

PICと最小限の部品を使った周波数計

Wolfgang "Wolf" Buscher, DL4YHF

最終更新 2008-11-10

訳者注：この PDF からのリンクは設定されていません。

この文書は安価な PIC といくつかの 7 セグメント LED を使った小型周波数計の製作について記述しています。

周波数計の主な仕様は以下の通りです。

- ・ 周波数範囲 1 Hz ~ 50 MHz (試作品では 60 MHz 以上まで動作しましたが、これは PIC のタイミング仕様を超えている)
- ・ 4 または 5 桁の表示 (例 x.xxx kHz x.xxx MHz または xx.xx MHz)
F8FH の改造では 6 桁
- ・ ゲートタイムの変更による自動レンジ切り替え
- ・ 加算、減算による任意の周波数オフセット (プログラム可)
- ・ 任意の入力プリアンプ
- ・ ブレッドボードタイプの片面基板に製作ができる。
- ・ アノードコモンと同様にカソードコモン LED に対応したファームウェア
- ・ たいへんに少ない部品数

PIC 16F628

4 または 5 個の 7 セグメント LED

4 または 20 MHz の Xtal 発振子 と いくつかの抵抗

5 桁にするためのトランジスタといくつかのダイオード (任意)

プリアンプ用トランジスタ (任意)

- ・ 2006 年 5 月以降：プログラム済み IF 4.0 MHz のトランシーバ周波数オフセット (40m QRP トランシーバ "Miss Mosquita" の製作者の実践をご覧ください)
- ・ 15 秒以内に周波数が変動しなかった場合、表示をオフにするパワーセーブモードは任意に設定可能。

この文書の内容

- ・ 製作例 1 (私の最初の試作品)
- ・ 製作例 2 (2 番目の試作品、レイアウト、回路)
- ・ 5 桁の表示 (NAND ゲートを使って 任意)
- ・ 電力消費
- ・ 周波数表示レンジ (自動レンジ切り替え)
- ・ オフセットの加算、減算の仕方
- ・ どのように動作するのか - ファームウェア機能説明
- ・ 他の人々による周波数計へのリンク (すばらしい思いつきのために)

PIC のファームウェアはこちらからダウンロードできます。(アノードコモン、カソードコモンに対応した h e x ファイル、ソースコードを含む)

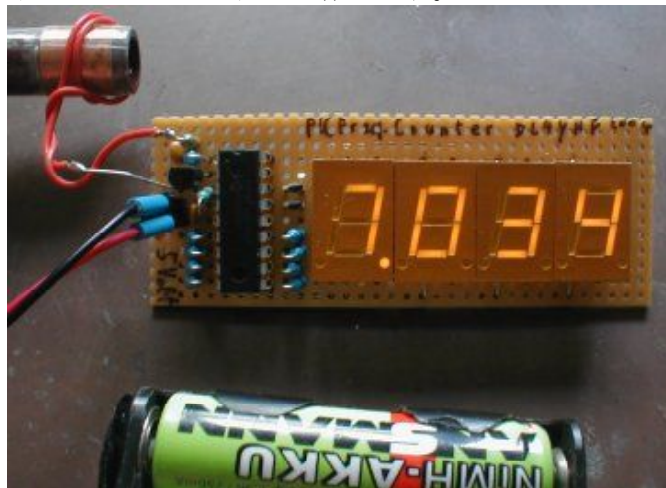
あなたの P I C 1 6 F 6 2 8 に書き込むために必要な PICprogrammer は DL4YHF のウェブサイトで入手可能です。

私にプログラム済み P I C を送ってほしいとか、回路基板を送ってほしいと言うことはご遠慮ください。私の時間的余裕はほとんどありません。これら自作のページの中であなたの欲することは見つけれられると思います。

もし、最初の段階であなたの作った周波数計が異なる周波数を表示するようでしたら、” setup mode ” に入り周波数オフセットをゼロに設定してください。(周波数オフセットを収納している EEPROM の消去ができていないことがときどきあります)

・製作例1

たいへん古い、低効率の 7 セグメント表示試作品です。

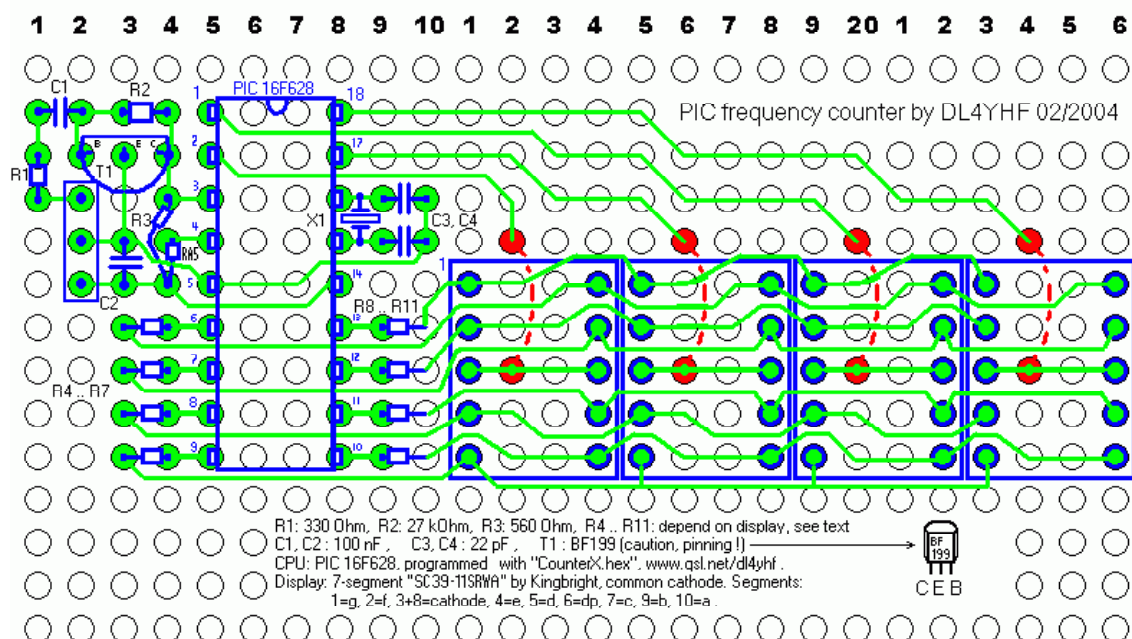


(グリッドディッパーにつないだ P I C 周波数計試作品)

点滅する小数点はキロヘルツを示しています。連続点灯の場合にはメガヘルツを示します。この方法はディップメーターや QRP トランシーバで普通に使われるようになっています。

配線図はたいへん簡単ですので、私は Eagle を使った作図はまだ行っていません。(他の製作者のやっているのはこちらをご覧ください)

最初の試作は丸いパッドのブレッドボード上に作りました。片面基板を使い、たいへんシンプルな配線図になっています。4つのブリッジ(下の図で赤で示した)は 7 セグメント LED を取り付ける前に配線しなくてはなりません。水晶発振子は裏側の配線面に取り付けます。特に、2つのパッドの間を通る配線など、いくつかの 7 セグメント LED への配線は細いエナメル銅線で行います。



(この回路には、ファームウェアは” counter1.hex ” 4 MHz 水晶発振子を使用)

抵抗 R4 ~ R11 は明るさとパワー消費調整に使用しています。もし、あなたが適当なローパワー 7セグ LED を見つけたのなら 1 k Ω かそれ以上の値のものを使ってください。古い LED ですとより多くの電流を消費します。そのような場合には 390 Ω を使います。PIC の一つのピンには入出力で 20 mA 流すことができます。そのため 7セグ LED に ” 8 ” を表示させた場合、それぞれのセグメントに 390 Ω を接続すると駆動電流はわずかにオーバーロードになってしまいます。電流消費量に注意せず、電流を大食いする古い LED を使う場合は、カソード側に PNP トランジスタを非反転駆動として使います。ベース抵抗は使いません。(ベース = PIC、コレクタ = GND、エミッタ = コモンカソード)

Kingbright の ” SC39-11SRWA ” にはドライバ・トランジスタは不要です。これら高効率ディスプレイは各セグメントに 1 k Ω の電流制限抵抗をつないでも十分な明るさがあります。ドイツではこれら高輝度、高効率ディスプレイを Reichelt に発注することができます。少なくともこれまでは発注できました。カタログの中から ” SC39-11RT ” を探してください。(実際には、超高輝度、高効率の SC39-11SRWA ディスプレイになるでしょう。SC39-EWA のような低効率のタイプは使わないでください。上記の抵抗値では暗いディスプレイになってしまいがっかりするでしょう。)

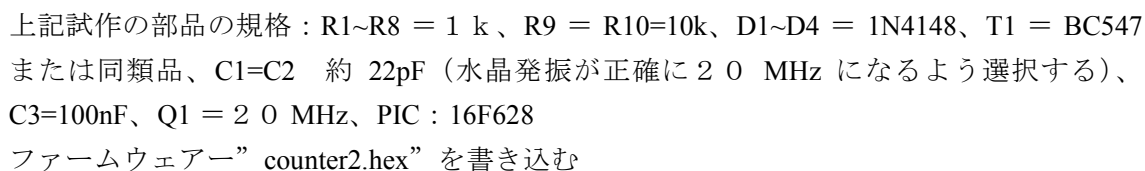
ご注意 他の電子部品販売店 (C のような) ではこのようなディスプレイはより高価です。

製作例2

2回目の試作品では、5桁のディスプレイを下のような一層の PCB で作りました。セグメント抵抗はディスプレイボードの後ろ側に 90度の角度で取り付けられた PIC の乗ったボードとの接続に使われています。

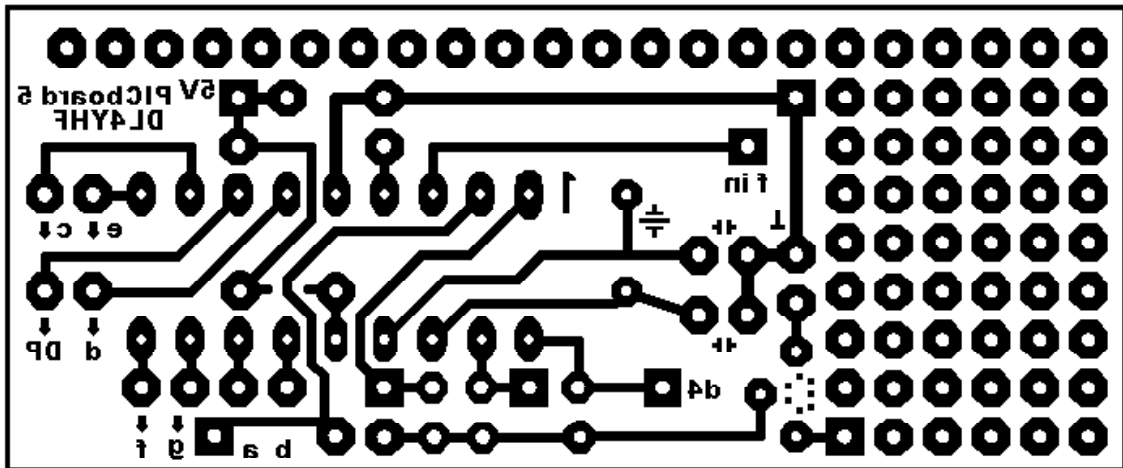
PIC の載ったボードとディスプレイボードを容易に接続するため、2番目のファームウェア

PIC ボードの回路図 (ディスプレイ試作 2)

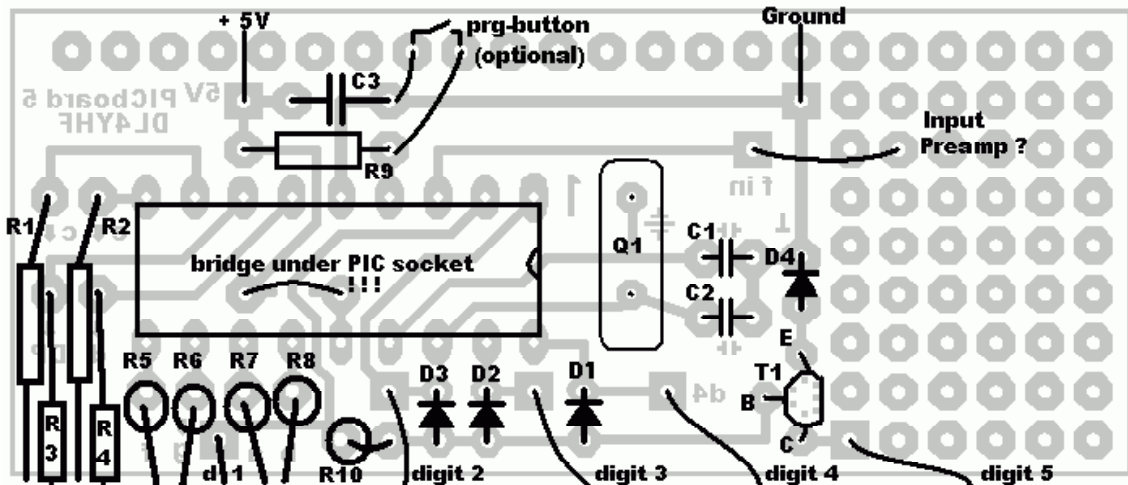


この基板には5桁のデコーダーと必要な場合に使用するプリアンプエリアが設けられています。このイメージを Irfan View などのユーティリティを使って正確に 300dpi でプリントします。このイメージは 700 * 300 pixels の大きさですのでボードは 2.33 * 1 inch または 5.93 * 2.54cm (エッチングの前にここをクリック)

- 4 -

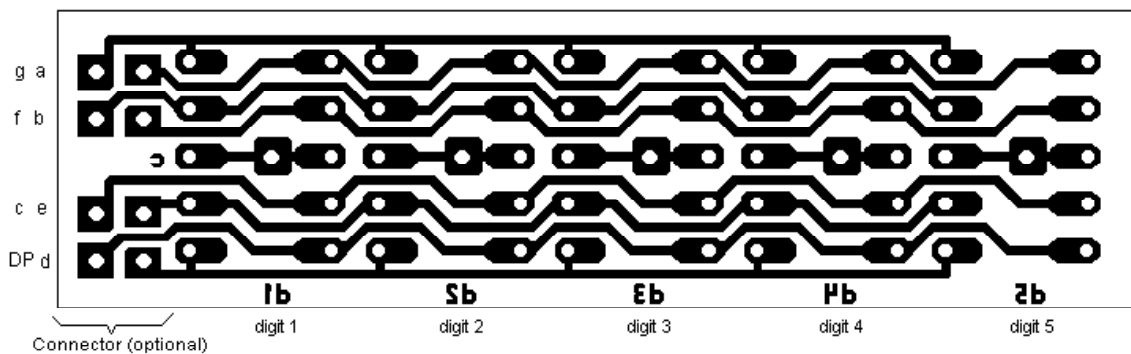


PIC ボード上の部品配置。ぱかぱかしいほど単純だと思いませんか？でも、半田付けの前に PIC ソケットの下のブリッジをつなげるのを忘れないでください！



これはディスプレイボードのレイアウトについての一つの提案です。(試作2 これも正確に 300dpi でプリント)

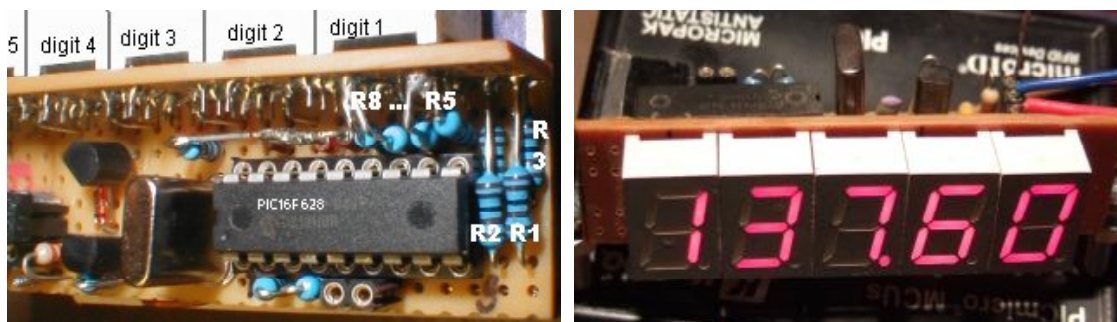
7-segment display, 5 digits SC 39-11 for DL4YHF's PIC frequency counter. Bottom layer, X-ray-view from TOP SIDE (not mirrored!)
Print with exactly 300 dpi.



このボードはたいへん多用途です。私はこれを PIC を使った DDS ジェネレータにも使う

つもりです。周波数計にはコネクタパッドの [g] [a] [f] [b] は使いません。これら4つの抵抗 R5～R8はPICボードから直接、後ろ側で7セグメントに接続しているからです。（下の写真参照）左側の文字はセグメントを示します。5つのカソードへの配線はディスプレイPCBの脇は通りません。このような小さな一層PCBでは難しいのです。小さなワイヤーでディスプレイボード上の方形カソードパッドとPCBボード上のd1～d5方形パッドを接続します。

フロントパネルのスペースを確保するため、PICボードとディスプレイボードの接続をしないようにします。そしてPICボード上のR5～R8を一番左端のセグメント（第1桁）のピンへ半田付けします。R1～R4はディスプレイボードの下側のパッドへ半田付けします。PICがラスターボードに取り付けられた第2試作品の写真をご覧ください。



抵抗を用いたCPUボードとディスプレイボードの接続はすてきだと思いませんか。

よりうまい方法があればお試しください。（他の方の実践はこちら [link](#)）

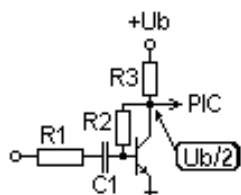
この試作品は Miss Mosquita と呼ばれる小さな QRP トランシーバにぴったりです。

コモンカソードまたはコモンアノードディスプレイ

上記の2つのディスプレイ試作（COUNTER1 と COUNTER2）はコモンカソード（CC）の7セグメントディスプレイでした。しかし、ダウンロードできるソフトウェアにはコモンアノード（CA）用の第3のファームウェア（COUNTER3. HEX）も含まれています。

COUNTER3 は COUNTER2 と同じピン接続を使っていますが、コントロール出力は逆になっています。回路では5桁目を制御する T1 トランジスタが NPN の代わりに、CA ディスプレイでは PNP トランジスタを用います。そのうえ、D1～D4 の接続が逆の極になるようにします。また、D4 に接続されるのは GND の代わりに Vsupp（正電圧）になります。

プリアンプ



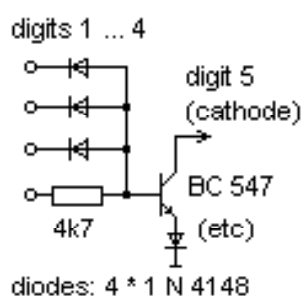
プリアンプはシングル HF シリコントランジスタで成り立っています。私はジャンク箱の中から安価な BF199 を使いました。これで 30 MHz 以上でもよく働き、50 MHz 以上では感度が低下します。コレクタから PIC のカウンター入力（pin3 = T0CKI）への接続はできるだけ短く行います。なぜなら配線上の pF 容量をカウントしてしまうからです。

プリアンプに BF199 を用いては、R2 は 27k Ω (+/-) R3 はバンド幅に応じてできるだけ 560 Ω より小さくします。R2 を調整してコレクタの電圧を $V_b/2$ となるようにします。

基板上のスペースを節約するため、プリアンプは取り去ることができます。TTL レベルのシグナルを入力とするときにはプリアンプは取り去ります。これにより 4mA 節約することができます。測定周波数が最大で 10MHz 以下ならば、R3 と R2 を同係数で増やします。(例 R3=1.2k Ω 、R2 = 56k Ω) 電池駆動でカウンタを使用する場合、幾ばくかの消費電流を減らせませす。R1 は入力インピーダンスと感度を調整します。R1=330 Ω のとき、試作機では 40MHz で入力電圧は 600mVpp (ピーク＝ピーク) 必要であり、15MHz では 150mVpp 必要でした。より高い入力レジスタンスが必要な場合には、バイポーラートランジスタの前に FET バッファを加えます。または、あなたのジャンク箱の中から見つけた、高速コンパレータを入力ステージとして使います。

2006 年 4 月追記 高入力インピーダンス・プリアンプを DL-QRP-AG のページで見つけることができます。

5桁めの駆動(オプション)



PIC のファームウェアは測定周波数が 10MHz 以上の場合には最小の桁である 5 番目の表示も駆動します。PIC16F628 にはもう一つの桁を駆動するための空いた出力ピンがありませんから、既に使われている 4 つの出力ピンから複合的に取り出した信号を論理的に組み合わせて使うことになります。

すべての桁の複合出力が受動のとき (1 ~ 4 桁 = high)、追加の 5 桁めが駆動されます。左図の回路図は 5 桁めを駆動する「解読器」の例です。これは基本的に、いくつかのダイオードと 1 つの抵抗、NPN トランジスタから構成された 4 入力 NAND ゲートです。エミッタと GND の間にダイオードが使われています。これは、このステージで他のダイオードのフォワード電圧 (1N4148 ではおよそ 0.6V から 0.7V) よりベース＝エミッタ間のスレッシュホールド電圧 (通常 0.5V から 0.6V) が少ない場合、このダイオードがないと不必要に導通してしまうためです。

ジャンク箱の中から BAT41 (または同類) を見つけるよりも 1N4148 の方が見つけやすいでしょうが、このエミッタダイオードとして、その他の低フォワード電圧のショットキーダイオードを使うことができます。

もし必要なければ、5 桁めを取り去れば多少の電流と部品の節約になります。同じ理由から「単独ゼロ」(入力信号がない場合) には 4 桁めは表示されますが 5 桁めは表示されません。

電力消費

試作機 (R4~R11=390 Ω) は平均電流が 40mA 流れました。(低効率の 5 桁で) 高効率、

低電流のディスプレイなら、これははっきりと現れます。セグメント抵抗が 1k Ω なら全体のディスプレイ電流は 20mA を下まわります。(5桁の”SC39-11”で) PIC 自体には 20MHz において約 4mA 流れます。4MHz で動作の時には 1mA 以下です。すなわちプリアンプにはコントローラ自体よりも多くの電流が流れることになります。

ディスプレイ・レンジ

ディスプレイ・レンジは最大精度表示を行うよう自動的にレンジを切り替えます。(4桁で) 下に示す表のように、ゲートタイムは自動的に選択されます。

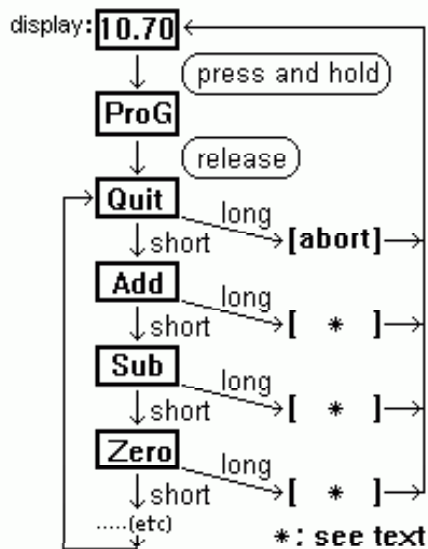
周波数レンジ	表示	ゲートタイム	小数点
0 ~ 9,999kHz	x.xxx	1 秒	点滅 kHz を示す
10 ~ 99.99kHz	xx.xx (x)	1/2 秒	点滅
100 ~ 999.9kHz	xxx.x (x)	1/4 秒	点滅
1 ~ 9.999MHz	x.xxx (x)	1/4 秒	点灯 MHz を示す
10 ~ 50.00MHz	xx.xx (x)	1/4 秒	点灯

(この場合 MHz はメガヘルツ、mHz はミリヘルツのことですが、別の話ですね・・・)

オフセット周波数の加算または減算

周波数計を短波受信機やトランシーバに組み込んで使用するとき、測定周波数からオフセット量を加えたり減じたりしたいでしょう。周波数計は普通、受信 VFO (可変周波数発振器) に接続されますから、オフセット周波数は多くのケースで中間周波数と同じです。このため、ファームウェアにはプログラムモード (またはセットアップモード) が組み込まれています。それで PIC のファームウェアを再度プログラムし直したり、アッセンブルし直したりせずとも、オフセット周波数を入れることができます。

RA5 (PIC16F628 の 4 ピン) の信号は通常周波数計モードからプログラムモードへの切り替えになっています。通常 RA5 のレベルはプルアップ抵抗 (10k または 22k) を通して電源電圧につながれていますので H (high) になっています。もしあなたがオフセット周波数をこれからのずっと加算・減算をしないのなら、これをずっと電源電圧に接続しておきます。(残念ながら内部にプルアップ抵抗がないので、こうして RA5 のレベルを規定します) RA5 を L (low) にする (PIC のピン 4 とピン 5 を小さなドライバーで接続する) ことによって、ファームウェアには測定した周波数を新たなオフセット量として取り込まれます。言葉を変えれば、周波数計の入力にオフセット周波数を入力し、それが正しく表示されるまで待って、下に説明するプログラミングモードに入ります。



左のプログラムの流れ図は、どのようにプログラムモードに入るか、どのようにメニューを選択するか、どのように関連する機能を実行するかを示します。

プログラムモードに入るためには、LED ディスプレーに PIC が” ProG”と表示されるまでプログラミングキーを押したままにします。【press&hold】（または PIC のピン4と5を小さなドライバーで接続する）

そうしましたら、キーを放します。これで最初のメニューに入ったことになります。

次のメニューを選択するために、キーを短時間だけ押します。（1秒以内）選択した機能を実行す

るためには、キーを長時間押します。（1秒以上）

メニューの機能は

- ” Quit” 何も変更せずプログラミングモードを出る
- ” Add” 前もって測定した周波数をずっと記録する。そして加算に用いる。
- ” Sub” 前もって測定した周波数をずっと記録する。そして減算に用いる。
- ” Zero” オフセット周波数をゼロにする。これによりオフセットがないので表示は測定周波数になる。以前に設定したオフセットは消去される。
- ” Table” Table に用意されたオフセット量の中から選択できるようになる。Table は PIC の EEPROM データの中に納められている。そのため、異なるオフセット量をその中に設定することができる。Table を眺めてみると、周波数は 455.0 (kHz)、4.1943 (MHz)、4.4336 (MHz)、10.700 (MHz) のように数値の形で納められていることがわかる。選択した後、エンター（キーの長押し）すると加算（Add）や減算（Sub）をするためにメインメニューに戻る。
- ” PSave” / ” NoPSV” パワーセービング on/off の切り替え。パワーセービングモードでは 15 秒間有効な周波数変化がないとディスプレイがオフになる。そして大きな周波数変化【a few dozen Hertz】（測定レンジが 3 ～ 4MHz において）があるとすぐにオンになる。2006 年 5 月、QRP トランシーバのようなバッテリー駆動機器のために追加された。

注：

ここで示したよりもっと多くのメニュー項目がありますが、原則は同じです。

周波数オフセット量は 32 ビットの整数値として PIC のデータ EEPROM に納められます。（EEPROM では最初の 4 つメモリーロケーション。high-byte が先、low-byte が後）もし、あなたがオフセット周波数を設定するための発振器を持っていないか、自作の短波受信機で BFO 周波数を動かす【tap】ことができないのなら、オフセット量を適当な PIC プログラマー（DL4YHF's WinPic のような）を使って入力することができます。

科学計算電卓を使って、周波数（Hertz、ポジティブまたはネガティブ）を 16 進数に変換します。この値を EEPROM メモリーの窓に入力します。もし WinPic を使うのなら、メモリー窓に入力する前に HEX エディターに入力します。

例	(はい、とても単純です)	
4.194304MHz	: Add= 00 40 00 00	Subtract= FF C0 00 00
4.433619MHz	: Add= 00 43 A6 D3	Subtract= FF BC 59 2D
0.455000MHz	: Add= 00 06 F1 58	Subtract= FF F9 0E A8
10.70000MHz	: Add= 00 A3 44 E0	Subtract= FF 5C BB 20

もし、減算したオフセット周波数が周波数計の入力周波数より高くなってしまった場合、減算の結果はネガティブになります。周波数計のカウンターは表示の前に結果をポジティブにします。この場合、 $f_{IF} = f_{RX} + f_{LO}$ または $f_{RX} = f_{IF} - f_{LO}$ の受信機において LO 周波数を加えた、RX 周波数を減じたものです。（カウンタが逆方向に動くのですが誤りではありません）

DL2YEO の 30m バンド QRP トランシーバを例にとると、 $f_{RX} = f_{LO} - f_{IF} = 14.314\text{MHz} - 4.194\text{MHz} = 10.120\text{MHz}$ 。これは内部で計算されたものです。（ f_{Lo} は測定された入力として、 f_{RX} はディスプレイ数値として、 f_{IF} はプログラムされたオフセット値となっています）

もし 10MHz 台の表示が不要なら、-4.194MHz の代わりに -14.194MHz をセットします。これにより、4 桁だけでより分解能のよい表示ができます。（受信機のチューニングレンジは 20kHz 程度ですから、 $f_{RX} = 10.120\text{MHz}$ が 120.0kHz となり、十分な表示です。）

ですからご自身で測定したり周波数を入力しなくても、よく使われる IF 周波数は Table メニューから呼び出すことができます。もしあなたが QRP の自作者で、標準的な中間周波数 Table に他の周波数を入れたい場合にはご連絡ください。

動作説明

このプログラムは最初の lamp テスト、プログラミングモード、ここでは説明していないパワーセービングモードの例外を除き、基本的にエンドレス・ループで動作しています。メインループは次のようなステップで進んでいます。

- 1 自動測定レンジ設定のために大まかな周波数測定を準備する：同期するプリスケアラは 64 で分周することで範囲外の周波数を除外する。（理論上は 64MHz、しかし、これは PIC の仕様を超える）
- 2 PIC の TIME0 モジュールのカウンターモードを使って 1/16 秒間、入力パルスのカウントする。
この間、ディスプレイのマルチプレクサは動作を続けている。カウントループは正確に 50 ミリ秒であり、マルチプレクスルーチンに含まれる。1/16 秒間に 1250 回のカウントループが行われる。ソースコードはサブルーティンの中で行われる。

- 3 ステップ2の測定結果に基づいて、どのプリスケラーを用い、どの測定間隔を使うかを判断する。
- 4 分周された測定周波数が 1MHz より低いのでカウンターのプリスケラーが再プログラムされる。(PIC のクロックが 4MHz の場合、ハードウェアカウンターの入力最高周波数は 1MHz である) 大まかに測定された周波数が 1MHz よりずっと低い場合、プリスケラーはよりよい分解能を得るためオフになる。
- 5 0.25、0.5、または1秒の測定間隔(ゲートタイム)の間パルスのカウントする。この間、ディスプレイのマルチプレクサは動作を続けている。8-bit timer register (hardware) のオーバーフローを他の2つの 8-bit register でソフト的にカウントする。これにより有効なパルスカウンタは24ビットになる。(8ハードウェアビット + 16ソフトウェアビット)
- 6 ゲートタイム終了 => パルスのカウント停止
- 7 カウントしているとき、ハードウェアのプリスケラーが働いていた場合(ステップ4参照)プリスケラーの係数がパルスカウントに掛けられる。そのため、これから後のステップにおいても、プリスケラーのセット状況については考慮しなくてよい。もしあなたがアセンブラ・プログラミングを少しでも知っているなら: マルチプリケータはいつも2つのパワーであり、パルスカウント数は(32bitに拡張されている)は左にシフトされPICにおいてより扱いやすくなる。
- 8 ゲートタイムが0.5秒だった場合、パルスカウントは2を掛けられる。ゲートタイムが0.25秒の場合はパルスカウントに4が掛けられる。どのプリスケラー係数であっても、ゲートタイムがいかようでも関わりなく結果として入力周波数は Hertz となる。前のステップのように、このマルチプリケーションはシンプルな bit-shifting 操作になっている。
- 9 (オプション) 周波数オフセットを行い、結果がネガティブになったとき、ポジティブに変換される
- 10 【この項目は訳者の守備範囲を超えているため訳不能】
- 11 前にあるゼロを飛び越えて、kHz または MHz の桁の後に小数点を挿入する。(kHz ポイントの場合は点滅動作も付加する)
- 12 ゼロでない桁の始まりで、2進コードから5桁の7セグメントパターンを生成する。そしてディスプレイレジスタに収納する。ディスプレイ複合ルーチンはカウントを

している間に LED ディスプレーに次のメインループのステップ 2 及び 5 で書き込みを行う。

- 1 3 プログラミング機能入力” RA5” この入力が L(low)の場合プログラミングモードに入り、そうでないならステップ 1 から次の測定が始まる。

トリッキーに思えますか？ あなたは内部の機能についてファームウェアを修正したいと思うまで理解しなくても大丈夫です。

他の人々による周波数計へのリンク

【これ以降の項目についてはコメントとリンクになっていますので訳はありません。】

////////////////////
たいへんシンプルで作りやすく使いやすい周波数カウンターです。国内の方でも作ってみようと思う方がいられると思い、紹介することにしました。

このpdf ファイルはできるだけ本物のページの体裁を尊重して作りましたが、リンクは張ってありません。本物のページにはたくさんの情報や資料へのリンクが張られていますので、是非訪問することをお勧めします。

DL4YHF Wolf さんのページ

http://www.mydrac.de/dl4yhf/freq_counter/freq_counter.html

注意して訳したつもりですが、内容の正確さを保証するものではありません。
不明の場合はWolfgangさんのページで確認をしてください。

2012.11.14

by Takayama "shig" Shigekazu, JA1XRQ