

# DL6ER - Keyer

---

## Benutzeranleitung

# Inhaltsverzeichnis

<b>1. Lizenz</b>	<b>5</b>
<b>2. Bedienung</b>	<b>6</b>
2.1. Einführung . . . . .	6
2.2. Erste Inbetriebnahme . . . . .	6
2.3. Betrieb des Keyers . . . . .	6
2.4. Abgesetztes Bedienteil . . . . .	7
2.5. Einstellungen zurücksetzen . . . . .	7
2.5.1. Zurücksetzen 1. Stufe . . . . .	8
2.5.2. Zurücksetzen 2. Stufe . . . . .	8
2.6. Eigenes Rufzeichen einspeichern . . . . .	8
2.7. Zeichenspeicher . . . . .	9
2.7.1. Wiedergabe . . . . .	9
2.7.2. Aufnahme . . . . .	9
<b>3. Menü</b>	<b>11</b>
3.1. 1 bis 9 - Sende Speicher 1 bis 9 . . . . .	11
3.2. A - Seitenton An/Aus . . . . .	11
3.3. B - Schleifensendung Speicher 1 (Bake) . . . . .	12
3.4. C - Eigenes Rufzeichen einspeichern . . . . .	12
3.5. D - Schleifensendung Intervall . . . . .	12
3.6. E - Sende Speicher 1 / Geschwindigkeit + 1 WpM . . . . .	13
3.7. F - Farnsworth Modus . . . . .	13
3.8. G - PTT Nachlaufzeit (geschwindigkeitsunabhängig) . . . . .	14
3.9. H - PTT Nachlaufzeit (geschwindigkeitsabhängig) . . . . .	14
3.10. I - Invertiere KEY-Ausgang . . . . .	15
3.11. J - Zeichenerkennungstoleranz . . . . .	15
3.12. K - Key/Paddle Betriebsmodus . . . . .	15
3.13. L - PTT Vorlaufzeit . . . . .	17
3.14. M - Stummschaltung für TRX-Leitungen . . . . .	17
3.15. N - Automatisches Leerzeichen zwischen Wörtern . . . . .	18
3.16. O - TRX-Leitungen 1 & 2 umschalten . . . . .	18
3.17. P - Programmiermodus betreten . . . . .	18
3.18. Q - nicht belegt . . . . .	19
3.19. R - Speicher aufzeichnen . . . . .	19
3.20. S - Einstellungen dauerhaft abspeichern . . . . .	19
3.21. T - Sende Speicher 2 / Geschwindigkeit - 1 WpM . . . . .	19
3.22. U - Dauersendung (Abstimmen) . . . . .	19
3.23. V - nicht belegt . . . . .	20

---

3.24. W - Zeichengeschwindigkeit in WpM . . . . .	20
3.25. X - Paddle-Seiten vertauschen . . . . .	20
3.26. Y - Tastaturfunktion einstellen . . . . .	20
3.27. Z - Seitenton Frequenz einstellen . . . . .	21
<b>4. Programmierung / Einspielen von Aktualisierungen</b>	<b>22</b>
4.1. Besonderheit: Aktualisierung . . . . .	23
4.2. Arduino-Module . . . . .	23
4.3. Sparkfun Pro Micro . . . . .	23
4.3.1. Linux/Mac: Arduino-Software . . . . .	23
4.3.2. Windows®: Treiber & Arduino-Software . . . . .	23
<b>5. Technische Beschreibung</b>	<b>24</b>
5.1. Konfiguration der Anschlüsse . . . . .	24
5.2. CW Timing . . . . .	24
5.2.1. Dit- / Dahlänge . . . . .	24
5.2.2. Abstand zwischen Buchstaben . . . . .	25
5.2.3. Abstand zwischen Wörtern . . . . .	25
5.2.4. Farnsworth Timing: Verlängerte Pausen . . . . .	25
5.3. PTT Timing . . . . .	26
5.3.1. PTT Vorlaufzeit $t_{\text{lead}}$ . . . . .	26
5.3.2. PTT Nachlaufzeit $t_{\text{hang}} + t_{\text{tail}}$ . . . . .	26
5.4. Tastaturfunktion des Keyers . . . . .	27
5.4.1. Steuerzeichen . . . . .	28
5.5. Zeichenerkennungsalgorithmus . . . . .	29
5.5.1. Automatisches Leerzeichen zwischen Wörtern . . . . .	30
<b>6. Winkey Emulation</b>	<b>31</b>
6.1. Echo . . . . .	31
6.2. Aktiviere Host-Modus . . . . .	31
6.3. Deaktiviere Host-Modus . . . . .	32
6.4. Setze Zeichengeschwindigkeit . . . . .	32
6.5. PTT Lead-In/Tail . . . . .	32
6.6. Zeichengeschwindigkeit abfragen . . . . .	32
6.7. Ausgang wählen . . . . .	33
6.8. Breche Sendung ab . . . . .	33
6.9. Tune TRX . . . . .	33
6.10. Winkey Modebyte . . . . .	34
6.11. Winkey Modebyte (2) . . . . .	34
6.12. Status abfragen . . . . .	34
6.13. PTT . . . . .	35
6.14. Betriebszeichen erzeugen . . . . .	35
6.15. Relais . . . . .	36
6.16. ASCII Zeichen . . . . .	36
6.17. Getestete Software . . . . .	37

## Appendix

<b>A. Bausatzbeschreibung</b>	<b>A1</b>
A.1. Benötigte Bauteile für den DL6ER-Keyer . . . . .	A2
A.1.1. Small (ohne zehn steuerbare Relais) . . . . .	A2
A.1.2. Big (inkl. zehn steuerbaren Relais) . . . . .	A3
A.1.3. Abgesetztes Bedienteil . . . . .	A4
A.1.4. Relais Kontrollboard . . . . .	A5
<b>B. Betriebsparameter und Standardeinstellungen</b>	<b>A6</b>

# 1. Lizenz

Die für dieses Projekt erstellte Software (d.h. der Quelltext) liegt in einer für den Menschen lesbaren und verständlichen Form vor. Es handelt sich hierbei um die für den Menschen im Verständnis verhältnismäßig einfach zu erlernende, strukturierte Programmiersprache C++.

Der Quelltext darf beliebig kopiert, verbreitet und genutzt werden. Es gibt insbesondere keine Einschränkungen bezüglich der Anzahl der Nutzer selbst, noch bezüglich der Anzahl der Installationen auf Mikrocontrollern insgesamt. Mit der kostenfreien Vervielfältigung und Verbreitung der Software ist der Autor ausdrücklich einverstanden, solange ein ausdrücklicher Hinweis auf die ursprüngliche Quelle (z.B. [www.dl6er.de](http://www.dl6er.de)) gegeben wird.

Die Software unterliegt dem deutschen Urheberrecht (UrhG). §69a schützt sowohl die Software, als auch das Entwurfsmaterial, das auch diese Dokumentation umfasst.

Durch den offengelegten Quellcode sind Veränderungen mit minimalem Aufwand für jeden möglich und grundsätzlich jederzeit möglich. Da bei eigenen Veränderungen miteinander in Konflikt stehende Versionen sowie u.U. unvorhersehbare Probleme entstehen können, ist eine Verbreitung von modifiziertem Quellcode in Übereinstimmung mit §23 UrhG ohne das schriftliche Einverständnis des Urhebers **nicht gestattet**. Unabhängig hiervon, bin ich immer gerne bereit über mögliche Veränderungen und das Hinzufügen von Funktionen zu diskutieren. So können aus einzelnen Änderungswünschen Vorteile für alle Anwender entstehen. Dieses Vorgehen hat sich in der Vergangenheit bereits mehrfach bewährt.

Supportanfragen werden ausdrücklich nur für offizielle Versionen bearbeitet, um den hiermit entstehenden Aufwand für den Autor in einem akzeptablen Rahmen zu halten. Weiterentwicklungen und Beantwortung von Anfragen erfolgen in einem gemeinsamen Zeitrahmen in der Freizeit des Autors. Es gibt keine garantierten Fristen für eine Antwort. Ich gebe mir jedoch Mühe, alle Anfragen möglichst zeitnah zu bearbeiten.

## **Kommerzielle Nutzung**

Eine kommerzielle Nutzung der hier vorgestellten Software und Beschreibung wird nicht ausgeschlossen. Die schriftliche Einwilligung des Autors ist für jegliche kommerzielle Nutzung unbedingt notwendig. Mögliche Szenarien könnten beispielsweise das Anbieten vorprogrammierter Module oder ganzer Bausätze sein. Da ich viele Stunden/Tage/Wochen in dieses Projekt investiert habe, ist es notwendig, dass ich am Gewinn etwaiger kommerzieller Projekte beteiligt werde. Es wäre schlichtweg nicht fair, wenn jemand aufgrund der Arbeit eines anderen Geld verdient, ohne dass der, der die ursprüngliche Arbeit hatte, keinen Anteil erhält.

## 2. Bedienung

### 2.1. Einführung

**[DIT]** symbolisiert das Betätigen des DIT-Arms des Paddles, so wie **[DAH]** das Betätigen des DAH-Arms bedeutet. Ohne Paddle oder mit einer Handtaste sind viele Funktionen des Keyers nicht verfügbar. **[MENU]** deutet an, dass der Taster zu drücken ist, mit dem man das Menü betritt.

**[Menü]»XYZ** symbolisiert eine Eingabe im Menü, bei der Schritte hintereinander in einer festgelegten Reihenfolge erfolgen. Ein Druck auf **[MENU]** führt zum Betreten des Menüs, was seitens des Keyers mit einem Dit in einer tieferen Tonlage quittiert wird. Wenn keine Eingabe getätigt wird, wird das Menü nach einer kurzen Zeit automatisch wieder verlassen (Ausgabe Dit-dit).

Diese Anleitung beschreibt die erste Inbetriebnahme (S. 6), den Betrieb des Keyers im Allgemeinen (S. 7, sämtliche Menüpunkte (S. 11 bis 21) sowie sonstige Funktionen wie die Wiedergabe (S. 9), die Programmieren von Zeichenspeichern (S. 9) und das Einspielen von Aktualisierungen (S. 22).

### 2.2. Erste Inbetriebnahme

Bei der ersten Inbetriebnahme werden die Grundeinstellungen geladen. Diese sind in Tabelle B.1 auf Seite A6 übersichtlich zusammengefasst.

Die Zeichen-Speicher 1 bis 9 werden vordefiniert (siehe Abschnitt 2.5.2 auf Seite 8).

Das Rufzeichen wird mit „DL6ER“ initialisiert. Es muss das eigene Rufzeichen eingespeichert werden (siehe Menüpunkt **[Menü]»C**, S. 12)!

### 2.3. Betrieb des Keyers

Es ist schwierig etwas zu diesem Thema zu schreiben, denn Telegraphisten, die bereits an ein Paddle gewöhnt sind, können ohne Berücksichtigung von Details sofort anfangen mit dem Keyer zu arbeiten. Üblicherweise sind das eigene Rufzeichen (**[Menü]»C**, S. 12) und die Zeichengeschwindigkeit (**[Menü]»W**, S. 20) einzustellen. Als Ausgangsgeschwindigkeit wurde 15 WpM gewählt. Als Betriebsart wurde die in Europa sehr weit verbreitete Betriebsart *Iambic A* voreingestellt. Selbst wenn man mit dieser Betriebsart nicht zurecht kommt, kann man in der Regel problemlos ins Menü wechseln und sie

dort umstellen (Einzelnes Antippen von **[DIT]** und **[DAH]** kann hilfreich sein). Alternativ können diese, wie auch andere Parameter, über den Computer mit einer Winkey kompatiblen Software (fast alle bekannten Logbuch-Programme sind hierzu in der Lage) verändert werden und sind dann unmittelbar am Keyer verfügbar.

Die Begrüßung des Keyers erfolgt bei angeschlossenem Lautsprecher mit dem einprogrammierten Rufzeichen, der aktuellen Zeichengeschwindigkeit in WpM und einem abschließenden „K“. Sie kann jederzeit unterbrochen werden.

Beispiel: DL6ER 24 K

## 2.4. Abgesetztes Bedienteil

Das abgesetzte Bedienteil ermöglicht den Aufruf des Menüs sowie das direkte Abspielen der neun Zeichenspeicher. Es besteht aus einer Widerstandskette, die durch Taste selektiv kurzgeschlossen wird. Da zur Auswertung des aktuell gedrückten Tasters Analogwerte verwendet werden, reagieren die Zeichenspeichertaster nur im Menü-Modus, da hier sichergestellt ist, dass der Transceiver auf Empfang ist. Ansonsten könnten (bspw. durch Einstrahlung) ungewollte Speichersendungen gestartet werden. Dies wird hierbei vermieden.

## 2.5. Einstellungen zurücksetzen

Normalerweise sollte dies niemals für einen selbst notwendig sein, da veränderte Parameter immer ausdrücklich gespeichert werden müssen (Menüpunkt „S“). Dies sollte erst nach einem gründlichen Test und entsprechender Zufriedenheit mit den neuen Einstellungen geschehen.

Dennoch kann es sein, dass man an einen Keyer (bspw. an einer gemeinschaftlich genutzten Station) kommt und es einem diverse Einstellungen unmöglich machen den Modus umzustellen (viel zu hohe Zeichengeschwindigkeit, für einen unbekannter Betriebsmodus, etc.). In einem solchen Fall kann ein (vorübergehendes) Zurückgehen auf die Standardeinstellungen sinnvoll sein. Dies ermöglicht u.A. auch ein völlig unabhängiges Arbeiten mehrerer Personen am gleichen Keyer, denn:



### Zurücksetzen der Einstellungen 1. Stufe

Beim Zurücksetzen werden die aktuell gespeicherte Einstellungen *nicht* gelöscht! Sollen die alten Einstellungen gelöscht werden, so sind die zurückgesetzten Einstellungen nach einem Reset abzuspeichern ( **[Menü]»S**, S. 19 )! Werden die Einstellungen nach einem Zurücksetzen nicht explizit abgespeichert, so werden beim nächsten Neustart des Keyer wieder die alten Einstellungen geladen!

### 2.5.1. Zurücksetzen 1. Stufe

[DIT]+[DAH] beim Einschalten gedrückt halten.

⇒ Die Standardeinstellungen werden geladen.

Bereits abgespeicherte Einstellungen werden *nicht* gelöscht.

### 2.5.2. Zurücksetzen 2. Stufe

[DIT]+[DAH]+[MENU] beim Einschalten gedrückt halten.

⇒ Setze die Einstellungen sowie die Zeichen-Speicher 1 & 2 zurück.

Evtl. existente Einstellungen werden unwiderruflich durch die Standardeinstellungen ersetzt. Außerdem werden die Zeichen-Speicher teilweise neu programmiert:

Speicher 1: cq cq de <CALL> cq cq de <CALL> <CALL> pse k

Speicher 2: <CALL>

Speicher 3: 5nn tu

Speicher 4: qrz? <CALL>

Speicher 5: de <CALL> - cfm - tnx fer fb qso 73 - de <CALL> k

Hierbei wird <CALL> automatisch durch das eingespeicherte Rufzeichen (s.u.) ersetzt. Selbstverständlich können die Speicher später wieder mit eigenen Inhalten überschrieben werden.

## 2.6. Eigenes Rufzeichen einspeichern

Vor Verwendung der vordefinierten Speicherinhalte muss das eigene Rufzeichen eingespeichert werden. Dies ist über den Menüpunkt [Menü]»C (S. 12) möglich.

Das eigene Rufzeichen kann beim Programmieren der Speicher durch das Prosign „CA“ („C“ und „A“ ohne Pause zwischen den Buchstaben) eingefügt werden.



Wechsel auf PA/..., .../P, .../MM, Clubstationscall, etc.

Bei Wechsel des Rufzeichens brauchen die Zeichenspeicher *nicht neu programmiert* werden. Nach Veränderung des eigenen Rufzeichens über [Menü]»C verwenden die Zeichenspeicher automatisch das neue Rufzeichen.



## 2.7. Zeichenspeicher

### 2.7.1. Wiedergabe

Es gibt mehrere Möglichkeiten eingespeicherte Texte wiederzugeben:

- Im Menü können die Zahlen 1 bis 9 gegeben werden. Diese Befehle beenden das Menü sofort. Die Aussendung des jeweiligen Speichers erfolgt unmittelbar.
- Es ist ebenso möglich die ersten beiden Speicher mit den Abkürzungen **Menü**»**E** (= Speicher 1) und **Menü**»**T** (= Speicher 2) aufzurufen.
- Die dritte Möglichkeit ist der Aufruf der Speicher über das abgesetzte Bedienteil.

### 2.7.2. Aufnahme

Der Keyer verfügt über neun (9) voneinander unabhängige Zeichenspeicher.



#### Nicht belegt muss nicht „leer“ bedeuten!

Möglicherweise sind die für die Zeichenspeicher verwendeten Speicherbereiche des Mikrocontrollers bereits mit anderen Inhalten (evtl. früherer Projekte) belegt. Die Zeichenspeicher sollten daher erst nach einer Aufnahme von Zeichen genutzt werden. Wird dies nicht befolgt, sind unvorhersehbare Effekte möglich.

Die Speicher werden über **Menü**»**R** aufgezeichnet. Auf die Nachfrage **R nr?** kann mit 1 bis 9 sowie „E“ (= Speicher 1) und „T“ (= Speicher 2) geantwortet werden. Außerdem kann der zu programmierende Speicher durch Drücken der entsprechenden Taste des abgesetzten Bedienteils ausgewählt werden. In diesem Fall wird die auszuzeichnende Speichernummer zur Kontrolle wiedergegeben. Nach der Bestätigung durch **K** beginnt die Aufzeichnung des ausgewählten Speichers unmittelbar.

Das eigene Rufzeichen kann beim Programmieren der Zeichenspeicher durch das Prosign „CA“ („C“ und „A“ ohne Pause zwischen den Buchstaben) eingefügt werden. Der Platzhalter für das Rufzeichen wird bei aktivierter Tastaturfunktion als „\$“ an den Computer übergeben.



#### Wechsel auf PA/..., .../P, .../MM, Clubstationscall, etc.

Bei Wechsel des Rufzeichens brauchen die Zeichenspeicher *nicht neu programmiert* werden. Nach Veränderung des eigenen Rufzeichens über **Menü**»**C** verwenden die Zeichenspeicher automatisch das neue Rufzeichen.

Bei aktivierter Tastaturfunktion (**Menü**»**Y**, S. 20) kann der eingegebene Text auf dem Computer (z.B. in einem Textverarbeitungsprogramm) überprüft werden. Fehler können sofort durch eine Irrung korrigiert werden. Es ist ausdrücklich nicht notwendig den Speicher bei einem Fehler komplett neu aufzunehmen. Da die Korrektur bei aktivierter Tastaturfunktion synchron auch auf dem Computer durchgeführt wird, kann man jederzeit seinen einzuspeichernden Text kontrollieren. Nicht erkannte Zeichen werden

im Aufnahme-Modus ignoriert. Löscht man ein Zeichen, so gibt der Keyer bis zu drei vorherige Zeichen erneut aus, sodass man einen Überblick über den aktuellen Speicherinhalt erhält.

Die automatische Einfügung von Leerzeichen ist für die Speicherprogrammierung ebenso von Relevanz wie für die Tastaturfunktion. Sie kann über **Menü** **N** (S. 18) deaktiviert bzw. aktiviert werden. Wenn aktiviert, wird nach dem Ablauf des jeweiligen Pausenabstandes zwischen zwei Wörtern nach allen Zeichen (außer einer Irrung) automatisch ein Leerzeichen eingefügt. Wenn ein falsches Zeichen gegeben wurde und die Irrung hierzu erst nach einer kurzen Pause, so müssen zwei Irrungen gegeben werden, da das automatisch erstellte Leerzeichen ebenfalls gelöscht werden muss (die Wiedergabe der letzten drei Zeichen ist hier sehr hilfreich). Alternativ kann die Pausenzeit mithilfe der Farnsworth-Einstellung vergrößert werden. Wenn das automatische Einfügen von Leerzeichen deaktiviert ist, müssen Pausen zwischen Wörtern manuell eingefügt werden (siehe Pausenzeichen, 5.4.1 auf Seite 28).

Es besteht weiterhin die Möglichkeit Speicher ohne Einsatz eines Paddles alleine über den Computer zu programmieren. Hierfür wird lediglich ein beliebiges, kompatibles Logbuch- oder Contestprogramm benötigt. Sobald der Keyer auf gegebene Zeichen zum Abspeichern wartet, kann man einen Text vom Computer aus geben lassen. Es wird automatisch erkannt, dass aktuell ein Speicher programmiert werden soll. PTT und KEY bleiben inaktiv. Der Keyer interpretiert die übergebenen Zeichen wie manuell gegebene Zeichen. So können vom Computer gegebene Zeichen bei Bedarf sogar noch korrigiert und/oder ergänzt werden.

Die Aufzeichnung kann durch rechtzeitiges Betätigung von **MENU** auch noch nach der Eingabe von Zeichen abgebrochen werden. Der Speicher wird dann nicht verändert und verbleibt im Ausgangszustand.

Wird die Aufzeichnung nicht abgebrochen und wurde mindestens ein Zeichen gegeben, so wird der Speicher nach Erreichen der maximalen Zeit zwischen zwei Buchstaben automatisch abgespeichert. Dies wird mit R <Speichernummer> K quittiert. Die Zeit bis zum automatischen Abspeichern beträgt exakt die doppelte Zeit nach der das Menü ansonsten automatisch verlassen werden würde:

$$t_{\text{saveMem}} = \frac{100 \text{ s}}{\text{Zeichengeschwindigkeit in WpM}} \approx 84 \cdot t_{\text{dit}} \quad (2.1)$$



#### Maximale Speicherlänge

Aus technischen Gründen habe ich eine maximale Speicherlänge von 100 Zeichen festgelegt. So ist neben den neun Zeichenspeichern genügend Platz für die Einstellungen sowie zukünftige Erweiterungen vorhanden.

## 3. Menü

Das Menü des Keyers bietet die Möglichkeit Einfluss auf alle Betriebsparameter zu nehmen. Das Menü wird durch Druck auf den Taster **MENU** betreten.

Befindet man sich im Menü, so werden vom Keyer selbst gegebene Zeichen mit einer Frequenz, die 100 Hz unter der aktuell eingestellten Seitentongfrequenz liegt (tieferer Ton), wiedergegeben, sodass sie klar von eigenen Zeichen zu unterscheiden sind.

Ein Verlassen des Menüs ist durch Warten möglich. Erfolgt eine bestimmte Zeit lang keine Eingabe, so wird das Menü automatisch beendet. Die Zeit bis zum automatischen Verlassen ist von der eingestellten Zeichengeschwindigkeit abhängig, da bei langsamen Zeichengeschwindigkeiten eine längere Zeit benötigt wird um diese einzugeben. Sie berechnet sich nach folgender Gleichung

$$t_{\text{menu}} = \frac{50 \text{ s}}{\text{Zeichengeschwindigkeit in WpM}} \approx 42 \cdot t_{\text{dit}} \quad (3.1)$$

ist jedoch mindestens eine Sekunde ( $\text{WpM} > 50$ ) lang. Während des Einspeicherns in einen der verfügbaren Zeichenspeicher, wird diese Zeitgrenze außer Kraft gesetzt.



### Abkürzung bestimmter Ziffern

Evtl. werden Ziffern abgekürzt. Es gilt dann: N = 9 und T = 0

### 3.1. 1 bis 9 - Sende Speicher 1 bis 9

Der entsprechende Speicher wird gegeben. Das Menü wird sofort verlassen.

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.2. A - Seitenton An/Aus

Abhängig von der aktuellen Einstellung wird der Seitenton entweder aktiviert oder deaktiviert.

- Seitenton ist aktiv  $\Rightarrow$  Seitenton wird deaktiviert.  
Ausgabe: A OFF
- Seitenton ist *nicht* aktiv  $\Rightarrow$  Seitenton wird aktiviert.  
Ausgabe: A ON

Hinweis: Der Seitenton ist innerhalb des Menüs immer aktiv.

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.3. B - Schleifensendung Speicher 1 (Bake)

Der Keyer beginnt mit der Aussendung des ersten Zeichen-Speichers. Nach Abschluss der Sendung wird der Speicher nach einer einstellbaren Zeitspanne (siehe „D“) wiederholt, insofern die Aussendung nicht anderweitig unterbrochen wird.

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.4. C - Eigenes Rufzeichen einspeichern

Mit diesem Befehl kann das eigene Rufzeichen (callsign) geändert werden. Das hier eingespeicherte Callsign wird während der Begrüßung des Keyers, sowie in den vorgegebenen Zeichenspeichern verwendet.

Dieser Menüpunkt erwartet die Eingabe des eigenen Rufzeichens. Es wird nach einer kurzen Wartezeit wiederholt:

Ausgabe: C <eingespeichertes Rufzeichen> K



#### Maximale Rufzeichenlänge

Aus technischen Gründen habe ich eine maximale Speichergröße von 24 Zeichen für das fest eingespeicherte Rufzeichen festgelegt. Dies sollte auch für ausgefallene Konstruktionen (beispielsweise hat PA/DLØEUR/AM nur 12 Zeichen) ausreichen.

### 3.5. D - Schleifensendung Intervall

Mit diesem Befehl wird die Zeit nach Abschluss einer Schleifensendung zum Beginn der nächsten Schleifensendung eingestellt.

Folgende Eingaben sind möglich:

Eingabe	entspricht
1 bis 9	1 s bis 9 s
T	0 s
E	1 s
R (reset)	4 s
N	9 s

Tabelle 3.1.: Parameter für Menüpunkt D

Ausgabe: D <eingestellte Zeit> K

### 3.6. E - Sende Speicher 1 / Geschwindigkeit + 1 WpM

- **MENU** wird gedrückt gehalten  
Zeichengeschwindigkeit wird um 1 WpM erhöht.  
Ausgabe: V in neuer Geschwindigkeit
- **MENU** wird nicht gedrückt gehalten  
Speicher 1 wird gegeben. Das Menü wird sofort verlassen.

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.7. F - Farnsworth Modus

Mit diesem Befehl wird der Farnsworth Parameter  $F$  eingestellt. Bei aktivierten Farnsworth Modus (Wert  $> 0$ ) werden die Pausen zwischen Zeichen / Wörtern künstlich verlängert, indem eine andere (geringere) Zeichengeschwindigkeit für die Berechnung der Längen der Pausen herangezogen wird. Eine detailliertere Beschreibung der Auswirkungen des hier eingestellten Werts ist in Kapitel 5.2.4 auf Seite 25 zu finden.

Folgende Eingaben sind möglich:

Eingabe	entspricht
1 bis 9	10 % bis 90 %
E	10 %
T	0 %

Tabelle 3.2.: Parameter für Menüpunkt F

Ausgabe: F <eingestellter Wert> K



#### Hohe Werte für $F$

Durch hohe Werte für  $F$  kann die Bedienung des Keyers erschwert werden. Daher sollte mit geringen Werten begonnen werden und der Wert nur schrittweise erhöht werden. Außerdem kann eine Verlängerung der Pausenintervalle zu Problemen mit verwendeter Software führen.

### 3.8. G - PTT Nachlaufzeit (geschwindigkeitsunabhängig)

Mit diesem Befehl wird die geschwindigkeitsunabhängige PTT Nachlaufzeit eingestellt. Nachdem der Key-Ausgang abgefallen ist, wartet der Keyer eine festgelegte Zeit, bis dass die PTT abgeschaltet wird. Dies ist insbesondere wichtig um ein zu schnelles Abfallen der PTT zu verhindern. Eine detaillierte Beschreibung der Auswirkungen der hier eingestellten Zeitspanne ist in Kapitel 5.3 auf Seite 26 zu finden. Um die Eingabe zu vereinfachen, sind nur einige übliche Zeitintervalle über das Menü einstellbar. Mithilfe der Computerschnittstelle (Winkey Emulation, siehe Kapitel 6.13 auf Seite 35) kann bei Bedarf jeder Wert von 0 ms bis 2500 ms in 10 ms Schritten eingestellt werden.

Folgende Eingaben sind möglich:

Eingabe	entspricht
0 bis 9	0 ms bis 90 ms
T	0 ms
E	100 ms
N	900 ms

Tabelle 3.3.: Parameter für Menüpunkt G

Ausgabe: G <Pause in msec> K

### 3.9. H - PTT Nachlaufzeit (geschwindigkeitsabhängig)

Mit diesem Befehl wird die geschwindigkeitsabhängige PTT Nachlaufzeit in Schritten von einzelnen Ditlängen eingestellt. Nachdem der Key-Ausgang abgefallen ist, wartet der Keyer eine festgelegte Zeit, bis dass die PTT abgeschaltet wird. Dies ist insbesondere wichtig um ein zu schnelles Abfallen der PTT zu verhindern. Eine detaillierte Beschreibung der Auswirkungen der hier eingestellten Zeitspanne ist in Kapitel 5.3 auf Seite 26 zu finden.

Folgende Eingaben sind möglich:

Eingabe	entspricht
0 bis 9	0 bis 9 Ditlängen
T	0 Ditlängen
E	3 Ditlängen
I	6 Ditlängen
N	9 Ditlängen
S	18 Ditlängen
H	36 Ditlängen

Tabelle 3.4.: Parameter für Menüpunkt H

Ausgabe: H <Pause in Ditlängen> K

### 3.10. I - Invertiere KEY-Ausgang

Abhängig von der aktuellen Einstellung wird der *Aktiv*-Zustand der KEY-Leitung gewechselt.

- Normal  $\Rightarrow$  Optokoppler aktiviert, wenn KEY angefordert wird. Sonst deaktiviert.  
Ausgabe: I OFF
- Invertiert  $\Rightarrow$  Optokoppler deaktiviert, wenn KEY angefordert wird. Sonst aktiviert.  
Ausgabe: I ON

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.11. J - Zeichenerkennungstoleranz

Mithilfe der Zeichenerkennungstoleranz kann man selbst festlegen, wie strikt der Zeichenerkennungsalgorithmus bei der Erkennung des Ende eines Zeichens vorgehen soll. Die Details zum Zeichenerkennungsalgorithmus sind in Kapitel 5.5 auf Seite 29 erläutert.

Folgende Eingaben sind möglich:

Eingabe	J	Pause
0 bis 8	0 bis 8	0 bis 2 Ditlängen
T	0	0 Ditlängen
E	1	0.25 Ditlängen
R (reset)	4	1 Ditlänge

Tabelle 3.5.: Parameter für Menüpunkt J

Ausgabe: J <Wert> K

### 3.12. K - Key/Paddle Betriebsmodus

Mit diesem Befehl wird die Betriebsart des angeschlossenen Paddles eingestellt.

- Iambische Betriebsarten
  - A: „europäischer“ Iambic-Modus. Keine Extrasendung nach Öffnen beider Kontakte. Das aktuell gesendete **DIT** oder **DAH** wird zu Ende gesendet.

- B: „amerikanischer“ Iambic-Mode. Werden beide Kontakte während der Sendung bspw. eines DIT geöffnet, so wird das DIT wie bei Iambic A fertig gestellt. *Zusätzlich* wird das gegenteilige Element nach dem Öffnen beider Kontakte (hier DAH) gesendet.
- Ultimatic  
Der zuletzt geschlossene Kontakt übernimmt die führende Rolle.
- Ein-Hebel Paddle Emulation  
Der zuerst geschlossene Kontakt übernimmt die führende Rolle. Diese Betriebsart wurde von Sverre Hol, LA3ZA, vorgeschlagen.
- Bug  
Im Bug-Modus formt man die DAH selbst, während der Keyer auf Betätigung der DIT Taste automatisch Punkte in der eingestellten Geschwindigkeit erzeugt. Zeichenerkennung in diesem Modus ist noch experimentell.
- Handtaste  
In diesem Modus wird ein Großteil der Keyer-Funktionalität deaktiviert. Die Zeichen werden ausschließlich selbst geformt, es gibt keinen Unterschied zwischen den DIT und DAH Eingängen.

Folgende Eingaben sind möglich:

Eingabe	entspricht
A	Iambic A
B	Iambic B
U	Ultimatic
S	Ein-Hebel Paddle Emulation
G	Bug
K	Handtaste (Straight key)

**Tabelle 3.6.:** Parameter für Menüpunkt K

Ausgabe: <eingestellter Modus> K



#### Handtaste

Durch die mangelnde Vorgabe einer Zeichengeschwindigkeit wird die Zeichenerkennung im Betriebsmodus Handtaste deaktiviert. Generierung von Morsecode mit dem Computer und Geben aus Speichern ist zwar weiterhin möglich, jedoch ist das Menü nicht zugänglich.



### 3.13. L - PTT Vorlaufzeit

Mit diesem Befehl wird die geschwindigkeitsunabhängige PTT Vorlaufzeit in Schritten von 10 Millisekunden eingestellt. Wird ein Tasten des KEY-Ausgangs angefordert (beispielsweise durch Berühren des Paddles oder durch Anforderung eines Zeichenspeichers oder Zeichengabe durch den Computer), so wird die PTT sofort aktiviert. Eine detaillierte Beschreibung der Auswirkungen der hier eingestellten Zeitspanne ist in Kapitel 5.3 auf Seite 26 zu finden. Um die Eingabe zu vereinfachen, sind nur einige übliche Zeitintervalle über das Menü einstellbar. Mithilfe der Computerschnittstelle (Winkey Emulation, siehe Kapitel 6.13 auf Seite 35) kann bei Bedarf jeder Wert von 0 ms bis 2500 ms in 10 ms Schritten eingestellt werden.

Folgende Eingaben sind möglich:

Eingabe	entspricht
T	0 ms
0 bis 9	0 ms bis 90 ms
E	100 ms
I	200 ms
S	300 ms
H	400 ms
N	900 ms

Tabelle 3.7.: Parameter für Menüpunkt L

Ausgabe: L <Pause in msec> K



#### Timeouts möglich

Durch eine hohe Vorlaufzeit könnte es Probleme mit hierfür nicht ausgelegter Software geben, denn über die Winkey-Schnittstelle wird erst der erste Buchstabe erst zurückgegeben, wenn er tatsächlich gegeben wird. Durch die PTT Vorlaufzeit können offensichtlich hohe Verzögerungen zwischen Anforderung und tatsächlichem Beginn des Sendens auftreten. Dies könnte von einem fälschlicherweise als Nicht-Reagieren des Keyers gedeutet werden.

### 3.14. M - Stummschaltung für TRX-Leitungen

Abhängig von der aktuellen Einstellung werden die KEY- und PTT-Leitungen entweder stummgeschaltet oder die Stummschaltung wird aufgehoben.

- Stummschaltung ist *nicht* aktiv  $\Rightarrow$  Stummschaltung wird aktiviert.  
Ausgabe: M ON
- Stummschaltung ist aktiv  $\Rightarrow$  Stummschaltung wird deaktiviert.  
Ausgabe: M OFF

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.



#### Stummschaltung und Seitenton

Damit der Keyer bei Stummschaltung der Leitungen zum Transceiver weiterhin bedienbar bleibt, wird der Seitenton aktiviert. Der Seitenton wird *nicht* automatisch wieder deaktiviert.

### 3.15. N - Automatisches Leerzeichen zwischen Wörtern

Abhängig von der aktuellen Einstellung werden das automatische Einfügen von Leerzeichen nach Buchstaben aktiviert oder deaktiviert. Details zu diesem Punkt sind in Kapitel 5.5.1 auf Seite 30 aufgeführt.

- Funktion *nicht* aktiv  $\Rightarrow$  Funktion wird aktiviert.  
Ausgabe: N ON
- Funktion aktiv  $\Rightarrow$  Funktion wird deaktiviert.  
Ausgabe: N OFF

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.16. O - TRX-Leitungen 1 & 2 umschalten



Abhängig von der aktuellen Auswahl wird zwischen den Ausgängen 1 & 2 umgeschaltet. Dies betrifft sowohl PTT als auch KEY.

- Ausgang 1 ist ausgewählt  $\Rightarrow$  Ausgang 2 wird ausgewählt.  
Ausgabe: O 2
- Ausgang 2 ist ausgewählt  $\Rightarrow$  Ausgang 1 wird ausgewählt.  
Ausgabe: O 1

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.17. P - Programmiermodus betreten

Da der Arduino durch einen Trick über die serielle Schnittstelle neu gestartet werden kann (z.B. um eine Programmierung einzuleiten), muss dieses Verhalten für den Normalbetrieb unterdrückt werden um spontane Neustarts zu vermeiden. Um das Arduino-Board dennoch programmieren zu können, gibt es drei Möglichkeiten:

1. Versetzen des Keyers in den Programmiermodus über  .
2. Ein externer Reset des Mikrocontrollers bringt diesen in den Bootloader-Modus, in dem er programmiert werden kann.

3. Beim Einschalten (oder nach einem Reset) kann der Keyer durch gedrückt halten von **MENU** in den Programmiermodus versetzt werden.

Hinweis: Der Keyer kann im Programmiermodus über die serielle Schnittstelle zum Neustarten gezwungen werden. Dies ist für die Programmierung notwendig, da das auf dem Arduino gespeicherte Programm nicht im laufenden Betrieb überschrieben werden kann.

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.18. Q - nicht belegt

### 3.19. R - Speicher aufzeichnen

Siehe Kapitel 2.7.2 auf Seite 9.

### 3.20. S - Einstellungen dauerhaft abspeichern

Die aktuellen Einstellungen werden im EEPROM-Speicher des Mikrokontroller gespeichert. Hier bleiben sie auch ohne Stromzufuhr dauerhaft erhalten und können beim nächsten Start wiederhergestellt werden.

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.21. T - Sende Speicher 2 / Geschwindigkeit - 1 WpM

- **MENU** wird gedrückt gehalten  
Zeichengeschwindigkeit wird um 1 WpM verringert.  
Ausgabe: V in neuer Geschwindigkeit
- **MENU** wird nicht gedrückt gehalten  
Speicher 2 wird gegeben. Das Menü wird sofort verlassen.

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.22. U - Dauersendung (Abstimmen)

KEY + PTT (evtl. mit Verzögerung) werden aktiviert. Das Menü wird sofort verlassen.

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.23. V - nicht belegt

### 3.24. W - Zeichengeschwindigkeit in WpM

Die Geschwindigkeit kann in ganzen WpM-Schritten geändert werden.

- **[DIT]** ⇒ Geschwindigkeit wird um 1 WpM erhöht  
Ausgabe: V in neuer Geschwindigkeit
- **[DAH]** ⇒ Geschwindigkeit wird um 1 WpM verringert  
Ausgabe: V in neuer Geschwindigkeit

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.25. X - Paddle-Seiten vertauschen

Die Eingänge **[DIT]** und **[DAH]** werden getauscht.

- **[DIT]** und **[DAH]** *nicht* vertauscht ⇒ **[DIT]** und **[DAH]** werden vertauscht.  
Ausgabe: X **[DAH]** **[DIT]**
- **[DIT]** und **[DAH]** vertauscht ⇒ **[DIT]** und **[DAH]** werden richtig zugewiesen.  
Ausgabe: X **[DIT]** **[DAH]**

Dieser Menüpunkt bietet keine weiteren Eingabemöglichkeiten.

### 3.26. Y - Tastaturfunktion einstellen

Mit diesem Befehl wird die Zeichenübergabe an den Computer mithilfe der Tastaturemulation eingestellt.

Folgende Eingaben sind möglich:

Eingabe	entspricht
T	deaktiviert
P	nur Paddle
M	Zeichenspeicher und Paddle
A	alle Zeichen

**Tabelle 3.8.:** Parameter für Menüpunkt Y

Ausgabe: K

### 3.27. Z - Seitenton Frequenz einstellen

Mit diesem Befehl wird die Frequenz des Seitentons eingestellt.

Folgende Eingaben sind möglich:

Eingabe	entspricht
2 bis 9	200 Hz bis 900 Hz
R (reset)	700 Hz
N	900 Hz

**Tabelle 3.9.:** Parameter für Menüpunkt Z

Ausgabe: Z <eingestellte Frequenz in Hz> K

## 4. Programmierung / Einspielen von Aktualisierungen

Wenn kein vorprogrammiertes Modul als Grundlage des Bausatzes dient, muss vor der Inbetriebnahme eine Programmierung des Moduls erfolgen. Hierfür kann die beliebte und für alle gängigen Betriebssysteme verfügbare Arduino-Software verwendet werden.

Quelle: <http://arduino.cc/en/pmwiki.php?n=Main/Software>

Der grobe Ablauf der Programmierung (nach Installation der Arduino-Software) ist:

1. Download der aktuellen Version der Keyer-Software (siehe <http://blog.dl6er.de/DL6ER-Keyer.html>)

Aufgrund der Art der automatischen Erstellung dieses Archivs zum Zeitpunkt des Downloads ist garantiert, dass nur Text-Dateien enthalten sind. Es ist somit immer virenfrei.

2. Der Inhalt des heruntergeladenen Archivs muss in einen neuen Ordner namens „DL6ER\_Keyer“ entpackt werden. Existiert dieser Ordner bereits, so muss der Inhalt bei geschlossener Arduino-Software zuvor entfernt werden.
3. Die Arduino-Software wird mit der Datei `DL6ER_Keyer.ino` gestartet.
4. Vor der Programmierung müssen unter »Werkzeuge: Platine« (Tools: Board) sowie »Werkzeuge: Port« (Tools: Port) die richtigen Werte ausgewählt werden.
5. Wenn das Arduino-Board noch nicht programmiert ist, kann mit dem folgenden Schritt fortgefahren werden. Falls eine existierende Softwareversion aktualisiert werden soll, muss der Keyer zuerst in den Programmiermodus versetzt werden (siehe 4.1).
6. Nach einem Klick auf "Hochladen" (alternativ Tastenkombination **Strg** + **U**) wird der Programmcode in Maschinensprache übersetzt (kompiliert) und auf das Arduino-Board hochgeladen.

Die Programmierung/Aktualisierung ist hiermit abgeschlossen

## 4.1. Besonderheit: Aktualisierung

Da der Arduino durch einen Trick über die serielle Schnittstelle neu gestartet werden kann (z.B. um eine Programmierung einzuleiten), muss dieses Verhalten für den Normalbetrieb unterdrückt werden um spontane Neustarts zu vermeiden. Um das Arduino-Board dennoch programmieren zu können, gibt es drei Möglichkeiten:

1. Versetzen des Keyers in den Programmiermodus über `Menü` `»` `P`.
2. Ein externer Reset des Mikrocontrollers bringt diesen in den Bootloader-Modus, in dem er programmiert werden kann.
3. Beim Einschalten (oder nach einem Reset) kann der Keyer durch gedrückt halten von `MENU` in den Programmiermodus versetzt werden.

Hinweis: Der Keyer kann im Programmiermodus über die serielle Schnittstelle zum Neustarten gezwungen werden. Dies ist für die Programmierung notwendig, da das auf dem Arduino gespeicherte Programm nicht im laufenden Betrieb überschrieben werden kann.

## 4.2. Arduino-Module

Originale Arduino-Module (die nicht von einem Fremdanbieter hergestellt werden), werden ohne weitere Eingriffe von der Software erkannt. Unter Windows®-Betriebssystemen werden die benötigten Treiber bei der Installation der Arduino-Software mitgeliefert.

## 4.3. Sparkfun Pro Micro

Die Anleitung ist leider nur in Englisch verfügbar.

### 4.3.1. Linux/Mac: Arduino-Software

Unter Linux und Mac sind zum Betrieb dieses Boards *keine* Treiber erforderlich.

<https://learn.sparkfun.com/tutorials/pro-micro-fio-v3-hookup-guide#installing-mac-linux>

### 4.3.2. Windows®: Treiber & Arduino-Software

Unter Windows® sind zum Betrieb dieses Boards Treiber erforderlich.

<https://learn.sparkfun.com/tutorials/pro-micro-fio-v3-hookup-guide#installing-windows>

## 5. Technische Beschreibung

### 5.1. Konfiguration der Anschlüsse

Name	Beschreibung	Konfiguriert als	Pin am Pro Micro
CONN_DIT	Paddle DIT	INPUT_PULLUP	13 (D10)
CONN_DAH	Paddle DAH	INPUT_PULLUP	14 (D16)
CONN_BUTTON	Taster Anschluss	INPUT (ADC)	18 (A1)
CONN_KEY1	KEY Ausgang 1	OUTPUT	20 (A3)
CONN_KEY2	KEY Ausgang 2	OUTPUT	19 (A2)
CONN_PTT1	PTT Ausgang 1	OUTPUT	16 (D15)
CONN_PTT2	PTT Ausgang 2	OUTPUT	15 (D14)
CONN_SPEAKER	Lautsprecher	OUTPUT	17 (A0)
CONN_K1	Relais 1	OUTPUT	6 (D3)
CONN_K2	Relais 2	OUTPUT	5 (D2)
CONN_K3	Relais 3	OUTPUT	2 (D0)
CONN_K4	Relais 4	OUTPUT	1 (D1)
CONN_K5	Relais 5	OUTPUT	7 (D4)
CONN_K6	Relais 6	OUTPUT	11 (D8)
CONN_K7	Relais 7	OUTPUT	9 (D6)
CONN_K8	Relais 8	OUTPUT	10 (D7)
CONN_K9	Relais 9	OUTPUT	8 (D5)
CONN_K10	Relais 10	OUTPUT	12 (D9)

Tabelle 5.1.: Zuordnung der Anschlüsse entsprechend dem angebotenen Bausatz

### 5.2. CW Timing

#### 5.2.1. Dit- / Dahlänge

Die typische Geschwindigkeitsangabe Wörter pro Minute (WpM) bezieht sich auf das Wort *PARIS* mit insgesamt 50 Ditlängen. Wird das Wort *PARIS* zehnmal pro Minute gegeben, so ergibt sich eine Geschwindigkeit von  $G = 10$  WpM.

Hieraus folgt unmittelbar

$$t_{\text{dit}} = \frac{60 \text{ s}}{50 \cdot G} = \frac{1200 \text{ ms}}{G} . \quad (5.1)$$



Da die Einheit der Zeichengeschwindigkeit  $G$  (WpM) einen ausschließlich symbolischen Wert hat, ist die Einheit der Ditlänge die (Milli-)Sekunde.

Die Dahlänge wird direkt aus der Ditlänge berechnet:

$$t_{\text{dah}} = 3 \cdot t_{\text{dit}} \quad (5.2)$$

### 5.2.2. Abstand zwischen Buchstaben

Zum zuletzt gegeben Buchstaben gehört bereits eine Ditlänge als Pause, sodass zwischen Buchstaben zwei weitere Ditlängen eingefügt werden müssen.

$$t_{\text{letter}} = (1 + 2) \cdot t_{\text{dit}} = 3 \cdot t_{\text{dit}} = t_{\text{dah}} \quad (5.3)$$

### 5.2.3. Abstand zwischen Wörtern

Zum zuletzt gegeben Buchstaben gehört bereits eine Pause zwischen Buchstaben, sodass zwischen Wörtern vier weitere Ditlängen eingefügt werden müssen.

$$t_{\text{letter}} = (3 + 4) \cdot t_{\text{dit}} = t_{\text{letter}} + 4 \cdot t_{\text{dit}} \quad (5.4)$$

Zeichengeschwindigkeit $G$	1	5	10	15	20	30	40	60
$t_{\text{dit}}$ [ms]	1200	240	120	80	60	40	30	20
$t_{\text{dah}}$ [ms]	3600	720	360	240	180	120	90	60
$t_{\text{word}}$ [ms]	8400	1680	840	560	420	280	210	140

Tabelle 5.2.: Zeitintervalle zu unterschiedlichen Zeichengeschwindigkeiten  $G$

### 5.2.4. Farnsworth Timing: Verlängerte Pausen

Ist ein von Null verschiedener Wert für das Farnsworth Timing ( $F$ ) eingestellt (siehe Menü F, Kapitel 3.7 auf Seite 13), so werden die Pausen (5.3) und (5.4) aufgrund der Farnsworth-Ditlänge berechnet:

$$t_{\text{dit,Farnsworth}} = \frac{1200 \text{ ms}}{G \cdot (100\% - 10\% \cdot F)} \quad (5.5)$$

Beispiel: Aus  $G = 20$  WpM folgt  $t_{\text{dit}} = 60$  ms. Diese Zeit dient als Grundlage für alle abgeleiteten Zeiten. Für  $F > 0$  werden die Pausenzeiten jedoch abweichend nach (5.5) berechnet. Für  $F = 2$  ergibt sich folglich eine 20%ige Verlängerungen der Pausen gegenüber  $F = 0$ . In diesem Fall wird die Farnsworth-Ditlänge  $t_{\text{dit,Farnsworth}} = 75$  ms verwendet, um die Pausen zu berechnen.

### 5.3. PTT Timing

Der PTT-Ausgang kann verwendet werden um beispielsweise nicht Full-BK taugliche Geräte an den Keyer anschließen zu können. So ist es möglicherweise notwendig die PTT eine gewisse Zeit vor dem eigentlichen Geben der Zeichen zu aktivieren, damit eine langsame Sende-Empfangs-Umschaltung aktiviert werden kann. Die einzelnen Einstellmöglichkeiten der PTT-Verzögerung werden im Folgenden aufgelistet.

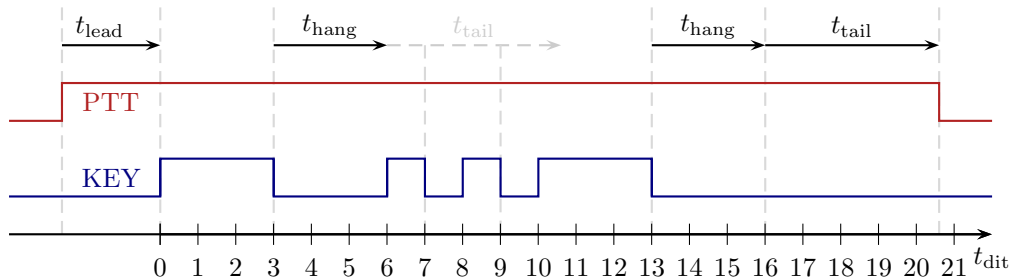


Abbildung 5.1.: Beispielhaftes Schaltverhalten der Leitungen PTT/KEY zum Buchstaben „U“.  $G = 25 \text{ WpM}$ ,  $t_{\text{lead}} = 100 \text{ ms}$ ,  $t_{\text{hang}} = 3 \text{ Ditlängen}$ ,  $t_{\text{tail}} = 200 \text{ ms}$

#### 5.3.1. PTT Vorlaufzeit $t_{\text{lead}}$

Wird ein Tasten des KEY-Ausgangs angefordert (beispielsweise durch Berühren des Paddles oder durch Anforderung eines Zeichenspeichers oder Zeichengabe durch den Computer), so wird die PTT sofort aktiviert. Wenn eine Vorlaufzeit definiert ist, wird der KEY-Ausgang erst nach Ablauf dieser Zeitspanne aktiviert ( $t_{\text{lead}}$  in Abbildung 5.1).

Die PTT-Vorlaufzeit kann entweder unter **Menü** **L** (Kapitel 3.13 auf Seite 17) oder über die Winkey-Emulation eingestellt werden.

#### 5.3.2. PTT Nachlaufzeit $t_{\text{hang}} + t_{\text{tail}}$

Nachdem der Key-Ausgang abgefallen ist, wartet der Keyer eine festgelegte Zeit, bis dass die PTT abgeschaltet wird. Dies ist insbesondere wichtig um ein zu schnelles Abfallen der PTT zu verhindern. Wenn eine Nachlaufzeit definiert ist, wird der PTT-Ausgang erst nach Ablauf dieser Zeitspanne deaktiviert ( $t_{\text{hang}} + t_{\text{tail}}$  in Abbildung 5.1).


Die PTT-Nachlaufzeit setzt sich aus einem geschwindigkeitabhängigen und einem geschwindigkeitsunabhängigen Teil zusammen, deren Summe als Zeitintervall ablaufen muss, bevor die PTT abfällt. Die Zeiten können unabhängig voneinander über **Menü** **H** (Kapitel 3.9 auf Seite 14) sowie **Menü** **G** (Kapitel 3.8 auf Seite 14) oder über die Winkey-Emulation eingestellt werden.



#### Genaueres Timing ist nicht garantiert!

Während der Einschaltzeitpunkt ohne Verzögerung identisch mit dem des KEY-Ausgangs ist, kann es bei deaktivierter Nachlaufzeit ( $t_{\text{hang}} + t_{\text{tail}} = 0$ ) trotzdem zu leichten Unterschieden zwischen KEY und PTT kommen. Unabhängig hiervon wird jedoch garantiert, dass die PTT erst *nach* Abfall des KEY-Ausgangs deaktiviert wird.

## 5.4. Tastaturfunktion des Keyers

Gegebene Buchstaben / Zahlen / Zeichen werden vom Keyer ausgewertet (vgl. Kapitel 5.5) und bei aktivierter Tastaturfunktion (  ) an den Computer übertragen. Der Keyer gibt sich hierfür für den Computer als gewöhnliche USB-Tastatur aus. Bei den meisten Betriebssystemen ist noch nicht einmal die Installation eines Treibers notwendig. Auf Wunsch kann zwischen Wörtern ein automatisches Leerzeichen eingefügt werden (siehe Kapitel 5.5.1).

Zusätzlich zum gewöhnlichen Buchstabenalphabet und den Zahlen 0 bis 9 werden folgende Zeichen unterstützt:

Zeichen	entspricht
.	ETETET
=	TEEET
,	TTEETT
/	TEETE
\	ETEETE
?	EETTEE
!	TETETT
'	ETTTTE
(	TETTE
)	TETTET
+, $\overline{\text{AR}}$	ETETE
-	TEEEET
:	TTTEEE
;	TETETE
ä	ETET
ü	EETT
ö	TTTE
ß	EEETTEE
@	ETTETE
$\overline{\text{CQ}}$	TETETTET
$\overline{\text{SK}}$	EEETET
$\overline{\text{BK}}$	TEEETET
$\overline{\text{KN}}$	TETTE

Tabelle 5.3.: Zusätzliche Morsezeichen, E = DIT, T = DAH

### 5.4.1. Steuerzeichen



Um auch längere Texte (z.B. E-Mails) angenehm und ausschließlich mit dem Paddle schreiben zu können, habe ich vier nicht im internationalen Morsealphabet definierte Steuerzeichen hinzugefügt, die im Folgenden kurz aufgeführt werden.

Unabhängig von der Einstellung zum automatischen Einfügen eines Leerzeichens, wird nach Geben eines Steuerzeichens *kein* Leerzeichen eingefügt.


#### Irrung (EEEEEE...)

Durch das Geben einer Irrung wird das Zeichen links von der aktuellen Position (d.h. im Allgemeinen das zuletzt eingegebene Zeichen) gelöscht.

#### Leerzeichen (TTTTTT...)

Durch das Geben dieses Zeichens wird ein Leerzeichen eingefügt. Dieses Steuerzeichen wird benötigt, wenn das automatische Einfügen von Leerzeichen ( , S. 18) deaktiviert ist.





#### Groß- / Kleinschreibung (ETETETET...)

Durch dieses Zeichen wird die linke Shift-Taste () der virtuellen USB-Tastatur gedrückt. Das nächste Zeichen wird somit groß geschrieben. Nach dem nächsten *gültigen* Zeichen wird die Shift-Taste automatisch wieder freigegeben.



#### Wirkung auf Zahlen und Zeichen

Gibt man andere Zeichen als Buchstaben, so können wie auf einer gewöhnlichen Tastatur Sonderzeichen gegeben werden. Auf deutschem Tastaturlayout unter anderem:

 + „ 1 “ = „ ! “    + „ 5 “ = „ % “    + „ . “ = „ : “    + „ + “ = „ \* “

#### Zeilenvorschub / Absatz (TETETETE...)

Es wird ein Absatz eingefügt.

## 5.5. Zeichenerkennungsalgorithmus

Der Zeichenerkennungsalgorithmus speichert  $\boxed{\text{DIT}}$  und  $\boxed{\text{DAH}}$  nur bei korrekter Gebeweise (genügend Pause zwischen den Buchstaben). Eine saubere Pause zwischen  $\boxed{\text{DIT}}$  und  $\boxed{\text{DAH}}$  wird in allen Paddle-Betriebsarten garantiert. Durch unsaubere Gebeweise, d.h. zu langen Pausen zwischen den  $\boxed{\text{DIT}}$  und  $\boxed{\text{DAH}}$  kann ein Buchstabe nicht korrekt erkannt werden, wenn auf perfektem Timing bestanden wird. Die Zeichenerkennungstoleranz  $J$  ist daher ein wesentlicher Parameter des Zeichenerkennungsalgorithmus des Keyers. Wenn die Zeichenerkennungstoleranz  $J$  auf einen Wert größer Null gesetzt wird, wartet der Keyer einen längeren Zeitraum, bevor ein Zeichen ausgewertet wird. Sobald ein Buchstabe als abgeschlossen angesehen wird, wird er mithilfe der internen Morsecodetabelle umgewandelt.

Mithilfe der Zeichenerkennungstoleranz kann man selbst festlegen, wie strikt der Algorithmus bei der Erkennung des Ende eines Zeichens vorgehen soll.

Wie aus Abbildung 5.2 und außerdem Tabelle 3.5 auf Seite 15 ersichtlich, kann  $J$  Werte zwischen Null und Acht einnehmen.

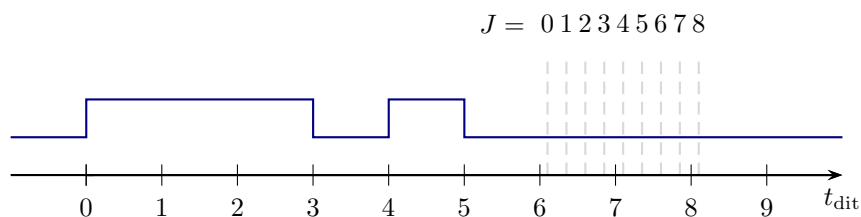


Abbildung 5.2.: Zeitpunkte der verschiedenen Zeichenerkennungstoleranzen nach einem „N“

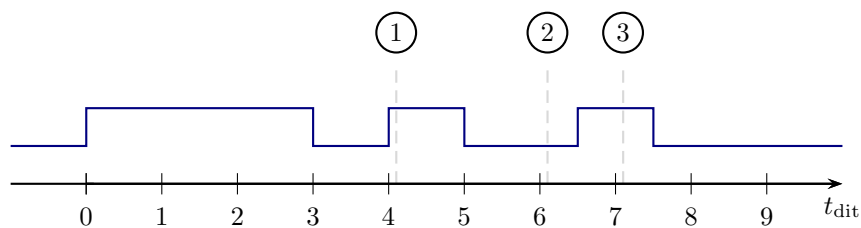


Abbildung 5.3.: Beispiel zur Zeichenerkennungstoleranz

Im oben gezeigten Beispiel kann das „D“ nicht erkannt werden, wenn man auf perfektem Timing ( $J = 0$ ) besteht, da das letzte  $\boxed{\text{DIT}}$  etwas zu spät einsetzt. In diesem Fall hätte der Keyer lediglich „ $\boxed{\text{DAH}} \boxed{\text{DIT}}$ “ abgespeichert, da eine weitere Ditlänge nach dem Ende des  $\boxed{\text{DIT}}$  keine weitere Aktivität des Paddles festgestellt wurde. Das Zeichen wird ausgewertet und ergibt somit „N“. Das darauffolgende einzelne  $\boxed{\text{DIT}}$  wird als „E“ gewertet. Die gesamte Ausgabe wäre demnach: **ne**.

- ① Nach dem DAH wird der Buchstaben nicht als abgeschlossen gewertet, da innerhalb der Kontrollzeit für  $J = 0$  ein DIT eingesetzt hat.
- ② Nach diesem DIT wird der Buchstabe als abgeschlossen gewertet, da innerhalb der Kontrollzeit für  $J = 0$  kein weiteres DIT oder DAH eingesetzt hat. Es wird der Buchstabe „N“ ausgegeben.
- ③ Wenn man die Zeichenerkennungstoleranz höher setzt (entspricht in diesem Fall  $J = 4$ ), so wird die Wartezeit nach dem DIT größer. Da hier das weitere DIT als zum Buchstaben zugehörig erkannt wird, kann der Buchstabe korrekt als „D“ ausgewertet werden.

### 5.5.1. Automatisches Leerzeichen zwischen Wörtern

Es ist möglich Leerzeichen automatisch zwischen Wörtern einfügen zu lassen wenn die Pause zwischen zwei Buchstaben größer ist als

$$t_{\text{space}} = t_{\text{word}} + J \cdot t_{\text{dit}} = (J + 7) \cdot t_{\text{dit}} \quad (5.6)$$

wobei  $t_{\text{word}}$  die Zeit zwischen zwei Wörtern (vgl. Kapitel 5.2.3 auf Seite 25) und  $J$  die Zeichenerkennungstoleranz ist.

Diese Funktion kann mit Menü N aktiviert und deaktiviert werden.

## 6. Winkey Emulation

Der Keyer setzt eine Minimalimplementierung des Winkey2-Protokolls mit Kompatibilität zum Winkey1-Protokoll um. Zu den Winkey-Befehlen, werden zusätzlich weitere Befehle akzeptiert. Es folgt eine Auflistung der akzeptierten Befehle und der daraus resultierenden Wirkungen am Keyer. Alle Befehle, die mit „0x“ beginnen, sind in hexadezimaler Schreibweise angegeben. Von Computer an den Keyer übergebene Befehle werden grün dargestellt. Vom Keyer an den Computer gesendete Nachrichten, werden rot gezeigt.

### 6.1. Echo

Computer → Keyer: 0x00 0x04 0xXX

Computer ← Keyer: 0xXX

Das nächste übertragene Byte nach 0x01 0x04 (hier als 0xXX gezeigt) wird an den Computer zurückgegeben.

Dieser Befehl ist nicht blockierend. Das bedeutet, dass der Keyer nur eine kurze Zeit nach Empfang des Befehls 0x01 0x04 wartet, dass ein Byte ankommt, welches dann zurückgegeben wird. Wird dieses Byte nicht innerhalb kurzer Zeit empfangen, so wird der Befehl verworfen und kein Zeichen zurückgegeben.

### 6.2. Aktiviere Host-Modus

Computer → Keyer: 0x00 0x02

Computer ← Keyer: 0x23 0xXX 0xC0

Der Host-Modus wird aktiviert. Der Host-Modus muss aktiv sein, damit Befehle und Zeichen, die vom Computer kommen und nicht mit 0x01 beginnen, akzeptiert und ausgewertet werden.

Der erste zurückgegebene Wert (0x23) entspricht der zugrundeliegenden Winkey Protokoll-Revision.

Der zweite zurückgegebene Wert (0xXX) entspricht der kodierten aktuellen Zeichengeschwindigkeit, berechnet nach folgender Formel:

$$B_{\text{WpM}} = 0x80 + G_{\text{WpM}} \& 0x3F \quad (6.1)$$

Der dritte zurückgegebene Wert (**0xC0**) signalisiert, dass der Keyer jetzt in Bereitschaft ist.

### 6.3. Deaktiviere Host-Modus

Computer → Keyer: **0x00 0x03**

Computer ← Keyer: —

### 6.4. Setze Zeichengeschwindigkeit

Computer → Keyer: **0x02 0xXX**

Computer ← Keyer: —

Die übermittelte Zeichengeschwindigkeit (in WpM) wird eingestellt, insofern der Wert innerhalb der definierten Grenzen für die minimale und die maximale Geschwindigkeit liegt.

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

### 6.5. PTT Lead-In/Tail

Computer → Keyer: **0x04 0xXX 0xYY**

Computer ← Keyer: —

Die übermittelten Zeitintervalle werden eingestellt.

Der erste Wert **0xXX** wird für die Lead-In Zeit, der zweite Wert **0xYY** für die Tail Zeit eingestellt.

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

### 6.6. Zeichengeschwindigkeit abfragen

Computer → Keyer: **0x07**

Computer ← Keyer: **0xXX**

Der zurückgegebene Wert (**0xXX**) entspricht der kodierten aktuellen Zeichengeschwindigkeit, berechnet folgender Formel (6.1) (siehe S. 31).

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.



## 6.7. Ausgang wählen

Computer → Keyer: 0x09 0xXX

Computer ← Keyer: —

Entsprechend dem übergebenen Argument wird der aktuelle Ausgang ausgewählt.

- 0x09 0x04 PTT 1 + Key 1
- 0x09 0x08 PTT 2 + Key 2

Sind beide Bits gesetzt (0x12), so wird Ausgang 1 gewählt. Ist das empfangene Byte nicht mit obigen Regeln zu vereinbaren, so wird es ohne Änderung ignoriert.

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

## 6.8. Breche Sendung ab

Computer → Keyer: 0x0A

Computer ← Keyer: 0xC0

Die Aussendung der gepufferten Zeichen wird sofort unterbrochen und der Puffer wird geleert. Eventuell bereits an den Keyer gesendete Zeichen (Puffer der seriellen Schnittstelle) werden abgerufen und verworfen.

Der zurückgegebene Wert (0xC0) signalisiert, dass der Keyer jetzt in Bereitschaft ist.

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

## 6.9. Tune TRX

Computer → Keyer: 0x0B 0xXX

Computer ← Keyer: —

Entsprechend dem übergebenen Argument wird der aktuell gewählte TRX auf Sendung oder Empfang geschaltet.

- 0x0B 0x00 Empfang
- 0x0B 0x01 Dauersendung

Ist das Byte ungleich 0x00, so wird der Transceiver auf Sendung geschaltet. Die Dauersendung lässt sich durch Einsatz des Paddles oder über diesen (und andere) Protokoll-Befehle deaktivieren.

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

## 6.10. Winkey Modebyte

Computer → Keyer: 0x0E 0xXX

Computer ← Keyer: —

Entsprechend dem übergebenen Argument werden folgende Betriebsparameter ausgewählt:

- 0x0E 0bxxxx0xxx (0x00) Paddle Swap Off
- 0x0E 0bxxxx1xxx (0x08) Paddle Swap On
- 0x0E 0bxx11xxxx (0x30) Key: Bug
- 0x0E 0bxx01xxxx (0x10) Key: Iambic A
- 0x0E 0bxx10xxxx (0x20) Key: Ultimatic
- 0x0E 0bxx00xxxx (0x00) Key: Iambic B

Das Byte muss immer die Bits für beide Einstellungen enthalten.

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

## 6.11. Winkey Modebyte (2)

Computer → Keyer: 0x0F 0xXX 0xYY 0xZZ ...

Computer ← Keyer: —

Das erste Byte wird wie in 6.10 beschrieben ausgewertet. Das zweite Byte wird als Zeichengeschwindigkeit in WpM ausgewertet. Bis zu 13 weitere Bytes werden ignoriert.

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

## 6.12. Status abfragen

Computer → Keyer: 0x15

Computer ← Keyer: 0xYY

Der aktuelle Zustand des Keyers wird zurückgegeben:

- Keyer sendet Winkey Puffer → 0xC4
- Keyer sendet nicht aus Winkey Puffer → 0xC0

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

## 6.13. PTT

Computer → Keyer: 0x18 0xXX

Computer ← Keyer: —

Entsprechend dem übergebenen Argument wird die PTT-Leitung des aktuell gewählten TRX geschaltet.

- 0x0B 0x00 PTT Off
- 0x0B 0x01 PTT On

Ist das Byte ungleich 0x00, so wird die PTT aktiviert. Die aktivierte PTT kann durch zwischenzeitliches Geben mit dem Paddle auch wieder automatisch abfallen.

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

## 6.14. Betriebszeichen erzeugen

Computer → Keyer: 0x1B

Computer ← Keyer: —

Zwischen den beiden auf diesen Befehl folgenden Buchstaben wird *kein* Zeichenabstand gegeben.

Beispiel: Geben des Betriebszeichens <AS>:

Computer → Keyer: 0x1B 0x61 0x73

Der Keyer gibt das Betriebszeichen AS ohne Zeichenabstand.

Für folgende Betriebszeichen existiert außerdem ein Direktbefehl:

- 0x28 <KN>      HEX-Code entspricht ASCII-Zeichen „(“
- 0x29 <KK>      HEX-Code entspricht ASCII-Zeichen „)“
- 0x2B <AR>      HEX-Code entspricht ASCII-Zeichen „+“
- 0x3C <AR>      HEX-Code entspricht ASCII-Zeichen „<“
- 0x3E <SK>      HEX-Code entspricht ASCII-Zeichen „>“
- 0x40 <AC>      HEX-Code entspricht ASCII-Zeichen „@“

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

## 6.15. Relais

Computer → Keyer: `0x7F 0xXX 0xYY`

Computer ← Keyer: —

Entsprechend dem übergebenen Argument werden die zehn Relais des Bausatzes geschaltet. Nachfolgend einige Beispiele:

- `0x7F 0x00 0x00` Alle Relais deaktiviert
- `0x7F 0x00 0x01` Relais 1 aktiv, Rest deaktiviert
- `0x7F 0x00 0x02` Relais 2 aktiv, Rest deaktiviert
- `0x7F 0x00 0x03` Relais 1+2 aktiv, Rest deaktiviert
- `0x7F 0x00 0xFF` Relais 1+2+3+4+5+6+7+8 aktiv, Rest deaktiviert
- `0x7F 0x01 0x00` Relais 9 aktiv, Rest deaktiviert
- `0x7F 0x03 0x00` Relais 9+10 aktiv, Rest deaktiviert
- `0x7F 0x03 0xFF` Alle Relais aktiv

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

Andere mögliche Eingaben (auch ohne aktiviertem Host-Modus möglich):

- `0x7F 0x7F` Erkennung des DL6ER-Keyers

Antwort: `0x7F`

## 6.16. ASCII Zeichen

Computer → Keyer: `0xXX`

Computer ← Keyer: Gerade gegebenes Zeichen drei Ditlängen vor seiner Vollendung

Das übermittelte Zeichen `0xXX` wird dem Sendepuffer hinzugefügt insofern diesem Zeichen ein gültiges CW-Zeichen entspricht.

Der Befehl wird nur bei aktiviertem Host-Modus ausgewertet.

## 6.17. Getestete Software

Programm	Einstellungen ändern WpM, Mode, etc.	Zeichen/Speicher senden	Speicher programmieren	Relais steuern
UcxLog v7.46 Windows® XP	✓	✓	✓	✓
UcxLog v7.46 Ubuntu Linux	✓	✓	✓	✓
UcxLog v7.45 Windows® XP	✓	✓	✓	✗
CQRLog 1.5.2 Ubuntu Linux	✓	✓	✓	✗
RCKLog v3.27 Windows® XP	✓	✓	✓	✗
N1MM Logger 14.9.0 Windows® XP	✗	✓	✓	✗
SD by EI5DI v17.51 Windows® XP	?	✓	✓	✗

# ANHANG

## A. Bausatzbeschreibung

## A.1. Benötigte Bauteile für den DL6ER-Keyer

### A.1.1. Small (ohne zehn steuerbare Relais)



## DL6ER-Keyer :: Small v2.0

<https://secure.reichelt.de/index.html?&ACTION=20&LA=5010&AWKID=1023818&PROVID=2084>

reichelt elektronik GmbH & Co. KG  
Elektronikring 1  
26452 Sande

Tel.: +49 (0)4422 955-333  
Fax.: +49 (0)4422 955-111  
www.reichelt.de  
Datum: 05.02.2015

### Warenkorb

Referenznummer: **1712690687**

Artikel-Nr.	Bezeichnung	WG	Menge	Einzelpreis incl. MwSt.	Gesamtpreis incl. MwSt.
MPE 094-1-016	Buchsenleisten 2,54 mm, 1X16, gerade	1	2	0,38 €	0,76 €
GS 8P	IC-Sockel, 8-polig, superflach, gedreht, vergold.	1	1	0,17 €	0,17 €
75H 1,0K	Cermet-Miniatur-Regler, liegend, 6mm 1,0 K-Ohm	1	1	0,88 €	0,88 €
CBP 4	Cinchbuchse, Einbauprint, 4-fach	1	1	1,05 €	1,05 €
LUM 1503-07	Klinkeneinbaubuchse, 3,5mm Stereo, gew., PCB	1	1	0,59 €	0,59 €
BC 337-16	Transistor NPN TO-92 45V 0,8A 0,625W	1	1	0,050 €	0,05 €
TASTER 3301D	Kurzhubtaster 6x6mm, Höhe: 12,5mm, 12V, vert.	1	1	0,11 €	0,11 €
METALL 4,70K	Metallschichtwiderstand 4,70 K-Ohm	1	1	0,082 €	0,08 €
METALL 220	Metallschichtwiderstand 220 Ohm	1	2	0,082 €	0,16 €
METALL 150	Metallschichtwiderstand 150 Ohm	1	3	0,082 €	0,25 €
LTV 817	OPTOKOPPLER	1	2	0,19 €	0,38 €
LED 3MM GN	LED, 3mm, Low Cost, grün	1	2	0,060 €	0,12 €
LED 3MM RT	LED, 3mm, Low Cost, rot	1	2	0,060 €	0,12 €
SIL 7271-D 5V	Reedrelais, 5V, 1 Schließer mit Diode, 1A	1	2	1,50 €	3,00 €
SL 1X36G 2,54	36pol. Stiftleiste, gerade, RM 2,54	1	1	0,15 €	0,15 €
X7R-2,5 10N	Vielschicht-Keramikkondensator 10N, 10%	1	2	0,030 €	0,06 €
AKL 059-02	Anschlussklemme 2-polig, RM3,5	1	1	0,24 €	0,24 €

Warenwert 8,17 €

Zuzüglich Versandkosten innerhalb Deutschlands für ca. 0,0 kg bei Zahlung per Nachnahme 5,60 €

Zuzüglich Versandkosten innerhalb Deutschlands für ca. 0,0 kg 5,60 €

Alle Preise in Euro inklusive gesetzl. Mehrwertsteuer.

Mit freundlichen Grüßen  
Ihr reichelt elektronik Team

Volksbank Jever e. G. **BLZ 282 622 54** IBAN: DE10 2826 2254 5044 5500 00

Sparkasse **Kto. 50 44 55 00 00**  
Wilhelmshaven **BLZ 282 501 10** IBAN: DE36 2825 0110 0003 8161 62

Handelsregister: **Kto. 00 03 81 61 62**  
HRA 200654 Oldenburg USt-IdNr.: DE 259817039

Persönlich haftende Gesellschafterin: reichelt elektronik Verwaltungs-GmbH  
▪ Sitz Sande ▪ Amtsgericht Oldenburg, HRB 130281 ▪ Geschäftsführer: Ulf Timmermann, Hermann Herzog, Friedel Grunwald

**SWIFT: BIC:**  
GENODEF1JEV  
**SWIFT: BIC:**  
BRLADE21WHV  
**WEEE-Nr.:** DE 58715331





## A.1.2. Big (inkl. zehn steuerbaren Relais)



## DL6ER-Keyer :: Big v2.0

<https://secure.reichelt.de/index.html?&ACTION=20&LA=5010&AWKID=1023819&PROVID=2084>

reichelt elektronik GmbH & Co. KG  
Elektronikring 1  
26452 Sande

Tel.: +49 (0)4422 955-333  
Fax.: +49 (0)4422 955-111  
www.reichelt.de  
Datum: 05.02.2015

## Warenkorb

Referenznummer: **1715206379**

Artikel-Nr.	Bezeichnung	WG	Menge	Einzelpreis incl. MwSt.	Gesamtpreis incl. MwSt.
D-SUB BU 15US	D-SUB-Buchse, 15-polig, gewinkelt, RM 7,2	1	1	0,43 €	0,43 €
MPE 094-1-016	Buchsenleisten 2,54 mm, 1X16, gerade	1	2	0,38 €	0,76 €
GS 8P	IC-Sockel, 8-polig, superflach, gedreht, vergold.	1	1	0,17 €	0,17 €
75H 1,0K	Cermet-Miniatur-Regler, liegend, 6mm 1,0 K-Ohm	1	1	0,88 €	0,88 €
CBP 4	Cinchbuchse, Einbauprint, 4-fach	1	1	1,05 €	1,05 €
LUM 1503-07	Klinkeneinbaubuchse, 3,5mm Stereo, gew., PCB	1	1	0,59 €	0,59 €
BC 337-16	Transistor NPN TO-92 45V 0,8A 0,625W	1	1	0,050 €	0,05 €
TASTER 3301D	Kurzhubtaster 6x6mm, Höhe: 12,5mm, 12V, vert.	1	1	0,11 €	0,11 €
METALL 4,70K	Metallschichtwiderstand 4,70 K-Ohm	1	1	0,082 €	0,08 €
METALL 220	Metallschichtwiderstand 220 Ohm	1	2	0,082 €	0,16 €
METALL 150	Metallschichtwiderstand 150 Ohm	1	3	0,082 €	0,25 €
LTV 817	OPTOKOPPLER	1	2	0,19 €	0,38 €
LED 3MM GN	LED, 3mm, Low Cost, grün	1	2	0,060 €	0,12 €
LED 3MM RT	LED, 3mm, Low Cost, rot	1	2	0,060 €	0,12 €
SIL 7271-D 5V	Reedrelais, 5V, 1 Schließer mit Diode, 1A	1	12	1,50 €	18,00 €
SL 1X36G 2,54	36pol. Stiftleiste, gerade, RM 2,54	1	1	0,15 €	0,15 €
X7R-2,5 10N	Vielschicht-Keramikkondensator 10N, 10%	1	2	0,030 €	0,06 €
AKL 059-02	Anschlussklemme 2-polig, RM3,5	1	1	0,24 €	0,24 €

Warenwert 23,60 €

Zuzüglich Versandkosten innerhalb Deutschlands für ca. 0,0 kg bei Zahlung per Nachnahme 5,60 €  
Zuzüglich Versandkosten innerhalb Deutschlands für ca. 0,0 kg 5,60 €

Alle Preise in Euro inklusive gesetzl. Mehrwertsteuer.

Mit freundlichen Grüßen  
Ihr reichelt elektronik Team

Volksbank Jever e. G. **BLZ 282 622 54** IBAN: DE10 2826 2254 5044 5500 00

Sparkasse **BLZ 282 501 10** IBAN: DE36 2825 0110 0003 8161 62

Wilhelmshaven **Kto. 00 03 81 61 62**

Handelsregister: **HRA 200654 Oldenburg** USt-IdNr.: DE 259817039

Persönlich haftende Gesellschafterin: reichelt elektronik Verwaltungs-GmbH

• Sitz Sande • Amtsgericht Oldenburg, HRB 130281 • Geschäftsführer: Ulf Timmermann, Hermann Herzog, Friedel Grunwald

**SWIFT: BIC:**  
**GENODEF1JEV**  
**SWIFT: BIC:**  
**BRLADE21WHV**  
**WEEE-Nr.: DE 58715331**



Qualitätsmanagement  
zertifiziert nach ISO 9001

## A.1.3. Abgesetztes Bedienteil



## DL6ER-Keyer :: abgesetztes Bedienteil

<https://secure.reichelt.de/index.html?&ACTION=20&LA=5010&AWKID=1023799&PROVID=2084>

reichelt elektronik GmbH & Co. KG  
Elektronikring 1  
26452 Sande

Tel.: +49 (0)4422 955-333  
Fax.: +49 (0)4422 955-111  
www.reichelt.de  
Datum: 05.02.2015

## Warenkorb

Referenznummer: **1716445226**

Artikel-Nr.	Bezeichnung	WG	Menge	Einzelpreis incl. MwSt.	Gesamtpreis incl. MwSt.
AK MMK 710-DIGI	CD-ROM Audiokabel, SPDIF, 2-polig	1	1	0,95 €	0,95 €
METALL 2,20K	Metallschichtwiderstand 2,20 K-Ohm	1	9	0,082 €	0,74 €
TASTER 3301B	Kurzhubtaster 6x6mm, Höhe: 9,5mm, 12V, vertikal	1	1	0,13 €	0,13 €
TASTER 3301	Kurzhubtaster 6x6mm, Höhe: 4,3mm, 12V, vertikal	1	9	0,11 €	0,99 €

Warenwert	2,81 €
Zuzüglich Versandkosten innerhalb Deutschlands für ca. 0,0 kg bei Zahlung per Nachnahme	=====
Zuzüglich Versandkosten innerhalb Deutschlands für ca. 0,0 kg	5,60 €
	5,60 €

Alle Preise in Euro inklusive gesetzl. Mehrwertsteuer.

Mit freundlichen Grüßen  
Ihr reichelt elektronik Team

Volksbank Jever e. G. **BLZ 282 622 54** IBAN: DE10 2826 2254 5044 5500 00 **SWIFT: BIC: GENODEF1JEV**  
**Kto. 50 44 55 00 00**  
 Sparkasse **BLZ 282 501 10** IBAN: DE36 2825 0110 0003 8161 62 **SWIFT: BIC: BRLADE21WHV**  
 Wilhelmshaven **Kto. 00 03 81 61 62** **WEEE-Nr.: DE 58715331**  
 Handelsregister: **HRA 200654 Oldenburg** USt-IdNr.: DE 259817039  
 Persönlich haftende Gesellschafterin: reichelt elektronik Verwaltungs-GmbH  
 ▪ Sitz Sande ▪ Amtsgericht Oldenburg, HRB 130281 ▪ Geschäftsführer: Ulf Timmermann, Hermann Herzog, Friedel Grunwald



## A.1.4. Relais Kontrollboard



## DL6ER-Keyer :: Relaiskontrollboard

<https://secure.reichelt.de/index.html?&ACTION=20&LA=5010&AWKID=1023799&PROVID=2084>

reichelt elektronik GmbH & Co. KG  
Elektronikring 1  
26452 Sande

Tel.: +49 (0)4422 955-333  
Fax.: +49 (0)4422 955-111  
www.reichelt.de  
Datum: 05.02.2015

## Warenkorb

Referenznummer: **1717592512**

Artikel-Nr.	Bezeichnung	WG	Menge	Einzelpreis incl. MwSt.	Gesamtpreis incl. MwSt.
GS 20P	IC-Sockel, 20-polig, superflach, gedreht, vergold.	1	1	0,26 €	0,26 €
SL 1X36G 2,54	36pol. Stiftleiste, gerade, RM 2,54	1	1	0,15 €	0,15 €
METALL 100	Metallschichtwiderstand 100 Ohm	1	1	0,082 €	0,08 €
SIL 10-9 100	Widerstands-Netzwerk, 9Wid./10Pins, 100 Ohm	1	1	0,10 €	0,10 €
GBG 1000	Bargraph-Anzeige, 10 Elemente, grün	1	1	0,88 €	0,88 €
D-SUB BU 15US	D-SUB-Buchse, 15-polig, gewinkelt, RM 7,2	1	1	0,43 €	0,43 €
D-SUB ST 15US	D-SUB-Stecker, 15-polig, gewinkelt, RM 7,2	1	1	0,50 €	0,50 €

Warenwert 2,40 €

Zuzüglich Versandkosten innerhalb Deutschlands für ca. 0,0 kg bei Zahlung per Nachnahme 5,60 €  
Zuzüglich Versandkosten innerhalb Deutschlands für ca. 0,0 kg 5,60 €

Alle Preise in Euro inklusive gesetzl. Mehrwertsteuer.

Mit freundlichen Grüßen  
Ihr reichelt elektronik Team

Volksbank Jever e. G. **BLZ 282 622 54** IBAN: DE10 2826 2254 5044 5500 00 **SWIFT: BIC: GENODEF1JEV**  
**Kto. 50 44 55 00 00**  
 Sparkasse **BLZ 282 501 10** IBAN: DE36 2825 0110 0003 8161 62 **SWIFT: BIC: BRLADE21WHV**  
 Wilhelmshaven **Kto. 00 03 81 61 62** **WEEE-Nr.: DE 58715331**  
 Handelsregister: **HRA 200654 Oldenburg** USt-IdNr.: DE 259817039  
 Persönlich haftende Gesellschafterin: reichelt elektronik Verwaltungs-GmbH  
 ▪ Sitz Sande ▪ Amtsgericht Oldenburg, HRB 130281 ▪ Geschäftsführer: Ulf Timmermann, Hermann Herzog, Friedel Grunwald



## B. Betriebsparameter und Standardeinstellungen

Parameter	Kurzbeschreibung	Voreinstellung
<code>sidetoneActive</code>	Seitenton aktiv?	TRUE (Ja)
<code>key_output</code>	KEY/PTT Ausgang 1 oder 2	1
<code>toggleDitDah</code>	Dit/Dah Eingang vertauscht	FALSE (Nein)
<code>useKeyboard</code>	Tastaturfunktion	1 (nur Paddle)
<code>TRXMute</code>	TRX-Leitungen stumm geschaltet?	FALSE (Nein)
<code>TRXActive</code>	KEY Aktiv Zustand	HIGH
<code>keyboardAutospace</code>	Automatisches Leerzeichen?	TRUE (Ja)
<code>WpM</code>	Zeichengeschwindigkeit in WpM	15
<code>sidetoneFreq</code>	Frequenz des Seitentons	700 Hz
<code>mode</code>	Betriebsart des Paddles	IAMBIC_A
<code>pttLeadInTime</code>	PTT Vorlaufzeit	0
<code>pttTailTime</code>	PTT Nachlaufzeit (WpM unabh.)	0
<code>pttHangTime</code>	PTT Nachlaufzeit (WpM abh.)	3
<code>beaconInterval</code>	Schleifensendungsintervall	4
<code>Farnworth</code>	Verlängerte Pausen	0 (0%)
<code>charTolerance</code>	Zeichentoleranz	4
<code>LAYOUT</code>	Tastaturlayout <sup>1</sup>	DE (Deutsch)

Tabelle B.1.: Betriebsparameter des Keyers

<sup>1</sup>Nur durch Neuprogrammierung veränderbar