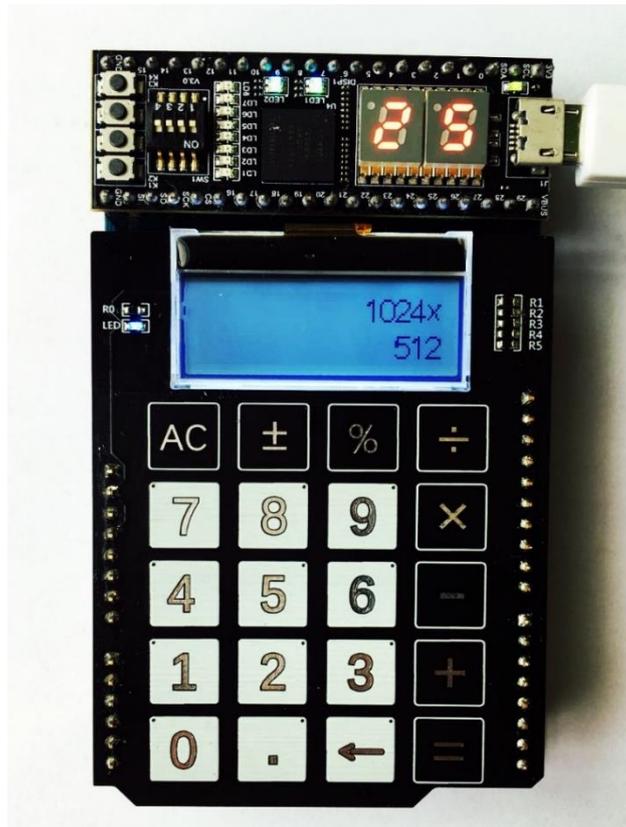


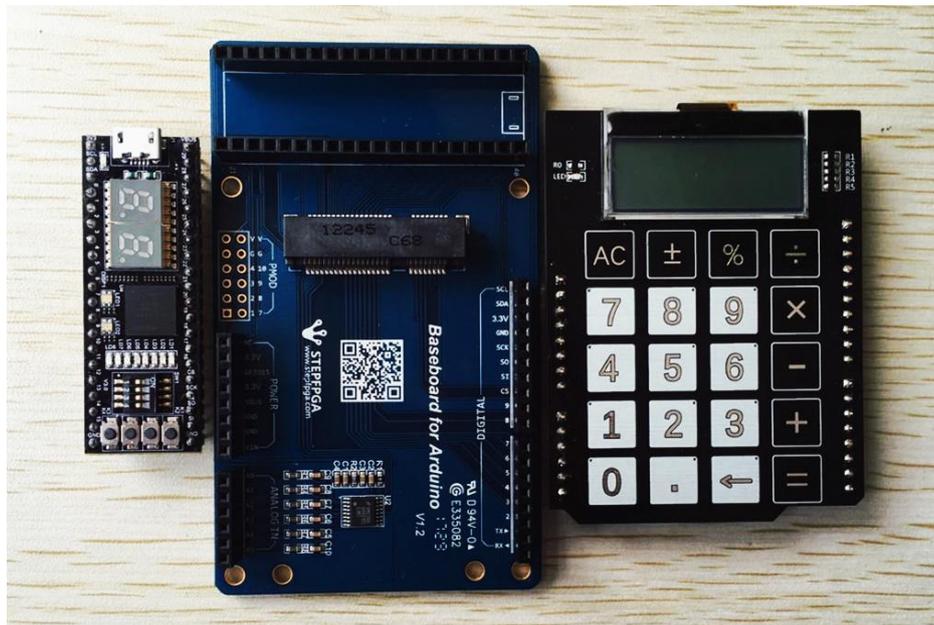
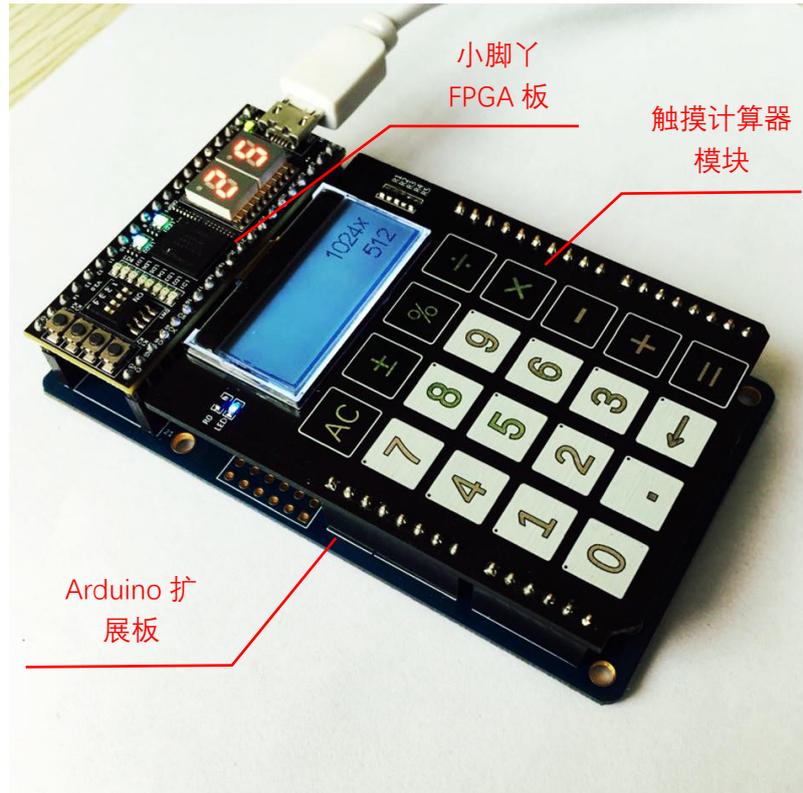
# 基于 STEP FPGA 的计算器模块用户开发手册

## 1. 计算器模块概述

小脚丫 FPGA 计算器模块是 STEP 团队推出的可配合小脚丫 FPGA 核心板和底板使用的扩展板卡, 基于该板卡可以实现一个计算器的触摸输入和基本计算功能。



触摸计算器模块需要配合小脚丫 FPGA 核心板(所有小脚丫版本都可以适用)和 Arduino 扩展底板使用。小脚丫核心板作为控制核心, Arduino 扩展板用来连接核心板和触摸计算器模块。



## 2. 计算器模块结构组成

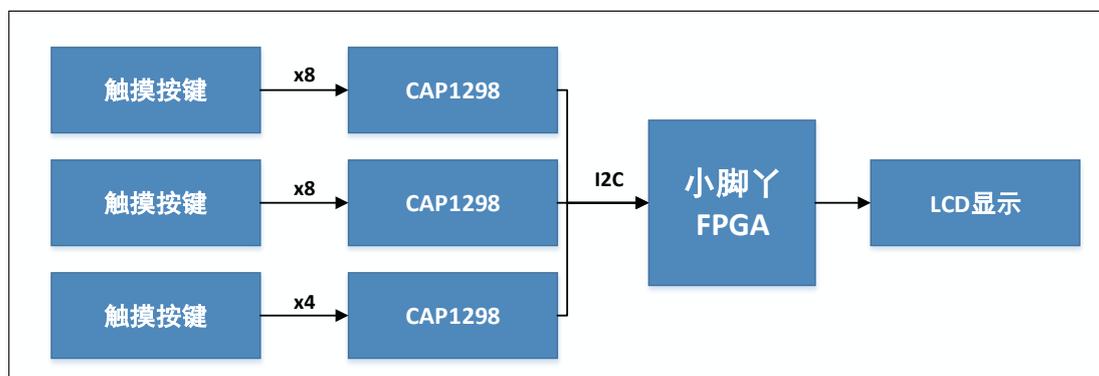
计算器模块结构如下图所示，主要由触摸按键、触摸检测芯片 CAP1298、128\*32 LCD 显示屏和 DIP 接口等四部分组成。



该模块 PCB 做有 20 个触摸按键，其中触摸检测芯片选用了 Microchip 的电容触摸控制器 CAP1298，该触摸芯片具有 8 路触摸按键检测功能，因此采用了 3 片 CAP1298 来检测 20 个触摸按键的状态。CAP1298 与 FPGA 采用 I2C 总线的通信方式传输按键状态。

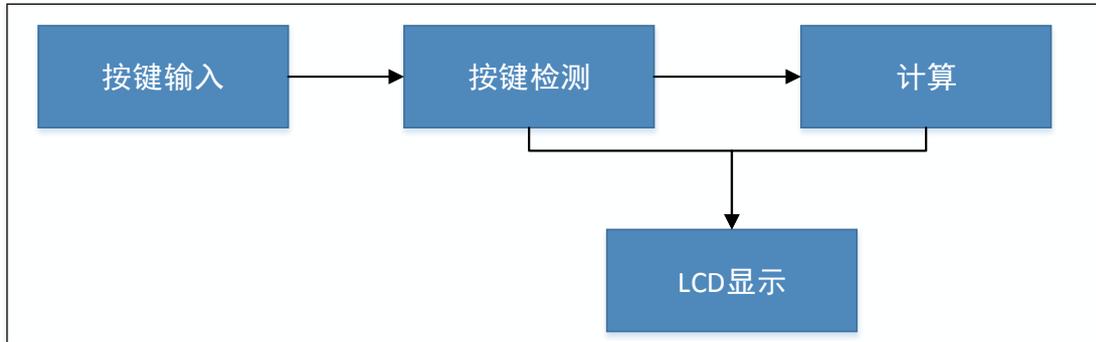
显示屏选用了晶联讯电子公司的 128\*32 的单色点阵液晶模块 JLX12832G-520，内置驱动芯片，FPGA 可以通过串行接口驱动该模块显示 16\*16 汉字或 16\*8 字符。

模块采用了 Arduino 的标准接口，完全兼容 Arduino UNO 接口。

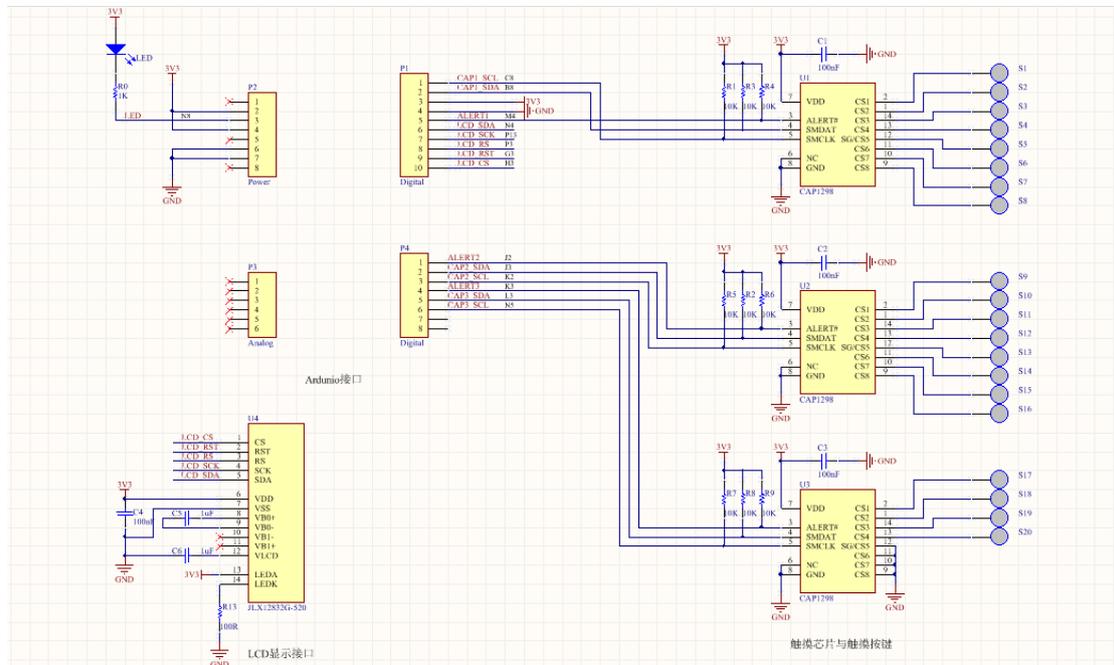


### 3.功能设计

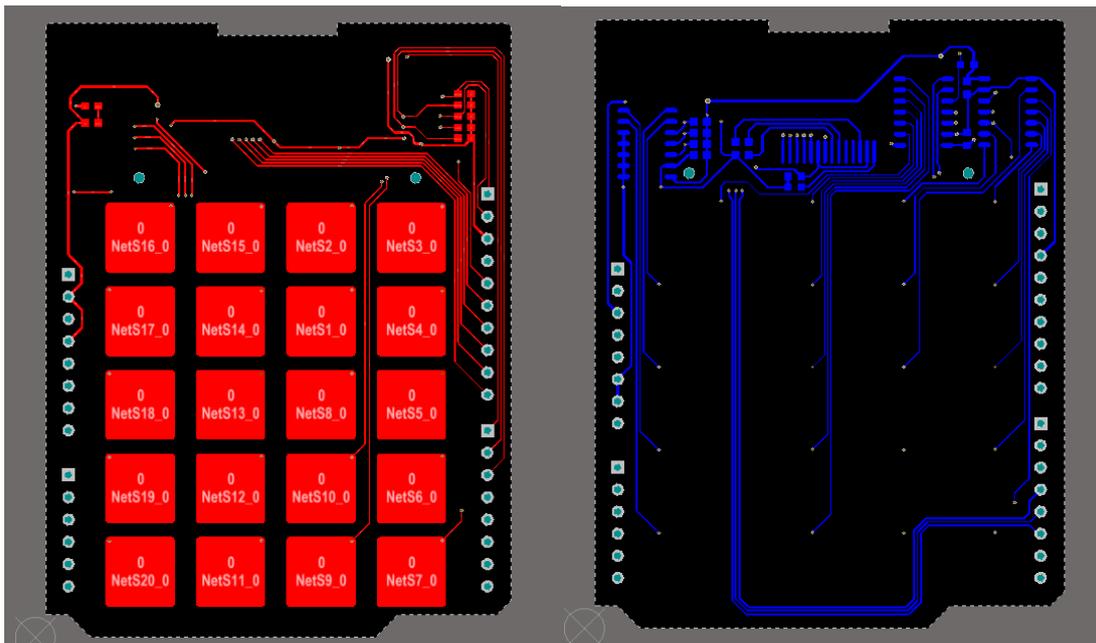
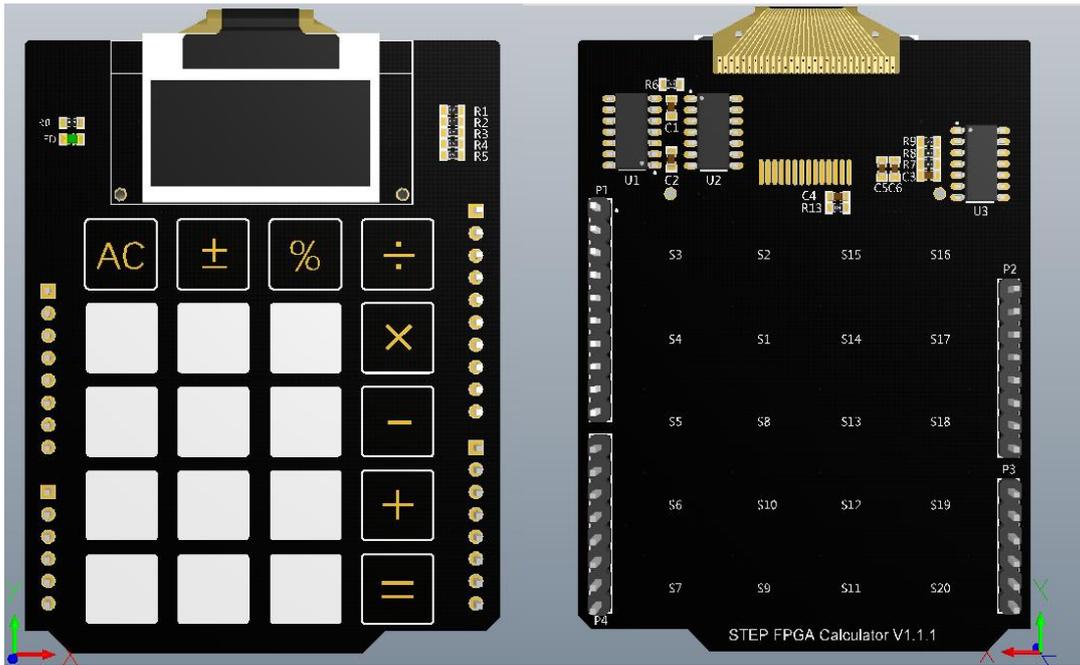
该模块配合小脚丫 FPGA 的核心板和 Arduino 扩展底板可以实现一个简易计算器的加减乘除功能，LCD 采用 8\*8 字符显示可以至少显示 12 位数字。



### 4.原理图说明



### 5.PCB 布局布线

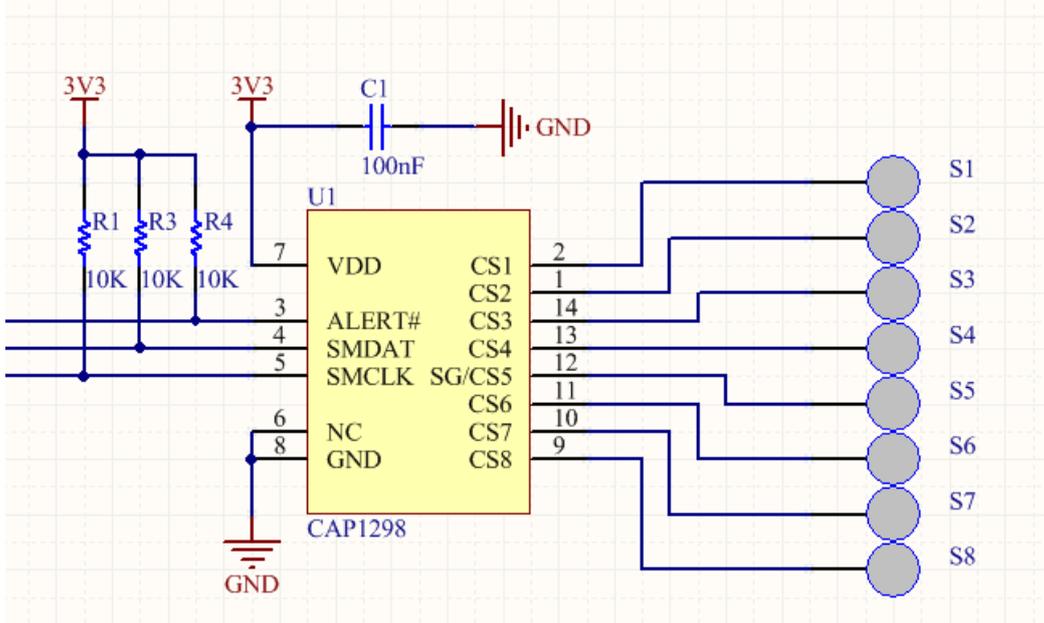


## 6.设计说明

### 1) 触摸检测

触摸按键检测部分采用 Microchip 公司的电容触摸控制器 CAP1298，该芯片内置电容触摸检测和校准功能，每个芯片可以独立检测 8 路的触摸按键容值变化

并将检测结果保存在内部的寄存器中，可以通过 SMBUS 或标准的 I2C 总线来读写芯片内的寄存器值。



芯片内部有众多的参数设置和状态寄存器，但是大部分寄存器在出厂时已经设置了默认值，一般情况下无需重新设置就可以使用。我们只需要操作 00h Main Control 寄存器和 03h Sensor Input Status 两个寄存器就可以实现触摸按键的检测工作。

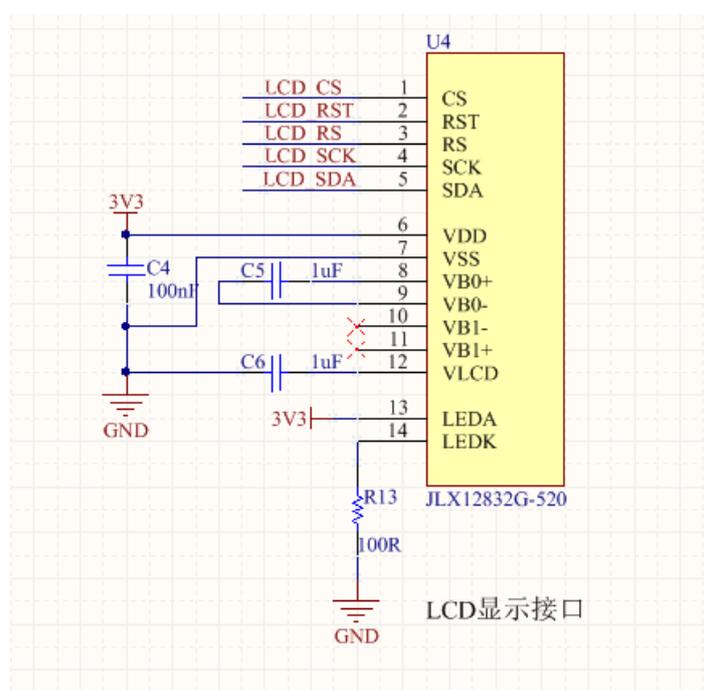
Register Address	R/W	Register Name	Function	Default Value	Page
00h	R/W	Main Control	Controls power states and indicates an interrupt	00h	Page 26
02h	R/W	General Status	Stores general status bits	00h	Page 28
03h	R	Sensor Input Status	Returns the state of the sampled capacitive touch sensor inputs	00h	Page 28
0Ah	R	Noise Flag Status	Stores the noise flags for sensor inputs	00h	Page 29

Main Control 寄存器控制芯片的电源状态和指示中断状态。Sensor Input Status 寄存器返回采样电容触摸传感器输入值，每一位对应一个触摸按键通道的返回值，当一个通道的触摸按键被触摸时，03h 寄存器对应位会被置 1，同时产生中断，当 01h 主控制器的 INT 位被清零时，该状态位会被清零。

FPGA 使用 I2C 总线操作步骤：首先读取 03h 地址寄存器值，如果 03h 寄存器的值不是 0，则表示有对应位的按键被触摸，然后向 00h 地址寄存器写 00h，则将 03h 清零，以便下一次检测。

## 2) 显示模块

显示模块选用了晶联讯电子公司的 128\*32 的单色点阵液晶模块 JLX12832G-520，内置矽创公司的驱动芯片 ST7567，FPGA 可以通过串行接口驱动该模块显示 16\*16 汉字或 16\*8 字符。



CS 是 LCD 的片选信号，低电平有效，CS 信号为低时 LCD 可以被驱动。

RST 是复位信号，低电平有效，上电时需要将该信号置低以对 LCD 复位。

SCK 和 SDA 分别是串行时钟和串行数据，符合 SPI 时序。

RS 是数据和命令选择信号，RS 为低时，表示写指令到 LCD，RS 为高时，表示写数据到 LCD。

从 CPU 写到 ST7567 (Writing Data from CPU to ST7567)

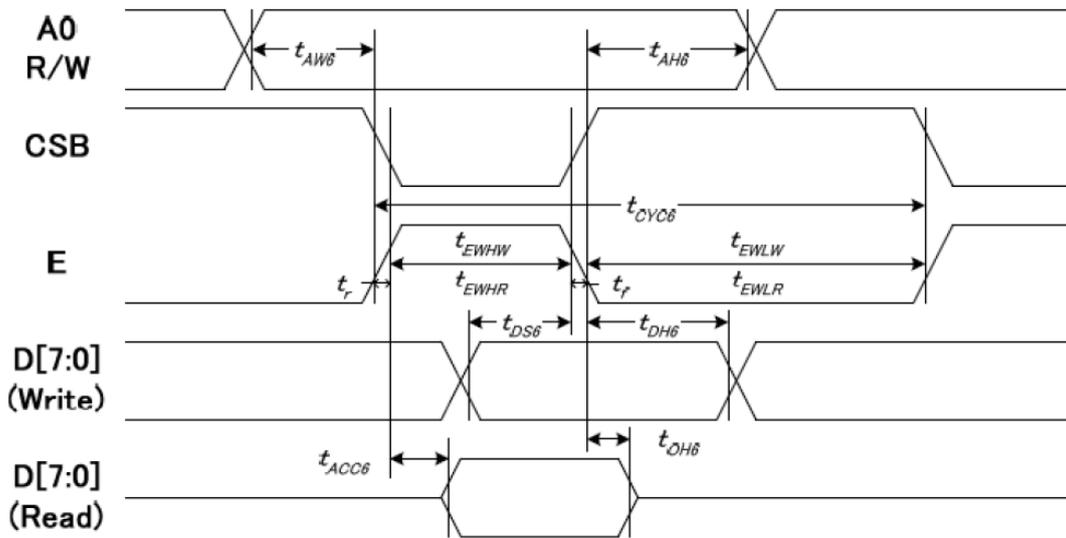


图 4. 从 CPU 写到 ST7567 (Writing Data from CPU to ST7567)

该显示屏使用下面指令表中的指令来完成驱动芯片的设置。

指令表

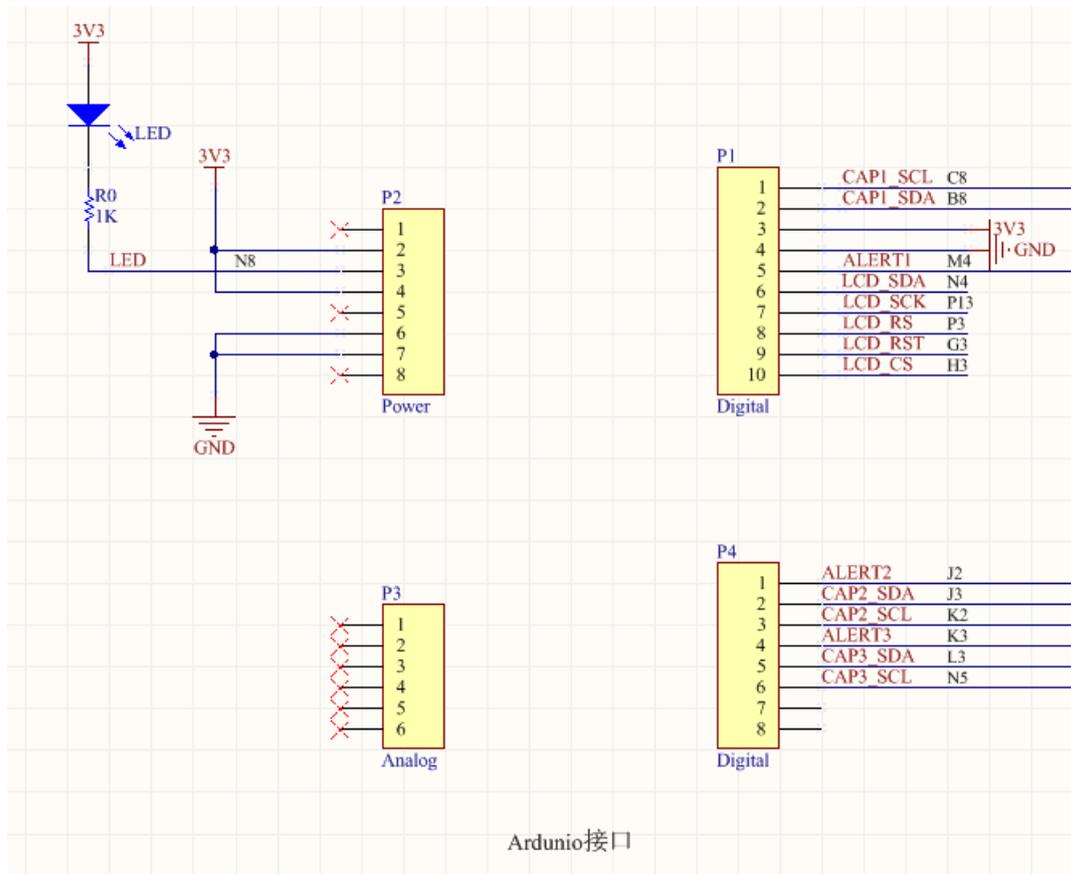
表 8.

指令名称	指令码									说明	
	RS	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
(1) 显示开/关 (display on/off)	0	1	0	1	0	1	1	1	0	显示开/关: 0XAE: 关, 0XAF: 开	
(2) 显示初始行设置 (Display start line set)	0	0	1	显示初始行地址, 共 6 位						设置显示存储器的显示初始行, 可设置值为 0X40~0X7F, 分别代表第 0~63 行, 针对该液晶屏一般设置为 0x40	
(3) 页地址设置 (Page address set)	0	1	0	1	1	显示页地址, 共 4 位				设置页地址。每 8 行为一个页, 64 行分为 8 个页, 可设置值为: 0XB0~0XB8 分别对应第一页到第九页, 第九页是一个单独的一行图标, 本液晶屏没有这一行图标, 所以设置值为 0XB0~0XB7 分别对应第一页~第八页。	
(4)	列地址高4位设置	0	0	0	0	1	列地址的高 4 位				高 4 位与低 4 位共同组成列地址, 指定 128 列中的其中一列。比如液晶模块的第 100 列地址十六进制为 0x64, 那么此指令由 2 个字节来表达: 0x16, 0x04
	列地址低4位设置		0	0	0	0	列地址的低 4 位				
(5) 读状态 (Status read)	0	状态				0	0	0	0	串口时: 读驱动 IC 的当前状态, 串口时不能用此指令	
(6) 写显示数据到液晶屏 (Display data write)	1	8 位显示数据									从 CPU 写数据到液晶屏, 每一位对应一个点阵, 1 个字节对应 8 个竖置的点阵
(7) 读液晶屏的显示数据 (Display data read)	1	8 位显示数据									串口时: 读已经显示到液晶屏上的点阵数据。串口时不能用此指令

### 3) 接口

该计算器模块需要小脚丫 FPGA 核心板和 Arduino 底板配合使用, 扩展接口

按照 Arduino 接口标准设计, 所以该板也可以配合 Arduino 开发板使用。



Arduino接口

## 7.引脚分配

设备	Net name	Arduino Pin	FPGA Pin	备注
CAP1298-1	CAP1_SCL	SCL	C8	CAP1298-1 I2C 时钟
	CAP1_SDA	SDA	B8	CAP1298-1 I2C 数据
	ALERT1	D13	M4	CAP1298-1 中断输出
CAP1298-2	CAP2_SCL	D5	K2	CAP1298-2 I2C 时钟
	CAP2_SDA	D6	J3	CAP1298-2 I2C 数据
	ALERT2	D7	J2	CAP1298-2 中断输出
CAP1298-3	CAP3_SCL	D2	N5	CAP1298-3 I2C 时钟
	CAP3_SDA	D3	L3	CAP1298-3 I2C 数据
	ALERT3	D4	K3	CAP1298-3 中断输出
LCD	LCD_SDA	D12	N4	LCD 数据线
	LCD_SCK	D11	P13	LCD 时钟线

	LCD_RS	D10	P3	LCD 数据/命令选择
	LCD_RST	D9	G3	LCD 复位信号
	LCD_CS	D8	H3	LCD 片选信号
LED	LED	RESET	N8	指示灯

8.版本

V1.0-2017.10.18 修改