

Color Display Driver Circuit and Driver Development for Embedded System Based on ARM Microcontroller

Harini H M¹, Ravikumar M N²

^{1,2}VTU University, Malnad College of Engineering, Salagame Road, 573202, Hassan

Abstract: Displays generally handle data input as character maps or bitmaps. In character-mapping mode, a display has a pre-allocated amount of pixel space for each character. In bitmap mode, it receives an exact representation of the screen image that is to be projected in the form of a sequence of bits that describe the color values for specific x and y coordinates starting from a given location on the screen. Displays that handle bitmaps are also known as all-points addressable displays. A display driver supports a particular family of display controllers and all displays which are connected to one or more of these controllers [2]. The drivers can be configured by modifying their configuration files whereas the driver itself does not need to be modified. The configuration files contain all required information for the driver including how the hardware is accessed and how the controllers are connected to the display. EmWin handles all access to the display.

Keywords: ST77335 TFT LCD, STM32F103 MCU, STemWin Library

1. Introduction

A display is a computer output surface and projecting mechanism that shows text and often graphic images to the computer user, using a cathode ray tube (CRT), liquid crystal display (LCD), light-emitting diode, gas plasma, or other image projection technology. The display is considered to include the screen or projection surface and the device that produces the information on the screen [5].

The terms display and monitor are often used interchangeably. Most computer displays use analog signals as input to the display image creation mechanism. This requirement and the need to continually refresh the display image mean that the computer also needs a display or video adapter. The video adapter takes the digital data sent by application programs, stores it in video random access memory and converts it to analog data for the display scanning mechanism using a digital-to-analog converter.

1.1 Display device

A display device is an output device for presentation of information in visual or tactile form. When the input information that is supplied has an electrical signal, the display is called an electronic display. Common applications for electronic visual displays are televisions or computer monitors. Some displays can show only digits or alphanumeric characters. They are called segment displays, because they are composed of several segments that switch on and off to give appearance of desired graph. The segments are usually single LEDs or liquid crystals. They are mostly used in digital watches and pocket calculators. There are several types:

The common segment displays are: 7-segment, 9-segment, 14-segment and 16-segment displays.

- Seven-segment display (most common, digits only)

- Fourteen-segment display
- Sixteen-segment display
- HD44780 LCD controller a widely accepted protocol for LCDs.

2. Display driver

A display driver supports a particular family of display controllers and all displays which are connected to one or more of these controllers. The drivers can be configured by modifying their configuration files whereas the driver itself does not need to be modified. The configuration files contain all required information for the driver including how the hardware is accessed and how the controller(s) are connected to the display. It explains the following in terms of each driver. Which display controllers can be accessed, as well as supported color depths and types of interfaces

- RAM requirements.
- Driver specific functions.
- How to access the hardware.
- Special configuration switches.
- Special requirements for particular display controllers.
- Screen and co-ordinates

The screen consists of many dots that can be controlled individually. These dots are called pixels. Most of the text and drawing functions that emWin offers in its API to the user program can write or draw on any specified pixel. The horizontal scale is called the X-axis, whereas the vertical scale is called the Y axis. Coordinates are denoted as a pair consisting of an X- and a Y-value (X, Y). The X-coordinate is always first in routines that require X and Y coordinates. The upper left corner of the display (or a window) has per default the coordinates (0,0). Positive X-values are always to the right; positive Y-values are always down. The below graph illustrates the coordinate system and directions of the X- and Y- axes. All coordinates passed to an API function are always specified in pixels.

Volume 7 Issue 5, May 2018

www.ijsr.net

Licensed Under Creative Commons Attribution CC BY

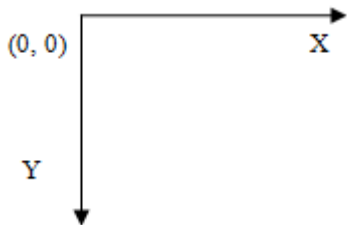


Figure 1: X and Y co-ordinate system of display screen

2.2 Display orientation

If the original display orientation does not match the requirements, there are different ways to change the display orientation:

- 1) Driver based configuration of the desired orientation
- 2) Function based configuration of display orientation

2.2.1 Driver based display orientation

If the display driver supports different orientations it is recommended to use the driver for setting up the right orientation. The way how to configure the display orientation then depends on the display driver to be used. Whereas the display orientation of the most common drivers is run-time configurable some drivers need to be configured at compile time.

2.2.2 Function based display orientation

Another possibility to set up the display orientation is to call GUI_SetOrientation(). Using this function is recommended if the display driver cannot be used. The number of required bytes can be calculated as follows:

$$\text{Virtual xSize} * \text{Virtual ySize} * \text{BytesPerPixel}$$

The number of bytes per pixel is for configurations from 1-8bpp 1, for systems with more than 8bpp up to 16bpp 2 and for systems with more than 16bpp 4. Each drawing operation first updates this buffer. After this the affected pixels are passed to the display driver device

3. STM32F103 MCU

The STM32F103xC/D/E high-density performance line family operates in the -40 to +105 °C temperature range, from a 2.0 to 3.6 V power supply. A comprehensive set of power-saving mode allows the design of low-power applications. These features make the STM32F103xC/D/E high-density performance line microcontroller family suitable for a wide range of applications such as motor drives, application control, medical and handheld equipment, PC and gaming peripherals, GPS platforms, industrial applications, PLCs, inverters, printers, scanners, alarm systems video intercom etc..

3.1 MCU parallel interface

The MCU can use one of following interfaces: 11-lines with 8-data parallel interface, 12-lines with 9-data parallel interface, 19-line with 16-data parallel interface or 21-lines with 18-data parallel interface. The chip-select CSX (active low) enables/disables the parallel interface. RESX (active low) is an external reset signal. WRX is the parallel data

write enable, RDX is the parallel data read enable and D [17:0] is parallel data bus [3].

The LCD driver reads the data at the rising edge of WRX signal. The D/CX is the data/command flag. When D/CX='1', D [17:0] bits is either display data or command parameter. When D/C='0', D [17:0] bits is command. The interface functions of 8080-series parallel interface are given in following table.

Table 1: The function of 8080-series parallel interface

IM2	IM1	IM0	Interface	D/CX	RDX	WRX	Read back selection
1	0	0	8 bit parallel	0	1	↑	Write 8-bit command (D7 to D0)
				1	1	↑	Write 8-bit display data or 8-bit parameter (D7 to D0)
				1	↑	1	Read 8-bit display data (D7 to D0)
				1	↑	1	Read 8-bit parameter or status (D7 to D0)

3.1.1 Write cycle sequence

The write cycle means that the host writes information (command or/and data) to the display via the interface. Each write cycle (WRX high-low-high sequence) consists of 3 control signals (D/CX, RDX, WRX) and data signals (D [17:0]). D/CX bit is a control signal, which tells if the data is a command or a data. The data signals are the command if the control signal is low (= '0') and vice versa it is data (= '1').

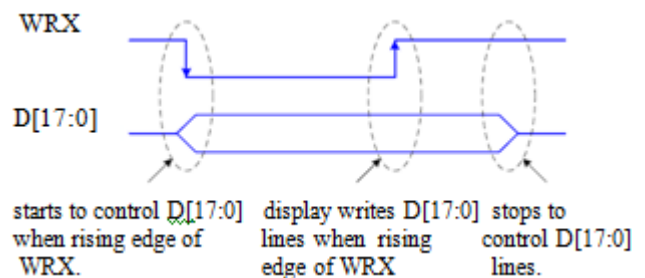


Figure 2: 8080-series WRX protocol

3.1.2 Read cycle sequence

The read cycle (RDX high-low-high sequence) means that the host reads information from LCD driver via interface. The driver sends data (D [17:0]) to the host when there is a falling edge of RDX and the host reads data when there is a rising edge of RDX.

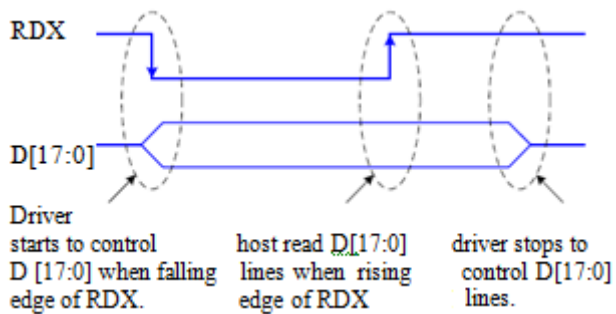


Figure 3: 8080-series RDX protocol

4. ST7735 262K Color Single-Chip TFT Driver

The ST7735 is a single-chip driver for 262K-color, graphic type TFT-LCD. It consists of 396 source line and 162 gate line driving circuits. The chip is capable of connecting directly to an external microprocessor, and accepts Serial Peripheral Interface (SPI), 8-bit/9-bit/16-bit/18-bit parallel interface. Display data can be stored in the on-chip display data RAM of 132 x 162 x 18 bits. It can perform display data RAM read/write operation with no external operation clock to minimize power consumption. In addition, because of the integrated power supply circuits necessary to drive liquid crystal, it is possible to make a display system with fewer components [2].

Table 2: General Features & Mechanical Specifications of TFT module

Item	Contents	Unit
LCD Type	a-Si TFT-Color Transmissive LCD	-
Display Format	Transmissive type	-
Display color	65K colors	-
Input Data	8 bit Interface	-
Viewing Direction	12 o'clock	-
Backlight	White(1x LED)	-
Driver IC	ST7735S	-
Resolution	128(RGB)(W) x160 (H)	Pixels
Dimensional outline	33.0(W)x42.5(H) x 2.15(T)	Mm
Active area	28.03(W) x35.04(H)	Mm
Diagonal length	1.77"	Inch

In addition to the main application, the user has to set and customize two essential interface files:

- LCD configuration file (LCDConf.c) – LCD Display initialization and configuration – LCD Display driver link and customizing – Additional hardware capability management
- GUI configuration file (GUICConf.c) – GUI management

STemWin Library includes two optimized drivers:

- Direct linear access (LIN) driver. This kind of driver is used for example on STM32F429, STM32F769, STM32H743 and any STM32 based on the LCD TFT Display Controller (LTDC) or LTDC/DSI (Display Serial Interface) hardware.
- FlexColor (indirect access) driver for serial and parallel bus external LCD controllers.

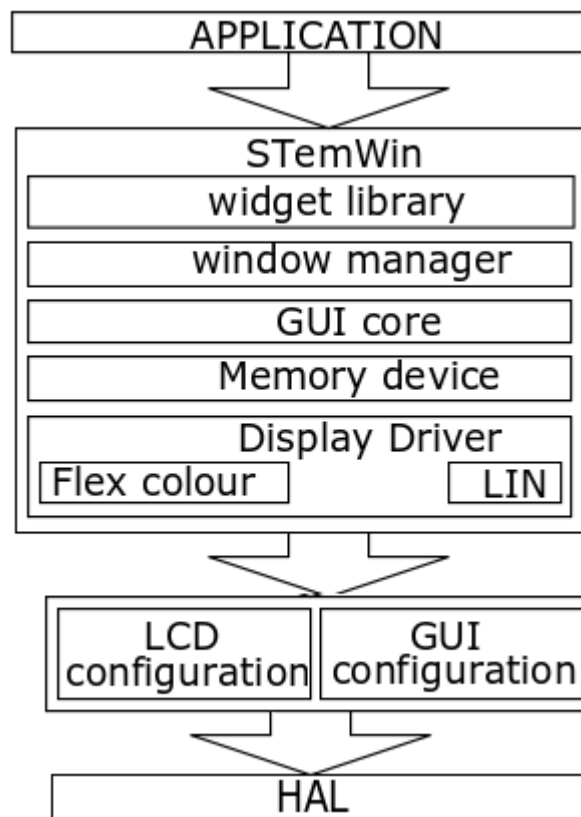


Figure 4: shows how STemWin is structured internally and how it can be implemented

5. Conclusion

Using STemWin library an efficient display driver code is developed and integrated to obtain a color LCD TFT display. The so obtained driver code is dumped to the ST7735 driver chip by using keil IDE. STM32F103 MCU is an efficient controller and the LCD display with 160*128 pixel resolution. An 8-bit parallel interface is seen between microcontroller and the display driver chip.

The implemented display is useful in display unit such as small smartphone displays and 2 way radio communication systems.

References

- [1] www.ece.usu.edu/ece_store/spec/S64128N-v11.
- [2] www.sitronix.com.tw > Products > LCD Driver for Mobile Phone
- [3] www.st.com > ... > STM32 Mainstream MCUs > STM32F1 Series > STM32F103
- [4] <https://www.sainsmart.com/sainsmart-stm32f103vct6-3-2-tft-lcd-gpio-serial-jtag>
- [5] https://www.webopedia.com/TERM/A/active_martix_display.html