**深 圳 大 学 实 验 报 告**

|  |
| --- |
| **课程名称 计算机系统1**  **项目名称 实验六 LC-3中断实验**  **学 院 计算机与软件学院**  **专 业 软件工程**  **指导教师 陈飞**  **报 告 人 学号**  **实验时间 2025年6月19日**  **提交时间 2025年6月22日** |

**教务处制**

# 一、实验目的与要求

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)。

# 二、实验内容与方法

实验任务包括实现以下三部分程序：

A. 用户程序

B. 键盘中断服务程序

C. 操作系统支持的代码

各程序描述见后。

A. 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

一个简单的实现方法是下面的程序段DELAY：

DELAY ST R1, SaveR1

LD R1, COUNT

REP ADD R1,R1,#-1

BRp REP

LD R1, SaveR1

RET

COUNT .FILL #2500

SaveR1 .BLKW 1

B. 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断服务程序中使用的那些寄存器。

C. 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：

1.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。

2.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。

3.操作系统应该设置KBSR的IE(Interrupt Enable)位。

**样例程序框架：用户程序**

.ORIG x3000

-- --- ; initialize the stack pointer

...

-- ---

-- --- ; set up the keyboard interrupt vector table entry

...

-- ---

-- --- ; enable keyboard interrupts

...

-- ---

-- --- ; start of actual user program to print ICS checkerboard

...

-- ---

.END

**样例程序框架：中断服务程序**

.ORIG x2000

-- --- ; the code

...

-- ---

RTI

-- --- ; buffer space as required

...

-- ---

.END

# 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）

**1.lc-3中断的具体流程**

①. 中断请求 某个设备（例如键盘、时钟等）发出中断请求。这个请求通过设置特定的中断请求标志来通知处理器。

②. 检测中断 处理器在每次执行指令后会检查是否有中断请求。处理器会检查中断启用标志（Interrupt Enable, IE）和中断请求标志（Interrupt Request, IR）。如果中断启用标志被设置，并且有中断请求，处理器将开始中断处理。

③. 保存当前状态 处理器需要保存当前程序的状态，以便在处理完中断后能够恢复。



④. 识别中断类型 中断向量表（Interrupt Vector Table, IVT）包含了各类中断的入口地址。处理器通过查找中断向量表，找到对应的中断服务程序的入口地址。

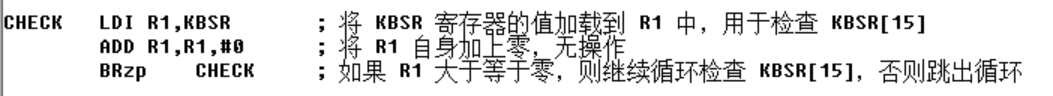
⑤. 跳转到中断服务程序 处理器将程序计数器（PC）设置为中断服务程序的入口地址，并开始执行中断服务程序的指令。

⑥. 执行中断服务程序

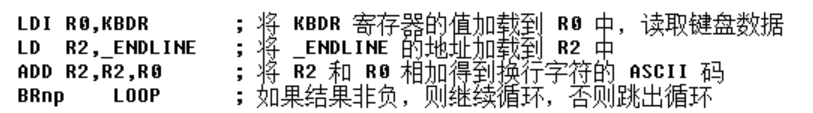
首先，将R0寄存器的值保存到SaveR0内存位置。



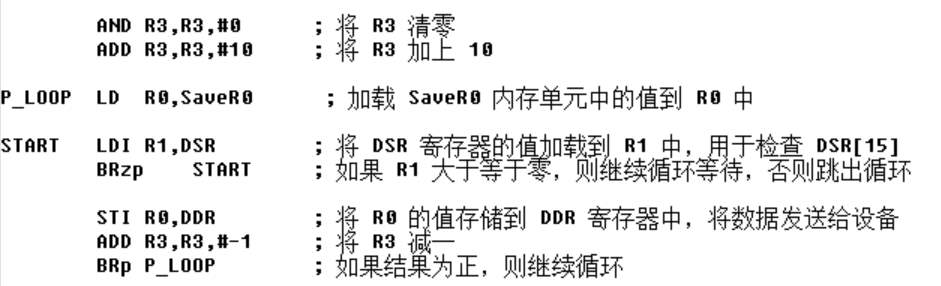
然后从键盘状态寄存器(KBSR)加载值到R1，并检查是否有按键输入。BRzp表示如果R1的值为零或正数，则继续循环CHECK。



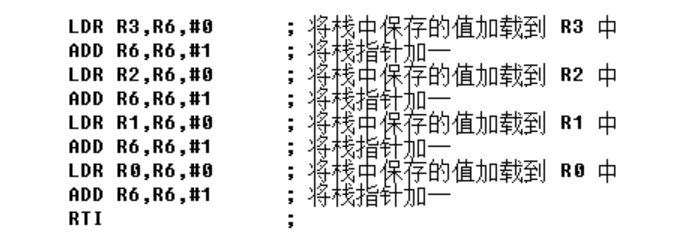
如果有按键输入，从键盘数据寄存器(KBDR)加载值到R0。然后加载特殊符号\_ENDLINE到R2，检查R0的输入是否为特定的结束符号，如果是则跳出循环。



将R3寄存器清零，然后设置为10。在P\_LOOP循环中，加载SaveR0的值到R0，检查显示状态寄存器(DSR)是否准备好接收数据。如果DSR[15] = 1，则将R0的值存储到显示数据寄存器(DDR)。然后R3递减1，继续循环直到R3为0。



⑦. 恢复之前的状态 中断服务程序完成后，处理器需要恢复之前保存的状态。通常包括恢复程序计数器（PC）和处理器状态寄存器（PSR）。 返回主程序 处理器继续执行中断之前的主程序，处理过程就此结束。



**2.user程序lc-3代码解释**

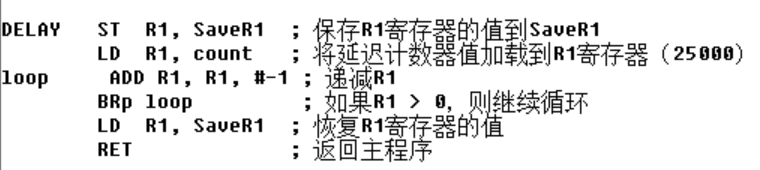
栈指针初始化，并将中断服务例程地址存储到中断向量表中，启用键盘中断。



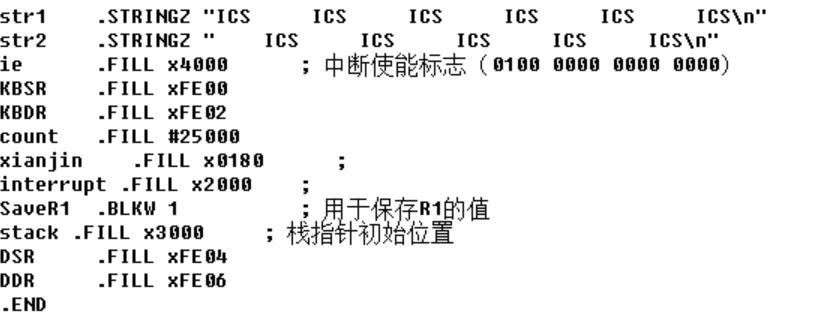
在主循环中，程序不断地输出两个字符串，每次输出之间插入延迟。TRAP x22用于输出字符串，JSR DELAY用于调用延迟子程序。



延迟子程序



字符串和常量



**3.运行结果**

①．加载user.obj 和 interrupt.obj

②．pc设置为3000

图形用户界面, 应用程序

描述已自动生成

③．开始运行并观察console输出

首先当键盘未输入字符时交替输出ICS字符

图片包含 表格

描述已自动生成

当键盘输入时程序停止，直到键盘输入换行，程序将输入的首个字符‘1’打印了10次

并继续进行循环交替打印

表格

中度可信度描述已自动生成

# 四、实验结论或体会

（撰写实验收获及思考）

1.中断处理程序中不能使用TRAP指令,则要显示字符时只能通过读写DSR的方式

2.中断处理程序中需备份用到的寄存器并恢复

3.为避免输出过快导致肉眼看不清,可加入计数延迟或无实际作用的代码以消耗时钟周期,达到程序sleep效果

4.通常情况下,当遇到中断发生之前,操作系统已经开辟好栈空间,保存PC和PSR,当执行到RTI时,PC和PSR会被弹栈，需要初始化R6为X3000指示一个空栈

5.操作系统会建立一张中断向量表,用来包含中断处理程序对应的起始执行地址,并将KBSR的lE(lnterrupt Enable)位置1

|  |
| --- |
| 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。