



Санкт-Петербургский  
государственный  
университет  
[www.spbu.ru](http://www.spbu.ru)

# СОЗДАНИЕ РЕГУЛЯТОРА СОСТОЯНИЯ ПРОСТОЯ ДЛЯ СИСТЕМЫ EAS

Студент: Пелогейко Макар Андреевич группа 19.Б11-мм

Научный руководитель: ст. преп. Сартасов С. Ю.

Санкт-Петербургский государственный университет

Кафедра системного программирования

2022г.

# Введение

- Смартфоны получили широкое распространение  
Требования и круг решаемых задач постоянно растут
- Одно из важнейших требований: время автономной работы
- Процессор потребляет энергию и не всегда используется
- Создание EAS для гетерогенных процессоров

# Цели и Задачи

## **Цель:**

Создание регулятора состояния простоя для системы EAS.

## **Задачи:**

1. Сделать обзор подсистемы CPU Idle.
2. Сделать обзор работы EAS.
3. Сделать обзор регулятора schedutil.
4. Разработать регулятор и встроить его.
5. Разработать методологию тестирования и подготовить окружение.
6. Провести тестирование.

## **CPU Idle состоит из 3 частей:**

- регуляторы (governors) выбор состояния простоя
- драйверы (drivers) передаёт решение регуляторов в аппаратную часть смартфона
- ядро (core) общая платформа, связывающая систему воедино

# Energy Aware Scheduling

- энергия = [джоуль] (ресурс, такой как батарея)
- мощность =  $\frac{\text{энергия}}{\text{время}}$  = [джоуль/секунда] = [ватт]

## Цель EAS

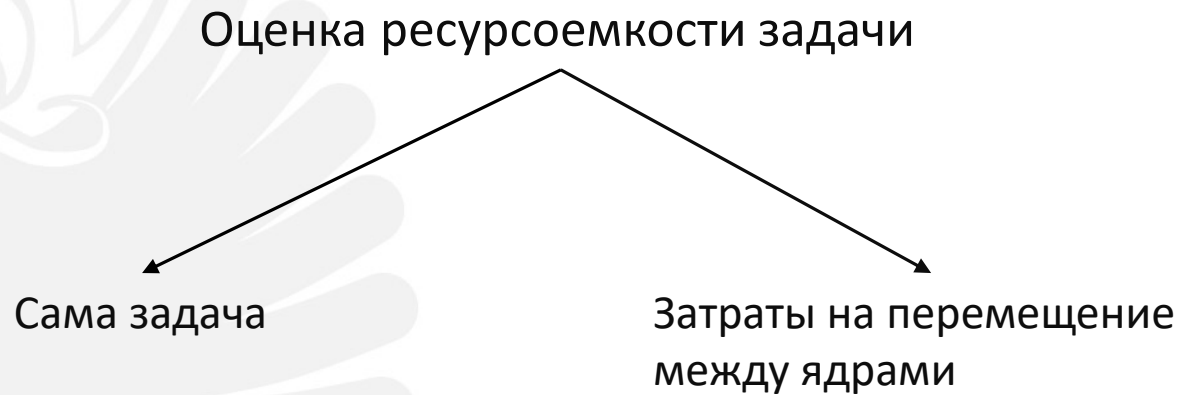
- Максимизировать:

$$\frac{\text{производительность[инструкция/сек]}}{\text{мощность[ватт]}}$$

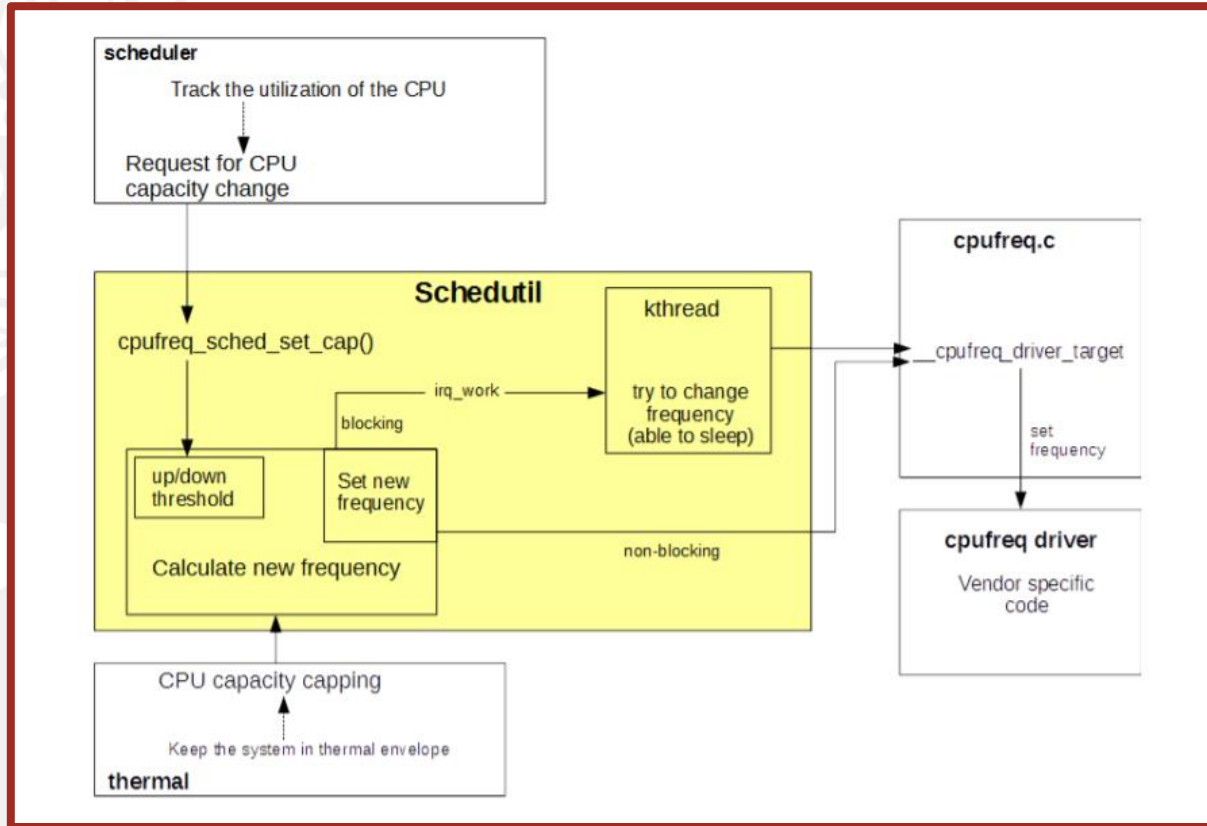
- Минимизировать:  $\frac{\text{энергия[Дж]}}{\text{инструкция}}$

# Energy Aware Scheduling

capacity — [0..1024] нормализация частоты



# Schedutil



[eas overview and integration guide r1p6](#)

# Тестовый стенд

- Samsung Galaxy s7 SM-G930FD
- Lineage-17.1-20210409 (android 10)
- Проект lazyflasher ветки no-verity-opt-encrypt
- Open source ядро на основе elite\_kernel с реализованной системой EAS



# Тестовый стенд

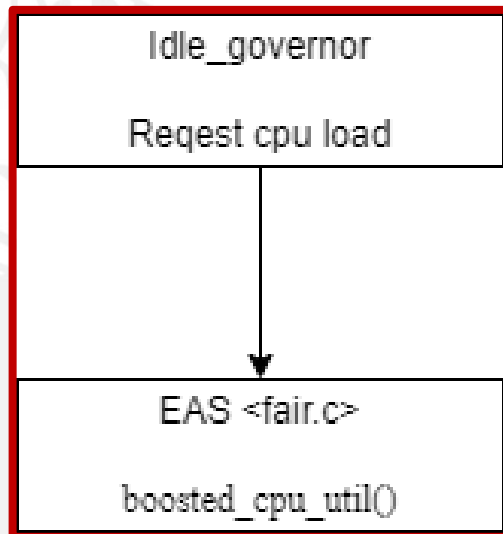
## CPU [0..3]

C-state 0, name: WFI,	desc: ARM WFI,	exit_latency: 1,	power_usage: -1, disabled: 0
C-state 1, name: cpu-sleep-boot,	desc: cpu-sleep-boot,	exit_latency: 125,	power_usage: 0, disabled: 0
C-state 2, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0
C-state 3, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0
C-state 4, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0
C-state 5, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0
C-state 6, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0
C-state 7, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0
C-state 8, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0
C-state 9, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0

## CPU [4..7]

C-state 0, name: WFI,	desc: ARM WFI,	exit_latency: 1,	power_usage: -1, disabled: 0
C-state 1, name: cpu-sleep-nonbo,	desc: cpu-sleep-nonboot,	exit_latency: 105,	power_usage: 0, disabled: 0
C-state 2, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0
.....			
C-state 9, name: ,	desc: ,	exit_latency: 0,	power_usage: 0, disabled: 0

# Модификация



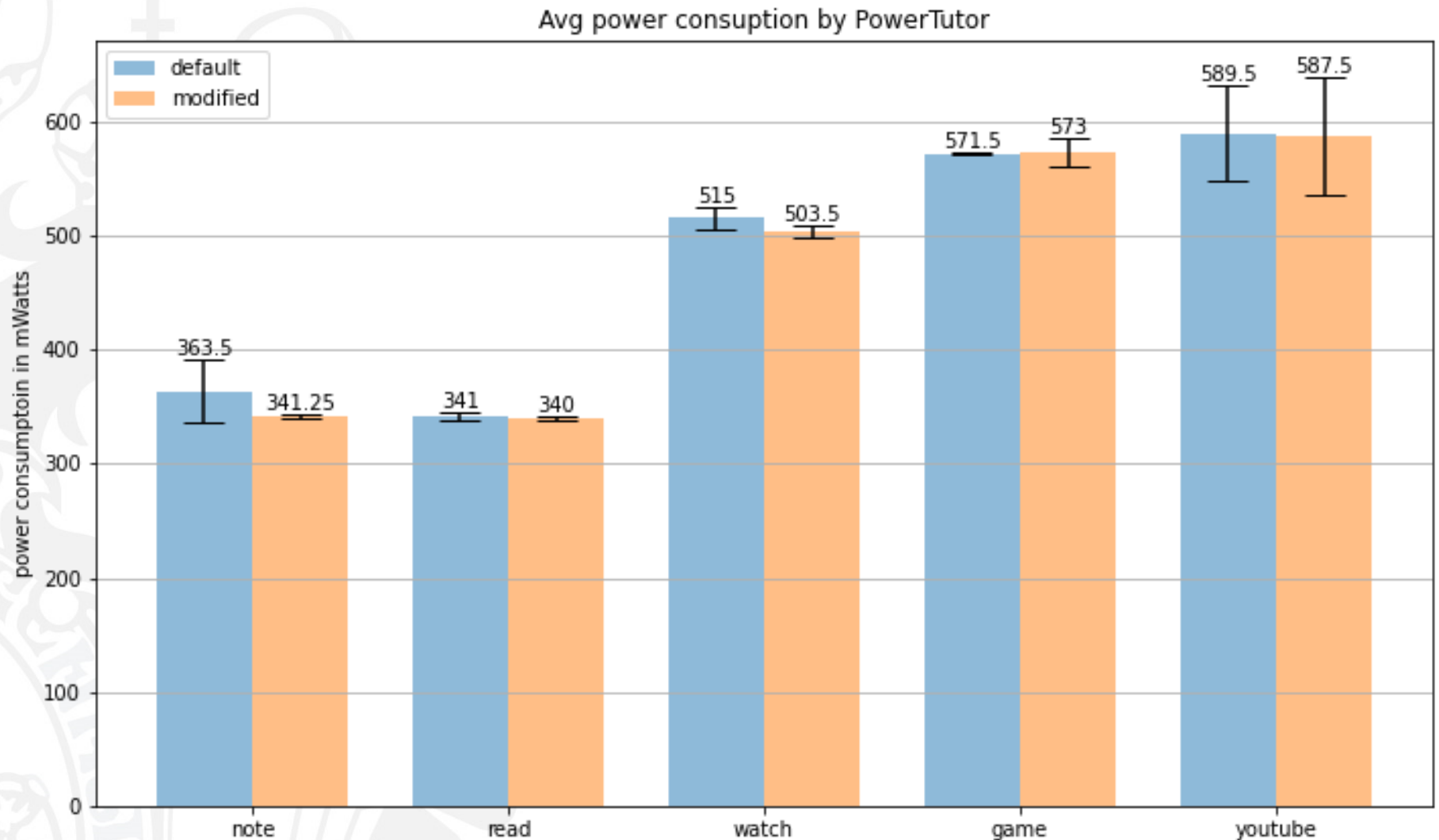
3 класса энергопотребления:

- 0. Little — 20%
- 1. Middle — 40 %
- 2. Big — 60%

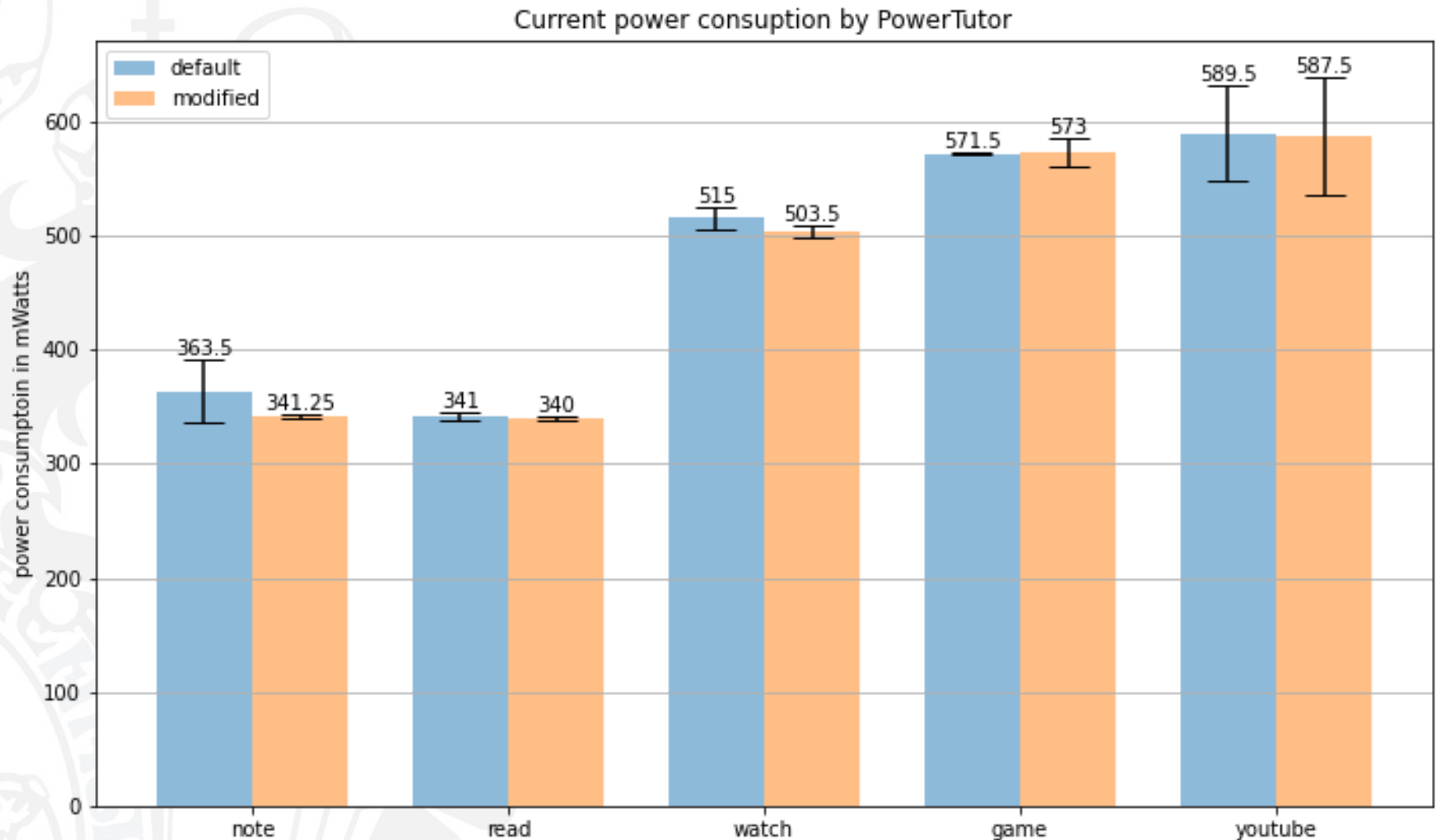
## Для каждого теста собраны следующие данные

- Значение времени и количества использований каждого состояния простоя для каждого ядра.
- Для каждого ядра количество времени проведенное данным ядром на определенной частоте
- Данные о энергопотреблении из PowerTurot
- Дамп batterystats
- Время длительности теста

# Анализ результатов

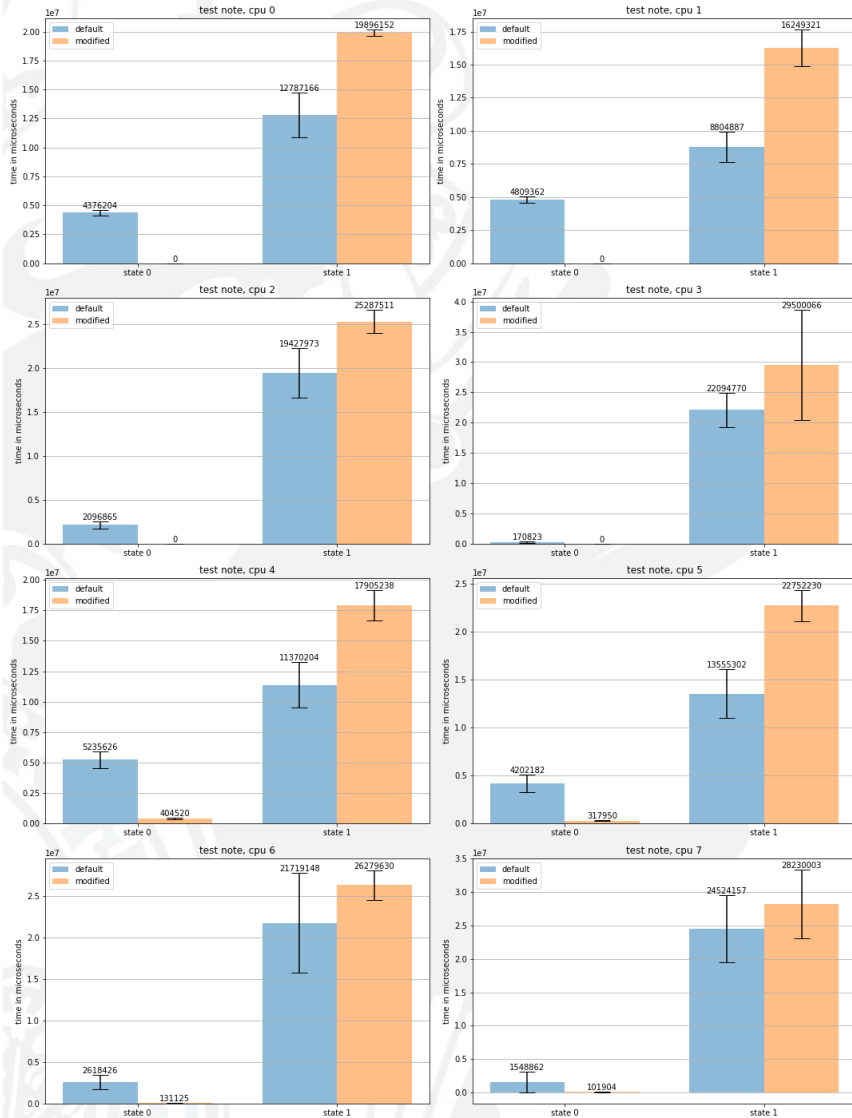


# Анализ результатов

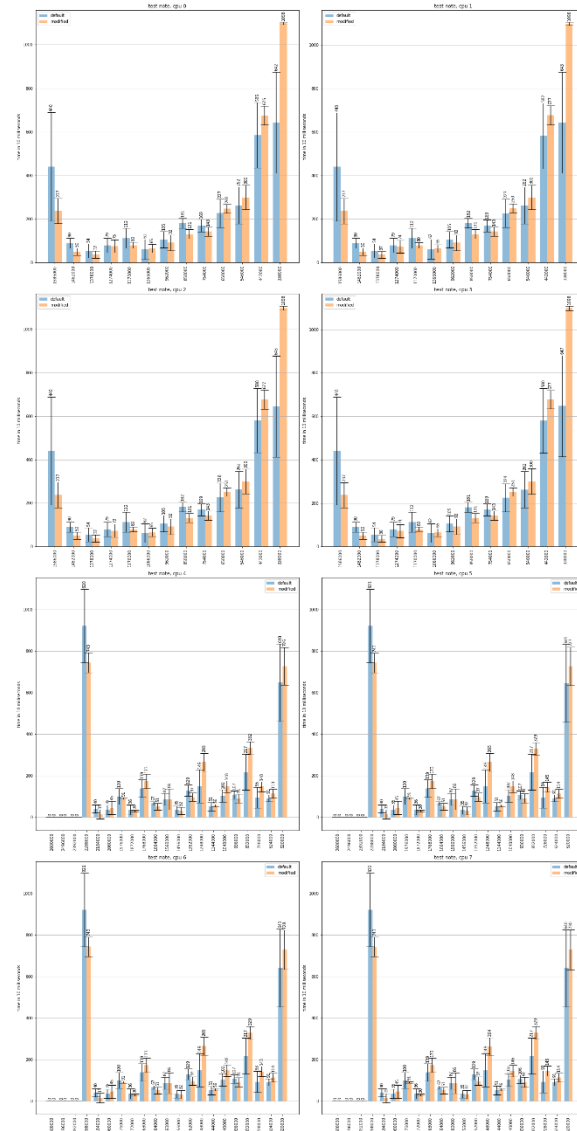


# Анализ результатов Note

Notes Idle test

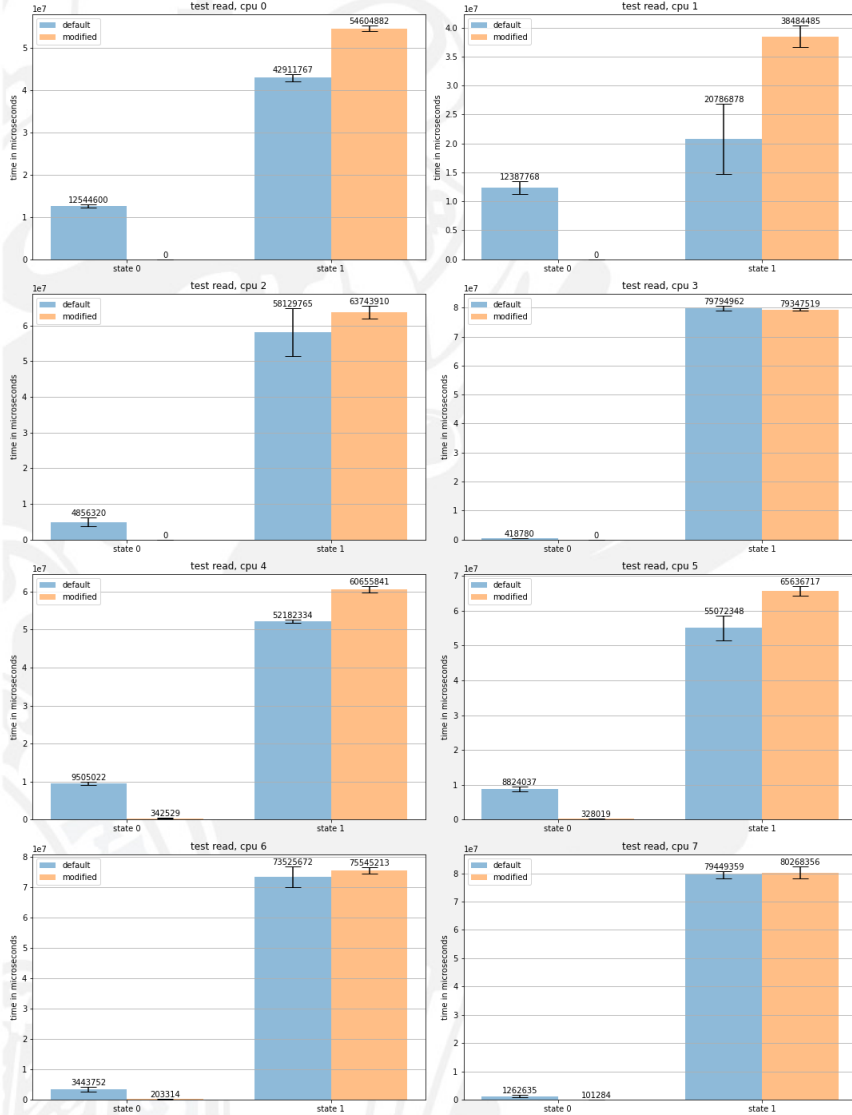


Note free test

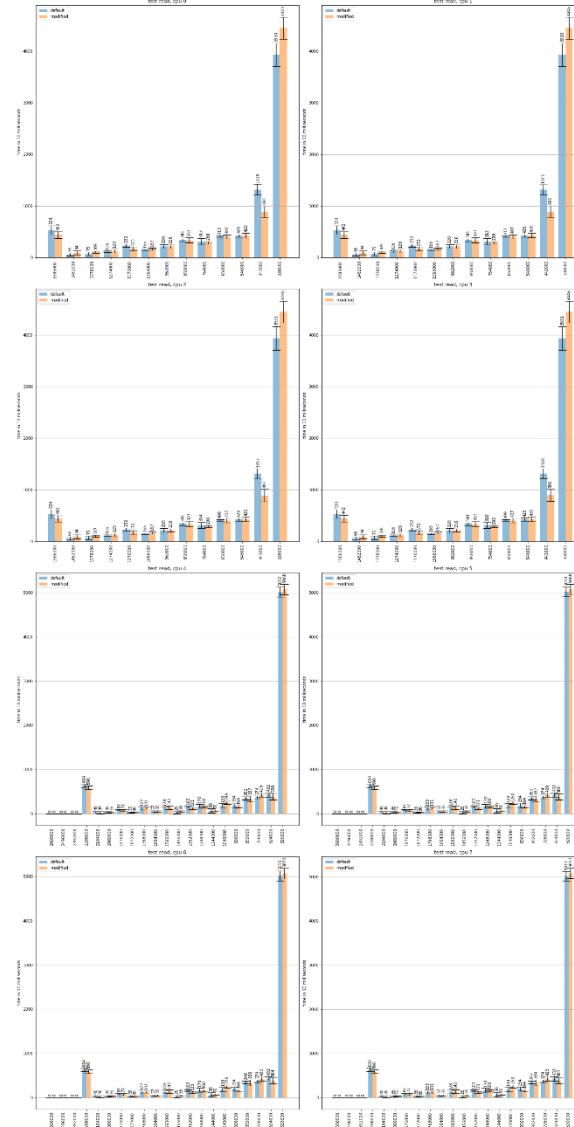


# Анализ результатов Read

Read Idle test

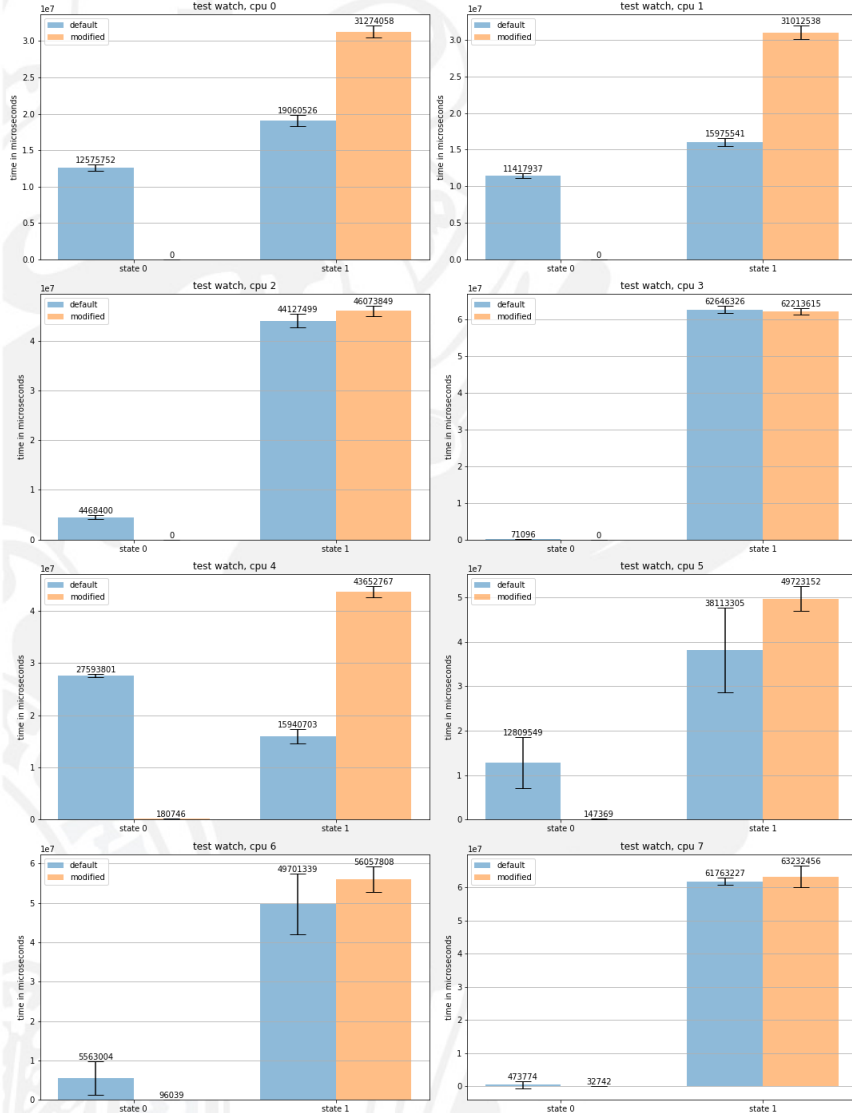


Read Ping test

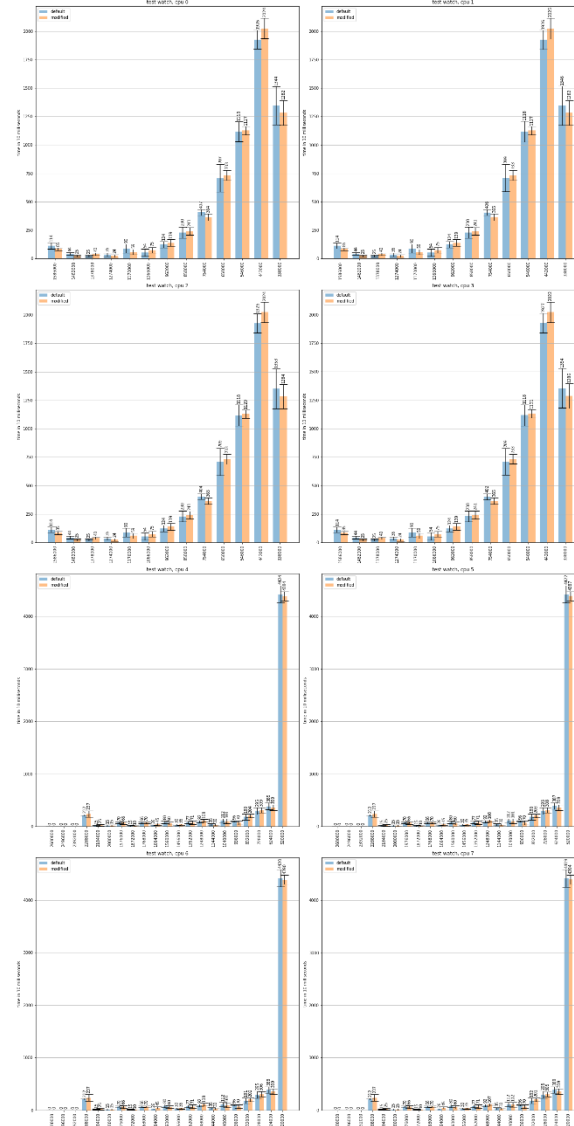


# Анализ результатов Video

Video Idle test



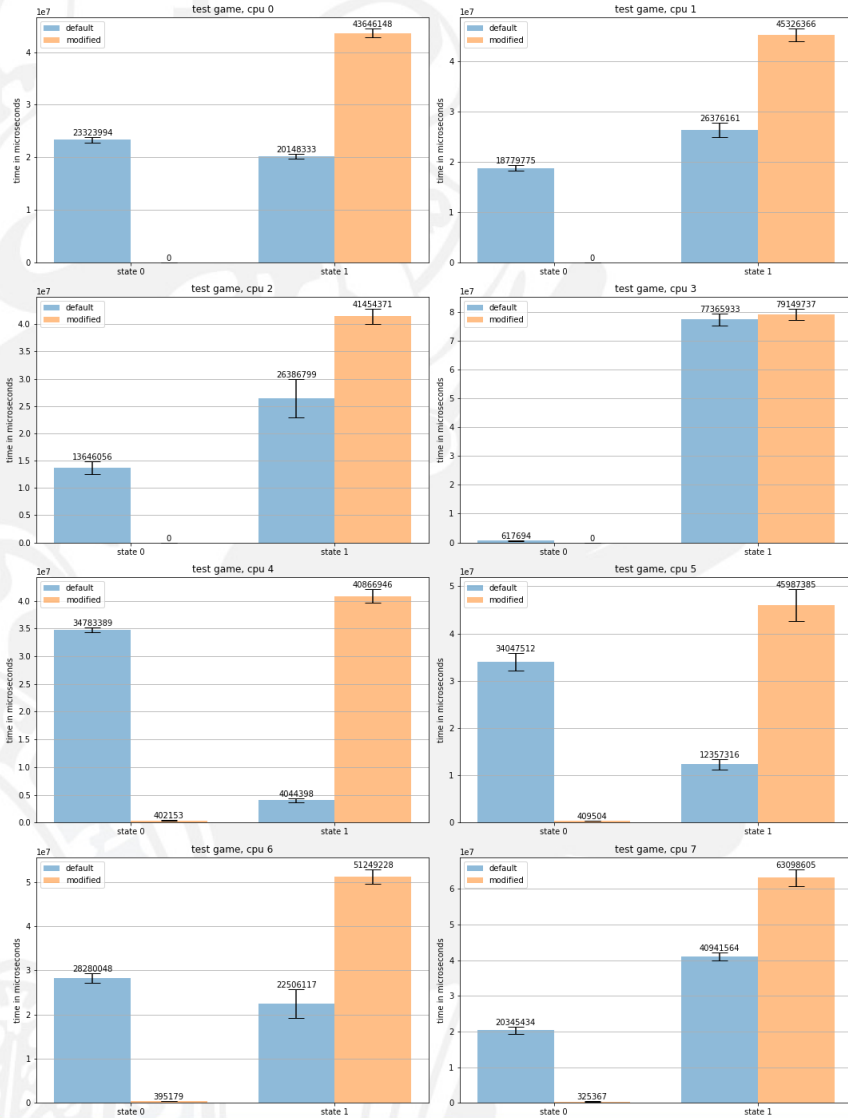
Video Prog test



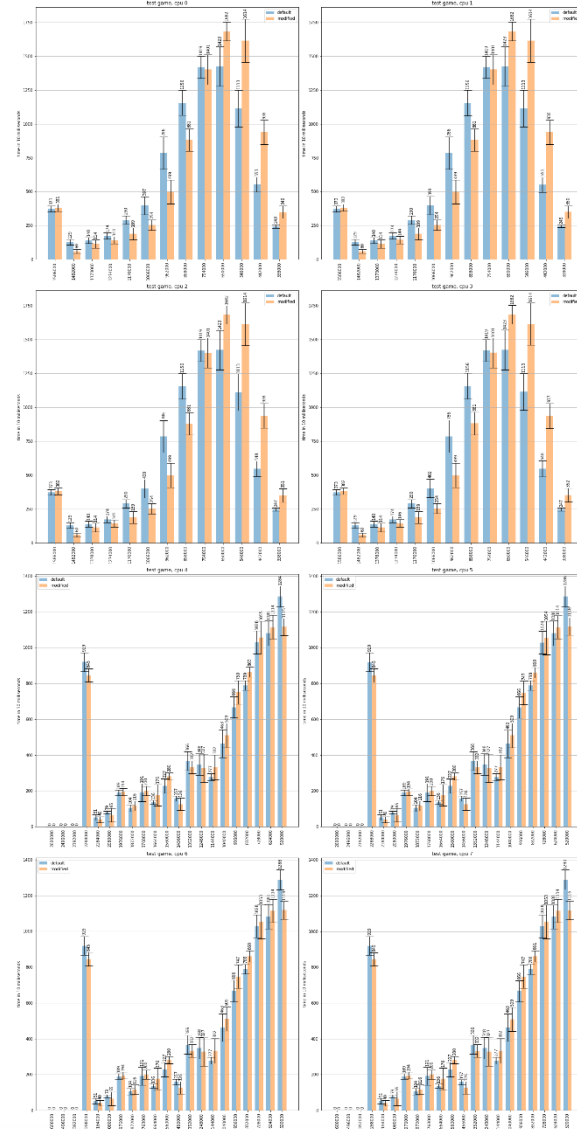


# Анализ результатов Game

Game Idle test

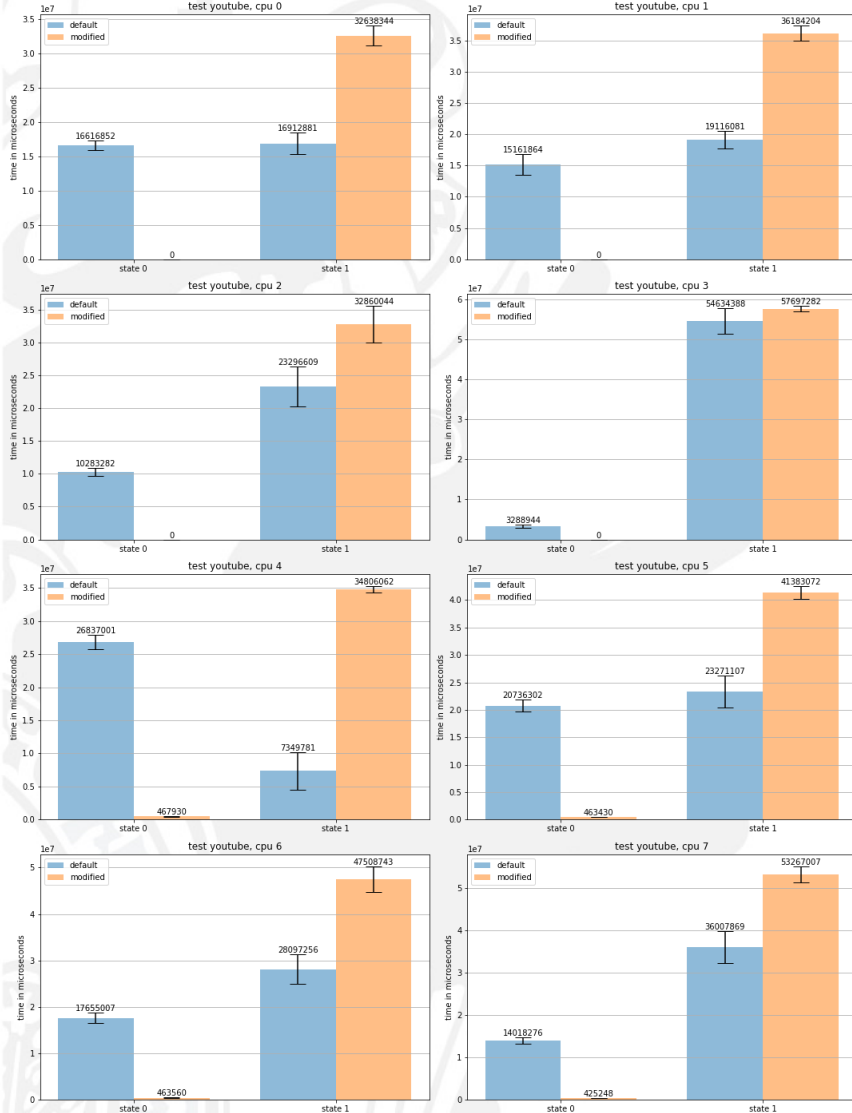


Game Trq test

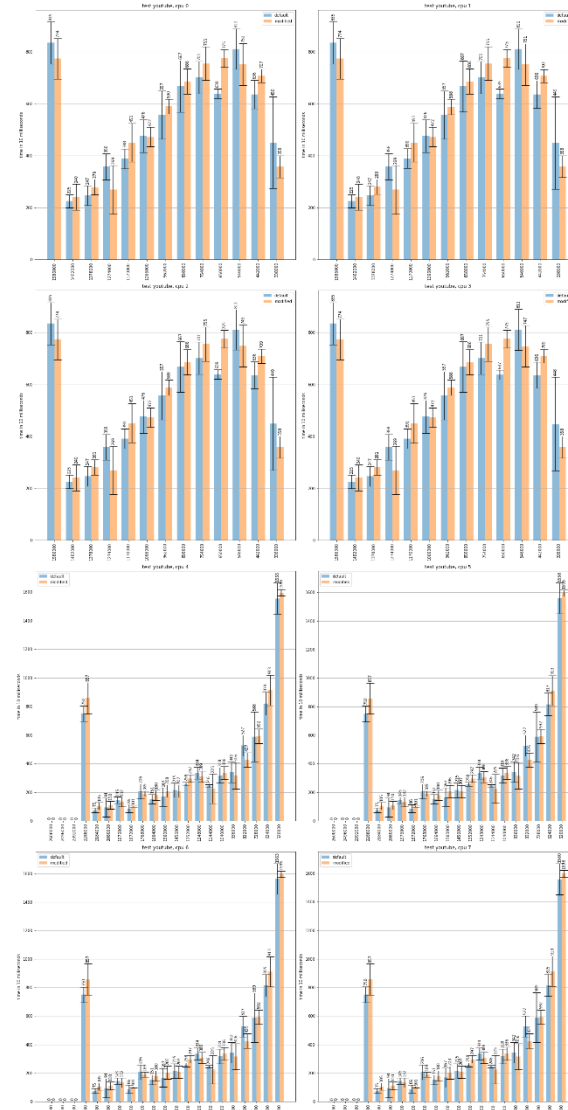


# Анализ результатов Youtube

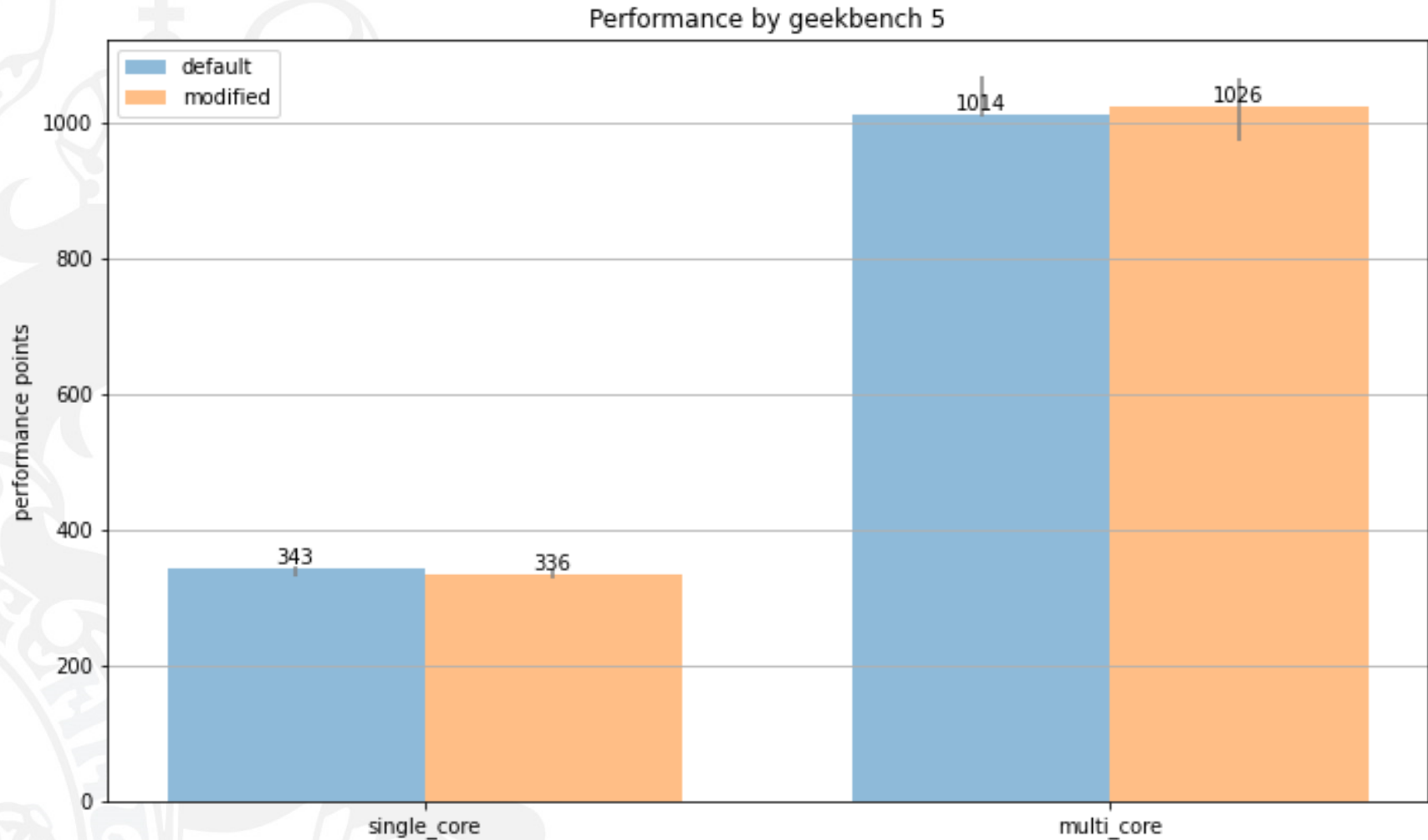
Youtube Idle test



Youtube Ping test



# Сравнение производительности



# Полученные результаты

1. Обзор подсистемы CPU Idle
2. Обзор работы EAS
3. Обзор регулятора schedutil
4. Подготовлен тестовый стенд
5. Разработан пробный алгоритм регулятора простоя
6. Созданы скрипты тестов и сбора данных
7. Проведен анализ результатов