Freescale Semiconductor
Application Note

Document Number: AN3823
Rev. 0, 2/2009

The LCD Driver for MC9S08LG32

by: Saurabh Jhamb
Reference Design and Applications Engineering
Microcontroller Solutions Group

1 Introduction

MC9S08LG32 is a member of Freescale's HCS08 family of MCUs. It uses the S08 core and integrates abundant peripherals, such as LCD, SPI, IIC, SCI, and ADC. This application note describes the LCD driver in software that allows you to use the LCD functionality to its fullest.

Figure 1 is a hardware connection on the MC9S08LG32 demo board for LCD setup. The PC communicates with the MC9S08LG32 target system via a USB (BDM) interface. With BDM protocol, the PC can update the MC9S08LG32 firmware.

In this note, the driver interfaces are explained. Various applications for MC9S08LG32 can make use of this driver. The following sections describe the details and the steps for creating an application using it.

Contents

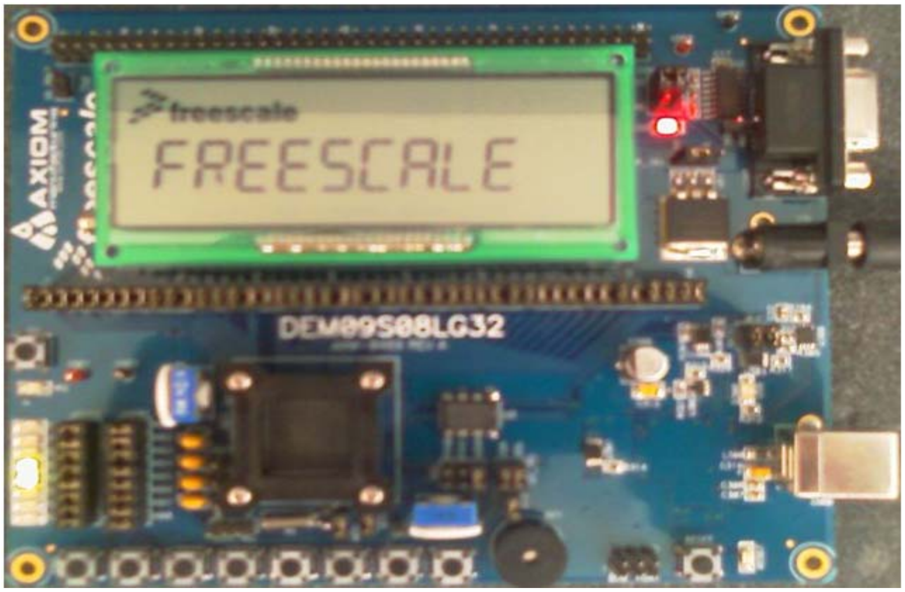


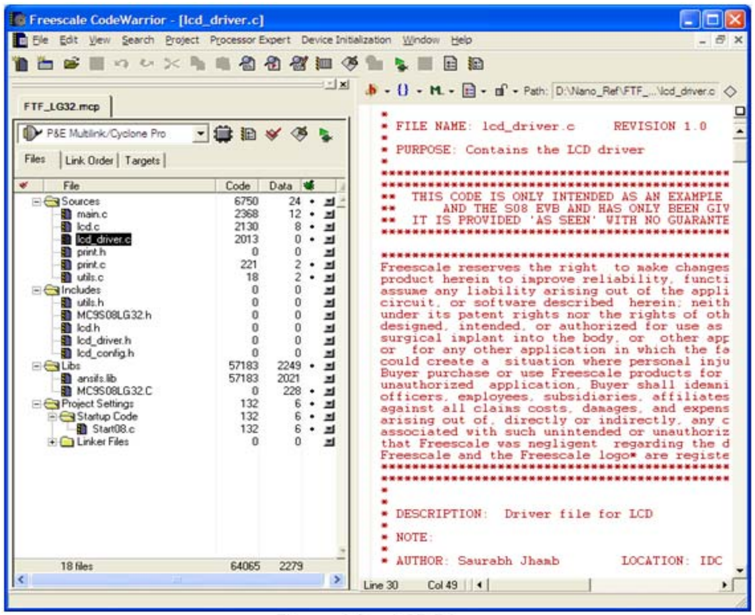
Figure 1. MC9S08LG32 Demo Board LCD Setup

2 LCD Driver Framework Overview

The LCD driver is provided as “C” code files. You can add these files to your applications. Some low level LCD driver files are specific to the LCD glass being used. Details are later in this document.

With the integration of the LCD driver, you can call LCD driver APIs to use the LCD functionality in your application.

Figure 2 illustrates the project for the MC9S08LG32 LCD driver.



File	Code	Data
Sources	6750	24
main.c	2368	12
lcd.c	2130	8
lcd_driver.c	2013	0
print.h	0	0
print.c	221	2
utils.c	18	2
Includes	0	0
utils.h	0	0
MC9S08LG32.h	0	0
lcd.h	0	0
lcd_driver.h	0	0
lcd_config.h	0	0
Libs	57183	2249
ansifs.lib	57183	2021
MC9S08LG32.C	0	228
Project Settings	132	6
Startup Code	132	6
Start08.c	132	6
Linker Files	0	0
18 files	64065	2279

```
*
* FILE NAME: lcd_driver.c     REVISION 1.0
*
* PURPOSE: Contains the LCD driver
*
*********************************************
*********************************************
**  THIS CODE IS ONLY INTENDED AS AN EXAMPLE
**      AND THE S08 EVB AND HAS ONLY BEEN GIV
**  IT IS PROVIDED 'AS SEEN' WITH NO GUARANTE
*********************************************

*********************************************
Freescale reserves the right  to make changes
product herein to improve reliability, functi
assume any liability arising out of the appli
circuit, or software described  herein; neith
under its patent rights nor the rights of oth
designed, intended, or authorized for use as
surgical implant into the body, or  other app
or  for any other application in which the fa
could create a  situation where personal inju
Buyer purchase or use Freescale products for
unauthorized  application, Buyer shall idemni
officers, employees, subsidiaries, affiliates
against all claims costs, damages, and expens
arising out of, directly or indirectly, any c
associated with such unintended or unauthoriz
that Freescale was negligent  regarding the d
Freescale and the Freescale logo* are registe
*********************************************
*********************************************
*
*
* DESCRIPTION:  Driver file for LCD
*
* NOTE:
*
* AUTHOR: Saurabh Jhamb        LOCATION: IDC
```

Figure 2. MC9S08LG32 LCD Driver

2.1 Files Introduction

There are five files associated with the driver.

- **lcd_driver.c** — It is the main file for the driver. It contains the various high-level API definitions exposed to the applications for LCD functionality.
- **lcd_driver.h** — This file contains the high-level API declarations. This file is included in the application that intends to use the LCD driver.
- **lcd.c** — The LCD glass–specific segment to pin mapping matrix is defined in this file.
- **lcd.h** — The LCD glass–specific macros are defined in this file.
- **lcd_config.h** — This file contains the LCD configuration specific flags that help the driver decide which LCD configuration is to be used. These flags are supposed to be edited as per the application LCD requirements.

2.2 External Interfaces

2.2.1 Data Structures

2.2.1.1 Configuration Macros

#define LCD_CLOCK

This macro defines the clock source for LCD operation.

NOTE

> The application must separately enable the clock being used by the LCD by configuring the ICS (internal clock source) module accordingly.

#define LCD_CHARGEPUMP

This macro defines whether the charge pump is on or off. 1–On, 0–Off.

#define LCD_VBIASSOURCE

This macro specifies the LCD-bias voltage generation source. 1–externally supplied, 0–internally generated.

#define LCD_INTERRUPT

This macro controls the on and off states of the LCD frame frequency interrupts that can be used to wake from wait and stop3 low power modes (refer to *MC9S08LG32 Reference Manual* for details). 1–on, 0–off.

#define LCD_PRESCLAR

This macro specifies the prescalar for dividing the LCD frame frequency to user-specified level. It can take values from 0 to 7.

2.2.2 APIs

Following is the detailed description of all the APIs exported to the application by the LCD driver. These interfaces are used by the application to access various LCD functions provided in MC9S08LG32.

2.2.2.1 lcd_Init

Prototype

unsigned char lcd_Init(void)

Description

This provides LCD initial configuration based on flags in "lcd_config.h" that include disabling clock gating to the LCD block, initializing various control registers, etc. It returns 0 if SUCCESS, otherwise 1.

NOTE

It configures the LCD to operate in 40x4 mode.

2.2.2.2 lcd_PrintString

Prototype

unsigned char lcd_PrintString(unsigned char *str)

Description

This takes a string as input and displays it on the alphanumeric character space on the LCD, starting from first alphanumeric digit. It returns 0 if SUCCESS, otherwise 1.

NOTE

It clears the digits that are not used for displaying the string passed.

2.2.2.3 lcd_PrintStringPos

Prototype

unsigned char lcd_PrintStringPos(unsigned char *str, unsigned char pos)

Description

This takes a string as input and displays it on the alphanumeric character space on the LCD, starting from alphanumeric digit as per the argument passed as pos. It returns 0 if SUCCESS, otherwise 1.

NOTE

It does not clear the digits that are not used for displaying the string passed.

2.2.2.4 lcd_SlideString

Prototype

unsigned char lcd_SlideString(unsigned char *str)

Description

This takes a string as input and displays it on the alphanumeric character space on the LCD, sliding the string from the rightmost digit to the leftmost digit. It returns 0 if SUCCESS, otherwise 1.

2.2.2.5 lcd_DispHexVal

Prototype

unsigned char lcd_DispHexVal(unsigned char val, unsigned char startloc)

Description

This takes a number as input and displays it in HEX format on the alphanumeric character space on the LCD, starting from alphanumeric digit passed as startloc. It returns 0 if SUCCESS, otherwise 1.

NOTE

It does not clear the digits that are not used for displaying the string passed.

2.2.2.6 lcd_DispDecVal

Prototype

unsigned char lcd_DispDecVal (float val, unsigned char startloc)

Description

This takes a float number as input and displays it in decimal base format on the alphanumeric character space on the LCD, starting from alphanumeric digit passed as startloc. It returns 0 if SUCCESS, otherwise 1.

NOTE

It does not clear the digits that are not used for displaying the string passed.

2.2.2.7 lcd_DispVal

Prototype

unsigned char lcd_DispVal(unsigned int val_int, unsigned char startloc)

Description

This takes an integer number as input and displays it in decimal base format on the alphanumeric character space on the LCD, starting from the alphanumeric digit passed as startloc. It returns 0 if SUCCESS, otherwise 1.

NOTE

It does not clear the digits that are not used for displaying the string passed.

2.2.2.8 lcd_StopBlinking

Prototype

unsigned char lcd_StopBlinking()

Description

This turns off the blinking feature of the LCD on the current display. It returns 0 if SUCCESS, otherwise 1.

NOTE

It does not alter the current display.

2.2.2.9 lcd_SetAltDisplay

Prototype

unsigned char lcd_SetAltDisplay(unsigned char *strnormal, unsigned char *stralt)

Description

This turns on the alternate blinking mode with the display switching between the two strings, passed as arguments. It returns 0 if SUCCESS, otherwise 1.

2.2.2.10 lcd_Clear

Prototype

unsigned char lcd_Clear(void)

Description

This clears all the LCD segments and enables the special character displaying "freescale". It returns 0 if SUCCESS, otherwise 1.

2.2.2.11 lcd_ActivateBlink

Prototype

unsigned char lcd_ActivateBlink(void)

Description

This activates LCD blink functionality for the current LCD display. It returns 0 if SUCCESS, otherwise 1.

NOTE

It does not alter the current display.

2.2.2.12 lcd_TestSpecialChars

Prototype

unsigned char lcd_TestSpecialChars(void)

Description

This is a test routine for all the special characters supported in the LCD. It returns 0 if SUCCESS, otherwise 1.

2.2.2.13 lcd_ScrollNumbersAndAlphabetbets

Prototype

unsigned char lcd_ScrollNumbersAndAlphabetbets(unsigned char digit)

Description

This provides a sample test routine to test the alphanumeric digit on the LCD by displaying 0–9 and A–Z on the digit number passed. It returns 0 if SUCCESS, otherwise 1.

2.2.2.14 lcd_AllDigitsTest

Prototype

unsigned char lcd_AllDigitsTest()

Description

This provides a sample test routine to test all the alphanumeric digits on the LCD by displaying 0–9 and A–Z on each of them one by one. It returns 0 if SUCCESS, otherwise 1.

2.2.2.15 lcd_Diagnostic

Prototype

unsigned char lcd_Diagnostic(void)

Description

This provides a test routine to test all the segments one by one. It returns 0 if SUCCESS, otherwise 1.

2.2.2.16 lcd_VolFunc

Prototype

unsigned char lcd_VolFunc(void)

Description

This is a utility function to display the VOLUME segments in a special manner. It returns 0 if SUCCESS, otherwise 1.

2.2.2.17 lcd_PowerFunc

Prototype

unsigned char lcd_PowerFunc(void)

Description

This is a utility function to display the POWER segments in a special manner. It returns 0 if SUCCESS, otherwise 1.

2.3 Assumptions

The descriptions in this document assumes the person reading it has full knowledge of all the configuration registers of all the blocks in MC9S08LG32, especially the LCD and ICS (internal clock source) blocks.

2.4 Design Decisions

- Provided all the functionalities present in the LCD block of MC9S08LG32 through external interfaces including access to all the special segments, capability to display strings, numbers on the LCD, blinking modes turning on and off.
- Provided test APIs to test the full functionality of the LCD in one go.

3 References

See S08LG Product Summary Page for more information and the documents released for MC9S08LG32.

THIS PAGE IS INTENTIONALLY BLANK

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Document Number: AN3823
Rev. 0
2/2009

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals", must be validated for each customer application by customer's technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

RoHS-compliant and/or Pb-free versions of Freescale products have the functionality and electrical characteristics as their non-RoHS-compliant and/or non-Pb-free counterparts. For further information, see http://www.freescale.com or contact your Freescale sales representative.

For information on Freescale's Environmental Products program, go to http://www.freescale.com/epp.

freescale™
semiconductor