

RTK コアはトランジスタ技術 2018 年 1 月号で公開したフリーのソースコードです。今回はその RTK コアに対して、u-blox 社 F9P の 2 周波に対応する改修を行いました。ここでは、追加部分のみ簡単に述べます。ソースコードの詳細は 2018 年 1 月号をご覧ください。

このソースコードは Visual Studio2010 で開発しました。Visual Studio2017 でも動作することは確認済みです。その際プロジェクトのプロパティで、プラットフォームのツールセットの変更が必要です。

今回追加した関数は 6 つで、基本的な動作は改訂前と同じです。追加した関数は以下です。

- ① calc_rtk_GQ_F9P→GPS と QZSS の 2 周波観測データより RTK 測位
- ② calc_rtk_GQE_F9P→GPS と QZSS、Galileo の 2 周波観測データより RTK 測位
- ③ calc_rtk_GQB_F9P→GPS と QZSS、BeiDou の 2 周波観測データより RTK 測位
- ④ calc_rtk_GQEB_F9P→GPS と QZSS、Galileo、BeiDou の 2 周波観測データより RTK 測位
- ⑤ check_cn→信号レベルでのチェックを少し厳しく行います
- ⑥ w_inverse_float2→基本的に前回の w_inv 関数と同じで二重位相差の重みを計算

上記の追加関数以外には、擬似距離の DGNSS を改訂前の二重位相差方式ではなく、擬似距離補正方式に変更しました。main 関数の基準局側の単独測位演算の後、Correction という変数に全衛星の補正データを格納し、移動側の測位演算で DGNSS の補正データとして利用しています。main 文内の移動側の処理を見るとわかるように、前回の 1 周波観測データの RTK の関数には入らないようにしています。もちろん利用もできます。

2 周波観測データ用の RTK は、最初に GPS と QZSS、Galileo、BeiDou の 4 つの測位システムでの計算を行い、FIX 解を得られなかった場合、順番に GQB、GQE、GQ の組み合わせでの RTK 測位を実施します。もし choose_sat 関数後に利用できる測位システムが GPS と QZSS のみだった場合は（ただし 5 機以上としています）、最初から GQ の組み合わせに入ります。GLONASS については動作確認が遅れており追加していません。choose_sat 関数では、2 周波用としてチェックをしています。もし 1 周波用として利用する場合は、L2 帯の信号チェックをはずす必要があります。

動作確認に使えるデータ・サンプルを付加しました。Rtk フォルダの下にあります。

20181107 フォルダには、研究室屋上で取得した 1m 程度の基線の 2 つの 1 時間データを準備しました。initial_setting テキスト内のデータ読み込み部分の 3 行のフォルダ名を 20181107 にすると、そのまま動作します。見通しの良い屋上で取得したデータなので、もちろん 100%FIX します。

20181115 フォルダには、海洋大周辺の移動体と基準局の約 30 分データを準備しました。initial_setting テキスト内のデータ読み込み部分の 3 行のフォルダ名を 20181115 にすると、

そのまま動作します。約 80%程度の **FIX** 率になります。なお、このとき同時に **u-blox** の **F9P** によるリアルタイムの **RTK** も実施しており、その **FIX** 率は 90%を超えていました。2 周波の観測データがあると、このように通常都市部でも 90%を超える **FIX** 率を得ることができます。プログラムを改良していけば、みなさんも 90%を超える **FIX** 率を達成することができます。例えば、フィルタ手法で **FLOAT** 解の精度を良くしたり、品質の良い衛星を選択したり、**FIX** した二重位相差のアンビギュイティを保持したりなど、改良する余地が残されています。

u-blox の **F9P** 受信機の 2 周波観測データは、**RTKLIB** の 2.4.3b31 の **RTKCONV** で変換できます。添付したデータも **RTKCONV** で変換したものです。他の受信機の 2 周波観測データも、読み込み部分さえ少し付加すれば、利用できるはずですが、ただし、異機種を受信機間では対処すべき事項が何かあるかもしれません。

出力結果ファイルについて以下にまとめました。中の詳細フォーマットはソースコードを見るとすぐにわかります。

dgnss.csv→擬似距離補正方式の **DGNSS** の結果を出力

pos.csv→基準側の単独測位結果を出力

rtk.csv→**RTK** の結果を出力

test.csv→移動側の **choose_sat** を通過後の衛星数とそれぞれの測位システムの数を出力

rtkplot.pos→**RTKLIB** の **RTKPLOT** で水平結果 (**Gnd Trk**) をチェックできます

注意

中国の **BEIDOU** や欧州の **GALILEO** 衛星はともに 30 機を越える予定で、このプログラム中であらかじめ確保した衛星番号が足りなくなっています。**GALILEO** 衛星はすでに 31 番があり、**BEIDOU** の 1 番と重なるため（このプログラム内では両方とも 71 番になってしまう）、強制的に計算から排除しています。