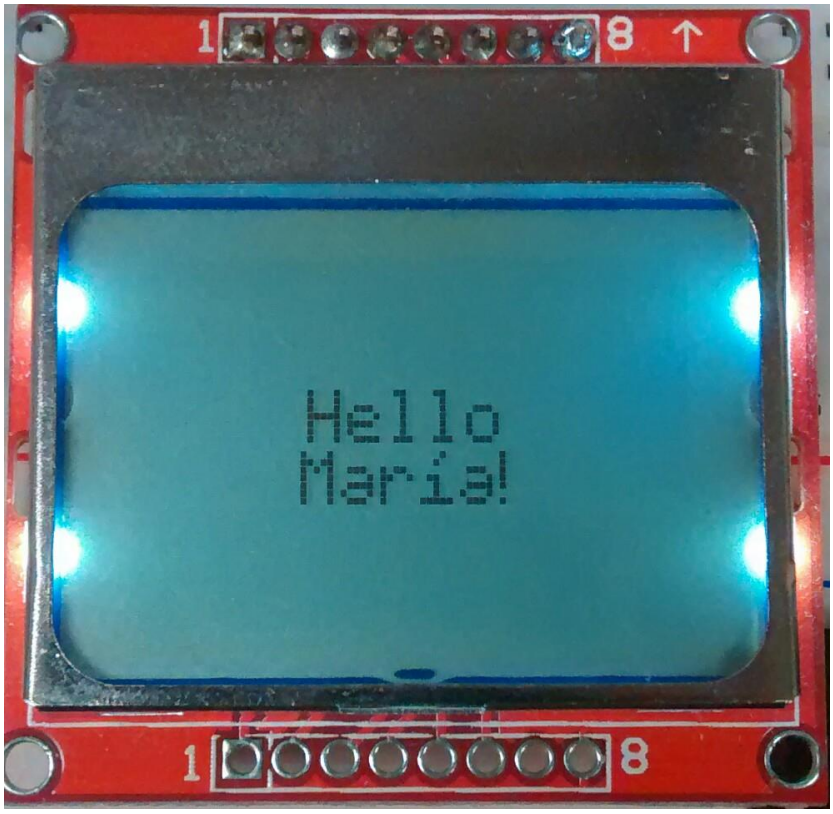
Nokia5110 Tiva C library

Version 2.2



CC BY NC SA

Javier Martínez Arrieta

martinezarrietajavier@gmail.com

Contents

Abstract

This is a short guide to be used as guidance in order to know how to use the library for the Nokia 5110 screen and also to dicover all the possibilities that it offers. This library has been tested with the TM4C123GH6PM microcontroller, so take it into account in case you desire to use it with another microcontroller as there might be necessary to make some changes.

License

This project is licensed under *Creative Commons Attribution-Non Commercial-Share Alike 4.0 International (CC BY-NC-SA 4.0)*. According to this license you are free to:

Share — copy and redistribute the material in any medium or format.

Adapt — remix, transform, and build upon the material.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Complete information about this license can be found at:
https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode

Files

There are three files used for this CCS project along with the Tivaware library, so in case you don't have it installed you will need it as some definitions come from that library. The main files are:

-main.c : This file contains references to the library files and example code.

-Nokia5110.h: This header file contains the definitions of functions and constants.

-Nokia5110.c: This file contains the implementation of the functions that make the screen work as desired.

Available languages

This library works for both English and Spanish languages, so accents, ñ and other characters are available.

Available functions

The functions available are the following:

-void clear_screen(short SSI) : Clears the whole screen. SSI indicates which SSI (SSI0, SSI1, SSI2 or SSI3) is being used.

-void screen_write(char string_to_write[], short alignment, short SSI): Writes the text sent through string_to_write[] parameter. Parameters are:

-string_to_write[]: Text to show on screen. For carriage return (enter character), write '\n' without the quotes.

-alignment: To make similar as when using word, the following alignments are available:

-ALIGN_LEFT_TOP

-ALIGN_CENTRE_TOP

-ALIGN _RIGHT_TOP

-ALIGN _LEFT_CENTRE

-ALIGN_CENTRE_CENTRE

-ALIGN _RIGHT_CENTRE

-ALIGN _LEFT_BOTTOM

-ALIGN _CENTRE_BOTTOM

-ALIGN _RIGHT_BOTTOM

-ALIGN_RANDOM: This one is thought for text such as 'Welcome' or similar and allows the text to be written in a random position.

-SSI: Indicates to which SSI is sent. Possible values are SSI0, SSI1, SSI2 and SSI3.

-void initialize_screen(short backlight,short SSI): Initializes the screen. Its parameters are:

-backlight: Enables or disables backlight. Possible values are BACKLIGHT_ON and BACKLIGHT_OFF.

-SSI: Indicates to which SSI is sent. Possible values are SSI0, SSI1, SSI2 and SSI3.

-void clear_columns(char ncolumns, short SSI): In case of desiring blank columns, it is possible to specify with this function the number of blank columns before writing new data.

-void fill_screen(short SSI): This method is useful in case of testing if all pixels are ok and inverse mode is not set.

-void startSSI0(): Configures SSI0 to be used.

-void startSSI1(): Configures SSI1 to be used.

-void startSSI2(): Configures SSI2 to be used.

-void startSSI3(): Configures SSI3 to be used.

-void enable_backlight(short SSI): Enables backlight to indicated SSI.

-void disable_backlight(short SSI): Disables backlight to indicated SSI.

-void set_menu(unsigned char[12][25]): Sets the menu for its later use.

-void show_menu(char current_position, short SSI): Shows the elements of the menu according the the current selected element in the list.

-void set_buttons_up_down(void): Sets the buttons to move up (PB0) and down (PB1) in the menu.

Bugs found

If using more than one SSI, I discovered that when configuring the clock of a second SSI, the clock of the first one was disabled (e.g. if startSSI0 is used, if startSSI1 is called, then SSI0 will not be usable). During the time I discover how to solve the problem, there is one thing that can be done, which is changing one line. If, for example, SSI0 and SSI1 are going to be used, go to startSSI1 in Nokia5110.c and change `SYSCTL_RCGCSSI_R|=SYSCTL_RCGCSSI_R1;` to `SYSCTL_RCGCSSI_R|=SYSCTL_RCGCSSI_R0+SYSCTL_RCGCSSI_R1;`

Connections

There are three possible SSIs to be used in order to connect the screen. Depending on which SSI is used, the connection should be as follows:

SSI0

Board	Screen
PA7	RST
PA3	CE
PA6	DC
PA5	DIN
PA2	CLK
3.3V	VCC
PA4	BL
GND	GND

SSI1

Board	Screen
PE2	RST
PF3	CE
PE1	DC
PF1	DIN
PF2	CLK
3.3V	VCC
PF0	BL
GND	GND

SSI2

Board	Screen
PB3	RST
PB5	CE
PB2	DC
PB7	DIN
PB4	CLK
3.3V	VCC
PB6	BL
GND	GND

SSI3

Board	Screen
PD7	RST
PD1	CE
PD6	DC
PD3	DIN
PD0	CLK
3.3V	VCC
PD2	BL
GND	GND