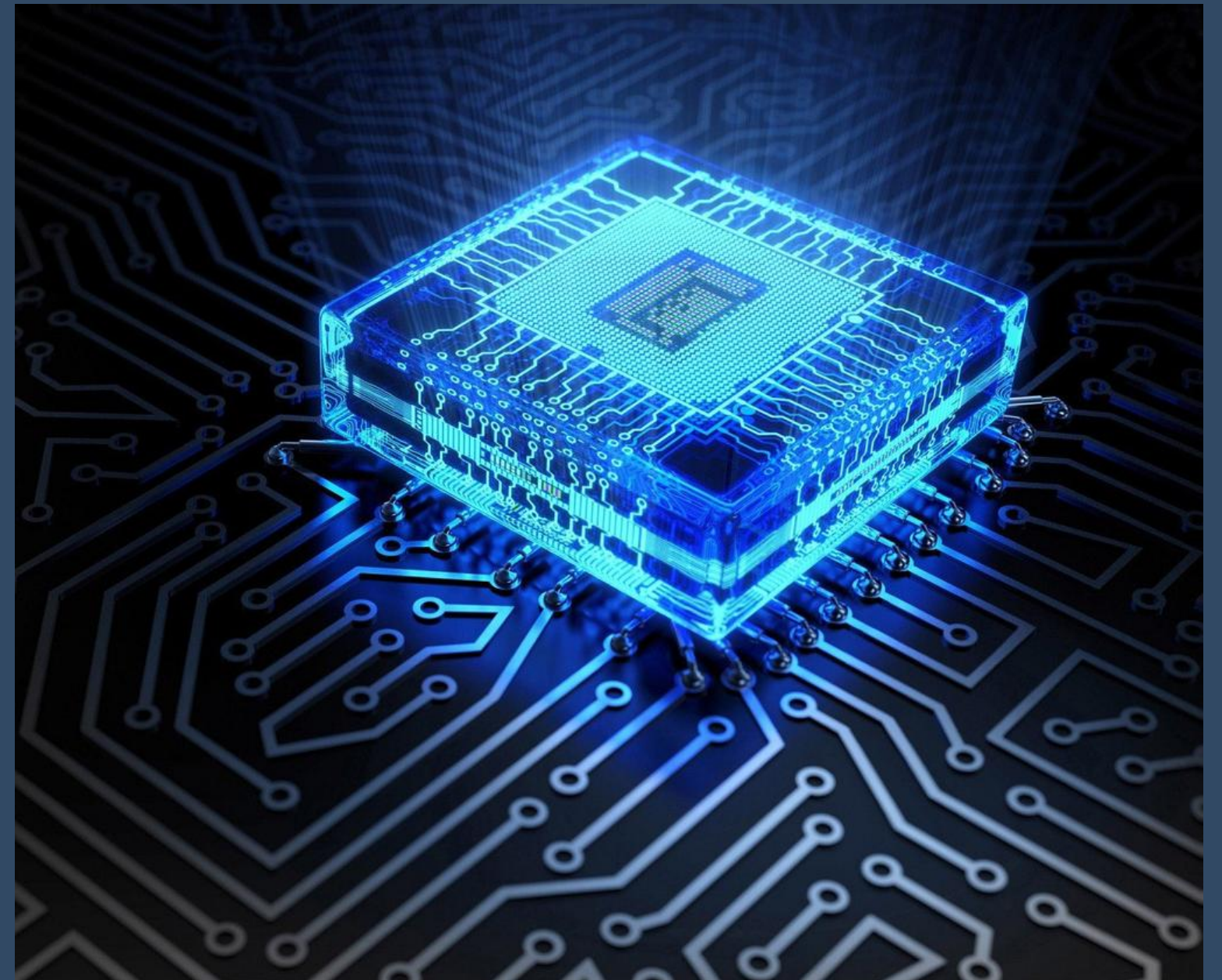




Пензенский  
государственный  
университет



Кафедра  
вычислительной  
техники



# Учебный кроссплатформенный эмулятор микропрограммируемой ЭВМ

АВТОРЫ: КОННОВ Н.Н., ПЧЕЛИНЦЕВ А.И., ФЕДУЛОВ В.Д., БЫЧКОВ А.С.

# Актуальность

---

**Подготовка современного специалиста в области вычислительной техники предполагает изучение архитектуры процессоров на двух уровнях:**

- Ассемблерном, с использованием стандартных средств низкоуровневого программирования
- Микропрограммном, с использованием эмуляторов ЭВМ

---

Использование программных эмуляторов микропрограммируемых ЭВМ в лабораторном практикуме и курсовом проектировании дисциплин, связанных с изучением архитектуры, позволяет студентам разрабатывать различные алгоритмы выполнения арифметических операций с фиксированной и плавающей точкой, выполнять микропрограммную имплементацию шестнадцатиразрядных ЭВМ с различной процессорной архитектурой

---

1

Разработаны под  
Windows платформу

---

2

Невозможность  
редактирования МК  
в процессе  
выполнения

---

3

Отсутствие  
графического  
редактора  
микропрограммы

---

4

Отсутствие  
возможности  
взаимодействия с  
периферийными  
устройствами

---

5

Отсутствие  
возможности  
обработки  
перерываний

---

6

Отсутствие  
возможности  
работы с ОЗУ и ПЗУ

# Недостатки известных эмуляторов

# Выбор решения



## Кроссплатформенность

Выбрали объектно-ориентированный язык программирования Java

## Редактирование МК

Доработали интерфейс программы

## Работа с ОЗУ

Разработали модуль и алгоритм взаимодействия с ОЗУ

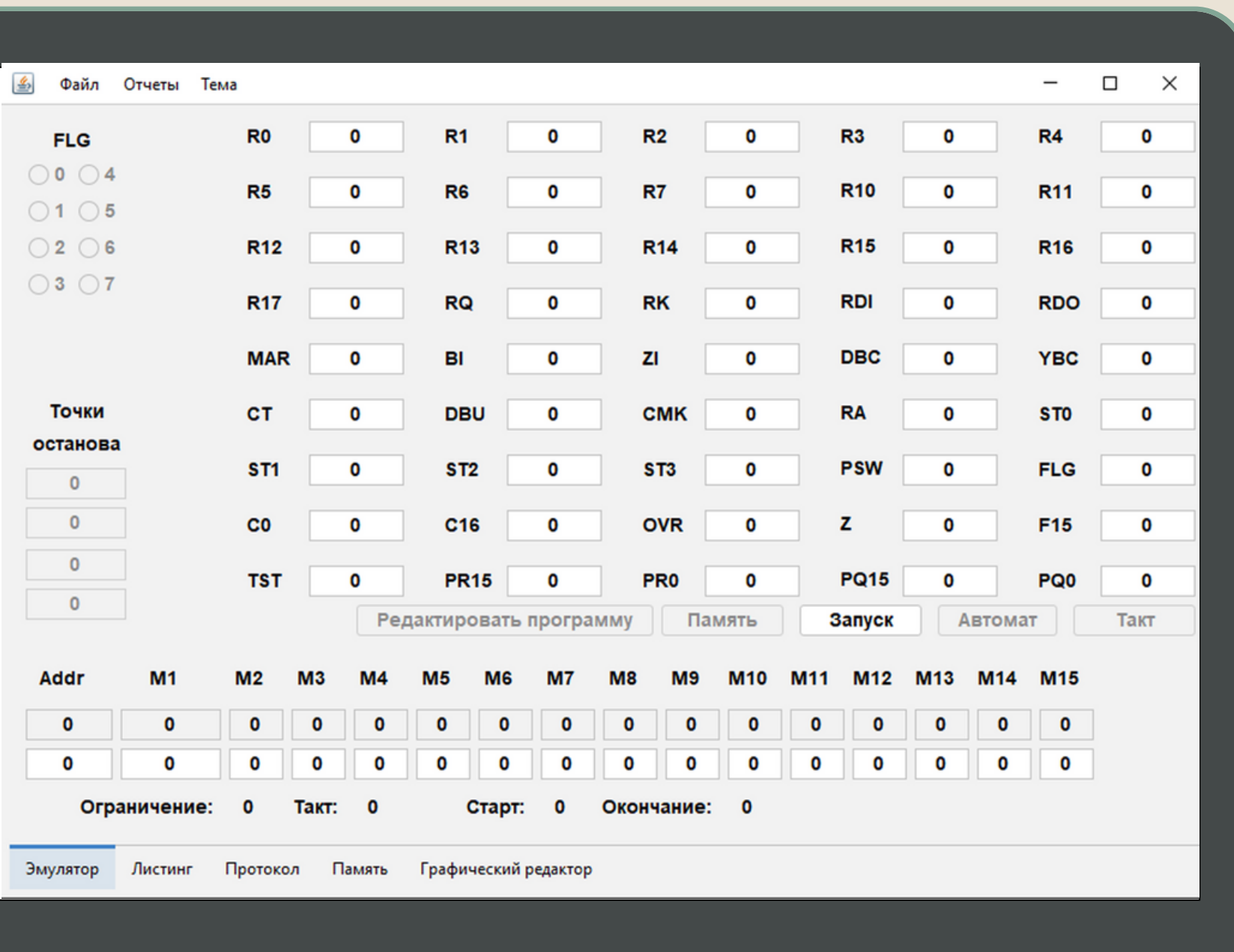
## Периферийная система

Доработали интерфейс программы

## Прерывания

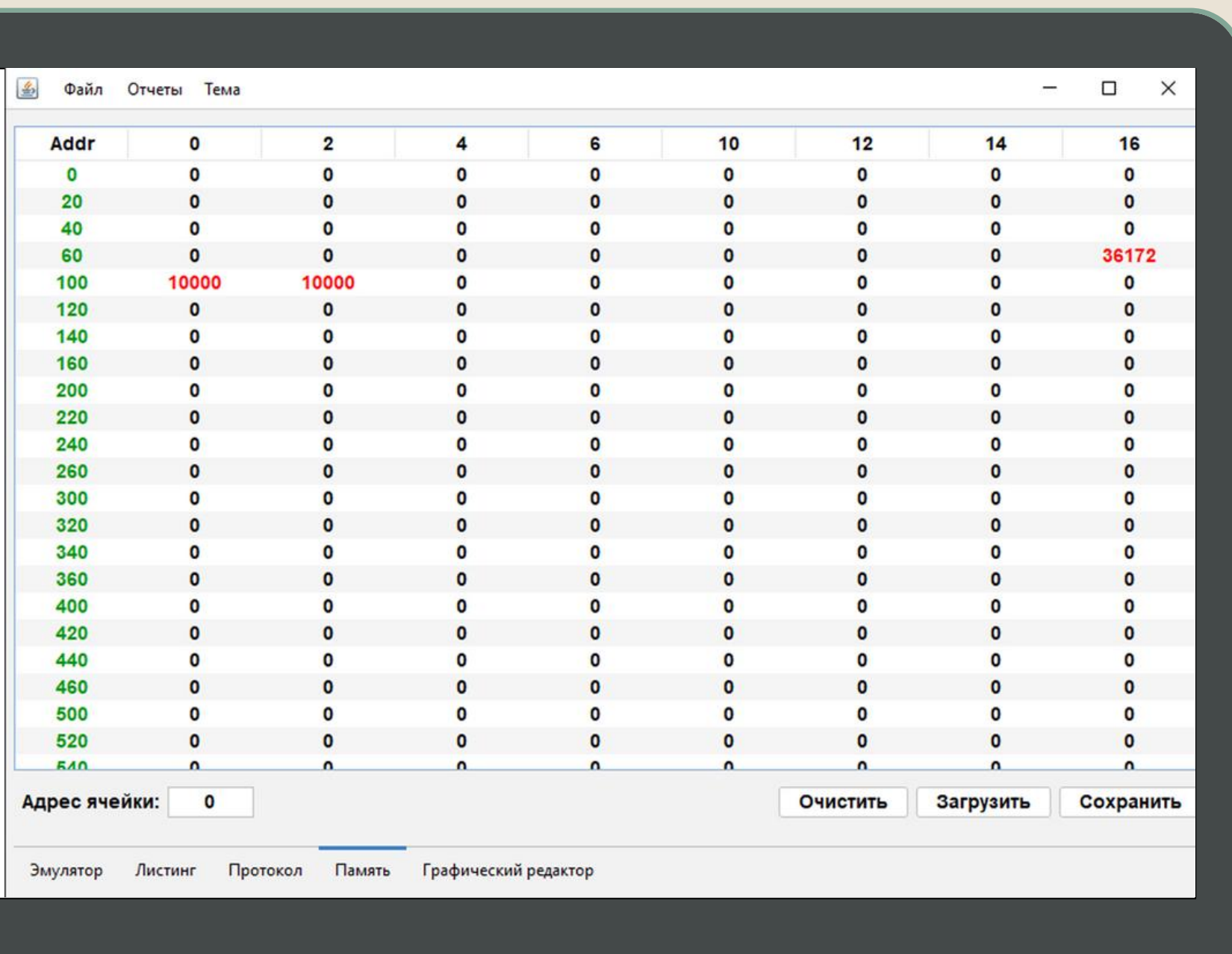
Разработали алгоритм обработки прерывания

# Главное окно



- Порты ввода внешних сигналов
- Точки останова
- Регистры микропроцессора
- Текущая и следующая микрокоманда
- Органы управление ходом отладки

# Окно памяти

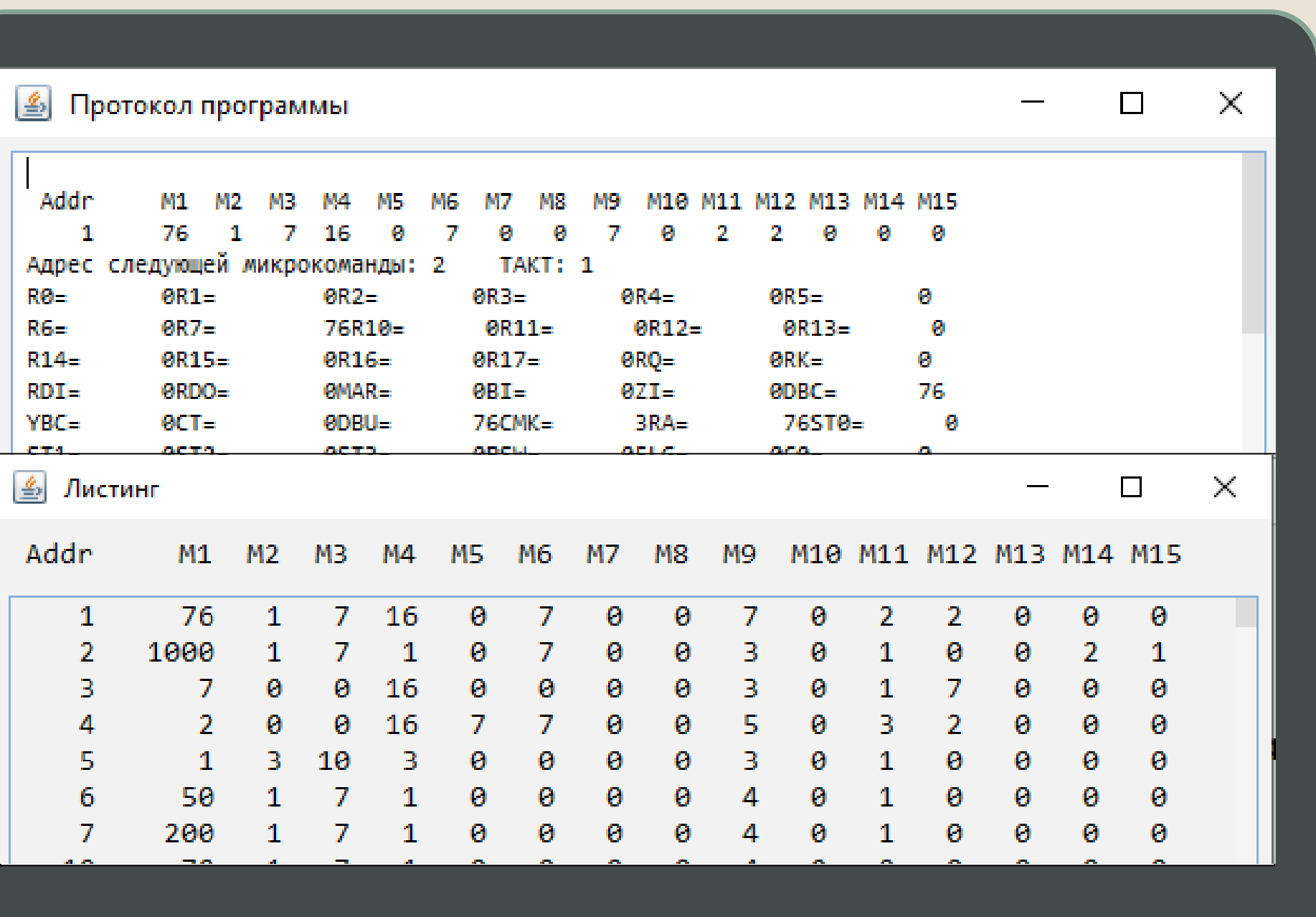


The screenshot shows a memory editor window with a menu bar (Файл, Отчеты, Тема) and a table of memory addresses. The table has columns for addresses 0, 2, 4, 6, 10, 12, 14, and 16. The values are mostly 0, with some non-zero values at addresses 100 and 60. Below the table is a search field for the address (0) and buttons for 'Очистить', 'Загрузить', and 'Сохранить'. At the bottom, there are tabs for 'Эмулятор', 'Листинг', 'Протокол', 'Память', and 'Графический редактор'.

Addr	0	2	4	6	10	12	14	16
0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	36172
100	10000	10000	0	0	0	0	0	0
120	0	0	0	0	0	0	0	0
140	0	0	0	0	0	0	0	0
160	0	0	0	0	0	0	0	0
200	0	0	0	0	0	0	0	0
220	0	0	0	0	0	0	0	0
240	0	0	0	0	0	0	0	0
260	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0
320	0	0	0	0	0	0	0	0
340	0	0	0	0	0	0	0	0
360	0	0	0	0	0	0	0	0
400	0	0	0	0	0	0	0	0
420	0	0	0	0	0	0	0	0
440	0	0	0	0	0	0	0	0
460	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0
520	0	0	0	0	0	0	0	0
540	0	0	0	0	0	0	0	0

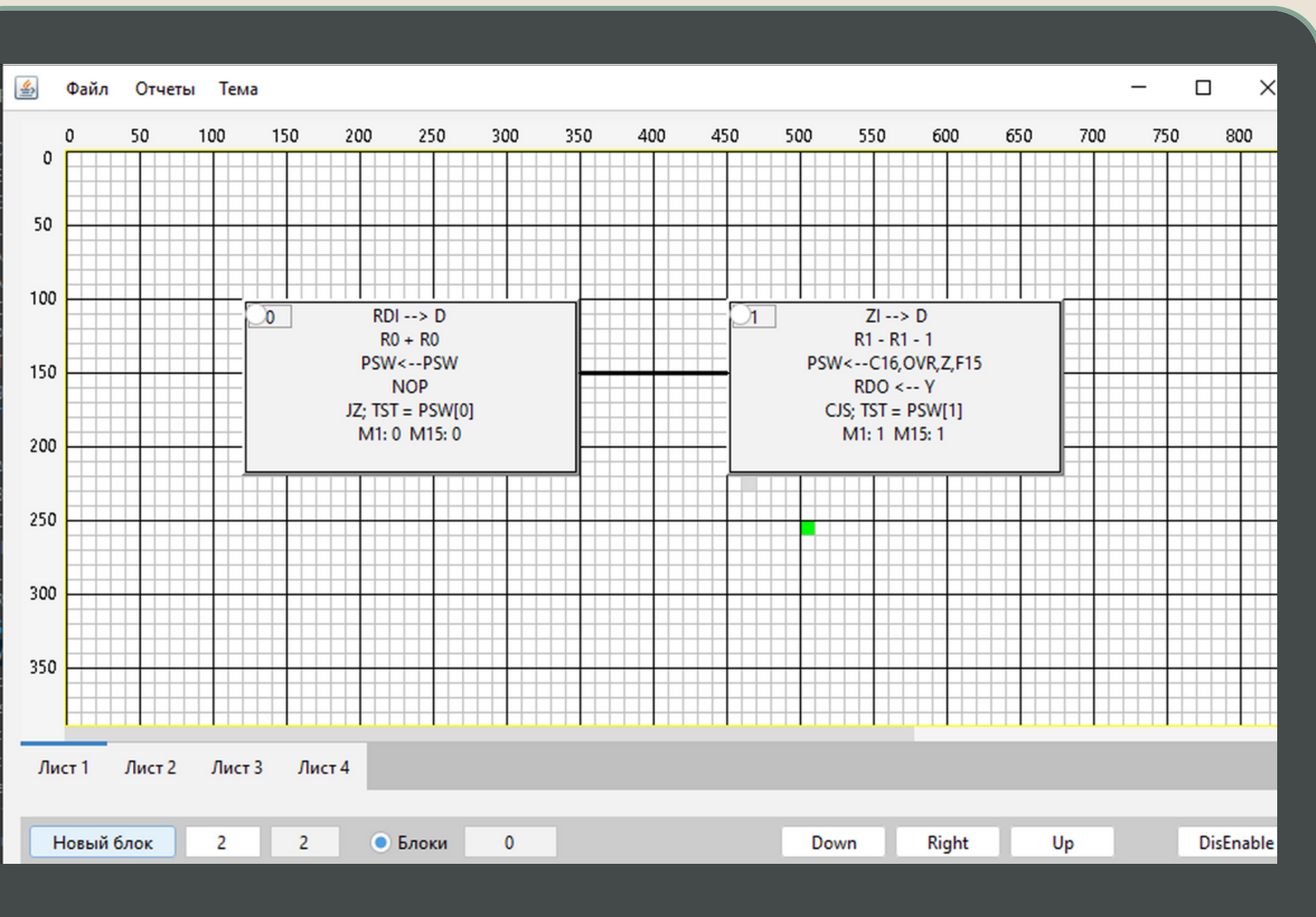
- Поиск ячейки памяти
- Очистка таблицы памяти
- Загрузка и сохранение
- Редактирование ячеек памяти

# Окна для отчетов



- Сохранение листинга и протокола
- Экспорт в текстовый файл
- Формирование протокола в процессе отладки

# Окно графического редактора

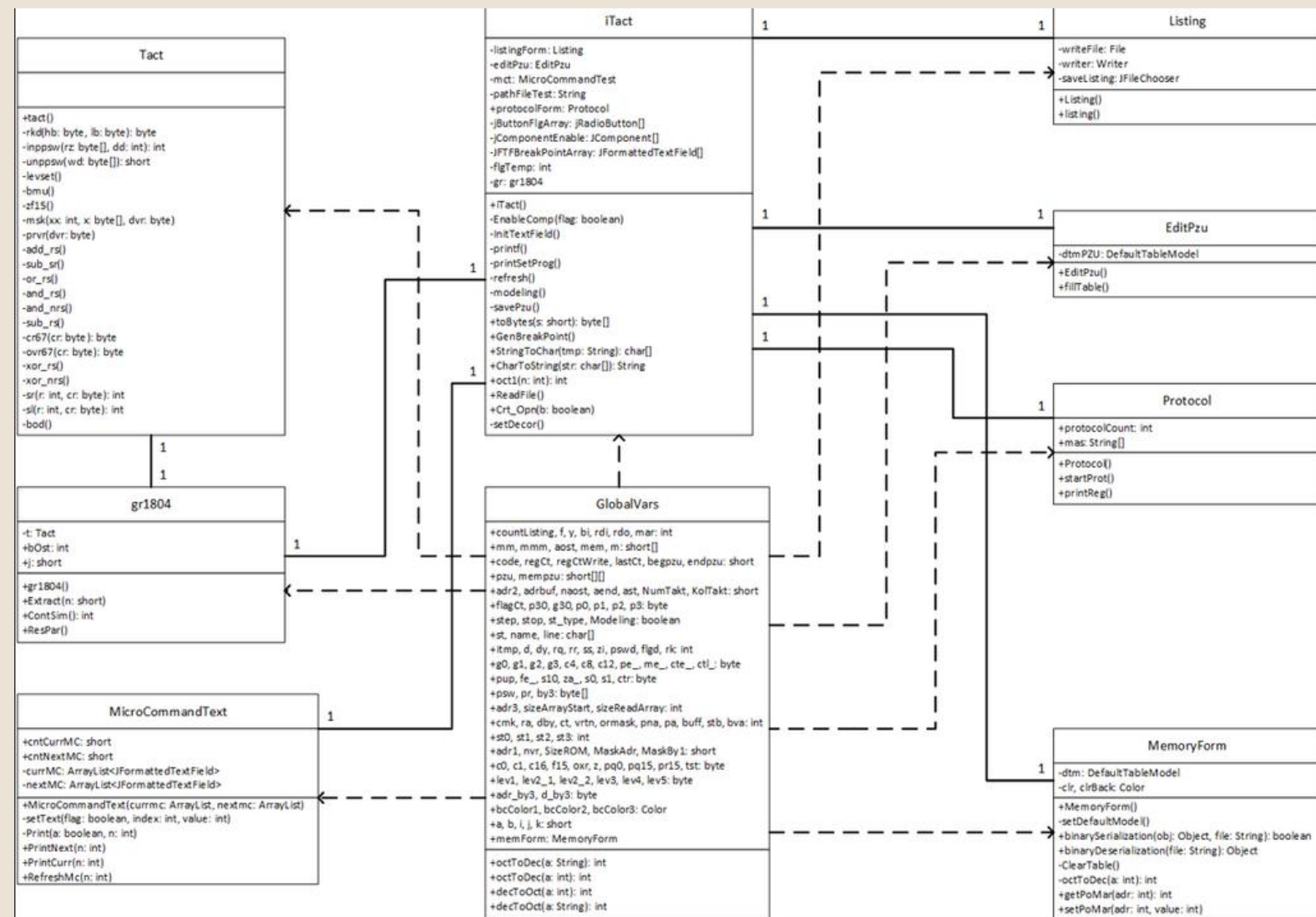


- Добавление и размещение блоков
- Редактирование МК
- Сохранение программы в памяти
- Возможность перемещения по рабочему пространству и листам программы



# Описание программы

Используется 7 основных классов, каждый из которых отвечает за свою область в программе и имеет свой набор методов



## Tact

Ядро микропроцессора

## Visio

Графический редактор

## EditPzu

Редактор программы

## Memory

Редактор памяти

## Protocol

Отчет отладки МП

## Listing

Листинг МП

## iTact

Главный класс реализующий окно эмулятора. Графический интерфейс реализован с помощью встроенной библиотеки Swing.

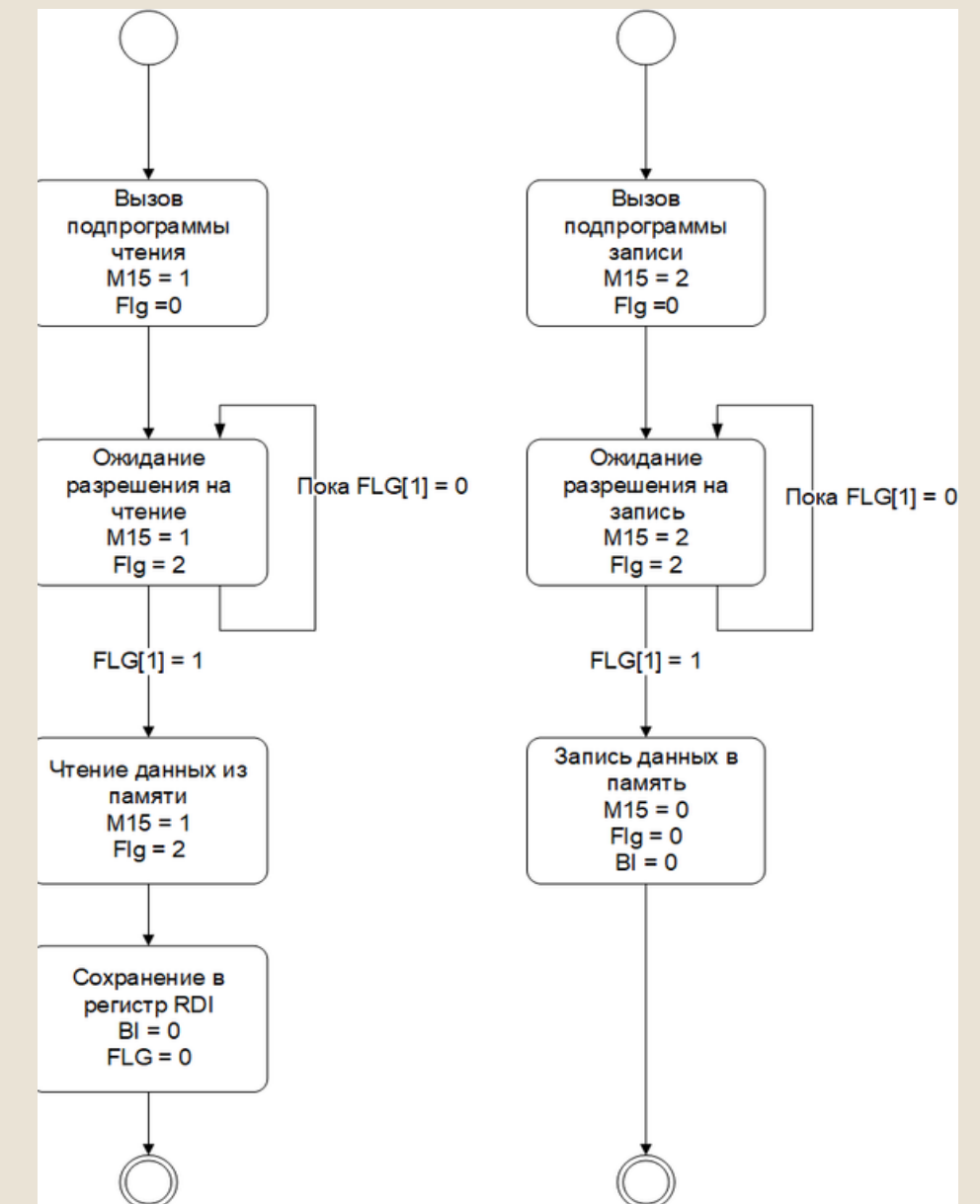
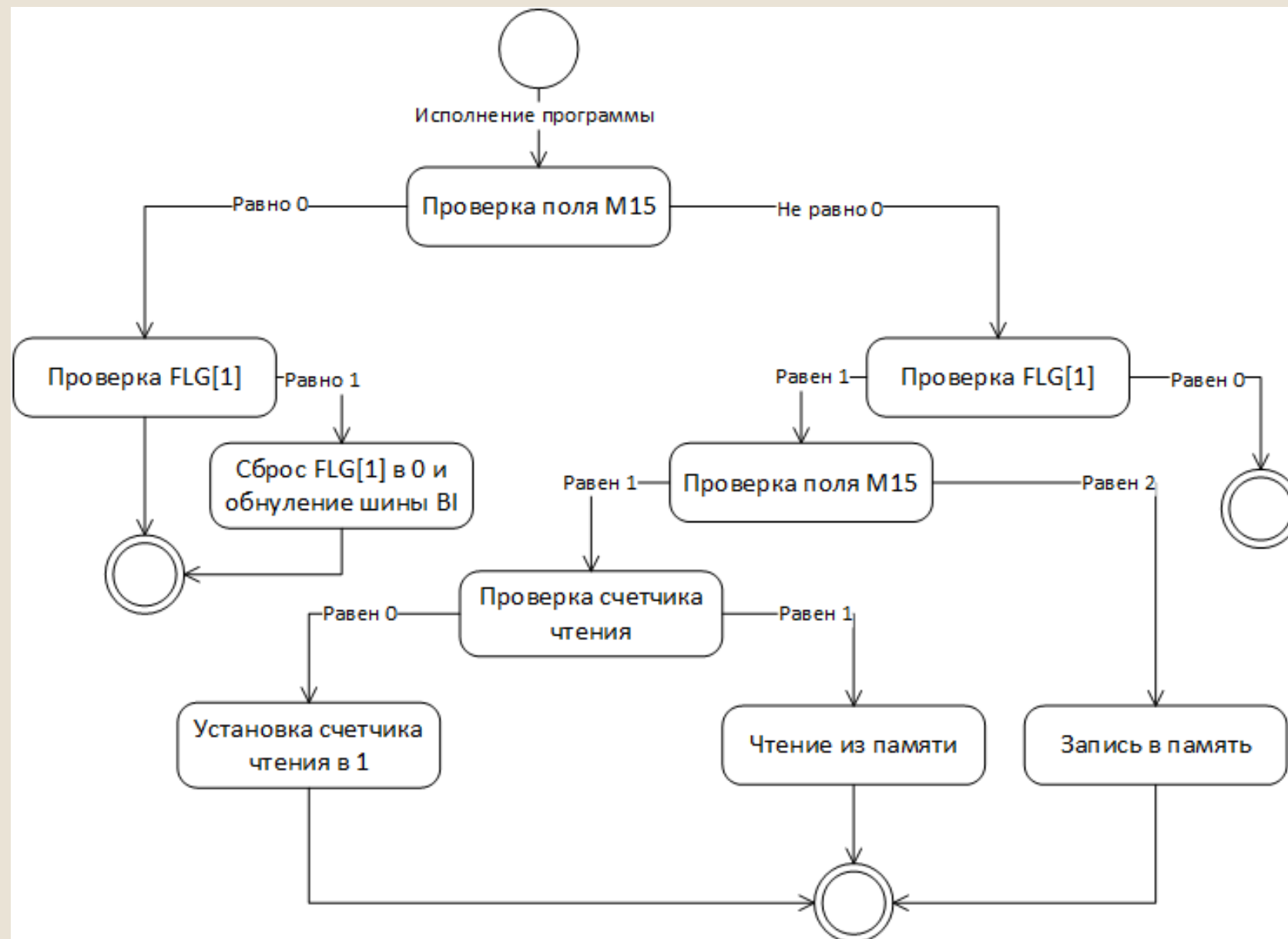
# Алгоритм работы с памятью

- Алгоритм чтения

Формирование сигнала запроса чтения для ОЗУ. Ожидание сигнала от ОЗУ, разрешающего чтение данных с шины BI. Чтение данных с шины в регистр RDI (Register Data Input)

- Алгоритм записи

Формирование сигнала запроса записи для ОЗУ. Ожидание сигнала от ОЗУ, разрешающего запись данных из регистра RDO (Register Data Output).



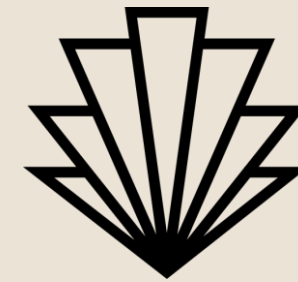
# Характеристики программы



Windows 7/8/10  
Linux Альт  
Образование/  
Ubuntu/Astra



JRE версия 1.8.0  
и выше



ОЗУ не менее  
256 Мбайт



Дисковая  
память не менее  
5 Мбайт

# Перспективы развития проекта

1

---

Добавление  
процессоров с  
аналогичной  
архитектурой

2

---

Интеграция  
эмулятора в  
образовательный  
процесс

3

---

Оптимизация  
программного кода



**Спасибо  
за внимание!**