**深 圳 大 学 实 验 报 告**

|  |
| --- |
| **课程名称 计算机系统1**  **项目名称 LC-3中断实验**  **学 院 计算机与软件学院**  **专 业 计算机科学与技术**  **指导教师 陈飞**  **报 告 人 刘俊楠 学号 2017303010**  **实验时间 2021/6/17**  **提交时间 2021/6/19** |

**教务处制**

# 一、实验目的与要求

实现以下三部分程序：

* 用户程序
* 键盘中断服务程序
* 操作系统支持的代码

**二、实验问题**

在计算机科学中，一个中断就是有硬件和软件发起的一个事件，表明需要立即处理。

高优先级的条件下，一个中断通知处理器需要中断正在运行的程序，处理器挂起正在处理的行为，保存执行程序的状态，然后执行一个小的中断处理程序（中断服务程序）来处理事件。

中断是临时的，当处理器执行完中断处理程序后，处理器继续执行之前被中断的程序，中断分硬件中断和软件中断。

使用实验证明：中断驱动的输入、输出可以中断一个正在运行的程序，执行中断服务程序，返回被中断的程序，从被中断位置下一个地址继续执行（好像什么也没发生似的)。

试验中我使用键盘作为输入设备，中断正在运行的程序。

试验任务包括实现以下三部分程序：

* **用户程序**

用户程序将会连续地输出纵横交替的ICS，通过交替，输出两个不同行，如下：

ICS ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS

确保输出不至于太快，以至于肉眼不能察觉。因而，需要延时操作，可以考虑如下实现：

即用户程序包含一小段代码用于每行间的计数，间隔为从2500开始倒计时，计时结束时，再进行输出

* **键盘中断服务程序**

键盘中断服务程序将会简单地在屏幕上写上十次，用户随机输入的字符并以Enter（x0A)结束。

中断服务程序中要求不使用TRAP指令。

注意，不要忘记保存和恢复在中端服务程序中使用的那些寄存器。

* **操作系统支持的代码**

不幸的是，我们还没在LC-3上安装windows或Linux，所以我们必须要求你在你的用户程序代码前先做到以下**三个步骤：**

* 正常情况下，操作系统将会先安装一些栈空间，所以当中断发生的时候 PC和PSR可以被放进栈中(正如你知道的，当程序执行RTI，PC和PSR都会被弹出栈，处理器返回到执行被中断的程序）由于没有操作系统，请先把R6初始化为x4000,表示一个空的栈。
* 正常情况下，操作系统会建立中断向量表，它包含对应中断服务程序的起始地址，你必须为键盘中断先建立一个中断向量表。中断向量表的开始地址是x0100，键盘中断的中断向量是x80。你必须在中断向量表提供一个入口供本实验使用。
* 操作系统应该设置KBSR的IE(Interrupt Enable)位。

**三、实验步骤与过程**

（依照实验内容，逐条撰写实验过程与实验所得结果：包括程序总体设计，核心数据结构及算法流程，调试过程。请附上核心代码，及注意格式排版的美观。实验提交时，以上为评分依据，请不删除本行）

实验环境：

硬件：Pc电脑、windows10系统

软件：LC3Edit、LC3Simulate

# 画流程图，明白思路

此题题意为实现中断处理，主要思路如下图（图3-1）：

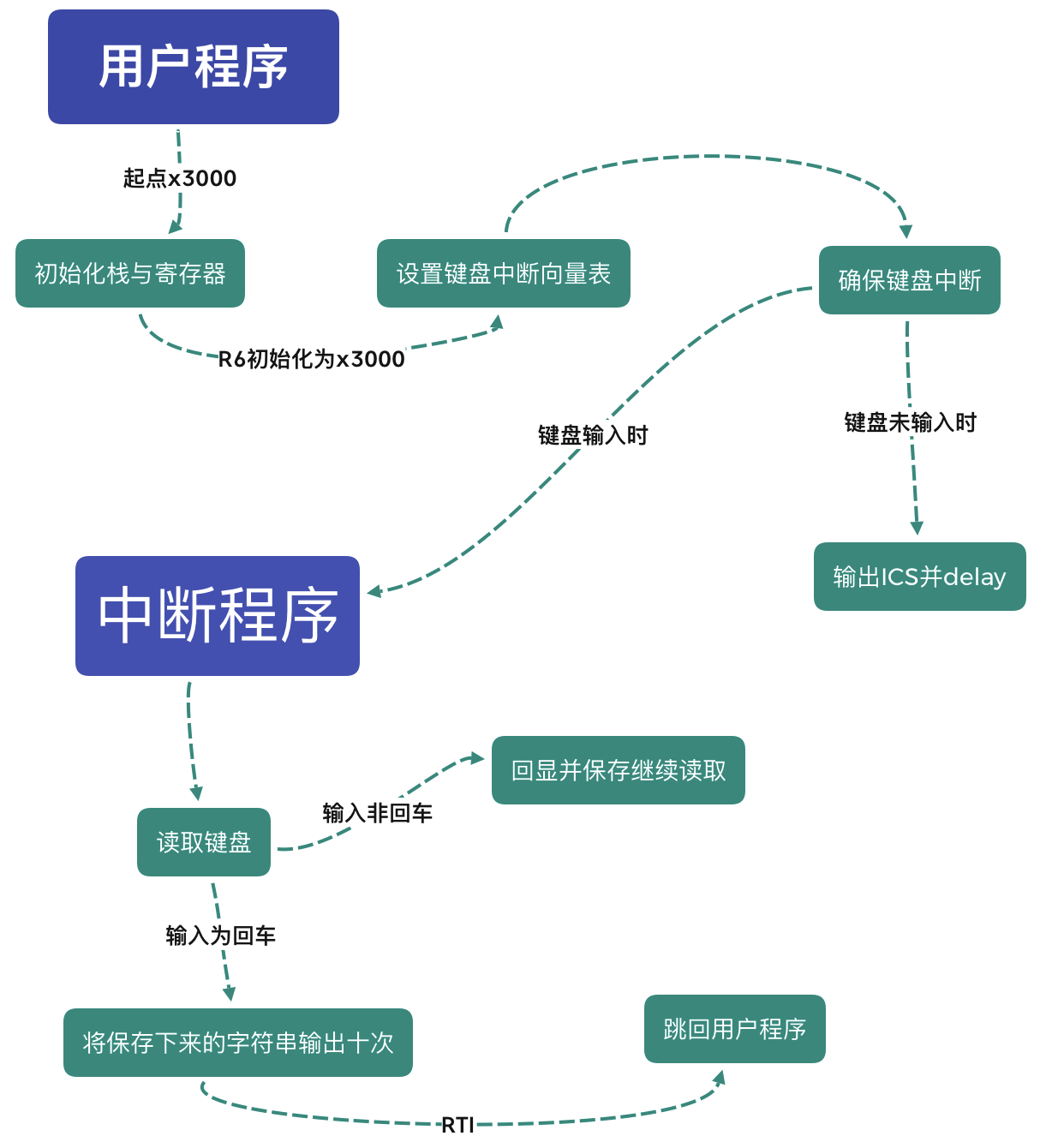


图3-1

# 代码实现

题目实现代码如下：

* Main函数（用户程序）：

.ORIG X3000

;初始化栈与寄存器

AND R0,R0,#0;

AND R1,R1,#0;

AND R2,R2,#0;

AND R3,R3,#0;

AND R4,R4,#0;

AND R5,R5,#0;

AND R7,R7,#0;

LD R6,RR6;

;设置键盘中断向量表

LD R5,ZDD;

STI R5,KWORD;

;确保键盘中断

LD R4,A; R4不可用

STI R4,KBSR;

;开始输出ICS并开始delay

LOP AND R1,R1,#0;

ADD R1,R1,#6;

LOOP LEA R0,ICS;

PUTS

JSR DELAY

ADD R1,R1,#-1;

BRp LOOP;

LEA R0,HC;

PUTS

ADD R1,R1,#6;

LOOP1 LEA R0,ICS1;

PUTS

JSR DELAY

ADD R1,R1,#-1;

JSR DELAY

BRp LOOP1

LEA R0,HC;

PUTS

BRnzp LOP

DELAY ST R1, SaveR1

LD R1, COUNT

REP ADD R1,R1,#-1

BRp REP

LD R1, SaveR1

RET

HC .stringz "\n"

COUNT .FILL #2500

SaveR1 .BLKW 1

ICS .stringz "ICS "

ICS1 .stringz " ICS"

KBDR .FILL XFE02

KBSR .FILL XFE00;

A .FILL X4000

SIX .FILL X09C4;是2500 delay

ZDD .FILL X2000

KWORD .FILL X0180

RR6 .FILL X3000

.END

* TRAP函数的代码如下：

.ORIG X2000

;被调用者保存

ST R0,SAVER0

ST R1,SAVER1;

ST R2,SAVER2

ST R3,SAVER3

ST R4,SAVER4

ST R5,SAVER5

;初始化

AND R1,R1,#0

ST R1,LENGTH

;读取键盘

LOOP0 LDI R3,KBDR

AND R1,R1,#0

ADD R1,R1,#-1

STI R3,DDR

STI R1,DSR

;判断是否为回车

LD R4,HUICHE

ADD R4,R3,R4

BRz NEXT

;输入的值计入长度

LEA R1,SHUZU

LD R2,LENGTH

ADD R1,R1,R2

STR R3,R1,#0

ADD R2,R2,#1

ST R2,LENGTH

AND R2,R2,#0

;输入过程回显

STI R2,KBSR

LDI R3,KBDR

NOT R3,R3

ADD R3,R3,#1

LD R1,NKBSR

STI R1,KBSR

;轮询等待输入

LOOP3 LDI R4,KBSR

BRzp LOOP3

BRnzp LOOP0

;输入为回车时

NEXT AND R3,R3,#0

ADD R3,R3,#10

AND R1,R1,#0;

ADD R1,R1,#-1

STI R3,DDR

STI R1,DSR

AND R2,R2,#0

ADD R2,R2,#10

LOOP1 LD R0,LENGTH

LEA R4,SHUZU

LOOP2 LDR R3,R4,#0

AND R1,R1,#0;

ADD R1,R1,#-1

STI R3,DDR

STI R1,DSR

ADD R4,R4,#1

ADD R0,R0,#-1

BRnp LOOP2

LD R5,SPACE

AND R1,R1,#0

AND R1,R1,#-1

STI R5,DDR

STI R1,DSR

;双重循环输出十次字符串

ADD R2,R2,#-1

BRp LOOP1

AND R1,R1,#0

ADD R1,R1,#-1

STI R1,KBSR

LD R1,NKBSR

STI R1,KBSR

;调用者读取

LD R0,SAVER0

LD R1,SAVER1

LD R2,SAVER2

LD R3,SAVER3

LD R4,SAVER4

LD R5,SAVER5

OVER RTI

HALT

SPACE .FILL X000A

NKBSR .FILL X7FFF

HUICHE .FILL #-10

SHUZU .BLKW X50

DDR .FILL XFE06

DSR .FILL XFE04

KBDR .FILL XFE02

KBSR .FILL XFE00

SAVER0 .FILL X0

SAVER1 .FILL X0

SAVER2 .FILL X0

SAVER3 .FILL X0

SAVER4 .FILL X0

SAVER5 .FILL X0

LENGTH .FILL X0

.END

载入界面如下图所示（图2-1）

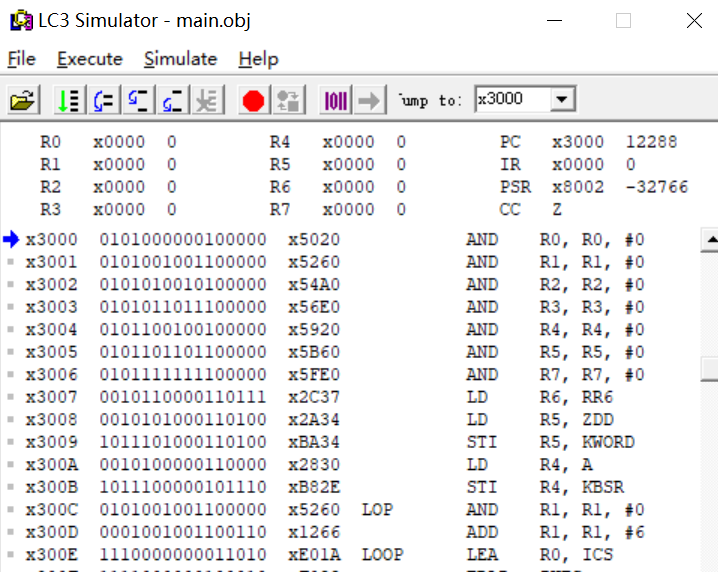


图2-1 代码载入界面

# 结果展示

3-1对载入的代码运行，可以见到以肉眼可见的速度连续输出ICS。（图3-1-1）

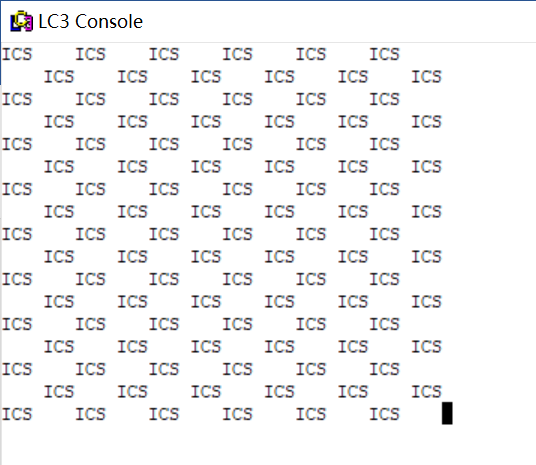


图3-1-1

3-2 当键盘输入Liujunnan\_2017303010时，程序自动暂停，进入trap程序，然后当键入回车时，程序自动回显输入十次，并返回用户程序继续输出ICS（图3-2-1）。

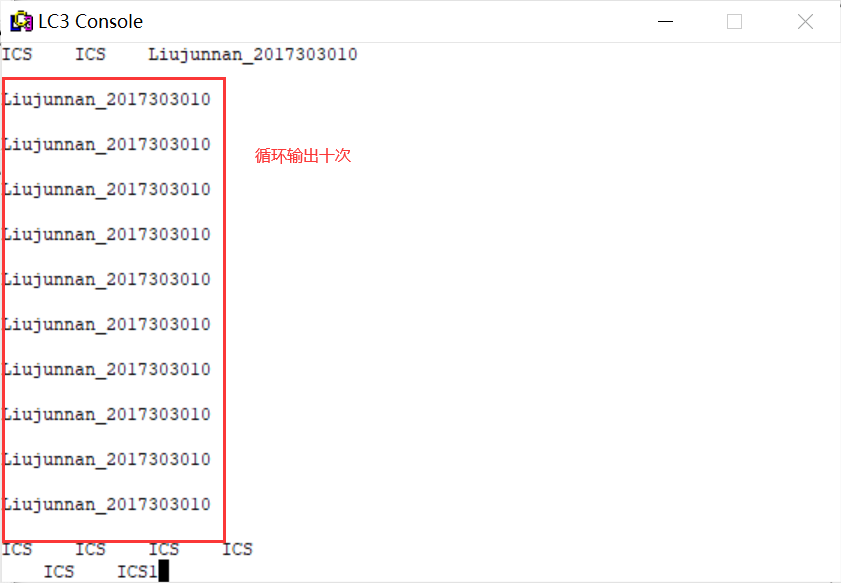


图3-2-1

# 错误与改正

实验过程中，我遇到了许多疏漏与错误，总结如下：

* 概念混淆，KBSR与KBDR概念错误，KBSR为键盘状态寄存器，地址为XFE00，当最高位为1时，表明已输入，为0时，表明未输入，键盘可继续输入。KBDR为键盘数据寄存器，地址为XFE02，可存储1位数据。
* KBSR、KBDR与DDR、DSR混淆，KBSR与KBDR为输入，DDR与DSR为输出DSR与KBSR类似，不过DSR最高位1时表明显示器空闲，为0时表示显示器忙。DDR为回显的数据。
* 用户程序必须在初始时初始化R6为用户程序起始地址，否则trap程序执行完后返回不到用户程序
* 在用户程序必须初始化键盘向量中断表，确保键盘中断是启动的

**四、实验结论或体会**

（撰写实验收获及思考）

**结论：**

通过本次实验，我学会了利用LC3EDIT软件设置中断程序，并且实现了用户程序与中断程序的一个跳转，让我对中断的概念有了更加清晰地认识，并且明白了保存现场与返回地址的重要性，此外，也明白了操作系统的复杂性，为以后的操作系统的认识打下基础。

|  |
| --- |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。