老师和各位同学，大家好，我们小组今天来进行计算机系统结构课程系统移植和应用开发的实验答辩。【翻页】

今日答辩内容将分成下面四部分来依次进行。【翻页】

本次实验中选用可以嵌入到 FPGA 中的 RISC 处理软核 MicroBlaze 嵌入式软核来进行操作系统的移植工作。移植的操作系统则选用了𝜇C/OS -III操作系统，这一操作系统架构简单，且可以支持MicroBlaze 软核。【翻页】

实验环境是基于N4开发板，在Windows10操作系统上，利用Vivado2019.1完成整个移植和开发过程。【翻页】

本次综合实验操作系统移植采用Vivado + Xilinx SDK + Micrium Xilinx BSP的方式进行移植。因此首先从官网下载开发所用的板级开发包，更好连接上层驱动程序和硬件设备寄存器。【翻页】

打开Block Design，先添加CPU部件 Micro Blaze，不使用缓存设备，利用片内存储32KB。其次添加提供串口模块字符串输出的硬件AXI Uartlite IP核。为了支持时钟中断，添加定时器AXI Timer，设置时钟频率为N4板上单端频率100Mhz。最后添加中断控制器AXI Interrupt Controller以支持时钟中断。【翻页】

添加好上述部件，接下来进行Auto Connection，手动连接CPU的中断接口和中断控制器；创建一个集线连接部件，连接所有Interrupt接口后送回中断控制器。最后验证布局成功。【翻页】

将布局设计输出为实际的ip核文件，并生成顶层模块，得到design\_1\_wrapper.v文件。进行接口约束， E3为N4板上时钟信号源，J15控制复位信号。最后，点击Generate Bitstream生成bit流。【翻页】

下面是移植操作系统的部分，将刚生成的CPU导出为硬件，注意要同时导出bit流；打开SDK设计界面，加入BSP目录地址，以便使用μcos操作系统封装模块。打开板级配置菜单，指定标准输入输出的串口，指定中断时钟信号，重新生成BSP。至此，移植操作系统部分实验完毕。【翻页】

新建application project，如果BSP目录配置正确，可以看到OS Platform中出现的选择项。如果硬件设计配置正确，可以看到Hardware platform以及对应的micro blaze处理器。选择 - III，至此新建工程完毕，应用开发环境配置成功。【翻页】

我们实现的是猜拳游戏，即用户可以和运行在Micro Blaze CPU、操作系统上的程序**进行交互**，模拟**人和计算机进行猜拳**的过程。【翻页】

具体的成果演示请看视频【翻页】