

1. データ伝送におけるクロック再生の重要性

デジタル通信システムにおいて、受信データのシンボルを正しいタイミングで判断することが必要です。つまり、アイバターンの最も開いた瞬間にシンボルを判定することが求められます。これをシンボル同期といいます。そのために、受信側でのクロック再生 (Clock Recovery) 技術は極めて重要です。特に無線伝送の場合には、クロックを独立して伝送しないで受信側で同期クロックを生成しますから、その良否がデータ無線伝送システム全体の性能を左右します。

クロック再生の手がかりは、データの遷移にあります。例えば、9,600bps のデータでは1シンボルは約 $104\mu\text{s}$ 長ですが、データの遷移はその整数倍 (1,2,3,4,5.....) 毎に起こりますから、データ自身が 4,800Hz 3,200Hz 2,400Hz 1,920Hz..... というスペクトルを含んでいます。この特定周期を利用してクロックを再生します。

具体的なクロック再生回路として、デジタル通信の解説書では、(1) バンドパスフィルタ法 (2) 閉ループ法 を紹介しています。

バンドパスフィルタ法では、上記の 9,600bps 場合に例をとると、データを全波整流するかあるいは遅延波と排他 OR して 9,600Hz の成分を強調し、バンドパスフィルタで精製する方式です。この場合、データに短時間の欠落があったとしても、フィルタのフライホール効果によって、正確なクロックを維持することができます。これと同種の技術がテレビジョンのカラー同期信号のバースト同期に使われています。

閉ループ法は、PLL によるクロック再生回路が一般的です。これでは、データのエッジを一方の入力とし、VCO のエッジをもう一方の入力とする位相比較器の出力で VCO の周波数を制御するループを構成します。PLL のループフィルタの定数設定の解説が参考書にありますので、それを使って設計します。

ただし、一般の PLL の設計は周波数シンセサイザのような単一スペクトラムの連続波のロックインについて書かれており、データ遷移が間欠的であるデータに対する位相同期を意識して書かれた参考書は、小生の知る限りでは「無線によるデータ変復調技術」(西村 芳一 著 CQ 出版社) のみです。同書では、特殊な位相比較器を構成しています。

本稿では、簡単なデータ伝送システムに使えるクロック再生回路を提案します。

2. PRBS でのデータ遷移

データそのものを一般化するために、取り扱うデータを PRBS(Pseudo Random Binary Sequence) とします。いわゆる疑似雑音です。この信号は、例えば PN9 符号の場合には、図 1 の回路構成で発生します。

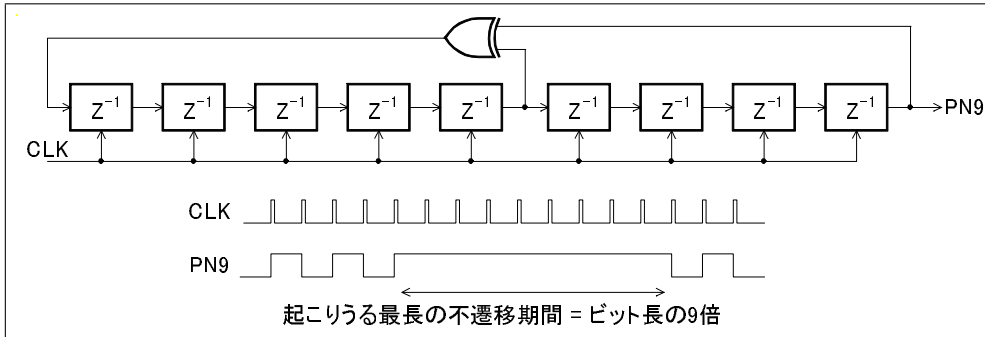


図 1: PRBS の符号構成

ここで、データにシンボル変化がないシーケンスに注目しましょう。PRBS では、シンボル '1' が PN9 では 9 ビット長の時間続くシーケンスが存在します。PN15 ではもっと厳しく、15 ビット長に亘って不遷移となる場合があります。

実際のデータ伝送では、この PRBS をそのまま使うのではなく、同一シンボルが長く連続しないマンチェスター符号等を選択して、クロック再生を容易にする工夫がされています。しかしここでは敢えて、PRBS のままのクロック再生に挑戦します。

3. 間欠遷移の問題を解決

さて、一般的な連続波を対象とする PLL クロック再生回路は、図 2 の黒字で表した部分の簡単な構成で済みます。

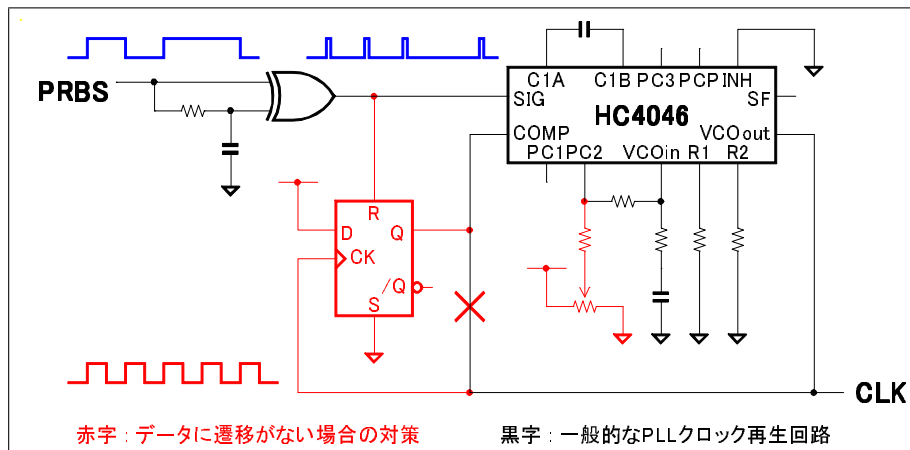


図 2: PLL を使ったクロック再生回路

問題はこの回路で、PRBS のような間欠遷移のデータから正しくクロックが再生できるかということです。HC4046 のエッジトリガ位相比較器 PC2 入力 SIG および COMP のパルスタイミングと PC2 出力との関係は、図 3(a) のようになります。この回路構成のままでは SIG のパルスが一部欠けると、COMP には連続パルスが加えられているので、図 3(b) のよ

うに VCO は目的周波数より下がる方向に動きます。これでは PLL は正常に動作しません。

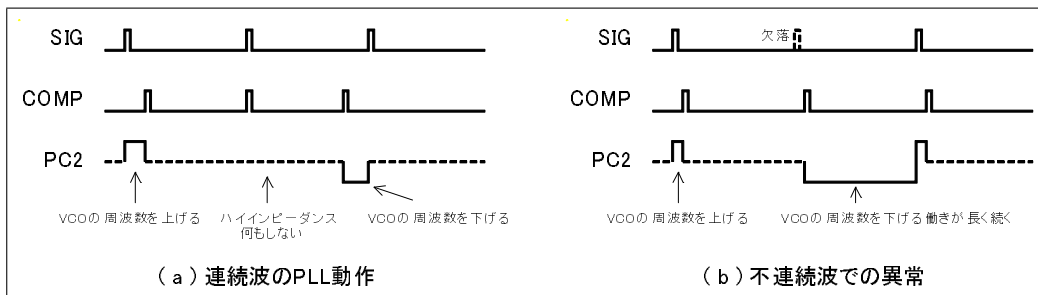


図 3: 位相比較器の二つの入力の関係

この不具合を解決するには、SIG 入力がない場合には COMP 入力を禁止する対策を講じればよく、先の図 2 の赤字で表した付加回路を挿入することで解決します。なお、VCO の固有発振周波数が目的周波数を超えると正常な動作をしなくなりますので、PC2 端子に追加した R で発振周波数を予め低い方にバイアスしておきます。この回路構成によって、PRBS に対するロックイン発振器を作ることができます。

自励発振器はシュミット NAND 回路でも構成できます。むしろこの方が回路が簡単でしょう。図 4 にその一例を示します。この回路も、PRBS にロックインするので、十分使えるクロック再生回路です。

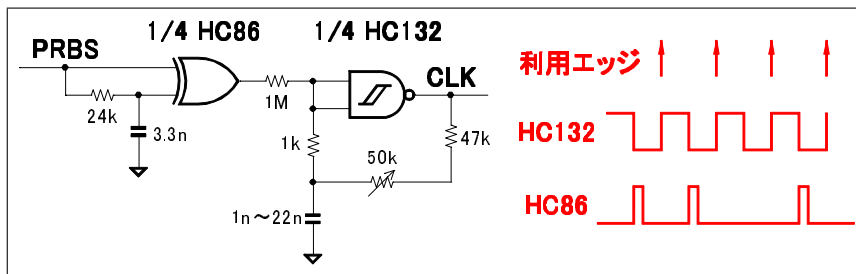


図 4: 簡単なクロック再生回路

この構成方法でも、VCO の固有発振周波数は目的周波数よりも少し低めに設定しておき、PRBS の遷移点で NAND 回路の出力を強制的に L とします。PRBS の遷移がない場合の NAND の L 転移および H 転移はフリーランで行われます。データ検波に利用するエッジは立上がりを利用します。

4. PIC によるデジタル的クロック再生法

上記の方法は発振回路をロックインする手順をとりましたが、これを純デジタル的に構成することもできます。PIC などのマイクロプロセッサを使うことを念頭に、そのソフトウェアの概要を紹介します。

シンボル検出に必要なクロックは、1 シンボル長に等しい周期をもちシンボルの時間的中央で立上がり (または立下がりどちらか) となるパルスです。つまり、周期とタイミングの両方が PRBS に同期している連続パルスです。この要件に合うパルスを、9,600bps の場合の 1 ビット長を $104\mu\text{s}$ として描いてみると、図 5 となります。

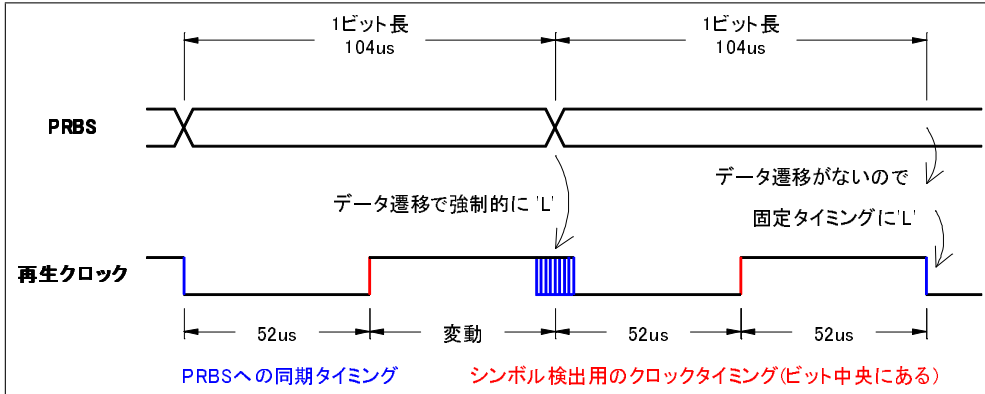


図 5: デジタル式クロック再生での同期 (9,600bps の例)

この方式では、PRBS にデータ遷移があると、再生クロックは強制的に L に移行します。これで再生クロックはタイミング同期がとられることになります。

その時点から 52µs 後に、再生クロックは内部的に H に移行します。H 状態は 52µs 間維持されますが、その時点で PRBS の遷移が検出されなければ、内部的に L に移行します。この 1 周期が 104µs ですので、9,600bps のシンボル長と一致しており、周期同期もタイミング同期もとれていることになります。図 6 はその実際の出力です。

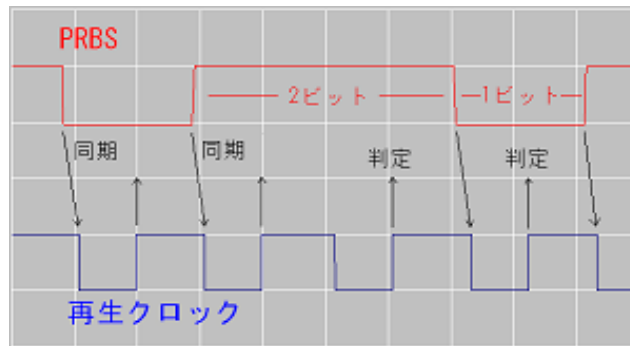


図 6: デジタル式クロック再生の波形

PIC 内部のクロック周期が 3% 以内程度に正確であれば、PN9 はもちろん PN15 のような長い不遷移シンボルが続いても、確実なクロック再生を行うことができます。シンボル検出には立上がりエッジを用います。

PIC12F675 用のアセンブラプログラムを掲載しますので、動作の詳細をご覧ください。

```

; clkrecover.asm (c) 2008 JA5FP
list p=12f675 ;
#include <p12f675.inc>;
errorlevel -302 ;
__CONFIG __CP_OFF & __CPD_OFF & __BODEN_OFF & __MCLRE_OFF
& __WDT_OFF & __PWRTE_ON & __INTRC_OSC_NOCLKOUT

bps EQU 0x20 ; 1200bps:0 2400bps:1 4800bps:2 9600bps:3
t_low EQU 0x21 ; 'L' 期間変数
t_high EQU 0x22 ; 'H' 期間変数
ORG 0x000 ;
goto main ;
ORG 0x004 ; 割込みあれば

```

```

BCF PIR1,CMIF ; フラグリセット
goto tol ; GP5 を'L' に同期

main BSF STATUS,RP0 ;Bank1
call 0x3ff ; 内部発振器の
MOVWF OSCCAL ; キャリブレート
MOVLW b'00001110'; CIN-,GP2,GP3 を
MOVWF TRISIO ; 入力
MOVLW b'00000010'; GP1 を
MOVWF ANSEL ; アナログ入力
MOVLW b'00000100'; GP2 を
MOVWF WPU ; プルアップ
MOVLW b'00001000'; コンパレータ割込みを
MOVWF PIE1 ; 使用
CLRF OPTION_REG ;
BCF STATUS,RP0 ;Bank0
BSF INTCON,PEIE ; ペリフェラル割込を使用
MOVLW b'00000100'; CIN-(GP1) が入力の
MOVWF CMCON ; コンパレータを構成
loop MOVLW b'00001100'; マスク
ANDWF GPIO,W ; GP2,GP3 の状態を
MOVWF bps ; bps に取得
BCF STATUS,C ; bps 変数に変換するため
RRF bps,F ; 桁移動
RRF bps,F ;
MOVF bps,W ; t_high を設定
call v_high ;
MOVWF t_high ;
MOVF bps,W ; t_low を設定
call v_low ;
MOVWF t_low ;
MOVF CMCON,W ; COUT の状態合わせ
BCF PIR1,CMIF ; 割込フラグを落とす
BSF INTCON,GIE ; 割込みを許可
sing DECFSZ t_high,F ; 同期に対応する期間
goto sing ;
tol BCF INTCON,GIE ; 割込み禁止
BCF GPIO,GP5 ; 強制的に GP5 を'L' に
ling DECFSZ t_low,F ; 'L' 期間
goto ling ;
BSF GPIO,GP5 ; クロックパルス発生
goto loop ;
v_high ADDWF PCL,F ;t_high 変換ルーチン
RETLW 0x82 ; 1200bps
RETLW 0x3d ; 2400bps
RETLW 0x19 ; 4800bps
RETLW 0x0a ; 9600bps
v_low ADDWF PCL,F ;t_low 変換ルーチン
RETLW 0x8b ; 1200bps
RETLW 0x45 ; 2400bps
RETLW 0x23 ; 4800bps
RETLW 0x0f ; 9600bps
END ;

```

以上