Федеральное государственное автономное образовательное учреждение  
высшего образования   
"Национальный исследовательский университет   
"Высшая школа экономики"

Московский институт электроники и математики им. А. Н. Тихонова

Департамент компьютерной инженерии

Отчёт

о выполнении практической работы № 1

Тема работы: «Разработка и программирование Soft-процессорных ядер с архитектурой однотактный MIPS. Часть 1»

по курсу «Системное проектирование цифровых устройств»

Выполнил:

Бригада № 10

Власов Р. В. БИВ174

Сегида Т. О. БИВ174

Принял

ассистент Американов А. А.

Оценка:

Москва 2020 г.

Оглавление

[1. Счетчик 3](#_Toc50994893)

[1.1. Выполнить компиляцию примера 3](#_Toc50994894)

[1.2. Выполнить дизассемблирование 4](#_Toc50994895)

[1.3. Провести моделирование программы 5](#_Toc50994896)

[1.4. Собрать проект на одной из плат 6](#_Toc50994897)

[2. Числа Фибоначчи 6](#_Toc50994898)

[3. Квадратный корень 6](#_Toc50994899)

[4. Самостоятельная работа 6](#_Toc50994900)

[5. Контрольные вопросы 6](#_Toc50994901)

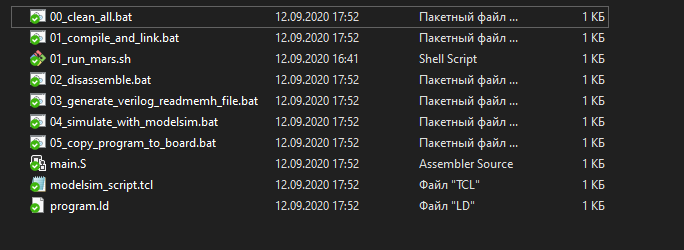
[6. Выводы 6](#_Toc50994902)

[7. Список литературы 7](#_Toc50994903)

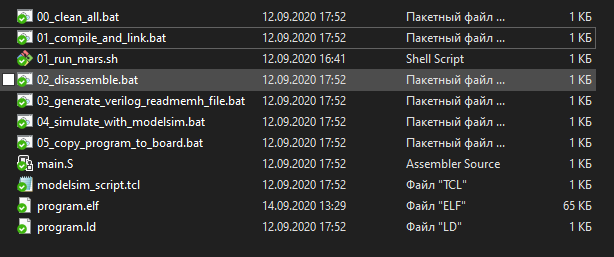
1. Счетчик

Задание:

1. выполнить компиляцию примера
2. выполнить дизассемблирование
3. провести моделирование программы
4. собрать проект на одной из плат
   1. Выполнить компиляцию примера



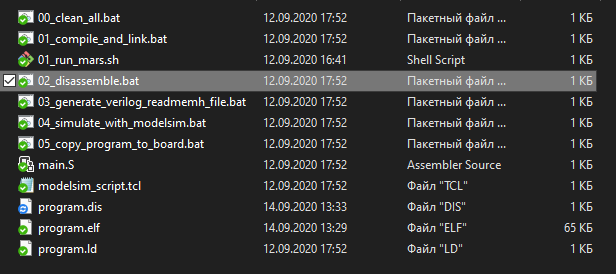
1. Файлы в папке примера до компиляции.



1. Файлы в папке примера после компиляции.

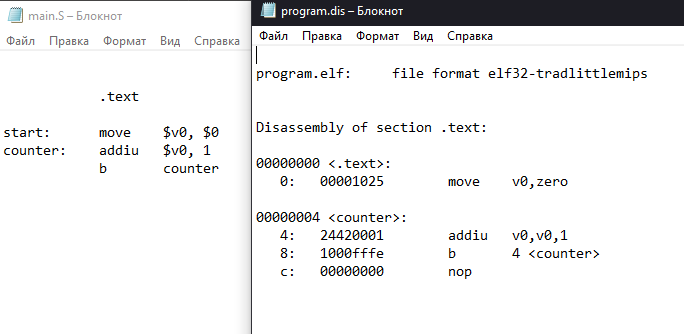
Добавился файл с расширением .elf – скомпилированный бинарник.

* 1. Выполнить дизассемблирование



1. Файлы в папке программы после дизассемблирования.

Добавился файл с расширением .dis – дизассемблированный файл, аналогичен исходнику main.S.



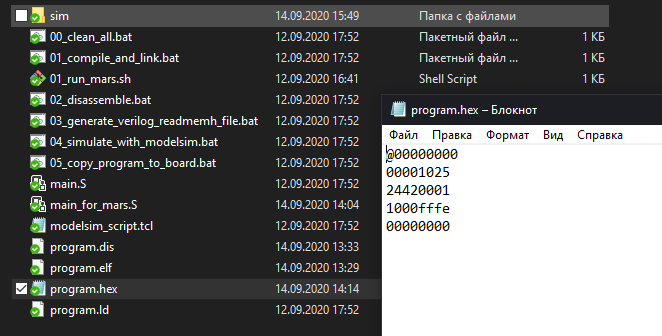
1. Сравнение файлов main.S и program.dis.

**ADDIU --** Add immediate unsigned (no overflow)

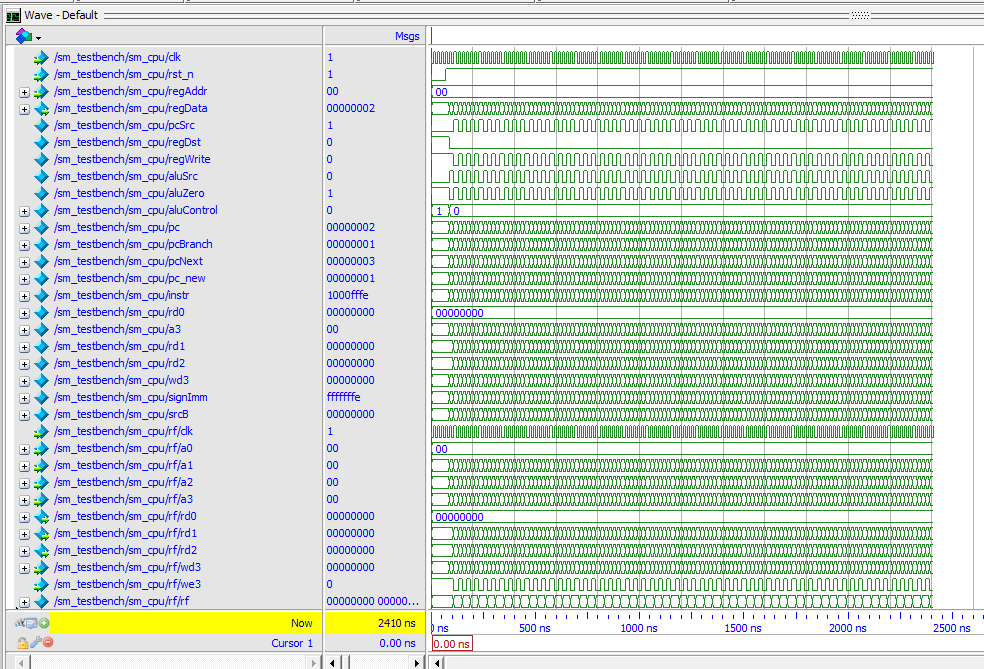
|  |  |
| --- | --- |
| Description: | Adds a register and a sign-extended immediate value and stores the result in a register |
| Operation: | $t = $s + imm; advance\_pc (4); |
| Syntax: | addiu $t, $s, imm |
| Encoding: | 0010 01ss ssst tttt iiii iiii iiii iiii |

* 1. Провести моделирование программы

Чтобы провести моделирование, сначала надо запустить скрипт 03\_generate\_verilog\_readmemh\_file.bat, который генерирует .hex файл необходимый ModelSim для симуляции.



1. .hex – файл.



1. Симуляция в ModelSim.

Из рисунка 4 видно, что программа обнуляет значение регистра v0 (регистр № 2), далее прибавляет единицу и возвращает выполнение программы безусловным переходом на предыдущую команду, создавая зацикленный счетчик.

* 1. Собрать проект на одной из плат

Бригада 10.

Основных плат в списке в [пункте 2](https://docs.google.com/document/d/1kJUwXja6cCaOC1bWOS4TUXvtFEZyd_Eth43VvBZxias/edit#heading=h.8nko2kqz0xp6) методички 8.

10 mod 8 + 1 = 3

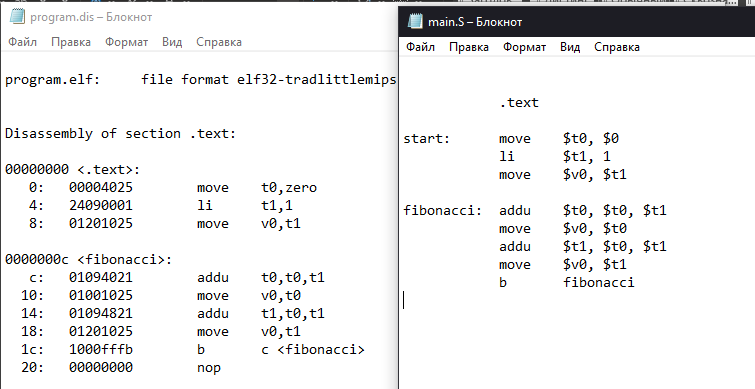
=> наша плата De10-Standard

На удаленном стенде все работало, но на видео, и тем более, на фото, этого не было бы видно из-за низкого качества. На веб-камере с удаленного стенда можно было различить лишь мерцания.

Для проверки работоспособности на удаленном стенде в файле de10\_standard.v заменили строку с назначением свичей, как устройства ввода номера регистра, на номер регистра (2 == 00010).

Порядок действий для загрузки и проверки на плате:

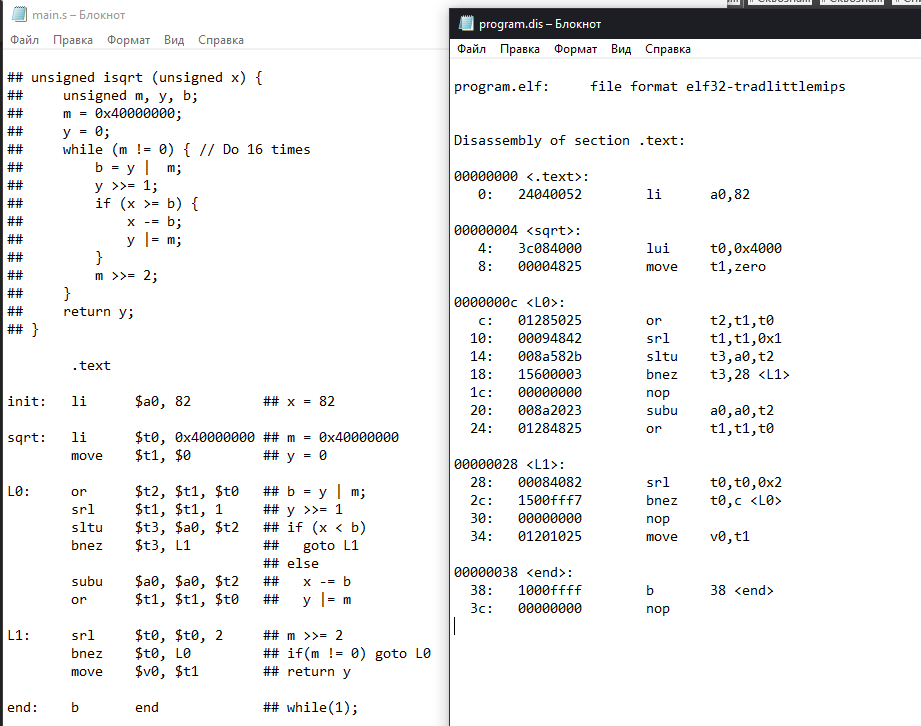
1. Прошить плату
2. Набрать на свичах второй регистр (SW[4]:SW[0] – 00010)
3. Убедиться, что на 7-сегментном дисплее увеличивается значение
4. Числа Фибоначчи



1. Сравнение файлов main.S и program.dis.

Остальные этапы не документируются, так как они в точности повторяют этапы из предыдущего пункта.

1. Квадратный корень



1. Сравнение файлов main.S и program.dis.

Остальные этапы не документируются, так как они в точности повторяют этапы из предыдущего пункта.

1. Самостоятельная работа

Задание № 1:

1. Добавить процессору команду jr
2. Добавить процессору команду xori
3. Добавить процессору команду srl
4. Добавить процессору команду xor

Не выполнено.

Задание № 2:

1. Разработать программу для демонстрации работоспособности команд (в соответствии с вариантом 5)
2. Добавить программу в проект микропроцессора (также, как и 3 программы-примера, как счетчик, числа Фибоначчи и вычисление квадратного корня)
3. Добавить в папку с программой файл описания (лучше сделать описание как в примере вычисления квадратного корня, рисунок 8)

Программа по варианту – найти сумму арифметической прогрессии, где количество членов прогрессии == 19, а разность арифметической прогрессии == 1.

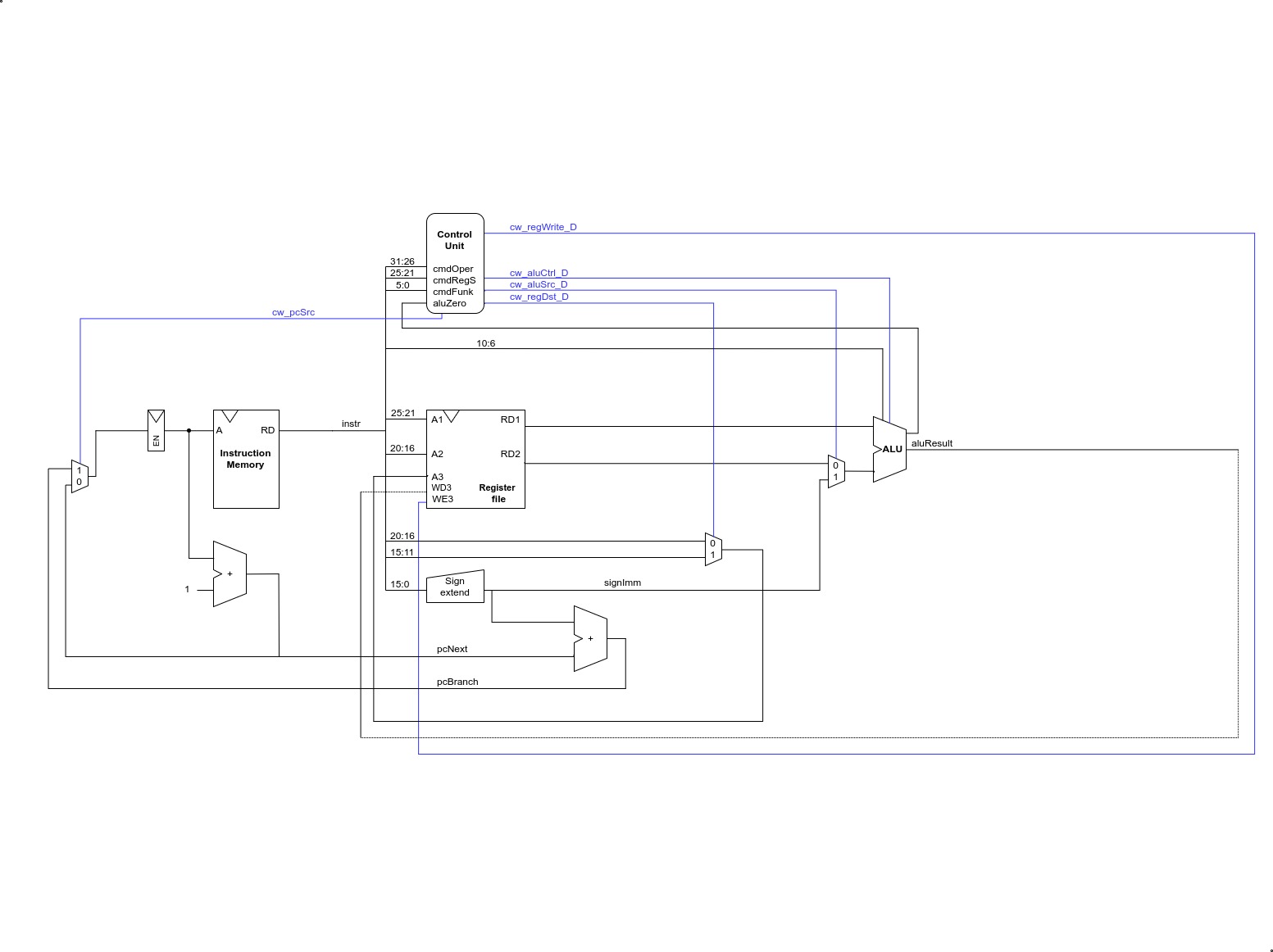
Не выполнено.

Задание № 3:

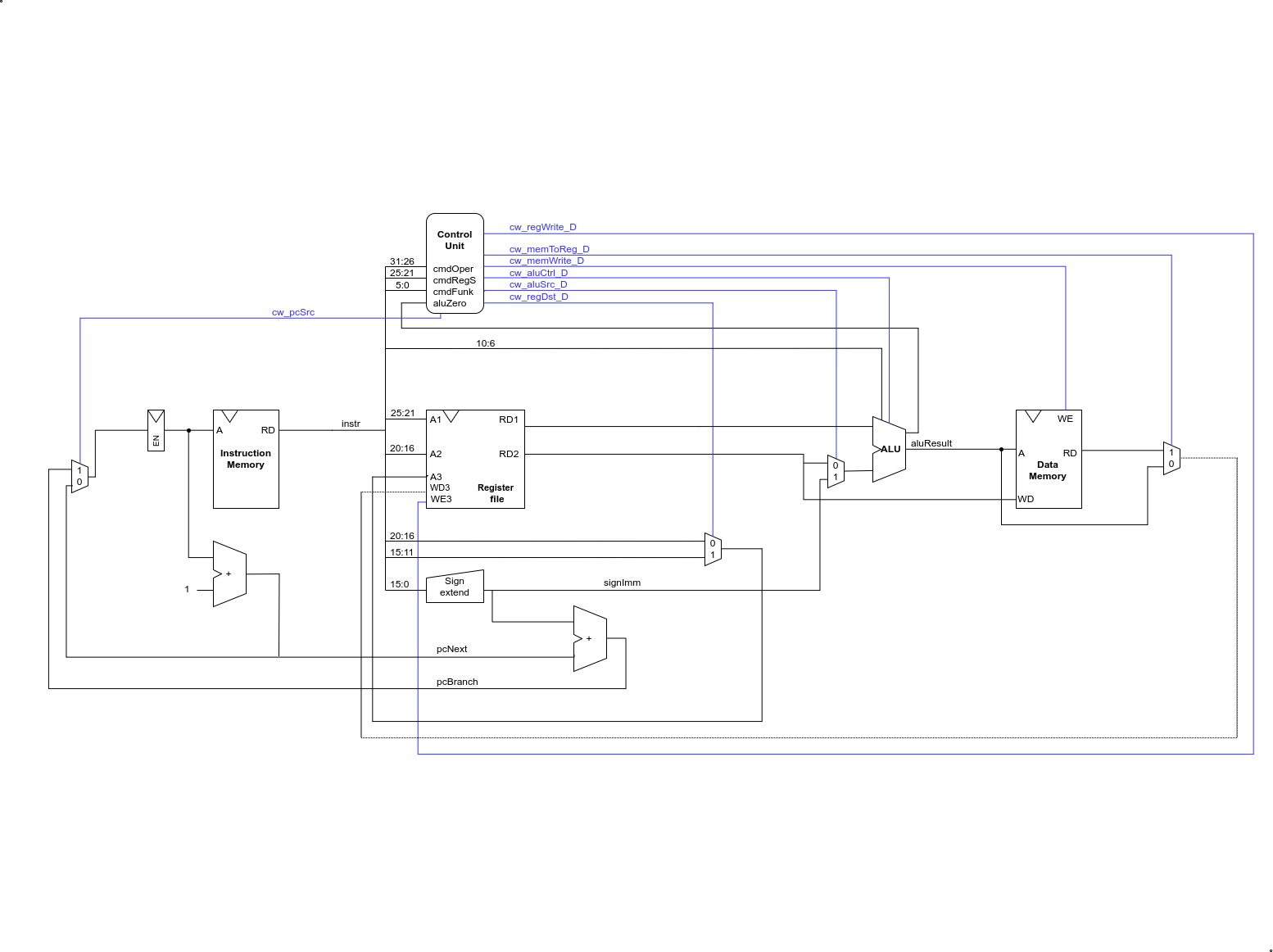
1. Скачать версию процессора 01\_mmio: <https://github.com/MIPSfpga/schoolMIPS/tree/01_mmio/>
2. Выполнить программы 00\_counter, 01\_fibonacci, 02\_sqrt и убедиться, что они работают также, как и на предыдущей версии
3. Выполнить программу 05\_pwm (в соответствии с вариантом 10 % 4 + 1)
4. Добавить комментарии к программе
5. Объяснить различия этой версии и базовой

Убедиться, что программы работают одинаково на различных версиях процессора можно на учебной плате в лаборатории, на удаленной плате ничего видно не будет.

Различия – the same but with data memory, simple system bus and peripherals (pwm, gpio, als)



1. Схема процессора версии 00\_simple.



1. Схема процессора версии 01\_mmio.  
   (system bus and peripheral devices are not shown)
2. Выводы

Во время выполнения данной работы мы познакомились с процессором schoolMIPS MIPS архитектуры. Сравнили две версии данного процессора. Получили навыки расширения функциональности процессора добавлением команд. Получили навыки работы с платой DE10-Standard. Вспомнили процессы дизассемблирования и моделирования.

1. Список литературы
2. SPDS\_Lab\_1\_2020 [Электронный ресурс]. – URL: [https://doc.google.com/document/d/1kJUwXja6cCaOC1bWOS4TUXvtFEZyd\_Eth43VvBZxias](https://docs.google.com/document/d/1kJUwXja6cCaOC1bWOS4TUXvtFEZyd_Eth43VvBZxias)
3. MIPSfpga/schoolMIPS Wiki [Электронный ресурс]. – URL: <https://github.com/MIPSfpga/schoolMIPS/wiki>