

ё.

Отчёт по лабораторной работе № 3 “Кэш”

Хромов Адам М3139

Вариант 2

Инструментарий:

Компилятор: gcc (GCC) 12.2.1

Репозиторий: <https://github.com/skkv-itmo2/itmo-comp-arch-2023-cache-Ad-mex>

Язык C11

Результат:

LRU: hit perc. 99.5244% time: 3186131

pLRU: hit perc. 99.4876% time: 3200031

Константы

MEM_SIZE	1 Мбайт
ADDR_LEN	$\log_2(\text{MEM_SIZE}) = 20$
CACHE_SIZE	4 Кб
CACHE_IDX_LEN	4
CACHE_OFFSET_LEN	6
CACHE_TAG_LEN	$\text{TAG_LEN} = (\text{ADDR_LEN} - \text{IDX_LEN} - \text{OFFSET_LEN}) = 10$
CACHE_LINE_SIZE	$2^{\text{CACHE_OFFSET_LEN}} = 64$
CACHE_SETS_COUNT	$2^{\text{CACHE_IDX_LEN}} = 16$
CACHE_LINE_COUNT	$\frac{\text{CACHE_SIZE}}{\text{CACHE_LINE_SIZE}} = 64$
CACHE_WAY	$\frac{\text{CACHE_LINE_COUNT}}{\text{CACHE_SETS_COUNT}} = 4$

Что было сделано? LRU и pLRU

Реализована память mem в виде массива байт размера “MEM_SIZE”. Эмулирую выделение памяти на ней.

Кэши реализованы как массивы [CACHE_WAY][CACHE_SETS_COUNT]

Есть структура кеш линии в обоих случаях, поддерживаем в ней доп флаг валидности линии и (pLRU: 1 бит под флаг вытеснения, LRU: 2 бита под флаг вытеснения, так как ассоциативность = 4 и 2 бит хватает).

pLRU

Ищем самый левый бит равный 0 или невалидную линию и замещаем ее, ставя туда единицу. Если все единицы - проходимся и зануляем всё.

LRU

Ищем невалидную кеш линию или линию со старостью = 3. Замещаем её. Ставим ей флаг 0, возраста остальных линий (возраста меньшего чем ту, которую выкинули) увеличиваем, поддерживая уникальность.

честно, я вообще не знаю, о чём ещё тут писать,