

高档单片机应用 实验报告

于海粟 无56 2015010901 yuhs15@mails.tsinghua.edu.cn

For source code, please visit:

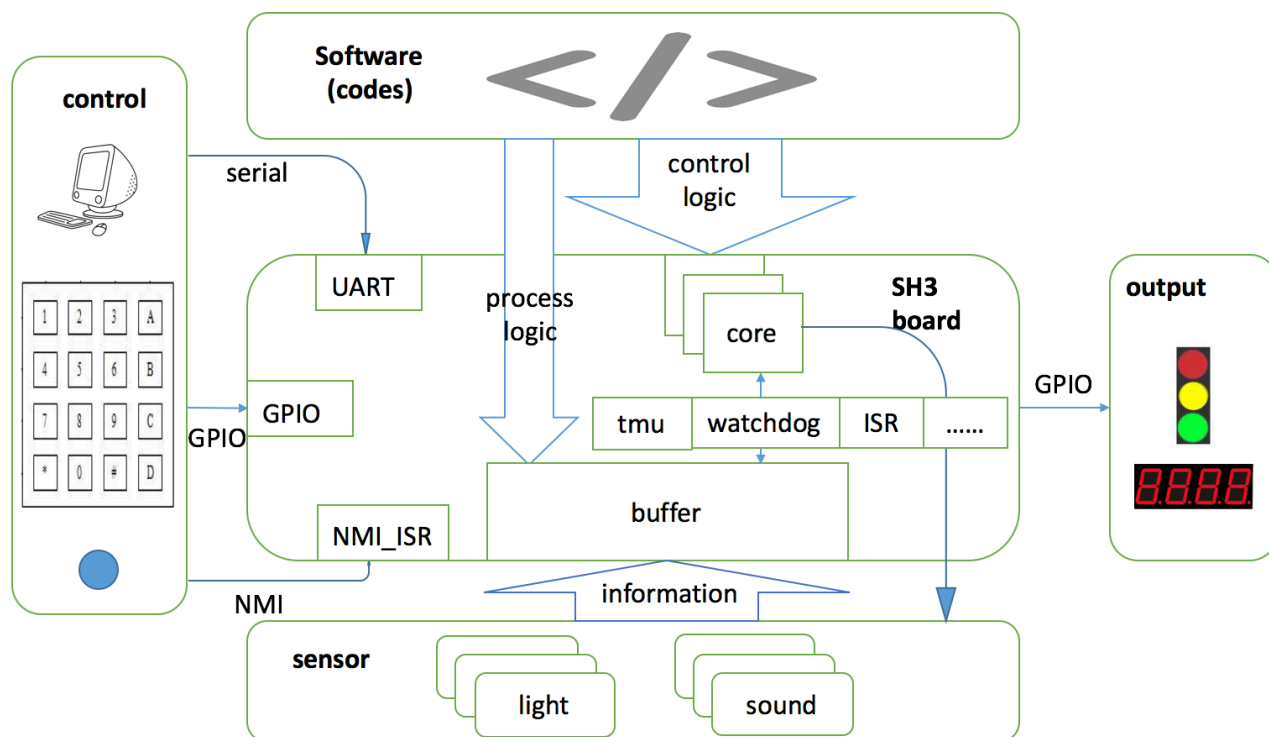
https://github.com/tsinghua-yuhs15/SH3_exp.git

1.基础实验部分

1-1 实验原理

基础实验实现了用单片机控制LED、数码管模拟智能交通系统，辅以光线传感器、声音传感器收集信息，同时采用了矩阵键盘、NMI中断键、串口、AD / DA模块辅助控制系统，完成了智能交通系统在不同模式下的工作和模式间的切换。

1-2 软硬件架构图



1-3 主要技术问题及解决方案

1-3-1 LED、数码管输出

由于LED和数码管已经提供管脚接口，只需要在相应的gpio管脚按照规则输出高低电平即可控制LED的亮暗；数码管想要显示两位数字则需要采用一个较高的扫描频率利用人眼特性交替显示数字。

1-3-2 NMI、矩阵键盘中断

中断是单片机为用户提供的方便的事件触发接口，利用特有的优化机制免去了用户自己通过while循环的判断浪费内存的困扰，采用异步方式完成事件触发并执行相关事件。这里采用矩阵键盘的A键作为ISR0以及NMI_ISR，按下按键时就会转入中断内部。为了优化程序运行，引入flag变量mode，只在中断内更改mode值，主函数中判断mode值来决定工作模式。

1-3-3 定时器

定时器是单片机提供的另一个方便的功能——本实验中使用TMU0，每1 / 1000秒处发一次中断代表1ms，抽象出函数delay_ms () 作为延时控制，可以实现特定时间的延时控制，该控制也广泛应用于本实验中。

1-3-4 传感器

本次试实验采用的传感器有光线传感器、声音传感器。光线传感器可以将光强转换为模拟电压输出，采用AD转换器并由单片机寄存器采集形成一个数值，用于之后对光强的判断；声音传感器直接判断声强是否超过阈值来输出高电平和低电平。单片机读取后通过一些处理逻辑使用这些信息。

1-3-5 串口

串口通讯广泛应用于单片机中，可以借助串口通信完成单片机与pc的信息交换，本实验中可以用串口完成程序的debug以及使用串口控制工作模式。

1-4 主要接线

本实验基本按照实验指导完成接线。

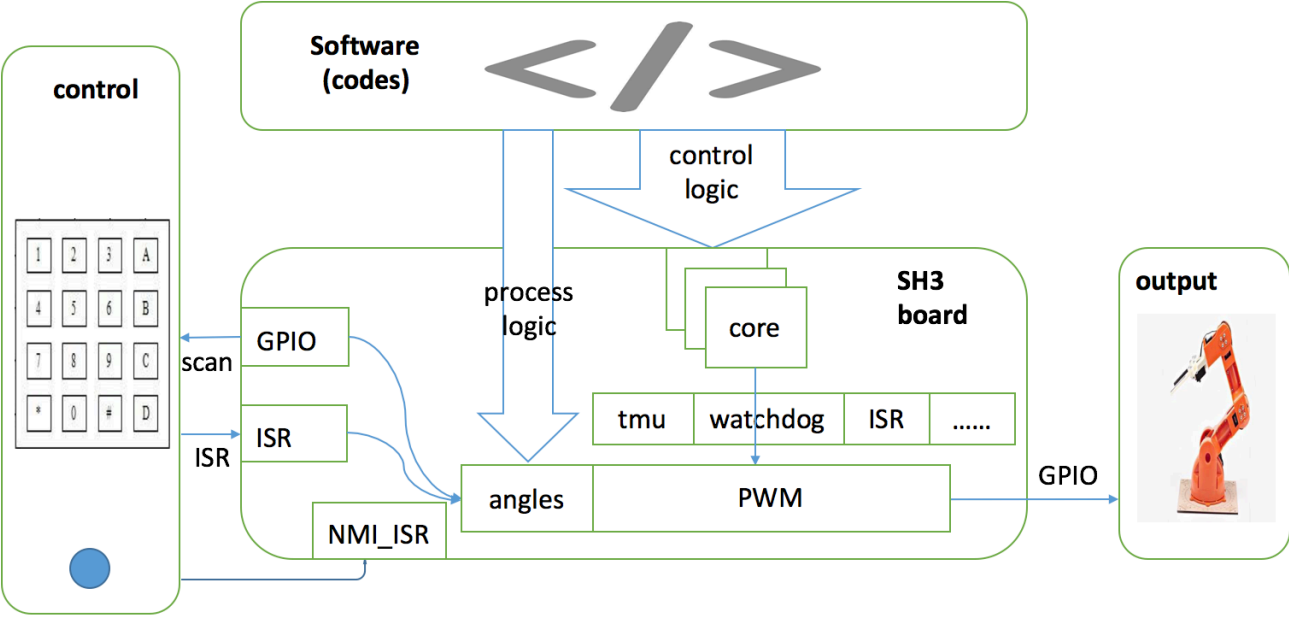
2.期末大作业部分

2-1 实验原理

本实验实验原理包括两部分：PWM控制信号的生成以及矩阵键盘全键盘控制。PWM信号即通过控制电平的高低以及不同的延时生成对应波形（即直流电机控制信号）；矩阵键盘则通过原理图和相关文档自己完成相应的控制代码，因为矩阵键盘的全键盘控制之前的实验没有涉及，所以考虑到实验尽量不要重复这里选择键盘控制。

键盘原理也比较简单，即通过扫描信号（0111，1011，1101，1110）不断输入四个横向信号输入，同时四个纵向信号连接单片机中断捕捉，不同的中断信号被捕捉到后会检查此时是哪个扫描信号以确定到底是哪个键被按下。这种扫描方式可以节省管脚数。

2-2 软硬件架构图



2-3 主要技术问题及解决方案

2-3-1 PWM信号的生成

PWM信号的原理就是通过控制一个周期性方波信号的占空比以改变输出电压的有效值从而使直流电机有不同的扭转力矩从而转动不同的角度。由于是方波输出可以直接抬高对应管脚电平一段时间后再拉低对应管脚电平来实现。采用的仍是自己完成的delay_mms函数写成的PWM逻辑。

2-3-2 矩阵键盘控制逻辑

矩阵键盘的原理刚才已经简要阐述过，本实验中采用矩阵键盘控制六个电机的旋转：

通过按“#”复位，再按“*”进入控制状态，然后按“1”~“6”选择要控制的电机，然后“A”是顺时针旋转13.5度，“B”是逆时针旋转13.5度，“C”是复位。其中每个按键的功能都已经抽象成了函数，函数内容就是修改六个PWM的占空比参数，主函数中循环调用PWM生成函数并根据修改过的占空比生成PWM信号。采用指针传递数组，基本的C语言语法。

2-4 主要接线

本实验接线：六个电动机分别接入PE0~5，矩阵键盘四个扫描gpio接PD0—~3，四个中断信号接ISR0~3.

3.课程收获

以前只用过stm32用于嵌入式开发，更接近上层高级语言的开发，有很多方便库函数和snippets，只需要根据特定的功能拼接即可；但是这门课上学到了一些关于定时器、中断、看门狗、AD / DA等单片机常用功能的概念和底层开发，掌握到了更多硬件的知识，同时配合以前软件的基础，这门课程比较顺利地完成了，最后感谢老师和助教的耐心帮助。