

计算机组成原理 实验(五)

实验项目名： 指令执行控制实验

实验类型： 设计性实验

实验设备与软件环境： PC 机、组成原理实验环境

实验要求：

(1) 理解 CPU 执行一条指令的过程中，主要寄存器 PC 和 IR 的作用及变化。

(2) 理解汇编语言指令和机器指令之间的关系，理解实验台给定指令系统的编码规则，能够运用其编写简单的程序。

(3) 运用单步方式调试程序，理解机器指令与微指令的关系，牢固建立计算机的整机概念。

实验说明：

本次实验用到前面四个实验中的所有电路，包括运算器、存储器、通用寄存器堆、程序计数器、指令寄存器、微程序控制器等，将几个模块组合成为一台简单计算机。因此，目前而言，这是最复杂的一个实验，也是最能得到收获的一个实验。

在前面的实验中，实验者本身作为“控制器”，完成数据通路的控制。而在本次实验中，数据通路的控制将由**微程序控制器**来完成。CPU 从内存取出一条机器指令到执行指令结束的一个机器指令周期，是由微指令组成的序列来完成的，即一条机器指令对应一个微程序。

1、本次实验的运行模式（仿真环境）

- 采用微程序方式；
- 单步运行-以微指令为单位，分解一个机器指令，执行程序。

2、实验台给定指令系统说明

实验台已经固化的微程序支持一个拥有 11 条指令的指令系统。这些指令均为单字节（8 位）指令，指令格式及功能如下表所示。

请结合课堂上学过的机器指令的格式，理解下列指令的操作码和地址码的编码、使用的寻址方式。**观察操作码的编码，请问还有多少可用的空闲编码？**

表 1 机器指令编码表

指令助记符	功能	指令格式							
		操作码				地址码			
		7	6	5	4	3	2	1	0
ADD Rd,Rs	(Rs)+(Rd)->Rd	0	0	0	0	RS1	RS0	RD1	RD0

SUB Rd,Rs	(Rd)-(Rs)->Rd	0	0	0	1	RS1	RS0	RD1	RD0
AND Rd,Rs	(Rd)&(Rs)->Rd	0	0	1	1	RS1	RS0	RD1	RD0
STA Rd,[Rs]	Rd->(Rs)	0	1	0	0	RS1	RS0	RD1	RD0
LDA Rd,[Rs]	(Rs)->Rd	0	1	0	1	RS1	RS0	RD1	RD0
JMP [Rs]	(Rs)->PC	1	0	0	0	RS1	RS0	X	X
JC D	若 C=1 , 则 (PC)+D ->PC	1	0	0	1	D3	D2	D1	D0
STP	暂停运行	0	1	1	0	X	X	X	X
IRET	返回断点	1	0	1	0	X	X	X	X
INTS	允许中断	1	0	1	1	X	X	X	X
INTC	禁止中断	1	1	0	0	X	X	X	X

3、助记符格式的汇编指令转换为二级制的机器指令

由于二进制形式的机器指令难记、调试困难，因此在实际使用时，计算机编程人员使用的最底层的编程语言并不是机器语言，而是汇编语言。将机器指令中的二进制编码使用英文缩写的助记符号进行表示，就得到了汇编语言，所以汇编指令基本上与机器指令存在一一对应的关系。

在本次实验中，我们怎样将助记符形式的指令转换为实验台能够理解的机器指令呢？可以通过以下两种方式：

(1) 由实验环境根据规则文件进行转换，类似与一个简单的汇编器。（仿真时无法使用此方式。）

实验台提供一个规则文件，该文件中定义了助记符指令与二进制机器指令的对应关系。我们可以在实验环境下的规则文件窗口中查看该文件的内容。在后续进行指令系统设计时，也可以根据自身设计的指令，修改或重新定义规则文件。

(2) 人工转换

根据表 1 的机器指令编码规则，采用人工方式，将书面表达的汇编程序，转换成机器码后，再输入到存储器中。

实验题目 1-1: 请将下表中的汇编程序，逐条手工编成十六进制的机器代码。

表 2 助记符小程序

地址	指令	机器代码
00H	LDA R0,[R2]	
01H	LDA R1,[R3]	
02H	SUB R0,R1	
03H	JC +5	

04H	AND R3,R2	
05H	SUB R3,R2	
06H	STA R3,[R2]	
07H	STP	

实验题目 1-2:

仔细研读表 1，尝试写出以下指令的助记符形式和机器码形式：

- ① 寄存器 R2 的内容减去 R1 的内容，结果存入 R2；
- ② 寄存器 R0 的内容加上 R1 的内容，结果存入 R0；
- ③ 向内存地址 F0H 存入一个数；
- ④ 向内存地址 F0H 取出数据；
- ⑤ 无条件跳转到地址 00H 处；
- ⑥ 根据运算产生的借位标志，如果有借位则跳转到地址 08H 处；

4、实验连线

仿真实验不需要连线，但需要了解为什么有些看似无关的控制信号会同时有效。本质上是由于硬件实验台将两个控制信号线连接在一起，以压缩微程序操作控制字段的长度。

本次实验由微程序控制器根据当前 IR 寄存器中存放的机器指令，产生相应的控制信号。

(1) 由于微指令编码过程中，将某些控制信号进行了合并，因此需要将以下信号两两连接在一起。这些信号如下：

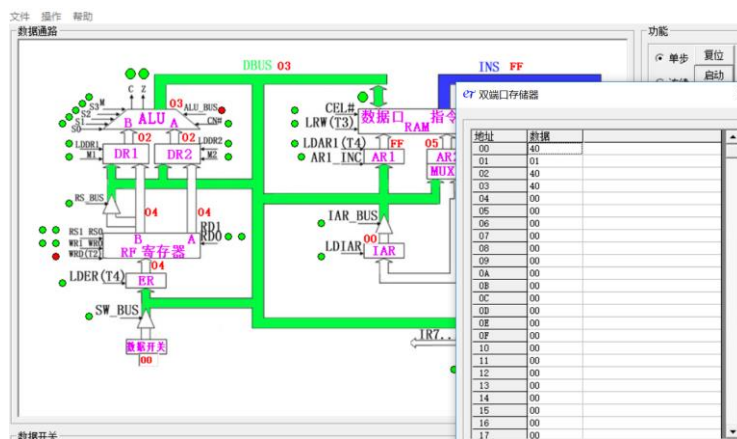
- LDAR1 和 LDAR2；
- LDIR 和 CER；
- LDDR1 和 LDDR2；
- LDPC 和 LDR4；
- M1 和 M2

(2) 由于寄存器号由 IR 寄存器中指令的地址码字段指出，因此需要将寄存器的选择信号与 IR 的相应位进行连接，包括：

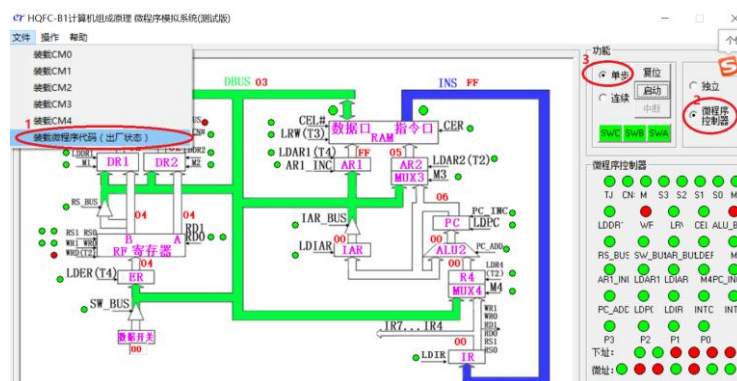
- IR3、IR2 分别连接 RS1、RS0；
- IR1、IR0 分别连接 RD1、RD0 和 WR1、WR0。

实验内容：

(1) 理解实验台给定微程序 CU 支持的指令系统。熟悉其中主要的 8 条指令，能够对照编码表，写出指令的助记符形式，并将其翻译为机器指令格式。通过编辑仿真环境中的双口 RAM，将机器指令格式的程序写入存储器中。



(2) 观察微程序方式下，采用**微程序方式、单步运行指令**。每一步都是运行一条微指令，该微指令包含多个微操作（发出多个微操作控制信号），请大家仔细观察取指周期中实验台发出的**控制信号**，观察 PC 和 IR 的变化，进一步理解取指周期微操作序列。



(3) 分析给定小程序，根据当前寄存器的值，估计每条执行的执行结果。单步运行改程序，**观察 PC、IR 以及指令操作的目的寄存器的值**，验证对程序功能的理解是否正确。

(4) 在理解指令的功能、汇编程序结构的基础上，尝试编写两个内存单元内容交换的小程序，并验证其正确性。

实验报告要求:

- (1) 写出实验题目 1-1、1-2。
- (2) 单步运行程序，分析每一步 CU 发出的控制信号，并写出对应的微操作。（记录）
- (3) 记录在单步运行实验题目 1-1 的过程中，观察到的 PC\IR 及相关寄存器的变化
- (4) 写出内存单元交换程序的清单，并分析是如何验证运行结果是否正确的。