

**Московский физико-технический институт (государственный университет)  
Физтех-школа радиотехники и компьютерных технологий  
Кафедра информатики и вычислительной техники**

**Разработка системных тестов видеоконтроллера MGA2.6  
для верификации микропроцессора R2000+**

**Выпускная квалификационная работа  
(бакалаврская работа)**

**студент: Ситников П. В.  
научный руководитель: Фролов П. В.**

**Москва, 2021**

# Введение

MGA2.6 — графическая подсистема микропроцессора R2000+.

MGA2.6 включает: модули блиттеров и подмодуль управления блиттерами.

Блиттер — устройство, задача которого состоит в быстром перемещении прямоугольных областей памяти с одновременным выполнением какого-либо вида обработки перемещаемого изображения.

AUC2 — подмодуль управления блиттерами: на основе программно-заданных очередей дескрипторов запускает исполнение задач на блиттерах.

# Цель работы

Разработка тестовых программ для функциональной верификации модулей блиттера и подмодуля автоматического управления блиттерами

# Требования

- 1) Разработка в кроссплатформенном окружении `periph_env`
- 2) Минимизация накладных расходов при запуске тестов на RTL-модели
- 3) Рандомизация тестовых данных

# Задачи

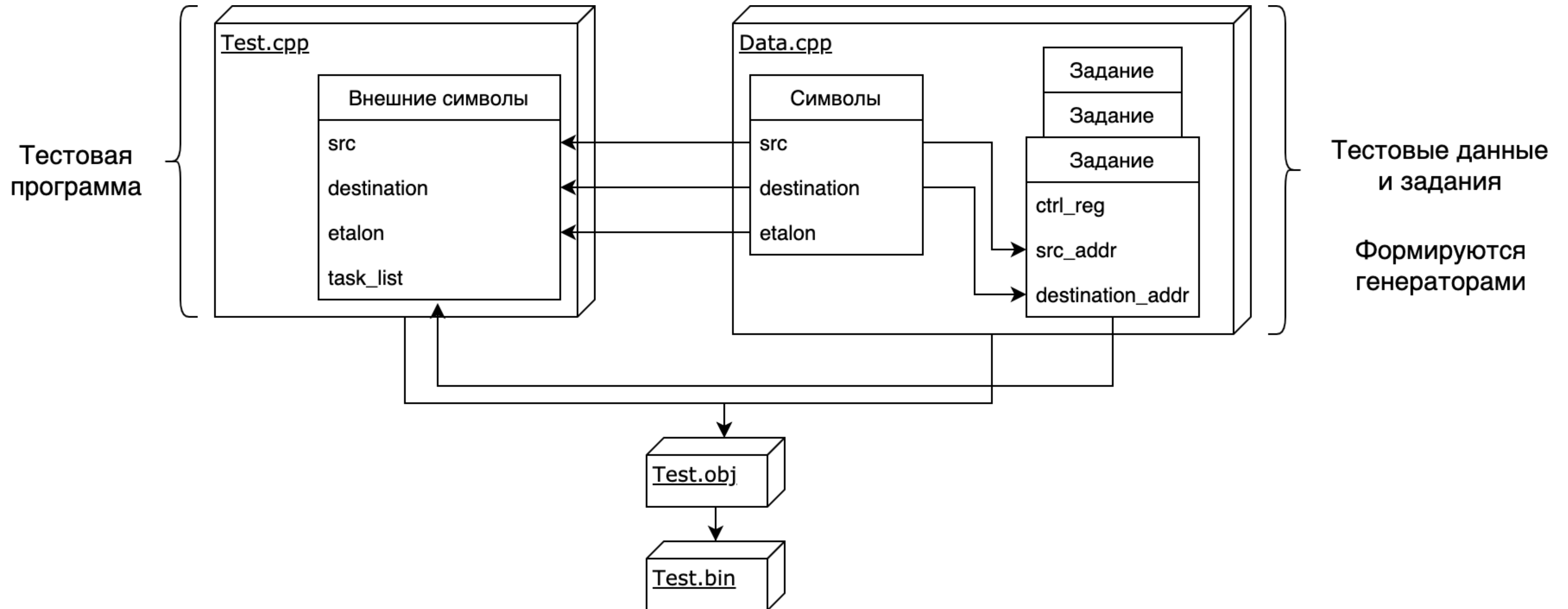
- Определить структуру теста
- Разработать генераторы:
  - исходных данных для преобразования модулями блиттера и эталонного результата
  - заданий для AUC2
  - последовательности заданий для AUC2

# Структура теста

Тестовая программа выполняется на вычислительном ядре R2000+ архитектуры SPARC v9. Программа реализует единый алгоритм для всех исходных данных и заданий:

- 1) Конфигурация AUC2 на основе исходного задания
- 3) Программирование и запуск локального таймера
- 4) Старт AUC2
- 5) Ожидание прерывания
- 6) Проверка на таймаут
- 7) Сравнение считанных данных с эталонными

# Структура теста



Сборка и компиляция теста производятся в среде `periph_env`. Эталонные данные формируются генератором и не требуют дополнительных вычислений в тесте, что ускоряет исполнение на RTL модели.

# Генератор данных

Разработан генератор на языке python.

Входные аргументы: blitter-mode, seed, src-width, src-height, color-depth.

**Поддержанные в генераторе режимы блиттера:**

BitBlt — копирование данных из области источника в область приемника.

src[width x height] → dest[width x height]

ColorExpansion — расширение цветом. Источник представляется как битовый поток. Нули в потоке расширяются цветом из BG\_REG. Единицы цветом из FG\_REG.

src[width x height] → dest[width x height x color\_depth x 8]

PatternCopy — размножение образа.

Образ источника копируется width x height раз в область приемника.

src[8 x 8 x color\_depth] → dest[width x height x color\_depth x 8]

Генератор формирует массивы данных источника, приемника и эталонных значений преобразований для выбранного режима в отдельном файле.

# Генератор заданий

Задание блиттера: запись типа 1. Содержит указатели на исходные данные и на массив с результатом.

Отправка прерывания: запись типа 0 (установка прерывания в регистр INTREQ).

## Запись тип 0

Биты	Описание
63	0 - последняя запись, 1 - последняя запись
51, 49	Ожидать полный останов блиттера #0, #1
47...32	Номер регистра в адресном пространстве MGA2.6
31...0	Записываемое в регистр значение

## Запись тип 1

Биты	Описание
63	0 - последняя запись, 1 - последняя запись
55...54	Выполнять операцию в указанном блиттере
52	Ожидать останова указанного блиттера
47...32	Маска используемых регистров
.	Значения используемых регистров (CTRL_REG, WINDOW_REG, PITCH_REG, SRC_ADDR, DEST_ADDR, BG_REG, FG_REG)

Разработан генератор заданий. Задания формируются в отдельном файле с тестовыми данными.



# Генератор последовательности заданий

Последовательность заданий — очередь записей в памяти теста.

Задания блиттера выполняются последовательно друг за другом на блиттерах.

Между заданиями формируется зависимость по данным для проверки механизма упорядочивания задач в модуле AUC2.



Разработан генератор последовательности заданий. Формируется массив заданий AUC2 в файле с тестовыми данными.

# Результаты

- Определена структура теста
- Разработан генератор эталонных данных
- Разработан генератор заданий для AUC2
- Разработан генератор последовательности заданий для AUC2