

华中科技大学

2017

系统能力培养综合实验 实验报告

题 目：智能小车系统

专 业：计算机科学与技术

班 级：CS1402

学 号：U201414590

姓 名：吴其其

电 话：15927588862

邮 件：abc_577@qq.com

完成日期：2017-9-19



计算机科学与技术学院

目 录

1	系统能力培养综合实践概述	2
1.1	课程目的	2
1.2	设计任务	2
1.3	课程要求	2
2	HOS 操作系统实践	3
2.1	Hos 操作系统搭建与运行	3
2.2	在 Hos 操作系统中添加应用	3
2.3	实验结果	4
3	蓝牙小车应用设计	5
3.1	小车驱动板接口设计	5
3.2	结果仿真	7
3.3	性能分析	8
3.4	主要故障与调试.....	8
3.5	实验进度	8
4	实验总结与心得	9
4.1	实验总结	9
4.2	实验心得	9

1 系统能力培养综合实践概述

1.1 课程目的

《计算机系统能力培养综合实践》课程是为了加强计算机专业学生的系统能力，顺应计算机教育改革的方向和近年的发展趋势，而开设的一门综合性的实践课程。课程的设置目标是，增强学生的动手能力、与其他组员配合与协调能力的同时，也会深入的理解学习计算机专业基础知识的目的与用途，增强其综合能力，同时也可增强对本专业的兴趣。

1.2 设计任务

课程分为三个部分：在第一部分，课程将指导学生在 Nexys4 DDR FPGA 开发板上搭建一个基于 MIPS 的处理器（软核），并在此基础上设计处理器的外围设备。具体的外围设备，包括 UART、COM 接口、蓝牙设备等。在第二部分，课程将指导学生在已开发出来的 MIPS 处理器上，搭建一个小型的操作系统。该操作系统基于清华的 ucore 操作系统改造而来，同时融合了一些新的特性，使得该操作系统相比于 ucore 而言更加小巧、实用。在第三部分，课程将指导学生将所开发出的系统投入到具体的应用中。课程计划引导学生设计一个，采用之前两个阶段所开发出来的计算机系统，加上步进马达系统、机械传动系统以及便携供电系统所共同构成的，由蓝牙设备遥控的小车。

1.3 课程要求

- (1) 完成 MIPSfpga 处理器硬件平台的搭建；
- (2) 完成 Hos 操作系统的实践；
- (3) 基于 MIPSfpga 硬件平台，在 Hos 操作系统中实现一个利用手机无线蓝牙控制的蓝牙小车应用，该应用能实现小车前进、倒车、转向等基本控制功能和更加复杂的小车控制如加减速等；
- (4) 蓝牙小车扩展功能；
- (5) 阶段验收检查；
- (6) 综合实践报告和总结。

2 Hos 操作系统实践

2.1 Hos 操作系统搭建与运行

2.1.1 交叉编译环境搭建

Hos 系统使用 C 语言编写，运行目标平台为 MIPS，所以在 Windows 下编译时就需要配置交叉编译环境。在实验中，使用 cygwin 来代替 Linux 进行实验。所以首先按照实验指导书安装配置 cygwin 软件并且安装所需软件。

2.1.2 Makefile 修改

因为在第一部分实验中，安装的 MIPS 交叉编译器是 mips-mti 编译器，而非原 Makefile 中的 mips-sde 编译器，所以将操作系统源码根目录下的 Makefile.config 文件中的 UCONFIG_CROSS_COMPILE="mips-sde-elf-"中的"sde"修改为"mti"。在这个文件中还可以看到 USER_APPLIST 项，该项是编译用户命令列表，可以看做是用户程序，在之后的 Hos 修改中，可以在这里直接新加入目标程序名以完成增加的程序的编译。

2.1.3 Hos 编译与运行

前面环境配置没有出错的情况下，只要在 cygwin 中，进入源码主目录下执行 make，就可以完成系统编译，之后烧好开发板，修改相应的端口名，再执行 run.bat，就可以完成系统的载入，完成之后在 putty 中就可以看到 Hos 的控制台，此时可以执行前面说的用户程序来验证正确性。

2.2 在 Hos 操作系统中添加应用

2.2.1 用户态打印“hello world”

这一步比较简单，在源码/user/user-ucore 目录下增加一个源程序，仿照该目录下的其他源程序编写一个打印 hello world 的程序，然后在前面提到的 USER_APPLIST 中增加该程序的程序名即可。然后重新编译生成操作系统，验证正确性。

2.2.2 内核态打印“hello world”

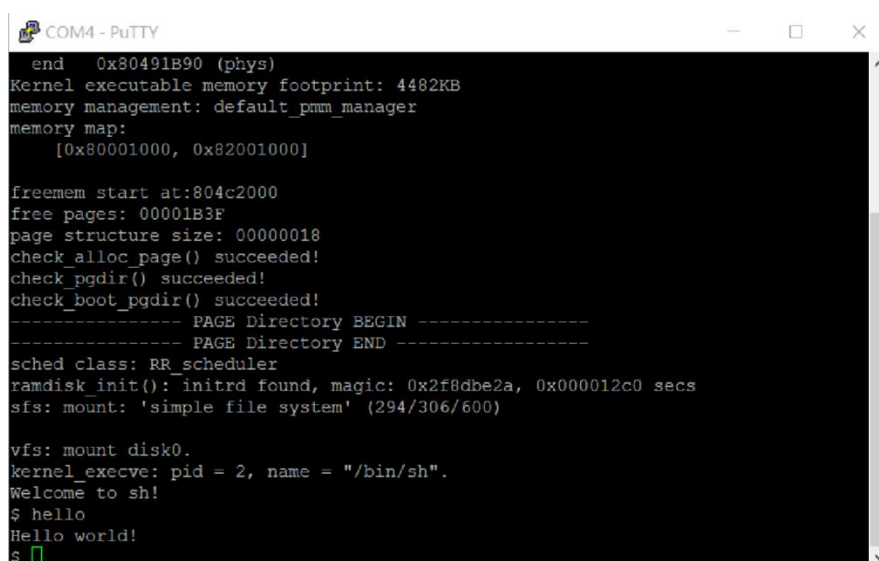
在内核态完成打印功能，需要以添加系统功能调用的形式来完成，修改/kern-ucore/syscall.c 在其中增加新的系统功能调用函数。再在/kern-ucore/include/lib/unistd.h 中为新增加的系统功能调用分配系统功能调用号。这样一个新的系统功能调用就完成了，不过在 Hos 中，用户态调用系统功能调用，还在/user 下多了一个 syscall 的调用，也就是实际上，用户态会先调用/user 下的 syscall，然后/user 下的程序才去调用真正的系统功能调用，为了统一，我们新增的系统功能调用也用这样的形式，所以再修改/user 下的对应文件即可。完成后，在用户源程序中添加对应的系统功能调用，重新编译，验证正确性。

2.2.3 打印 Hos 操作系统的空闲页面数量

这一部分内容需要我们去了解 Hos 对物理内存管理的机制，而在实现上则很简单，在/kern-ucore/pmm.c 中，nr_free_pages 函数的返回值即空闲页面数量，所以我们可以新增系统功能调用，然后逐层将这个返回值返回到用户态程序中，在用户态中打印出来数目即可。

2.3 实验结果

图 2.1 为在用户态打印“hello world”结果，说明操作系统的运行符合要求。



```
COM4 - PuTTY
end 0x80491B90 (phys)
Kernel executable memory footprint: 4482KB
memory management: default_pmm_manager
memory map:
[0x80001000, 0x82001000]

freemem start at:804c2000
free pages: 00001B3F
page structure size: 00000018
check_alloc_page() succeeded!
check_pgdir() succeeded!
check_boot_pgdir() succeeded!
----- PAGE Directory BEGIN -----
----- PAGE Directory END -----
sched class: RR_scheduler
ramdisk_init(): initrd found, magic: 0x2f8dbe2a, 0x000012c0 secs
sfs: mount: 'simple file system' (294/306/600)

vfs: mount disk0.
kernel_execve: pid = 2, name = "/bin/sh".
Welcome to sh!
$ hello
Hello world!
$
```

图 2.1 打印“hello world”结果

3 蓝牙小车应用设计

3.1 小车驱动板接口设计

在蓝牙智能小车实验中，我的组内分工是硬件部分的搭建，也就是开发板和小车驱动板的接口部分。另外蓝牙接口部分也是我负责，但是实验所用蓝牙芯片实质上就是一个串口设备，在 vivado 中有现成的串口 IP 核可以使用，就不作介绍了。

3.1.1 小车驱动板原理

本次实验所用的小车为四驱动小车，使用 L293D 驱动板驱动。该驱动板有十多个输入信号接口，看起来比较复杂，但是将板上芯片单独来看就比较清晰。首先，板子上有两块 L293D 芯片，这种芯片常被用来控制电机的转向和转速，每块电机最多可同时控制两个电机，所以两块芯片正好对应着小车的四个驱动电机。L293D 芯片主要的输入有控制信号和 PWM 调速信号。其中每个电机的控制信号有两根，这两根信号的高低电平组合构成小车的正转、反转和停止。

PWM 调速的原理为：输入占空比不同的数字信号，在输入信号的每个周期，信号为高电平时，电信号均可以对外做功。所以，相同周期的信号，占空比越大，所做的功就越多，其实可以看做数字信号波形中高电平对时间的积分。因为做功不同，电机转速就不同，从而达到通过占空比调速的目的。

L293D 所需的 PWM 信号，在硬件平台搭建时就实现了一个接口来输出这种信号，所以在这里就可以复用前面封装的 IP 核。四个电机分别控制，可以使用四个独立的 IP 核，也可以将其整合在一起，甚至根据实际应用情况，将其减少为 2 个也是可行的。而在本次实验中，我采用了单独 4 个 PWM 信号的方法，优点在于简单快捷。

而实际上，L293D 的转向控制信号并不需要我们直接从开发板上接过去。在驱动板上，这些输入已经连接好了，连接在 74HCT595N 芯片的输出上。74HCT595N 芯片，如图 3.1 所示，内部由一个移位寄存器、一个锁存器和一个三态输出构成。而我们队小车方向控制的 8 位信号，正是存放在这里。

74HCT595N 芯片工作时序逻辑为：在串行输入时钟的控制下，每个周期，移位寄存器右移一位，将输入的信号放在最高位。在锁存器时钟上升沿，移位寄存器中的

华中科技大学课程实验报告

值锁存进锁存器中。三态输出使能端为低电平时，将锁存器值输出。所以我们要将控制信号写入驱动板，需要：先使用同步串行通信的方法将值写入移位寄存器，写入完成后给锁存信号，然后使能三态门。

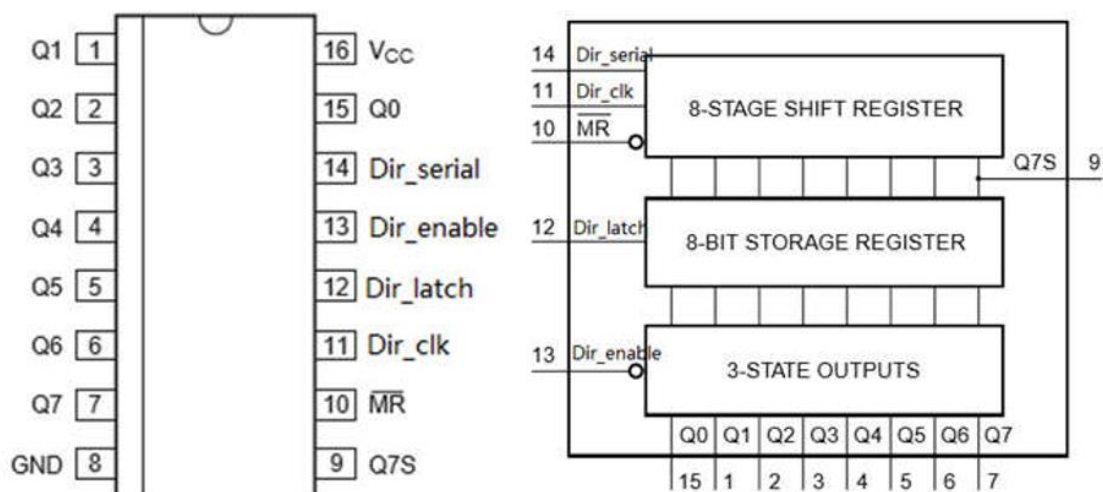


图 3.1 74HCT595N 芯片引脚图

3.1.2 接口模块设计

根据上述分析，驱动板接口有：四个独立的 PWM，一个 74HCT595N 芯片接口。PWMIP 核已经实现，所以直接在处理器系统总线上增加四个 PWM 模块即可。而 74HCT595N 芯片接口则需要自行设计。

74HCT595N 芯片接口有四根信号线：时钟(dir_clk)、串行信号线(dir_serial)、锁存器时钟(dir_latch)、三态门使能(dir_enable)。这些信号的时序关系在上节中已经说明，所以接下来设计这个接口。

根据应用环境，我们对小车的控制是一种无反馈的控制，所以该接口仅需要可写，也就是说，我们控制小车的前进后退转向等，实际上是控制每个车轮的正反转，所以需要 8 位数据。这 8 位数据，先写到 AXI4 总线寄存器中，然后再由一定的逻辑模块传输到驱动板上。在查阅资料的过程中我发现，74HCT595N 芯片是标准的串行外设接口(SPI)，所以我一开始考虑用 vivado 中的 SPI 的 IP 核，但是最终未能成功实现传输，所以我还是按照同步串行通信原理重新实现了一个逻辑模块。

因为前面说的时序使用有限状态机能比较直接的描述，所以我设计了一个有限状态机来实现数据发送，状态机的状态变迁如图 3.2 所示。

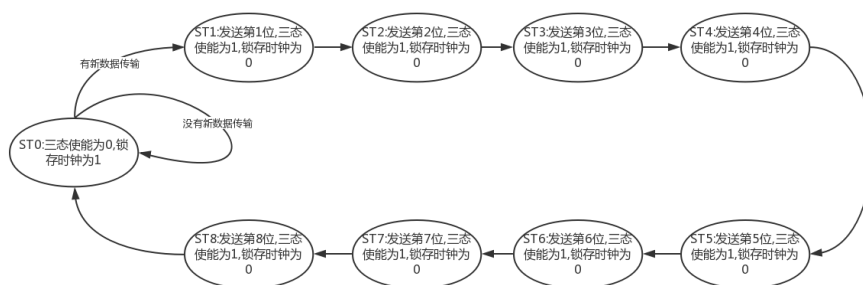


图 3.2 状态机状态变迁图

上述状态机在每次 CPU 向端口写入数据之后开始传输数据，在传输完成后给出相应的锁存信号和使能信号即可。

而这个设计的难点在于确定何时 CPU 向端口里写了数据，一种方案是使用 AXI4 总线的控制信号来控制。在 AXI4 总线的传输握手完成时 `axi_wready`、`S_AXI_WVALID`、`axi_awready` 和 `S_AXI_AWVALID` 信号均为有效时，数据在下一个时钟上升沿被写入总线寄存器。所以我们可以利用这些信号的有效与否来控制传输是否开始。另一种方法是在状态机中加入一个寄存器，保存上次传输的数据，然后在每次时钟上升沿时和总线寄存器中的数据进行比较，如果数据不一致，则发送总线寄存器中的数据同时将该数据存储用作下次比较。而在本次实验中我们选取了后一种方案，因为前一种方案在本应用中存在一些问题：因为和驱动板的接口类似于一个同步串行传输，CPU 时钟频率也就是总线频率是 50MHz，在高频下，传输更容易因干扰出错，而且 L293D 驱动板的建议频率为 1000Hz，所以在传输时我们采用了低频。这样就带来一个问题，总线的高频决定了上述握手信号的存在时间远远小于低频的一个周期，所以在低频模块里在每个时钟沿对上述信号进行判断是不可行的。或者将上述信号作为低频模块的敏感事件列表之一，但是这种想法很难在硬件上实现。

3.2 结果仿真

上述接口整体仿真比较麻烦，所以在实现后我针对最主要的数据传输模块，也就是上述状态机模块进行了仿真。

图 3.3 为数据传输接口模块的功能仿真。从图中可以看到，当输入变为 1 之后的下一个时钟周期下降沿，开始传输最低位的 1，`latch` 变为低电平，`enable` 变为高电平。在完成最后一位传输后，`latch` 变为高电平从而产生上升沿来使之前传输的值锁存进锁存器，`enable` 变为低电平，使能小车驱动板控制信号。直到下次需要传输的值改变，重复

华中科技大学课程实验报告

上述过程，进行新一轮的数据传输。

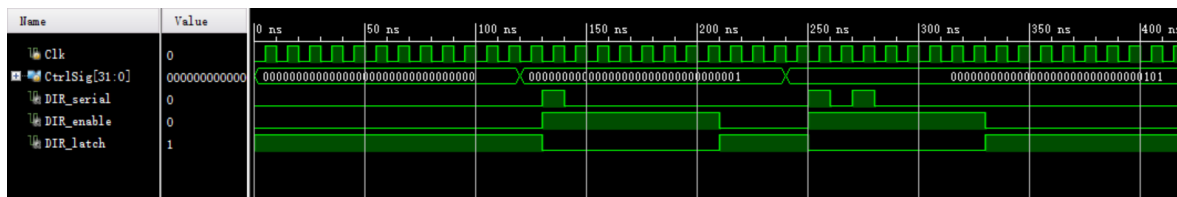


图 3.3 数据传输模块功能仿真结果

3.3 性能分析

在设计中采用的传输时钟频率为 1000Hz，但是单独对上述模块进行实现后仿真时，采用 100MHz 的时钟频率模块功能依然正常，所以如果在使用 74HCT595N 芯片时有更高速的传输需要上述模块也能胜任。

将上述模块与分频模块一起组成一个基于 AXI4 总线的接口，挂载在前面完成的处理器系统上，给接口分配一个地址，我们就可以通过向接口中写入数据来直接控制小车的运动。

3.4 主要故障与调试

故障现象：在实现基于 AXI4 总线握手信号开始传输的接口后，直接进行测试发现无法驱动小车。

原因分析：经过逐层查找，最终确定是因为接口内时钟频率不一致导致，上述握手信号使用高频时钟，存在时间短，而发送模块使用低频时钟，需要控制信号存在较长时间，所以上述设计不满足时序逻辑。

解决方案：修改硬件模块，使用数值比较判断进行传输开始的控制。

3.5 实验进度

表 3.1 实验进度表

时间	进度
9月4日-9月8日	硬件平台搭建，完成全部实验内容
9月8日-9月12日	操作系统部分实验
9月13日-9月15日	小车驱动板接口的设计与实现
9月16日-9月18日	调试蓝牙部分，协助队友完成蓝牙智能小车整体实现

4 实验总结与心得

4.1 实验总结

在本次实验中，我和同组同学各自发挥所长，通力合作，最终实现了一个较为成熟的蓝牙智能小车。我们实现的智能小车实现了以下几点功能：

1. 在手机或者电脑的控制端软件，通过蓝牙连接小车，实现遥控；
2. 使用 PWM 调速原理，实现了小车的可变速行进和差速转向功能；
3. 记录控制信号序列实现轨迹记录和轨迹重放功能；
4. 使用自定义接口实现了音乐播放功能。

4.2 实验心得

本次实验是目前为止综合程度最高的一个是实验，其中用到了组成原理、计算机接口技术、操作系统、软件工程等方面的知识。这次实验，既是对以前知识的回顾，也是一次很好的提升，通过这次实验，我们将以前所学知识融合起来，形成了一个完整的计算机系统的概念。

通过自己的动手实验，我主要有以下收获：

1. 掌握计算机处理器系统的基本原理及开发过程；
2. 掌握了基于 AXI4 总线标准的接口设计与实现；
3. 进一步理解计算机中断的软硬件实现原理；
4. 掌握了计算机操作系统最基本结构及其功能；
5. 学会了蓝牙通信编程。

在实验中，我也面对过很多障碍。比如在开发小车驱动板接口的时候，很多自己觉得很合理的逻辑无法综合，这启示我，硬件开发和软件开发的思路有着本质上的区别，要做好硬件模块，就要尝试着从硬件的角度去思考问题，从基本的逻辑开始分析。另外，实验所用驱动板的芯片的各种资料很多，但是这块驱动板的说明就很少，这就要我们学会自己去结合以前所学知识，分析其结构和原理，从而为其设计接口。

对于这样较为综合的实验，同组同学配合必不可少。组内同学在不同的方面都有自己的特长。我对硬件部分较为熟悉，在前期硬件平台搭建中就多去帮助队友，保持

华中科技大学课程实验报告

进度，而在操作系统部分，对 Hos 有着深入掌握的同学则给队友指出关键点。在后期的小车总体设计中，我们各自分工明确，各个模块定义和接口定义合理，全组成员都在计划时间内完成自己的分工，使得我们的小车基础部分提前一个星期完工。在这种合作中，我体会到了团队合作的效率和重要性。

虽然这次实验我们组完成了要求的内容，但是因为时间关系和课程安排在了同学们升学、找工作的时段，所以我们没能投入全部的精力去实现一个更为完善智能小车，我们组实现了音乐播放，但是未能将其集成在小车上；开发板上带有 SD 卡槽和 USB 接口，将这些利用起来，我们也可以进一步扩展存储系统等。希望以后实验可以改善这一状况。这次实验的选做部分有两个，我建议在今后的实验中，将不同的扩展功能分配给不同的组，更大化的利用开发板的功能，这样就能在实验中积累更多的成果。

最后在这里也感谢三位老师对于我在本次实验中问题的耐心解答，也感谢本组所有成员在实验中对于我的帮助和建议。感谢 Imagination Technologies 公司提供的 MIPS 处理器软核，这为我们实验提供了很大便利，不过我也希望，今后我们的组成原理课程设计会越来越好，能让同学们用上自己实现的 CPU 来完成这个系统。我相信这次综合能力培养实验中我的收获将会对我今后的学习有很大帮助。

• 指导教师评定意见 •

一、原创性声明

本人郑重声明本报告内容，是由作者本人独立完成的。有关观点、方法、数据和文献等的引用已在文中指出。除文中已注明引用的内容外，本报告不包含任何其他个人或集体已经公开发表的作品成果，不存在剽窃、抄袭行为。

特此声明！

作者签字：

吴其其

二、对课程实验的学术评语（教师填写）

三、对实验的评分（教师填写）

评分项目 (分值)	报告撰写 (30 分)	课设过程 (70 分)	最终评定 (100 分)
得分			

指导教师签字：_____ 年 月 日