In der vorliegenden Version sind insgesamt 63 Buchstaben, Ziffern, Zeichen oder Phrasen als jeweils 1 Byte speicherbar. Darin sind bereits die erforderlichen Wortabstände enthalten. Der Text „CQ DE DL7UMO" würde also zunächst insgesamt 12 Speicherplätze beanspruchen.

Das Programm des PICs (s. Tabelle) realisiert alle 26 Buchstaben (ohne Umlaute!), die 10 Ziffern, + (ar), / und die sogenannten Phrasen DE, PSE K, CQ, DX, TEST, SK, BK, QRP, dazu den Wortabstand (00) sowie das Textende (63) in jeweils einem Speicherplatz. Letztere werden nur intern verarbeitet.

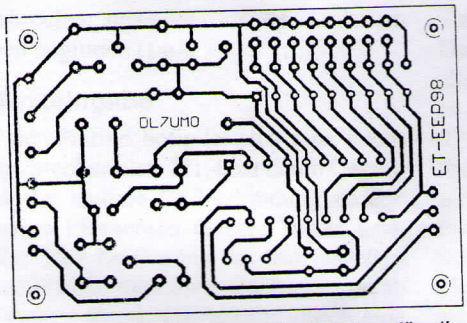

Bild 2: Leitungsführung der Platine für die elektronische Taste, einseitig

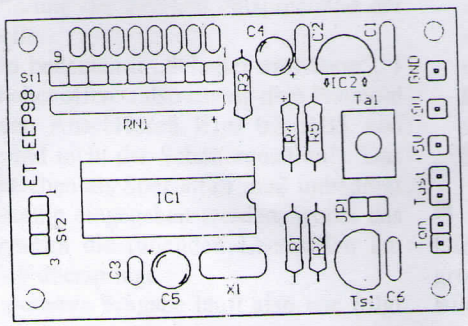

Bild 3: Bestückungsplan der Leiterplatte der elektronischen Taste

Textende oder kurz „Ende" dient dazu, die Ausgabe unmittelbar nach dem letzten Morsezeichen zu beenden. Das ist z.B. nützlich, wenn der Speicher neu beschrieben wurde. Sollten sich nämlich im Speicher noch weitere (alte) Zeichen befinden, würden sie sonst programmgemäß mit abgearbeitet. Das dauert nicht nur zu lange, sondern stört selbstverständlich durch unerwünschte Zeichen. Der Befehl „Ende" versetzt das Programm in den Ruhezustand, d.h., die evtl. verbliebenen Zeichen werden nicht mehr erreicht.

Die Tabelle enthält noch einige besondere Elemente: So dient „leer" (36h) dazu, den für den gewünschten Text nicht benötigten Rest des Rufzeichenspeichers zu füllen. „Wortabst." ist das Element für einen Wortabstand im Text.

Der Start der Ausgabe des internen Speichers erfolgt durch kurzes Betätigen der Taste „Text int" an St1/6. Je nach Stellung des Schalters an St1/5 kann man den Textspeicher des PIC entweder einmalig oder permanent auslesen. Hierbei gilt es allerdings zu beachten, daß sich auf den Speicherplätzen nach dem gewünschten Text bis zum Speicherende keine weitere Information befindet. Empfehlenswert ist also der Abschluß der Eingabe mit dem Element „Ende" (3Fh). Bei einmaliger Speicherausgabe wird damit auch das Auslesen beendet.

Rufzeichen als Sub-Block

Unabhängig davon, ob das Auslesen eines Textes mit oder ohne „Ende" sowie im Status „1x" oder „Dauer" erfolgt: Betätigen eines der Geberarme unterbricht den laufenden Text, und das vom Geber initiierte Morseelement beginnt – allerdings erst, nachdem noch das aus dem Speicher gelesene Morsezeichen komplettiert wurde.

Der EEPROM-Bereich des PIC (64 Bytes) wurde in drei Sektoren unterteilt. Die höchste Stelle (3Fh) ist für das aktuelle Gebetempo zuständig, acht Bytes (37h bis 3Eh) enthalten das Rufzeichen, und die restlichen 55 Bytes stehen für den Text des Nutzers zur Verfügung.

Mit den Rufzeichenbytes läßt sich sowohl ein fünfstelliges (z.B. DJ1AB) als auch eines mit angehängtem Suffix wie /a, /p, /m realisieren (z.B. DK2BCD/a). In der Kodetabelle ist dieses selbstprogrammierte Rufzeichen unter „Rfz" (2Dh) zu finden.

Es braucht dann später bei der (ihm übergeordneten) Textzusammenstellung nicht mehr jedesmal als entsprechende Folge von Einzelelementen zusammengestellt zu werden. Diese Variante des Selbstprogrammierens des Rufzeichens macht unabhängig von PIC-Programmiergeräten o. ä., und es dürfte deshalb sehr willkommen sein.

Vorprogrammierter CQ-Ruf

Nachdem das Rufzeichen im EEPROM verewigt wurde, kann man bereits einen vorgefertigten CQ-Ruf absetzen. Dazu enthält die Software die mit der CQ-Taste abrufbare Sequenz (alle Elementbezeichnungen im Hex-Format) „30 30 30 2E 2D 2D 30 30 30 2E 2D 2D 2A 2F 2B 2B", wobei das Rufzeichen unter „Rfz" (2Dh) auftaucht. In diesen 16 Speicherplätzen ist folgender Text enthalten: „CQ CQ CQ DE Rfz Rfz CQ CQ CQ DE Rfz Rfz + PSE K" plus 20 s Pause bis zum Neubeginn.

Zur Ausgabe sind also drei Tasten vorgesehen. Betätigen der Taste „Rfz" (St1/7) gibt einmalig das Rufzeichen aus. Die CQ-Taste (St1/8) aktiviert einen bereits fest programmierten CQ-Ruf, und das Auslesen der restlichen max. 55 Bytes erfolgt durch die Taste „Text int" an St1/6. Kriterium hierbei sind die Elemente der Kodetabelle.

Der Kniff mit den Phrasen

Da genügend (Programm-)Platz im PIC vorhanden war, wurden die in der Tabelle aufgeführten Phrasen bzw. Abkürzungen die gleich mehrere Zeichen ersetzen, mit in das Programm geschrieben. Dabei gehört zu allen in der Tabelle mit einem Stern gekennzeichneten Phrasen noch ein fester Wortabstand (insgesamt 7 GE). Auch das spart wieder Bytes im EEPROM.

Die Einsparung durch spezielle Phrasen verdeutlicht folgendes Beispiel: „CQ DX CQ DX CQ DX DE DL7UMO DL7UMO + PSE K". In der ausführlichen Variante sind das 42 Bytes, ohne komprimiertes Rufzeichen 23, mit Kompression dagegen nur noch 11 Bytes! Damit ist erst 1/5 des verfügbaren Textspeicherplatzes gefüllt.

Kodierungen der Buchstaben, Ziffern, Zeichen und Phrasen

Element	Nr. dez.	Nr. hex	Bit 6	5	4	3	2	1
Wortabst.	0	00	0	0	0	0	0	0
A	1	01	0	0	0	0	0	1
B	2	02	0	0	0	0	1	0
C	3	03	0	0	0	0	1	1
D	4	04	0	0	0	1	0	0
E	5	05	0	0	0	1	0	1
F	6	06	0	0	0	1	1	0
G	7	07	0	0	0	1	1	1
H	8	08	0	0	1	0	0	0
I	9	09	0	0	1	0	0	1
J	10	0A	0	0	1	0	1	0
K	11	0B	0	0	1	0	1	1
L	12	0C	0	0	1	1	0	0
M	13	0D	0	0	1	1	0	1
N	14	0E	0	0	1	1	1	0
O	15	0F	0	0	1	1	1	1
P	16	10	0	1	0	0	0	0
Q	17	11	0	1	0	0	0	1
R	18	12	0	1	0	0	1	0
S	19	13	0	1	0	0	1	1
T	20	14	0	1	0	1	0	0
U	21	15	0	1	0	1	0	1
V	22	16	0	1	0	1	1	0
W	23	17	0	1	0	1	1	1
X	24	18	0	1	1	0	0	0
Y	25	19	0	1	1	0	0	1
Z	26	1A	0	1	1	0	1	0
0	32	20	1	0	0	0	0	0
1	33	21	1	0	0	0	0	1
2	34	22	1	0	0	0	1	0
3	35	23	1	0	0	0	1	1
4	36	24	1	0	0	1	0	0
5	37	25	1	0	0	1	0	1
6	38	26	1	0	0	1	1	0
7	39	27	1	0	0	1	1	1
8	40	28	1	0	1	0	0	0
9	41	29	1	0	1	0	0	1
+ (ar)*	42	2A	1	0	1	0	1	0
10sP	43	2B	1	0	1	0	1	1
/	44	2C	1	0	1	1	0	0
Rfz*	45	2D	1	0	1	1	0	1
DE**	46	2E	1	0	1	1	1	0
PSE K	47	2F	1	0	1	1	1	1
CQ*	48	30	1	1	0	0	0	0
DX*	49	31	1	1	0	0	0	1
TEST*	50	32	1	1	0	0	1	0
BK	51	33	1	1	0	0	1	1
SK	52	34	1	1	0	1	0	0
QRP	53	35	1	1	0	1	0	1
leer	54	36	1	1	0	1	1	0
Ende	63	3F	1	1	1	1	1	1

* Phrase mit angehängtem Wortabstand
** Phrase mit angehängtem Wortabstand plus Rufzeichen im Bereich des EEPROMs kodiert