**《嵌入式系统及应用》实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **年级、专业、班级** | | **2019级计算机科学与技术卓越** | | | **姓名** | **李燕琴 李放** |
| **实验题目** | **使用中断产生方波发声** | | | | | |
| **实验时间** | **2022年4月17日** | | **实验地点** | **DS3 305** | | |
| **实验成绩** |  | | **实验性质** | **■验证性 □设计性 □综合性** | | |
| 教师评价：  ■算法/实验过程正确； ■源程序/实验内容提交 ■程序结构/实验步骤合理；  ■实验结果正确； ■语法、语义正确； ■报告规范；  其他：  评价教师签名： | | | | | | |
| 1. 实验目的 2. 理解和学习中断软件 3. 将接口输出到扬声器，使其能够将一个440HZ的方波音频发声出来。 4. 学习中断创造的音频的准确性。 | | | | | | |
| 1. 实验项目内容 2. 学习中断相关知识。 3. 编写代码实现中断程序。 4. 利用中断将输入电流转换成440HZ的方波输出电流，使其能够在扬声器上发出对应音频的声音。 | | | | | | |
| 1. 实验过程或算法（源程序） 2. 首先需要按照本次实验的要求编写代码，利用中断实现将输入电流转换为440HZ的方波输出电流。但是由于本次实验能够找到源码，所以我们的任务变成了需要读懂实验代码并了解掌握是如何借助这些代码利用中断进行电流的转换。 3. 按照接线图接线。在实验指导材料上有本次实验要用到的接线图，我们可以根据这个接线图完成接线工作,我们将PA3作为输入，PA2作为输出，将由PA3输入的电流转换为440HZ的方波电流由PA2输出。 4. 进行实验，验证实验结果的正确性。将PA2的输出接到扬声器上，观察扬声器的发声情况。 5. 通过改变中断处理程序的处理频率，中断处理程序的频率和音频之间的关系为：     输出不同频率的方波电流，得到不同音频，得到不同的声音，观察修改频率后的扬声器发声情况。   1. 实验电路连线图        1. 实验代码 2. #include "PLL.h" 3. #include "SysTick.h" 4. #include "UART.h" 6. #define GPIO\_PORTF\_DATA\_R       (\*((volatile unsigned long \*)0x400253FC)) 7. #define GPIO\_PORTF\_DIR\_R        (\*((volatile unsigned long \*)0x40025400)) 8. #define GPIO\_PORTF\_AFSEL\_R      (\*((volatile unsigned long \*)0x40025420)) 9. #define GPIO\_PORTF\_PUR\_R        (\*((volatile unsigned long \*)0x40025510)) 10. #define GPIO\_PORTF\_DEN\_R        (\*((volatile unsigned long \*)0x4002551C)) 11. #define GPIO\_PORTF\_LOCK\_R       (\*((volatile unsigned long \*)0x40025520)) 12. #define GPIO\_PORTF\_CR\_R         (\*((volatile unsigned long \*)0x40025524)) 13. #define GPIO\_PORTF\_AMSEL\_R      (\*((volatile unsigned long \*)0x40025528)) 14. #define GPIO\_PORTF\_PCTL\_R       (\*((volatile unsigned long \*)0x4002552C)) 15. #define SYSCTL\_RCGC2\_R          (\*((volatile unsigned long \*)0x400FE108)) 16. #define GPIO\_PORTF\_PDR\_R        (\*((volatile unsigned long \*)0x40025514)) 18. **void** PortF\_Init(**void**){ **volatile** unsigned **long** delay; 19. SYSCTL\_RCGC2\_R |= 0x20;         // 1) 激活Port F 20. delay = SYSCTL\_RCGC2\_R;         // allow time for clock to stabilize 21. GPIO\_PORTF\_LOCK\_R = 0x4C4F434B; // 2) unlock Port F lock 22. GPIO\_PORTF\_CR\_R = 0x1F;         //   允许PF4-0改变配置 23. GPIO\_PORTF\_AMSEL\_R = 0x00;      // 3) 禁用PF4-0的模拟信号量 24. GPIO\_PORTF\_PCTL\_R = 0x00000000; // 4) 配置 PF4-0 为 GPIO并行输入输出端口 25. GPIO\_PORTF\_DIR\_R = 0x0E;        // 5) PF4 and PF0 为输入, PF3-1 为输出 01110 26. GPIO\_PORTF\_AFSEL\_R = 0x00;      // 6) disable alt funct on PF4-0 27. GPIO\_PORTF\_DEN\_R = 0x1F;        // 7) 启用PF4-0的I/O信号量 28. // GPIO\_PORTF\_PUR\_R = 0x11;        //    enable pull-up on PF4 and PF0 29. GPIO\_PORTF\_PUR\_R = 0x00;        //    disable pull-up on PF4 and PF0 30. GPIO\_PORTF\_PDR\_R = 0x11;        //    enable pull-down on PF4 and PF0 31. } 33. //  red, yellow, green, light blue, blue, purple,  white,  dark 34. **const** **long** ColorWheel[8] = {0x02,0x0A,0x08,0x0C,0x04,0x06,0x0E,0x00}; 35. **int** main(**void**){ 36. unsigned **long** SW1,SW2; 37. **long** prevSW1 = 0;        // SW1的先值 38. **long** prevSW2 = 0;        // SW2的先值 39. unsigned **char** inColor;   // 其他实验板传入的颜色的值 40. unsigned **char** color = 0; // 当前LED灯的颜色 41. PLL\_Init();              // 设置系统时钟为80 MHz 42. SysTick\_Init();          // initialize SysTick 43. UART\_Init();             // initialize UART 44. PortF\_Init();            // initialize buttons and LEDs on Port F 45. **while**(1){ 46. SW1 = GPIO\_PORTF\_DATA\_R&0x10; // Read SW1 47. **if**((SW1 == 0) && prevSW1){    // falling of SW1? 48. color = (color+1)&0x07;     // 在按下开关SW1时进入ColorWheel的下一种颜色 49. } 50. prevSW1 = SW1; // 当前按下的是SW1 51. SW2 = GPIO\_PORTF\_DATA\_R&0x01; // Read SW2 52. **if**((SW2 == 0) && prevSW2){    // falling of SW2? 53. UART\_OutChar(color+0x30);   // 在按下开关SW2时发送颜色值的ASCALL码，as '0' - '7' 54. } 55. prevSW2 = SW2; // 当前按下的是SW2 56. inColor = UART\_InCharNonBlocking(); 57. **if**(inColor){ // FIFO是否有新到达的数据 58. color = inColor&0x07;     // 更新当前实验板的颜色 59. } 60. GPIO\_PORTF\_DATA\_R = ColorWheel[color];  // 更新LED灯 61. SysTick\_Wait10ms(2);        // 防止开关反跳，即在20ms之内按下两次 62. } 63. } | | | | | | |
| 1. 实验结果及分析和（或）源程序调试过程 2. 针对自己的LaunchPad，下图展示为接线方式。      1. 按下开关，耳机能清晰地听到声音。 | | | | | | |
| 1. 实验心得   本实验学习了使用中断产生方波发声。其中我们学会了编写代码实现中断程序并利用中断将输入电流转换成440HZ的方波输出电流，使其能够在扬声器上发出对应音频的声音。也学会了五脚耳机座的使用。 | | | | | | |