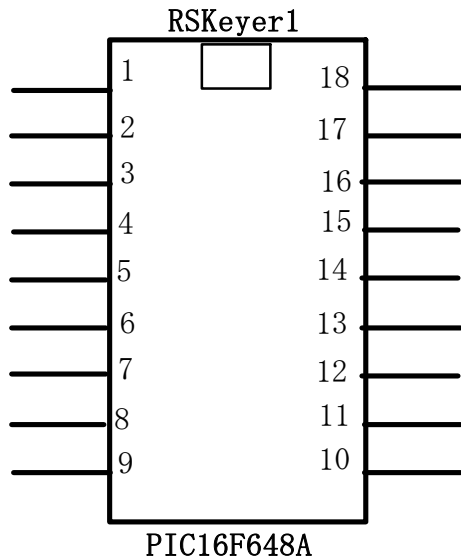


概要

RSKeyer1は、18ピンDIPタイプのIC(PIC16F648A)でシリアル入力より受信した文字列をモールスコードに変換してキーイングします。RS232C出力をもったPCと無線機の間において使用します。また、アイアンビックパドルを接続することによって短点メモリ付のエレキーとしても使用可能です。



PIN1:	[I]	Baud rate 9600/19200bps
PIN2:	[I]	Buzzer enable
PIN3:	[I]	Flow control
PIN4:	[I]	N/A
PIN5:		Vss (ground)
PIN6:	[O]	LED
PIN7:	[I]	RS232C IN
PIN8:	[O]	RS232C OUT
PIN9:	[O]	Buzzer
PIN10:	[I]	Paddle DASH
PIN11:	[I]	Paddle DOT
PIN12:	[I]	Speed down
PIN13:	[I]	Speed up
PIN14:		Vdd DC+5V
PIN15:		OSC (20MHz)
PIN16:		OSC (20MHz)
PIN17:	[O]	Keying
PIN18:	[O]	TX (PTT)

特徴

ワンチップマイコン(16F648A)を使ったエレキー

シリアルポートで受信した文字列をモールスコードに変換してキーイング

シリアルポートの受信バッファは80文字

シリアルポートからコマンドを送信することによって速度等の動作設定が可能。

RSKeyer1チップのピン詳細

Pin #	I/O	Description	説明	
Pin1	I	Baud rate	デジスイッチ等によってPCとの接続用RS232Cインターフェイスのボーレートを決定します。	0V:19200bps 5V:9600bps
Pin2	I	Buzzer enable	Pin9に接続されたブザーを有効にするか無効にするかを選択します。有効にするとキーイング時にPin9がHi(5V)になります。	0V:有効 5V:無効
Pin3	I	Flow control	RS232Cでフローコントロールを行う場合は有効にします。有効にするとRSKeyer1が文字列送信を始める際にbusyコードをPCに送信、文字列送信完了時にreadyコードをPCに送信します。	0V:有効 5V:無効 (フロー制御を有効にした場合に、キーイング中にPCより文字列を送信すると、最初の1文字がうまく受信できないケースがあります)
Pin4	I	N/A	未使用	
Pin5		Vss (ground)	0Vを接続します。	
Pin6	O	LED	動作確認用のLEDを接続します。キーイング時に長点、短点に応じて点滅します。	5V出力時にLEDが点灯するような回路構成にします。
Pin7	I	RS232C IN	RS232C受信部です。	通常はMAX232等のレベル変換ICを介してPCのRS232Cポートに接続します。
Pin8	O	RS232C OUT	RS232C送信部です。	通常はMAX232等のレベル変換ICを介してPCのRS232Cポートに接続します。
Pin9	O	Buzzer	ブザー用出力です。キーイング時に長点、短点に応じてブザーになります。	5V出力時にブザーが鳴るような回路構成にします。単純なON/OFF出力なので発信器内蔵のブザーを使用します。16F648Aの出力端子の電流量の制限から通常はドライブ用のTr(2SC1815等)を介してブザーを接続します。
Pin10	I	Paddle DASH	パドルを接続する場合のDASH用入力です。	0V DASH入力 5V 無入力
Pin11	I	paddle DOT	パドルを接続する場合のDOT用入力です。	0V DOT入力 5V 無入力
Pin12	I	Speed down	キーイングスピードを下げるための入力です。この入力がLoになるとキーイング速度WPMが1つマイナスされ、Pin2の入力に関係なくブザーが3回なります。	0V スピードダウン 5V 現スピード維持
Pin13	I	Speed up	キーイングスピードを上げるための入力です。この入力がLoになるとキーイング速度WPMが1つプラスされ、Pin2の入力に関係なくブザーが2回なります。	0V スピードアップ 5V 現スピード維持
Pin14		Vdd DC+5V	5V電源接続端子	レギュレータを介して5Vを供給します。
Pin15		OSC 20MHz	水晶発振子用端子	20MHzの水晶発振子かセラロックを接続します。
Pin16		OSC 20MHz	水晶発振子用端子	20MHzの水晶発振子かセラロックを接続します。
Pin17	O	Keying	キーイング出力。ドライブ用のTr(2sc1815等)を介して無線機をキーイングします。	5V キーイング
Pin18	O	TX(PTT)	PTT用出力。受信した文字列を送信中はHiを維持します。バッファ内の文字列送信後Loになります。	5V 送信 0V 受信

動作説明

適切にハードウェアが製作された前提で、動作を説明します。

電源投入

RSKeyer1チップに電源を投入すると、内部で初期化を行ったあとに約1秒間LEDを点灯します。初期化時には、PIN1－PIN4の入力状態を読み取り、RS232Cのボーレート、フロー制御の内部設定を行います。キーイング速度は初期値として20WPMにセットされます。初期化完了時にRS232Cに[Power ON]と出力されます。

PC側の設定

PC側のターミナルは、

ボーレート: 9600 or 19200 bps (RSKeyer1のPIN1の設定による)

キャラクタ: 8bit

パリティ: なし

ストップビット: 1

ソフトウェアフロー制御(Xon,Xoff): 有効 or 無効 (RSKeyer1のPIN3の設定による) とします。PCとRSKeyer1との接続は信号線2本とグラウンドの3本のみでハードウェアフロー制御は使用していません。また、信号線以外の線で接続相手を確認しているようなターミナルソフトは使用できませんので注意してください。

電源断

PC側のターミナルソフトを終了し、RSKeyer1への電源を切ります。

電源投入後の操作

RSKeyer1とPC(ターミナルソフト)接続完了後の操作は以下のようになります。

キーイング速度の設定

キーイング速度を設定するには、RSKeyer1に*(アスタリスク)で始まる速度設定コマンドを送信します。速度は5－60WPMの範囲で設定可能です。

送信文字列例

25WPMに設定 → “*02 0025”

30WPMに設定 → “*02 0030”

(コマンド送信の詳細に関しては、「RS232C通信」参照)

ウエイトの設定

初期化時に短点と長点は1:3の割合になっていますが、このウエイトを変える場合は、以下のコマンド文字列を送信します。

送信文字列例

ウエイトを55に設定 → “*03 0055” (デフォルトは50)

ウエイトを45に設定→ “*03 0045”

(コマンド送信の詳細に関しては、「RS232C通信」参照)

PCから文字列を送信してキーイング

キーイングを行う場合は、PCのターミナルソフトから半角大文字からなる文字列を送信します。(詳細は「RS232C通信」参照)

送信文字列例

CQの例→ “CQ CQ CQ DE JA1ABC JA1ABC TEST K”

(受信バッファの文字列をキーイング中にパドル操作をするとキーイングを中断し、受信バッファをクリアします。)

注意

フロー制御が無効になっている場合は、文字列をキーイング中に次に送信する文字列を送信しても無視されてしまいます。また、フロー制御を有効にしてもターミナルソフトによっては、次に送信する文字列の最初の1文字が無視されてしまう場合があります。

RS232C通信

RSKeyer1の受信バッファは80文字分確保しており、PCより送信される文字を1文字ずつ受信し、バッファに格納していきます。受信した文字が0x0a(LF)か0x0d(CR)のときにそれまで受信バッファに格納された文字を順次モルスコードに変換し、キーイングしていきます。受信バッファをすべてキーイングすると、受信バッファをクリアして次の文字受信に備えます。(デリミタ1文字を考えると一回に送信できる文字列は79文字となります。)

キーイング可能な文字は、PICのEEPROMの格納されていて下表の赤色の文字とスペースがキーイング可能な文字です。(キーイング不可能な文字を受信した場合はキーイングせず、スペースになります。)

EEPROMイメージ	0	1	2	3	4	5	6	7	8	9	10	11	12
0x00	0		1		2		3		4		5		6
0x10	8		9		A		B		C		D		E
0x20	G		H		I		J		K		L		M
0x30	O		P		Q		R		S		T		U
0x40	W		X		Y		Z		.		,		?
0x50	;		'		"		()		+		-
0x60	=		\$		<		>		@				
0x70													
0x80													
0x90													
0xA0													
0xB0													
0xC0													

注) アルファベットはすべて半角の大文字です。小文字はキーイング不可能ですのでPCから送信時に大文字に変換するなどして対応してください。

RSKeyer1が文字列を受信した際に、その文字列をキーイングする文字として認識するモードと、コマンドとして認識するモードがあります。

RSKeyer1は受信文字列の先頭文字が'*'(アスタリスク)のときは、その文字列はコマンドと認識します。先頭文字列が'*'でないときはキーイングする文字と認識し、キーイングを開始します。

キーイング文字処理

PCより、キーイング可能な文字列の最後にLF(またはCR)を付加した文字列を送信することによってRSKeyer1はキーイングを開始します。

文字チルダ“~”を受信すると次の文字をキーイングしたあとのスペースが1短点分(通常は3短点)となります。“~BK”や“~AR”を受信するとBとK、AとRの文字間の間隔が短くなり、連続してキーイングされます。

コマンド文字処理

RSKeyer1が受信した文字列の先頭文字が'*'のときにその文字列はコマンドと解釈します。コマンド文字列は以下の書式になっています。

*02 0025

先頭文字が‘*’、次の2文字(02)がコマンド番号、次にスペース、次の4文字(0025)がそのコマンドのパラメータになります。最後にLFまたはCRが付加されます。

コマンド種類

02(通信速度設定)

コマンドパラメータにて、キーイング速度(WPM)を設定します。設定可能範囲は5－60です。

03(ウエイト設定)

短点と長点の比率を調整します。設定可能範囲は10－90です。

04(PTTコントロールリードタイムの設定)

232Cよに受信した文字列送信中はポートA1がHiを維持しますが、文字列送信前何msからHiにするかの設定です。設定範囲は0－200です。

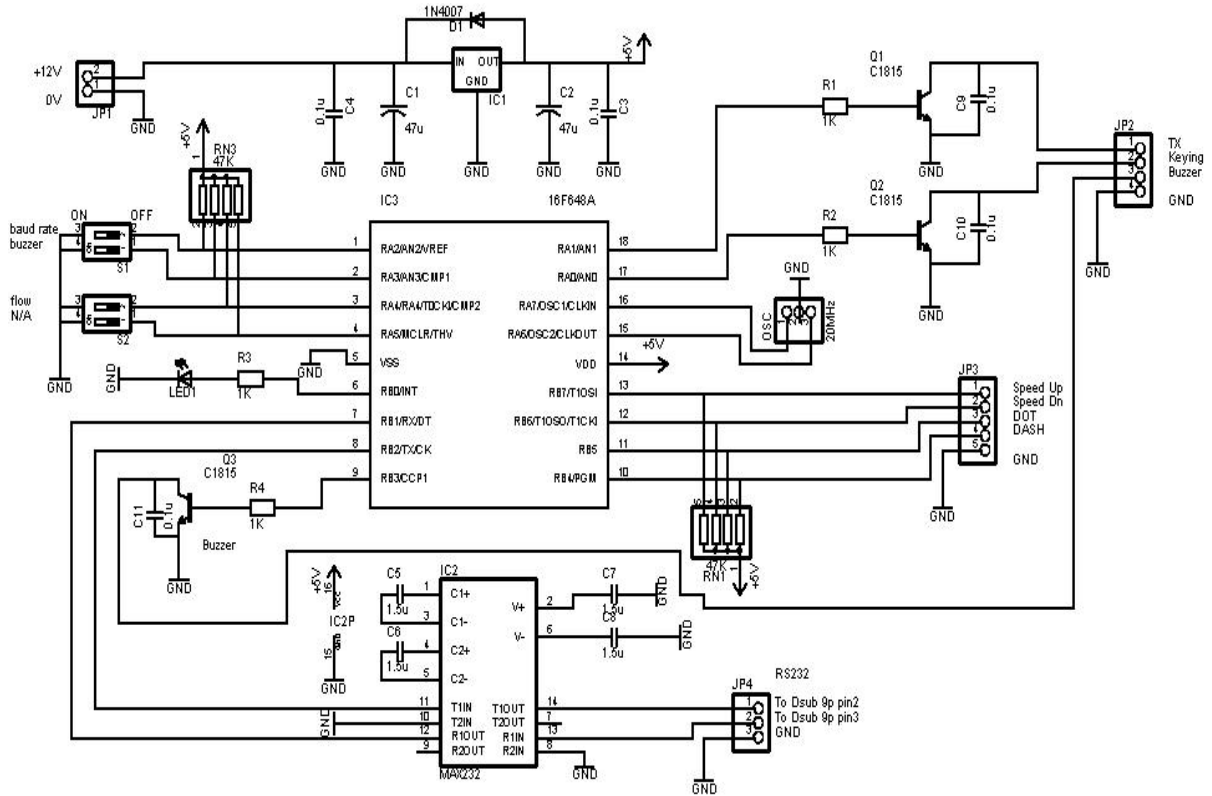
05(PTTコントロールテールタイムの設定)

232Cよに受信した文字列送信中はポートA1がHiを維持しますが、文字列送信後何msHiを維持するかの設定です。設定範囲は0－200です。

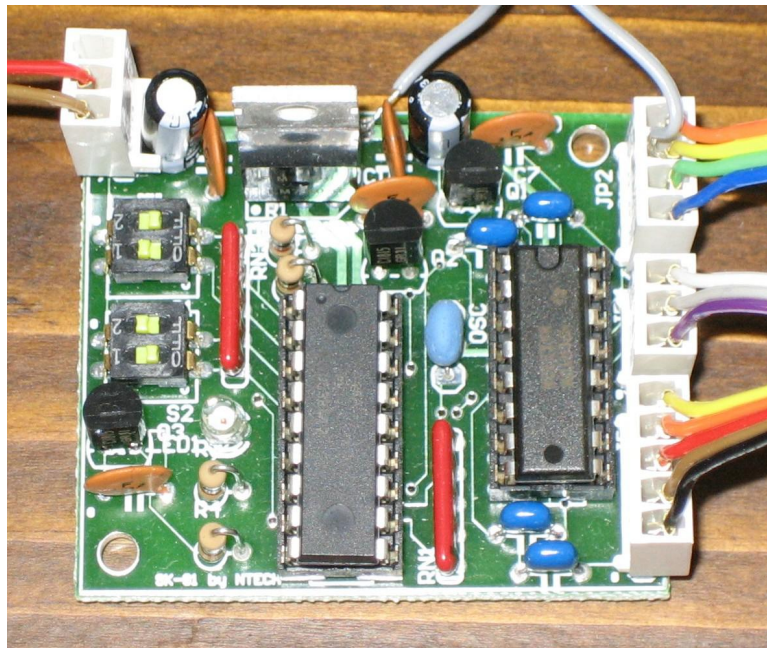
RSKeyer1を使った製作例

発振子にセラロックを使った製作例を以下に示します。

回路図

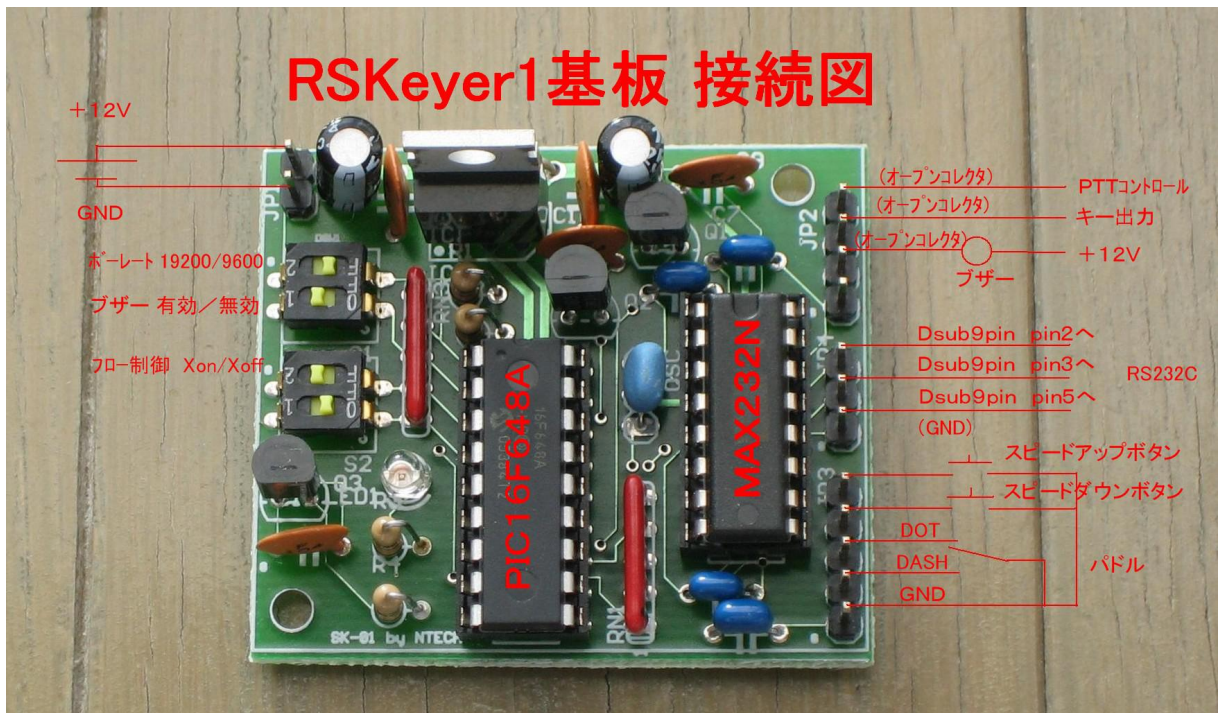
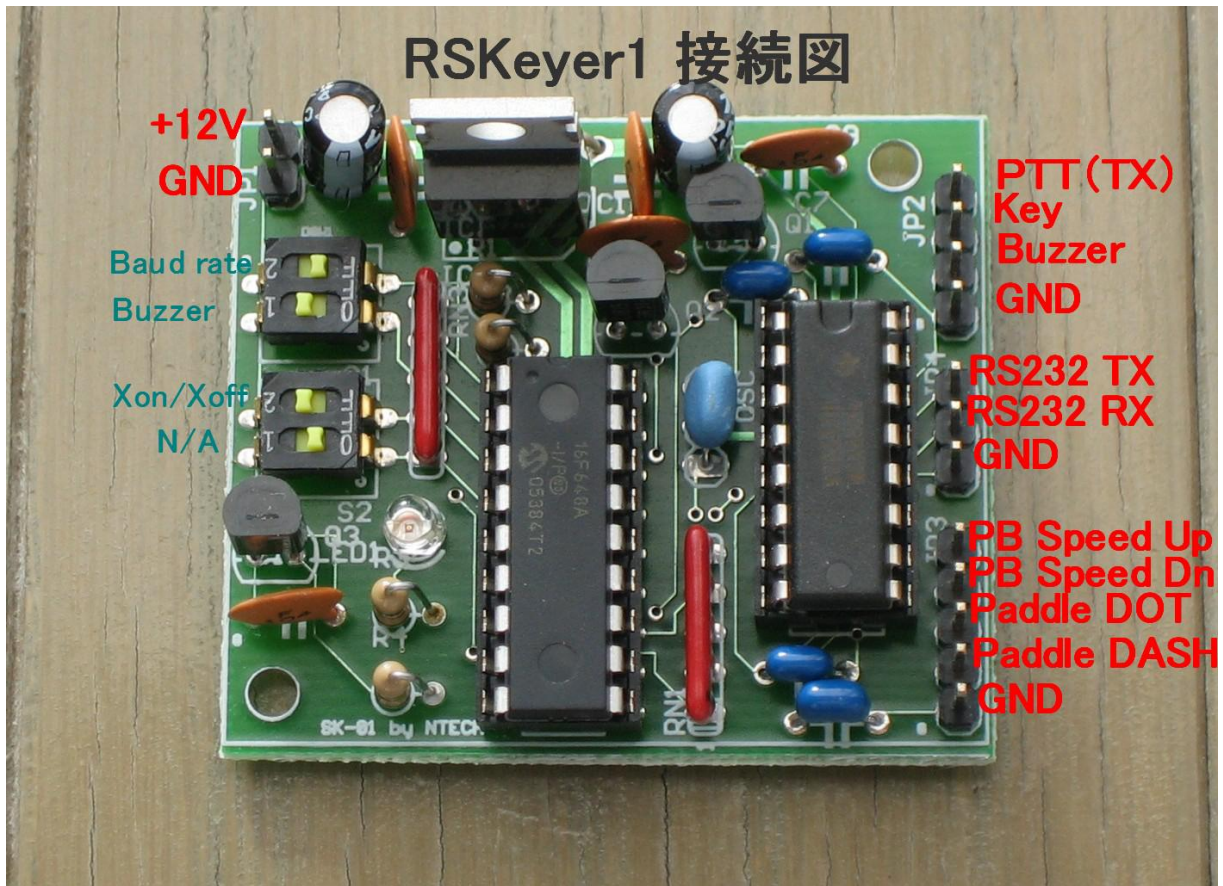


完成基板



50WのトランシーバでSWR1.1のアンテナでは特に回り込み等の問題は起きませんでした。動作環境によってはコモンモードノイズ対策が必要になる場合もあるかとおもいます。

外部との接続はピンヘッダにしています。



サンプルプログラムの説明

RSKeyer1の動作確認用のプログラムを作成しました。機能としては、テキストの送信、速度の設定、ウエイトの設定の3つです。

製作環境は

WinXp+Delphi2005 (NEC LavieノートPC)

動作環境は、

シリアルポートがあるWinXpマシンです。

動作確認は

NEC LavieノートPC+エレコムUC-SGT(USB->RS232C変換)

実行すると以下の画面が表示されます。(インストール等の作業は必要なく実行形式ファイルをダブルクリックすることによって実行します。)



使い方

最初にシリアルポートをオープンします。

RSKeyer1が接続されているCOMポートを確認して選択します。(動作確認時ではCOM4が該当ポートになりました。)通信速度、Xon/Xoffの設定はRSKeyer1のポートA2、ポートA4の入力にあわせて選択します。ボタン「ポートオープン」をクリックすると選択したポートがオープンされます。

「送信」ボタン:これをクリックするとボタンの左側に入力されている文字列がRSKeyer1に送信され、キーイングされます。

「速度WPMの設定」-「設定」ボタン:これをクリックするとRSKeyer1のキーイング速度

が選択した速度に設定されます。

「ウエイト50→1:3」-「設定」ボタン:これをクリックするとRSKeyer1のウエイトが選択した値に設定されます。

「受信文字列」下のメモコントロール

RSKeyer1からPCに送られた文字列を表示します。

PC側のソフトで比較的簡単にメモリキーヤー等を実現できるかとおもいます。