

	´	¨	ˇ	ˆ	˘	˙		˘	˝	ˇ	˘	˙	˚	˛	˜	˝	˙	
<b>A a</b>	á		ǎ	(â)				à	(ă)	(ǎ)	ą			ā				
<b>B b</b>																	(b)	
<b>C c</b>	ć					ç				č								
<b>D d</b>										ď		đ					(d)	
<b>E e</b>	é		ě	(ê)		(ę)		è	(è)	ě	é			ē				
<b>F f</b>																		
<b>G g</b>	(ǵ)		ǧ							ǧ		ǧ	(ǧ)					
<b>H h</b>					(h)	(h)	(h)			ħ							(h)	
<b>I i</b>	í		ï	(î)				ì	(ï)	(ï)	ı			ī				
<b>J j</b>										(j)								
<b>K k</b>	(k)						(k)			ķ		ķ					(k)	
<b>L l</b>	ĺ									ł		ł					(l)	ł
<b>M m</b>	(m)																	
<b>N n</b>	ń						(n)	ñ		ň		ņ					(n)	
<b>O o</b>	ó	õ	õ	(ô)				ò	(ö)	(ö)	ó	ơ		ō	(ø)			
<b>P p</b>	(p)																	
<b>Q q</b>																		
<b>R r</b>	ř			(r)					(ř)	ř		ŗ					(r)	
<b>S s f</b>	ś					ş				š		ş						
<b>T t</b>						(t)				ť		ţ					(t)	
<b>U u</b>	ú	ů	ű	(û)				ù	(ü)	(ű)	ú	ư		ū				
<b>V v</b>																		
<b>W w</b>	ŵ							Ẁ										
<b>X x</b>																		
<b>Y y</b>	ý							ÿ			ÿ			ÿ				
<b>Z z</b>	ź						(z)			ž							(z)	
<b>Σ</b>	<b>13</b>	<b>2</b>	<b>6</b>	<b>-</b>	<b>-</b>	<b>2</b>	<b>-</b>	<b>8</b>	<b>-</b>	<b>12</b>	<b>6</b>	<b>2</b>	<b>7</b>	<b>6</b>	<b>1</b>	<b>-</b>	<b>1</b>	

˘	˙	˚	-	˚	˚	/ /	˜	˜	˜	˚	˚	˚	˚	ˆ	ˆ	Σ
ą	ą	ą			ą	(ą)	(ą)	ą			ą			ą		12
	(b)	(b)	(b)										(b)			-
	ć						(c)							ć		5
	(d)	(d)	đ										(d)		(d)	3
ę	ę	ę					(e)	ę		(e)	ę			ę	(ę)	12
	(f)															-
	g		g				(g)							g		6
	(h)	h	h								h			h		5
ı	ı	ı	(i)					ı		(i)	ı			ı		11
			(j)											j		1
		(k)	(k)				(k)									2
		ł	(l)	(l)			ł		(l)						(l)	6
	(m)	(m)														-
	ń	(n)					(n)	ń							(n)	6
o	o	o					o	o			o			o		14
	(p)		(p)													-
			(q)				(q)									-
	(r)	(r)	(r)				(r)									3
	(s f)	(s)	(f)				(s f)							ŝ		5
	(t)	(t)	t				(t)				(t)				(t)	3
u		(u)			u			(u)		(u)	u	(u)		u	(u)	11
		(v)					(v)	(v)								-
	(w)	(w)			(w)						w			w		4
	x										x					2
	(y)	(y)	(y)		(y)			ÿ			ÿ			ÿ		7
	z	(z)	(z)											z		4
5	9	6	4	-	2	-	2	6	-	-	9	-	-	13	-	122



Außerdem gibt es noch so genannte „*Named Sequences*“, kombinierte Sequenzen, die nicht als gesonderte Unicode-Zeichen aufgenommen wurden:<sup>1</sup>

Litauisch	diverse
ą ą̃	à
ė ė̃ é é̃	ẽ ẽ̃
ì ì̃ í í̃	ì í
ĵ	
ĩ	
ĩ̃	
ñ	
	ñ̃
ř	
ú û ú̃ û̃	ù

## Entwicklung einer praktikablen Eingabemethode

Als Titel der Eingabemethode habe ich *Šibboleth* gewählt. Hintergrund ist eine Stelle aus dem Alten Testament, Buch Richter Kapitel 12 Vers 5ff. Dort heißt es:

»(...) Wenn nun einer von den Flüchtlingen Ephraims sprach: Lass mich hinübergehen!, so sprachen die Männer von Gilead zu ihm: Bist du ein Ephraimter? Wenn er dann antwortete: Nein!, ließen sie ihn sprechen: „Šibbolet“.

<sup>1</sup> <http://unicode.org/Public/UNIDATA/NamedSequences.txt> (11.02.2012)

Sprach er aber: „Sibbolet“, weil er's nicht richtig aussprechen konnte, dann ergriffen sie ihn und erschlugen ihn an den Furten des Jordans, sodass zu der Zeit von Ephraim fielen zweiundvierzigtausend.«

## Grundsätzliche Erwägungen

1. Bestehende Eingabemethoden funktionieren oft nur unter bestimmten Programmen (z. B. MS Word) oder nur für bestimmte Untermenüen (z. B. Latein-1). Es besteht daher Bedarf an einer universal einsetzbaren Eingabemethode für alle lateinischen Schriftzeichen moderner europäischer Sprachen (zumindest auf MS-Windows-Systemen, besser noch plattformübergreifend).
2. Eingabemethoden basierend auf dezimalen bzw. hexadezimalen *Zahlenwerten* sind nicht praktikabel, da die Anzahl der Werte per Definition der Anzahl der Schriftzeichen entspricht. Dem durchschnittlichen Nutzer ist es nicht zuzumuten, über 400 Zahlenwerte im Gedächtnis zu behalten oder stets eine entsprechende Liste griffbereit zu haben.
3. Die Ausführung aller notwendigen Diakritika als *Tottasten* würde die Einführung von mindestens 14 zusätzlichen Tastenbelegungen erfordern. Diese neuen Belegungen müssten auf Tastaturen physisch markiert werden. Langfristig müssten bestehende internationale Standards für Tastatur-Layouts entsprechend angepasst werden.
4. Die Ausführung von Diakritika als Tottasten war eine technische Notwendigkeit der mechanischen Schreibmaschine. Im Zeitalter der elektronischen Textverarbeitung ist dieses historisch bedingte Zugeständnis nicht mehr zwingend notwendig.

5. Einige Systeme verwenden voreingestellte „Escape-Kodierungen“ basierend auf den bestehenden Tastenbelegungen. Das Ä lässt sich beispielsweise mittels  $\backslash$  " A (T<sub>E</sub>X-Quelltext) oder **Strg**  $\uparrow$  : A (MS Word) erzeugen.
6. Der Nachteil von Tottasten und vorgestellten Escape-Kodierungen ist, dass sie sowohl dem händischen Schreibfluss als auch der Nomenklatur von Schriftzeichen widersprechen („A mit Trema“, nicht „Trema über A“).
7. Escape-Kodierungen außerhalb von Quelltext erfordern eine Kombinierfunktion (*Compose*). MS Word beispielsweise verwendet hierfür die **Strg**-Taste. Der Nachteil ist, dass der Nutzer bis zu drei Tasten gleichzeitig drücken muss, z. B. **Strg**  $\uparrow$  : für ein Trema.
8. Eine Alternative zu Tastenkombinationen sind so genannte *Hotstrings*, i. e. Zeichenketten, die Ereignisse auslösen. Die Zeichenkette A : beispielsweise würde als Hotstring automatisch durch Ä ersetzt. Das Problem dabei ist, dass die Ersetzung dann auch an Textstellen stattfinden würde, an denen tatsächlich ein Doppelpunkt gewünscht war. Die Zeichenkette muss also durch einen zusätzlichen Tastendruck „scharf gemacht“ werden. Diese Taste sollte eine Steuertaste sein oder hilfsweise ein Zeichen, das in Texten nicht oder nur selten vorkommt. Auf einer deutschen Tastatur empfiehlt sich die Raute #, auf den Tastaturen der meisten anderen Sprachen ebenfalls die Raute oder das Plus-Zeichen +.
9. Es gibt grundsätzlich zwei Möglichkeiten für die Reihenfolge: # A : und A : #. Berücksichtigt man die Tatsache, dass Buchstaben mehrere Diakritika tragen können (z. B. Ǟ), ist die zweite Variante A : -

zu bevorzugen, da dann das Ende der Escape-Kodierung eindeutig ist (während     sowohl Ä als auch Ä- bedeuten könnte).

10. Die Escape-Kodierungen sollten zwecks maximaler Kompatibilität und Interoperabilität ausschließlich aus *ASCII-Zeichen* bestehen (d. h. aus Zeichen, die in der überwiegenden Zahl der Zeichensätze durch den gleichen Zahlenwert kodiert werden).
11. Die Escape-Kodierungen sollten *intuitiv* sein, d. h. in ihrer Form an das jeweilige diakritische Zeichen erinnern.

Die intuitive Kombination wäre „Grundbuchstabe, Diakritikum-Ersatz, Zusammensetzen“. Sie würde für die meisten Buchstaben mit Diakritika sowie für Ligaturen und Digraphen funktionieren. In Einzelfällen bzw. bei einer möglichen Erweiterung des Systems würden jedoch diejenigen Diakritika ein Problem verursachen, die sowohl über-, als auch untergesetzt werden können. Es besteht also Bedarf an einer zweiten Funktionstaste, nämlich für das Untersetzen. Intuitiv bietet sich hierzu der Unterstrich  an. Probleme können eigentlich nur beim Vergeben von Dateinamen auftreten, wenn der Unterstrich benutzt wird, um ein Leerzeichen zu ersetzen.

1. Grundbuchstabe, Diakritikum-Ersatz, Zusammensetzen
2. Grundbuchstabe1, Grundbuchstabe 2, Zusammensetzen
3. Grundbuchstabe1, Grundbuchstabe 2, Ligatur-/Digraph-Signal, Zusammensetzen
4. Grundbuchstabe, Diakritikum-Ersatz, Untersetzen (für Diakritika, die vorwiegend übergesetzt werden)
5. Grundbuchstabe, Diakritikum-Ersatz, Überlagern (für - ~)

Besteht Bedarf an einer dritten Funktionstaste für Buchstaben, die durch das Diakritikum überlagert werden? Ich würde einfach das  benutzen.

## Konfliktfälle

Einige Konfliktfälle müssen jedoch trotzdem gelöst werden:

Die Kombination `[a][e][#]` kommt prinzipiell als Auslöser für ä, å, æ und ę in Frage. Ich habe mir folgende Kombinationen gewählt: `[a][:][#]` = ä, `[a][l][e][#]` bzw. `[ä][e][#]` = å, `[a][e][#]` = æ und `[e][&][#]` = ę. Diese Kombinationen würden entsprechende Auswirkungen auf die Erzeugung von œ/ô/ö sowie von û/ü haben.

Die Ligatur ij sowie die kroatischen Digraphen lj und nj, die alle weitaus häufiger zu erwarten ist als das j (j ohne Punkt, vermutlich nur in mittelalterlichen Texten). Hier sollte `[i][j][#]` = ij den Vorzug bekommen und das j via `[j][0][#]` erzeugt werden.

Ein weiterer Konfliktfall ist die Kombination `[d][z][#]`, die als Auslöser für die Buchstabenvariante ȝ sowie für den Digraphen dz in Frage kommt. Erschwerend kommt noch hinzu, dass sowohl ȝ als auch dz mit einem Háček (ž, dž) vorkommen können. Setzen wir `[d][z][#]` = ȝ, dann muss dz via `[d][z][j][#]` explizit als Digraph erzeugt werden, auch wenn dies die Kohärenz mit lj und nj stört. Daraus folgt `[d][z][j][<][#]` = ž und `[d][z][<][j][#]` = dž. Beide Lösungen sind mit fünf Tastendrücken vergleichsweise tipp-intensiv, aber vermutlich nicht weiter optimierbar. Ebenso steht `[a][e][#]` für æ und `[a][e][j][_][#]` / `[a][e][j]['][#]` für æ̃ / ǽ.

## Escape-Kodierungen mit Diakritika

Unter Berücksichtigung bestehender Escape-Kodierungen (z. B. aus  $\text{\TeX}$  und MS Word) habe ich folgende Zuordnungen gewählt:

Akut ´	Apostroph [ ]
Doppelakut ¨	Anführungszeichen [ ]
Breve ˘	öffnende Klammer [ (
umgekehrtes Breve ˇ	schließende Klammer [ )
breites Breve ˘	zweimal öffnende Klammer [ ( [ (
breites umgekehrtes Breve ˇ	zweimal schließende Klammer [ ) [ )
Breve unter ˘	schließende Klammer, Unterstrich [ ) [ ]
umgekehrtes Breve unter ˇ	öffnende Klammer, Unterstrich [ ( [ ]
Cedille ,	Semikolon [ ;
Deszender	Ziffer [ 4
Gravis `	umgekehrter Schrägstrich [ \
Doppelgravis ``	zweimal umgekehrter Schrägstrich [ \ [ \
Háček ˇ	Kleiner-als-Zeichen [ <
Háček unter ˇ	Kleiner-als-Zeichen, Unterstrich [ < [ ]
Haken ˆ	Fragezeichen [ ?
Haken über/unterhalb	(experimentell) [ § ] [kein ASCII-Zeichen]
Horn ˆ	Ziffer [ 9
Komma ,	Komma [ ,
Makron ¯	Bindestrich/Minus [ -
Makron in der Mitte	Bindestrich/Minus + x [ - [ x
Makron unter ¯	Bindestrich/Minus, Unterstrich [ - [ ]
doppeltes Makron unter __	zweimal Bindestrich/Minus, Unterstrich [ - [ - [ ]

Mittelpunkt $\cdot$	(siehe Punkt) [separater Mittelpunkt: $\boxed{\cdot}$ ]
Ogonek $\underset{\cdot}{\text{c}}$	&-Zeichen $\boxed{\&}$
Punkt $\cdot$	Punkt $\boxed{\cdot}$
Punkt unter $\underset{\cdot}{\text{c}}$ (alternativ)	Ausrufezeichen $\boxed{!}$
Querstrich -	Gleich-Zeichen $\boxed{=}$
Ring $\circ$	Sternchen $\boxed{*}$ / Grad-Zeichen $\boxed{\circ}$ [kein ASCII-Zeichen]
Schrägstrich $/$	Schrägstrich $\boxed{/}$
Tilde $\sim$	Tilde $\boxed{\sim}$
Tilde in der Mitte	Tilde + x $\boxed{\sim} \boxed{x}$
Tilde unten $\underset{\sim}{\text{c}}$	Tilde, Unterstrich $\boxed{\sim} \boxed{\_}$
breite Tilde $\widetilde{\text{c}}$	zweimal Tilde $\boxed{\sim} \boxed{\sim}$
Trema $\ddot{\text{c}}$	Doppelpunkt $\boxed{:}$
<i>topbar</i> ]	schließende eckige Klammer $\boxed{]}$
Zirkumflex $\hat{\text{c}}$	Größer-als-Zeichen $\boxed{>}$
Zirkumflex unter	Größer-als-Zeichen, Unterstrich $\boxed{>} \boxed{\_}$
mit langem Bein	vertikale Linie $\boxed{ }$
Buchstabe spiegelverkehrt	vertikale Linie $\boxed{ }$
alleinstehendes Diakritikum	$\boxed{\text{Leerz.}}$ + Diakritikum
kombinierendes Diakritikum	öffnende eckige Klammer $\boxed{[}$ + Diakritikum

Die Bedienung ist denkbar einfach: Den Grundbuchstaben eingeben, dann das Diakritikum / die Diakritika, Raute/Unterstrich-Taste drücken.

Beispiele: `n' #` = `ń`, `u" #` = `ú`, `g( #` = `ğ`, `e) #` = `ê`,  
`i) ) #` = `îâ`, `c; #` = `ç`, `n4 #` = `ñ`, `y\ #` = `ÿ`, `r\ \ #` = `ř`,  
`c< #` = `č`, `a? #` = `ą`, `z$ #` = `ž`, `o9 #` = `ó`, `t, #` = `ţ`,  
`u- #` = `û`, `o-x #` = `ø`, `e& #` = `ę`, `e. #` = `è`, `s! #` = `ş`,  
`z= #` = `z`, `u° #` = `ů`, `o/ #` = `ø`, `a~ #` = `ã`, `l~x #` = `ł`,  
`u~_ #` = `u`, `x: #` = `ẋ`, `b] #` = `ḅ`, `y> #` = `ÿ`, `d>_ #` = `ḍ`

Für mehrere Diakritika werden erst die Diakritika unter dem Grundbuchstaben und dann die Diakritika über dem Grundbuchstaben eingegeben (aufsteigend)<sup>2</sup>.

Beispiele: `e; ( #` = `ě`, `u: < #` = `ů`

## Escape-Kodierungen ohne Diakritika

Einige Buchstaben werden direkt (ohne Diakritika) eingegeben:

Ð	<code>D</code> (Eth)
İ	<code>I</code> (türkisches I)
κ	<code>k</code> (kra) – (Der Großbuchstabe ist K', d.h. K + einfaches Anführungszeichen oben.)
Ə	<code>Ä</code> (Schwa, zusätzlich) [kein ASCII-Zeichen]
f	<code>f</code> (Florin / Gulden)
Ql	<code>Q</code> (Gha)
Ƶ	<code>W</code> (Wynn)
Θ	<code>Ö</code> (zusätzlich) [kein ASCII-Zeichen]

<sup>2</sup> Ausnahme Vietnamesisch.

Q	Ⓚ Ⓥ (zusätzlich)
W	Ⓥ Ⓥ (zusätzlich)
X	Ⓚ Ⓢ (zusätzlich)
Å	Ⓐ Ⓐ (zusätzlich)
Æ	Ⓐ Ⓔ (AE-Ligatur)
Ñ	Ⓝ Ⓝ (zusätzlich)
Þ	Ⓣ Ⓜ (Thorn) / Þ Ⓣ Ⓜ Ⓜ Ⓜ (Thorn mit Querstrich, <i>that</i> )
ß	Ⓢ Ⓢ (Eszett)
IJ	Ⓜ Ⓜ (IJ-Ligatur)
ŋ	Ⓝ Ⓜ (Eng)
Œ	Ⓞ Ⓔ (OE-Ligatur)
Ʒ	Ⓣ Ⓢ (Ezh) / Ʒ Ⓣ Ⓢ Ⓜ Ⓜ
Θ	Ⓣ Ⓞ (griech. Theta)
Ĥ	Ⓜ Ⓥ (Hwair, zur Transkription des Buchstabens Θ aus der gotischen Schrift)
ŋ	Ⓝ Ⓜ (N mit verlängertem rechtem Bein)
ᚱ	Ⓨ Ⓜ (zur Transkription der altnordischen Rune <i>elhaz</i> ᚱ)
DZ	Ⓣ Ⓢ Ⓜ (DZ-Digraph) / DŽ Ⓣ Ⓢ Ⓜ Ⓜ (DŽ-Digraph)
LJ	Ⓦ Ⓜ (LJ-Digraph)
NJ	Ⓝ Ⓜ (NJ-Digraph)
8	Ⓞ Ⓤ (OU-Ligatur, in indianischen Sprachen)
F	Ⓦ Ⓜ (claudianischer <i>Sonus medius</i> )

Ɔ	ⓑ Ⓢ, ⓑ Ⓢ (claudianisches <i>Antisigma</i> )
Ꝛ	Ꝛ Ꝛ ( <i>I longa</i> ), Ꝛ Ꝛ
𐌆	𐌆 𐌆 (urtümliches M)
ȝ	ȝ 0 (j ohne Punkt)
f	ſ Ꝛ (langes s)
ȝ	ȝ ȝ (Yogh)
Ǽ	Ǽ e (zusätzlich - übergesetztes e in frühen Formen von ä) [kein ASCII-Zeichen]
Ǿ	Ǿ e (zusätzlich) [kein ASCII-Zeichen]
ǘ	ǘ e (zusätzlich) [kein ASCII-Zeichen]
‡	Ꝛ = = ( <i>alveolar click</i> )
!	! Ꝛ ( <i>retroflex click</i> )
ə	Ⓐ : Ꝛ (Schwa)
ll	ll Ꝛ ( <i>Middle-Welsh LL</i> )
aa	Ⓐ Ⓐ Ꝛ
ao	Ⓐ 0 Ꝛ
aj	Ⓐ u Ꝛ
av	Ⓐ v Ꝛ / Ꝛ Ⓐ v Ꝛ =
ay	Ⓐ y Ꝛ
oo	0 0 Ꝛ
w	v y Ꝛ

Für die übrigen Schriftzeichen habe ich folgende Zuordnungen gewählt:

‰	<input type="text" value="‰"/> (Promille)
@	<input type="text" value="a"/> <input type="text" value="t"/>
¤	<input type="text" value="i"/> <input type="text" value="w"/> <input type="text" value="z"/> (internationales Währungszeichen), <input type="text" value="i"/> <input type="text" value="c"/> <input type="text" value="s"/>
	<input type="text" value="Leerz."/> <input type="text" value=" "/>
0 <sup>a</sup> ...9 <sup>a</sup>	<input type="text" value="0"/> <input type="text" value="a"/> ... <input type="text" value="9"/> <input type="text" value="a"/>
0 <sup>o</sup> ...9 <sup>o</sup>	<input type="text" value="0"/> <input type="text" value="o"/> ... <input type="text" value="9"/> <input type="text" value="o"/>
¬	<input type="text" value="N"/> <input type="text" value="O"/> <input type="text" value="T"/>
°	<input type="text" value="g"/> <input type="text" value="r"/> <input type="text" value="a"/> <input type="text" value="d"/> , <input type="text" value="d"/> <input type="text" value="e"/> <input type="text" value="g"/>
¼	<input type="text" value="Leerz."/> <input type="text" value="1"/> <input type="text" value="/"/> <input type="text" value="4"/>
½	<input type="text" value="Leerz."/> <input type="text" value="1"/> <input type="text" value="/"/> <input type="text" value="2"/>
¾	<input type="text" value="Leerz."/> <input type="text" value="3"/> <input type="text" value="/"/> <input type="text" value="4"/>
⅓	<input type="text" value="Leerz."/> <input type="text" value="1"/> <input type="text" value="/"/> <input type="text" value="3"/>
⅔	<input type="text" value="Leerz."/> <input type="text" value="2"/> <input type="text" value="/"/> <input type="text" value="3"/>
⅕	<input type="text" value="Leerz."/> <input type="text" value="1"/> <input type="text" value="/"/> <input type="text" value="5"/>
⅖	<input type="text" value="Leerz."/> <input type="text" value="2"/> <input type="text" value="/"/> <input type="text" value="5"/>
⅜	<input type="text" value="Leerz."/> <input type="text" value="3"/> <input type="text" value="/"/> <input type="text" value="5"/>
⅘	<input type="text" value="Leerz."/> <input type="text" value="4"/> <input type="text" value="/"/> <input type="text" value="5"/>
⅙	<input type="text" value="Leerz."/> <input type="text" value="1"/> <input type="text" value="/"/> <input type="text" value="6"/>
⅚	<input type="text" value="Leerz."/> <input type="text" value="5"/> <input type="text" value="/"/> <input type="text" value="6"/>
⅛	<input type="text" value="Leerz."/> <input type="text" value="1"/> <input type="text" value="/"/> <input type="text" value="8"/>
⅜	<input type="text" value="Leerz."/> <input type="text" value="3"/> <input type="text" value="/"/> <input type="text" value="8"/>
⅝	<input type="text" value="Leerz."/> <input type="text" value="5"/> <input type="text" value="/"/> <input type="text" value="8"/>
⅞	<input type="text" value="Leerz."/> <input type="text" value="7"/> <input type="text" value="/"/> <input type="text" value="8"/>
1/	<input type="text" value="Leerz."/> <input type="text" value="1"/> <input type="text" value="/"/>
"	<input type="text" value="Leerz."/> <input type="text" value="'"/> <input type="text" value="'"/> <input type="text" value="Leerz."/> (Zoll)

?	<b>[?] [!]</b> (Interrobang)
7	<b>[&amp;]</b> (Tironisches et)
$\%_c$	<b>[a] [/ c]</b> ( <i>account of ...</i> )
$\%_s$	<b>[a] [/ s]</b> ( <i>addressed to the subject ...</i> )
$\%$	<b>[c] [/ o]</b> ( <i>care of ...</i> )
$\%_u$	<b>[c] [/ u]</b> ( <i>cada una</i> – pro Stück)
°C	<b>[°] [C], [d] [C]</b>
°F	<b>[°] [F], [d] [F]</b>
N <sub>o</sub>	<b>[N] [o]</b>
™	<b>[T] [M]</b>
$\%_s$	<b>[A] [/ S]</b> ( <i>Aktieselskab</i> )
→	<b>[-] [-] &gt;</b>
↗	<b>[-] [/ -] &gt;</b>
↔	<b>&lt; [-] &gt;</b>
⇒	<b>[=] [=] &gt;</b>
⇔	<b>&lt; [=] &gt;</b>
Σ	<b>[s] [u] [m]</b>
⊕	<b>[ ( ] [ + ] [ ) ]</b>
⊖	<b>[ ( ] [ - ] [ ) ]</b>
∅	<b>[o] [/]</b> (Durchmesser)
℞	<b>[x] [p]</b> (Chi-Rho)
☾	<b>[ ( ] [ * ]</b> (Halbmond und Stern)

☞        ☐☐ (Hammer und Sichel)

☯        ☐☐ (Yin-Yang)

☺        ☐☐

☺        ☐☐ (*Smiley*)

|| ☐# nicht vergessen! ||

## Unterstrich

Mit Unterstrich ☐ anstelle von Raute ☐ werden folgende Buchstaben gebildet:

⊖        ☐☐ (afrikanisches O)

⊣        ☐☐ (claudianisches *Digamma inversum*)

∞        ☐☐

⊍        ☐☐☐

sonstige umgekehrte Buchstaben ☐☐ etc.

ı        ☐☐

¿        ☐☐

|        ☐☐ (*dental click*)

||       ☐☐☐ (*lateral click*)

alleinstehende/kombinierbare Diakritika: siehe Tabelle

## Sonstige Zeichen

» «       ☐☐ ... ☐☐ (*Guillemets*)

±	<input type="button" value="+"/> <input type="button" value="-"/> <input type="button" value="Leerz."/>
-	<input type="button" value="-"/> (echtes Minuszeichen, auf Ziffernblock)
×	<input type="button" value="Leerz."/> <input type="button" value="*"/> <input type="button" value="*"/> (echtes Malzeichen)
÷	<input type="button" value="Leerz."/> <input type="button" value="/"/> <input type="button" value="/"/> (echtes Geteiltzeichen)
-	<input type="button" value="Leerz."/> <input type="button" value="-"/> <input type="button" value="-"/> (n-Strich)
-	<input type="button" value="-"/> <input type="button" value="-"/> <input type="button" value="-"/> (m-Strich)
„“	<input type="button" value="Leerz."/> <input type="button" value=""/> <input type="button" value=""/> ... <input type="button" value=""/> <input type="button" value=""/> <input type="button" value="Leerz."/>
≥	<input type="button" value="&gt;"/> <input type="button" value="="/>
≤	<input type="button" value="&lt;"/> <input type="button" value="="/>
≠	<input type="button" value="="/> <input type="button" value="/"/>
≅	<input type="button" value="="/> <input type="button" value="&gt;"/>
≈	<input type="button" value="="/> <input type="button" value="~"/>
⇌	<input type="button" value="&lt;"/> <input type="button" value="&gt;"/> (chemisches Gleichgewicht)
†	<input type="button" value="+"/> <input type="button" value=" "/>
‡	<input type="button" value="+"/> <input type="button" value="+"/>
...	<input type="button" value="."/> <input type="button" value="."/> <input type="button" value="."/> (Ellipse)
♥	<input type="button" value="&lt;"/> <input type="button" value="3"/>

## Deadkeys

Der Umfang der Deadkeys wurde erweitert. Jetzt ist es möglich, alle Buchstaben mit ´ ` , ¨ direkt einzugeben (auch kombiniert).

-

**Bernd Kappenberg: Zeichen setzen für Europa – Der Gebrauch europäischer lateinischer Sonderzeichen in der deutschen Öffentlichkeit, ISBN 3-89821-749-3**