

# OLED-DISPLAYS

## INKL. KONTROLLER FÜR 8-BIT UND 4-BIT



OLED										
Part-Number	Row x Column	Char height	Module size			Viewing Area		Accessories (Frames)	Hints	Drawing page
			B	H	T	B	H			
EA W082-XLG	2x8	5.5	58.0	32.0	10.0	38.0	16.0	---	yellow/green	7
EA W162-X3LW	2x16	5.5	80.0	36.0	10.0	66.0	16.0	EA 017-2U	icewhite	7
EA W162-X3LG	2x16	5.5						EA 017-2U	yellow/green	7
EA W162-X9LG	2x16	5.5	85.0	36.0	10.0	66.0	16.0	EA 017-2U	yellow/green	8
EA W162-XLG	2x16	5.5	84.0	44.0	10.0	66.0	16.0	EA 017-2U	yellow/green	8
EA W162-XBLW	2x16	8.9	122.0	55.0	10.0	99.0	24.0	EA 017-12U	icewhite	9
EA W162-XBLG	2x16	8.9						EA 017-12U	yellow/green	9
EA W202-XLG	2x20	5.5	116.0	37.0	9.8	85.0	18.6	EA 017-7U	yellow/green	9
EA W204-XLG	4x20	5.5	98.0	60.0	10.0	70.0	25.2	EA 017-9U	yellow/green	10

### TECHNISCHE DATEN

- \* INTEGRIERTER KONTROLLER (HD44780-ÄHNLICH)
- \* EINGANG 4- ODER 8-BIT DATENBUS, 3 STEUERLEITUNGEN (R/W, E, RS)
- \* ASCII-ZEICHENSATZ UND SONDERZEICHEN IM CHARACTER-ROM
- \* BIS ZU 8 ZEICHEN (ASCII-CODE 0..7) KÖNNEN FREI DEFINIERT WERDEN
- \* VERSCHIEDENE FUNKTIONEN MIT EINEM BEFEHL PROGRAMMIERBAR:
  - CLEAR DISPLAY, CURSOR HOME, CURSOR ON/OFF, BLINKING CURSOR
  - SHIFT DISPLAY, SHIFT CURSOR, READ/WRITE DISPLAY DATA, ETC.
- \* EINFACHE SPANNUNGSVERSORGUNG (3.3..5V).
- \* GERINGER STROMVERBRAUCH (15..50 mA)
- \* BETRIEBSTEMPERATUR -40..+80°C
- \* 3 INTEGRIERTE FONTS

### ZUBEHÖR

- \* ABDECKRAHMEN (SIEHE TABELLE)

**ELECTRONIC  
ASSEMBLY**

making things easy

## ZEICHENSATZ

ENGLISH\_JAPANESE CHARACTER FONT TABLE (default FT[1:0]= 00)

Upper ABC	Lower ABC	LLLL	LLLN	LLHL	LLHH	LHLL	LHLN	LHHL	LHHH	NLLL	NLHN	NLHL	NLHN	NHLL	NHHL	NHHH
CO RAM (1)																
CO RAM (2)																
CO RAM (3)																
CO RAM (4)																
CO RAM (5)																
CO RAM (6)																
CO RAM (7)																
CO RAM (8)																
CO RAM (9)																
CO RAM (10)																
CO RAM (11)																
CO RAM (12)																
CO RAM (13)																
CO RAM (14)																
CO RAM (15)																
CO RAM (16)																
CO RAM (17)																
CO RAM (18)																
CO RAM (19)																
CO RAM (20)																

WESTERN EUROPEAN CHARACTER FONT TABLE I (FT[1:0]=01)

Upper ABC	Lower ABC	LLLL	LLLN	LLHL	LLHH	LHLL	LHLN	LHHL	LHHH	NLLL	NLHN	NLHL	NLHN	NHLL	NHHL	NHHH
CO RAM (1)																
CO RAM (2)																
CO RAM (3)																
CO RAM (4)																
CO RAM (5)																
CO RAM (6)																
CO RAM (7)																
CO RAM (8)																
CO RAM (9)																
CO RAM (10)																
CO RAM (11)																
CO RAM (12)																
CO RAM (13)																
CO RAM (14)																
CO RAM (15)																
CO RAM (16)																
CO RAM (17)																
CO RAM (18)																
CO RAM (19)																
CO RAM (20)																

ENGLISH RUSSIAN CHARACTER FONT TABLE (FT[1:0]=10)

Upper ABC	Lower ABC	LLLL	LLLN	LLHL	LLHH	LHLL	LHLN	LHHL	LHHH	NLLL	NLHN	NLHL	NLHN	NHLL	NHHL	NHHH
CO RAM (1)																
CO RAM (2)																
CO RAM (3)																
CO RAM (4)																
CO RAM (5)																
CO RAM (6)																
CO RAM (7)																
CO RAM (8)																
CO RAM (9)																
CO RAM (10)																
CO RAM (11)																
CO RAM (12)																
CO RAM (13)																
CO RAM (14)																
CO RAM (15)																
CO RAM (16)																
CO RAM (17)																
CO RAM (18)																
CO RAM (19)																
CO RAM (20)																

Es ist möglich aus einem der hier abgebildeten Zeichensätze frei zu wählen. Standardmäßig wird der Englisch/Japanische Zeichensatz aktiviert. Hierfür müssen die Bits FT1/FT0 aus dem FunctionSet-Register entsprechend gesetzt werden:

FT 1	FT 0	Description
0	0	Englisch Japanese character font table
0	1	Western European character font table
1	0	English Russian character font table
1	1	not available

Anmerkung: Der Zeichensatz muss vor allen anderen Befehlen und Einstellungen ausgewählt werden (ausgenommen Busy-Flag-Abfrage und Adressoperationen).

## PROGRAMMIERUNG VON SELBSTDEFINIERTEN ZEICHEN

Bei allen hier angebotenen OLED-Displays können zusätzlich zu den 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								
				7	6	5	4	3	2	1	0	Hex
0	1	0	0	0	0	0	0	0	1	0	0	\$04
			0	0	1	0	0	1	0	0	0	\$04
			0	1	0	0	1	0	0	0	0	\$04
			0	1	1	0	1	0	0	0	0	\$04
			1	0	0	0	1	0	1	0	1	\$15
			1	0	1	0	1	1	1	0	0	\$0E
			1	1	0	0	1	0	0	0	0	\$04
			1	1	1	0	0	0	0	0	0	\$00

## PINBELEGUNG

PIN	SYMBOL	Level	DESCRIPTION	
			8 BIT-Mode	4 BIT-Mode
1	GND	L (0V)	Ground	Ground
2	VCC	H (5V)	Supply Voltage	Supply Voltage
3	N.C.	---	not connected	not connected
4	RS	H/L	H: Data, L: Instruction code	H: Data, L: Instruction code
5	R/W	H/L	H: Read, L: Write	H: Read, L: Write
6	E	H, H->L	Enable	Enable
7	DB0	H/L	Data bit 0 (LSB)	Do not connect
8	DB1	H/L	Data bit 1	Do not connect
9	DB2	H/L	Data bit 2	Do not connect
10	DB3	H/L	Data bit 3	Do not connect
11	DB4	H/L	Data bit 4	Data bit 0 (LSB)
12	DB5	H/L	Data bit 5	Data bit 1
13	DB6	H/L	Data bit 6	Data bit 2
14	DB7	H/L	Data bit 7 (MSB)	Data bit 3 (MSB)
15	N.C.	---	not connected	not connected
16	N.C.	---	not connected	not connected

## BEFEHLSSATZ

Instruction	Code										Description	Max. execution time when fsp or fosc=250KHz
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear Display	0	0	0	0	0	0	0	0	0	1	Clears entire Display, Sets DDRAM-address 0 into addresscounter	6.2ms
Return Home	0	0	0	0	0	0	0	0	1	0	Sets DDRAM-address 0 into addresscounter. Returns shifted display to original position. DDRAM contents remain unchanged.	0ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S	Sets cursor move direction and specifies display shift.(These operations are performed during data write and read.)	0ms
Display On/Off Control	0	0	0	0	0	0	1	D	C	B	Sets entire Display (D) ON/OFF. Sets Cursor (C) ON/OFF. Sets Blinking (B) of Cursor Position Character.	0ms
Cursor/Display Shift/Mode/Pwr	0	0	0	0	0	1	S/C	R/L	0	0	Moves cursor and shifts display without changing DDRAM contents.	0ms
							G/C (0)	PWR	1	1	Sets Graphic/Character Mode Sets internal power on/off	0ms
Function Set	0	0	0	0	1	DL	N	F	FT1	FT0	Sets interface data length (DL). Sets number of display lines (N). Sets Character Font (F). Sets Font Table (FT).	0ms
Set CGRAM Address	0	0	0	1	ACG	ACG	ACG	ACG	ACG	ACG	Sets CGRAM Address. CGRAM data is sent and received after this setting.	0ms
Set DDRAM Address	0	0	1	ADD	ADD	ADD	ADD	ADD	ADD	ADD	Sets DDRAM Address. The DDRAM data is sent and received after this setting.	0ms
Read Busy Flag and Address	0	1	BF	AC	AC	AC	AC	AC	AC	AC	Reads Busy Flag (BF) indicating that internal operation is being performed. Reads Address Counter contents.	0ms
Write data into the CGRAM or DDRAM	1	0	Write Data							Writes data into the CGRAM or DDRAM		0ms
Read data from the CGRAM or DDRAM	1	1	Read Data							Reads data from the CGRAM or DDRAM		0ms

### Anmerkungen zum Befehlssatz:

1. Nach Ausführen von CGRAM/DDRAM Read or Write Instruction, wird der RAM Address Counter incrementiert oder decrementiert. Nachdem das Busy Flag nicht mehr gesetzt ist, wird die RAM Adresse übernommen.
2. **I/D**=Increment/Decrement Bit  
I/D="1": Incrementieren  
I/D="0": Decrementieren
3. **S**=Shift Entire Display Control Bit.  
S="0", shift function deaktiviert.  
S="1", shift function aktiviert
4. **BF**=Busy Flag  
BF="1": Interne Operationen werden durchgeführt  
BF="0": Keine internen Operationen, der nächste Befehl kann entgegen genommen werden.
5. **R/L**=Shift Right/Left  
R/L="1": Shift nach Rechts  
R/L="0": Shift nach Links
6. **S/C**=Display Shift/Cursor Move  
S/C="1": Display Shift  
S/C="0": Cursor Move
7. **G/C**=Graphic/Character mode selection.  
G/C="0", Character mode is selected.  
G/C="1", Graphic mode is selected.
8. **PWR**=Internal DCDC on/of control.  
PWR="1", DCDC an  
PWR="0", DCDC aus
9. **DDRAM**=Display Data RAM
10. **CGRAM**=Character Generator RAM
11. **ACG**=CGRAM Address
12. **ADD**=Address Counter Address (corresponds to cursor address)
13. **AC**=Address Counter (used for DDRAM and CGRAM Addresses)
14. **F**=Character Pattern Mode  
F="1": 5 x 10 dots  
F="0": 5 x 8 dots
15. **N**=Number of Lines Displayed  
N="1": 2- und 4-Line Display  
N="0": 1-Line Display

## INITIALISIERUNGSBEISPIELE

Initialisation example: 8-Bit											
RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	HEX	Description
0	0	0	0	1	1	1	0	0	1	\$39	Function Set, western european character set, 8-Bit
0	0	0	0	0	0	1	0	0	0	\$08	Display off
0	0	0	0	0	0	0	1	1	0	\$06	Entry mode set, increment cursor by 1 not shifting display
0	0	0	0	0	1	0	1	1	1	\$17	Character mode and internal power on (have to turn on internal power to get the best brightness)
0	0	0	0	0	0	0	0	0	1	\$01	Clear display
0	0	0	0	0	0	0	0	1	0	\$02	Return home
0	0	0	0	0	0	1	1	0	0	\$0C	Display on

*void* `initDisplay(void)`

```
{
    RS_DD=1; //RS Pin as output
    RW_DD=1; //RW Pin as output
    EN_DD=1; //EN Pin as output
    WriteIns(0x39); //function set european character set
    WriteIns(0x08); //display off
    WriteIns(0x06); //entry mode set increment cursor by 1 not shifting display
    WriteIns(0x17); //Character mode and internal power on
    WriteIns(0x01); //clear display
    WriteIns(0x02); //return home
    WriteIns(0x0C); //display on
}
```

*void* `WriteIns(char instruction)`

```
{
    DATA_PORT_DD=0xFF; //Dataport as Output
    RS = 0;
    RW = 0;
    DATA_PORT = instruction; //set Data on Outputport
    EN = 1; //set Enable to high
    Wait(10); //wait 1us (stabilize Outputport)
    EN = 0; //reset Enable to low
    CheckBusy();
}
```

Initialisation example: 4-Bit											
RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	HEX	Description
0	0	0	0	1	0	1	0	0	0	\$28	Function Set, English/Japanese char set, 4-Bit
0	0	0	0	0	0	1	0	0	0	\$08	Display off
0	0	0	0	0	0	0	1	1	0	\$06	Entry mode set, increment cursor by 1 not shifting display
0	0	0	0	0	1	0	1	1	1	\$17	Character mode and internal power on (have to turn on internal power to get the best brightness)
0	0	0	0	0	0	0	0	0	1	\$01	Clear display
0	0	0	0	0	0	0	0	1	0	\$02	Return home
0	0	0	0	0	0	1	1	0	0	\$0C	Display on

*void* `initDisplay(void)`

```
{
    RS_DD=1; //RS-Pin as Output
    EN_DD=1; //EN-Pin as Output
    RW_DD=1; //RW-Pin as Output
    RS = 0; //RS-Pin to low
    RW = 0; //RW-Pin to low
    EN = 0; //EN-Pin to low
    send_nibble(0x03); //Be sure to
    send_nibble(0x03); //be in
    send_nibble(0x03); //8-Bit-Mode
    send_nibble(0x02); //Switch to 4 Bit
    Wait(50); //Wait 5us
    WriteIns(0x28); //4-Bit-Mode
    WriteIns(0x08); //display off
    WriteIns(0x06); //entry mode set increment cursor by 1 not shifting display
    WriteIns(0x17); //Character mode and internal power on
    WriteIns(0x01); //clear display
    WriteIns(0x02); //return home
    WriteIns(0x0C); //display on
}
```

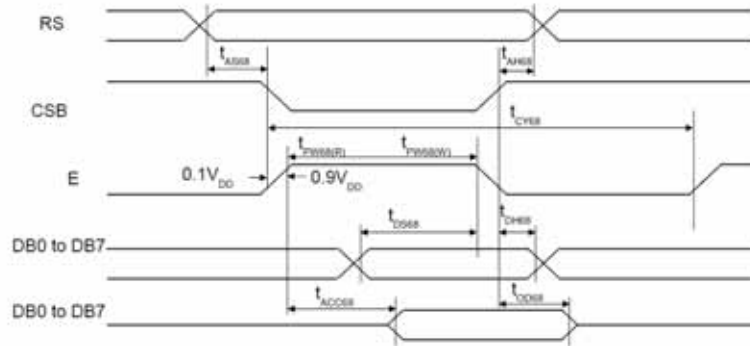
*void* `send_nibble(char data)`

```
{
    DATA_PORT = data; //output data
    EN=1;
    Wait(10); //wait 1us (stabilize output)
    EN=0;
    Wait(10); //wait 1us (stabilize output)
}
```

*void* `WriteIns(char instruction)`

```
{
    DATA_PORT_DD=0x0F; //Dataport as Output
    RS = 0;
    RW = 0;
    send_nibble((instruction&0xF0)>>4); //Highbyte
    send_nibble(instruction&0x0F); //Lowbyte
    CheckBusy();
}
```

## TIMING 8-BIT INTERFACE



(VDD = 3.0 to 5.3V, Ta = 25°C)

Item	Signal	Symbol	Min.	Typ.	Max.	Unit	Remark
Address setup time	RS	t <sub>AS68</sub>	20	-	-	ns	
Address hold time	RS	t <sub>AH68</sub>	0	-	-	ns	
System cycle time		t <sub>CY68</sub>	500	-	-	ns	
Pulse width (E)	E_RDB	t <sub>PW68(W)</sub>	250	-	-	ns	
Pulse width (E)	E_RDB	t <sub>PW68(R)</sub>	250	-	-	ns	
Data setup time	DB7 to DB0	t <sub>DS68</sub>	40	-	-	ns	
Data hold time	DB7 to DB0	t <sub>DH68</sub>	20	-	-	ns	
Read access time	DB0	t <sub>ACC68</sub>	-	-	180	ns	C <sub>L</sub> = 100pF
Output disable time	DB0	t <sub>OD68</sub>	10	-	-	ns	

## ELECTRICAL CHARACTERISTICS

Item	Symbol	Test Condition	Standard Value			Unit
			min.	typ.	max.	
Input "high" voltage	V <sub>IH</sub>	-	0.9 VDD	-	VDD	V
Input "low" voltage	V <sub>IL</sub>	-	GND	-	0.1 VDD	V
Output "high" voltage	V <sub>OH</sub>	I <sub>OH</sub> =-0.5mA	0.8 VDD	-	VDD	V
Output "low" voltage	V <sub>OL</sub>	I <sub>OL</sub> =0.5mA	GND	-	0.2 VDD	V
Power supply current	I <sub>CC</sub>	VDD=5V		15..50mA		A

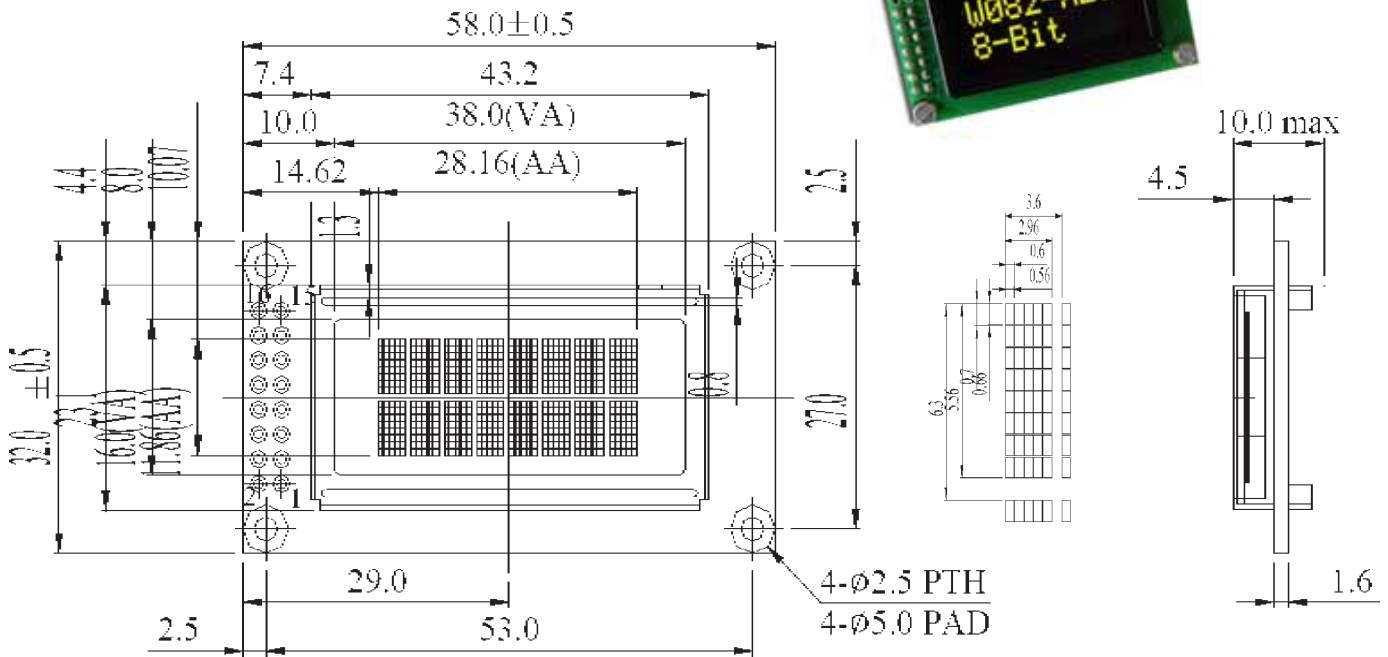
## ABSOLUTE MAXIMUM RATINGS

VCC=5,0V, Ta=25°C

Item	Symbol	Standard Value		Unit
		min.	max.	
Power supply voltage for logic	VDD-GND	-0.3	5.3	V
Input Voltage	V <sub>I</sub>	-0.3	VDD	V
Operating temperature	T <sub>OP</sub>	-40	80	°C
Storage Temperature	T <sub>ST</sub>	-40	80	°C

Bei Versorgung mit 3,3V, ist die Helligkeit gegenüber 5V reduziert.

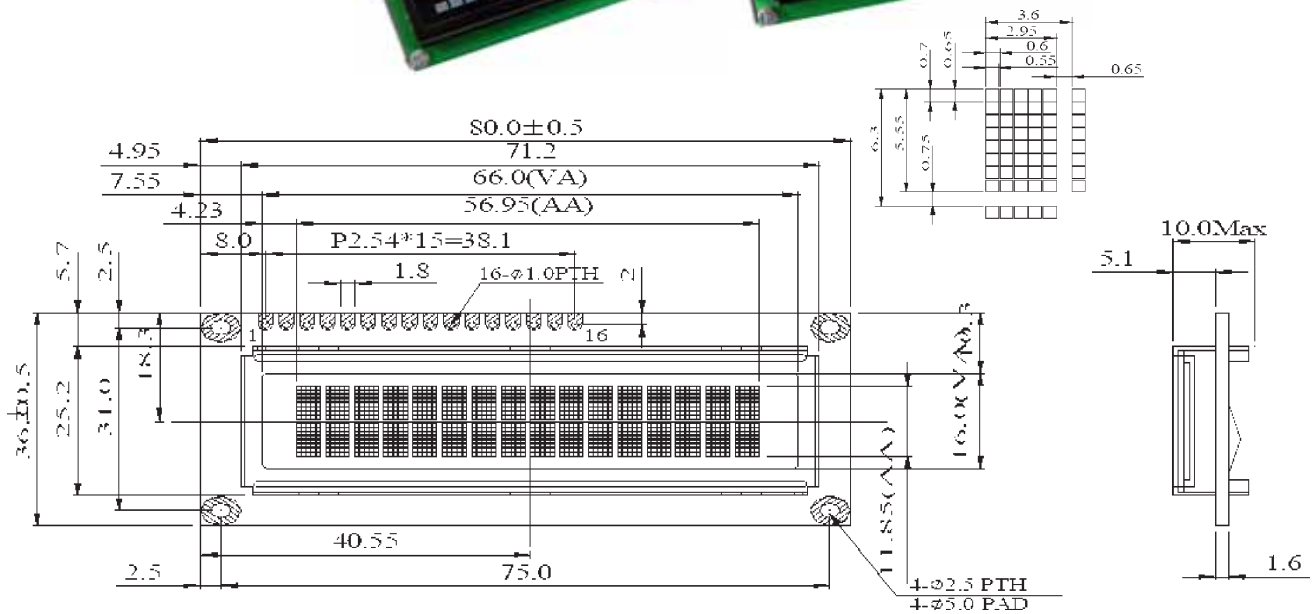
**EA W082-XLG**



**EA W162-X3LW (ICEWHITE)**  
**EA W162-X3LG (GELB/GRÜN)**

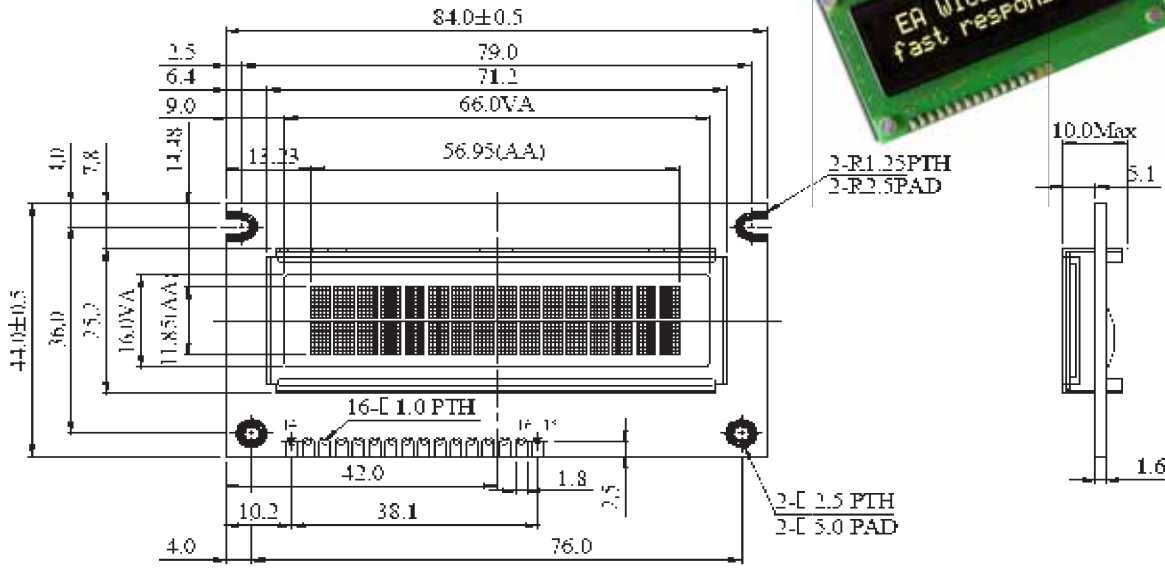


**RAHMEN**  
**EA 017-2U**



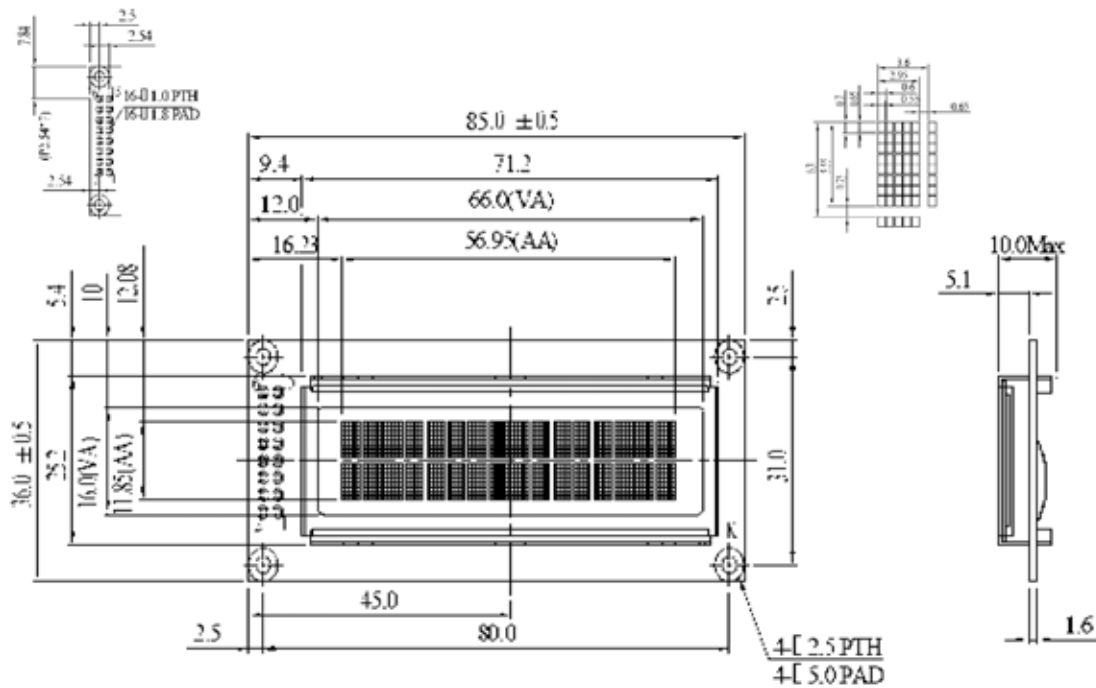
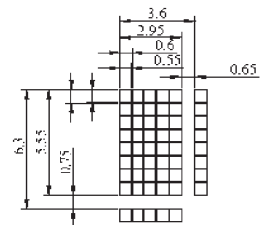
## EA W162-XLG (GELB/GRÜN)

**RAHMEN  
EA 017-2U**



## EA W162-X9LG

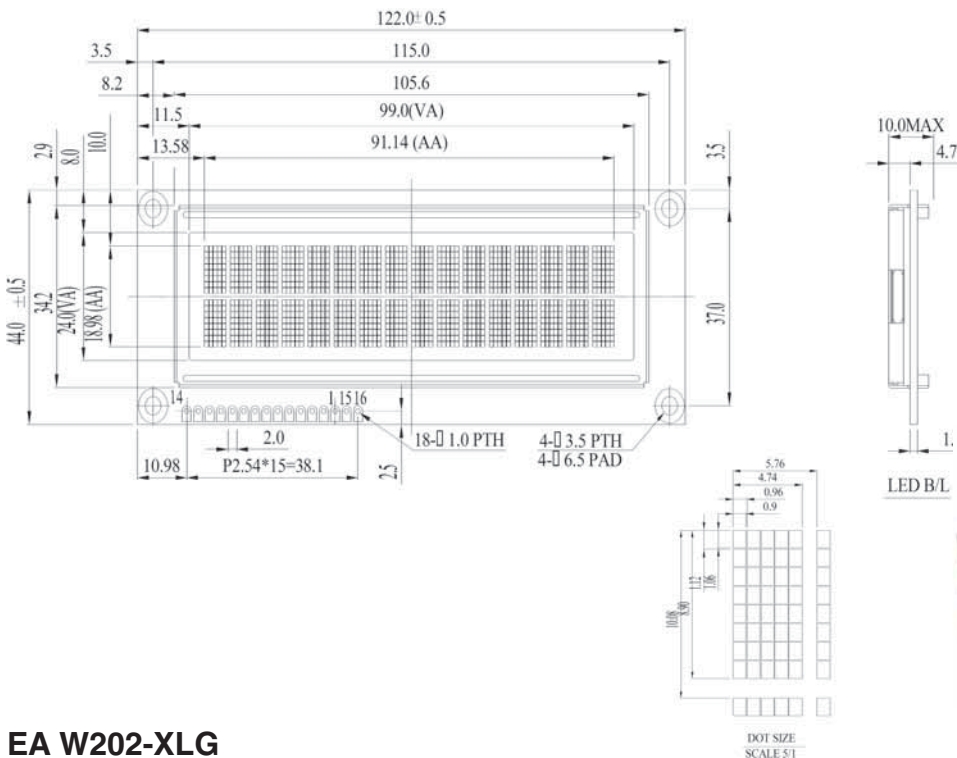
**RAHMEN  
EA 017-2U**





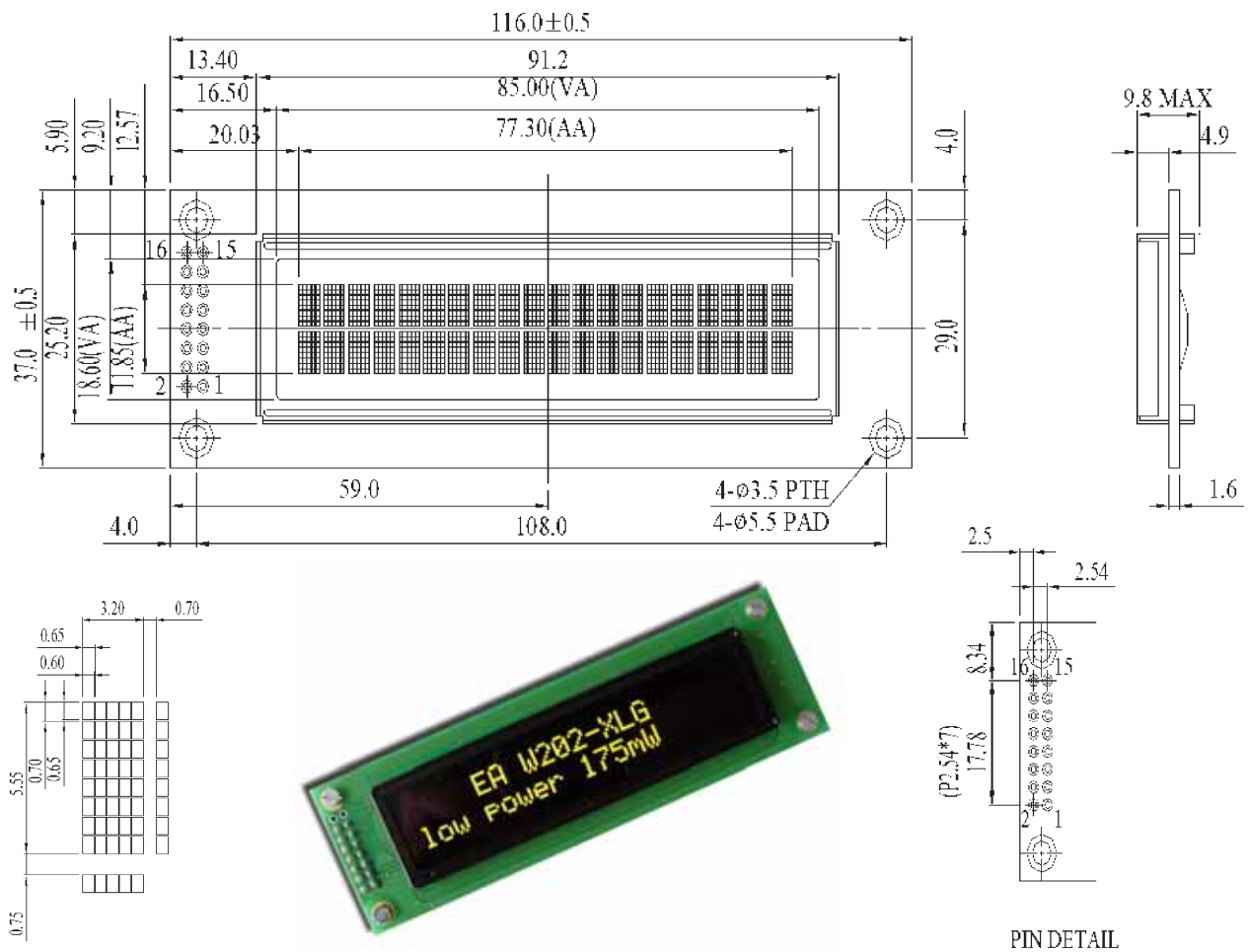
**EA W162-XBLW (ICEWHITE)**  
**EA W162-XBLW (GELB/GRÜN)**

**RAHMEN**  
**EA 017-12U**



**EA W202-XLG**

**RAHMEN**  
**EA 017-7U**



## EA W204-XLG

## RAHMEN EA 017-9U

