

# Ansteuerung eines LCD-Screens

Marcel Meinersen

4. Mai 2013

## 1 Allgemeines

- Was ist ein LCD-Screen?
- LCD 162C LED

- 1 Allgemeines
  - Was ist ein LCD-Screen?
  - LCD 162C LED
- 2 Beschaltung mit Mikrocontroller
  - Blockschaltbild unserer Gruppe
  - LCD-Screen-Anschlüsse
  - How to connect: LCD-ATmega32
  - Control and Display-Commands

- 1 Allgemeines
  - Was ist ein LCD-Screen?
  - LCD 162C LED
- 2 Beschaltung mit Mikrocontroller
  - Blockschaltbild unserer Gruppe
  - LCD-Screen-Anschlüsse
  - How to connect: LCD-ATmega32
  - Control and Display-Commands
- 3 AVR-GCC-Anwendung
  - Wiederholung der Grundoperationen
  - AVR-GCC-Beispiel
  - LCD-Commands

- 1 Allgemeines
  - Was ist ein LCD-Screen?
  - LCD 162C LED
- 2 Beschaltung mit Mikrocontroller
  - Blockschaltbild unserer Gruppe
  - LCD-Screen-Anschlüsse
  - How to connect: LCD-ATmega32
  - Control and Display-Commands
- 3 AVR-GCC-Anwendung
  - Wiederholung der Grundoperationen
  - AVR-GCC-Beispiel
  - LCD-Commands
- 4 Quellenangabe, Schlusswort

## 1 Allgemeines

- Was ist ein LCD-Screen?
- LCD 162C LED

## 2 Beschaltung mit Mikrocontroller

- Blockschaltbild unserer Gruppe
- LCD-Screen-Anschlüsse
- How to connect: LCD-ATmega32
- Control and Display-Commands

## 3 AVR-GCC-Anwendung

- Wiederholung der Grundoperationen
- AVR-GCC-Beispiel
- LCD-Commands

## 4 Quellenangabe, Schlusswort

# Was ist ein LCD-Screen?

# Was ist ein LCD-Screen?

## Definition

Der LCD (engl. liquid crystal display) ist ein Flüssigkristallbildschirm, der mithilfe elektrischer Spannungen (elektrische Felder) die Ausrichtung und die Durchlässigkeit der Flüssigkristalle, und damit die Polarisationsrichtung des Lichtes, verändern kann.

# Was ist ein LCD-Screen?

## Definition

Der LCD (engl. liquid crystal display) ist ein Flüssigkristallbildschirm, der mithilfe elektrischer Spannungen (elektrische Felder) die Ausrichtung und die Durchlässigkeit der Flüssigkristalle, und damit die Polarisationsrichtung des Lichtes, verändern kann.



# Was ist ein LCD-Screen?

## Definition

Der LCD (engl. liquid crystal display) ist ein Flüssigkristallbildschirm, der mithilfe elektrischer Spannungen (elektrische Felder) die Ausrichtung und die Durchlässigkeit der Flüssigkristalle, und damit die Polarisationsrichtung des Lichtes, verändern kann.



# Beispiel: LCD 162C LED

# Beispiel: LCD 162C LED



Quelle: [www.reichelt.de](http://www.reichelt.de)

# Beispiel: LCD 162C LED



Quelle: [www.reichelt.de](http://www.reichelt.de)

## Features

# Beispiel: LCD 162C LED



Quelle: [www.reichelt.de](http://www.reichelt.de)

## Features

- 2 x 16 Zeichen (ASCII-Code /Standard character)

# Beispiel: LCD 162C LED



Quelle: [www.reichelt.de](http://www.reichelt.de)

## Features

- 2 x 16 Zeichen (ASCII-Code /Standard character)
- Physikalische Auflösung: 16\*80 Pixel

# Beispiel: LCD 162C LED



Quelle: [www.reichelt.de](http://www.reichelt.de)

## Features

- 2 x 16 Zeichen (ASCII-Code /Standard character)
- Physikalische Auflösung: 16\*80 Pixel
- Controller KS0070B integriert

# Beispiel: LCD 162C LED



Quelle: [www.reichelt.de](http://www.reichelt.de)

## Features

- 2 x 16 Zeichen (ASCII-Code /Standard character)
- Physikalische Auflösung: 16\*80 Pixel
- Controller KS0070B integriert
- Betriebsspannung: 4,5...5,1 V

# Beispiel: LCD 162C LED



Quelle: [www.reichelt.de](http://www.reichelt.de)

## Features

- 2 x 16 Zeichen (ASCII-Code /Standard character)
- Physikalische Auflösung: 16\*80 Pixel
- Controller KS0070B integriert
- Betriebsspannung: 4,5...5,1 V
- LED-Backlight (grün-reflektiv)

## 1 Allgemeines

- Was ist ein LCD-Screen?
- LCD 162C LED

## 2 Beschaltung mit Mikrocontroller

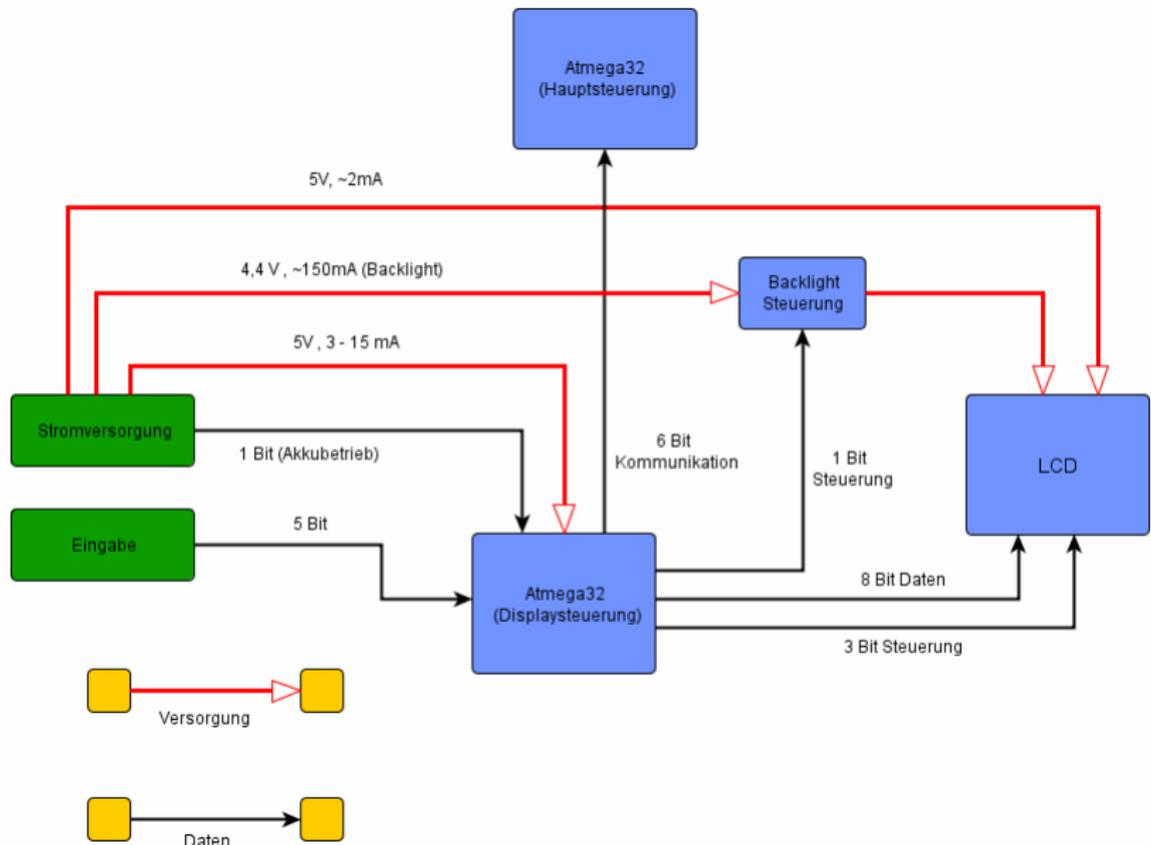
- Blockschaltbild unserer Gruppe
- LCD-Screen-Anschlüsse
- How to connect: LCD-ATmega32
- Control and Display-Commands

## 3 AVR-GCC-Anwendung

- Wiederholung der Grundoperationen
- AVR-GCC-Beispiel
- LCD-Commands

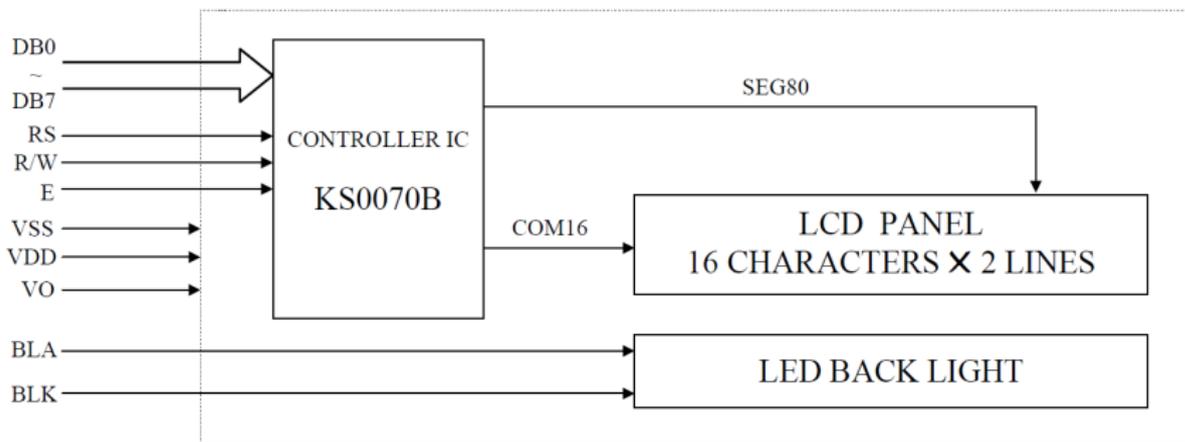
## 4 Quellenangabe, Schlusswort

# Blockschaltbild unserer Gruppe



# LCD-Screen-Anschlüsse

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
VSS	VDD	VO	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	BLA	BLK



# LCD-Screen-Anschlüsse

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
VSS	VDD	VO	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	BLA	BLK

Abbildung: Pinbelegung

# LCD-Screen-Anschlüsse

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
VSS	VDD	VO	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	BLA	BLK

Abbildung: Pinbelegung

# LCD-Screen-Anschlüsse

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
VSS	VDD	VO	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	BLA	BLK

Abbildung: Pinbelegung

## Stromversorgung (übliche Verschaltung)

- VSS= GND
- VDD= VCC = 5 V
- VEE = VO= GND  
(Kontrast-Poti)
- BLA / BLK =  
Anode/Kathode BL

# LCD-Screen-Anschlüsse

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
VSS	VDD	VO	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	BLA	BLK

Abbildung: Pinbelegung

## Stromversorgung (übliche Verschaltung)

- VSS= GND
- VDD= VCC = 5 V
- VEE = VO= GND (Kontrast-Poti)
- BLA / BLK = Anode/Kathode BL

## Steuerleitungen

- RS= Register Select
- RW = Read/Write
- E = enable
- DB0 - DB7 = Datenleitungen

# How to connect: LCD - ATmega32

# How to connect: LCD - ATmega32

## Wahl des Betriebsmodus

- 4- Bit Modus
  - Vorteil: nur ein Port des uC belegt
  - Nachteil: Nibble-Übertragung
- 8- Bit Modus
  - Vorteil: Byte-Übertragung
  - Nachteil: mehr als 1 Port des uC belegt

# How to connect: LCD - ATmega32

## Wahl des Betriebsmodus

- 4- Bit Modus
  - Vorteil: nur ein Port des uC belegt
  - Nachteil: Nibble-Übertragung
- 8- Bit Modus
  - Vorteil: Byte-Übertragung
  - Nachteil: mehr als 1 Port des uC belegt

## Initialisierung des Displays

Da wir den 4-Bit Modus ( 8.Bit=Standarteinstellung) nutzen wollen, muss das Display darauf eingestellt werden.

- Verwendung der Control and Display-Commands

# How to connect: LCD - ATmega32 (4-Bit Modus)

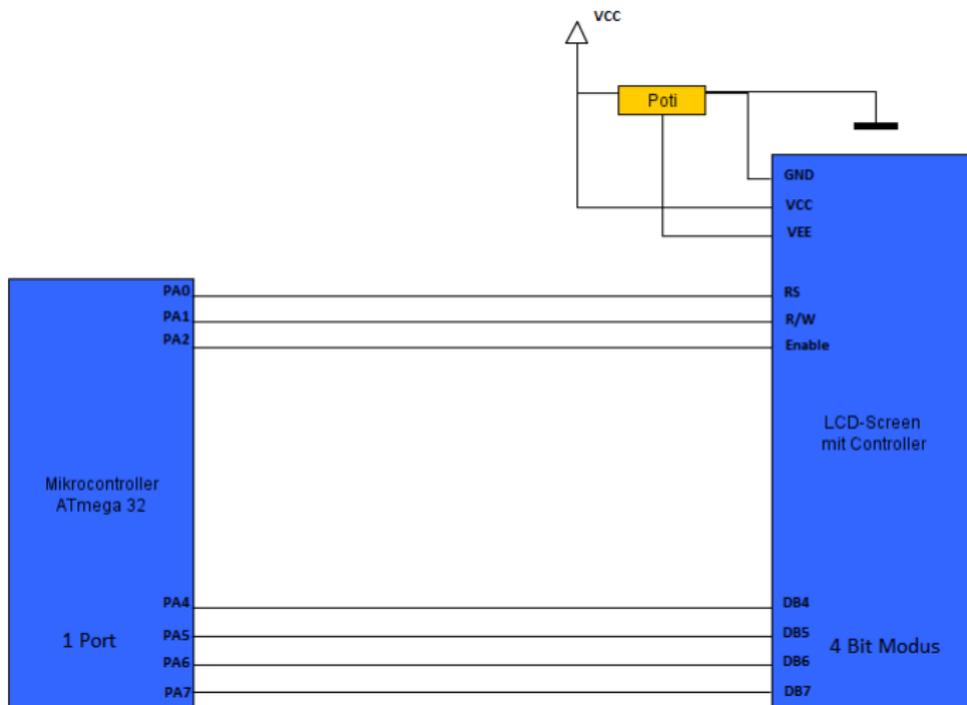


Abbildung: Connection: uC - LCD

# Control and Display-Commands

## Definition

- sind vordefinierte kombinierte Signale an den Pins
- werden vom KS0070B interpretiert

Befehl	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Ausführungszeit bei $f_{osc}=250kHz$
Clear Display	0	0	0	0	0	0	0	0	0	1	1.64mS
Cursor Home	0	0	0	0	0	0	0	0	1	*	1.64mS
Entry mode set	0	0	0	0	0	0	1	I/D	S		40uS
Display on/off Control	0	0	0	0	0	1	D	C	B		40uS
Cursor/Display shift	0	0	0	0	1	S/C	R/L	*	*		40uS
Function set	0	0	0	1	DL	N	F	*	*		40uS
Set CGRAM Address	0	0	0	1	CGRAM Adresse						40uS
Set DDRAM Address	0	0	1	DDRAM Adresse						40uS	
Read busy-flag and address counter	0	1	BF	CGRAM / DDRAM Adresse						0uS	
Write to CGRAM or DDRAM	1	0	Schreib Daten						40uS		
read from CGRAM or DDRAM	1	1	Lese Daten						40uS		

Abbildung: Grundbefehle mit Verarbeitungszeit

# Control and Display-Commands

# Control and Display-Commands

## Cursor Home

setzt den Cursor (an dem geschrieben wird) auf die erste Zeile/erste Spalte

# Control and Display-Commands

## Cursor Home

setzt den Cursor (an dem geschrieben wird) auf die erste Zeile/erste Spalte

## Entry mode set

setzt Cursorrichtung, Schiebeoperation

# Control and Display-Commands

## Cursor Home

setzt den Cursor (an dem geschrieben wird) auf die erste Zeile/erste Spalte

## Entry mode set

setzt Cursorrichtung, Schiebeoperation

## Clear Display

löscht Display und setzt Cursor auf erste Zeile/erste Spalte

# Control and Display-Commands

Bit Name	Einstellung / Status	
I/D	0 = Erniedrige Cursor Position	1 = Erhöhe Cursor Position
S	0 = Display nicht schieben	1 = Display schieben
D	0 = Display aus	1 = Display an
C	0 = Cursor aus	1 = Cursor an
B	0 = Cursor blinken aus	1 = Cursor blinken an
S/C	0 = Bewege Cursor	1 = Schiebe Display
R/L	0 = Schiebe nach links	1 = Schiebe nach rechts
DL	0 = 4-bit Interface	1 = 8-bit Interface
N	0 = 1/8 oder 1/11 Duty (1 Zeile)	1 = 1/16 Duty (2 Zeilen)
F	0 = 5x7 Punkte	1 = 5x10 Punkte
BF	0 = Befehle werden akzeptiert	1 = Interne Operation wird ausgeführt

Abbildung: Bitfunktionalitäten

# Control and Display-Commands

## Initialisierungsarten

- Initialisierung durch internen Reset Circuit
- Initialisierung durch Instruktion

# Control and Display-Commands

## Ablauf der Initialisierung durch Instruktion

- 3 x den Init-Befehl mit mehr als circa 5 ms warten
- Function-Set-Befehl mit N=1, F=0, DL=0 für 4.Bit Modus (2.Zeilen + 5x7 Font)
- weitere Einstellungen für Display on, Display Clear, etc.

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	1	*	*	*	*

Abbildung: Pinbelegung für Init

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	DL	N	F	*	*

Abbildung: Function-Set für 4.Bit

## 1 Allgemeines

- Was ist ein LCD-Screen?
- LCD 162C LED

## 2 Beschaltung mit Mikrocontroller

- Blockschaltbild unserer Gruppe
- LCD-Screen-Anschlüsse
- How to connect: LCD-ATmega32
- Control and Display-Commands

## 3 AVR-GCC-Anwendung

- Wiederholung der Grundoperationen
- AVR-GCC-Beispiel
- LCD-Commands

## 4 Quellenangabe, Schlusswort

# Bitoperationen und Konstanten

# Bitoperationen und Konstanten

## Operatoren

# Bitoperationen und Konstanten

## Operatoren

- << Links schieben

# Bitoperationen und Konstanten

## Operatoren

- << Links schieben
- >> Rechts schieben

# Bitoperationen und Konstanten

## Operatoren

- << Links schieben
- >> Rechts schieben
- | binäres Oder

# Bitoperationen und Konstanten

## Operatoren

- << Links schieben
- >> Rechts schieben
- | binäres Oder
- & binäres Und

# Bitoperationen und Konstanten

## Operatoren

- << Links schieben
- >> Rechts schieben
- | binäres Oder
- & binäres Und
- ^binäres XOR

# Bitoperationen und Konstanten

## Operatoren

- << Links schieben
- >> Rechts schieben
- | binäres Oder
- & binäres Und
- ^binäres XOR

## Definition von Konstanten:

# Bitoperationen und Konstanten

## Operatoren

- << Links schieben
- >> Rechts schieben
- | binäres Oder
- & binäres Und
- ^binäres XOR

## Definition von Konstanten:

- `#define LCD_CLEAR_DISPLAY 0x01`

# Bitoperationen und Konstanten

## Operatoren

- << Links schieben
- >> Rechts schieben
- | binäres Oder
- & binäres Und
- ^binäres XOR

## Definition von Konstanten:

- `#define LCD_CLEAR_DISPLAY 0x01`
- `const int 42`

# I/O Register

# I/O Register

## I/O Register

<b>DDRx</b>	DATA-Direction Register (1=Ausgang, 0=Eingang)
<b>PINx</b>	Eingangsregister (1=High, 0=Low)
<b>PORTx</b>	Ausgangsregister

# Beispiel im 4-Bit Modus

## !Vereinfachtes! Beispiel der Übertragung

z.B. 0b00100101 soll an PORT A an den LCD-Screen übertragen werden

```
...  
DDRA=0xFF;  
// Ausgaenge aktiv  
// PORTA=0xFF , ist auf Null initialisiert  
PORTA|=(1<<PA5);  
//PORTA=0b00100000 , upper Nibble uebertragen  
PORTA&=~(1<<PA5);  
//PORTA=0b00000000 , zuruecksetzen  
PORTA=(1<<PA4)|(1<<PA6);  
//PORTA=0b01010000 , lower Nibble uebertragen  
...
```

# ASCII-Code to LCD

Wieder das Beispiel von vorhin:  
0b00100101 soll an PORT A an den  
LCD-Screen übertragen werden. Was  
verbirgt sich dahinter?

# ASCII-Code to LCD

upper 4 bit	lower 4 bit	0000	0010	0011	0100
0000	CG RAM (1)				
0001	(2)				
0010	(3)				
0011	(4)				
0100	(5)				
0101	(6)				

Wieder das Beispiel von vorhin:  
0b00100101 soll an PORT A an den  
LCD-Screen übertragen werden. Was  
verbirgt sich dahinter?

Abbildung: Ascii-Tabelle

# Beispiel im 4-Bit Modus

# Beispiel im 4-Bit Modus

## Problem der Vereinfachung

- Initialisierungszeiten des KS0070B nicht berücksichtigt (siehe Tab.)
- enable, RS, R/W Leitung nicht gesetzt
- busyFlag nicht überprüft
- statische Ausgabe
- Welche Zeile, welche Spalte ?

# HW-Beispiel mit ATMEL-Studio 6

Beispiel ist im Anhang zu finden!!

# LCD-Commands

## LCD-Befehle

- vereinfachen wesentlich den wiederholten Aufruf von Daten/Befehlen

```
...
#define LCD_EN          PA2
#define LCD_ENABLE_US  20
...
static void lcd_enable( void )
{
    PORTA |= (1<<LCD_EN);    // Enable auf 1 setzen
    _delay_us( LCD_ENABLE_US ); // kurze Pause
    PORTA &= ~(1<<LCD_EN);  // Enable auf 0 setzen
}
...
```

## 1 Allgemeines

- Was ist ein LCD-Screen?
- LCD 162C LED

## 2 Beschaltung mit Mikrocontroller

- Blockschaltbild unserer Gruppe
- LCD-Screen-Anschlüsse
- How to connect: LCD-ATmega32
- Control and Display-Commands

## 3 AVR-GCC-Anwendung

- Wiederholung der Grundoperationen
- AVR-GCC-Beispiel
- LCD-Commands

## 4 Quellenangabe, Schlusswort

# Quellenangabe

- <http://www.mikrocontroller.net/articles/AVR-GCC-Tutorial/LCD-Ansteuerung>
- <http://de.wikipedia.org/wiki/Flüssigkristallanzeige>
- [http://www.rn-wissen.de/index.php/LCD-Modul\\_am\\_AVR](http://www.rn-wissen.de/index.php/LCD-Modul_am_AVR)
- Projektlaboreinführung in Mikrocontroller
- <http://www.reichelt.de/>
- Datasheets: KS0070B, ATmega\_32, LCD 162C LED

Vielen Dank für ihre Aufmerksamkeit!