

Automatic Switching of Filters – Brooks Shera’s GPS-DO Yoshiyuki Takeyasu / JA6XKQ

フィルタの自動切換え Brooks Shera s GPS-DO 武安義幸 / JA6XKQ

2005 年から 2006 年にかけて自作した GPS-DO (GPS Disciplined Oscillator = GPS 同期オシレータ) の PLL ループ・フィルタの時定数切替えを、手動から自動切換えに変更したので、備忘録として残す。

はじめに

Brooks Shera / W5OJM による GPS-DO [1] を自作し [2] ,測定器や X-band (8.4 GHz) ダウンコンバータ等の周波数リファレンス信号源として使用してきた。GPS-DO の電源を使用時のみ通電していたので、電源 ON 直後から 2~3 時間程度は、OCXO オシレータの初期変動に応じて GPS-DO の PLL ループ・フィルタの時定数を手動で切替える必要があった。GPS に同期した高安定度な基準周波数を得るには、ループ・フィルタの時定数を長い値とするが、一方、通電直後の OCXO の急激な周波数変動に PLL が追従するには、ループ・フィルタの時定数を短くしなくてはならない。Brooks Shera の GPS-DO では、7 つのループ・フィルタ時定数が選択可能となっている [1] 。使用している OCXO [3] の安定度の特性から、時定数が短いほうから 1~3 のフィルタを、凡そ 30 分 ~ 1 時間毎にサムホイール・スイッチで切替えるが、連続通電しておくほどの使用頻度でないものの、ある程度使用頻度が上がってくると、毎回の手動切替えが面倒になってきた。そこで、GPS-DO が出力する位相誤差 (Phase Error) をマイクロプロセッサでモニタして、位相誤差の変化具合に応じてマイクロプロセッサがループ・フィルタ時定数を切替えるようにした。

位相誤差モニタとフィルタ切替え

オリジナルの設計では、マイクロプロセッサ PIC16C73 が GPS-DO の制御を行い、サムホイール・スイッチの設定を読み込んで、ループ・フィルタの時定数を切替える。また、制御の基本となる位相誤差と、OCXO オシレータの周波数を調整する DAC (Digital-to-Analog Converter) の値を、RS-232C シリアル通信で ASCII 文字列として出力する機能を備えている。オリジナルの GPS-DO には表示器がないので、GPS-DO の動作状態を PC でモニタするための機能である。

9600 bps のシリアル通信で、位相誤差、DAC 値、および、フィルタ値を次のようなスペース区切りの ASCII 文字列で、30 秒毎に出力する。

```
...  
00820 64485 00003  
00811 64417 00003  
00778 64151 00003  
...
```

追加のマイクロプロセッサ

今回は、このシリアル通信で出力される位相誤差情報を PIC マイクロプロセッサとは別に設けたマイクロプロセッサで読取り、位相誤差の変化具合を計算して、サムホイール・スイッチを代行する。

マイクロプロセッサとプログラム開発環境には、Arduino Uno 互換ボードと Arduino IDE を用いる。シリアル通信の読取りは、PIC16C73 の TTL 出力を Arduino のソフトウェア・シリアル・ポートに取込む。サムホイール・スイッチの代行は、オープン・ドレインの FET をバッファとして用いて、マイクロプロセッサの DI/O (Digital Input/Output) で ON/OFF する。

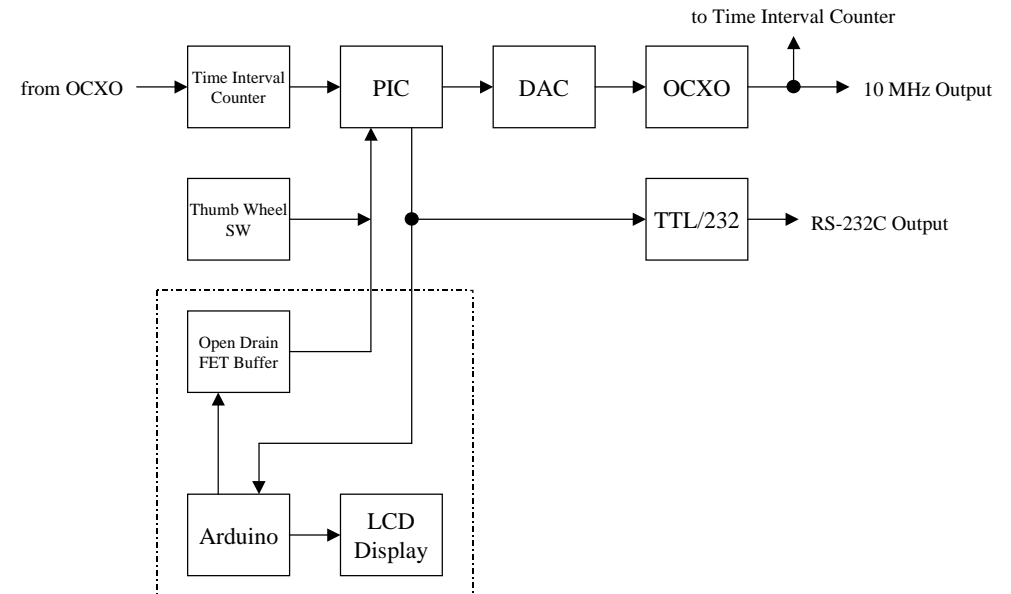


図-1 : ブロック・ダイアグラム

上記の構成を 図-1 のブロック・ダイアグラムに示す。

フィルタ切替えのアルゴリズム

フィルタを手動で切替えた時の位相誤差の変化の様子を 図-2 に示す。グラフの横軸は時間経過を示すが、目盛り値に 30 秒を乗じた値が秒数を示す。すなわち、ひと目盛りは 30 分を示す。

位相の安定具合に応じてフィルタを切替える訳であるが、「位相の安定具合」を言い換えると、「ある期間内で」「ある範囲内の位相変動に」「より長く収まっている」という、瞬時値ではない統計的な指標を用いてフィルタを切替えることとなる。

本プロジェクトと同様な例として、PIC マイクロプロセッサを用いた Brooks Shera / W5OJM 版を Arduino に移植した Jeffrey Anderson / K6JCA 版 [4] がある。そこで行っているフィルタの自動切換えは、フィルタの時定数（原文では settling time としている）に応じた時間後に、位相誤差がある範囲内に収まっていればフィルタを切替えるという、位相誤差が収まっているであろう時期（settling time）を見計らって瞬時値をチェックするアルゴリズムとなっている。

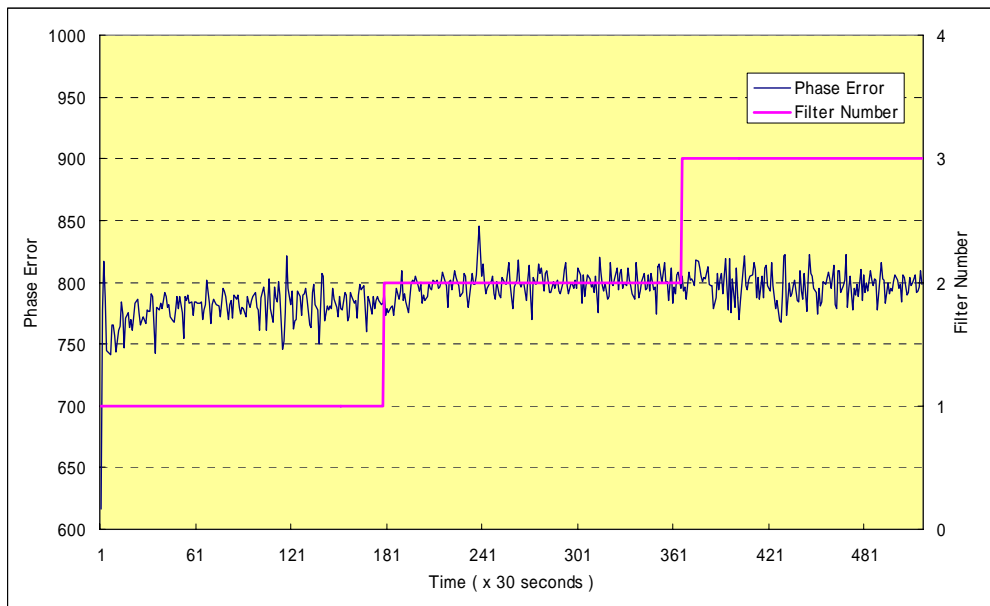


図-2：手動切替え時の位相誤差の変化

「位相の安定具合」を統計的/数学的に取り扱う指標として Allan Variance（アラン分散）あるいは、その平方根の Allan Deviation（アラン偏差）がある。本 GPS-DO の性能を評価する際には [5]，収集した位相誤差データを AlaVar [6] なるツールを用いて事後処理を行い、Allan Deviation を求めた。Allan Deviation の計算は専ら事後処理であると理解していたが、位相誤差データを収集しつつ Allan Deviation を同時に計算してプロットする、John Miles / KE5FX による TimeLab がある [7]。これに倣って、Allan Deviation あるいは、Allan Variance をリアルタイムに計算してフィルタを切替えることも可能であろう。Allan Variance の計算は四則演算のみなので、マイクロプロセッサでの計算も容易だと想像する。しかし、計算には複数個のデータが必要であり、フィルタ切替えの判定に必要なデータ数の見積もりを怠り、マイクロプロセッサのメモリ容量に関して漠然とした不安を感じたので、手動切替えで行っている「人の感覚」をインプリメントする安易な手法とした。

「人の感覚」とは、冒頭にあるように「ある期間内で」「ある範囲内の位相変動に」「より長く収まっている」である。「ある期間内で」をアルゴリズムとして記述すると、マイクロプロセッサの PIC は 30 秒毎に位相誤差を出力する/サンプリングするので、サンプリングの回数が時間の長さとなる。「ある範囲内の位相変動」とは、GPS-DO は位相誤差 = 800 を目標値として制御を行うので、 $| \text{位相誤差データ} - 800 |$ （ $| |$ は絶対値を示す）である。「より長く収まっている」とは、前述の $| \text{位相誤差データ} - 800 |$ が閾値以下となっているサンプリングをカウントし、閾値以上であればカウントを減ずる。位相の安定具合に応じて前述のカウント数は増減し、カウントがフィルタの時定数に対応した回数に達すれば「より長く収まっている」と判定して、フィルタをより長い時定数に切替える。これは、安定度をサンプル値の出現頻度に換算するアルゴリズムと言える。

プログラム

具体的な数値を用いてプログラムの概要を示す。各数値は、フィルタ手動切替えでの位相誤差の挙動を基に決定した。

PIC が出力する位相誤差データを PhaseCount として、位相変動具合を示す PhaseError を次のように定義/計算する。

$$\text{PhaseError} = \text{abs}(\text{PhaseCount} - 800)$$

使用している OCXO 単体の周波数安定度特性から、フィルタは 1 ~ 3 の三種類を使用する。4 以上のフィルタでは単体での安定度が不足していることが理由である。「ある範囲内の位相変動に」という、各フィルタで期待する安定度(位相誤差)に収まっている状

態を示すカウント値(頻度)を, state1, state2, state3 とする. 期待する安定度は,

- Case-1 state1 : PhaseError >=50
- Case-2 state2 : 25 <= PhaseError < 50
- Case-3 state3 : PhaseError < 25

PIC が出力する位相誤差データが Case-1 の状態では, state2 と state3 カウンタを 1 減算する. ただし, 減算の結果が負となる場合は, 値をゼロとする. 言い換えると, Case-1 では, state2 と state3 がともにゼロとなっている. Case-2 の場合は, state2 カウンタを 1 加算する. ただし, 加算の結果が 10 を超える場合は, 10 を上限値とする. また, state3 カウンタは 1 減算する. ただし, 減算の結果が負となる場合は, 値をゼロとする. Case-3 の場合は, state3 カウンタを 1 加算する. ただし, 加算の結果が 20 を超える場合は, 20 を上限値とする.

「ある期間内で」「より長く収まっている」という時間軸を示す変数として state1, state2, state3 を定義しているが、「より長く」という指標はフィルタ毎に異なった値とする. これは, フィルタ -1 ~ -3 の時定数が -1 ~ -3 で二倍ずつ長くなっていることに対応している. フィルタ 1 である条件は,

state2 == 0 && state3 == 0

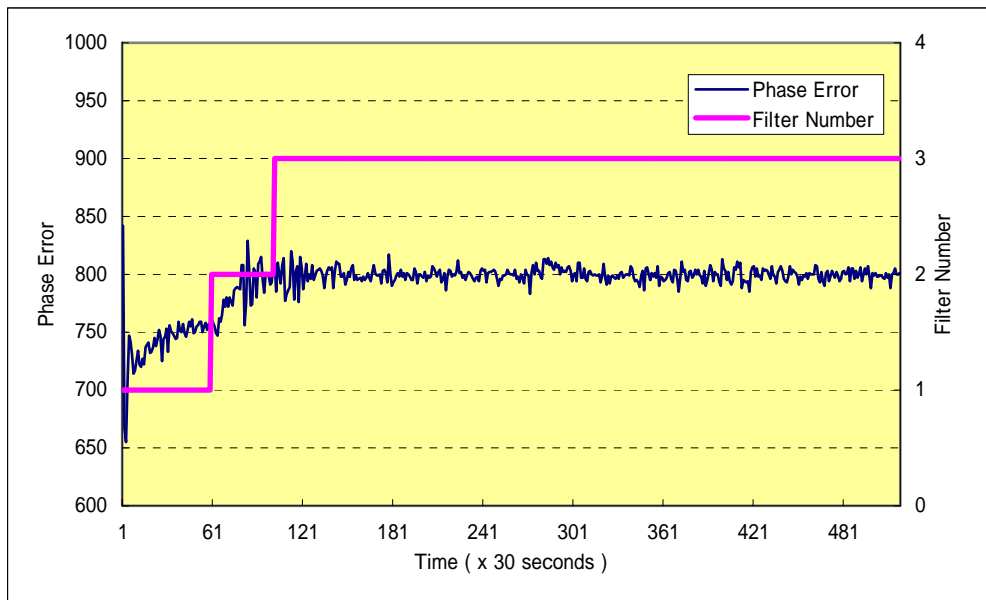


図-3 : 自動切替え時の位相誤差の変化

とする. 電源 ON 時のプログラムとしての初期値は state2 と state3 共にゼロであるので, フィルタ 1 から起動する. 自動切替えの過程で位相誤差が悪化して Case-1 の状態となれば, フィルタ-2 あるいは, フィルタ -3 からでもフィルタ 1 へと切替わる. フィルタ 2 である条件は,

state2 == 10 && state3 < 14

とする. フィルタ 3 である条件は,

state3 == 20

とする. ここで, フィルタ 2 である条件に “ state3 < 14 “ を加えているのは, フィルタ -3 から 2 への切戻しにハンチングが発生することを避けるためである. この条件 (いわゆるヒステリシス) がないと, フィルタ-3 の状態で Case -3 の条件を一回でも満足しないと, すなわち, state3 カウンタが 19 になるとフィルタ -2 へ切替り, その直後に Case 3 の条件を満足すればフィルタ 3 へ戻るというハンチングが, 検討過程で発生したので追加した.

フィルタ自動切替えの結果

フィルタを自動で切替えた時の位相誤差の変化の様子を 図-3 に示す. この例では, フィルタがスムーズに 1 から 3 へと順次切替わり, OCXO の安定度が落ち着いていく様子が分かる.

一方, 図-4 は, 位相誤差が急激に変化する, いわゆる位相ジャンプ時のフィルタ切替えの様子を示した例である. 位相ジャンプを生じた原因はさておき, 急激な位相変化に対してフィルタ切替えのアルゴリズムが適切に働き, フィルタを時定数が短いフィルタ 1 あるいは 2 へ切り戻している様子が分かる. フィルタ番号が途中で 4 以上を示している (値としては 101) のは, 時定数が最小のフィルタ 1 でも追従できない状態となり, GPS-DO としての PLL がアンロックとなっているためである. この状態においてフィルタ切替えのアルゴリズムはフィルタ 1 を維持して, 位相誤差が小さくなればフィルタ 3 へ切替えている.

まとめ

経験則から、およそ 30 分 ~ 1 時間の間隔でループ・フィルタの時定数を切替えれば GPS-DO の PLL としてのロックを維持しつつ、OCXO の位相安定度を高めていくことができていた。この経験則を基にフィルタの切替を自動化した結果、急激な位相変動にも対応しつつ、最短の時間で OCXO を高安定度な状態とすることができた。GPS-DO としてオリジナルな構成のまま追加機能/回路として付加したので、オリジナルの性能を損なうことなく、機能を改善することができた。

//

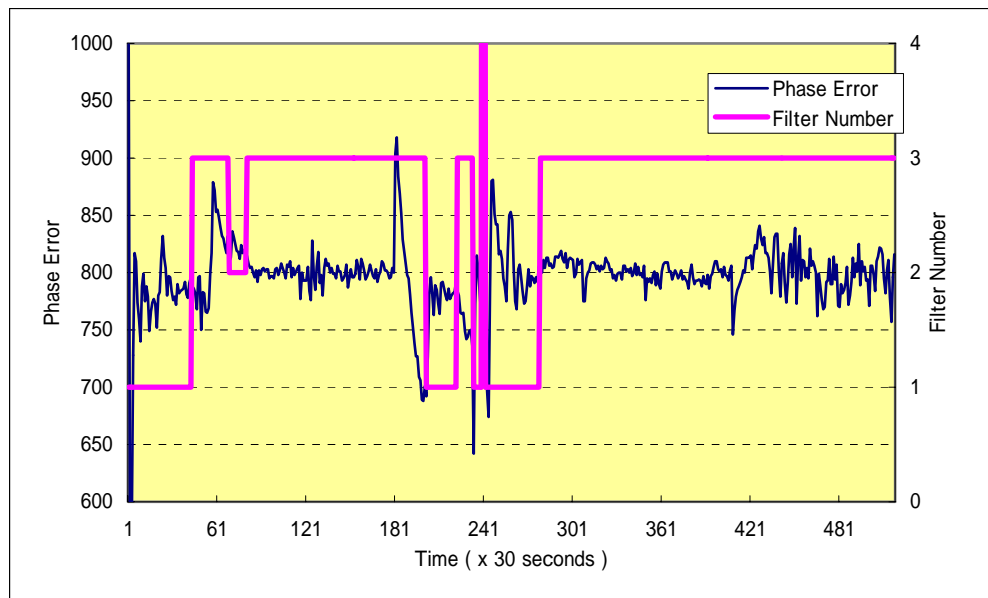


図-4 : 自動切替え時の位相誤差の変化 位相ジャンプあり

参考文献

- [1] SHERA, Brooks. W5OJM. "A GPS-Based Frequency Standard," QST, July 1998.
- [2] 武安義幸, JA6XKQ, "GPS 同期オシレータ 製作編," 4 March 2006.
http://www.terra.dti.ne.jp/~takeyasu/GpsOsc_2.pdf
- [3] "10811A/B Quartz Crystal Oscillator," HEWLETT-PACKARD COMPANY, August 1980.
<http://hparchive.com/Manuals/HP-10811AB-Manual.pdf>
- [4] ANDERSON, Jeffrey. K6JCA, "An Arduino Version of Brooks Shera's GPSDO," Feb. 6, 2019.
<https://k6jca.blogspot.com/2019/02/an-arduino-version-of-brooks-sheras.html>
- [5] 武安義幸, JA6XKQ, "GPS 同期オシレータ 測定データ編," 25 June 2006.
http://www.terra.dti.ne.jp/~takeyasu/GpsMsrData_1.pdf
- [6] MAKDISSI, Alaa, "ALAVAR: ALLAN VARIANCE SOFTWARE."
<http://www.alavar.org/>
- [7] MILES, John. KE5FX. "TimeLab.."
<http://www.miles.io/timelab/beta.htm>