

## ADuC703xシリーズのLINボーレート計算

著者：Aude Richard

### はじめに

このアプリケーション・ノートの目的は、アナログ・デバイセズが発売するADuC703xシリーズのUARTで使用するデバイズ値 (COMDIV0、COMDIV1、COMDIV2) の計算に習熟していただくことです。このアプリケーション・ノートでは、読者がLocal Interconnect Network (LIN) 2.0仕様に精通しているものと想定します。

本書は、以下の3つの節で構成されます。

- LINフレームヘッダ：LINフレームヘッダとSyncバイトについて説明します。
- LINボーレート計算：LIN Hardware Synchronization (LHS) 機能を使用してUARTの分周値を計算する方法について説明します。
- LINボーレート計算のCコード例：「LINボーレート計算」で説明した計算のCコードによる実装例を示します。

### LINフレームヘッダ

図1は、標準のLIN通信フレームを示します。このフレームは、ブレイクシンボル、Syncバイト、プロテクトID、データ、チェックサムで構成されます。

- ブレイクシンボルは、LINパケットの先頭を表します。
- Syncバイトは、スレーブのボーレートをキャリブレーションします。
- プロテクトIDは、スレーブを識別します。
- チェックサムは、送信データについて計算される標準チェックサム、またはプロテクトIDとデータについて計算される拡張チェックサムです。

図2は、Syncバイトの詳細を示します。Syncバイトは、マスターの所望のボーレートで送信される0xAAです。マスターのビットレートを判断する標準的な方法としては、最初の立下がりエッジから5番目の立下がりエッジまでの時間を測定します。この値を8で割ると、所望のビットレートが得られます。その後、この値はUARTの分周値の計算に使用されます。計算の詳細は、以降の節で説明します。このアプリケーション・ノートでは、Syncバイトの全8ビットのタイミングをとるために、LHS MMRが設定されているものと想定します (LHSCON1=0x62)。

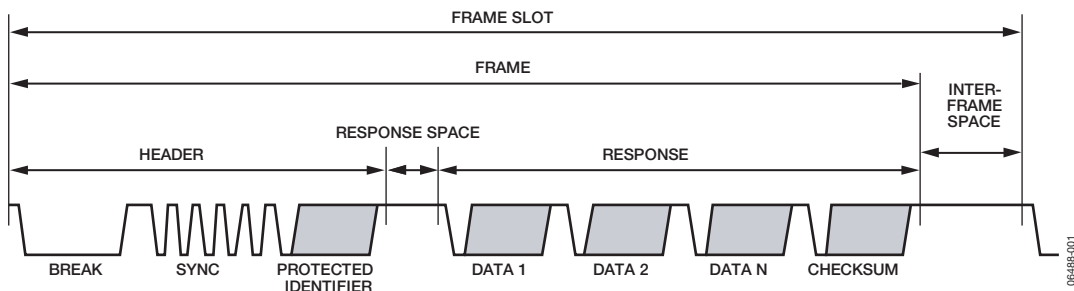


図1. LINフレーム



図2. LIN Syncバイト

REV. 0

**アナログ・デバイセズ株式会社**

本社 / 〒105-6891 東京都港区海岸1-16-1 ニューピア竹芝サウスタワービル  
電話03(5402)8200  
大阪営業所 / 〒532-0003 大阪府大阪市淀川区宮原3-5-36 新大阪MTビル2号  
電話06(6350)6868

## LINボーレート計算

LHSシステムを使用すると、Syncバイトの受信後にLHSVVAL0から値が得られます。LHSVVAL0には8つの $T_{BIT}$ に相当する値が含まれています。この値はUARTのデバイダCOMDIV0とCOMDIV1、およびフラクショナル・デバイダCOMDIV2の値の生成に使用されます。UARTの詳細については関連するADuC703xのデータシートを参照してください。

標準のボーレート・ジェネレータを使用してCOMDIV0/COMDIV1の値を計算するには、次の基本UART式を使用します。

$$DL = \frac{20.48 \text{ MHz}}{\text{ボーレート} \times 2^{CD} \times 16 \times 2}$$

ここで、

DLはCOMDIV0とCOMDIV1の値です。  
CDはクロック・デバイダです。

LHSVVAL0に基づいて、所望のボーレートは次のようになります。

$$\text{所望のボーレート} = \frac{5.12 \text{ MHz} \times 8}{LHSVVAL0}$$

(LHSVVAL0は内部の5.12MHzクロックから駆動されます。LHSCON1は8つの $T_{BIT}$ を測定するように設定されていると想定します。)

標準のボーレート式と所望のボーレート式を組み合わせた場合、

$$DL = \frac{20.48 \text{ MHz} \times LHSVVAL0}{5.12 \text{ MHz} \times 2^{CD} \times 16 \times 2 \times 8}$$

$$DL = \frac{LHSVVAL0}{2^{CD} \times 16 \times 2 \times 2}$$

$$DL = \frac{LHSVVAL0}{2^{CD+6}}$$

標準のボーレート・ジェネレータ式だけを使用すると、COMDIV0/COMDIV1の必要な値が得られます。

精度を高めるには、DL値 (COMDIV0/COMDIV1) を標準のボーレート・ジェネレータ向けにあらかじめ計算して、ADuC703xフラクショナル・デバイダを使用します。フラクショナル・デバイダを使用する式は、次のようになります。

$$\text{ボーレート} = \frac{20.48 \text{ MHz}}{DL \times 2^{CD} \times 16 \times 2 \times \left( M + \frac{N}{2048} \right)}$$

ここで、MとNはCOMDIV2の値です。

$$M + \frac{N}{2048} = \frac{20.48 \text{ MHz}}{\text{ボーレート} \times DL \times 2^{CD} \times 16 \times 2}$$

ボーレートを置き換えると、

$$M + \frac{N}{2048} = \frac{20.48 \text{ MHz} \times LHSVVAL0}{5.12 \text{ MHz} \times 8 \times DL \times 2^{CD} \times 16 \times 2}$$

これは次のように簡略化されます。

$$M + \frac{N}{2048} = \frac{LHSVVAL0}{DL \times 2^{CD} \times 2 \times 16 \times 2}$$

$$M + \frac{N}{2048} = \frac{LHSVVAL0}{DL \times 2^{CD+6}}$$

フラクショナル・デバイダの計算に使用される複雑な数値演算を簡略化するため、DL (COMDIV0/COMDIV1)の値を2のべき乗に制限します。たとえば、DL=17の場合、Nの計算でDL=16=2<sup>4</sup>を使用します。これによりDLの変更による誤差を補償するようにNの値が自動的に調整されます。

$$DL = 2^{DL\_Power}$$

$$M + \frac{N}{2048} = \frac{LHSVVAL0}{2^{DL\_Power} \times 2^{CD+6}}$$

$$M + \frac{N}{2048} = \frac{LHSVVAL0}{2^{DL\_Power+CD+6}}$$

M=1に設定された場合は、

$$N = \frac{2^{11} \times LHSVVAL0}{2^{DL\_Power+CD+6}} - 2048$$

$$N = 2^{5-DL\_Power-CD} \times LHSVVAL0 - 2048$$

たとえば、19,200bpsのボーレートで、CD=0、DL=33、LHSVVAL0=2133の場合、N=21でボーレートは19,197bpsです。DL=32とN=85を使用した場合、ボーレートは19,203bpsとなります。

## LINボーレート計算のCコード例

Cでプログラミングするときは、シフト・コマンド<<and>>を使用して前述の式を簡単に記述できます。

```
// DL = LHSVAL0 >> CD_Bits + 6
iDL = LHSVAL0 >> (( POWCON & 0x7)+6);

// writing DL as 2^iDL_Power
iDL_Power = 0;
iDL_temp = iDL;
while(iDL >> (iDL_Power +1 ))
{
    iDL_Power++;
}

// Configuration of the fractional divider:
// M = 1
// N = LHSVAL0 × 2 ^ (5 - (iDL_Power + CD)) - 2048
iDL_temp = iDL_Power + (POWCON & 0x7);
if (iDL_temp > 5)
{
    iDL_temp = (LHSVAL0 >> (iDL_temp -5)) - 2048;
}
else
{
    iDL_temp= (LHSVAL0 << (5 - iDL_temp)) - 2048;
}

COMDIV2 = 0x8800 + iDL_temp;

COMCON0 = 0x080; // Setting DLAB
// Setting DIV0 and DIV1 to DL calculated
COMDIV0 = (1<< iDL_Power) & 0xff;
COMDIV1 = (1<< iDL_Power) & 0xff00;
COMCON0 = 0x03; // Setting DLAB
COMIEN0 = 0x1; // Enable RX interrupt
```