

实验十二 自选大实验

2020 年秋季学期

“Do you mean to tell me that you’re thinking seriously of building that way, when and if you are an architect?”

“Yes.”

“My dear fellow, who will let you?”

“That’s not the point. The point is, who will stop me?”

– “The Fountainhead”, Ayn Rand

本实验为同学自选实验，可以自行选择题目，也可以在下列课题中选择一个。大实验为 2 人一组，如果做计算机系统，可以 3 人一组。学期最后一周之前验收并提交实验报告。

12.1 音乐播放器

难度系数 ★
工作量 ★★
趣味性 ★★

利用 DE10-Standard 实现一个简单的音乐播放器，可以播放 3 首以上乐曲。乐曲可以用乐谱输入调整输出频率的方式来实现。

基本功能：

- 3 首以上的乐曲播放
- 循环播放、随机选曲播放
- 暂停与继续功能

可选扩展功能：

- 利用波形存储的方法实现钢琴、长笛的乐器的音色（参考 MIDI sound-font），用不同乐器演奏同一乐谱。此项功能难度系数为两星至三星，视演奏效果而定。
- 能够记录键盘电子琴演奏的音乐，并重放该演奏片段。

12.2 模拟时钟

难度系数 ★★

工作量 ★★

趣味性 ★

在显示器上绘制一个时钟表盘，包含时分秒信息。同时要求在 FPGA 开发板上的七段数码管中显示时间。

基本功能：

- 通过键盘设置时间
- 闹钟功能
- 整点报时，音频实现

可选其他功能

- 显示器上的时钟表盘以模拟时钟形式呈现，自行绘制时分秒指针及钟面。
建议自行查找计算机图形学相关教材（如电子工业出版社，计算机图形学，Hearn & Baker，提供电子书），了解坐标计算、画线算法的基本原理（相关内容较多，只需了解必要关键算法即可）。
- 万年历、星期的设置。

12.3 复刻打字小游戏

难度系数 ★★

工作量 ★★

趣味性 ★★

请用 FPGA 实现计算机系统基础课 PA4 中的打字小游戏，即屏幕上随机产生一些字母，并以不同随机速度掉落。玩家需要按对应的键在字符掉落到屏幕底部之前消除字符，如图 12-1 所示。

基本功能：

- 只需实现界面中的字符部分。VGA 采用 60Hz 刷新率，分辨率和字符大小可与 PA4 不同
- 实现随机字符产生，字符掉落，按键消除字符功能和游戏逻辑

- 游戏信息提示，如“FPS”等字符，色彩可以自选。

👉 思考：

- 如何实现字符的“平滑”下落？
- 用 FPGA 实现与用 NEMU 实现有什么相同的地方，有什么地方不同？

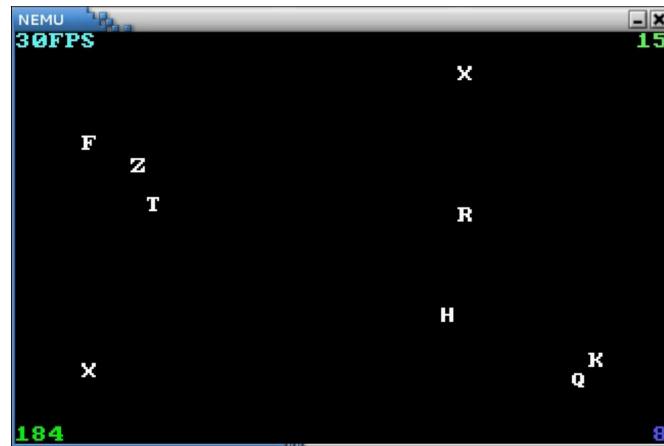


图 12-1: 打字小游戏界面

12.4 MD5 摘要

难度系数 ★★
工作量 ★★
趣味性 ★

MD5 是常用的信息摘要算法，给定一段信息，通过 MD5 算法可以产生 128 位的散列值。这个散列值可以用来验证数据是否被篡改过，修改信息的任何一个 bit 都会造成散列值的巨大变化。例如：“The quick brown fox jumps over the lazy dog” 的散列值是：

0x9e107d9d372bb6826bd81d3542a419d6

而增加一个句号，“The quick brown fox jumps over the lazy dog.” 的散列值就变成了：

0xe4d909c290d0fb1ca068ffaddf22cbd0

MD5 常用于验证网上下载的文件完整性。常用的操作系统和编程语言均提供 MD5 计算的接口。

在本实验中，你将根据 MD5 算法的描述 (如: <https://en.wikipedia.org/wiki/MD5>) 自己通过 FPGA 实现 MD5 算法。事实上，MD5 算法设计上是非常易于用硬件实现的。

基本功能:

利用实验十一的简单文本交互系统，通过键盘输入一串 ASCII 字符。在输入完毕键入回车之后，进行该段字符 (不含回车) 的 MD5 摘要计算。并将计算结果以文本格式显示在屏幕上。需要与操作系统提供的 MD5 算法比对，保证结果的正确性。

12.5 简单计算机系统

难度系数 ★★★★★

工作量 ★★★★★

趣味性 ★★

还记得本学期的 PA 实验么？在 FPGA 上也可以实现一个简单的计算机系统，完成起来并不像你想像的那么困难。本实验的目标是实现拥有基本指令和输入输出功能的计算机，基本实现思路如下：

- **处理器部分：**使用 FPGA 实现取指令，指令分析，指令执行以及数据写回功能。可以自行设计指令格式，建议指令长度统一为 32 位，数据也为 32 位，实现 4-8 个寄存器。实现基本的 RAM 读取/存储、立即数赋值、ALU（加减法，移位等等）、比较、跳转指令等。可以参考计算机组成原理中的单周期 CPU 实现，并进行适当简化。
- **RAM 部分：**利用 FPGA 实现简单的数据和指令 RAM，容量在 1KByte 以上。
- **输入输出部分：**支持键盘输入。输出可以通过 VGA 显示实现，也可以通过七段显示输出结果（处理器直接写某个寄存器输出到七段显示器）。代码可以通过自己编写的简单汇编代码转换为机器码写入 RAM。
- **基本功能：**能接收键盘输入命令并执行，能执行简单的算术、循环、分支判断代码。例如，计算斐波那契数列的第 N 位或计算 N 的阶乘。感兴趣的同学也可以利用汇编实现数据结构中提到的中缀表达式的计算，让用户可以输入表达式并计算结果输出。

- **扩展功能：**支持更多指令，支持 bootloader 功能，支持单步调试等等。

注：请勿直接使用网上现成的 CPU 开源工程，请自行设计 CPU 结构和系统结构。

✎ 具体实现思路参阅附加文档。

12.6 自定选题

允许自己提出新的大作业的课题，但必须提前与任课教师说明课题内容，确定后方可自选。