

# “一生一芯” 太理工作室二组寒假研学最终次学习路线

学习目标：结束本次寒假研学学习内容，至此大家就完成了数字设计，数字电路，复习C语言，数据结构与算法的学习，额外的我们还学习了makefile的使用，debug工具的使用等等计算机基础知识。在之后我们会彻底完成Learn C The Hard Way的学习，在补全所有知识后，大家就要去挑战南大数电实验和PA了！完成这部分学习后，就可以报名预学习答辩了！大家加油！



预防针：本次作业有一定难度：

数字设计迎来很难的一章——有限状态机，思路理不清楚很有可能坐一天也想不出来，这时候就试着画画图吧！理清楚思路再写这部分代码效果会好很多哦！并且这一部分是有固定套路的，大家在做了很多道相关题之后，做笔记归纳总结，就可以悟出状态机的构建流程了！

## 学习作业

1. 随时记录自己的学习记录，一定要如实记录，我们会不定期查看各位的学习情况。
2. Learn C The Hard Way中8-18，20-22节完成，最后将章节练习任务放在命名为“C”的文件夹中。

！这部分已经完成的同学可以将Learn C The Hard Way继续向后做，其中的26、37-41、43、45-47不需要学习，性价比比较低，不推荐学习，其他内容我们都很推荐学习

3. Verilog实践：Verilog在线学习网站——veilog学习实践，全英文（在这里推荐一个浏览器插件：沉浸式翻译，如果看不懂的话，就用这个插件配合学习吧！），完成如下的Shift Registers与Finite State Machines的内容，中间的可跳过，最后完成部分上传截图即可

！有限状态机部分题目有很多重复，大家对这里有自己的方法和理解即可，可以跳过重复性较高的题！！

至此，我们数字设计部分学习结束。若已完成这部分，可选择拔高中的南大数电实践。

## Shift Registers

- 4-bit shift register ○
- Left/right rotator ○
- Left/right arithmetic shift by 1 or 8 ○
- 5-bit LFSR ○
- 3-bit LFSR ○
- 32-bit LFSR ○
- Shift register ○
- Shift register ○
- 3-input LUT ○

## Finite State Machines

- Simple FSM 1 (asynchronous reset) ○
  - Simple FSM 1 (synchronous reset) ○
  - Simple FSM 2 (asynchronous reset) ○
  - Simple FSM 2 (synchronous reset) ○
  - Simple state transitions 3 ○
  - Simple one-hot state transitions 3 ○
  - Simple FSM 3 (asynchronous reset) ○
  - Simple FSM 3 (synchronous reset) ○
  - Design a Moore FSM ○
  - Lemmings 1 ○
  - Lemmings 2 ○
  - Lemmings 3 ○
  - Lemmings 4 ○
  - One-hot FSM ○
  - PS/2 packet parser ○
  - PS/2 packet parser and datapath ○
  - Serial receiver ○
  - Serial receiver and datapath ○
  - Serial receiver with parity checking ○
  - Sequence recognition ○
  - Q8: Design a Mealy FSM ○
  - Q5a: Serial two's complementer (Moore FSM) ○
  - Q5b: Serial two's complementer (Mealy FSM) ○
  - Q3a: FSM ○
  - Q3b: FSM ○
  - Q3c: FSM logic ○
  - Q6b: FSM next-state logic ○
  - Q6c: FSM one-hot next-state logic ○
  - Q6: FSM ○
  - Q2a: FSM ○
  - Q2b: One-hot FSM equations ○
  - Q2a: FSM ○
  - Q2b: Another FSM ○
-

最后将以上内容放入一个文件夹里，文件夹命名为 **你的姓名-专业班级**，然后压缩为一个压缩文件即可。

请**严格按照上述要求**发送到邮箱：[YunDing\\_YSYX@163.com](mailto:YunDing_YSYX@163.com)

## 拔高

### 1. 计算机组成原理

#### 地位

考研408数据结构 操作系统 **计算机组成原理** 计算机网络，计算机知识硬件部分精华科目，理工科学生必修课之一，无论考研还是工作都占据极其重要的地位，同时更是我们后续实践的基础科目。

#### ? 为什么有这么重要的地位呢？

那就要知道机组具体学的内容了，以下是机组完整学习的思维导图，大家自取了解：

链接：[https://pan.baidu.com/s/1YDMg\\_OsAa2Kn5wpj7k99JA?pwd=9999](https://pan.baidu.com/s/1YDMg_OsAa2Kn5wpj7k99JA?pwd=9999) 提取码：9999

看完之后，你应该就明白为何如此重要了——这不就是我们设计芯片的基础蓝图吗？

第六七章的IO和总线部分难度大，内容多，大家在后续才会接触，因此我们寒假只需要学习完第五章处理器即可。

#### 内容

以下视频的**CH1-CH5五章内容学习**，记得随手typora记录笔记和继续自己的学习记录哦！

[https://www.bilibili.com/video/BV1Ba4y1V7GD/?vd\\_source=4ec31615294fd2510d5fd40f0183648f](https://www.bilibili.com/video/BV1Ba4y1V7GD/?vd_source=4ec31615294fd2510d5fd40f0183648f)

机组部分学习会大量涉及你学过的**数电**知识，如果当时没有好好学，那就要比较吃力了。

**!** 注意！该课程是使用教材为**MIPS指令集版本**，我们之后设计芯片，使用的是**RISCV指令集版本**，但是考虑到该课程体系完善，效果好，指令集思想是相通的，因此我们仍然选择了该视频课程，配套书籍为黑皮书：计算机组成与设计：硬件软件接口，图示如下（**群内有电子版资料**），想搭配书籍的同学可以参考——不推荐纯看书，黑皮书阅读难度大，知识点深而且广，很难理清知识点。

因此，**也可以选择你觉得好的计组课程或者硬啃黑皮书**，学到知识就好！



## 计算机组成与设计：硬件软件接口（原书第5版）

戴维 A. 帕特森 (David A. Patterson), 约翰 L. 亨尼斯 (John L. Hennessy)

☆ 5.0 / 5.0    9 comments    心    书    目

《计算机组成与设计：硬件/软件接口》是计算机组成与设计的经典畅销教材，第5版经过全面更新，关注后PC时代发生在计算机体系结构领域的革命性变革——从单核处理器到多核微处理器，从串行到并行。本书特别关注移动计算和云计算，通过平板电脑、云体系结构以及ARM（移动计算设备）和x86（云计算）体系结构来探索和揭示这场技术变革。

与前几版一样，本书采用MIPS处理器讲解计算机硬件技术、汇编语言、计算机算术、流水线、存储器层次结构以及I/O等基本功能。

种类: Computers <  
年: 2015  
出版社: 机械工业出版社  
页: 557  
ISBN 13: 9787111504825  
文件: PDF, 153.37 MB

Content Type: 书籍  
出版: 5th  
语言: Chinese  
ISBN 10: 7111504828  
系列: 计算机科学丛书  
IPFS: CID, CID Blake2b

▶ 线上阅读

↓ 下载 (PDF, 153.37 MB)



发送到 邮件 PB a

平装

书籍有问题吗?



### 学完机组我能干什么？

1. 为之后的数字设计，硬件设计，芯片设计等等相关方面打基础，缺少机组知识你将寸步难行
2. 对考研有很大的帮助
3. 硬件相关学校课程你的修读难度会很低

## 任务

交回相关markdown笔记（我们**推荐**大家进行**总结归纳和心得记录**，这真的很有帮助），以此说明你完成了此部分的学习。

## 2. 南大数字电路实验

**!** 注意！PA0中绝大多数操作我们已经学习并实践过了，因此请认真阅读按照如下这一部分讲义进行，请勿盲目按照PA0中讲义操作！

### A. 获取工具及框架源码

首先，我们需要安装PA中需要的工具：

```
1 apt-get install build-essential # build-essential packages, include binary utilities, gcc, make, and so on
```

```
2 apt-get install man # on-line reference manual
3 apt-get install gcc-doc # on-line reference manual for gcc
4 apt-get install gdb # GNU debugger
5 apt-get install git # revision control system
6 apt-get install libreadline-dev # a library used later
7 apt-get install libsdl2-dev # a library used later
```

随后阅读如下讲义以获取PA代码：

PA0获取包部分讲义：<https://ysyx.oscc.cc/docs/ics-pa/0.6.html#getting-source-code>



当你发现如下提醒时，阅读该讲义：<https://ysyx.oscc.cc/docs/2306/preliminary/0.2.html>  
获取属于YSYX的代码框架

**！参加"一生一芯"的同学, 请参考"一生一芯"讲义获取代码链接**

如果你参加"一生一芯", 请勿使用下面的代码链接. 此外, PA讲义中关于作业提交的要求, "一生一芯"的同学可以全部忽略, 但需要关注"一生一芯"讲义中的提交要求.

## B. Verilator仿真环境搭建

<https://ysyx.oscc.cc/docs/2306/preliminary/0.4.html>

从这里开始，我们就要在自己的Linux上进行数字设计的编程学习了，逐步搭建电路器件并进行仿真，直到最后设计出一款属于自己的CPU！

## C. 南大数字电路实验

实现YSYX预学习讲义中如下内容。

## 借助NVBoard完成数字电路实验

我们首先推荐南京大学的[数字电路与计算机组成实验](#)。

南京大学开展教学改革, 将"数字电路"与"计算机组成原理"两门课程进行融合, 其实验内容贯穿从数字电路基础到简单的处理器设计, 最近尝试加入程序运行时环境的相关内容, 与"一生一芯"的主线内容非常契合。

以下部分是**必做题**:

- 实验一 选择器
- 实验二 译码器和编码器
- 实验三 加法器与ALU
- 实验六 移位寄存器及桶形移位器
- 实验七 状态机及键盘输入

其他内容作为了解可以选做, 在预学习部分不作规定。有了NVBoard之后, 你就可以把它当作FPGA来使用, 用它来实现需要FPGA支持的实验内容。

## 3. 一生一芯课程PA

PA是我们后续学习中非常重要的一部分内容, 目前我们已经把PA0相关的基础知识进行了补全, 大家可以去尝试PA0的相关内容学习, 不过想开PA1还需要一些时间, PA1的内容需要数据结构, Lchtw都学到不错的地步并具备一定的编程思想, 然后就需要你的时间和精力花费了。

本作品《"太理工一生一芯工作室前置讲义寒假研学篇"》由 许鹏远 创作, 并采用 CC BY-SA 4.0 协议进行授权。

遵循CC BY-SA 4.0开源协议: <https://creativecommons.org/licenses/by-sa/4.0/deed.en>

转载或使用请标注所有者: 许鹏远, 太理 “一生一芯” 工作室