Freescale Semiconductor
Application Note

Document Number: AN3606
Rev. 0, 03/2008

Understanding LCD Memory and Bus Bandwidth Requirements

ColdFire, LCD, and Crossbar Switch

by: Melissa Hunter
TSPG Applications

The need for graphical user interfaces in industrial and consumer applications is rising steadily. For instance, LCDs are common in appliances such as washers, dryers, refrigerators, and stoves for enhanced human control. Security and HVAC control systems are other examples of applications which require advanced graphical display interfaces to enable human control of the various system functions. To address this need, Freescale is introducing a family of cost effective and highly integrated ColdFire microprocessors that feature an integrated LCD controller.

Running a graphical display is a data intensive task. Memory must be allocated to store the graphic data, and bus bandwidth is also needed to write and read data.

In particular, large displays running at high refresh rates with color depths greater than 8-bits require a substantial amount of bus bandwidth and can starve the CPU and other modules in the microprocessor system which share memory for program and data storage and the LCD frame buffer.

Contents

freescale™ semiconductor

This application note discusses some of the concerns that must be taken into account while designing a system with a graphical LCD. Memory and bus bandwidth requirements are discussed with a focus on how ColdFire's integrated LCDC module uses graphic data. This application note also highlights strategies for the most effective use of system resources.

1 LCD Frame Buffer Memory Requirements

The memory requirements to support LCDs are fairly simple to calculate and understand. The LCDC fetches data from sequential memory locations beginning from the screen start address pointer programmed into one of the LCDC's on-chip registers. This means that enough contiguous data to store an entire frame's worth of data must be set aside in the system.

Table 1 shows the amount of memory required for some common panel sizes and color depths (bits per pixel or bpp). The bpp column lists two values. The first value is the actual number of color bits that are output from the LCDC to the LCD panel itself. The second value is the number of bits in memory used to store this data. In most cases, these numbers are equal but there are exceptions. For example if 18 bpp is used, the LCDC fetches one longword of data (32-bits) where the lower 18-bits are used as the pixel data, and the other bits are discarded. The memory bits per pixel is the actual amount of space used to store the data, so it is used to calculate the total memory requirement.

Table 1. LCDC Memory Requirements

Panel Resolution	Total Pixels	Color Depth bpp (pixel/memory)	Required Memory for Single Frame Buffer
800x600 (SVGA)	480K	18bpp (32bpp)	1920 KB
		12/16bpp (16bpp)	960 KB
		8bpp (8bpp)	480 KB
		4bpp (4bpp)	240 KB
		2bpp (2bpp)	120 KB
		1bpp (1bpp)	60 KB
640x480 (VGA)	307.2K	18bpp (32bpp)	1228.8 KB
		12/16bpp (16bpp)	614.4 KB
		8bpp (8bpp)	307.2 KB
		4bpp (4bpp)	153.6 KB
		2bpp (2bpp)	76.8 KB
		1bpp (1bpp)	38.4 KB
320x240 (QVGA)	76.8K	18bpp (32bpp)	307.2 KB
		12/16bpp (16bpp)	153.6 KB
		8bpp (8bpp)	76.8 KB
		4bpp (4bpp)	38.4 KB
		2bpp (2bpp)	19.2 KB
		1bpp (1bpp)	9.6 KB

For other LCD panel configurations the required memory can be calculated using the following equation:

Required Memory = Horizontal Resolution x Vertical Resolution x Memory bpp ***Eqn. 1***

The memory requirements shown here are the minimum amounts required. In some cases, space for additional frame buffers may be needed. For instance, it is common for some graphics software to maintain two separate frame buffers. This enables the display of one frame buffer while another frame buffer is being updated by the software.

If the new buffer is ready, the LCDC's start address pointer is updated to point to the new frame buffer, and the old buffer becomes the "new" buffer. This ping pong method ensures that an incomplete or partially drawn frame buffer is not displayed on the LCD.

In addition to maintaining multiple frame buffers so that one or more can be updated as needed, the LCDC supports two buffer pointers. The first pointer is for a background plane (the default pointer used by the LCDC) and the second is for a graphic window. The idea is similar to a desktop environment, where a background on the desktop takes up the entire screen and an open application window resides on top of the background. The graphic window must use the same color depth (bpp) setting as the background plane, but it can be a different size and discuss all or part of the background. Like the background plane, the graphic window has a start address register that is set to point to a sequential memory block with the data for the graphic window. Again, the size of the buffer depends on the resolution and number of bits per pixel in memory required to display the image, but the size can differ from that of the full panel.

For most applications, a single frame buffer that is displayed and updated at the same time is sufficient. This can result in some temporary artifacts on the LCD panel as a result of displaying graphic data which is in the process of being updated. If this is not acceptable or if use of the graphic window is desired, then you need to allocate memory to maintain multiple frame buffers.

2 LCD System Bus Bandwidth Requirements

If the LCDC is enabled, it continuously fetches data from the active frame buffer stored in system memory. The LCDC reads frame buffer data on a line by line basis. This means the LCD reads one line worth of data at a time at a predictable rate. This rate is a function of the pixel clock rate and the number of bits in memory used to store the color data for each pixel.

Table 2 shows the bus bandwidth calculations for some common panel sizes and color depths. The pixel clock frequencies listed come from LCD manufacturer specifications. For a given panel the manufacturer must recommend a pixel clock frequency or range of frequencies that result in an acceptable refresh rate for the screen. Panels can be run with a lower frequency pixel clock, but the lower refresh rate could cause flicker.

The SDRAM read throughput in Table 2 is the measured throughput for continuous read bursts from a 32-bit wide single data rate (SDR) SDRAM with a CAS latency of two clocks using the SDRAM controller on the ColdFire MCF532x processor. This is the maximum amount of data that the LCD can fetch under these conditions. Dividing the required max LCD bus bandwidth by the SDRAM read throughput gives the percentage of time the SDRAM bus is used by the LCD for a given pixel clock frequency and color depth.

Table 2. Bus Bandwidth Usage for Color LCD Panels

Panel Resolution	Typical LCD Manufacturer Recommended Pixel Clock Frequency	Color Depth bpp (pixel/memory)	Max LCD Bus Bandwidth (MBps)	SDRAM Read Throughput (MegaBytes /second)	Percentage of SDRAM Throughput Used by LCD
800x600 (SVGA)[1]	35–42 MHz (26.66 MHz)	18bpp (32bpp)	140–168 (106.64)	128 MBps	109.4%–131.25% (83.3%)
		12/16bpp (16bpp)	70–84 (53.32)	128 MBps	54.7%–65.6% (41.66%)
		8bpp (8bpp)	35–42 (26.66)	128 MBps	27.3%–32.8% (20.8%)
		4bpp (4bpp)	17.5–21 (13.33)	128 MBps	13.67%–16.4% (10.4%)
640x480 (VGA)	24.3–26.1 MHz	18bpp (32bpp)	97.2–104.4	128 MBps	75.9%–81.56%
		12/16bpp (16bpp)	48.6–52.2	128 MBps	37.9%–40.8%
		8bpp (8bpp)	24.3–26.1	128 MBps	18.98%–20.4%
		4bpp (4bpp)	12.15–13.05	128 MBps	9.5%–10.2%
320x240 (QVGA)	4.5–6.8 MHz	18bpp (32bpp)	18–27.2	128 MBps	14.1%–21.3%
		12/16bpp (16bpp)	9–13.6	128 MBps	7.0%–10.6%
		8bpp (8bpp)	4.5–6.8	128 MBps	3.5%–5.3%
		4bpp (4bpp)	2.25–3.4	128 MBps	1.76%–2.66%

[1] Currently, the ColdFire processors that include the graphical LCDC support a maximum pixel clock frequency of 26.66 MHz; therefore, most SVGA panels cannot be configured for the recommended screen refresh rate. Bus bandwidth calculations for the max allowable 26.66 MHz clock rate are shown in parentheses.

For other configurations the required bandwidth can be calculated using the following equation:

Max required LCDC Bandwidth (MBps) = (bpp in memory/8) x LSCLK frequency (MHz) ***Eqn. 2***

Over time the LCD actually uses less bandwidth than the calculated maximum. The LCD expects one pixel's worth of color data on each pixel clock when reading in a line, but after an entire line is read in; there is a delay before the next line starts. Then if the entire frame (all of the horizontal lines) is sent, there is another delay before the next frame starts. These delays decrease the system bus bandwidth required by the LCDC to collect data for the LCD panel. The actual amount of delay required varies depending on the requirements of the LCD panel used.

Most systems operate such that LCDC bus traffic is asynchronous to other ongoing events in the system, so it is better to use the maximum LCDC bus bandwidth number as an estimate when figuring system bus loading. The maximum number gives the worst case measurement of system bus bandwidth needed by the LCDC, which in turn can be used to estimate the remaining bandwidth available for the core and other on-chip bus masters (FEC, USB, etc.).

3 ColdFire with LCD System Optimization

3.1 Enable LCD Burst Accesses

The greatest impact on the LCD system bus usage can be made by enabling the LCD controller to request burst accesses on the external bus. Bursts allow for the fastest, most efficient transfer of data (approximately a 2x performance boost). This means that the LCD can get the data it needs faster, freeing up additional bus bandwidth for other masters.

The burst configuration register (BCR) in the system control module is used to enable (or disable) bursting for all of the on-chip bus masters other than the CPU core. Freescale recommends setting the BCR to 0x3FF to allow for bursting by on-chip masters, including the LCD module. Enabling bursting in the BCR is necessary to get the SDRAM read throughput used in Table 2.

If bursting is not enabled the LCD needs a significantly larger percentage of the SDRAM bus's time.

3.2 Crossbar Switch Priorities

The settings for the system bus arbitration between masters can also have an impact. The crossbar switch (XBS) on ColdFire devices with a graphical LCD controller allows the priority of on-chip masters to be set on a slave-by-slave basis. Depending on the needs of the system, the XBS must be programmed accordingly. If LCD performance is critical, then the LCD controller must be given high priority access to the memory where the frame buffer is stored.

3.3 Memory Allocation

One of the main benefits of the crossbar switch is that it allows for concurrent accesses by multiple masters as long as those masters are accessing different slave ports. If care is taken while allocating system memory, the crossbar switch can yield significant system performance gains. Splitting up the slave memory resources between masters helps to decrease competition between masters accessing the same resources simultaneously and creates additional system bus bandwidth.

Here is an example to show how this can be done. Consider a system with three active masters—the CPU core, a USB controller, and the LCD controller. If the CPU core is executing code that is stored in SDRAM with the cache enabled, the core may not require a large amount of bandwidth on the SDRAM bus.

The USB can be set up so that its data structures are stored in on-chip SRAM, and the LCD can use SDRAM for the frame buffer. In this configuration, accesses from the USB controller can access the on-chip SRAM at the same time as the LCD and/or core is accessing SDRAM. The LCD controller and core have to arbitrate for access to the SDRAM, but because the SDRAM bandwidth required by the core is minimized by the cache, the likelihood of either the LCD controller or core having to wait to access SDRAM is reduced.

4 Conclusion

While designing a system that uses a graphical LCD, it is important to take into account the amount of bus bandwidth that is required to refresh the screen. Refreshing a graphical LCD display can require a large amount of system bus bandwidth; however, the ColdFire architecture does offer features that can help to reduce the overall system impact.

THIS PAGE IS INTENTIONALLY BLANK

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

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)
support@freescale.com

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 Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
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: AN3606
Rev. 0
03/2008

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