LCD library for AVR uC

00.00.01

Generated by Doxygen 1.8.2

Sun Nov 11 2012 19:12:33

Contents

1 Main Page

1.1 Project

LCD functions based on the HD4470 / KS0066u chips

Author

Patrice Nadeau
email : patricen@telwarwick.net
www : http://nadeaup.homeip.net:8080/

1.2 Specifications

1.2.1 Language

- C

1.2.2 Target

- ATmega48
- ATmega88
- ATmega168
- ATmega328

1.2.3 Tested

- ATmega168

1.3 Files

- lcd.h
- lcd.c

1.4 Dependency

- utils.h

1.5 More informations

Datasheet Datasheet Newhaven Display (NHD-0216BZ-RN-YBW) March 6, 2008

1.6 Disclaimer

2 Todo List

File lcd.h

utils.h, delay.h should be replaced

Global lcd_DisplayOnOff (uint8_t display, uint8_t cursor, uint8_t blink)

Optimise the code

Global lcd_EnterMode (uint8_t rightleft, uint8_t shift)

Optimise the code

Global lcd_FunctionSet (uint8_t line, uint8_t font)

use of a case to suport 4 lines

use of a case to suport other fonts

3 Bug List

Global lcd_BusyFlag (void)

Can't work, _Send alway return 0

need to be modified for 4 bits

Global lcd_ReadData (void)

Can't work, _Send alway return 0

4 File Documentation

4.1 lcd.c File Reference

LCD on AVR.

Functions

- static void **_Clock** (void)
- static uint8_t **_Send** (uint8_t command)
- static void **_SendData** (uint8_t data)
- static void **_SendCommand** (uint8_t command)
- void lcd_InitPort (void)

  Initialize the data and command register for output.
- void lcd_FunctionSet (uint8_t line, uint8_t font)

Set the number of lines and font.

- void lcd_ClearDisplay (void)

  Clear the display and set DDRA to 20H and DDRAM to 00h.
- void lcd_EnterMode (uint8_t rightleft, uint8_t shift)

  lcd_EnterMode
- void lcd_DisplayOnOff (uint8_t display, uint8_t cursor, uint8_t blink)

  Turn on/off the display, cursor and blink of the LCD.
- void lcd_DDRAM (uint8_t position)

  Place the cursor at position in display memory.
- void lcd_CGRAM (uint8_t address)

  Set the GCRAM address.
- void lcd_WriteData (uint8_t character)

  Display character on the lcd.
- uint8_t **lcdShift** (uint8_t cursor_display, uint8_t direction)
- void lcd_WriteString (char *string)

  Write a string to the LCD.
- void **lcdReturnHome** (void)
- uint8_t lcd_BusyFlag (void)

  Return the busy flag and the address counter.
- uint8_t lcd_ReadData (void)

  Read the 8 bit data from position set earlier by lcdCGRAM or lcdDDRAM.
- void lcd_ClearLine (uint8_t line)

  Clear every character on line and place cursor at beginning of line.
- void lcd_Byte (uint8_t data)

  Display the binary version of a byte.
- void lcd_DoubleByte (uint16_t data)

  Display the binary version of 16 bits.

4.1.1 Detailed Description

LCD on AVR.

Author

Patrice Nadeau

See Also

lcd.h

4.1.2 Function Documentation

4.1.2.1 uint8_t lcd_BusyFlag (void)

Return the busy flag and the address counter.

Returns

Busy flag, address counter

- bit **XXXXX** is busy flag () 1 busy, 0 rdy to accept command
- bit 6-0 is the address counter

Bug Can't work, _Send alway return 0

need to be modified for 4 bits

4.1.2.2 void lcd_Byte (uint8_t *data*)

Display the binary version of a byte.

Parameters

in	*data*	Byte to display

Note

Mostly for debugging

4.1.2.3 void lcd_CGRAM (uint8_t *address*)

Set the GCRAM address.

Parameters

in	*address*	(last 6 bits)

4.1.2.4 void lcd_ClearLine (uint8_t *line*)

Clear every character on line and place cursor at beginning of line.

Parameters

in	*line*	The line numer to clear • lcd_LINE_x

4.1.2.5 void lcd_DDRAM (uint8_t *position*)

Place the cursor at *position* in display memory.

Parameters

in	*position*	Position in memory

Note

First line 00H to 27H, second line 40H to 67H

4.1.2.6 void lcd_DisplayOnOff (uint8_t *display,* uint8_t *cursor,* uint8_t *blink*)

Turn on/off the display, cursor and blink of the LCD.

Todo Optimise the code

4.1.2.7 void lcd_DoubleByte (uint16_t *data*)

Display the binary version of 16 bits.

Parameters

in	*data*	Double byte to display

Note

Mostly for debugging

4.1.2.8 void lcd_EnterMode (uint8_t *rightleft,* uint8_t *shift*)

lcd_EnterMode

Todo Optimise the code

4.1.2.9 void lcd_FunctionSet (uint8_t *line,* uint8_t *font*)

Set the number of *lines* and *font*.

Note

Only support 2 lines

Todo use of a case to suport 4 lines

Todo use of a case to suport other fonts

4.1.2.10 void lcd_InitPort (void)

Initialize the data and command register for output.

Returns

Note

Maybe should provide software reset

4.1.2.11 uint8_t lcd_ReadData (void)

Read the 8 bit data from position set earlier by lcdCGRAM or lcdDDRAM.

Returns

uint8_t Data

Note

Address counter will also be affected (like a write)

Bug Can't work, _Send alway return 0

4.1.2.12 void lcd_WriteData (uint8_t *character*)

Display *character* on the lcd.

Parameters

in	*character*	Character to display

Note

Works for the first 127 ASCII codes

4.1.2.13 void lcd_WriteString (char * *string*)

Write a string to the LCD.

Parameters

in	*string	Pointer to a string

Note

Do not check the length

4.2 lcd.h File Reference

LCD on AVR.

Macros

For the dusplay functions

- #define lcd_LINES_1 1
  1 line
- #define lcd_LINES_2 2
  2 lines
- #define lcd_LINES_4 4
  Not implemented yet.

- #define lcd_FONT_8 8
  5 x 8 font
- #define lcd_FONT_11 11
  8 x 11
- #define lcd_SHIFT_DISPLAY_OFF 0
  Shift the whole display.
- #define lcd_SHIFT_DISPLAY_RIGHTLEFT 0
  Direction to shift the display if enable.
- #define lcd_DISPLAY_OFF 0
  1 line
- #define lcd_CURSOR_OFF 0
  Use !lcd_CURSOR_OFF to enable.
- #define lcd_CURSOR_BLINK_OFF 0
  Use !lcd_CURSOR_BLINK_OFF to enable.

Functions

- void lcd_InitPort (void)
  Initialize the data and command register for output.
- void lcd_FunctionSet (uint8_t line, uint8_t font)
  Set the number of lines and font.
- void lcd_ClearDisplay (void)
  Clear the display and set DDRA to 20H and DDRAM to 00h.
- void lcd_EnterMode (uint8_t rightleft, uint8_t shift)
  lcd_EnterMode
- void lcd_DisplayOnOff (uint8_t display, uint8_t cursor, uint8_t blink)
  Turn on/off the display, cursor and blink of the LCD.
- void lcd_DDRAM (uint8_t position)
  Place the cursor at position in display memory.
- void lcd_WriteData (uint8_t character)
  Display character on the lcd.
- void lcd_CGRAM (uint8_t address)
  Set the GCRAM address.
- uint8_t lcd_Shift (uint8_t wich, uint8_t direction)
  Select which display or cursor move and to the right or left.
- void lcd_WriteString (char *string)
  Write a string to the LCD.
- void lcd_ReturnHome (void)
  Return cursor at 00H, return display original status if shifted.
- uint8_t lcd_BusyFlag (void)
  Return the busy flag and the address counter.
- uint8_t lcd_ReadData (void)
  Read the 8 bit data from position set earlier by lcdCGRAM or lcdDDRAM.
- void lcd_ClearLine (uint8_t line)
  Clear every character on line and place cursor at beginning of line.
- void lcd_Byte (uint8_t data)
  Display the binary version of a byte.
- void lcd_DoubleByte (uint16_t data)
  Display the binary version of 16 bits.

4.2.1 Detailed Description

LCD on AVR. LCD functions based on the HD4470 / KS0066u chips (Datasheet Newhaven Display (NHD-0216BZ-RN-YBW) March 6, 2008)

It take for granted that the Data pins on the LCD and on the AVR are in the same order and all on the same port

Author

Patrice Nadeau

Note

For ATmega48PA/88PA/168PA/328P

Todo utils.h, delay.h should be replaced

4.2.2 Function Documentation

4.2.2.1 uint8_t lcd_BusyFlag (void)

Return the busy flag and the address counter.

Returns

Busy flag, address counter

- bit **XXXXX** is busy flag () 1 busy, 0 rdy to accept command
- bit 6-0 is the address counter

Bug Can't work, _Send alway return 0

need to be modified for 4 bits

4.2.2.2 void lcd_Byte (uint8_t *data*)

Display the binary version of a byte.

Parameters

in	*data*	Byte to display

Note

Mostly for debugging

4.2.2.3 void lcd_CGRAM (uint8_t *address*)

Set the GCRAM address.

Parameters

in	*address*	(last 6 bits)

4.2.2.4 void lcd_ClearLine (uint8_t *line*)

Clear every character on line and place cursor at beginning of line.

Parameters

in	*line*	The line numer to clear • lcd_LINE_x

4.2.2.5 void lcd_DDRAM (uint8_t *position*)

Place the cursor at *position* in display memory.

Parameters

in	*position*	Position in memory

Note

First line 00H to 27H, second line 40H to 67H

4.2.2.6 void lcd_DisplayOnOff (uint8_t *display,* uint8_t *cursor,* uint8_t *blink*)

Turn on/off the display, cursor and blink of the LCD.

Parameters

in	*display*	Show the display • lcd_DISPLAY_OFF • !lcd_DISPLAY_OFF
in	*cursor*	Show the cursor • lcd_CURSOR_OFF • !lcd_CURSOR_OFF
in	*blink*	Blink the cursor • lcd_CURSOR_BLINK_OFF • !lcd_CURSOR_BLINK_OFF

Todo Optimise the code

4.2.2.7 void lcd_DoubleByte (uint16_t *data*)

Display the binary version of 16 bits.

Parameters

in	*data*	Double byte to display

Note

Mostly for debugging

4.2.2.8 void lcd_EnterMode (uint8_t *rightleft*, uint8_t *shift*)

lcd_EnterMode

Parameters

in	*rightleft*	Direction to shift the display • lcd_SHIFT_DISPLAY_RIGHTLEFT • !lcd_SHIFT_DISPLAY_RIGHTLEFT
in	*shift*	Show the dispplay • lcd_SHIFT_DISPLAY_OFF • !lcd_SHIFT_DISPLAY_OFF

Todo Optimise the code

4.2.2.9 void lcd_FunctionSet (uint8_t *line*, uint8_t *font*)

Set the number of *lines* and *font*.

Parameters

in	*line*	Number of lines • lcd_LINES_x
in	*font*	Font to use • lcd_FONT_x

Note

Only support 2 lines

Todo use of a case to suport 4 lines

Todo use of a case to suport other fonts

4.2.2.10 void lcd_InitPort (void)

Initialize the data and command register for output.

Returns

Note

Maybe should provide software reset

4.2.2.11 uint8_t lcd_ReadData (void)

Read the 8 bit data from position set earlier by lcdCGRAM or lcdDDRAM.

Returns

uint8_t Data

Note

Address counter will also be affected (like a write)

Bug Can't work, _Send alway return 0

4.2.2.12 void lcd_ReturnHome (void)

Return cursor at 00H, return display original status if shifted.

Note

Does not erase DDRAM
Unshift if shifted

4.2.2.13 uint8_t lcd_Shift (uint8_t *wich*, uint8_t *direction*)

Select which display or cursor move and to the right or left.

Parameters

in	*wich*	Cursor or dislplay
in	*direction*	Direction to shift

Returns

Error code

Return values

0	successful
1	wrong cursor/display option
2	wrong direction

4.2.2.14 void lcd_WriteData (uint8_t *character*)

Display *character* on the lcd.

Parameters

in	*character*	Character to display

Note

Works for the first 127 ASCII codes

4.2.2.15 void lcd_WriteString (char * *string*)

Write a string to the LCD.

Parameters

in	*string	Pointer to a string

Note

Do not check the length

Index