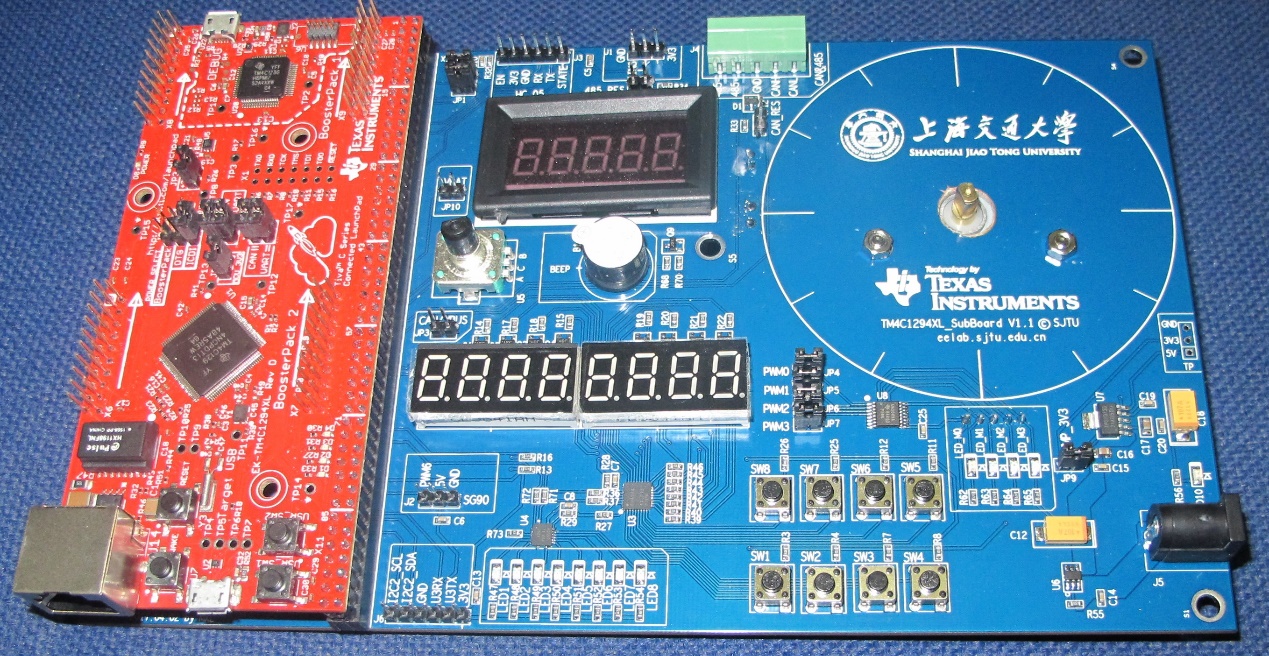
1. S800板介绍



***S800板 = TM4C1294XL(红) + TM4C1294XL\_SUB(蓝)***

红色板为TI提供的基于TM4C1294NCPDT（ARM CORTEX-M4F内核, 128PIN）CPU的评估板;蓝色板为扩展功能板,提供实验的各种对象及执行部件.

扩展板提供的功能如下:

整板静电ESD保护及过电流负载保护

整板电流显示

多路可选电源输入,包括

DC5V

MICRO-USB5V输入

MICRO-USB OTG 5V输入

I2C扩展GPIO

8位共阴数码管

8位输入按键

8位共阴LED

USART- RS485总线串行接口

CAN总线串行接口

5V直流有刷电机或步进电机接口

PWM输出

DAC输出

外部模拟量输入

可调电位器模拟量输入

蜂鸣器

SD卡接口

QEI数字电位器接口

* 蓝牙模块接口
* 舵机控制接口
* RTC备用电池

1. 板上芯片简介

红板芯片

|  |  |  |
| --- | --- | --- |
| 序号 | 芯片型号 | 说明 |
| 1 | TM4C1294NCPDT | 主CPU，ARM-CORTEX-M4F |
| 2 | TM4C123GH6PMI | ICDI板载仿真器CPU，，ARM-CORTEX-M4F |
| 3 | TPD4S012 | 4通道ESD保护用于USB接口保护 |
| 4 | TPS2052B | 过流保护及负载分配开关 |
| 5 | TPS73733 | 3.3VLDO稳压器 |

蓝板芯片

|  |  |  |
| --- | --- | --- |
| 序号 | 芯片型号 | 说明 |
| 1 | PCA9557 | I2C扩展8位GPIO |
| 2 | TCA6424 | I2C扩展24位GPIO |
| 3 | TPD3S044 | USB接口限流保护及ESD保护 |
| 4 | TPD4E001 | 4通道ESD保护 |
| 5 | SN65HVD08 | RS-485收发器 |
| 6 | SN65HVD1050 | CAN总线收发器 |
| 7 | ULN2003A | 7位达林顿管阵列 |
| 8 | S8050D | 40V0.5A0.3W NPN管 |
| 9 | EC12 | 数字电位器 |
| 10 | CL3641AH | 4位共阴数码管 |

1. 按键及LED资源列表

红板

|  |  |  |
| --- | --- | --- |
| 名称 | 对应管脚 | 说明 |
| RESET | RESET | TM4C1294NCPDT芯片复位按键低有效 |
| WAKE | WAKE | 从睡眠模式唤醒按键低有效 |
| USR\_SW1 | PJ0 | 用户输入按键低有效 |
| USR\_SW2 | PJ1 | 用户输入按键低有效 |
| D0 |  | 3.3V电源指示，绿LED高有效 |
| D1 | PN1 | 用户控制绿LED高有效 |
| D2 | PN0 | 用户控制绿LED高有效 |
| D3 | PF4 | 用户控制绿LED高有效 |
| D4 | PF0 | 用户控制绿LED高有效 |

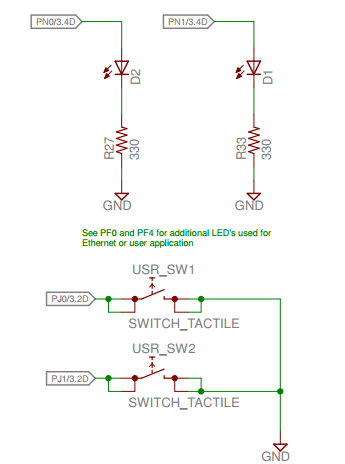
蓝板

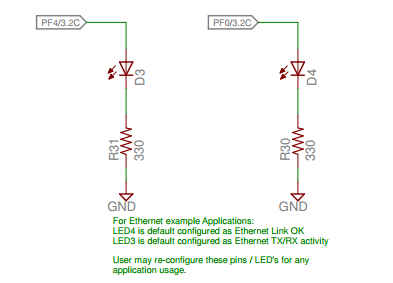
|  |  |  |
| --- | --- | --- |
| 名称 | 对应管脚 | 说明 |
| SW1-SW8 | TCA6424-P01~P08 | TCA6424 I2C展GPIO芯片P0口低有效 |
| LED1-LED8 | PCA9557-P0-P7 | PCA9557 I2C展GPIO芯片P0口低有效 |
| LED\_M0 | PF0 | 用户控制LED低有效 |
| LED\_M1 | PF1 | 用户控制LED低有效 |
| LED\_M2 | PF2 | 用户控制LED低有效 |
| LED\_M3 | PF3 | 用户控制LED低有效 |
| D10 |  | 3.3V电源指示，红LED高有效 |

* 红板与蓝板均有PF0对应控制LED，两个LED一绿（高有效）一红（低有效），显示效果互补。

1. 板上器件编程与使用

4.1 红板上资源编程





红板上共有两个外部晶体振荡器供选择，一个为外部25M高精度无源晶振HSE，为正常使用时提供外部时钟信号；一个为32.768K无源晶振LSE，主要用于低功耗或电池供电时提供外部时钟信号。

同时在CPU内部也提供一个低精度的16M振荡器HSI（正负50%误差），可以用于不需要精确定时的场合。不能用于同步通讯如USART，CAN，ETHERNET等。

如果不使用PLL倍频电路，则只能使用25M的HSE或16M的HSI作为时钟。

如果使用PLL倍频电路，则可以升频到480M后再分频作为CPU时钟使用，最大可接受频率为120M时钟。

频率越高，则速度越快，功耗越大。

红板上共有四个用户可控LED，如上图所示。分别为PF0,PF4,PN0,PN1管脚控制。只有当管脚输出高电平时，LED才会亮。

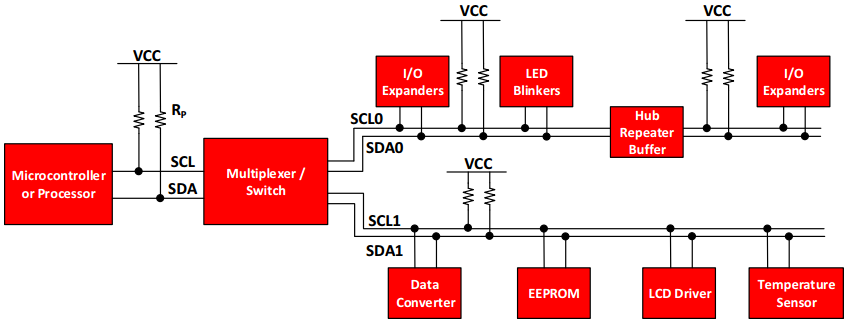
红板上共有两个用户可控按键输入，如上图所示。分别为USR\_SW1，USR\_SW2接到PJ0与PJ1，当按键按下时，与地相通。这两个引脚并没有外接上拉电阻，因此作为输入使用时，必须配置为内部弱上拉，才能清楚地分开未按下与按下状态。

4.2 蓝板上资源编程

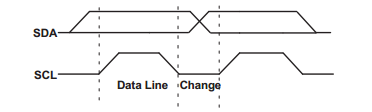
4.2.1 I2C通讯简述

I2C总线是一种简单的双向一线制串行通信总线。多个符合I2C总线标准的器件都可以通过同一条I2C总线进行通信，而不需要额外的地址译码器。

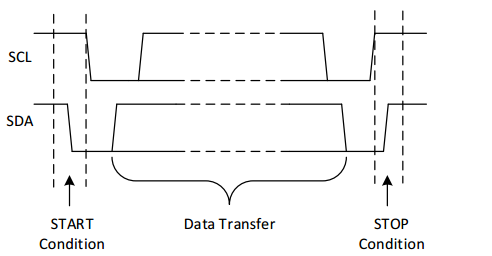
I2C总线仅使用两个信号：SDA和SCL。SDA是双向串行数据线，SCL是双向串行时钟线。当SDA和SCL线为高电平时，总线为空闲状态。I2C总线连接方式如下图所示。



在时钟SCL的高电平期间，SDA线上的数据必须保持稳定。SDA仅可在时钟SCL为低电平时改变，如下图所示。



I2C总线的协议定义了两种状态：起始和停止。当SCL为高电平时，在SDA线上从高到低的跳变被定义为起始条件；而当SCL为高电平时，在SDA线上从低到高的天边则被定义为停止条件。总线在起始条件之后被看作为忙状态。总线在停止条件之后被看作为空闲。



SDA线上的每个字节必须为8位长。不限制每次传输的字节数。每个字节后面必须带有一个应答位。数据传输是MSB在前。当接收器不能接受另一个完整的字节时，它可以将时钟线SCL拉到低电平，以迫使发送器进入等待状态。当接收器释放时钟SCL时继续进行数据传输。

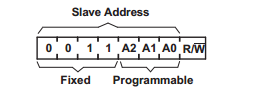
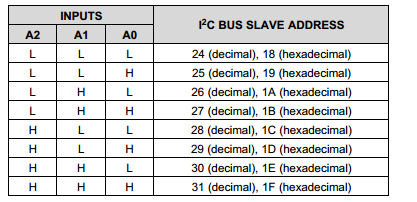
从机地址在起始条件之后发送。该地址为7位，后面跟的是第8位是数据方向位，这个数据方向位决定了下一个操作是接受（高电平）还是发送（低电平），0表示传输（发送）；1表示请求数据（接收）。数据传输始终由主机产生的停止条件来终止。然而，通过产生重复的起始条件和寻址另一个从机（而无需先产生停止条件），主机仍然可以再总线上通信。因此，在这种传输过程中可能会有接收/发送格式的不同组合。

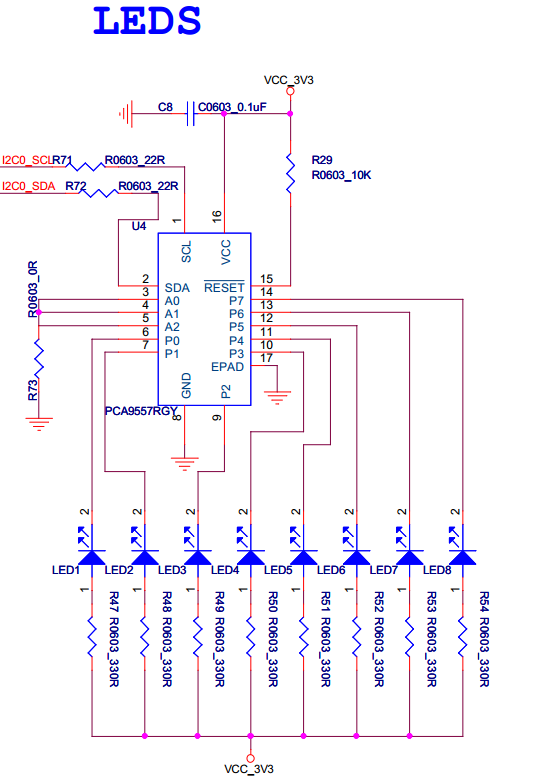
首字节的前面7位组成了从机地址。第8位决定了消息的方向。首字节的R/S位为0表示主机将向所选择的从机发送信息。该位为1表示主机将接收来自从机的信息。

带有I2C总线的器件除了有从机地址（Slave Address）外，还有数据地址（也称子地址）。从机地址是指该器件在I2C总线上被主机寻址的地址，而数据地址是指该器件内部不同部件和存储单元的编址。

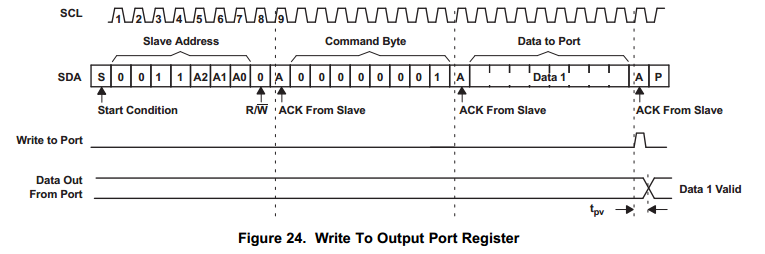
4.2.2 I2C芯片PCA9557

地址定义



从上图可以看出，蓝板使用的PCA9557芯片的地址为0x18



因为PCA9557为I2C转8位GPIO扩展芯片，在板上为LED驱动，低有效。即当管脚为低电平时，对应的LED亮。

PCA9557的使用分为两步：

1. PCA9557初始化，即将端口配置为输出。
2. 对PCA9557的输出端口赋值，例赋0x0时，则8个LED全亮，

当为0x0ff时，8个LED全灭。

固件库文件驱动方式从上图的对输出端口写数据可见，首先给I2C总线输出从设备即PCA9557地址，然后给出两字节数据，第一个字节为命令，第二个字节为数据。命令即输出端口01h，数据则为从0x00-0x0ff。

因此固件库使用如下：其中DevAddr为设备地址即0x18，RegAddr为输出端口地址即0x01，WriteData为写给输出端口的数据。

*uint8\_t I2C0\_WriteByte(uint8\_t DevAddr, uint8\_t RegAddr, uint8\_t WriteData)*

*{*

*uint8\_t rop;*

*while(I2CMasterBusy(I2C0\_BASE)){};*

*I2CMasterSlaveAddrSet(I2C0\_BASE, DevAddr, false);*

*I2CMasterDataPut(I2C0\_BASE, RegAddr);*

*I2CMasterControl(I2C0\_BASE, I2C\_MASTER\_CMD\_BURST\_SEND\_START);*

*while(I2CMasterBusy(I2C0\_BASE)){};*

*rop = (uint8\_t)I2CMasterErr(I2C0\_BASE);*

*I2CMasterDataPut(I2C0\_BASE, WriteData);*

*I2CMasterControl(I2C0\_BASE, I2C\_MASTER\_CMD\_BURST\_SEND\_FINISH);*

*while(I2CMasterBusy(I2C0\_BASE)){};*

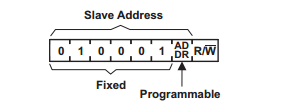
*rop = (uint8\_t)I2CMasterErr(I2C0\_BASE);*

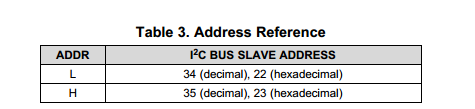
*return rop;*

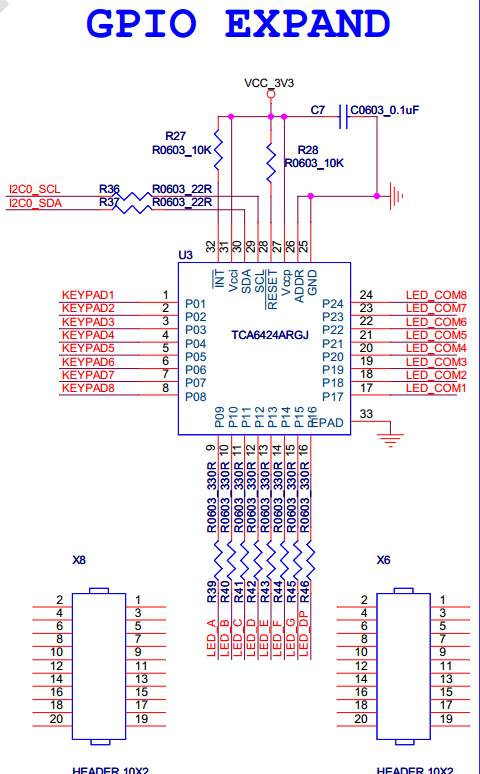
*}*

I2C芯片TCA6424

地址定义







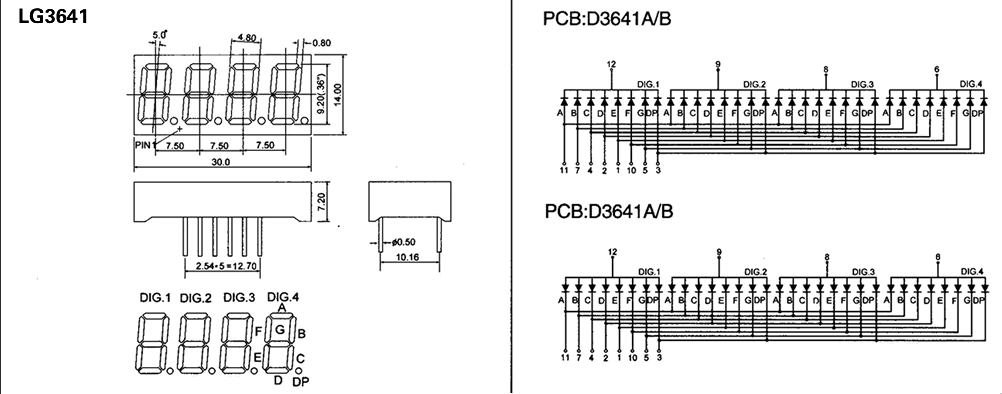
从上图可以看出，蓝板使用的TCA6424芯片的地址为0x22

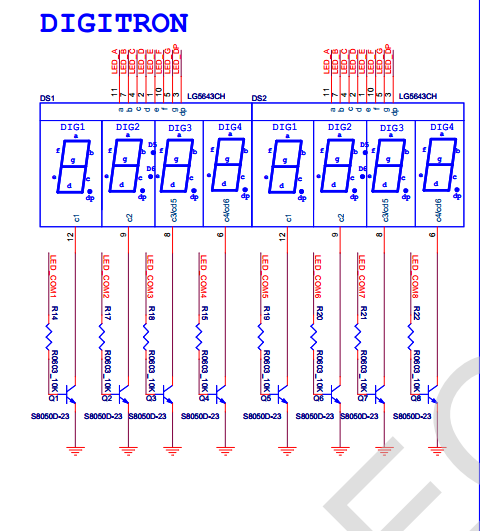
因为TCA6424为I2C转24位GPIO扩展芯片，分为3组，每组8位。在板上分别为：

P0为按键SW1-SW8；

P1为动态共阴数码管的脚位信号，高电平时点亮相应的笔划；

P2为动态共阴数码管的片选信号，当为高电平时，驱动对应的8050三极管导通，从而选通对应的位。





TCA6424的使用分为两步：

1. TCA6424初始化，即将端口P0配置为输入，P1，P2配置为输出。
2. 对PCA9557的输出端口赋值。从而点亮动态数码管。