

CW-Trainer

Version 2
von DJ3YB und DJ1MHR



Der CW-Trainer ist zum Erlernen und Üben der Morsezeichen vorgesehen. Die Bedienung erfolgt im wesentlichen über eine AT-Tastatur. Die Eingaben werden auf dem 2x20 Zeichen Display angezeigt. Die Wiedergabe der CW-Zeichen geschieht über den eingebauten Kleinlautsprecher oder über einen anzuschließenden Kopfhörer. Die Lautstärke kann über ein Potentiometer eingestellt werden. Das Gerät verfügt über einen Ausgang zur TX-Tastung (aktiv Low) und zur Modulation eines Funkgerätes liefert es ein NF-Signal (300 mVss). Über drei LEDs wird der Status angezeigt. Es kann eine Morsetaste oder ein Paddle angeschlossen werden. Die über die Morsetaste (oder Paddle) eingegebenen Zeichen werden am Ausgang TX-Tastung und am Lautsprecher/Hörer wieder ausgegeben und von dem Gerät decodiert und auf dem Display angezeigt.

Zum Üben für unterwegs können die gespeicherten Lektionen auch über eine „Play-Taste“ aufgerufen und gestartet werden ohne dass eine Tastatur erforderlich ist.

Hier soll die Hard- und Software des Gerätes näher beschrieben werden. Eine Bedienungsanleitung ist in dem Dokument CW-TrainerBanl_V2.pdf zu finden.

Beschreibung der Schaltung:

Die Hardware des Gerätes ist sehr einfach, sie besteht im wesentlichen aus dem ATMEL Prozessor Atmega168 der die externen Komponenten wie Display, LEDs, Hörer und TX-Tastausgang direkt ansteuert. Die Eingabelemente wie CW-Taste oder Paddle, AT-Tastatur und Bedientaste werden durch den Prozessor mittels der Interrupt Eingänge INT0 und INT1 direkt abgefragt. Der Prozessor arbeitet mit einem 16 MHz Quarzoszillator.

Zur Stromversorgung dient die eingebaute 9V-Blockbatterie und es kann auch eine externe Stromversorgung angeschlossen werden. Die +5V für die Versorgung des Prozessors werden von dem Spannungsregler IC3 erzeugt. Die Stromaufnahme beträgt ca. 10mA + Strom für die Tastatur.

Beschreibung der Software:

Im Gegensatz zur Hardware des CW-Trainers ist dessen Software ziemlich komplex. Wegen des großen Umfangs musste auch ein Prozessor mit mindestens 16K Programmspeicher verwendet werden. Vorab einige Anmerkungen zur Programmstruktur:

Zu Beginn des Programmcodes sind einige Compilerdirektiven vorgesehen, mit Debug kann eine Testausgabe aktiviert werden, CW-Geber und CW-Decoder Software können unabhängig voneinander aktiviert werden und die Quarzfrequenz kann entsprechend dem eingesetzten Quarz umgeschaltet werden. Dann erfolgt die Definition aller im Programm verwendeten Konstanten und Variablen. Nach Initialisierung der Ports, des LCDs, der AT-Tastatur, der Timers 1 und nach Ausgabe des Logos beginnt der Ablauf mit dem Erstellen des Tabellenindex für die einzelnen im EEPROM gespeicherten Lektionen und dem Lesen einiger im EEPROM abgelegten Konstanten.

Nach dem Einschalten wird die aktuelle Spannung der Batterie angezeigt.

Mit dem Timer0 wird ein periodischer Interrupt im Raster von 10ms zu erzeugt. Dieser Interrupt dient beim CW-Geber zur Ausgabe der CW-Zeichen und beim CW-Decoder zur Geschwindigkeitsmessung der Eingabe.

Der Timer2 dient zur Erzeugung des CW-Tones und er wird zur Seed-Einstellung für die Random-Funktion zur zufälligen Zeichenausgabe benutzt.

Schließlich mündet der Ablauf in der Hauptschleife (Do ... Loop). Der genaue Ablauf der Software ist auch in den Flussdiagrammen im Anhang zu finden.

Codierung des CW-Zeichens im Programm

Für die Darstellung eines CW-Zeichens im Programm wird folgende Codierung, die pro Zeichen nur ein Byte belegt, benutzt:

Die meisten CW-Zeichen bestehen aus 1 bis 5 Symbolen (Punkte bzw. Striche). Mit den Bits 7...3 werden die Symbole dargestellt, ein Strich ist mit einer 1 und ein Punkt ist mit einer 0 codiert. Die Symbolcodierung beginnt dabei immer mit dem Bit 7 (MSB). Die Bits 2...0 geben die Anzahl der Symbole eines Zeichens an. Das t hat also den Code 10000001, das e den Code 00000001 und das o den Code 11100011.

Darüber hinaus gibt es noch die Sonderfälle, Zeichen mit mehr als 5 Symbolen, die ebenfalls berücksichtigt werden. Bei Zeichen mit 6 Symbolen die mit einem Strich enden, wird die 1 auf Bit 2 (Länge 110) als 6. Symbol = Strich ausgegeben. Bei Zeichen mit 6 Symbolen, die mit einem Punkt enden, werden die ersten 5 Symbole wie beschrieben und mit Länge 7 codiert. Bei Länge 7 werden

von dem Programm nur die ersten 5 Symbole entsprechend dem Code ausgegeben und ein Punkt angehängt. Der Sonderfall IRRUNG wird mit Hex: 00000111 Dez: 7 codiert und es werden 8 Punkte ausgegeben. Auch andere Spezialzeichen (!, \$, &, @) sind im Code realisiert.

Das Programm unterstützt auch die Ausgabe von Doppelzeichen wie [ar], [ch], [eb], [ka], [ve] und [sk]. Hier werden die Einzelzeichen ohne Wortabstand ausgegeben.

CW-Geber:

Die AT-Tastatur ist am Port D.1 (INT1) angeschlossen und löst bei Betätigung, wenn das Programm auf CW-Decoder geschaltet ist, den Interrupt 1 aus. Damit wird der CW-Geber aktiviert und der Decoder deaktiviert. In der While-Schleife des CW-Gebers wird zuerst die Tastatur abgefragt. Dies geschieht mittels der in Bascom enthaltenen Library „Getatkbd()“ Diese Routine interpretiert den Scan-Code der auf der Tastatur gedrückten Taste und liefert das in der Übersetzungstabelle „Keydata“ für diesen Scan-Code definierte CW-Zeichen als „Key“. Die Scan-Codes der AT-Tastatur sind bei den Funktionstasten F5 und F7 anders codiert als die übrigen Funktionstasten. Um den uneingeschränkten Zugriff auf alle Funktionstasten auch im Shift-Mode zu erhalten, werden diese Tasten Codes von der Software abgeändert. Dazu ist der Zugriff auf das interne Register R6 erforderlich, welches von der Getatkbd() Routine benutzt wird. Auf die weiteren Details dieser Umkodierung soll hier verzichtet werden.

In einer Select Case Verzweigung werden dann entsprechend der gedrückten Taste verschiedene Aktionen ausgeführt:

Steuerfunktionen

Die Tasten des Ziffernblocks liefern die Werte KP_0 = 40 bis KP_9 = 49. In der Subroutine „CWtempo“ wird mit den Ziffern 0 bis 5 die CW-Geschwindigkeit und mit den Ziffern 6 bis 9 der Zeichenabstand eingestellt. Die veränderten Werte werden dann auf dem Display in der 2. Zeile ausgegeben.

Wenn eine der Steuertasten gedrückt wurde, wird die Steuerfunktion, entsprechend der folgenden Tabelle ausgeführt:

Auf-Pfeil:	CW-Tonfrequenz um 100 Hz erhöhen
Ab-Pfeil:	CW-Tonfrequenz um 100 Hz reduzieren
Links-Pfeil:	Zeichenausgabe auf LCD abschalten
Rechts-Pfeil:	Zeichenausgabe auf LCD einschalten
Einf:	Speichern des Tastaturpuffer Inhaltes im EEPROM
Entf:	Lesen des Tastaturpuffer Inhaltes aus dem EEPROM
Pos1:	Wiederholen des Tastaturpuffer Inhaltes
Ende:	Abbruch der Ausgabe aus den Tastaturpuffer
Rück-Taste:	Löscht letztes Zeichen im Tastaturpuffer
ESC:	Rücksetzen bzw. Löschen des Tastaturpuffers
Shift:	Übliche Shift Funktion zur Doppelbelegung der Tasten
Bild Auf:	Einschalten des Editier Modus für F1 bis F12 und SF1 bis SF12
Bild Ab:	Beenden des Editier Modus und Speichern der Änderungen im EEPROM
TAB:	Einschalten der Lektionenausgabe ab F1 bis Fx bzw. SF1 bis SFx
Shift Lock:	Abschalten der Lektionenausgabe ab F1 bzw SF1, Teil-Lektionen
Rechts Strg:	Einschalten der Random Ausgabe der Lektionenspeicher in 5er Gruppen
Links Strg:	Abschalten der Random Ausgabe der Lektionenspeicher
ALT:	Einschalten der Dauerausgabe der Lektionenspeicher
Alt GR:	Abschalten der Dauerausgabe der Lektionenspeicher
Zahlenblock +:	Einschalten des Abfrage Modes
Zahlenblock -:	Abschalten des Abfrage Modes
Space:	Abbruch des Abfrage Mmode
Taste ^:	Aktivierung der CW-Eingabe mit Taste
Taste °:	Aktivierung der CW-Eingabe mit Paddle
Druck:	Ausgabe der internen Programm Parameter auf dem LCD Display
NUM:	Ein-/Ausschalten des Kalibrier-Modes für die Batteriespannungsanzeige
Links-Pfeil:	Bei eingeschaltetem Kalibrier-Mode, Reduzieren der angezeigten Spannung
Rechts-Pfeil:	Bei eingeschaltetem Kalibrier-Mode, der Erhöhen angezeigten Spannung

Ausgabe der Lektionen

Die Funktionstasten F1 bis F12 bzw. Shift-F1 bis Shift-F12 starten die Ausgabe der Zeichen von einer der insgesamt 2 x 12 Lektionen, die im EEPROM gespeichert sind. Wenn vorher mit der TAB - Taste die Gesamtausgabe aktiviert wurde (ET=1) wird die Ausgabe immer mit Lektion F1 bzw. Shift-F1 begonnen und bis zum Ende der ausgewählten Lektion fortgesetzt.

Ausgabe der am Keyboard eingegebenen Zeichen

Wenn keine der oben genannte Funktionen aufgerufen wurde, wird das Stop-Flag für den Abfrage Mode geprüft. Wenn es gesetzt ist, wird in den Abfrage Mode, siehe unten, verzweigt. In allen anderen Fällen wird das eingegebene Zeichen in den Ausgabepuffer geschrieben und auf dem Display angezeigt. Alle Morsezeichen, die direkt über die Tastatur eingegeben wurden und auch die aus dem EEPROM aufgerufenen Zeichen der Lektionen, werden in der 1. Zeile am Display ausgegeben. Wenn die Zeile voll ist, wird der Text nach links verschoben, sodass immer die letzten 20 Zeichen am Display sichtbar sind.

Die Ausgabe der CW-Zeichen über den Hörer bzw. den TX-Tastenausgang erfolgt dann zeitlich unabhängig von der Eingabe in der Interrupt Routine.

Schließlich wird im Falle L=2 aus dem momentanen Stand des Timer2 ein neuer Seed für die Randomfunktion eingetragen.

Hier ist dann die While Schleife des CW-Geber Betriebs zu Ende und der Ablauf beginnt von vorne.

Editier Mode

Wenn vor dem Betätigen einer Funktionstaste mit „Bild Auf“ der Edit-Mode eingeschaltet wurde, kann der Inhalt der betreffenden Lektionen editiert werden. Die einzelnen Lektionen können unterschiedlich lang sein. Die Summe der Zeichen aller Lektionen (inklusive aller Trennzeichen) ist auf die Länge des Tastaturpuffers (250) beschränkt. Wenn durch Verändern einer Lektion diese Länge überschritten wird, wird die letzte Lektion automatisch entsprechend gekürzt. Beim Abschalten des Edit Mode mit „Bild Ab“ werden die geänderten Zeichen in das EEPROM zurück geschrieben. Der Ablauf innerhalb des Edit Mode ist in einem separaten Flussdiagramm im Anhang dargestellt.

Abfrage Mode

Der Abfrage Mode wird mit der Taste „+“ im Ziffernblock eingeschaltet. Nach dem Aufruf der gewünschten Lektion mittels Funktionstaste wird zunächst nur 1 Zeichen ausgegeben und die grün leuchtende LED macht darauf aufmerksam, dass der „Schüler“ das nach dem Gehör dekodierte Zeichen über die Tastatur eingeben soll. Ist das eingegebene Zeichen falsch, wird ein Brummtone abgegeben und auf eine richtige Eingabe gewartet. Ist das eingegebene richtig wird nach einer kurzen Pause das nächste Zeichen hörbar.

Der Abfrage Mode kann jederzeit mit der „-“Taste des Ziffernblocks oder durch die Eingabe von Blank wieder ausgeschaltet werden. Die Subroutine des Abfrage Mode ist im Abschnitt Subroutinen beschrieben.

Interruptroutine für den CW-Geber:

Die Interruptroutine beginnt mit der Einstellung des Timers für den 10m Zyklus, dann wird entweder in den Programmteil für den Geber oder für den CW-Decoder verzweigt. Hier die Beschreibung der Abläufe bei CW-Geber:

Als erstes wird das Stop-Flag für den Abfrage Mode getestet. Ist das Stop-Flag gesetzt erfolgt keine weitere Aktion und die Interruptroutine wird beendet, d.h. Im Abfragemode wird das aus einer Lektion ausgesuchte Zeichen nicht ausgegeben.

Zu Beginn, oder wenn ein Zeichen vollständig ausgegeben wurde ist der Längenindikator L=0. Bei L=0 wird ein neues Zeichen für die Ausgabe geholt. Solange der Tastaturpufferzeiger ungleich der im Tastaturpuffer eingetragenen Zeichen ist, wird das nächste Zeichen aus dem Tastaturpuffer für die Ausgabe geholt. Wenn eine Funktionstaste gedrückt wurde, wird das nächste Zeichen aus der ausgewählten Lektionentabelle geholt. Dabei werden mit N 5-er Gruppen gebildet, d.h. nach jeweils 5 Zeichen wird ein Wortabstand eingefügt. Wenn die Random-Ausgabe aktiviert wurde, werden die Zeichen zufällig aus der Lektion ausgewählt. Wenn die zufällige Auswahl auf ein Ende-Zeichen trifft, wird die Ausgabe nicht beendet, sondern das Zeichen wird verworfen. Die kontinuierliche Ausgabe im Random Mode kann durch Drücken einer beliebigen Taste beendet werden. Ein Textende Zeichen (Trennzeichen zwischen aufeinander folgenden Lektionen) bei der Ausgabe mehrerer Lektionen, wird

durch einen Wortabstand ersetzt. Aus der Zeichencodierung wird nun die Anzahl der Symbole gelesen und L auf diesen Wert gesetzt. Bei L=6 oder L=7 erfolgt die Behandlung der Sonderfälle für IRRUNG bzw. für Zeichen mit 6 Symbolen, die mit einem Punkt enden. Nach diesem Eintrag eines Zeichens für die Ausgabe wird die Interruptroutine beendet.

Wird zu Beginn $L > 0$ festgestellt, wird zur Ausgabe des Zeichens verzweigt. Wenn kein Wortabstand vorliegt ($P=0$) und die Ausgabe des letzten Symbols abgeschlossen ist ($A=0$) werden der Ton, die TX-Tastung und die CW-LED eingeschaltet und mit $P=1$ der nächste Symbolabstand eingestellt. In Abhängigkeit vom MSB des Zeichens wird A (Dauer des Symbols, Punkt oder Strich) zugewiesen und die Interruptroutine beendet.

Beim nächsten Interrupt wird $P=1$ und $A>0$ vorgefunden. Jetzt wird lediglich A dekrementiert und die Interruptroutine wieder beendet. Die in A angegebene Zahl beschreibt die Anzahl der noch abzuwartenden 10ms Inkremente eines Symbols.

Wird beim Interrupt schließlich $A=0$ vorgefunden, werden der Ton, die TX-Tastung und die CW-LED wieder abgeschaltet und mit $Decr L$ die Restlänge der noch verbleibenden Symbole eingestellt. Ist Solange $L>0$ wird eine Silbenpause mit der Dauer eines Punktes eingefügt, das nächste Symbol auf die MSB-Position geschoben und mit $P=0$ die Ausgabe vorbereitet. Wird dagegen $L=0$ vorgefunden, wird im Falle einer Lektion Ausgabe das ausgegebene Zeichen auf dem Display ausgegeben, A mit dem Zeichenabstand belegt und mit Z das nächste Zeichen adressiert. Im Abfrage Mode wird das Stop-Flag gesetzt und die Abfrage-LED eingeschaltet.

Die Interruptroutine ist so in der Lage vollkommen selbständig die Ausgabe der vom Hauptprogramm in den Tastaturpuffer eingetragenen Zeichen, oder die Zeichen der zur Ausgabe anstehenden Lektionen auszugeben. Die Ausgabe Parameter können während einer Ausgabe mit den betreffenden Steuertasten verändert werden. Eine laufende Ausgabe wird dadurch nicht unterbrochen.

CW-Decoder:

Die CW-Taste oder ein Paddle ist am Port D.2 (INT0) angeschlossen und löst bei Betätigung den Interrupt 0 aus. Damit wird der CW-Decoder aktiviert und der CW-Geber deaktiviert. Als erstes werden Startwerte für Punkt- und Strichdauer vorgegeben mit der die Dekodierung begonnen werden soll. Diese Werte werden dann entsprechend der tatsächlich gemessenen Geschwindigkeiten adaptiert. Außerdem werden die benötigten Zählervariablen initialisiert.

Achtung: wenn durch Betätigung der CW-Taste oder Paddle das Gerät in den CW-Decoder-Mode geschaltet wurde, kann es durch Betätigung einer beliebigen Taste auf der Tastatur (oder durch Aus- und Einschalten) wieder in den CW-Geber-Mode umgeschaltet werden. Dieser erste Tastendruck wird nicht als Eingabezeichen oder als Steuerzeichen interpretiert.

Die Umschaltung in den Trainer-Mode kann auch über die Play-Taste erfolgen.

Tastatureingaben sind nur möglich, wenn auf dem Display in der zweiten Zeile „Trainer“ angezeigt wird.

Interruptroutine für den CW-Decoder:

Die Interruptroutine beginnt mit der Einstellung des Timers für den 10m Zyklus, dann wird entweder in den Programmteil für den CW-Geber oder für den CW-Decoder verzweigt. Hier die Beschreibung der Abläufe bei CW-Decoder:

Über den Timer0 Interrupt wird alle 10 ms der Zustand der CW-Taste abgefragt. Wird eine gedrückte Taste vorgefunden, wird mit $Taste = 0$ geprüft ob die Taste vor 10 ms noch nicht gedrückt war. Ist das der Fall, d.h. ein neues Symbol hat innerhalb der letzten 10 ms begonnen, wird das Zeichenausgabeflag $ChrOut$ gelöscht und der Pausenzähler auf Null gesetzt. Bei gedrückter Taste wird dann immer mit $Taste = 1$ gespeichert, dass die Taste gedrückt war und die aktuelle Symbolzeit wird inkrementiert.

Wird eine nicht gedrückte Taste vorgefunden, wird mit $Taste = 1$ geprüft ob die Taste vor 10 ms noch gedrückt war. Ist das der Fall, d.h. ein Symbol wurde innerhalb der letzten 10 ms beendet, wird der Symbolzähler inkrementiert. Bei nicht gedrückter Taste wird dann immer mit $Taste = 0$ gespeichert, dass die Taste nicht gedrückt war und der Pausenzähler bis max. 255 inkrementiert.

Am Ende der Interruptroutine wird das CW-Decoder Bearbeitungs Flag ($PrFlag$) gesetzt.

Zeichen Decodierung:

Die Schleife des CW-Decoders beginnt mit der Abfrage der CW-Taste zur direkten Steuerung (kein 10 ms Raster) der CW-Ausgänge Ton-Ausgang, TX-Tastung, inverse TX-Tastung und CW-LED.

Wenn dann das CW-Decoder Bearbeitungs Flag (PrFlag) nach einem Interrupt gesetzt ist, der aktuelle Pausenwert den Mittelwert überschreitet und auch noch keine Zeichen Dekodierung durchgeführt wurde, wird die Zeichen Dekodierung begonnen.

Aus den verschiedenen Symbolen eines Zeichens werden die kürzeste Dauer der Punkteinheit und die längste Dauer der Stricheinheit zugewiesen. Wenn die Stricheinheit größer als die doppelte Punkteinheit, d.h. es kamen Punkte und Striche im Zeichen vor, werden die Werte für Punkt und Strich adaptiert und daraus der neue Mittelwert aus Punkt- und Strichdauer (Einh15) berechnet. Wenn die Anzahl der vorgefundenen Symbole = 6 ist und die Dauer des 6. Symbols einem Punkt entspricht, wird die Symbolanzahl auf 7 gesetzt anderenfalls auf 2. Der Mittelwert aus Punkt- und Strichdauer wird dann zur Decodierung des Zeichens benutzt. Ein Symbol kürzer als der Mittelwert wird als Punkt und ein Symbol länger als der Mittelwert wird als Strich interpretiert. Schließlich wird zum Zeichencode noch die Anzahl der Symbole (S) addiert. Das so decodierte Zeichen wird dann auf dem Display ausgegeben.

Subroutinen:

GetIndex: Sucht die Textende Marken in der Lektionentabelle im EEPROM ab Adresse 256 und speichert deren Offset zu 256 in dem Array CwTabIdx für den späteren Zugriff.

Save: Speichert den Inhalt des Tastaturpuffers bei Beenden des EDIT Mode im EEPROM als neue Lektionentabelle ab und aktualisiert mit GetIndex das CwTabIdx Array.

Kontrolle: Bedient den Abfrage Mode. Blank (Key=255) beendet den Abfrage Mode. Anderenfalls wird das eingegebene Zeichen mit dem vorgespielten Zeichen verglichen. Bei Übereinstimmung wird das AM-Stop Flag auf Null gesetzt und die Abfrage LED abgeschaltet. Vom Hauptprogramm wird dann das nächste Zeichen zur Abfrage gestellt. Bei falscher Eingabe wird ein Fehlersignal im Hörer abgegeben und das AM-Stop Flag bleibt stehen bis eine richtige Eingabe erfolgt.

LDX: Lädt die internen Register 26 und 27 mit der Adresse \$100 des RAM zur Zwischenspeicherung von Register R6. Damit wird die Umkodierung der Funktionstasten F5 und F7 vorgenommen.

TestOut: Ausgabe Routine für die Debug Hilfe. Normalerweise deaktiviert.

LCDout: Holt den ASCII-Code des CW-Zeichens aus der Tabelle CW2ASCII und fügt den Wert dem LCD-Ausgabebetext an. Zeichen deren ASCII-Wert < 32 sind Doppelzeichen wie [ar], [ch], [eb], [ka], [ve] und [sk]. In diesem Fall werden beide Zeichen aus der Tabelle Doppelzeichen gelesen und dem LCD-Ausgabebetext angefügt. Wenn die Länge des Ausgabebetextes den Wert 20 überschreitet, wird der Ausgabebetext auf die letzten 20 Zeichen gekürzt. Wenn der Ausgabebetext nur aus einem Zeichen besteht wird vorher das LCD gelöscht. Die Routine wird dann mit LCDout2 fortgesetzt.

LCDout2: Der Ausgabebetext wird in Zeile 1 ausgegeben. Die Routine wird dann mit LCDout3 fortgesetzt.

LCDout3: In der 2. Zeile werden beim CW-Geber die aktuellen Werte für Geschwindigkeit und Zeichenabstand ausgegeben. Beim CW-Decoder werden die gemessenen Längen für Punkt und Strich ausgegeben.

LCDout4: Ausgabe des eingestellten CW-Gebers Taste oder Paddle.

ParOut: Gibt einige Programmparameter nach Betätigung der Taste „Druck“ auf dem Display aus.

ResBuff: Löscht den Tastaturpuffer und setzt die Zeiger auf Null. Die Routine wird dann, wenn Logo noch Null mit LCDout2 fortgesetzt.

CWtempo: Über den eingegebenen Steuercode werden aus der Tabelle CWtab die Werte für Geschwindigkeit oder Wortabstand geholt und abschließend im EEPROM als neue Startwerte gespeichert.

SavePaddle: Speichert die Einstellung Paddle oder Taste im EEPROM, wenn diese geändert wurde.

SetCwTon: Mit dem Aufwärtspfeil wird die CW-Ton bis zu einem Wert von max. 20 erhöht, mit dem Abwärtspfeil wird er bis zu einem Wert von min. 5 reduziert. Abhängig von der Quarzfrequenz wird

daraus ein Wert für die Einstellung des Waveformgenerators (OCR2) des Timer2 berechnet und eingetragen.

TastOn: Steuert die Tast-Ausgänge und den Mithörton in den On-Zustand.

TastOff: Steuert die Tast-Ausgänge und den Mithörton in den Off-Zustand.

ActKBD: Schaltet bei Interrupt 1 auf den CW-Geber um. Interrupt1 wird deaktiviert und Interrupt0 wird nach Löschen des Interrupt0-Flags aktiviert.

ActCwDec: Schaltet bei Interrupt 0 auf den CW-Decoder um. Interrupt0 wird deaktiviert und Interrupt1 wird nach Löschen des Interrupt1-Flags aktiviert. Durch setzen des Err-Flags wird die Polling-Routine von GetKBD() beendet.

ReadUB: Misst die Batteriespannung und gibt den gemessenen Wert in der 2. Zeile am Display aus.

T0Int: Timer0 Interrupt Routine wie oben beschrieben.

Writeeprdaten: Setzt das WRITE-Flag zum Speichern des Tastaturpuffers im EEPROM und fährt mit der Routine Readeprdaten fort.

Readeprdaten: Bei WrFlag = 1 werden die Daten aus dem Tastaturpuffer in das EEPROM geschrieben. Bei WrFlag = 0 werden die Daten aus dem EEPROM in den Tastaturpuffer gelesen.

TastInp: Aktiviert und führt durch das Einstell-Menue zur Bedienung des CW-Trainers mit der „Play-Taste“ (ohne Tastatur).

StatDspl: Statusanzeige bei Bedienung über die Bedientaste.

Tabellen im Flash-Speicher:

Keydata: Scan-Code Übersetzungstabelle. Sie enthält die codierten CW-Zeichen und Steuercodes.

CW2ASCII: Übersetzungstabelle CW-Code nach ASCII. ASCII-Code des CW-Zeichens zur Anzeige.

Doppelzeichen: ASCII-Wert des jeweils ersten und zweiten Zeichens eines Doppelzeichens.

CWtab: Werte für CWtempo und Wortabstand.

Tabellen im EEPROM:

Tastaturpuffer von Adresse 0 ... 253.

Startwerte für Geschwindigkeit und Wortabstand bei Adresse 254 und 255.

Lektionen von Adresse 256 ... 511.

Hinweis: Die Flussdiagramme aus dem Software-Stand 9.1, wurden noch nicht aktualisiert!