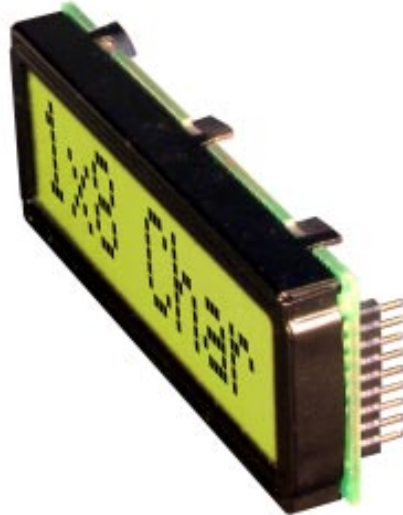


LCD- MODUL 1x8 - 11,48mm**INKL. KONTROLLER HD 44780****keine Montage
erforderlich**

*Abmessungen 68 x 27 mm
auch mit LED nur 11mm flach*

TECHNISCHE DATEN

- * KONTRASTREICHE LCD-SUPERTWIST ANZEIGE
- * HD 44780 KOMPATIBEL
- * ANSCHLUSS AN 4- ODER 8-BIT DATENBUS
- * SPANNUNGSVERSORGUNG +5V
- * BETRIEBSTEMPERATURBEREICH 0..+50°C
- * LED-HINTERGRUNDBELEUCHTUNG GB/GN typ. 70mA@4,1V, max.150mA
- * WEITERE MODULE IM GLEICHEN GEHÄUSE, GLEICHES PINOUT:
 - DOTMATRIX 2x16, 4x20
 - GRAFIK 122x32
- * KEINE MONTAGE ERFORDERLICH: EINFACH NUR IN PCB EINLÖTEN
- * STECKBAR ÜBER BUCHSENLEISTEN EA B200-9 (2 STK. ERFORDERLICH)

BESTELLBEZEICHNUNG

LCD-MODUL 1x8 - 11,48mm MIT LED-BELEUCHTUNG gb/gn
9-POL. BUCHSENLEISTE (1 STÜCK)
PASSENDER FRONTRAHMEN (FENSTER 60,0x14,8mm)

EA DIP081-CNLED
EA B200-9
EA 017-2UKE

PINBELEGUNG

Pin	Symbol	Level	Funktion	Pin	Symbol	Level	Funktion
1	VSS	L	Stromversorgung 0V (GND)	10	D3	H / L	Display Data
2	VDD	H	Stromversorgung +5V	11	D4 (D0)	H / L	Display Data
3	VEE	-	Kontrastspg. (ca. 0V)	12	D5 (D1)	H / L	Display Data
4	RS	H / L	Umschaltung Befehl / Daten	13	D6 (D2)	H / L	Display Data
5	R/W	H / L	H=Read, L=Write	14	D7 (D3)	H / L	Display Data, MSB
6	E	H	Enable (fallende Flanke)	15	-	-	frei (siehe EA DIP122-5N)
7	D0	H / L	Display Data, LSB	16	-	-	frei (siehe EA DIP122-5N)
8	D1	H / L	Display Data	17	A	-	LED-Bel. + (RV erford.)
9	D2	H / L	Display Data	18	C	-	LED-Bel. -

BEFEHLSTABELLE

Instruction	Code										Description	Execute Time (max.)
	RS	R/W	DB 7	DB 6	DB 5	DB 4	DB 3	DB 2	DB 1	DB 0		
Clear Display	0	0	0	0	0	0	0	0	0	1	Clears all display and returns the cursor to the home position (Address 0).	1.64ms
Cursor At Home	0	0	0	0	0	0	0	0	1	*	Returns the Cursor to the home position (Address 0). Also returns the display being shifted to the original position. DD RAM contents remain unchanged.	1.64ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S	Sets the Cursor move direction and specifies or not to shift the display. These operation are performed during data write and read.	40µs
Display On/Off Control	0	0	0	0	0	0	1	D	C	B	Sets ON/OFF of all display (D) cursor ON/OFF (C), and blink of cursor position character (B).	40µs
Cursor / Display Shift	0	0	0	0	0	1	S/C	R/L	*	*	Moves the Cursor and shifts the display without changing DD RAM contents.	40µs
Function Set	0	0	0	0	1	DL	N	F	*	*	Sets interface data length (DL) number of display lines (L) and character font (F).	40µs
CG RAM Address Set	0	0	0	1	ACG					Sets the CG RAM address. CG RAM data is sent and received after this setting.		40µs
DD RAM Address Set	0	0	1	ADD					Sets the DD RAM address. DD RAM data is sent and received after this setting.		40µs	
Busy Flag / Address Read	0	1	BF	AC					Reads Busy flag (BF) indicating internal operation is being performed and reads address counter contents.		-	
CG RAM / DD RAM Data write	1	0	Write Data					Writes data into DD RAM or CG RAM		40µs		
CG RAM / DD RAM Data Read	1	1	Read Data					Reads data from DD RAM or CG RAM		40µs		

INITIALISIERUNGSBEISPIEL FÜR DEN 8-BIT MODUS

Befehl	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Bemerkung
Function Set	0	0	0	0	1	1	0	0	0	0	8-Bit Datenlänge, 1-zeiliges Display, 5x7 Font
Display ON/OFF	0	0	0	0	0	0	1	1	1	1	Display ein, Cursor ein, Cursor blinken
Clear Display	0	0	0	0	0	0	0	0	0	1	Display löschen, Cursor auf 1. Spalte von 1. Zeile
Entry Mode Set	0	0	0	0	0	0	0	1	1	0	Cursor Auto-Increment

ZEICHENSATZ

Unten abgebildeter Zeichensatz ist bereits integriert. Zusätzlich können 8 eigene Zeichen frei definiert werden.

Lower 4 bit \ Upper 4 bit	0000 (\$0x)	0010 (\$2x)	0011 (\$3x)	0100 (\$4x)	0101 (\$5x)	0110 (\$6x)	0111 (\$7x)	1010 (\$Ax)	1011 (\$Bx)	1100 (\$Cx)	1101 (\$Dx)	1110 (\$Ex)	1111 (\$Fx)
xxxx0000 (\$x0)	CG RAM (0)	0	@	P	`	F		-	ウ	ミ	α	p	
xxxx0001 (\$x1)	(1)	!	1	A	Q	a	q	μ	ア	チ	△	ä	q
xxxx0010 (\$x2)	(2)	"	2	B	R	b	r	Γ	イ	ウ	×	β	θ
xxxx0011 (\$x3)	(3)	#	3	C	S	c	s	┘	ウ	テ	ε	ε	ω
xxxx0100 (\$x4)	(4)	\$	4	D	T	d	t	√	I	ト	ト	μ	Ω
xxxx0101 (\$x5)	(5)	%	5	E	U	e	u	•	オ	ナ	1	ε	Ü
xxxx0110 (\$x6)	(6)	&	6	F	V	f	v	ヲ	カ	ニ	ヨ	ρ	Σ
xxxx0111 (\$x7)	(7)	'	7	G	W	g	w	ア	キ	ヌ	ラ	g	π
xxxx1000 (\$x8)	CG RAM (0)	(8	H	X	h	x	イ	ウ	ネ	リ	フ	Σ
xxxx1001 (\$x9)	(1))	9	I	Y	i	y	ウ	ケ	ル	ル	"	y
xxxx1010 (\$xA)	(2)	*	:	J	Z	j	z	エ	コ	ハ	レ	j	≠
xxxx1011 (\$xB)	(3)	+	:	K	[k	[オ	サ	ヒ	ロ	*	≠
xxxx1100 (\$xC)	(4)	,	<	L	¥	l	l	パ	シ	フ	ワ	φ	≠
xxxx1101 (\$xD)	(5)	-	=	M]	m]	ユ	ズ	ハ	ン	ε	÷
xxxx1110 (\$xE)	(6)	.	>	N	^	n	→	ヨ	セ	ホ	ハ	ñ	
xxxx1111 (\$xF)	(7)	/	?	O	_	o	←	ウ	リ	マ	"	ö	■

PROGRAMMIERUNG VON SELBSTDEFINIERTEN ZEICHEN

Bei allen hier angebotenen Dotmatrixdisplays (Text) können zusätzlich zu den 192 im ROM fest einprogrammierten Zeichen bis zu 8 weitere frei definiert werden (ASCII Codes 0..7).

- 1.) Mit dem Kommando "CG RAM Address Set" wird der ASCII Code (Bit 3,4,5) und die entsprechende Pixelzeile (Bit 0,1,2) des Zeichens angewählt. Im Beispiel wird ein Zeichen mit dem Code \$00 definiert.
- 2.) Mit dem Befehl "Data Write" wird nun Pixelzeile für Pixelzeile das Zeichen in das CG RAM geschrieben. Ein Zeichen benötigt 8 Schreiboperationen, wobei die 8. Zeile der Cursorzeile entspricht.
- 3.) Das neu definierte Zeichen wird genauso behandelt wie ein "normales" ASCII Zeichen (Verwendung: "DD RAM Address Set", "Data Write").

Adresse im CG RAM setzen				Daten des Zeichens									
Adresse		Hex	Bit								Hex		
7	6	5	4	3	2	1	0						
0	1	0	0	0	0	0	0	0	0	1	0	0	\$04
								0	0	1	0	0	\$04
								0	0	1	0	0	\$04
								0	0	1	0	0	\$04
								0	0	1	0	0	\$04
								1	0	1	0	1	\$15
								0	1	1	1	0	\$0E
								0	0	1	0	0	\$04
								0	0	0	0	\$00	

