - HSI
  - LSE
- Wakeupのためのソース
  - 通常の RXNE 割込み
  - 特別な wakeup イベント
    - スタートビット
    - アドレスマッチ
    - 任意の受信データ



USARTは、クロックソースがHSIまたはLSEの場合、Stopモードからマイコンをwake up可能  
Wakeupのためのソースは、スタートビット、アドレスマッチ、任意の受信データ

| 割込みイベント                         | 説明                                      |
|---------------------------------|---|
| Transmit data register empty    | 送信データレジスタが空のときセット                       |
| Transmit complete               | データ送信が終了してデータとシフトレジスタの両方が空のときセット        |
| CTS                             | nCTS入力がトグルしたときセット                       |
| Receive data register not empty | 受信レジスタにデータがあるときセット                      |
| Idle line                       | アイドルラインを検知したときセット                       |
| Character match                 | 受信データがプログラムされたアドレスのときセット                |
| Receiver timeout                | プログラムされたタイムアウト期間Rxラインになにもデータがこなかったときセット |



いくつかのイベントが割込みになる:

- Transmit Data Register Empty フラグは、送信データレジスタが空で書き込み準備のできたときにセット
- Transmit Complete フラグは、データ送信が終了してデータとシフトレジスタの両方が空のときにセット
- CTS フラグは、nCTS入力がトグルしたときセット
- Receive Data Register Not Empty フラグは、受信データレジスタにデータがあって読み出し準備ができたときにセット
- Idle Line フラグは、アイドルラインを検知したときセット
- Character Match フラグは、受信データがプログラムされたアドレスのときセット
- Receiver Timeout フラグは、プログラムされたタイムアウト期間Rxラインになにもデータがこなかったときセット

| 割り込みイベント              | 説明                   |
|-----------------------|----------------------|
| End of block          | 完全なブロックを受信したときにセット   |
| Wakeup from stop mode | wakeupイベントを確認したときセット |
| LIN break             | LIN ブレークを検知したときセット   |

- DMAリクエストは、送信データレジスタが空、および受信データレジスタがフルによってトリガされる



End of Block フラグは、完全なブロックを受信したときにセット

Wakeup from Stop Mode フラグは、wakeupイベントを確認したときセット

DMAリクエストは、送信データレジスタが空、および受信データレジスタがフルがセットされたときに生成される



| 割込みイベント             | 説明                     |
|---------------------|------------------------|
| Overrun error       | オーバーランエラーが発生したときセット    |
| Parity error        | パリティエラーが発生したときセット      |
| Framing error       | フレーミングエラーが発生したときセット    |
| Noise error         | 受信フレーム上でノイズが検出されたときセット |
| Auto-baudrate error | ボーレート計測に失敗したときセット      |



いくつかのエラーフラグが生成される:

- Overrun error フラグはオーバーランエラーが発生したときにセットされる
- Parity error フラグはパリティエラーが発生したときセットされる
- Framing error フラグはフレーミングエラーが発生したときセットされる
- Noise error フラグは受信フレーム上でノイズが検出されたときセットされる
- Auto-baudrate error フラグはボーレート計測に失敗したときセットされる



| モード             | 説明   |
|-----------------|--|
| Run             | アクティブ  |
| Sleep           | アクティブ。ペリフェラルの割込みはデバイスをSleepモードから起こす  |
| Low-power run   | アクティブ  |
| Low-power sleep | アクティブ。ペリフェラルの割込みはデバイスをLow-power sleepモードから起こす  |
| Stop 1          | USARTは、クロックがHSI16かLSEがセットされているときマイコンをStop 1モードから起こすことが可能。マイコンは通常のRXNE割込みまたはWUFイベントで、Stop 1モードからおきることが可能。 |
| Stop 2          | 停止。ペリフェラルのレジスタは保持する  |
| Standby         | パワーダウン。ペリフェラルはStandbyモードから抜けた後に初期化が必要  |
| Shutdown        | パワーダウン。ペリフェラルはShutdownモードから抜けた後に初期化が必要   |



USARTペリフェラルは、Runモード、Low-power run、Sleepとlow-power sleepモードでアクティブ。I2C割込みは、デバイスがSleepまたはモードを終了するための、要因になる。USARTはクロックがHSI16またはLSEに設定されている場合、Stop 1モードからマイコンを起動することができる。マイコンは、標準RXNE割込みやWUFイベントのいずれかを使用してStop 1モードからウェイクアップすることができる。Stop 2において、デバイスは、通信を行うことができません。StandbyおよびShutdownモードでは、ペリフェラルは、パワーダウンにあり、StandbyまたはShutdownモードを終了した後に再初期化する必要がある。

# STM32L4 USART インスタンス機能一覧

24

| USART 機能                        | USART1/2/3 | UART4/5 |
|---------------------------------|------------|---------|
| モデムのためのハードウェアフロー制御              | X          | X       |
| マルチプロセッサ通信                      | X          | X       |
| 同期モード                           | X          | -       |
| スマートカードモード                      | X          | -       |
| 単線半二重通信                         | X          | X       |
| IrDA SIR ENDEC                  | X          | X       |
| LIN モード                         | X          | X       |
| デュアルクロックドメインおよびStopモードからのwakeup | X          | X       |
| 受信タイムアウト                        | X          | X       |
| Modbus 通信                       | X          | X       |
| 自動ボーレート検知                       | X          | X       |
| RS485 Driver enable             | X          | X       |



STM32L4デバイスは5つのインスタンスを内蔵する:

- USART1、2、3はフル機能
- 4 と5は同期モードとスマートカードモードをサポートしない

- 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
  - GPIO (オルタネート機能設定)
  - Reset と clock controller (RCC)
  - Power controller (PWR)
  - Interrupts (NVIC と EXTI)
  - Direct memory access controller (DMA)



これらはUSARTに関連するペリフェラルのリストである。必要な場合、これらのペリフェラルのトレーニングを参照のこと。

- General-purpose input/outputs
- Reset とclock controller
- Power controller
- Interrupts controller
- Direct memory access controller