

# Die BASIC-Briefmarke® II

Die BASIC-Briefmarke® II besitzt gegenüber ihrem Vorgänger, der BASIC-Briefmarke® I, folgende Verbesserungen:

- 2048 Byte EEPROM
- 32 Byte RAM-Speicher
- 16 I/O-Pins
- 20 MHz Taktca.
- 10000 BASIC Befehle/s
- RS232 mit 9600 Baud

Die BASIC-Briefmarke® II stellt 32 Byte des RAMs als Variablen-Speicherplatz zur Verfügung. Zwei Bytes werden für die I/O-Leitungen benötigt (1 Byte = Daten-Inhalt der 8 I/O-Pins, 1 Byte = Richtung der einzelnen 8 I/O-Pins). Die verbleibenden 29 Daten-Bytes können als Word-Variablen, Byte-Variablen, Nibble-Variablen oder als Bit-Variablen genutzt werden.

## Programmierung

Zur Definition von Variablen, Konstanten und Daten stehen die Befehle **VAR**, **CON** und **DATA** zur Verfügung. Mit folgenden Befehlen können Variablen definiert bzw. verändert werden:

sym1	var	bit	erzeugt eine Bit-Variablen
sym2	var	nib	erzeugt eine Nibble-Variablen
sym3	var	byte	erzeugt eine Byte-Variablen
sym4	var	word	erzeugt eine Word-Variablen
sym5	var	bit(10)	erzeugt eine 10 Bit-Variablen
sym6	var	sym4.lowbit	erzeugt eine Bit-Variablen aus den niederw. Bits von sym4

LOWBYTE	niederwertiges Byte eines Wortes
HIGHBYTE	höherwertiges Byte eines Wortes
BYTE0	Byte 0 (niederw. Byte) eines Wortes
BYTE1	Byte 1 (höherw. Byte) eines Wortes
LOWNIB	niederwertiges Nibble
HIGNIB	höherwertiges Nibble
NIB(n)	Nibbles n (n=0...3)
LOWBIT	niederwertiges Bit
HIGHBIT	höherwertiges Bit
BIT(n)	Bit(n) (n=0...15) eines Wortes, Byte, Nibble



Wilke Techno GmbH

Krefelder Str. 147, D-52070 Aachen

Telefon: 0241/918 900, Telefax: 0241/918 9044

## Editor-Funktionstasten:

<b>ALT-I</b>	Zeigt Versionsnr. des Interpreters
<b>Alt-R</b>	Programm zur Briefmarke, starten
<b>Alt-L</b>	Programm von Disk laden
<b>ALT-M</b>	Zeigt Speicherbelegung an
<b>Alt-S</b>	Programm auf Disk speichern
<b>Alt-Q oder ESC</b>	Editor verlassen, nach DOS

<b>Enter</b>	Information eingeben, neue Zeile
<b>Tab</b>	Wie Enter

<b>Cursor links</b>	Ein Zeichen nach links
<b>Cursor rechts</b>	Ein Zeichen nach rechts
<b>Cursor auf</b>	Eine Zeile nach oben
<b>Cursor runter</b>	Eine Zeile nach unten
<b>Ctrl-Links</b>	Ein Wort nach links
<b>Ctrl-Rechts</b>	Ein Wort nach rechts

<b>Home/Pos1</b>	Zum Anfang der Zeile
<b>End</b>	Zum Ende der Zeile
<b>Page Up</b>	Eine Seite zurück
<b>Page Down</b>	Eine Seite nach vorne
<b>Ctrl-Page Up</b>	Zum Anfang der Datei
<b>Ctrl-Page Down</b>	Zum Ende der Datei

<b>Shift-Left</b>	Ein Zeichen nach links markieren
<b>Shift-Right</b>	Ein Zeichen rechts markieren
<b>Shift-Up</b>	Eine Zeile nach oben markieren
<b>Shift-Down</b>	Eine Zeile nach unten markieren
<b>Shift-Ctrl-Left</b>	Ein Wort nach links markieren
<b>Shift-Ctrl-Right</b>	Ein Wort nach rechts markieren

<b>Shift-Home</b>	Bis Zeilenanfang markieren
<b>Shift-End</b>	Bis Zeilenende markieren
<b>Shift-Page Up</b>	Eine Seite rückwärts markieren
<b>Shift-Page Down</b>	Eine Seite vorwärts markieren
<b>Shift-Ctrl-Page Up</b>	Bis Datei-Anfang markieren
<b>Shift-Ctrl-Pg Down</b>	Bis Datei-Ende markieren

<b>Shift-Einfügen</b>	Wort unter Cursor markieren
<b>ESC</b>	Markierung löschen

<b>Backspace</b>	Zeichen vor Cursor löschen
<b>Delete</b>	Zeichen unter Cursor löschen
<b>Shift-Backspace</b>	Zeilenanfang bis Cursor löschen
<b>Shift-Delete</b>	Bis Zeilenende löschen
<b>Ctrl-Backspace</b>	Zeile löschen

<b>Alt-X</b>	Markierten Text ausschneiden
<b>Alt-C</b>	Markierten Text kopieren
<b>Alt-V</b>	Markierten Text einfügen
<b>Alt-F</b>	Suche Text
<b>Alt-N</b>	Suche nächsten Text

## BASIC-Befehle:

<b>BRANCH</b>	BRANCH offset,[adr-0,adr-1,...adr-N]
<b>BUTTON</b>	BUTTON pin,downstate,delay,rate,bytevar.,targetstate,adr
<b>COUNT</b>	◆ COUNT pin,time,variable
<b>DEBUG</b>	DEBUG
<b>DTMFOUT</b>	◆ DTMFOUT pin,[ontime,offtime,][key,key,...]
<b>END</b>	END
<b>FOR...NEXT</b>	FOR var=start TO end [step {-}incr]... NEXT {var}
<b>FREQOUT</b>	◆ FREQOUT pin,time,frequenz1{,frequenz2}
<b>GOSUB</b>	GOSUB address
<b>GOTO</b>	GOTO address
<b>HIGH</b>	HIGH pin
<b>IF...THEN</b>	IF var ?? value {AND/OR var ?? value...} THEN adr
<b>INPUT</b>	INPUT pin
<b>LOOKDOWN</b>	LOOKDOWN target,[value0,value1,...valueN],var
<b>LOOKUP</b>	LOOKUP offset,[value0,value1,...valueN],var
<b>LOW</b>	LOW pin
<b>NAP</b>	NAP period
<b>OUTPUT</b>	OUTPUT pin
<b>PAUSE</b>	PAUSE milliseconds
<b>PULSIN</b>	PULSIN pin,state,var
<b>PULSOUT</b>	PULSOUT pin,time
<b>PWM</b>	PWM pin,duty,cycles
<b>RANDOM</b>	RANDOM wordvar
<b>RCTIME</b>	◆ RCTIME pin,status,variable
<b>READ</b>	READ location,var
<b>RETURN</b>	RETURN
<b>REVERSE</b>	REVERSE pin
<b>SERIN</b>	◆ SERIN pin{vpin},baudmode,{plabel}{timeout,tlabel}[inputdata]
<b>SEROUT</b>	◆ SEROUT pin,baudmode,{pace,}[outputdata]
<b>SHIFTIN</b>	◆ SHIFTIN dpin,cpin,modus,[var{bits}...]
<b>SHIFTOUT</b>	◆ SHIFTOUT dpin,cpin,modus,[data{bits}...]
<b>SLEEP</b>	SLEEP seconds
<b>TOGGLE</b>	TOGGLE pin
<b>WRITE</b>	WRITE location,data
<b>XOUT</b>	◆ XOUT mpin,zpin,[housekeyorcommand{cycles,...}]

◆ Durch dieses Symbol werden Unterschiede zur BASIC-Briefmarke® gekennzeichnet

## Erklärung:

Springen auf durch offset spezifizierte Adr. Taste, Auto-Repeat, springe Adr falls Taste.

Mißt einen Impuls in Schritten von 8µs. Sende Variablen zur Ansicht an den PC. Sendet ein DTMF-Signal über einen Pin. unterminierte "Schlaf-Modus". Einrichten einer FOR-NEXT Schleife.

Gibt eine oder zwei Sinuswelle(n) aus. Springen auf Unterprogramm bei Adresse. Springen auf bestimmte Programmadresse. Macht Pin zu Output (high). Vergleichen und bedarfsweise springen. Macht Pin zu Input. Position von Zielwert in Tabelle -> in Var.

Tabellen-Eintrag bei "offset" in Variable. Macht Pin zu Output (low). "Schlummer-Modus". Strombedarf reduziert. Macht Pin zu Output. Pause 0-65536 Millisekunden. Mißt einen Input-Impuls. Impuls-Ausgabe (Toggle Pin).

Output PWM, danach Pin wieder Input. Generieren einer Pseudo-Zufallszahl. Mißt die Lade-/Entladezeit eines RC-Glied. Liest EEPROM Byte in eine Variable. Rücksprung aus Unterprogramm. Macht Input- zu Output-Pin und umgekehrt. Serieller Input mit opt. Triggern und Var.

Daten seriell senden. Liest Daten synchron seriell von Pin ein. Schiebt Daten seriell über einen Pin raus zeitlich begrenzter "Schlaf-Modus". Macht Pin zu Output und wechselt Status. Schreibe Byte ins EEPROM. Generiert einen X-10 Netzleitungscode

## Mathematische Operatoren:

+	-	addieren, subtrahieren
*		multiplizieren (Ergebnis=low word)
**		multiplizieren (Ergebnis=high word)
*/		multiplizieren (Ergebnis=middle word)
/,//		dividieren (Quotient), dividieren (Rest)
MIN		begrenzt Variable auf größer oder gleich
MAX		begrenzt Variable auf kleiner oder gleich
&,		logisch AND, OR
COS,SIN		Cosinus/Sinus einer 8-Bit-Zahl (Wertebereich ± 127)
~, -		1-er-Komplement, 2-er-Komplement

DD	2^n Decoder eines 4-Bit wertes
NOD	Prioritätscodierung
^	logisch XOR
&/	logisch AND NOT
//	logisch OR NOT
^/	logisch XOR NOT
>>,<<	rechts shift, links shift
FEV	wechselt den Zustand der Bits

## Vergleichs-Operatoren:

= < > < < > >