



华中农业大学
HUAZHONG AGRICULTURAL UNIVERSITY

实 验 报 告

EXPERIMENT REPORT

姓名 邢广杰

学号 2017308210404

专业 计算机科学与技术

教师 高俊祥

科目 接口技术 A

信 息 学 院
COLLEGE OF INFORMATICS

8086CPU 与多种接口芯片协同工作的计时抢答系统的设计与实现

一. 实验目的

本实验报告所包含的实验为《接口技术 A》课程的课程设计实验，为创新型、综合型的大型实验项目。实验中需要用到在《接口技术 A》课程上学习到的有关 8086CPU 芯片及常见的各种接口芯片的原理和功能。同时，实验使用到了 EDA 仿真工具软件 proteus。本实验项目的综合型较强，旨在达成以下目的：

1. 通过完成复杂的电路设计，培养数字逻辑电路设计的能力。
2. 通过对 CPU 芯片和接口芯片的功能设计与协同工作设计，加强对《接口技术 A》课程上学习到的微机原理与接口技术的理解。
3. 通过编写汇编代码完成对设计功能的实现，加深对各种接口芯片的工作方式和控制字写入方式的认知。
4. 在 Proteus 仿真软件中完成对实验设计的仿真，培养学生对现代 EDA 软件的熟练使用。
5. 通过对各种功能的设计与实现，加深对 CPU 芯片中中断等机制实现方式与工作流程的理解。
6. 通过协同各种接口芯片与 CPU 芯片的接口通信，了解 CPU 访问接口芯片的方式，理解 CPU 芯片工作流中的时序概念，理解 CPU 芯片在硬件层面的工作原理。

二. 实验原理

8086CPU

Intel 8086 拥有四个 16 位的通用寄存器，也能够当作八个 8 位寄存器来存取，以及四个 16 位索引寄存器（包含了堆栈指标）。资料寄存器通常由指令隐含地使用，针对暂存值需要复杂的寄存器配置。它提供 64K 8 位元的输出输入（或 32K 16 位

元），以及固定的向量中断。大部分的指令只能够存取一个内存位址，所以其中一个操作数必须是一个寄存器。运算结果会储存在操作数中的一个寄存器。

Intel 8086 有四个 内存区段 (segment) 寄存器，可以从索引寄存器来设定。区段寄存器可以让 CPU 利用特殊的方式存取 1 MB 内存。8086 把段地址左移 4 位然后把它加上偏移地址。大部分的人都认为这是一个很不好的设计，因为这样的结果是会让各分段有重叠。尽管这样对组合语言而言大部分被接受（也甚至有用），可以完全地控制分段，使在编程中使用指针（如 C 编程语言）变得困难。它导致指针的高效率表示变得困难，且有可能产生两个指向同一个地方的指针拥有不同的地址。更坏的是，这种方式产生要让内存扩充到大于 1 MB 的困难。而 8086 的寻址方式改变让内存扩充较有效率。

8086 处理器的时钟频率介于 4.77MHz（在原先的 IBM PC 频率）和 10 MHz 之间。8086 没有包含浮点指令部分（FPU），但是可以通过外接数学辅助处理器来增强浮点计算能力。Intel 8087 是标准版本。

8255 芯片

8255 芯片是 Intel 公司生产的可编程并行 I/O 接口芯片，有 3 个 8 位并行 I/O 口。具有 3 个通道 3 种工作方式的可编程并行接口芯片（40 引脚）。其各口功能可由软件选择，使用灵活，通用性强。8255 可作为单片机与多种外设连接时的中间接口电路。

8255 作为主机与外设的连接芯片，必须提供与主机相连的 3 个总线接口，即数据线、地址线、控制线接口。同时必须具有与外设连接的接口 A、B、C 口。由于 8255 可编程，所以必须具有逻辑控制部分，因而 8255 内部结构分为 3 个部分：与 CPU 连接部分、与外设连接部分、控制部分。

8253 芯片

8253 芯片是可编程计数器/定时器。这种芯片外形引脚都是兼容性的。8253 内部有三个计数器，分别称为计数器 0、计数器 1 和计数器 2，他们的结构完全相同。

每个计数器的输入和输出都决定于设置在控制寄存器中的控制字，互相之间工作完全独立。每个计数器通过三个引脚和外部联系，一个为时钟输入端 CLK，一个为门控信号输入端 GATE，另一个为输出端 OUT。每个计数器内部有一个 8 位的控制寄存器，还有一个 16 位的计数初值寄存器 CR、一个计数执行部件 CE 和一个输出锁存器 OL。

8259A 芯片

8259A 是专门为了对 8085A 和 8086/8088 进行中断控制而设计的芯片，它是可以用程序控制的中断控制器。单个的 8259A 能管理 8 级向量优先级中断。在不增加其他电路的情况下，最多可以级联成 64 级的向量优先级中断系统。8259A 有多种工作方式，能用于各种系统。各种工作方式的设定是在初始化时通过软件进行的。在总线控制器的控制下，8259A 芯片可以处于编程状态和操作状态，编程状态是 CPU 使用 IN 或 OUT 指令对 8259A 芯片进行初始化编程的状态。

三. 主要功能

本实验产出的电路系统为基于微机原理和接口技术的围绕 8086CPU 的多接口芯片协同工作的电子计时抢答系统。

该系统主要由以下部分组成：

1. 8086CPU 及锁存译码系统

该部分的主要功能为完成对 8086CPU 的地址数据总线的地址锁存，再将 8086CPU 访问的 16 位地址译码成 3 个接口芯片的片选信号，以完成 CPU 对接口芯片端口的访问。

2. 基于 8255A 并行传输芯片的显示系统

该部分主要为整个系统的输出显示部分，利用到了一块 8255A 接口芯片控制两个七段数码管和若干 LED 灯，完成对倒计时的显示和对抢答结果的显示。

3. 基于 8253 的计时/脉冲系统

此部分的功能主要为通过 8253 芯片的计时功能完成抢答的倒计时和产生一定频率的中断脉冲，以控制七段数码管显示的刷新。

4. 基于 8259A 的中断控制系统

基于 8259A 中断控制芯片主要为整个系统提供中断功能，整个系统的所有功能实现也是基于中断完成的，中断控制芯片用于控制中断的产生与执行，通过和 8086CPU 通信完成中断，以实现抢答、计时显示、抢答显示、重置等功能。

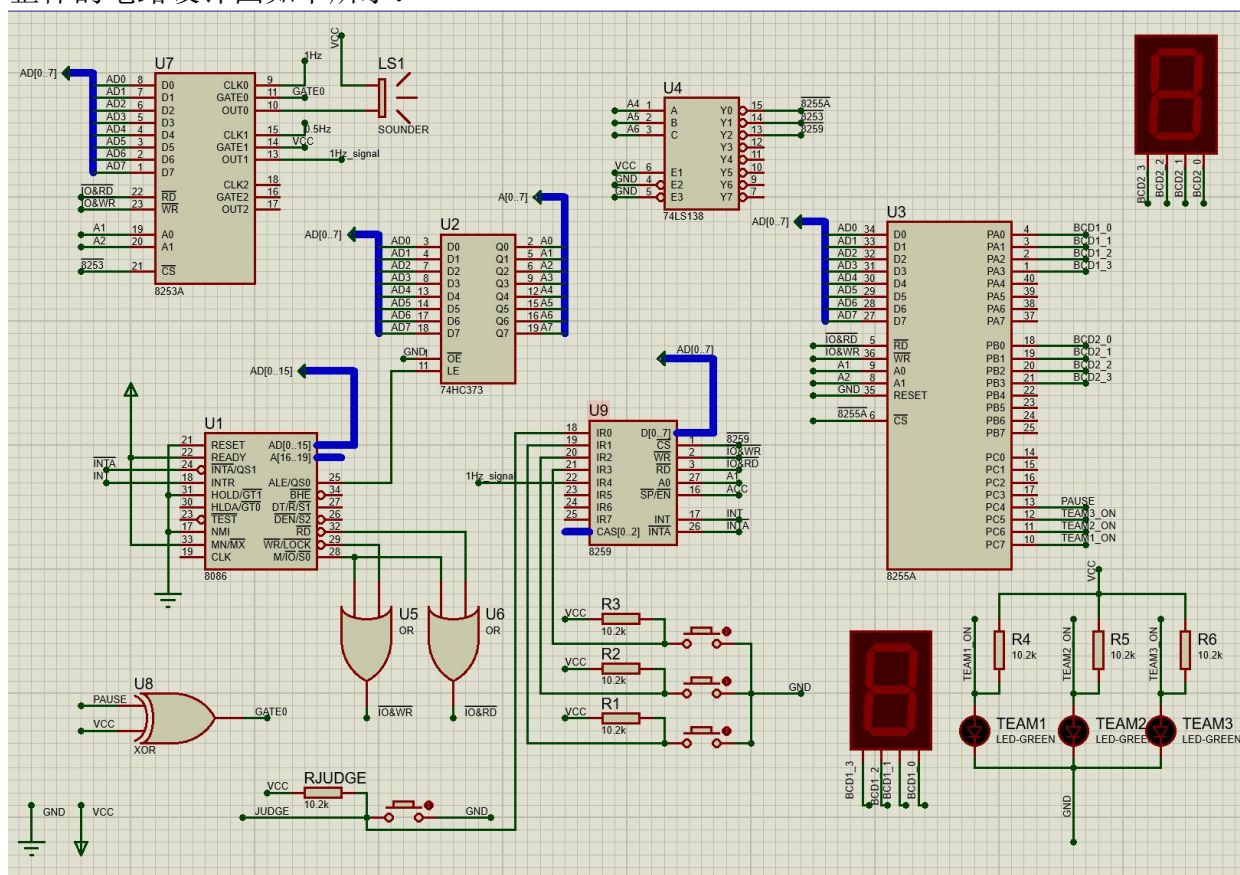
综上所述，该基于 8086CPU 和多种接口芯片的计时抢答系统实现了以下功能：

1. 裁判通过按 JUDGE 按钮可以将计时重置，并开放所有队伍的抢答按钮生效
2. 计时板会显示目前所剩下的时间，在此时间内所有队伍的抢答按钮均有效

3. 在抢答按钮有效时，任何队伍都可以通过按下抢答按钮完成抢答，完成抢答后，显示板上会显示抢答成功的队伍编号，队伍对应的 LED 灯会亮，计时板会停止计时，未抢答成功的队伍抢答按钮均失效
4. 裁判可以通过按 JUDGE 按钮重新开始计时，此时计时板和显示板及 LED 灯会重置，所有队伍的抢答按钮恢复有效
5. 若计时结束仍没有队伍完成抢答，蜂鸣器会响

四. 电路设计

整体的电路设计图如下所示：



下面分系统的几个模块描述电路设计细节：

1. 8086CPU 及锁存译码系统

此部分主要为 8086CPU 的 AD 线直接与 74HC373 的 D 排线相连，74HC373 的输出 Q 排线的若干端口再与 74LS138 的输入 A、B、C 相连，74LS138 的三个输出口再分别连上其他接口芯片的片选信号，以完成对 8086CPU 产生的地址译码，选中对应的接口芯片完成数据传输。

2. 基于 8255A 并行传输芯片的显示系统

8255A 的 D0~7 口与 8086CPU 的数据地址总线相连，以完成对控制寄存器和三个并行传输寄存器的读写，同时，A 口与一个七段数码管相连，完成抢答成功队伍的显示，B 口与另一个七段数码管相连，完成剩余时间的显示。PC4~7 分别与控制计时暂停的位于 8253 的 GATE0 以及三个 LED 灯相连，完成抢答成功时的状态输出。

3. 基于 8253 的计时/脉冲系统

8253A 也是将 D 排线连上了数据地址总线，用于接收 8086CPU 传输的数据，同时，计时器 0 为倒计时计时器，其控制了一个蜂鸣器，计时器 1 为固定频率脉冲发生器，其输出信号接到 8259A 中断控制芯片上。

4. 基于 8259A 的中断控制系统

8259A 作为整个系统的核心芯片，几乎与其他部分都有相连，首先是其 D 口排线同样连上了地址数据总线，用以完成对 8086 产生读写数据的接收和发送，同时，其一些控制端口利于 INT、INTA 与 8086 相应的端口相连，完成中断信号的发送和中断控制字的发送，以协助 8086 完成中断控制。其 IR0 口接了一个按钮，用于接受裁判按钮的中断信号，IR1~IR3 分别接了三个队伍按钮，而 IR4 接了 8253 产生的定时脉冲的输出信号。

五. 实现方法

本系统主要是用到了 8086CPU 及其他若干接口芯片实现了所有的功能，下面对于每个芯片单独阐述其实现功能的方式：

● 8086CPU

8086CPU 作为整个系统的核心，主要完成对整个系统功能的产生、控制以及调度，通过在 8086CPU 上执行汇编代码，完成对系统其他芯片的初始化、数据传送以及中断控制和执行。

此项目中 8086CPU 工作在最小模式下，数据和地址共用 16 位总线，对接口芯片的片选通过译码系统完成，其 16 位的第 1、2 位接到所有 CPU 的控制端口上，同时译码系统根据其 4、5、6 位完成译码产生对接口芯片的片选信号。

● 8255A 并行数据传输芯片

8086CPU 开始执行程序时完成对 8255A 的初始化，对其写入的控制字为 10000001B，即工作在方式 0，A 口用于输出，C 口的 4~7 引脚用于输出，B 口用于输出，C 口 0~3 引脚用于输入

方式 0 是最基本的输入输出方式，本项目用其进行无条件传送

- 8253 计时控制芯片

8086CPU 开始执行时也完成对 8253 的初始化，对计时器 0 的控制字写入为 00010001B，使其工作在方式 0，低字节有效，以 BCD 方式计数，对计时器 1 写入控制字为 01010111B，以方式 3 工作，低字节有效，同样以 BCD 计数，计时器 0 用于倒计时，计时器 1 用于方波发生，即产生一个固定频率的脉冲。

- 8259A 中断控制芯片

8086CPU 在完成对上述芯片的初始化后，开始初始化 8259A 中断控制芯片，并安装中断控制程序。其中，对 8259A 的初始化要分 4 次，因为有四个中断程序，每一次初始化又要分三步执行，首先写入 ICW1 寄存器，控制字为 00010011B，表明要写 ICW4，再对 ICW2 寄存器写入中断控制字的信息，最后对 ICW4 吸入中断信号产生和传送的方式以及中断结束的方式。接着，对操作命令字 OCW2 写入控制字，关闭除 IRO 外的所有中断，等待裁判启动系统。安装中断控制程序需要将 4 个不同的中断控制程序的入口地址分别写入段地址为 0，偏移地址为 4*中断控制字的内存单元。此时，整个系统的初始化就完成了，8086CPU 进入死循环，等待系统继续运行。

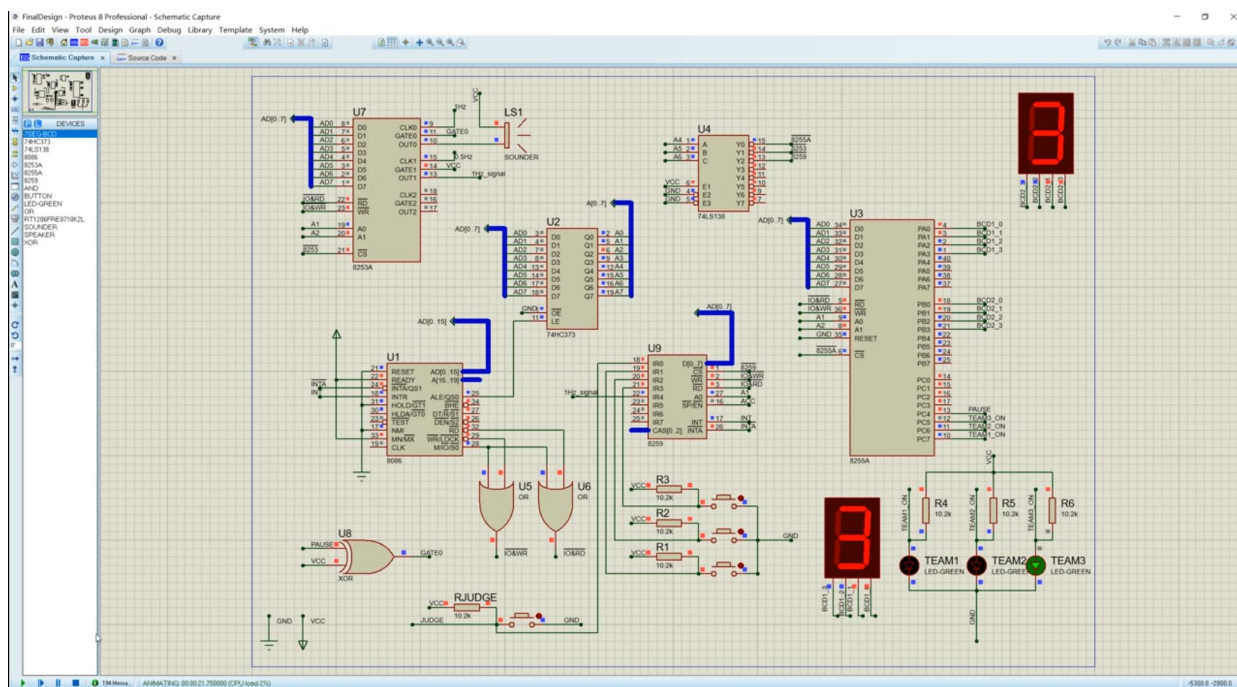
在系统运行期间，若裁判按钮按下，则会产生一次中断，此中断将计时器 0 置初始值 5，工作在方式 0 的计时器收到初始值时会马上重新开始计数。接着，中断控制程序会对 8259A 的操作命令字 OCW2 写入控制字，用于打开其他的中断。

在进入倒计时之后，由于各队伍按钮触发的终端被置为有效，当各队伍按下对应的按钮时，会进入相应的中断处理程序执行。用于处理队伍抢答的中断处理程序首先会向 8259A 的 OCW 寄存器写入控制字，屏蔽其他队伍按钮抢答的终端，并关闭由计时脉冲产生的用于刷新计时显示的中断，让显示板停留在当前的计时时间。同时，会向 8255 的 C 口写入数据，用于控制对应的 LED 灯亮，并暂停计时。

8253 的计时器 0 用于完成抢答计时的倒计时，当寄存器中的值变为 0 时，OUT0 引脚的电平会发生变化，将除 JUDGE 按钮外的所有中断屏蔽，不再接受选手的抢答。

8253 的计时器 1 工作在方式 3，可以看作是一个方波发生器，CLK1 引脚连接一个 2Hz 的脉冲信号，则从 OUT1 可以得到一个稳定的 1Hz 电平，用作 8259A 的一个时钟中断，用于提供给 8255 刷新显示板。

六. 结果分析



系统运行结果如上图所示，整个系统能够正常运行，所有设计的功能均能够正确实现，符合预期结果，课程设计获得成功。

七. 总结

作为《接口技术与微机原理》的课程设计项目，本系统从设计出发，尽量多运用了理论课程学习的各种接口芯片，使其协同工作，完成各种各样的功能，以此加深对理论课程学习到的相关知识的理解。

本次设计的系统整体来说还是非常庞大的，课程设计的时间比较紧张，但为了使学习过的接口芯片尽量都能在系统中出现，还是费了一些心思的，虽然系统功能不多，但

为了将设计好的功能实现，首先学习了软件 Proteus 的使用，该软件是一个专门用于 EDA 电路设计的软件，通过该软件可以将电路作图在网格上，轻松将相应的引脚连线，然后对 8086CPU 编写汇编代码，实现对其他接口芯片的控制，从而完成相应的功能。

在编写代码的过程中，发现对汇编语言有些遗忘，写起来比较困难，也存在代码组织不够清晰的问题，后来也发生了出现 BUG 但无从下手调试的情况，这说明之后对于任何一门课程的学习，一定要做到掌握牢固，计算机的学科需要融会贯通，将所有的技术综合应用，才能实现一个表现优秀的系统，

《接口技术与微机原理》课程作为一门专业选修课，上完之后感觉还是非常有必要学的，它其实是在告诉我们各种硬件是怎样在 CPU 的控制下协同工作完成各种功能的，通过学习这门课程，可以让我们更好的理解硬件是如何进行通信的以及如何完成 CPU 交给的各种任务的。接口芯片有很多，但是通过学习典型的几个，可以让我们了解其原理，方便后续对任何硬件的学习。