AlphaLCD helper for Arduino

1.0.4

Generated by Doxygen 1.8.6

Tue Jan 27 2015 20:28:26

Contents

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Chapter 4

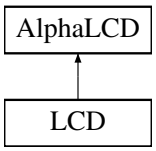
Class Documentation

4.1 LCD Class Reference

Manages the Alphanumeric display for program output messages.

`#include "LCD.h"`

Inheritance diagram for LCD:



Public Member Functions

- LCD ()
- ~LCD ()
- void enable (bool s)

  Set the display on or off.
- void blink (bool set)

  Set blink mode.
- void error (String m)

  shows an error message
- void error (String m, int x, int y)

  shows an error message at specified coordinates
- void message (String m)

  shows a string message
- void message (String m, int x, int y)

  shows a string message at specified coordinates
- void clean ()

  clean the LCD screen
- void dec (int n)

  shows an integer in decimal format
- void hex (int n)

  shows an integer in hexadeciaml format
- void bin (int n)

  shows an integer in binary format

- void oct (int n)

  shows an integer in octal format
- void welcome ()

  shows the program welcome message
- void menu (String sect1, String sect2, String sect3, String sect4)

Private Member Functions

- LCD (const LCD &c)
- LCD & operator= (const LCD &c)

Private Attributes

- AlphaLCD lcd

  AlphaLCD class inherited instance.

4.1.1 Detailed Description

Manages the Alphanumeric display for program output messages.

This class implements the *AlphaLCD* class that manages the Alphanumeric LCD display hardware using three digital Arduino pins via a shift-out register.

Definition at line 56 of file LCD.h.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 LCD::LCD ( )

4.1.2.2 LCD::~LCD ( )

4.1.2.3 LCD::LCD (const LCD & *c*) [private]

4.1.3 Member Function Documentation

4.1.3.1 void LCD::bin (int *n*)

shows an integer in binary format

4.1.3.2 void LCD::blink (bool *set*)

Set blink mode.

4.1.3.3 void LCD::clean ( )

clean the LCD screen

4.1.3.4 void LCD::dec (int *n*)

shows an integer in decimal format

4.1.3.5 void LCD::enable (bool *s*)

Set the display on or off.

4.1.3.6 void LCD::error (String *m*)

shows an error message

4.1.3.7 void LCD::error (String *m*, int *x*, int *y*)

shows an error message at specified coordinates

4.1.3.8 void LCD::hex (int *n*)

shows an integer in hexadeciaml format

4.1.3.9 void LCD::menu (String *sect1*, String *sect2*, String *sect3*, String *sect4*)

4.1.3.10 void LCD::message (String *m*)

shows a string message

4.1.3.11 void LCD::message (String *m*, int *x*, int *y*)

shows a string message at specified coordinates

4.1.3.12 void LCD::oct (int *n*)

shows an integer in octal format

4.1.3.13 LCD& LCD::operator= (const LCD & *c*) [private]

4.1.3.14 void LCD::welcome ( )

shows the program welcome message

4.1.4 Member Data Documentation

4.1.4.1 AlphaLCD LCD::lcd [private]

AlphaLCD class inherited instance.

Definition at line 61 of file LCD.h.

The documentation for this class was generated from the following file:

- /Volumes/John Doe/Firmware/Arduino Sketches and components/Display/Display_Example/LCD.h

Chapter 5

File Documentation

5.1 /Volumes/John Doe/Firmware/Arduino Sketches and components/Display/Display_-Example/Display_Example.ino File Reference

```
#include <AlphaLCD.h>
#include <Streaming.h>
#include "LCD.h"
#include "Strings.h"
#include "Version.h"
```

Functions

- AlphaLCD lcd (2, 3, 4)

  AlphaLCD class instance for display hardware control.
- void setup ()
- void loop (void)
- void welcome ()

  Welcome message shown at device power-on.
- void message (String m)

  Display a string on the LCD at the cursor position.
- void dec (int n)

  Display an integer value in decimal format at the cursor position.
- void hex (int n)

  Display an integer value in hex format at the cursor position.
- void error (String m, int x, int y)

  Display an error message at the specified cursor coordinates.
- void error (String m)

  Display an error message at the cursor position.
- void message (String m, int x, int y)

  Display a string on the LCD at the specified cursor coordinates.
- void clean ()

  Clean the display.
- void menu (String sect1, String sect2, String sect3, String sect4)

  Creates a menu screen.

5.1.1 Function Documentation

5.1.1.1 void clean ( )

Clean the display.

A delay of 100 ms is added after the hardware clear() call to give the display the time to complete the operation.

Definition at line 135 of file Display_Example.ino.

References lcd(), and LCDCLEAR_DELAY.

Referenced by menu().

```
135             {
136     lcd.clear();
137     delay(LCDCLEAR_DELAY);
138 }
```

5.1.1.2 void dec (int *n*)

Display an integer value in decimal format at the cursor position.

Parameters

n	the integer to show in decimal format

Definition at line 71 of file Display_Example.ino.

References lcd().

```
71                 {
72     lcd.print(n, DEC);
73 }
```

5.1.1.3 void error (String *m*, int *x*, int *y*)

Display an error message at the specified cursor coordinates.

The error message is shown for a LCDERROR_DELAY milliseconds. After the timeout expires the screen is not cleared so the next steps should be managed by the program flow. It is expected that error messages are shown in a calling code that manages the error conditions.

Parameters

m	the message string
x	the cursor column zero based
y	the row number zero based

Definition at line 97 of file Display_Example.ino.

References LCDERROR_DELAY, and message().

```
97                                      {
98     message(m, x, y);
99     delay(LCDERROR_DELAY);
100 }
```

5.1.1.4 void error (String *m*)

Display an error message at the cursor position.

The error message is shown for a LCDERROR_DELAY milliseconds. After the timeout expires the screen is not cleared so the next steps should be managed by the program flow. It is expected that error messages are shown in a calling code that manages the error conditions.

Parameters

m	the string message

Definition at line 112 of file Display_Example.ino.

References LCDERROR_DELAY, and message().

```
112                     {
113     message(m);
114     delay(LCDERROR_DELAY);
115 }
```

5.1.1.5 void hex (int *n*)

Display an integer value in hex format at the cursor position.

Parameters

n	the integer to show in hexadecimal format

Definition at line 80 of file Display_Example.ino.

References lcd().

```
80                 {
81     lcd.print("0x");
82         lcd.print(n, HEX);
83 }
```

5.1.1.6 AlphaLCD lcd (2 , 3 , 4)

AlphaLCD class instance for display hardware control.

Referenced by clean(), dec(), hex(), message(), setup(), and welcome().

5.1.1.7 void loop (void)

Definition at line 29 of file Display_Example.ino.

```
30 {
31
32   // ...
33
34 }
```

5.1.1.8 void menu (String *sect1*, String *sect2*, String *sect3*, String *sect4*)

Creates a menu screen.

The 2-lines LCD screen is divided in four sectors, that can be used or not. The LCD sectors length and position are defined and based on the LCD size. Every sector is filled with one of the four parameters string. Sectors 1 & 2 are in the top row, sectors 3 & 4 in the bottom row

Parameters

sect1	The Upper Left display sector

sect2	The Upper Right display sector
sect3	The Lower Left display sector
sect4	The Lower Right display sector

Definition at line 153 of file Display_Example.ino.

References clean(), LCD_SECTOR1, LCD_SECTOR2, LCD_SECTOR3, LCD_SECTOR4, LCDBOTTOMROW, LCDTOPROW, and message().

```
153                                                                 {
154     clean();
155     message(sect1, LCD_SECTOR1, LCDTOPROW);
156     message(sect2, LCD_SECTOR2, LCDTOPROW);
157     message(sect3, LCD_SECTOR3, LCDBOTTOMROW);
158     message(sect4, LCD_SECTOR4, LCDBOTTOMROW);
159 }
```

5.1.1.9 void message (String *m*)

Display a string on the LCD at the cursor position.

Parameters

m	the message string

Definition at line 62 of file Display_Example.ino.

References lcd().

Referenced by error(), menu(), and message().

```
62                          {
63     lcd.print(m);
64 }
```

5.1.1.10 void message (String *m*, int *x*, int *y*)

Display a string on the LCD at the specified cursor coordinates.

Parameters

m	the string message
x	the cursor column zero based
y	the row number zero based

Definition at line 124 of file Display_Example.ino.

References lcd(), and message().

```
124                                         {
125     lcd.setCursor(x, y);
126     message(m);
127 }
```

5.1.1.11 void setup ( )

Definition at line 15 of file Display_Example.ino.

References lcd(), LCDCHARS, LCDROWS, and welcome().

```
16 {
17
18     // Initializes the LCD library
19     lcd.begin(LCDCHARS, LCDROWS);
```

```
20
21      // Turn LCD On
22      lcd.display();
23
24      welcome();
25
26  }
```

5.1.1.12 void welcome ( )

Welcome message shown at device power-on.

Definition at line 42 of file Display_Example.ino.

References _SPACING, _VERSION, lcd(), LCDBOTTOMROW, LCDMESSAGE_DELAY, LCDTOPROW, and version.

Referenced by setup().

```
42                {
43
44      lcd.clear();
45      lcd.setCursor(0, LCDTOPROW);
46      lcd << "Electrosch...";
47      delay(LCDMESSAGE_DELAY);
48      lcd.clear();
49
50      lcd.setCursor(0, LCDBOTTOMROW);
51          lcd << _VERSION << _SPACING << version();
52      delay(LCDMESSAGE_DELAY);
53      lcd.clear();
54
55  }
```

5.2 /Volumes/John Doe/Firmware/Arduino Sketches and components/Display/Display_Example/LCD.h File Reference

LCD display Manager include file.

```
#include <inttypes.h>
#include <Print.h>
#include <AlphaLCD.h>
#include <Streaming.h>
```

Classes

- class LCD

  Manages the Alphanumeric display for program output messages.

Macros

- #define LCDclockPin 2

  LCD Shift control pin - Clock signal Define this value accordingly with the Arduino board connections.
- #define LCDlatchPin 3

  LCD Shift control pin - Latch signal Define this value accordingly with the Arduino board connections.
- #define LCDdataPin 4

  LCD Shift control pin - Data signal Define this value accordingly with the Arduino board connections.
- #define LCDCHARS 16

  Display characters per line Define this value accordingly with the LCD Hardware datasheet.

- #define LCDROWS 2

  Display rows.
- #define LCDTOPROW 0

  The top row number of the LCD.
- #define LCDBOTTOMROW 1

  The bottom row number of the LCD.
- #define LCD_SECTOR1 0

  Top Left display sector column.
- #define LCD_SECTOR2 LCDCHARS / 2

  Top Right display sector column.
- #define LCD_SECTOR3 0

  Bottom Left display sector column.
- #define LCD_SECTOR4 LCDCHARS / 2

  Bottom Right display sector column.
- #define LCDERROR_DELAY 3000

  Delay after showing an error.
- #define LCDMESSAGE_DELAY 2000

  Delay after showing a temporary message e.g. the welcome screen.
- #define LCDCLEAR_DELAY 100

  Delay after a clear display call to hardware has been done.

5.2.1 Detailed Description

LCD display Manager include file. Methods to manage the LCD output and display features, including some hard-coded strings like the welcome message.

Definition in file LCD.h.

5.2.2 Macro Definition Documentation

5.2.2.1 #define LCD_SECTOR1 0

Top Left display sector column.

Definition at line 35 of file LCD.h.

Referenced by menu().

5.2.2.2 #define LCD_SECTOR2 LCDCHARS / 2

Top Right display sector column.

Definition at line 37 of file LCD.h.

Referenced by menu().

5.2.2.3 #define LCD_SECTOR3 0

Bottom Left display sector column.

Definition at line 39 of file LCD.h.

Referenced by menu().

5.2.2.4 #define LCD_SECTOR4 LCDCHARS / 2

Bottom Right display sector column.

Definition at line 41 of file LCD.h.

Referenced by menu().

5.2.2.5 #define LCDBOTTOMROW 1

The bottom row number of the LCD.

Definition at line 33 of file LCD.h.

Referenced by menu(), and welcome().

5.2.2.6 #define LCDCHARS 16

Display characters per line Define this value accordingly with the LCD Hardware datasheet.

Definition at line 27 of file LCD.h.

Referenced by setup().

5.2.2.7 #define LCDCLEAR_DELAY 100

Delay after a clear display call to hardware has been done.

Definition at line 48 of file LCD.h.

Referenced by clean().

5.2.2.8 #define LCDclockPin 2

LCD Shift control pin - Clock signal Define this value accordingly with the Arduino board connections.

Definition at line 18 of file LCD.h.

5.2.2.9 #define LCDdataPin 4

LCD Shift control pin - Data signal Define this value accordingly with the Arduino board connections.

Definition at line 24 of file LCD.h.

5.2.2.10 #define LCDERROR_DELAY 3000

Delay after showing an error.

Definition at line 44 of file LCD.h.

Referenced by error().

5.2.2.11 #define LCDlatchPin 3

LCD Shift control pin - Latch signal Define this value accordingly with the Arduino board connections.

Definition at line 21 of file LCD.h.

5.2.2.12 #define LCDMESSAGE_DELAY 2000

Delay after showing a temporary message e.g. the welcome screen.

Definition at line 46 of file LCD.h.

Referenced by welcome().

5.2.2.13 #define LCDROWS 2

Display rows.

Definition at line 29 of file LCD.h.

Referenced by setup().

5.2.2.14 #define LCDTOPROW 0

The top row number of the LCD.

Definition at line 31 of file LCD.h.

Referenced by menu(), and welcome().

5.3 /Volumes/John Doe/Firmware/Arduino Sketches and components/Display/Display_-Example/Strings.h File Reference

LCD Display base strings.

```
#include "Version.h"
```

Macros

- #define _SPACING " "
- #define _EMPTY_HALF_LINE " "

  Half text line empty (based on 20 character lines LCD)
- #define _VERSION "V"
- #define _DIRECT_PHOTO "Photo"

  Direct function: Shoot a photo.
- #define _DIRECT_FOCUS "Focus"

  Direct function: Setup auto focus on the Camera.
- #define _DIRECT_BURST "Burst"

  Direct function: Burst a manual sequence.
- #define _MAIN_MESSAGE "'Info' last action"
- #define _MENU_MESSAGE "'Down' for items"
- #define _READY_MESSAGE "Select any option"
- #define _STANDBY_TIT "Standby"

  Standby status.
- #define _CHOICE_TIT "Ready"

  Choice from the menu or waiting for action.
- #define _RUNNING_TIT "Running"

  Function running.
- #define _PAUSED_TIT "Paused"

  Function paused.

5.3.1 Detailed Description

LCD Display base strings. The strings used to build the board interface.

Definition in file Strings.h.

5.3.2 Macro Definition Documentation

5.3.2.1 #define _CHOICE_TIT "Ready"

Choice from the menu or waiting for action.

Definition at line 27 of file Strings.h.

5.3.2.2 #define _DIRECT_BURST "Burst"

Direct function: Burst a manual sequence.

Definition at line 20 of file Strings.h.

5.3.2.3 #define _DIRECT_FOCUS "Focus"

Direct function: Setup auto focus on the Camera.

Definition at line 19 of file Strings.h.

5.3.2.4 #define _DIRECT_PHOTO "Photo"

Direct function: Shoot a photo.

Definition at line 18 of file Strings.h.

5.3.2.5 #define _EMPTY_HALF_LINE " "

Half text line empty (based on 20 character lines LCD)

Definition at line 14 of file Strings.h.

5.3.2.6 #define _MAIN_MESSAGE "'Info' last action"

Definition at line 22 of file Strings.h.

5.3.2.7 #define _MENU_MESSAGE "'Down' for items"

Definition at line 23 of file Strings.h.

5.3.2.8 #define _PAUSED_TIT "Paused"

Function paused.

Definition at line 29 of file Strings.h.

5.3.2.9 #define _READY_MESSAGE "Select any option"

Definition at line 24 of file Strings.h.

5.3.2.10 #define _RUNNING_TIT "Running"

Function running.

Definition at line 28 of file Strings.h.

5.3.2.11 #define _SPACING " "

Definition at line 13 of file Strings.h.

Referenced by welcome().

5.3.2.12 #define _STANDBY_TIT "Standby"

Standby status.

Definition at line 26 of file Strings.h.

5.3.2.13 #define _VERSION "V"

Definition at line 16 of file Strings.h.

Referenced by welcome().

5.4 /Volumes/John Doe/Firmware/Arduino Sketches and components/Display/Display_-Example/Version.h File Reference

Version and Build NumberHelper Class.

Macros

- #define build() "1.0.4 rc"

  Incremental build number.
- #define version() "1.1"

  Firmware version.
- #define project() "Electroschematics"

  Project name.

5.4.1 Detailed Description

Version and Build NumberHelper Class. This helper macros exposes the static methods to get the firmware version and the build number. Use the build() and version() metho5ds anywhere in the program including this file

Definition in file Version.h.

5.4.2 Macro Definition Documentation

5.4.2.1 #define build( ) "1.0.4 rc"

Incremental build number.

Definition at line 12 of file Version.h.

5.4.2.2 #define project( ) "Electroschematics"

Project name.

Definition at line 16 of file Version.h.

5.4.2.3 #define version( ) "1.1"

Firmware version.

Definition at line 14 of file Version.h.

Referenced by welcome().

Index