Educational Booster Pack Library Reference Manual

Generated by Doxygen 1.8.3.1

Contents

Chapter 1

Educational BoosterPack Library

Library for the Educational BoosterPack

Developed with `embedXcode`

Author

Rei VILO
embedXcode.weebly.com

Date

Feb 11, 2013

Version

106

Copyright

© Rei VILO, 2013
CC = BY NC SA

See Also

- Educational BoosterPack wiki
  `http://boosterpackdepot.com/wiki/index.php?title=Educational_Booster-Pack`
- Library for LCD
  based on Arduino library for Dog character LCD's
  `http://code.google.com/p/doglcd/`
  Copyright: 2010 Eberhard Fahle `e.fahle@wayoda.org`
  License: GNU Lesser GPL
- Library for ADXL335
  based on Accelerometer_FRAUNCHPAD library
  `http://github.com/energia/Energia/tree/master/examples/6.Sensors/-Accelerometer_FRAUNCHPAD`
  Copyright: Rei VILO, 2012
  Licence: CC = BY NC SA

- Library for RGB LED
  based on Stellar_Fading_RGB example
  http://github.com/energia/Energia/tree/master/examples/1.Basics/-
  Stellar_Fading_RGB
  Copyright: Rei VILO, 2012
  Licence: CC = BY NC SA

- sprintf
  http://www.cplusplus.com/reference/cstdio/printf/

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 documented files with brief descriptions:

Project EducationalBoosterPack_main
Developed with `embedXcode` 13

Chapter 4

Class Documentation

4.1 ebpACC Class Reference

Accelerometer on Educational BoosterPack.

```
#include <EducationalBoosterPack.h>
```

Public Member Functions

- ebpACC ()

    Constructor.

- void begin ()

    Initialisation.

- void calibrate ()

    Calibrate the accelerometer.

- void get ()

    Acquire acceleration.

- void gravityX100 (int32_t &x, int32_t &y, int32_t &z)

    Return gravity, X100 to avoid float.

- void degreeX100 (int32_t &x, int32_t &y, int32_t &z)

    Return angle in degrees, X100 to avoid float.

4.1.1 Detailed Description

Accelerometer on Educational BoosterPack.

4.1.2 Member Function Documentation

4.1.2.1 void ebpACC::calibrate ( )

Calibrate the accelerometer.

Note

   To perform a calibration, place the FraunchPad on a horizontal table

4.1.2.2 void ebpACC::degreeX100 (int32_t & *x,* int32_t & *y,* int32_t & *z*)

Return angle in degrees, X100 to avoid float.

```
Serial.print(x/10, DEC);      // integer part
Serial.print(".");            // decimal separator
Serial.print(x%100/10, DEC);  // decimal part, first digit
Serial.print(x%10, DEC);      // decimal part, second digit
```

Parameters

x	angle on x axis
y	angle on y axis
z	angle on z axis

4.1.2.3 void ebpACC::get ( )

Acquire acceleration.

Note

> Call this function prior to gravityX100 or degreeX100

4.1.2.4 void ebpACC::gravityX100 (int32_t & *x,* int32_t & *y,* int32_t & *z*)

Return gravity, X100 to avoid float.

```
Serial.print(x/10, DEC);      // integer part
Serial.print(".");            // decimal separator
Serial.print(x%100/10, DEC);  // decimal part, first digit
Serial.print(x%10, DEC);      // decimal part, second digit
```

Parameters

x	gravity on x axis
y	gravity on y axis
z	gravity on z axis

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

- EducationalBoosterPack.h
- EducationalBoosterPack.cpp

4.2 ebpLCD Class Reference

LCD on Educational BoosterPack.

```
#include <EducationalBoosterPack.h>
```

Public Member Functions

- ebpLCD ()

    Constructor.
- void begin ()

Initialisation.

- void clear ()

  Clear the screen.

- void setRowLine (uint8_t row, uint8_t line)

  Set the coordinates.

- void setContrast (uint8_t contrast)

  Set contrast.

- void setFont (uint8_t font)

  Select font.

- void print (String text)

  Print a string.

- void scrollLeft ()

  Scroll the display on column left.

- void scrollRight ()

  Scroll the display on column right.

4.2.1 Detailed Description

LCD on Educational BoosterPack.

4.2.2 Member Function Documentation

4.2.2.1 void ebpLCD::clear ( )

Clear the screen.

Clear the screen and place the coordinates at 0,0

4.2.2.2 void ebpLCD::print (String *text*)

Print a string.

Parameters

text	string

4.2.2.3 void ebpLCD::scrollLeft ( )

Scroll the display on column left.

Note

The coordinates are not updated.

4.2.2.4 void ebpLCD::scrollRight ( )

Scroll the display on column right.

Note

The coordinates are not updated.

4.2.2.5 void ebpLCD::setContrast (uint8_t *contrast*)

Set contrast.

Parameters

contrast	constrast value=32..48

4.2.2.6 void ebpLCD::setFont (uint8_t *font*)

Select font.

Parameters

font	font=0=small or 1=big

4.2.2.7 void ebpLCD::setRowLine (uint8_t *row,* uint8_t *line*)

Set the coordinates.

Parameters

row	row number, 0..40
line	line number, 0..1 with small font, 0 with big font

Note

Visible rows and lines

- Small font: 2 visible lines and 16 visible characters
- Big font: 1 visible line and 16 visible characters
  The coordinates are not impacted by the scrolling.

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

- EducationalBoosterPack.h
- EducationalBoosterPack.cpp

4.3 ebpMIC Class Reference

Microphone on Educational BoosterPack.

```
#include <EducationalBoosterPack.h>
```

Public Member Functions

- ebpMIC ()

  Constructor.
- void begin ()

  Initialisation.
- void get (uint32_t &value)

  Acquire microphone level.

4.3.1 Detailed Description

Microphone on Educational BoosterPack.

4.3.2 Member Function Documentation

4.3.2.1 void ebpMIC::get (uint32_t & *value*)

Acquire microphone level.

Parameters

value	level of the microphone, 10-bit coded=0..1023

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

- EducationalBoosterPack.h
- EducationalBoosterPack.cpp

4.4 ebpPOT Class Reference

Potentiometer on Educational BoosterPack.

```
#include <EducationalBoosterPack.h>
```

Public Member Functions

- ebpPOT ()
    Constructor.
- void begin ()
    Initialisation.
- void get (uint32_t &value)
    Acquire potentiometer position.

4.4.1 Detailed Description

Potentiometer on Educational BoosterPack.

4.4.2 Member Function Documentation

4.4.2.1 void ebpPOT::get (uint32_t & *value*)

Acquire potentiometer position.

Parameters

value	position of the potentiometer, 10-bit coded=0..1023

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

- EducationalBoosterPack.h
- EducationalBoosterPack.cpp

4.5 ebpRGB Class Reference

RGB LED on Educational BoosterPack.

```
#include <EducationalBoosterPack.h>
```

Public Member Functions

- ebpRGB ()

  Constructor.
- void begin ()

  Initialisation.
- void set (uint8_t red, uint8_t green, uint8_t blue)

  Set colour.

4.5.1 Detailed Description

RGB LED on Educational BoosterPack.

4.5.2 Member Function Documentation

4.5.2.1 void ebpRGB::set (uint8_t *red,* uint8_t *green,* uint8_t *blue*)

Set colour.

Parameters

red	red component, 0..255
green	green component, 0..255
blue	blue component, 0..255

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

- EducationalBoosterPack.h
- EducationalBoosterPack.cpp

Chapter 5

File Documentation

5.1 EducationalBoosterPack.h File Reference

Class library header

Project EducationalBoosterPack_main

Developed with `embedXcode`

Classes

- class ebpPOT

    Potentiometer on Educational BoosterPack.
- class ebpMIC

    Microphone on Educational BoosterPack.
- class ebpRGB

    RGB LED on Educational BoosterPack.
- class ebpACC

    Accelerometer on Educational BoosterPack.
- class ebpLCD

    LCD on Educational BoosterPack.

Macros

- #define **EducationalBoosterPack_Library_h**

Pins for white and RGB LEDs

- #define EBP_WHITE_LED P2_5

    PWM pin for white LED.
- #define EBP_RED_LED P2_1

    PWM pin for red component of RGB LED.
- #define EBP_GREEN_LED P2_2

    PWM pin for green component of RGB LED.
- #define EBP_BLUE_LED P2_4

    PWM pin for blue component of RGB LED.

Pins for accelerometer

- #define EBP_ACC_X A0

      analog pin for X-axis
- #define EBP_ACC_Y A3

      analog pin for Y-axis
- #define EBP_ACC_Z A4

      analog pin for Z-axis

Pins and constants for LCD

- #define EBP_LCD_SCK P1_5

      SPI clock pin for LCD.
- #define EBP_LCD_MOSI P1_7

      SPI data pin for LCD.
- #define EBP_LCD_RS P2_3

      command/data pin for LCD
- #define EBP_LCD_command 0

      command constant for LCD
- #define EBP_LCD_data 1

      data constant for LCD

Pins for other devices

- #define EBP_POT A3

      analog pin for potentiometer
- #define EBP_MIC A4

      analog pin for microphone
- #define EBP_BUZZER P2_6

      PWM pin for buzzer.
- #define EBP_GATOR P2_7

      pin for alligator hole

5.1.1 Detailed Description

Class library header

Project EducationalBoosterPack_main

Developed with `embedXcode`

Author

    Rei VILO
    embedXcode.weebly.com

Date

    Feb 11, 2013

Version

    106

Copyright

    © Rei VILO, 2013
    CC = BY NC SA