

# 一种面向智能仪器的可重构嵌入式彩色 LCD 系统设计

廖观亮, 蒋东方, 魏子阶

(西北工业大学 自动化学院, 西安 710129)

**摘要:** 按照使用方便、配置灵活、适用面广的特点要求, 论证了一种以 LPC1788 为显示控制器并面向智能仪器的 LCD 彩色图像图形显示系统的设计方案; 首先分析了智能仪器图像图形显示系统的特点和要求; 然后给出设计思路和总体方案; 接着逐项讨论嵌入式图形显示控制器、显示数据的信息传输与变换、LCD 面板接口以及显示数据源所用显示 API 函数库等系统软硬件的设计与实现; 最后采用 115200 的波特率从外部向显示系统发送显示命令对系统进行实际测试并给出了最终的实验结果; 实验结果表明所设计的显示系统满足普通智能仪器的使用需求。

**关键词:** 图形显示系统; LCD; 嵌入式系统; 硬件可编程; CPLD

## A Reconfigurable Embedded Color LCD System Design for Intelligent Instrument

Liao Guanliang, Jiang Dongfang, Wei Zijie

(College of Automation, Northwest Polytechnical University, Xi'an 710129, China)

**Abstract:** An embedded color graphic display system for intelligent instrument is constructed, which uses LPC1788 as the display controller and aims at convenient usage, flexible configuration and width different applications. The characteristics and demands of the graphic display system of intelligent instrument are analyzed first. Guideline and outline of the embedded system is proposed followed. Then, the design of the hardware and software of the system, including the embedded graphic display controller, the transmission and transformation of display data, the interfaces between the controller and LCD panel, and the API library used by display data source are discussed one by one. The system test is carried out at last, in which a baud rate of 115200 is used to send display commands to the system from outside of the system and experimental results show that the designed system can meet the requirements of ordinary intelligent instruments.

**Keywords:** graphics display system; LCD; embedded system; programmable hardware; CPLD

### 0 引言

图像显示是最直观的人机交互手段<sup>[1-3]</sup>。由于实现技术的复杂性或成本因素, 目前的一般智能仪器, 常避免引入图像显示<sup>[3-6]</sup>。随着电子技术的发展, 彩色 LCD 面板和高性价比微控制器的售价已十分低廉。在智能仪器图像显示系统设计中, 成本因素已成为次要因素, 而接口简单、可重配置、显示功能丰富等要求则成为主要考虑因素。

普通台式计算机, 笔记本计算机和移动通信设备等都配有彩色 LCD 显示系统。但考虑到智能仪器存在诸多局限, 如存储容量有限等。而图像显示又需要大量的存储资源。如果直接借用当前的彩色 LCD 显示系统则会对智能仪器既有的软硬件设计提出巨大的挑战<sup>[6]</sup>。若能设计出一个独立的显示模块, 使其在正确的初始化配置后, 既能依照简单的命令, 完成图像的显示控制, 又能兼容不同大小和分辨率的彩色 LCD 面板, 则该显示模块对于提高智能仪器的设计水平以及增强智能仪器的使用灵活性具有重

要价值。

### 1 LCD 显示系统的组成及工作原理

台式计算机的 LCD 显示系统结构如图 1 所示<sup>[1-3]</sup>。台式计算机主板上的扩展总线 (PCI 或 PCIe 总线) 将显示信息传递给插接在总线槽中的显卡。显卡根据主板 CPU 的显示要求形成显示图像, 并将显示图像转换成 VGA 接口格式的扫描时钟、数字信号后输出给带 VGA 接口的独立显示屏。笔记本计算机将显卡、LCD 显示屏集成在一起, 因而省去了主板与显卡之间的 PCI/PCIe 总线、VGA/LVDS 转换、显示屏交流电源等部分。但由于笔记本计算机的 GPU 子系统与主板及所采

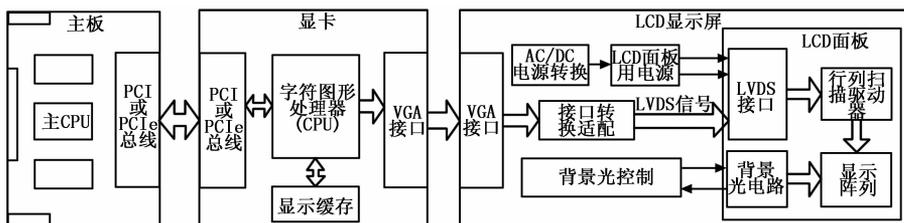


图 1 台式计算机显示系统结构示意图

用 LCD 面板紧密结合, 其设计细节多属商业技术秘密。

从上文可知, LCD 图像图形显示系统主要包括 3 个部分: 数据源、显示面板和介于此二者之间的显示处理系统。

收稿日期: 2014 - 02 - 13; 修回日期: 2014 - 03 - 23。

作者简介: 廖观亮 (1989 -), 男, 江西赣州人, 硕士研究生, 主要从事嵌入式系统和智能仪器仪表方向的研究。

数据源提供显示数据。在台式计算机中, 数据源的硬件就是主 CPU, 数据的类型取决于应用程序。数据源在要求显示系统完成显示时, 若待显示的信息是字符, 则数据源只给出待显示字符的代码、屏幕位置、大小等, 并不给出显示字符的点阵字模, 也不关心 LCD 的具体驱动方法。若待显示的信息是图形, 则数据源只给出待显示图形的描述参数, 并不给出图形的点阵信息。

显示面板是具体的人机界面。目前市售显示面板通常集成本点阵扫描控制器。该扫描控制器根据所支持的时钟、数据信号, 对 LCD 点阵进行扫描, 从而形成显示的图形图像。

可见, 将数据源与 LCD 显示面板直接连接并不能形成显示系统。在二者之间需设计显示处理系统, 用以完成对数据源给出的代码或数据的解释、变换和对 LCD 显示面板的驱动。在智能仪器中, 数据源就是仪器的主功能控制器, 嵌入式显示处理系统与数据源、LCD 面板之间的关系如图 2 所示。

本文结合数据源与 LCD 显示面板的一般特点, 设计一种面向智能仪器使用的嵌入式独立显示处理系统。

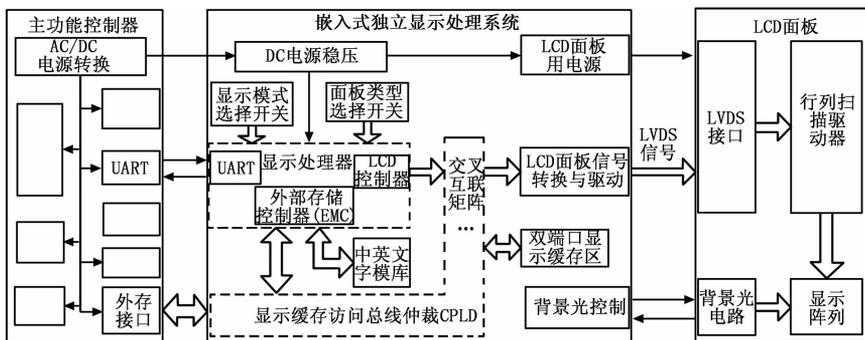


图 2 智能仪表显示系统结构图

## 2 系统设计目标与思路

设计目标是构建一块采用低压直流供电、面向市售普通 LCD 显示面板、具有通用显示数据和命令传输接口和完善的底层显示 API 的彩色 LCD 智能模块。该模块的总体结构框架如图 2 中间部分所示。

为了能广泛适配现有智能仪器, 在本显示系统与数据源的接口端, 配置了 RS232 接口和公共 RAM 双接口。使显示处理系统既可通过 RS232 接口得到数据源的数据, 又可以让数据源将显示数据通过公共 RAM 双接口直接写入显示缓存区。

## 3 硬件设计

根据图 2, 可将嵌入式显示处理系统硬件分为显示处理器与接口子系统、双端口显示缓存区、LCD 面板信号转换与驱动、字符图形字模库等主要部分。

### 3.1 显示处理器与接口子系统

显示处理器与接口子系统主要完成显示数据的获取, 显示命令和数据的解析, 显示色彩和图案、字符点阵的生成, 显示点阵在 LCD 面板上的刷新控制。

本系统采用 LPC1788 作为显示处理器, LPC1788 是 NXP 公司设计的一款 32 位 ARM 核微控制器, 自带 UART、外部存储器控制 (EMC)、LCD 控制等外围接口。因此, 只需在 LPC1788 UART 接口外增添驱动电路即可实现显示处理系统与数据源之间的串口通信。而显示处理系统与 LCD 面板之间

也只需在 LPC1788 LCD 控制器接口外设计 LCD 面板所需信号的接口转换与驱动电路。显示处理系统对显示缓存器和字符图形字模库的访问可通过 LPC1788 的外部存储器控制接口实现。可见, 采用 LPC1788 作为显示处理器可简化硬件设计, 缩小系统体积。

### 3.2 双端口显示缓存区子系统

显示缓存区存储需要显示的内容。由于 LCD 显示面板按扫描方式获得显示输出。因此显示处理系统需按照一定的信号格式和频率, 反复地依次刷新显示面板的各显示像素并将其输出到 LCD 面板进行显示<sup>[5-6]</sup>。为保证刷新频率, 各显示像素的数据必须预先按扫描次序存储在显示缓存区中, 再由硬件电路自动周期性地依次读取并转换成 LCD 面板识别的格式传输给 LCD 面板进行显示。

当数据源通过串口传输显示命令时, 由显示处理系统根据命令自主生成显示缓存区内容。当数据源由公共 RAM 接口传输显示内容时, 由数据源直接填充显示缓存区。以上两种情况, 最终都由显示处理器读取显示缓存区的数据并通过其

LCD 接口及附加电路输出到 LCD 面板。可见, 显示缓存区需接受数据源与显示处理器的双重访问。因此可采用双端口 RAM 结构。本设计中没有采用市售的双端口 RAM, 而是利用普通 SDRAM, 通过配置一片 CPLD 来实现地址、数据和控制总线切换, 从而实现双端口 RAM 访问功能。SDRAM 采用 K4S561632H, 其容量为 4M × 16bit × 4 Banks, 最大工作频率为 133 MHz。由于其数据总线宽度是 16 位。为最大限度地发挥 LPC1788 的性能, 可采用两片 K4S561632H 拼接成 32 位总线宽度。CPLD 选用 EPM240T100C5N。该芯片实现的双端口总线切换逻辑如图 3 所示。图 3 中的 Busy 信号是由 LPC1788 给出的显示缓存总线使用权仲裁信号, 当 Busy 为高电平时, 显示处理器获得显示缓存的访问权限。反之, 则数据源获得访问权限。其中连接数据源的信号线用 EXT 前缀标识, 连接显示处理器的信号用 1788 前缀标识。SDRAM 的 nCAS、nRAS、CLKOUT、CKE-OUT 等其余信号的切换实现方式与 WE 信号实现方式类似。

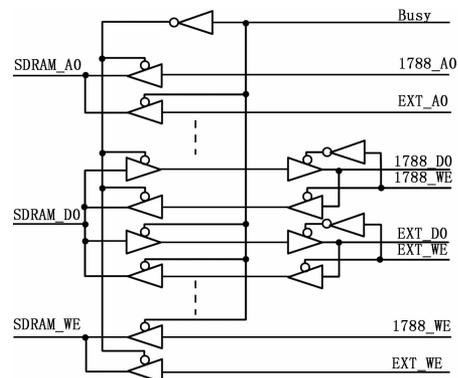


图 3 CPLD 实现的显存总线仲裁逻辑示意

### 3.3 LCD 面板信号转换与驱动

由于市售 LCD 面板种类繁多, 各面板的对外接口定义也各不相同, 为了能方便的兼容不同接口类型的 LCD 面板, 在

设计中采用可编程逻辑器件 CPLD 作为交叉互联矩阵来实现不同的显示引脚配置。实验所用 LCD 面板采用 LVDS 接口，其信号转换与驱动结构示意图如图 4 所示。

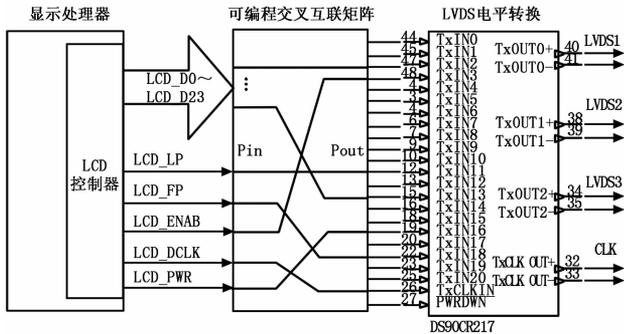


图 4 LCD 面板信号转换与驱动结构示意图

### 3.4 字符图形字模库

为简化使用界面，保证显示界面的一致性，并考虑到 LCD 面板需采用像素点阵的形式显示字符，智能仪器一般不使用矢量或轮廓字库，而采用点阵字模库。字符图形字模库用于存储字符或汉字的点阵字模。本设计采用 16×16 点阵字模文件。字模库存储芯片选用存储容量大且使用广泛的 Nand-Flash 型芯片 K9F1208U0A，它的容量大小为 64 MB，而存储 GB2312 规定的常用汉字和字符的 16×16 点阵字模库只需 256KB 容量，因此满足设计要求。在该芯片的最低地址端安置点阵字模库，其余容量可用于存储特定仪器所需要的特定字符和固有图形、图像。

## 4 软件设计

软件设计分为两部分。一部分是运行在显示处理器上的软件固件。另一部分是本系统提供给数据源处理器使用的 API 软件。从上文讨论可知，在不同的工作模式下，软件完成的任务不同。目前系统支持以下几种工作模式。

模式 1：将双端口 RAM 全部当作显示缓存区，数据源直接填写显示缓存区。该模式下显示处理器只需将显示缓存区的数据取出，并在驱动电路的配合下转换成 LCD 面板支持的 LVDS 串行数据流与同步信号流即可。

模式 2：将双端口 RAM 的一部分当作公共存储区，另一

部分当作显示缓存区。数据源利用公共存储区向显示处理器发送显示命令，由显示处理器生成显示数据并填充显示缓冲区。在必要时，数据源也可以直接改写显示缓冲区。其他工作同模式 1。

模式 3：数据源通过 RS232 接口传输显示命令。显示处理器完成显示数据的生成并填充显示缓冲区，然后显示处理器等将其转换成 LCD 面板要求的 LVDS 信号。

由于在模式 2、模式 3 时都需接收数据源发来的显示命令。因此需要定义显示命令的格式。

### 4.1 显示命令格式及命令参数

为方便使用，本系统设计了如表 1 所示的 8 字节显示命令格式。其中联合型短参数与长参数的 C 语言格式定义分别为

```
Union sParam {
    uChar uCh[2];
    short int psInt;
}
```

```
Union lParam {
    uChar uCh[4];
    short int slnt[2];
    long int llnt;
}
```

部分已实现的命令码及对应参数定义如表 2 所示。

表 1 显示命令格式

字段次序	字段名称	字段内容	字段长度	字段类型
1	Cmd	命令码	1 Byte	uChar
2	Attr	属性	1 Byte	uChar
3	sParam	短参数	2 Byte	union
4	lParam	长参数	4 Byte	union

### 4.2 显示处理器固件设计 (流程图)

显示处理器固件的设计流程图如图 5 所示。限于篇幅，这里不再对具体细节进行描述。

### 4.3 供数据源调用的 API 函数库设计

考虑到移植性，本系统提供给数据源调用的所有 API 函数均采用 C 语言编写，目前所设计的 API 函数库中各函数原形如下。

表 2 显示命令及其参数定义

命令码	命令功能	属性定义	短参数定义	长参数定义
0x03	Disp_SetPen 设置画笔	0x00 保留	sParm. sInt 画笔宽度(Pixels)	lParam. llnt 画笔颜色(RGB)
0x05	Disp_SetBackground 设置背景颜色	0x00 保留	0x00 保留	lParam. llnt 背景颜色(RGB)
0x06	Disp_PenMoveTo 移动画笔到给定点	0x00 保留	0x00 保留	lParam. sInt[2] 给定点坐标(x,y)
0x09	Disp_LineTo 从当前点画直线到给定点	0x00 保留	0x00 保留	lParam. sInt[2] 给定点坐标(x,y)
0x0a	Disp_DrawArc 以当前笔位置为圆心,给定半径及起止角画圆弧	0x00 保留	sParm. sInt 圆的半径(Pixels)	lParam. sInt[2] 起止角度(Sa, Ta)
0x0c	Disp_DrawStr 以当前位置为左上角坐标显示字符	0x00: 英文字符 0x05: 汉字字符	前 2 个字符的 ASCII 码 第 1 个汉字国标码	后 4 个字符的 ASCII 码 后 2 个汉字国标码
0x0f	Disp_DrawPicture 在指定点画特定图形图像	0x00 保留	sParm. sInt 图形图像的存储编号	lParam. sInt[2] 图形左上角坐标

```

int Disp_Init(uchar Mode, long int
QueueAddr, long int QueueSize);
int Virtual_DispBuff_AccessStatus(Void);
int Disp_GetFromQueue(long int
QueueAddr, uchar * data);
int Disp_StoreToQueue(long int QueueAddr, uchar * data);
int Disp_SetPen(short int Width, long int Color);
int Disp_SetBackground(long int Color);
int Disp_PenMoveTo( short int x, short int y);
int Disp_LineTo(short int x, short int y);
int Disp_DrawArc(short int StartAngle, short int EndAngle, short
int Radii);
int Disp_DrawString(short int Right, short int Up, w_char * str);
int Disp_DrawPicture(short int Right, short int Up, long int Pic-
Num);
    
```

以上所有的 API 函数的返回值如果为 0, 表明调用成功, 其它返回值表明执行中出错。返回值为错误编号。

在使用 API 函数库中其他函数前, 需先调用初始化函数 Disp\_Init。该函数的 Mode 参数必须与系统硬件中的显示模式选择开关的设定值相同, 用以通知运行在数据源端的 API 函数当前显示系统的工作模式。该函数中的 QueueAddr 参数在不同的工作模式下的意义不同。当显示系统工作在模式 1 或模式 2 时, QueueAddr 要求设定为显示缓冲区在数据源处理系统存储空间的起始地址, QueueSize 参数指明数据源硬件分配的缓冲存储器容量。在这两种模式下, 数据源必须先具体实现一个函数 Virtual\_DispBuff\_AccessStatus, 用非零返回值指明数据源可访问显示缓冲区; 当显示系统工作在模式 3 时, 数据源需管理一个与显示系统通信的 RS232 接口, 以便将显示 API 函数生成的数据包经该接口发往显示系统。本设计中, 显示 API 与 RS232 服务程序之间的数据交换是通过一个设置在数据源硬件 RAM 中的 FIFO 队列完成的。此时, Disp\_Init 函数中 QueueAddr 指定这个 FIFO 的 RAM 首字节地址, QueueSize 指定该 FIFO 的大小。Disp\_Init 函数会根据这两个参数自动建立并管理该 FIFO 队列。数据源处理器利用 GetFromQueue 函数, 从队列 QueueAddr 处获得数据, 并将获得的数据存储到 \* data 指针给出的地址处。当队列中有数据时, 每次调用 GetfromQueue 都将得到一个 12 字节数据的数据包。数据源的 RS232 服务程序只需将所得到的数据包一次传输出去即可。

显示 API 中的 StoreToQueue 函数为 API 内部使用。其他函数含义自明, 这里不再细述。

### 5 系统测试

采用一块 LPC1768 开发板作为数据源, 一块 LVDS 接口面板 B141XN04V2 为测试面板。构建了测试系统。显示系统工作在模式 3, 测试结果表明所设计的显示系统可用于智能仪器显示。

### 6 结束语

借用 LPC1788 内嵌的外设接口, 同时充分利用 CPLD 器

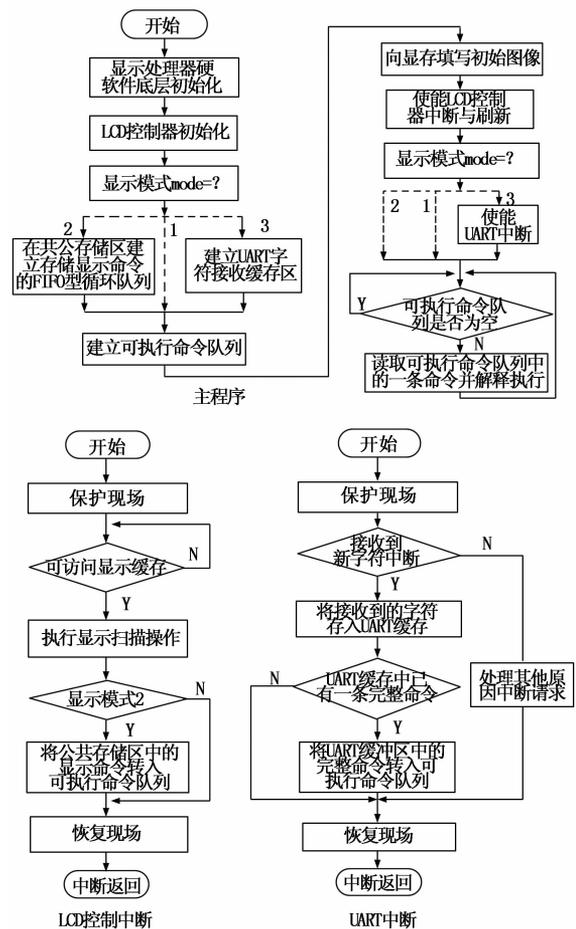


图 5 软件硬件设计流程图

件的可编程灵活性, 为智能仪器设计出功能完善的独立显示系统硬件。在软件设计上仔细划分软件层级, 为智能仪器提供方便的图形图像显示 API 接口。综合以上软硬件设计, 可以以很小的代价, 在智能仪器中配置 LCD 彩色图像图形显示系统。本文介绍的显示系统硬软件设计既有独立的结构又适用性广, 具有很强的扩展性, 为进一步扩充显示系统功能提供了方便。

### 参考文献:

- [1] David F. Rogers 计算机图形学的算法基础 [M]. 北京: 机械工业出版社出版社, 2002.
- [2] Randal E. Bryant. 深入理解计算机系统 [M]. 北京: 机械工业出版社出版社, 2011.
- [3] 马舜峰, 金龙旭. 一种基于 ARM9 的彩色 TFT-LCD 模块设计与实现 [J]. 液晶与显示, 2010, 25 (5): 718-723.
- [4] 张超建, 王厚军. 一种基于 FPGA 的 TFT-LCD 液晶显示模块设计 [J]. 自动化信息, 2009, (9): 52-54.
- [5] 陈建群, 段晨东. 液晶模块的显示绘图方法 [J]. 计算机测量与控制, 2009, 17 (4): 748-750.
- [6] 胡爱华, 杨郁池. 液晶显示模块及其在智能仪表当中的应用 [J]. 计算机测量与控制, 2007, 15 (2): 275-277.