Федеральное агентство связи Федеральное государственное бюджетное учреждение

высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

Лабораторная работа 6

“Подсистема прерываний ЭВМ. Сигналы и их обработка. ”

Выполнили

Студенты группы ИП-813

Бурдуковский И.А

Стояк Ю.К

Проверил: Преподаватель кафедры В С Токмашева Елизавета Ивановна

Новосибирск, 2020

# 1. Постановка задачи

1. Прочитайте главу 6 практикума по курсу «Организация ЭВМ и систем». Изучите страницу man для функций signal, setitimer.
2. Доработайте консоль Simple Computer. Создайте обработчик прерываний от системного таймера так, чтобы при каждом его срабатывании при нулевом значении флага «игнорирование сигналов системного таймера» значение регистра – InstructionCounter увеличивалось на 1, а при поступлении сигнала SIGUSR1 состояние Simple Computer возвращалось в исходное. Обработка нажатых клавиш осуществляется только в случае, если сигналы от таймера не игнорируются

# 2. Описание алгоритма работы

**Функции:**

void sigHandler(int signal)- функция, проверяющая, какой сигнал поступил на вход функции. Если SIGALRM(Посылается процессу ядром при срабатывании таймера.

Каждый процесс может устанавливать не менее трѐх таймеров. Первый из них измеряет прошедшее реальное время.), то при нулевом значении флага

«игнорирование сигналов системного таймера» значение instructionCounter увеличивается на 1. Если же на вход функции подаётся сигнал SIGUSR1(пользовательские сигналы. Так же как и сигнал SIGTERM, эти сигналы никогда не посылаются ядром и могут использоваться для любых целей по выбору пользователя.), то значение флага «игнорирование сигналов системного таймера» устанавливается в 1, а состояние Simple Computer сбрасывается и возвращается в исходное.

При нажатии клавиши в терминале происходит проверка с помощью вызова системной функции signal для того, чтобы определить, какое действие необходимо выполнить при получении сигнала. В зависимости от полученного сигнала функция void sigHandler(int signal) выполняет определенные действия, описанные выше.

void sigHandler (int signal) {

switch (signal) {

case SIGALRM: {

int val;

if (!sc\_regGet(FLAG\_IGNORE\_CLOCK, &val) && !val) {

CU();

interface\_print();

}

}

break;

case SIGUSR1: {

sc\_memoryInit();

sc\_regInit();

registers.instruction\_counter = 0;

registers.accumulator = 0;

sc\_regSet(FLAG\_IGNORE\_CLOCK, 1);

select\_cell = 0;

}

break;

}

}

void create\_timer(double interval) {

struct itimerval nval;

nval.it\_value.tv\_sec = nval.it\_interval.tv\_sec = (long)interval;

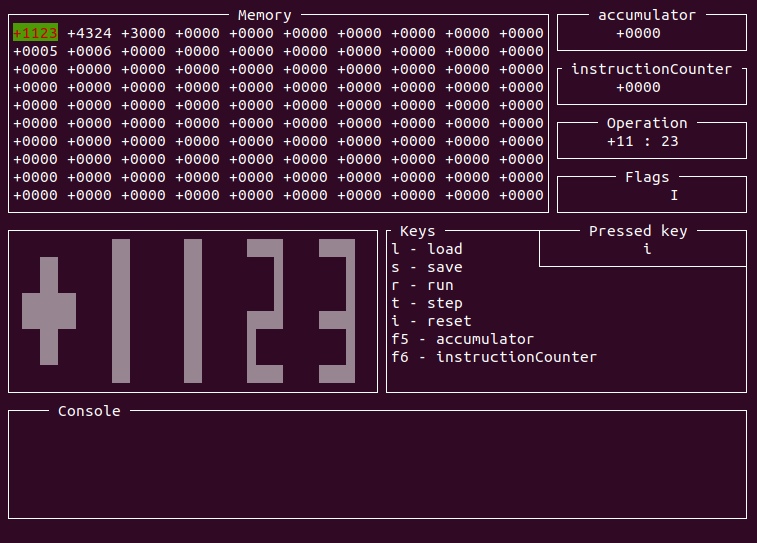
nval.it\_value.tv\_usec = nval.it\_interval.tv\_usec = (long)((interval - (long)interval) \* 1000000);

setitimer (ITIMER\_REAL, &nval, NULL);

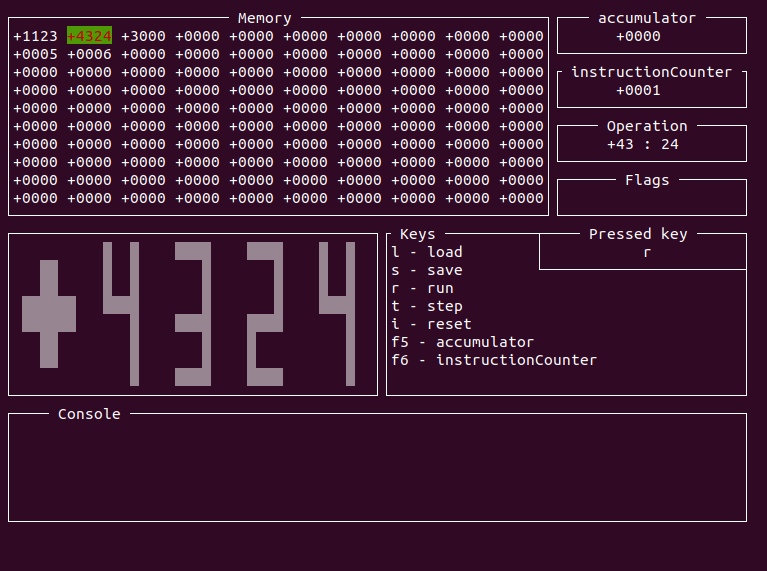
}

# 3.Результат работы алгоритма

**Начальное положение терминала:**



**После нажатия r, через несколько секунд:**



**После нажатия I изменяется значение флага, сброс значений:**

