

文章编号:1007-2780(2011)05-0665-08

采用 SOPC IP 核技术实现液晶屏显示

刘 敏,戴曙光,穆平安

(上海理工大学 光电信息与计算机工程学院,上海 200093, E-mail: mark_lewis@foxmail.com)

摘 要: 设计了 ST7920 系列 12864 液晶显示模组符合 Avalon MM 接口规范的 IP 核,建立了基于 Nios II 的 SOPC 系统,实现了液晶屏的图形与字符显示。按照开发流程详细介绍了系统设计步骤与关键技术,其中 LCD12864 IP 核的硬件设计部分采用 Verilog HDL 编写并遵循 Altera 所建议的 Avalon 信号类型最新命名规则, LCD12864 IP 核的软件设计部分给出了详细的 C 语言驱动代码与程序说明,并给出了图形显示的应用程序范例。对所设计的 SOPC 系统进行了实验验证,得到了理想的效果。

关 键 词: Nios II ; LCD12864; ST7920; SOPC; IP 核

中图分类号: TN27; TP311 **文献标识码:** A **DOI:** 10.3788/YJYXS20112605.0665

Implementation of LCD Display with IP-Based SOPC Technology

LIU Min, DAI Shu-guang, MU Ping-an

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China, E-mail: mark_lewis@foxmail.com)

Abstract: An IP core of ST7920 series 12864 LCM was designed meeting the Avalon MM interface specification, an NiosII-based SOPC system was set up and the graphical and character display of LCD were realized. The system design and key technologies were introduced step by step following the sequence of development process. The hardware design of LCD12864 IP core was written in Verilog HDL, following the newest signal naming conventions for automatic type and interface recognition suggested by Altera, while detailed C driver code and program specifications were provided in the software design of LCD12864 IP core. An application program example of graphical display was also provided. The IP-based SOPC system has been proved feasible according to the actual experiment performances which get the desired results.

Key words: Nios II ; LCD12864; ST7920; SOPC; IP core

1 引 言

可编程片上系统(SOPC)技术最早是由美国 Altera 公司于 2000 年提出的,其目标是将尽可能大而完整的电子系统在一块 FPGA 中实现,即由单个芯片完成整个系统的主要逻辑功能^[1]。NiosII 是 Altera 公司专门针对 FPGA 推出的一款采用

流水线技术和哈佛结构的通用 RISC(Refined Instruction Set Computer)嵌入式软核处理器。结合丰富的外设、专用指令和硬件加速单元,Nios II 可以低成本地提供极度灵活和功能强大的 SOPC 系统,实现用户定制系统的设计要求。

IP 核(Intellectual Property Core),即知识产权核,是 SOPC 设计的关键技术之一。IP 核是指

收稿日期: 2011-03-28; 修订日期: 2011-05-17
基金项目: 上海市教委重点学科资助项目(No. J50505)
作者简介: 刘敏(1986—),男,江西安福人,硕士研究生,主要从事图像测试与处理的研究。

一些在数字电路中常用但比较复杂的功能块,如 SDRAM 控制器、PCI 接口都设计成参数可修改的模块,供用户直接调用^[2]。

本设计以 ST7920 控制驱动的重庆育松电子生产的型号为 QC12864B 的 128×64 点阵汉字图形液晶显示模块为例,在 Quartus II 10.1 和 Nios II IDE 10.1 的开发环境下,选用 Cyclone II 系列的型号为 EP2C8Q208C8 的 FPGA 芯片,完成了 12864 液晶模块的 SOPC IP 核的硬件架构与软件驱动设计,完整构建起了基于 Nios II 的 SOPC 系统,实现了 12864 液晶模组的图文显示。

2 背景简介

2.1 ST7920 系列 12864 LCM

12864 液晶模块的显示分辨率为 128×64,其软件特性主要由控制驱动器决定。其中 ST7920 系列 LCM 接口灵活、指令简单、操作方便,既可显示 8×4 行 16×16 点阵汉字,又可完成图形显示,构成全中文的人机交互图形界面。由于内置了汉字库而无须人工取字模,且在程序控制上简洁明了,这个巨大的优势使得 ST7920 超越 KS0107 等其他类控制器逐渐成为市场的主流^[3]。

ST7920 是台湾矽创电子公司出品的一款中文图形控制芯片。它可提供 32 路行输出和 64 路列输出,具有 4 位/8 位并行、2 线/3 线串行的多种接口方式,带有国标一级、二级简体中文字库,内置 8 192 个 16×16 点阵汉字、128 个 16×8 点阵 ASCII 字符集和 64×256 的点阵显示 RAM (GDRAM)。此外,ST7920 内部还设计有 2 Mbyte 的中文字型 CGROM 和 64×256 点阵的 GDRAM 绘图区域;并提供 4 组可编程控制的 16×16 点阵造字空间。ST7920 系列 12864 液晶显示模块的原理框图如图 1 所示。单片 ST7920 只能控制输出 32COM×64SEG 点阵,ST7920 系列 12864 液晶模组采用 ST7920 作为 LCD 控制器,通过控制信号使内部的 ST7921 与之配合共同完成上下两屏 128×32 点阵的显示,即共同组成一个 128×64 的点阵模块。

2.2 基于 Nios II 的 SOPC 系统开发

基于 Nios II 的 SOPC 系统开发在逻辑语言和开发软件上使用的是 C/C++ 语言或者汇编语言的 Nios II IDE 集成开发环境、HDL 描述的 Quartus II 软件和 GUI 表格录入的 SOPC Builder

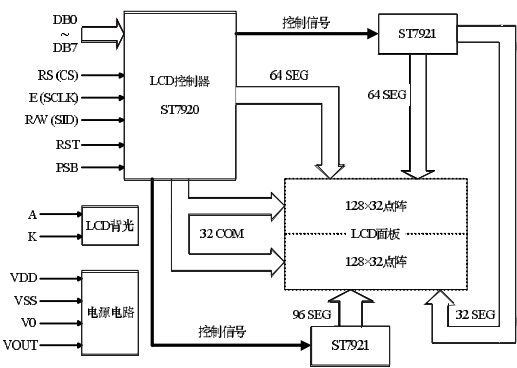


图 1 ST7920 系列 12864 LCM 原理框图

Fig. 1 Block diagram of ST7920 series 12864 LCM

工具。目前 Quartus 8.0 以上版本的 Quartus II 软件都将 SOPC Builder 集成在其内部。

图 2 是基于 Nios II 的 SOPC 系统开发流程图。其基本流程为:首先由 SOPC Builder 图形用户界面(GUI)生成并配置 Nios II 嵌入式软核处理器,选择并配置外设和 IP 核(用户既可以从标准库中添加也可以编写 HDL 代码自行定制指令与 IP 模块),连接各个模块生成用户定制化的 Nios II 处理器模块;在此过程中,SOPC Builder 将为硬件开发自动生成该模块的 HDL 源文件与 Test bench 文件,为软件开发自动生成与目标硬件相匹配的软件开发环境,包括生成一个相关的头文件和整个系统的仿真模型;而后在 Quartus II 中完成硬件开发,将 Nios II 处理器模块与用户代码及其他 IP 模块互联成一个完整的系统架构,进行

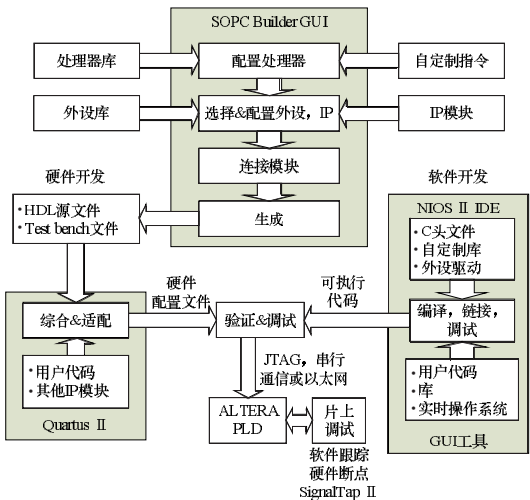


图 2 基于 Nios II 的 SOPC 系统开发流程图

Fig. 2 Flow chart of SOPC system development based on Nios II

综合与适配,产生硬件配置文件;在 Nios II IDE 中进行软件开发,根据系统应用的需求,利用 C/C++ 或汇编语言和系统所带的 API(应用程序接口)函数编写实现特定功能的程序并进行编译、链接与调试,生成与硬件系统无缝连接的可执行代码;最后通过下载电缆将硬件配置数据与软件执行代码下载到 FPGA 中,进行验证与调试。

3 硬件设计

3.1 基于 Avalon MM 接口的 LCD12864 IP 核硬件设计

由于 Altera SOPC Builder 并未提供本设计中所采用的 ST7920 系列 12864 液晶模块的 IP 核,因此需要运用自定义逻辑的方法在 SOPC 设计中添加 IP 模块以实现对其的接口与驱动。

Avalon 接口规范是 Altera 公司专门为 SOPC 环境下外设的开发而设计的,它定义了外设与 Avalon 开关互连结构(Avalon Switch Fabric)之间的数据传输。最新版本的 Avalon 接口规范^[4]定义了 6 种接口类型,分别为:

(1)Avalon-MM,一种典型的基于地址的主从结构读写接口;

(2)Avalon-ST,一种支持单向数据流传输的接口,包括多路复用数据流,包和 DSP 数据;

(3)Avalon Memory Mapped Tristate Interface,一种用于支持外设的基于地址的读写接口,多个外设可以共享数据和地址总线以减少对 FPGA 引脚的占用和 PCB 走线;

(4)Avalon Clock,一种驱动或接收 clock 和 reset 信号的接口,用于同步接口和提供 reset 连接;

(5)Avalon Interrupt,一种允许组件发送事件信号给其他组件的接口;

(6)Avalon Conduit,一种允许信号被传输出 SOPC Builder 系统的顶层之外,以连接至设计的其他模块或 FPGA 引脚的接口。

其中,Avalon-MM 接口是 Avalon Memory-Mapped 接口的简称,是用于在内存映射系统下描述主从元件的读/写接口。

本设计需要把 ST7920 系列 LCD12864 模块做成一个符合 Avalon-MM 接口规范的 Slave 设备,即在 SOPC 中添加基于 Avalon-MM 总线的 LCD12864 IP 核硬件模块。与传统的将 LCD 接

口当作普通外设进行 PIO 操作的方式相比,这种方式控制灵活,易于实现复杂的控制时序^[5],并且利用了总线的特性,具有 IP 核的“一次设计即可重复调用”的巨大优势,大大减少了重复工作,提高了设计效率。

具体实现步骤为:

(1)查阅 ST7920 数据手册^[6]和 12864 LCM 模组的说明文档^[7],理解分析管脚说明、控制逻辑、时序要求等,本设计选用 8 位并行接口方式实现对该系列 LCD12864 的驱动与图文显示。

(2)参考表 1 所示的“<接口类型>有效值”列表,Altera 公司最新公布的 Avalon 信号类型命名规则的建议为:

与指定接口相关的信号——<接口类型>_<接口名>_<信号类型>[_n]

其中“<接口名>”可以在同一条 Avalon 总线上拥有多个 Slave 端口时用于指示信号与端口的对应关系;“_n”可以附加上以指示该信号为低电平有效。遵循该规则进行信号命名的好处是,SOPC 中的元件编辑器将能够根据 HDL 源文件中的信号命名自动识别出信号类型与相应 Avalon 接口之间的关系。

表 1 <接口类型>有效值^[8]

Table 1 Valid values for <Interface Type>

Value	Meaning
avs	Avalon-MM slave
avm	Avalon-MM master
ats	Avalon-MM tristate slave
aso	Avalon-ST source
asi	Avalon-ST sink
cso	Clock output
csi	Clock input
coe	conduit
inr	Interrupt receiver
ins	Interrupt sender
ncm	Nios II custom instruction master
ncs	Nios II custom instruction slave

在 Quartus II 10.1 中依据 Verilog 2001 规范,用 Verilog HDL 语言编写生成符合这一命名规则的 ST7920 系列 LCD12864 的 Avalon MM Slave 接口描述文件,其 .V 文件格式下的完整代

码如下:

```
module ST7920_lcd12864 (
//Clock Input
input csi_clockreset_clk,
input csi_clockreset_reset_n,
// Avalon-MM Slave
input [1:0]avs_12864slave_address,
input avs_12864slave_read,
output [31:0] avs_12864slave_readdata,
input avs_12864slave_write,
input [31:0] avs_12864slave_writedata,
// lcd12864 interface, Conduit
output reg coe_e,
output reg coe_rs,
output reg coe_rw,
inout [7:0] coe_data_io
);
// 往 12864 写操作
reg [7 : 0] coe_lcd12864_data_out;
always@ (posedge csi_clockreset_clk, negedge
csi_clockreset_reset_n)
begin
if (! csi_clockreset_reset_n)
begin
coe_e <= 1'b0;
    coe_rs <=1'b0;
    coe_rw <=1'b0;
    coe_lcd12864_data_out <= 8'b0;
end
else if(avs_12864slave_write)
begin
    case(avs_12864slave_address)
0:coe_e
<= avs_12864slave_writedata[0];
    1:coe_rs
<= avs_12864slave_writedata[0];
    2:coe_rw
<= avs_12864slave_writedata[0];
    3:coe_lcd12864_data_out
<= avs_12864slave_writedata[7:0];
    endcase
end
end
```

```
// 从 12864 读操作
reg [7:0] readdata_r;
wire [7:0] coe_lcd12864_data_in;
always@(posedge csi_clockreset_clk)
if (avs_12864slave_read)
begin
if (avs_12864slave_address == 3)
readdata_r <= coe_lcd12864_data_in;
else
readdata_r <= 8'b0;
end
else
readdata_r <= 8'b0;
assign avs_12864slave_readdata = {24'b0,
readdata_r};
// 双向口数据流控制
reg coe_lcd12864_data_out_en;
always@(posedge csi_clockreset_clk)
if (avs_12864slave_write)
coe_lcd12864_data_out_en <= 1'b0;
else if (avs_12864slave_read)
coe_lcd12864_data_out_en <= 1'b1;
assign coe_lcd12864_data_in = coe_data_io;
assign coe_data_io =
coe_lcd12864_data_out_en ?
8'bz :coe_lcd12864_data_out;
endmodule
```

其中由于 Avalon MM 的数据总线宽度为 32 位,因此文件中的 readdata 和 writedata 均设置为 32 位,通过在前补零和读取低位的方式实现 ST7920 系列 12864 液晶外设的 8 位数据与 Avalon MM Slave 的数据交互,也可以将 readdata 和 writedata 设置为 8 位,这需把地址对齐方式改为静态对齐。

(3)编制完成. v 格式的符合 Avalon MM 规范的接口描述文件之后,就可以在 SOPC Builder 图形用户界面中添加此组件,完成 IP 核的硬件设计了。具体操作步骤为:双击 Altera SOPC Builder 中的 New Component...,在 HDL Files 标签下添加所建立的. v 格式接口描述文件,系统将自动对文件进行分析读取 IO 信号与参数信息;由于所编制的 HDL 文件完全符合规范,因此各项设置均会自动生成,可以直接点击“Library

Info”标签填写 IP 核信息完成硬件设计部分,如有特殊时序或设置要求,可以点击“Interface”标签进行修改。图 3 所示为由规范化的接口描述文件自动生成的自定义逻辑信号信息。

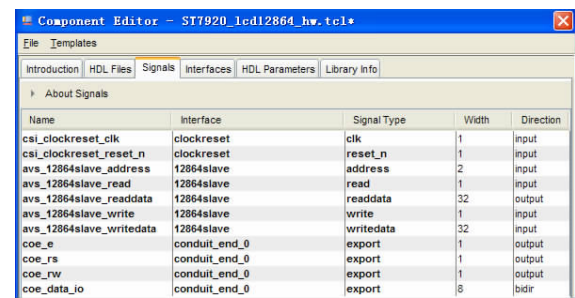


图 3 自动生成的自定义逻辑信号

Fig. 3 Customized logic signals automatically generated by ST7920_lcd12864. v

至此,完成了基于 Avalon MM Slave 接口的 LCD12864 IP 核硬件设计部分。生成的 LCD12864 SOPC IP 组件包含 ST7920_lcd12864. v 和 ST7920_lcd12864_hw.tcl 两个文件,将它们拷贝至任意新建工程的文件夹中即可实现对该 IP 核组件的复用。

3.2 系统硬件设计

图 4 显示了本设计构建的含有 ST7920_lcd12864 IP 的 Avalon 系统架构框图。定制封装化的 ST7920_lcd12864 IP 组件可以在 SOPC Builder 环境中像其他标准组件(如 JTAG UART、SDRAM Controller 等)同样使用。

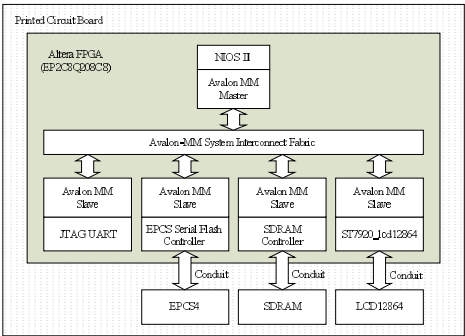


图 4 含有 ST7920_lcd12864 IP 的 Avalon 系统框图

Fig. 4 Avalon system chart with ST7920_lcd12864 IP

图 5 显示了由 SOPC Builder 创建的自定义的 NiosII 配置及其地址映射。添加配置所需组件并进行地址与中断分配后即可进一步生成 Nios II 软核。

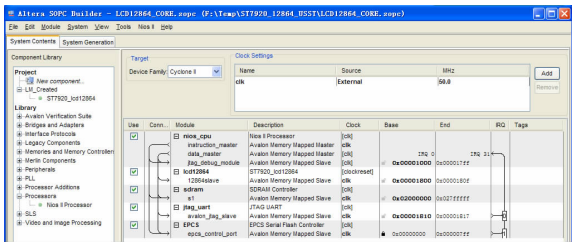


图 5 Nios II 软核配置及其地址映射

Fig. 5 Configuration and address mapping of Nios II

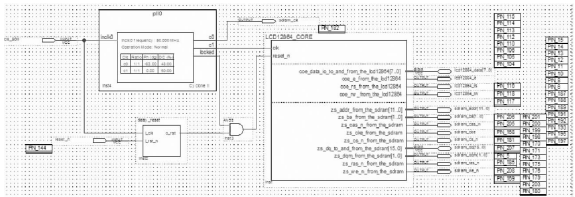


图 6 整体硬件设计原理图

Fig. 6 Top-level block diagram of hardware design

完成 NiosII 软核的创建后,在 Quartus II 10. 1 中新建一个原理图文件作为顶层模块,双击空白的原理图界面,在弹出的 Symbol 窗口的 Libraries->Project 目录下选择即可调入自定义创建的 LCD12864_CORE 软核,添加输入输出引脚和 PLL 模块、延时复位(delay_reset)模块等基本单元,将它们互连成完整的硬件系统并进行引脚锁定,在 Quartus II 中编译通过。最终建立的硬件设计原理图如图 6 所示。其中外部时钟输入为 50 MHz;delay_reset 模块实现 300 ms 延时复位,起到按键消抖的作用;SDRAM 采用的是 HY57V1620, SDRAM 时钟相对于控制时钟的相位偏移设置为 -63°,其计算公式如下: Read Lag = $t_{OH}(\text{SDRAM}) - t_{H_MAX}(\text{FPGA})$ Write Lag = $t_{CLK} - t_{CO_MAX}(\text{FPGA}) - t_{DS}(\text{SDRAM})$ Read Lead = $t_{CO_MIN}(\text{FPGA}) - t_{DH}(\text{SDRAM})$ Write Lead = $t_{CLK} - t_{HZ}(\text{SDRAM}) - t_{SU_MAX}(\text{FPGA})$ Maximum Lag = min imun(Read Lag, Write Lag) Maximum Lead = min imun(Read Lead, Write Lead) Maximum Lead = min imun(Read Lead, Write Lead)

最后的 SDRAM 时钟相移理论值由(Maximum Lag + Maximum Lead)/ 2 得到,即相移值取数据读和写有效的最小区间的中间值。其中计算所需的各项参数可以通过查阅 HY57V1620 和 EP2K8Q208C8 的 Datasheet 得到,FPGA 相关的最精确的时序信息也可以通过查看 Quartus II 编

译报告的时序分析器得到。这一计算结果的前提是假设 PCB 延迟和从 PLL 时钟输出节点到目标处的延迟引起的相位偏移均为 0,实际设定相移值时需将它们的影响进行综合考虑加以微调。可以从参考文献[9]得知更详尽的设计描述。

4 软件设计

4.1 创建软件工程

SOPC 硬件系统构建完成之后,便可在 Nios II IDE 中建立软件工程,进行应用程序开发。其具体步骤如下:打开 Nios II IDE 10.1,点击 File -> Switch Workspace 创建并指向 Nios 软件工程文件夹;在工程向导的指引下建立 C/C++ 应用程序工程(Nios II C/C++ Application)并为其指定目标硬件的 PTF 文件,该文件是在之前编译 SOPC 硬件系统时产生的;进一步即可自动产生系统库工程(System Library),而后对空白工程进行第一次编译产生 system.h 文件;该文件在 Nios_12864_syslib/Debug/ system_description/ 目录下,编译器根据之前硬件设计时用户自定义的软核架构生成的相应寄存器,如各个设备的基地址、中断号等,均体现在此文件中,如图 7 所示。利用这些地址对 Nios 软核进行寄存器操作,就可以达到与单片机相同的寄存器操作方式。

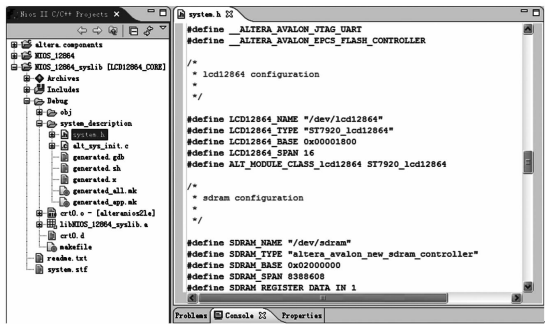


图 7 System.h 文件包含有与硬件架构无缝对接的设备寄存器信息

Fig.7 System.h file contains the information of device registers which build a seamless connection between hardware architecture and software design

4.2 LCD12864 IP 核的驱动设计

LCD12864 IP 核的驱动设计部分包含驱动宏文件 ST7920_lcd12864.h 和驱动源文件 ST7920_lcd12864.c 的编制。

(1) ST7920_lcd12864.h 的内容包括参照 system.h 编写的与 IP 核 HDL 硬件设计相关的宏定义、基地址定义和功能函数声明,例如:

```
#define lcd12864_addr LCD12864_BASE;
// 这是对 LCD12864 基地址的宏定义。
```

```
#define IOWR_LCD12864_RW(base, data)
IOWR(base, 2, data), 是根据 HDL 文件
ST7920_lcd12864.v 编写的寄存器映射的宏定义。其中 IOWR()是 Nios II 提供的写函数,base
是基地址,2 为地址偏移量,data 为要写入的数据,结合 ST7920_lcd12864.v 可知,当地址偏移量
为 2 时(传递给 ST7920_lcd12864.v 中的形参 avs_12864slave_address 值为 2),即实现了对 RS 引脚
的数据写入。其他寄存器的映射宏定义编写与此类似。Nios II 提供的寄存器读写函数的定义包含
在系统自带的 io.h 头文件中。
```

类似地,管脚操作的宏定义格式举例如下,分别表示将 RS 置 1,向 LCD12864 写数据,从 LCD12864 读数据:

```
#define SET_RS
IOWR_LCD12864_RS(lcd12864_addr, 2)
#define WRITE_DATA(data)
IOWR_LCD12864_DATA(lcd12864_addr, data)
#define READ_DATA
IORD_LCD12864_DATA(lcd12864_addr)
```

又如,extern void ST7920_WriteCommand(alt_u8 cmd);是向 LCD12864 写指令的功能函数声明。

(2)ST7920_lcd12864.c 中包含根据 ST7920 数据手册^[6]和 12864 LCM 模组的说明文档^[7]编写的功能函数,以下是其中忙碌状态检测、写函数与初始化函数的例子:

```
void ST7920_CheckBusy(void)
{
    // 忙碌状态检测
    CLEAR_RS; // RS=0,指令操作
    SET_RW; // RW=1,读操作
    SET_E; //E=1,操作使能
    while((READ_DATA&0x80) == 0x80);
    // 检测忙碌标志位
    CLEAR_E; //操作结束
    usleep(72); // Wait 72µs
}

void ST7920_WriteCommand(alt_u8 cmd)
```

```
{
    // 写指令函数
    ST7920_CheckBusy();
    CLEAR_RS;    // RS=0,指令操作
    CLEAR_RW;    // RW=0,写操作
    SET_E;
    WRITE_DATA(cmd);
    CLEAR_E;
    usleep(72);    // Wait 72μs
}

void ST7920_WriteData(alt_u8 data)
{
    // 写数据函数
    ST7920_CheckBusy();
    SET_RS;    // 数据操作
    CLEAR_RW;    // 写操作
    SET_E;
    WRITE_DATA(data);
    CLEAR_E;
    usleep(72);    // Wait 72μs
}

void ST7920_Init(void)
{
    usleep(40 * 1000); // Wait 40ms
    ST7920_WriteCommand(0x32);
//Function Set
    usleep(39);
    ST7920_WriteCommand(0x36);
//Function Set
    usleep(37);
    ST7920_WriteCommand(0x0C);
    //整体显示开,光标关,反白关
    usleep(100);    //Wait 100μs
    ST7920_WriteCommand(0x01);
    //Display Clear
}
```

4.3 LCD12864 图文显示的实现

建立了基于 Nios II IP 核技术的 SOPC 软硬件系统后,通过编写特定功能函数并在主程序中调用,用户能够方便地实现 LCD12864 的图文显示。

其中,图形显示函数的 C 语言代码如下:

```
void Lcd12864_drawpic(unsigned char * p)
{ int row_p1, seg_p1, row_p2, seg_p2;
  for(seg_p1=0; seg_p1<0x20; seg_p1++)
```

```
{for(row_p1=0; row_p1<0x08; row_p1++)
  {ST7920_WriteCommand(0x80+seg_p1);
   //写列地址
   ST7920_WriteCommand(0x80+row_p1);
   //写行地址(12864 上半屏)
   ST7920_WriteData(* p++);
   //写 GDRAM 的高八位数据
   ST7920_WriteData(* p++);
   //写 GDRAM 的低八位数据}
}

for(seg_p2=0; seg_p2<0x20; seg_p2++)
{ for(row_p2=0; row_p2<0x08; row_p2++)
  {ST7920_WriteCommand(0x80+seg_p2);
   //写列地址
   ST7920_WriteCommand(0x88+row_p2);
   //写行地址(12864 下半屏)
   ST7920_WriteData(* p++);
   //写 GDRAM 的高八位数据
   ST7920_WriteData(* p++);
   //写 GDRAM 的低八位数据}
}
}
```

主函数调用程序为:

```
int main()
{
    ST7920_Init(); // 初始化 Lcd12864
    Lcd12864_drawpic(buf); //调用绘图函数
    usleep(1000 * 1000);
    return 0;
}
```

实验效果如图 8 所示。



图 8 LCD12864 的图形显示

Fig. 8 Graphical display example of LCD12864

类似地,也可以根据 ST7920 数据手册,利用建立好的 LCD12864 的 SOPC IP 核和 ST7920 内

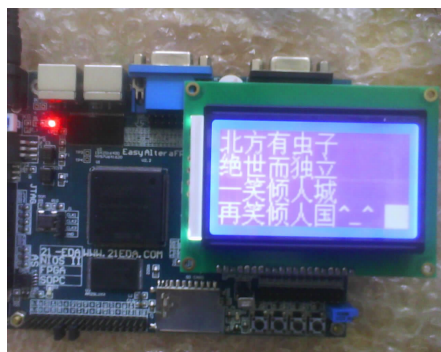


图 9 LCD12864 的字符显示

Fig. 9 Character display example of LCD12864

置汉字库与字符集,编制出字符显示的应用程序。依据本设计中 LCD12864 IP 核的特点而编制出的 LCD12864 字符显示程序实验效果如图 9 所示。

5 结 论

由于所设计的 ST7920 系列 LCD12864 IP 核内已经建立好符合 Avalon MM 总线规范硬件架构和符合 ST7920 时序要求的读写函数及其他功能模块,开发者不需要再编写底层驱动,大大减少了重复劳动,提高了设计效率。所论述的设计方法也适用于基于 SOPC IP 核技术的其他液晶显示系统的开发。

参 考 文 献:

- [1] 林振营, 马刚. 基于 Nios II 的 SOPC 嵌入式系统开发解决方案 [J]. 现代电子技术, 2007, 30(6):79-81.
- [2] 方苗, 陈卓文, 彭澄廉. SOPC 设计中的用户自定义逻辑 [J]. 计算机工程, 2004, 30(17):42-44.
- [3] 舒鑫, 郭其一. ST7920 点阵式液晶显示模块的应用研究和编程 [J]. 液晶与显示, 2007, 22(2):192-196.
- [4] Altera Corporation. Avalon Interface Specifications Version 1.3 [EB/OL]. [2010-8-01]. http://www.altera.com.cn/literature/manual/mnl_avalon_spec.pdf.
- [5] 侯长宏, 袁慧梅. 基于 SOPC 的 Nios II 与液晶模块接口及程序设计 [J]. 液晶与显示, 2008, 23(3):307-311.
- [6] 台湾矽创电子公司. ST7920 Specification Version 4.0 [EB/OL]. [2008-08-18]. [http://www.sitronix.com.tw/sitronix/SASpecDoc.nsf/FileDownload/ST7920713600/\\$FILE/ST7920V40_E.pdf](http://www.sitronix.com.tw/sitronix/SASpecDoc.nsf/FileDownload/ST7920713600/$FILE/ST7920V40_E.pdf).
- [7] 重庆育松电子. 液晶显示模块 QC12864B 使用说明书 [R]. 重庆, 2008.
- [8] Altera Corporation. SOPC Builder User Guide Version 1.0 [EB/OL]. [2010-12-01]. http://www.altera.com.cn/literature/ug/ug_sopc_builder.pdf.
- [9] Altera Corporation. Embedded Peripherals IP User Guide [EB/OL]. [2010-12-01]. http://www.altera.com.cn/literature/ug/ug_embedded_ip.pdf.