

Winkey2 – Morsekeyer mit Steuerung unter Windows

PETER ROSSBACH – DK4MX

Multitasking unter Windows ist ein Problem für Hardwaresteuerungen. Diverse Contest- und Logbuchprogramme reagieren dadurch bei der Telegrafieausgabe mit Stottern und abgehackten Zeichen. Doch seit geraumer Zeit löst dieses Problem ein Winkey-Chip von K1EL. Auch Winkey2 als neue Version generiert die Morsezeichen selbst. Wir stellen eine sowohl über USB ansprechbare als autark verwendbare Baugruppe mit galvanischer Trennung vor.

Schon vor Jahren fiel mir auf, dass meine Windows-Contestsoftware bei der Ausgabe von Telegrafiezeichen stottert. Auch durch Ändern diverser Parameter ist dieses Verhalten nicht in den Griff zu bekommen. Der Tipp für eine Lösung kam von einem befreundeten Funkamateurl: Versuche es doch mal mit dem Winkey-Chip. Die Internetrecherche ergab eine Bezugsmöglichkeit bei Steven T. Elliott, K1EL [1].

oder Logbuch-Software über eine virtuelle COM-Schnittstelle mit PTT-Ausgang – herkömmliche softwaregesteuerte Telegrafieausgabe über die Leitungen DTR (Key) und RTS (PTT) einer virtuellen COM-Schnittstelle, wobei der Winkey2-Chip nicht notwendig. Letzterer Betriebsfall ist als Alternative für Contest- und Logsoftware gedacht, die



Nach kurzer Zeit lag die Sendung im Briefkasten. Also schnell eine Platine nach dem Original von K1EL gefertigt, aufgebaut und getestet. Und siehe da: einwandfreie Zeichengabe! Und ein externes Paddle ließ sich auch noch anschließen. Doch die Entwicklung ging weiter. K1EL programmierte im vergangenen Jahr einen neuen Keyer in einem PIC 16F688, und nannte ihn Winkey2. Er ist abwärtskompatibel zu Winkey, wird aber jetzt im 14-poligen DIL-Gehäuse geliefert.

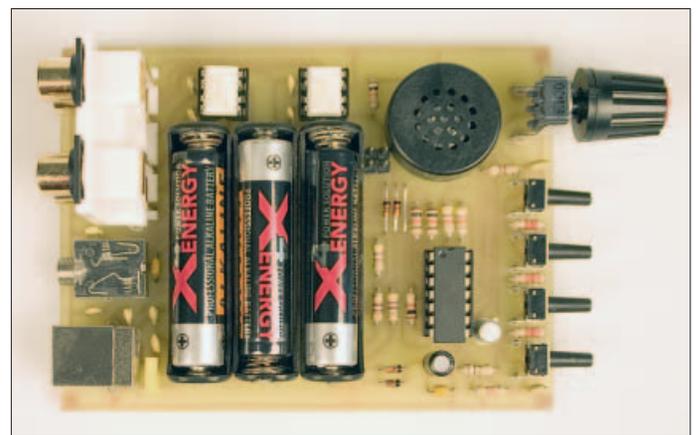
In der vorliegenden Schaltung wurden neben der galvanischen Trennung der Tastend PTT-Ausgänge drei Betriebsarten realisiert:

- autarker Betrieb als Elbug mit PTT-Ausgang
- Betrieb mit Rechnerkopplung (Host-Mode) zur softwaregesteuerte Telegrafieausgabe unter dem Betriebssystem Windows mit entsprechender Contest-

Bild 1: Das Gehäuse von Fischer Elektronik bietet selbst bei rauem Einsatz sehr guten Schutz.

Fotos und Screenshots: DK4MX

Bild 2: Ansicht der bestückten Musterplatine mit dem Winkey2-Chip rechts



nicht über eine Winkey-Unterstützung verfügt und/oder für neuere PCs ohne COM-Schnittstelle.

■ Schaltungsbeschreibung und Aufbau

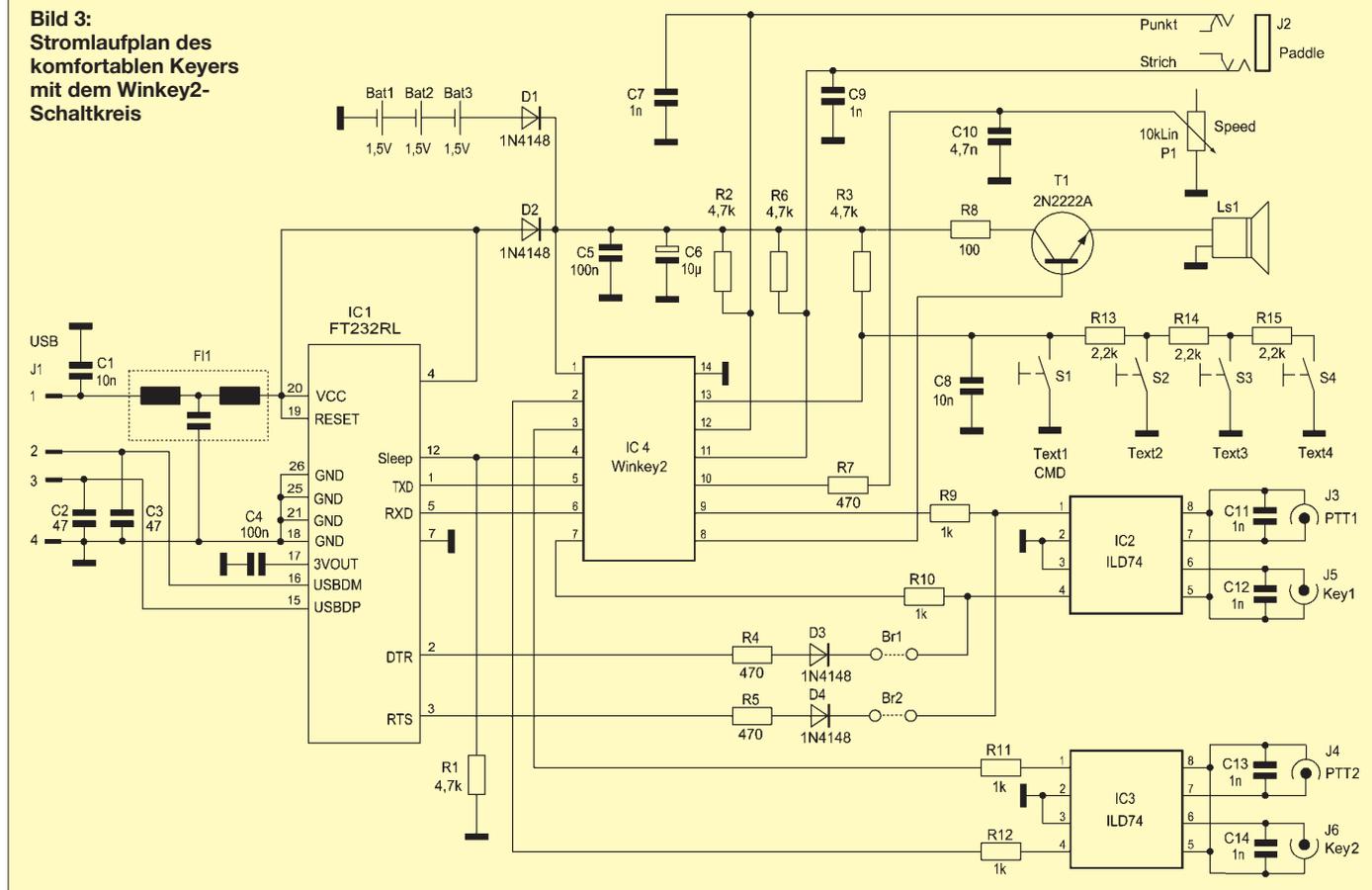
Hauptakteur in der Schaltung nach Bild 3 ist der Winkey-PIC-Controller, in unserem Fall IC4. IC1 ist der mittlerweile bekannte USB-Seriell-Umsetzer FT232RL der in einem 28-poligen SSOP-Gehäuse geliefert wird und auf der Unterseite der Platine untergebracht wurde. Der Pinabstand von 0,65 mm ist aber mit ruhiger Hand zu beherrschen. Das Pin 1 ist auf dem Chip als Zahl ge-

kennzeichnet. Die Stromversorgung der Baugruppe erfolgt gleitend entweder aus der eventuell angeschlossenen USB-Schnittstelle oder aus drei intern einzusetzenden Micro-Batterien (Baugröße AAA). Dabei versorgt die Batterie im PC-unabhängigen Betrieb die Schaltung über VD1 mit 4,5V – die Stromaufnahme im Standby-Betrieb beträgt zirka 10 μ A. Bei Tastung fließen etwa 5 bis 20 mA, je nach Schallwandler. Er stellt zwar den größten Verbraucher dar, ist aber abschaltbar. Trotz Schallwandler im Innern des Gehäuses ist dieser gut zu hören, wenn die hinteren Gehäusebohrungen nicht zu eng bemessen werden. VT1 ist der Treiber für den Schallwandler. Es kann jeder beliebige Si-NPN-Typ eingesetzt werden. Mit R8 im Bereich von 47 bis 100 Ω ist die Lautstärke des Mithörtones in Grenzen festlegbar. Sobald die Baugruppe an den USB-Port angeschlossen wird, übernimmt dieser über Fi1 und VD2, aufgrund der dann größeren Betriebsspannung von 5 V die Versorgung der Baugruppe. Die Batterie wird nur noch minimal belastet. Diese Systematik erspart einen Umschalter und die Batterie kann fest im Gerät verbleiben. Sie hält leicht mehrere Jahre. Der Pin 4 des IC4 wird über den Ausgang Sleep von IC1 (Pin 12) aufgeweckt.

Alle verwendeten Kondensatoren dienen der Vermeidung von HF-Einstreuungen.

Fi1 filtert über die +5-V-Leitung des USB-Ports eventuell eintreffende HF-Reste aus. Die Ausgänge TXD und RXD des FT232RL sind mit den Eingangsports des Winkey2-Chips verbunden und realisieren die serielle Datenübertragung mit 1200 Bd. DTR und RTS von IC1 führen über eine Widerstands-Dioden-Reihenschaltung und zwei Steckbrücken direkt auf die Eingänge des Doppeloptokopplers IC2. Damit ist die schon genannte dritte Betriebsart realisierbar. Das Paddle ist über einen 3,5-mm-Stereo-Klinkenstecker an J2 anschließbar. Bei Tasten mit 6,3-mm-Klinkenstecker ist ein handelsüblicher Adapter verwendbar.

Bild 3:
Stromlaufplan des
komfortablen Keyers
mit dem Winkey2-
Schaltkreis



Über die vier Taster werden die Standardtexte 1 bis 4 aufgerufen – die linke Taste ist außerdem für die Kommando steuerung vorgesehen. Interessant ist hierbei die Programmierung über nur einen Port, dem Pin 13 von IC4. R3, R13 bis R15 fungieren als Spannungsteiler. Die entsprechend der gedrückten Taste anliegende Spannung an Pin 13 wird vom internen A/D-Umsetzer erkannt.

Über das Potenziometer P1 ist die Gebe geschwindigkeit einstellbar. Alternativ kann dies auch über die Schnittstelle von der

Steuersoftware erfolgen, doch der Griff zum Knopf ist in vielen Fällen schneller. Um den Drehknopf bündig auf die Frontplatte setzen zu können, muss die Achse des Potenziometers etwas gekürzt werden.

Zur Potenzialtrennung werden zwei Doppeloptokoppler vom Typ ILD74 verwendet. Ähnliche Typen, wie z. B. CNY74-2, sind ebenfalls verwendbar. Die Tast- und PTT-Ausgänge führen direkt auf robuste Cinchbuchsen. Die einseitige Platine hat die Maße 100 mm × 75 mm und passt in das Gehäuse FR 80 42 100 ME von Fischer-Elektronik. Wer einen Bausatz (Bestell-Nr. BX-008) über den FA-Leserservice bezieht, kann auf ein Gehäuse mit vorgestanzter Front- und Rückplatte zurückgreifen. Der Winkey2-Chip ist auch einzeln über den FA-Leserservice erhältlich: Bestell-Nr. WINKEY2-V21.

Für alle ICs wurden, bis auf IC1, Steckfassungen eingesetzt. Br1 ist eine Drahtbrücke, Br2 und Br3 eine Steckerleiste. Bei der Bestückung ist die übliche Reihenfolge einzuhalten, also zuerst die Widerstände, dann die Kondensatoren und zum Schluss die ICs und Buchsen. Nach nochmaliger optischer Prüfung der Platine kann die Batterie eingesetzt werden.

■ Inbetriebnahme und Test

Sinnvoll ist zuerst ein Test im autarken Betrieb mit angeschlossenem Paddle. Nach

Einlegen der Batterie ertönt ein R und der Keyer ist betriebsbereit. Bei Betätigen des Paddles müssen entsprechende Zeichen ausgegeben werden. Über das Potenziometer P1 kann die Gebe geschwindigkeit eingestellt werden. Damit ist der Hardwaraufbau abgeschlossen und die Baugruppe im autarken Betrieb verwendbar.

Wer möchte, kann einen Transceiver anschließen und schon mal testen. Bei Betätigung eines der vier Taster ergeben sich noch keine sinnvolle Ausgabe, da noch keine Texte programmiert wurden.

Die Batterie kann im Gerät verbleiben. Etwa 2 s nach Betätigung des letzten Paddles schaltet der PIC in den Schlafmodus. Er wacht aber sofort bei einer Paddlebetätigung wieder auf.

■ Autarker Betrieb

Die Einstellungen für den PC-unabhängigen Betrieb können sowohl über eine angeschlossene Morsetaste und die Taster auf der Frontplatte als auch komfortabel über das im nächsten Abschnitt näher erläuterte Programm WK2MGR von [1] eingestellt werden. Zur Eingabe über ein Paddle sind die in Tabelle 2 aufgelisteten Kommandos nutzbar. Die Aktivierung des Eingabemodus erfolgt über die Taste S1 (CMD). Diese ist länger als 2 s zu drücken. Es erfolgt ein eine Bestätigung durch ein R.

Tabelle 1: Technische Daten

serielles RXD/TXD-Interface mit 1200 Bd
Lambic CW-Paddle-Interface
zwei Tast- und PTT-Ausgänge, umschaltbar
justierbare PTT-Ansprech- und Abfallzeit
einstellbares Punkt-Strich-Verhältnis
Farnsworth-Eigenschaft wählbar
automatischer Zwischenraum möglich
Mithörton abschaltbar
Paddle-Schaltpunkt und Punkt-Strichkorrektur beim ersten Zeichen einstellbar
LambicA, B und „Bug“ Mode
Geschwindigkeitseinstellung über Potenziometer oder Software möglich
Speicherung von Standardtexten mit eingebetteten Kommandos möglich
Schlafmodus im autarken zur Batterieschonung
für Hochgeschwindigkeits-Telegrafie (HSCW) und extrem langsame Telegrafie (QRSS) verwendbar
Spannungsversorgung über USB-Schnittstelle oder intern über drei 1,5-V-Micro-Zellen

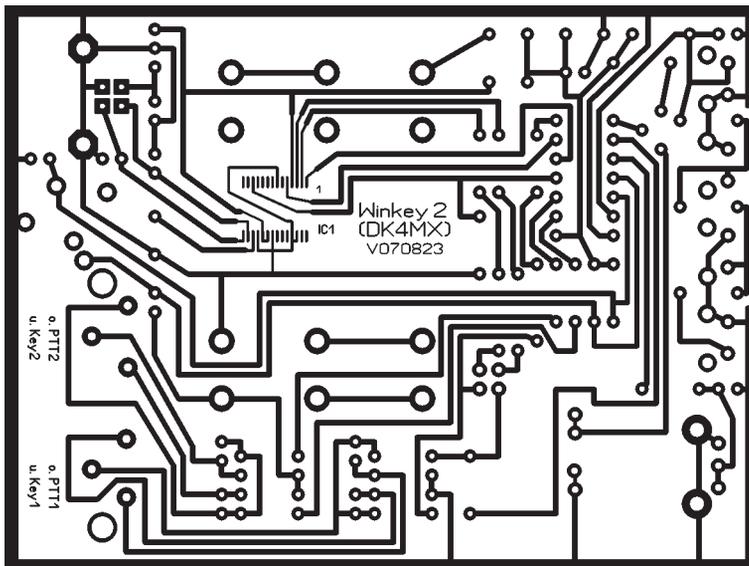


Bild 4: Layout der Platine

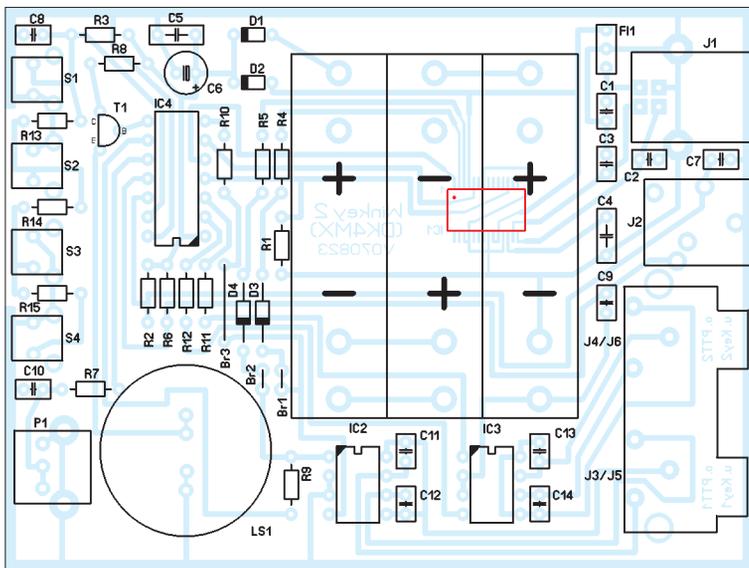


Bild 5: Bestückungsplan der Baugruppe

Danach ist das gewünschte Kommando durch Paddlebetätigung einzugeben. Folgt ohne Eingabe ein ? ist man aus der Eingabeschleife raus. Ebenfalls mit ? anwortet Winkey2 bei Falscheingabe. Nach Abschluss aller EingabeprozEDUREN können die Parameter mit P im EEPROM gespeichert werden.

■ USB-Treiber installieren

Wem das zu aufwendig ist, der kann die Software WK2MGR verwenden. Mit ihr lassen sich unter anderem die Standardtexte komfortabel eingeben. Zudem besteht die Möglichkeit, in die Standardtexte Kommandos einzufügen. Diese werden immer mit einem Schrägstrich eingeleitet. Doch vor der Nutzung des Programms ist es notwendig, einen Treiber für eine virtuelle COM-Schnittstelle zu installieren. Nur auf diesem Weg ist es möglich, auf den FT232RL von FTDI zuzugreifen, der die Schnittstellenanpassung im Gerät übernimmt. Auf der Internetseite von FTDI [2] sind die Treiber für diverse Betriebssysteme

downloadbar. Je nach vorhandenem Betriebssystem ist der notwendige Treiber auszuwählen. Nach Anklicken startet der Download. Die auf dem PC abgelegte ZIP-Datei ist zu öffnen und die enthaltenen Dateien in einen temporären Ordner anzulegen.

Nun kann Winkey2 über ein USB-Kabel an den PC angeschlossen werden. Der Gerätemanager erkennt daraufhin ein neues USB-Gerät und verlangt nach einem Treiber. Es sind zuerst *Software von einer Liste oder bestimmten Quelle installieren* auszuwählen und dann der vorher gewählte temporäre Ordner anzugeben. Die weitere Installation läuft über den Hardware-Assistenten automatisch ab.

Nach Abschluss ist das neue USB-Gerät verwendbar und der virtuelle COM-Port ist im Gerätemanager unter *Anschlüsse* zu finden. Sollte eine höhere Schnittstelle als COM 4 eingetragen sein, ist es sinnvoll, auf eine unbenutzte zwischen COM 1 bis 4 zu wechseln. Um die COM-Anschlussnummer zu ändern, klickt man im *Geräte-*

Manager unter *Anschlüsse* auf die neue Schnittstelle, sodass das Fenster *Anschluss-Eigenschaften* erscheint. Über diesen Punkt gelangt man zu den *Erweiterten Einstellungen* (Advanced) für die neue Schnittstelle, in der der Schnittstelle eine neue Anschlussnummer zugewiesen werden kann.

Somit sind alle Voraussetzungen vorhanden, um Parametereingaben unter WK2MGR und Programmierungen unter MProg durchzuführen.

■ Installation von WK2MGR

Alle Informationen nebst zugehöriger Software sind auf [1] zu finden. Auszuwählen ist zunächst *WK2MGR*. Darauf folgt die übliche Lade- und Installationsprozedur. Ein weiteres interessantes Produkt ist *WKTest*. Dies ist ebenfalls ein Test und Eingabeprogramm, allerdings mit interessanten virtuellen Kontroll- und

Tabelle 2: Hauptkommandos im autarken Betrieb

Zeichen ¹⁾	Funktion
A	Mithörton ein-/ausschalten
H	PTT-Umschaltzeit auf Empfang einstellen
K	Tastmodus: A → Lambic A, B → Lambic B, U → Ultimatic, S → Straight Key, E → Dit Priority, T → Dah Priority
L	Standardtext speichern
M	Sendertastung ein-/ausschalten
N[nnnn]	Anfangswert der vierstellige Seriennummer eingeben
O	Umschaltung der aktiven Tast- und der PTT-Leitungen
P	Parameter im EEPROM speichern
R	Abspielen eines Standardtextes ohne Sendertastung
T	Dauerton zum Abstimmen
U	automatischen Zwischenraum einfügen
X	Punkt- und Strichpaddles wechseln
Z	Höhe des Mithörtöns festlegen

¹⁾ Eingabe über Paddle

Tabelle 3: Beispielkommandos in Standardtexten

Zeichen	Funktion
/Bnn	Bakenperiode/s (nn = 00...99).
/Cn	Standartext n aufrufen und danach zurückkehren (n = 1...4)
/Hn	HSCW-Geschwindigkeit/WPM: 0 → 200, 1 → 300, 2 → 400, 3 → 600, 4 → 800, 5 → 1200
/Knn	Sender für nn Sekunden tasten
/N	Seriennummer ausgeben und anschließend erhöhen
/P	auf Paddleingabe oder Tasteneingabe für Standarttext warten
/Qn	Punkttdauer für QRSS-Betrieb: 0 → 3 s, 1 → 6 s, 2 → 10 s, 3 → 12 s, 4 → 30 s, 5 → 60 s
/Snn	Geschwindigkeit/WPM auf nn setzen (nn = 5...59)
/Wnn	nn Sekunden warten (nn = 00...99)
/X	Einstellungen der Kommandos /Snn, /Hn oder /Qn löschen
/n	zum Standardtext n springen (n = 1...4)

Eingabeeffekten. Interessant ist die Verfolgung der Speedeinstellung über einen Schieberegler und die Rücksendung des eingegebenen Textes über das Schnittstellenecho. Auch *Winkeyer2* animiert zum komfortablen Testen der Hardware. Integriert ist hier noch ein Logprogramm, das man ausprobieren sollte.

Es sei an dieser Stelle betont, dass sowohl über *WK2MGR* als auch über die anderen o. g. Programme nur die Parameter für den autarken Betriebs eingestellt werden. Im Host-Mode wird die Steuerung des Geräts durch das jeweilige Steuerprogramm vorgenommen, nach dessen Beendigung wieder die Parameter für den autarken Betriebs aus dem EEPROM geladen werden.

Beschränken wir uns aber auf *WK2MGR*. In ihm ist zuerst die vorher gewählte COM-Schnittstelle unter *Comm* einzustellen. Nach Drücken des Buttons *Read WK* wird die Schnittstelle geöffnet und die Versionsnummer eingelesen. Sämtliche Parameter lassen sich in den sieben Fenstern auf der rechten Seite ändern. Als erster Test kann die Höhe des Mithörtons

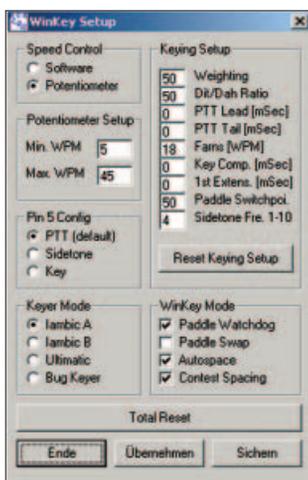


Bild 6: Screenshot des Winkey-Setup-Menüs in RCKLog

tuitiv nachvollziehen. Beim Verlassen von *WK2MGR* kehrt *Winkey2* wieder in den autarken Betrieb zurück – die Einstellungen werden vorher im EEPROM abgelegt.

■ **Host-Modus**

Ich beschränke mich beim Betrieb mit einem PC ausschließlich auf die *Plug and Play* Informationen. Wer eigene Applikationen schreiben möchten, dem stellt

Untermenüs in *RCKLog* [5]. Erreichbar ist es über *Special* und *Winkey Setup on/off*. Zu sehen sind wieder die bekannten Buttons für diverse Einstellungen. Die Geschwindigkeit ist über einen Schieber wählbar. Zugriff auf den Einstellbereich bietet das Fenster darunter. Ich persönlich bevorzuge diese manuelle Einstellmöglichkeit des Gebetempos, ohne in der Contest-Hektik erst den entsprechenden Button in der Software suchen zu müssen. Wird *Pin 5 Config* auf *PTT* gesetzt, ist der Mithörton abgeschaltet. Mit *Sichern* werden die Einstellungen in *RCKLog* abgelegt und mit *Übernehmen* zu *Winkey2* übertragen. Jede Änderung erfordert eine neue Übertragung. Die geänderten Parameter sind jedoch sofort, auch ohne einen Neustart von *RCKLog* nutzbar. Die Ausgabe eines vorher eingegebenen Textes ist durch eine Paddlebetätigung unterbrechbar. Dies erlaubt flexibles Reagieren außerhalb der Standarttexte.

Nach dem Schließen des Programms kehrt *Winkey* wieder in den autarken Betrieb zurück. Dies funktioniert bei der aktuellen Version von *RCKLog* zuverlässig. Voraussetzung ist, dass der Programmierer das entsprechende Schließkommando für die Verbindung verwendet hat. Ebenfalls bei Abschalten des PCs oder Trennen der USB-Verbindung kehrt *Winkey2* zum autarken Betriebs zurück. Bei Problemen mit

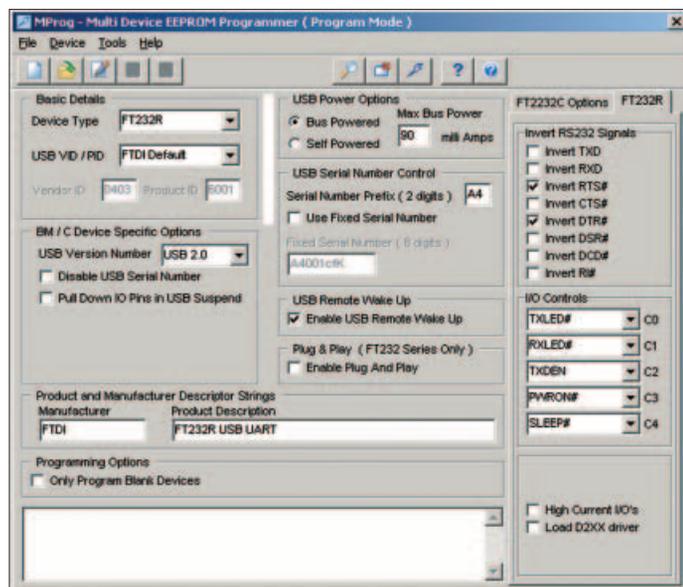


Bild 7: Screenshot MProg

(Sidetone) dienen – einfach mal ändern und über *Write WK* ans Gerät schicken. Die Änderung ist danach sofort akustisch wahrnehmbar.

Dies ist die generelle Prozedur für alle Einstellungen. Auf der linken Seite sind die Fenster für die Standardtexte zu sehen. Gewünschten Text, eventuell mit integrierten Kommandos, einfach eingeben. Nach dem Laden ist der jeweilige Text über die Taster abrufbar. Eine vorherige Überprüfung ist mit den Buttons *Test* möglich. Hardwaremäßig sind nur vier Texte vorgesehen, was für die üblichen Anwendungen ausreichen dürfte – der *Winkey*-Chip selbst erlaubt bis zu sechs. Alle anderen Einstellungen lassen sich in-

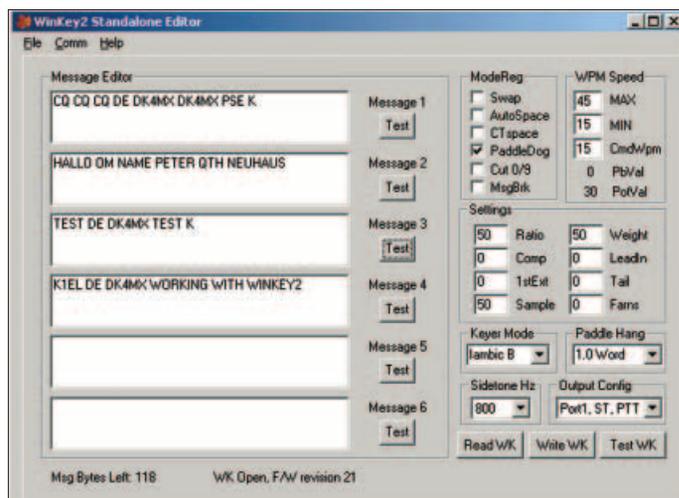


Bild 8: Screenshot WinKey2MGR

K1EL eine umfangreiche Kommandosammlung und Programmieranweisungen zur Verfügung. Die gebräuchlichsten Contest- und Logprogramme integrieren mittlerweile einen *Winkey2*-Modus, wie z. B. *NIMMLogger*, *RCKLog*, *Win-Test*, *TurboLog*, *Winlog32*, *DXBase*, *Logger32*, *DX4WIN*, *HamRadioDeluxe*, *Luxlog*, *TRXManager* und *WriteLog*.

Beim Aufrufen wird nach vorheriger COM-Port-Einstellung die Telegrafieausgabe über *Winkey2* sofort unterstützt. Die Standardtexttasten sind außer Betrieb. Bild 6 zeigt das Beispiel des *Winkey-*

dieser Rückkehr hilft immer ein Reset, also das Drücken der Taste *CMD* für zirka 6 s – als Bestätigung werden sechs Punkte ausgegeben.

■ **Ausgabe über DTR und RTS**

Die Möglichkeit zur softwaregesteuerten Ausgabe von Telegrafiezeichen über die Leitungen DTR und RTS einer COM-Schnittstelle wurde für Programme vorgesehen, die *Winkey* noch nicht unterstützen und/oder für Nutzer, die nicht über einen PC mit COM-Schnittstelle verfügen. Der *Winkey2*-Chip ist dabei nicht notwendig,

kann aber eingesetzt werden, wobei der er dann parallel zur DTR/RTS-Ausgabe arbeitet. Die Brücken Br1 und Br2 müssen gesteckt sein. Selbstverständlich ist die Sendung dann nicht unterbrechbar. Ebenfalls kann das Ausgabtempo nur von der Software bestimmt werden. Eine unbedingte Voraussetzung für diese Betriebsart ist die Umprogrammierung des FT232RL auf ein invertiertes DTR- und RTS-Signal. Dies ist über das Programm *MProg* von *FTDI* [4] möglich. Nach Verbinden mit dem USB-Anschluss und Aufrufen von *MProg* wird das Hauptmenü angezeigt. Über *File-New* ist ein neuer Parameterblock erstellbar. Als erstes wählt man den Device-Typ: FT232R. Danach wird das Menü in Bild 7 sichtbar, in dem nur drei Einstellungen durchzuführen sind: Auf der rechten Seite *invert DTR* und *invert RTS* aktivieren und links unten den Haken bei *only program blank device* entfernen.

Danach über das Diskettensymbol speichern, denn ohne Speichern ist kein Programmieren möglich! Zum Abschluss über das Blitz-Symbol die Parameteränderung in den Chip brennen – fertig. Es wird eine neue Seriennummer generiert. Diese bewirkt beim erneuten Verbinden von Winkey2 mit dem PC die Erkennung eines neuen USB-Gerätes, respektive die Vergabe einer wiederum höheren virtuellen COM-Schnittstellenummer, die wie o.g. zu ändern ist. Eine erneute Treiberinstallation ist nicht notwendig. Hardwaremäßig ist für diesen Betriebsfall nur die Tastung von Transceiver 1 vorgesehen.

■ Zusammenfassung

Telegrafie macht Spaß und ist als Fremdsprache jedem Funkamateurliebhaber durchaus zu empfehlen. Wer dazu noch Interesse am Selbstbau einer komfortablen Elbug hat,

findet in Winkey2 ideale Unterstützung. Sowohl im autarken Betrieb als auch softwaregesteuert über diverse Contest und Logprogramme gibt das Modul sauberste Zeichen mit vielen frei wählbaren Parametern ohne Multitaskingprobleme aus. Der Aufbau der Platine dürfte mit etwas SMD-Erfahrung problemlos gelingen. Ein Bausatz für diesen Keyer (BX-008) ist über den FA-Leserservice erhältlich. Weitergehende Informationen sind auf [1] nachzulesen. Bei Problemen im Aufbau gebe ich im vertretbaren Rahmen unter Unterstützung. **dk4mx@gmx.de**

Literatur und Bezugsmöglichkeiten

- [1] Steven T. Elliott, K1EL: Winkey2 Information. <http://k1el.tripod.com/wk2info.html>
- [2] FTDI: Virtual COM Port Drivers: www.ftdichip.com → Drivers → VCP
- [3] Dallmeier W., DL4RCK: RCKLog. www.rcklog.de
- [4] FTDI: FTDI Utilities: www.ftdichip.com → Resources → Utilities