

# 冯·诺依曼计算机指令执行虚拟仿真实验

实验平台链接 <http://101.76.229.34:8001/>

## 一、 实验目的

冯·诺依曼体系结构是当前计算机设计的重要基础，其包含三个基本原则：二进制逻辑、程序存储执行以及计算机由五个部分组成（即运算器、控制器、存储器、输入设备、输出设备）。了解冯·诺依曼体系结构的基本设计原则对理解计算机的结构组织、运行方式具有非常重要的意义。现代人工智能技术不断发展，深度学习应用到如人脸识别、推荐系统、增强现实等诸多领域，计算机为深度学习算法的部署与执行提供了支撑，为了更好地理解这一过程，我们选取了卷积计算这一最常见的深度学习运算模块，结合基于冯·诺依曼体系结构的模型机，在虚拟仿真环境下为学生介绍计算机基本组成和指令执行流程，并引导学生完成卷积计算完整流程，激发学生的实验兴趣，提升学生对计算机系统原理的理解及解决实际问题的能力。

## 二、 实验要求

实验要求学生具备冯·诺依曼体系结构核心原理、卷积运算基本流程等知识储备，初步认识计算机系统基本结构及指令，并掌握灵活组合指令完成特定运算任务的方法，并应具有良好的查阅文献与计算机操作能力。

## 三、 实验方法

采用虚拟仿真的实验方法建立基于冯·诺依曼体系结构模型机并设计专用的指令集，实验者可通过设置系统运行关键设置与编写指令解决实际问题等互动手段深入了解计算机系统基本组成原理及指令系统和卷积计算流程等重要概念，提升计算机系统逻辑思维和解决实际问题的能力。

## 四、 实验注意事项

1. 注册账号时，用户名使用学号，姓名使用“中文姓名+班级”（例如：张三 4，表示张三是 4 班的学生），并选择正确的学校名称，方便教师后台检查完成情况。

用户名: 4-16位数字、字母组合

密码: 2-10位中文或4-20位字母数字

输入邮箱, 找回密码唯一凭证

输入密码

再次输入密码

选择省份

选择学校

已有账号, 请登录

注册

2. 进行实验时需进入全屏模式并认真阅读实验引导页，掌握虚拟仿真环境操作规范。



## 五、实验内容

整个实验主要分为三个部分，学生首先通过“指令周期演示”的动画场景完成指令执行周期的学习，通过“指令计算原理”体会冯·诺依曼计算机的基本构成、指令的种类及执行流程，最后在“卷积计算原理”模块中完成卷积计算任务所需的程序各区段设计及指令设计工作，最终完成卷积计算任务。

### 1、指令周期演示

学习模块，用以了解一条指令的取指、译码、执行的过程及相关知识了解。

### 2、指令计算原理



(1) 进入指令计算原理实验，查看实验说明：在仿真计算机结构中，通过学习和操作模拟计算机  $2 \times 3 + 1$  的运算。

(2) 查看实验提供的模拟冯·诺依曼结构计算机，包括存储器、CPU 内的控制器和运算器。存储器开辟了 0x00-0x8F 的地址空间。控制器包括取指部件、PC 寄存器和译码部件，并通过地址总线访问存储器，通过数据总线读写数据。运算器包括乘法器和加法器。还额外提供了 3 个寄存器。

(3) 开始实验后，根据底部提示信息，点击右下角初始化按钮，清空 PC、存储器及寄存器。初始化后提示清理成功。

(4) 根据提示设置存储器空间，点击选择存储器内的单元，分别设置代码段、数据段的区间范围。

(5) 分析运算过程，从候选指令区将正确的指令拖拽至对应的存储器代码段，如果未能正确选择指令，根据提示重新选择。

(6) 指令填写正确后，点击 PC 寄存器，设置指令运行的开始地址，确认并开始执行指令。

(7) 首先点击 PC 寄存器，取指部件根据 PC 提供的初始地址，通过地址总线访问 0x00，并通过数据总线将指令读回取指部件，同时 PC 寄存器进行加 1 操作。然后点击译码部件，对取指部件取回的指令进行解析。最后，执行 Load 0x60,R0 指令，将 0x60 地址上的数据读取到 R0 寄存器。

(8) 根据 PC 寄存器当前地址，读取 Load 0x61,R1 指令，并进行译码，执行指令，将 0x61 地址上的数据读取到 R1 寄存器，同时 PC 寄存器进行加 1 操作。

(9) 根据 PC 寄存器当前地址，读取 Load 0x62,R2 指令，并进行译码，执行指令，将 0x62 地址上的数据读取到 R2 寄存器，同时 PC 寄存器进行加 1 操作。

(10) 根据 PC 寄存器当前地址，读取 Multi R0,R1,R1 指令，并进行译码，执

行指令，将寄存器 R0 和 R1 的数据存储到乘法器，执行乘法运算，并将结果存入到 R1 寄存器。

(11) 根据 PC 寄存器当前地址，读取 Add R1,R2,R2 指令，并进行译码，执行指令，将寄存器 R1 和 R2 的数据存储到加法器，执行加法运算，并将结果存入到 R2 寄存器。

(12) 根据 PC 寄存器当前地址，读取 Store 0x63,R2 指令，并进行译码，执行指令，将寄存器 R2 的数据存储到存储器 0x63 地址。

(13) 根据 PC 寄存器当前地址，读取 HALT 指令，并进行译码，执行指令，终止指令执行。

(14) 计算机根据指令得到最终正确的计算结果，完成教学实验。

### 3、卷积计算原理



(1)点击“卷积计算原理”进入测评模式。进入测评模式后首先会看到弹出的“实验说明”窗口，在这个模式中，学生需要根据前面学习模式中学到的基本操作方法自主完成如窗口中所示的卷积计算过程，即完成一个 3X3 矩阵与一个 2X2 矩阵的卷积计算，得到一个 2X2 的结果矩阵，并将结果写入到存储器中。

(2)开始测评模式后，首先点击右下角的初始化清空存储器，然后依次设置代码段，卷积矩阵存放数据段，卷积核存放方式，被卷积矩阵数据段，被卷积矩阵存放方式以及存放临时数据的数据段，设置完成后准备开始设置指令并按照指令开始计算。

(3)开始输入指令，点击第二步中设置的代码段，每个地址可以存储一条指令。指令分为 Load / Multi/ Add / Store / HALT 五种类型，输入时首先输入对应的指令类型，然后根据指令类型输入所需的地址或数据。请学生根据学习模式中

的经验自主输入可以完成卷积计算的指令，并在最后添加一条 **HALT** 结束指令，指令输入完成后点击左下方的“完成填写”即可。

(4)指令输入结束开始准备指令的执行，设置 **PC** 寄存器的初始值，此值应指向卷积计算过程的第一条指令。

(5)开始指令的执行过程，**PC** 设置完成后虚拟仿真程序会从初始值指向的存储器地址依次取指、译指、执行，并依次累加直至执行到 **HALT** 指令结束。程序运行失败时应返回步骤三、四重新设置，运行成功时还需校验卷积计算结果。

(6)正确的卷积计算指令应得到 **2X2** 的结果矩阵，并存储在存储器中相应的位置，填写结果矩阵在存储器中的存储位置，与正确结果做比对。结果正确时会有相应显示，错误时可选择返回之前的步骤重新设置，直至全部结果正确，实验结束。