Контрольная работа
по курсу
«Машинная арифметика в рациональных числах»

Москва, 2020

ОСОБЕННОСТИ ВЫЧИСЛЕНИЙ С ИСКЛЮЧЕНИЕМ ОШИБОК ОКРУГЛЕНИЙ

1. Необходимость вычислений с очень «длинными числами», что является их серьёзным недостатком

Например

$A_1/B_1 + A_2/B_2 = (A_1B_2+A_2B_1)/(B_1B_2)$

2. Если псевдопереполнение возникают в процессе вычислений, но не в конечном результате, то он будет правильным. Например, для дробей Фарея 3-го порядка

(1/2)*(1/2) + (3/2)*(1/2) = 1

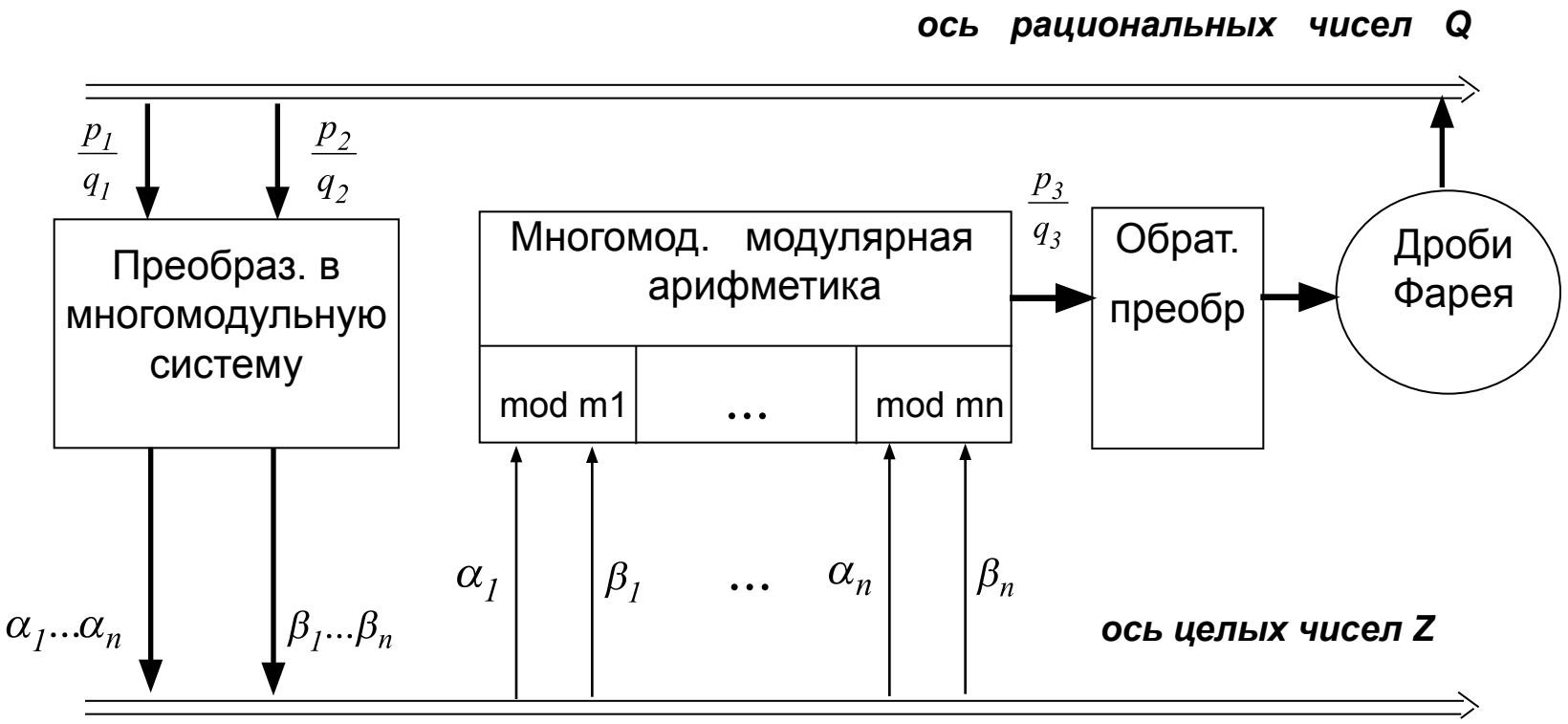
10*10+11*10=100+110 = 210 mod 19 = 1

ГДЕ ПРИМЕНЯТЬ ВЫЧИСЛЕНИЙ С ИСКЛЮЧЕНИЕМ ОШИБОК ОКРУГЛЕНИЙ?

1. Для любых вычислительных задач о которых известно, что решения это дроби Фарея определенного порядка или есть оценка сверху для них.

2) Для задач, где требуется очень высокая точность вычислений.

МОДЕЛЬ ВЫЧИСЛЕНИЙ С ИСКЛЮЧЕНИЕМ ОШИБОК ОКРУГЛЕНИЯ НА ОСНОВЕ МНОГОМОДУЛЬНОЙ АРИФМЕТИКИ



Порядок дробей Фарея

$$N = \left\lfloor \sqrt{\frac{m_1 \cdot m_2 \cdot \ldots \cdot m_n - 1}{2}} \right\rfloor$$

Примере для схемы вычислений с исключением ошибок округления по нескольким модулям

$(A-d0)*(m_1\ ^{-1}) = d_1 + m_2 * d_2$

$(A-d0)*(m_1\ ^{-1}) - d_1 = m_2 * d_2$

(?, 3, 4) – 3 = (?,0,1)

$m_2\ ^{(-1)}$ mod m3 = 3

(?,?,3)

d_2=(?,0,1)*(?,?,3) = 3

Оценки сверху для задач

$(A-d0)*(m_1\ ^{-1}) = d_1 + m_2 * d_2$

$(A-d0)*(m_1\ ^{-1}) - d_1 = m_2 * d_2$

(?, 3, 4) – 3 = (?,0,1)

$m_2\ ^{(-1)}$ mod m3 = 3

(?,?,3)

d_2=(?,0,1)*(?,?,3) = 3

Параллельная реализация вычислений с исключением ошибок округления

$(A-d0)*(m_1\ ^{-1}) = d_1 + m_2 * d_2$

$(A-d0)*(m_1\ ^{-1}) - d_1 = m_2 * d_2$

(?, 3, 4) – 3 = (?,0,1)

$m_2\ ^{(-1)}$ mod m3 = 3

(?,?,3)

d_2=(?,0,1)*(?,?,3) = 3

Избыточная система счисления

В системах счисления рассмотренных выше с фиксированным основанием, набор цифр был ограничен $\{0,\ldots,\ r-1\}$. Расширим этот набор цифр:

$$\{\overline{(r-1)},\overline{(r-2)}\ldots,\ \bar{1},0,1,\ldots,\ r-1\},$$

где

$$\bar{i} = -i$$

Каждая цифра или положительная или отрицательная, таким образом, нет необходимости в отдельном знаковом разряде. Такая система счисления называется знакоразрядной системой счисления (signed-digit number system)

Рассмотрим пример $r = 10$ (т.е. множество допустимых цифр $\{\overline{9},\overline{8},\ldots,\ \overline{1},0,1,\ldots,\ 8,9\}$ |

Избыточная система счисления

Пусть число разрядов $n = 2$, тогда диапазон представления двухразрядных чисел $\overline{99} < X < 99$, который включает в себя 199 чисел. Однако, общее количество двухразрядных чисел равно $19^2 = 361$ и, следовательно, некоторые числа имеют более чем одно представление.

Такая система счисления называется избыточной.

Например, чисел $(01) = (1\overline{9}) = 1$, число $(0\overline{2}) = (\overline{1}8) = -2$.

Число 0 представляется единственным образом. Таким образом, 361 – 199 = 162 числа являются избыточными, т.е. избыточность составляет 81%.

Избыточность системы счисления используется для ускорения арифметических операций. С другой стороны, высокий уровень избыточности требует большего количества числа битов для представления каждой цифры.

Избыточная система счисления

Каждая цифра принадлежит диапазону $x_i \in \{\overline{a}, \overline{a-1}, \ldots, \overline{1}, 0, 1, \ldots a\}$ и выполняется неравенство:

$$\left\lceil \frac{r-1}{2} \right\rceil \le a \le r-1, \qquad (1)$$

где $\lceil x \rceil$ обозначает наименьшее целое, большее или равное x.

Верхняя граница этого неравенство следует из того, что по меньшей мере r различный цифр требуются для представления цифры по основанию r. Каждая цифра принадлежит диапазону $\overline{a} \le x \le a$, их количество – $2 \cdot a + 1$. Следовательно, для избыточности системы счисления должно выполняться неравенство $2 \cdot a + 1 \ge r$, отсюда следует нижняя граница этого неравенства.

Избыточная система счисления

Важное свойство отсутствие распространение переноса далее чем на соседний разряд при сложении. Рассмотрим правило сложения чисел в этой системе счисления.

Пусть даны два числа $(x_{n-1}, \ldots, x_0)$ и $(y_{n-1}, \ldots, y_0)$. Сумму этих чисел обозначим через $(s_{n-1}, \ldots, s_0)$.

Время сложения чисел не зависит от разрядности чисел и выполняется с помощью следующих шагов:

Избыточная система счисления

Время сложения чисел не зависит от разрядности чисел и выполняется с помощью следующих шагов:

1. Определение суммы $u_i = x_i + y_i - r \cdot c_i$, где

$$c_i = \begin{cases} 1, если\ (x_i + y_i) \geq a \\ -1, если\ (x_i + y_i) \leq -a \\ 0, в\ противном\ случае\ (|x_i + y_i| < a) \end{cases}$$

2. Определение окончательной суммы $s_i = u_i + c_{i-1}$

Избыточная система счисления

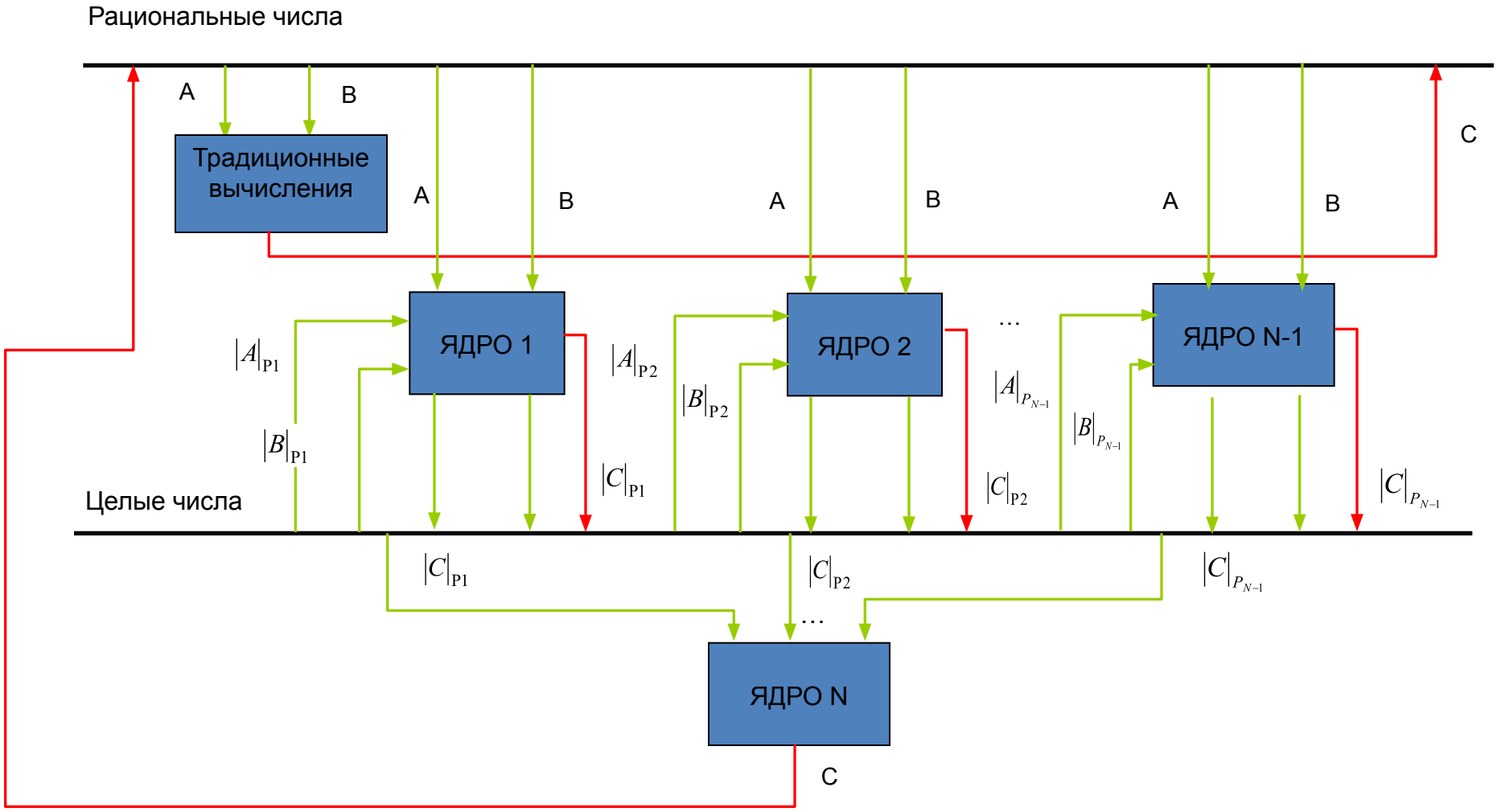
Утверждение

Если выполняется условие:

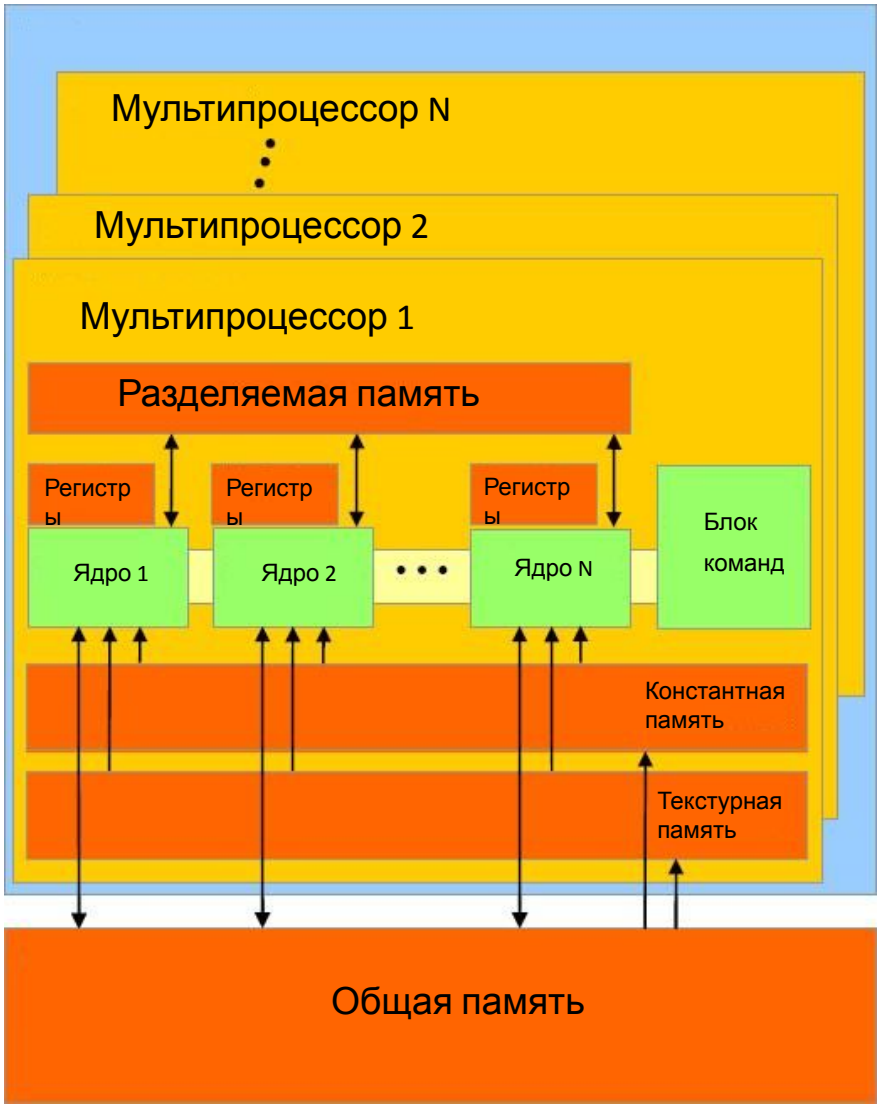
$$\left\lceil \frac{r+1}{2} \right\rceil \le a \le r-1 ,$$

то при сложении перенос не распространяется далее чем на соседний разряд.

Возможная реализация схемы высокоточных вычислений с отложенным округлением с использованием многоядерного процессора

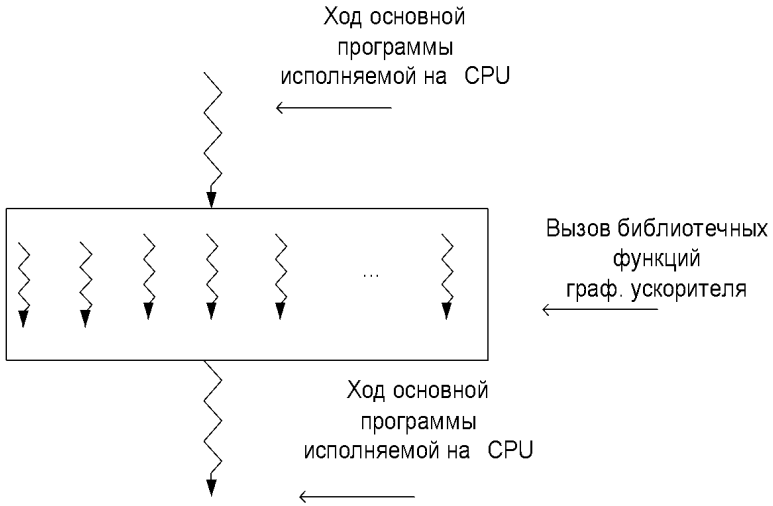


Структура многоядерного графического ускорителя NVIDIA.



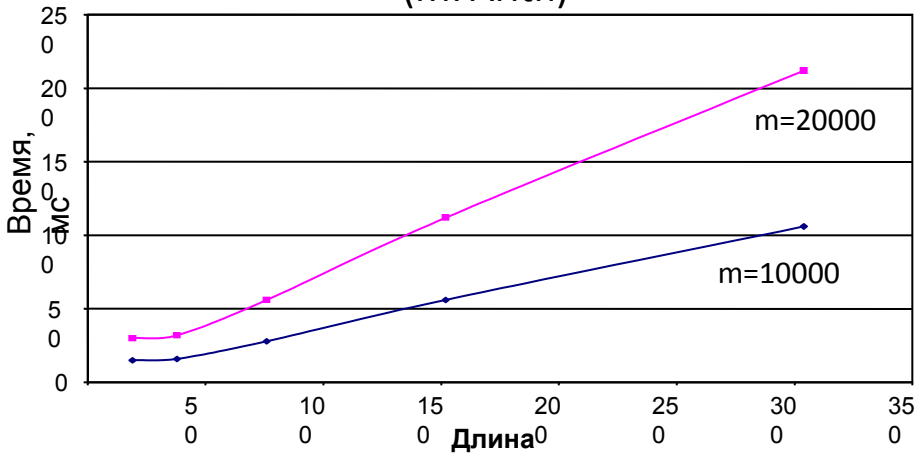
Графический ускоритель GeForce 9600M GT: 32 ядра, 4 мультипроцессора, 512Мб общей памяти.

Архитектура SIMD. Порядок использования графического ускорителя:

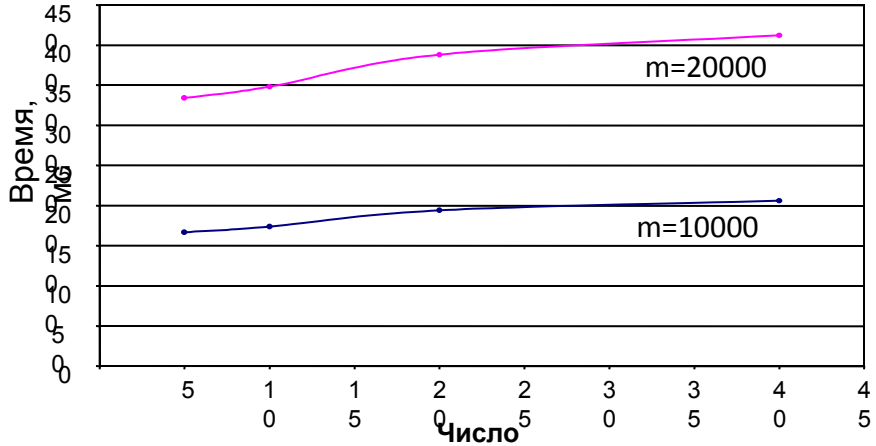


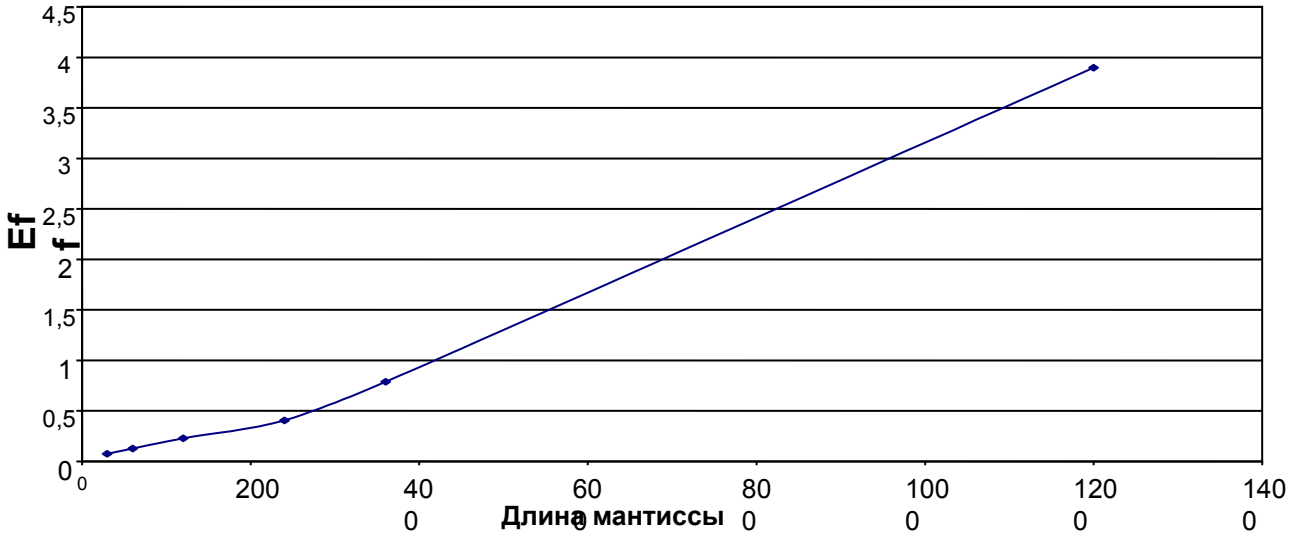
Оценка эффективности модулярной арифметики на примере вычисления скалярного произведения

Зависимость времени вычисления скалярного произведения от длины мантиссы в библиотеке высокоточных вычислений (MPArith)



Зависимость времени вычисления скалярного произведения в модулярной системе счисления от числа модулей (на многоядерном графическом ускорителе)





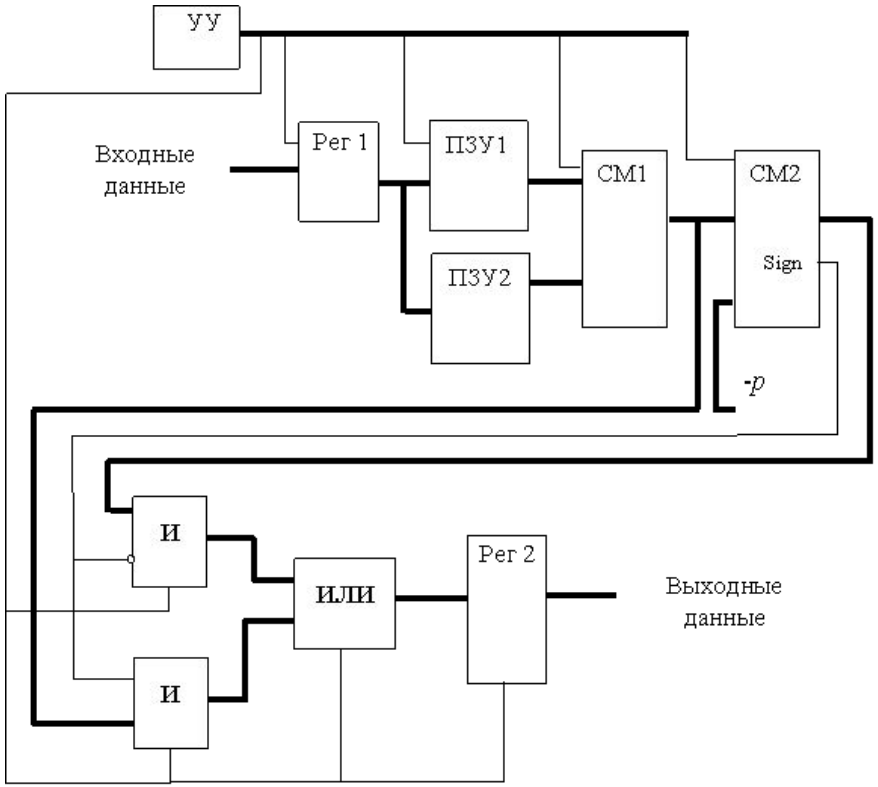
$$Eff = \frac{T_1}{T_2},$$

T_1 —время вычислений с использованием библиотеки MPArith,

T_2 — время вычислений в модулярной арифметике при одинаковой точности

Структурные схемы узлов модулярного сопроцессора высокоточных вычислений

1. Структурная схема устройства преобразования целых чисел из позиционной системы в модулярную систему счисления



Исходные данные:

N – целое число

p – модуль

Результат:

$|N|_p$

Патенты РФ: 2235423, 2293437, 2305861.

$$|N|_p \equiv \left|\sum_{i=0}^{t-1} c_i \cdot b^i\right|_p \equiv \sum_{i=0}^{t-1} \left|c_i \cdot b^i\right|_p \equiv \sum_{i=0}^{t-1} \left||c_i|_p \cdot |b^i|_p\right|_p \equiv$$

$$\equiv \left|\sum_{i=0}^{t-1} |k_i|_p\right|_p = \left|\sum_{i=0}^{\lfloor (t-1)/2 \rfloor} |k_i|_p + \sum_{i=\lceil (t-1)/2 \rceil}^{(t-1)} |k_i|_p\right|_p = |k_r + k_l|_p,$$

где

$$k_i = |c_i|_p * |b^i|_p,$$

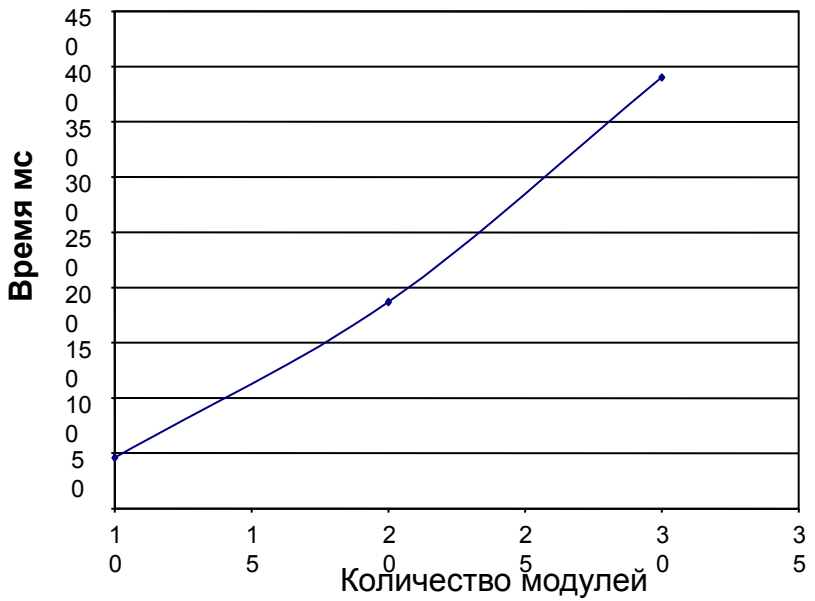
$$k_r = \sum_{i=0}^{\lfloor (t-1)/2 \rfloor} |k_i|_p,$$

$$k_l = \sum_{i=\lceil (t-1)/2 \rceil}^{(t-1)} |k_i|_p,$$

$$t = \lceil \log_b N \rceil$$

Рекомендации по применению высокоточных вычислений в модулярной арифметике

Зависимость времени обратного преобразования чисел из МСС от числа модулей.



В отличие от времени вычислений время обратного преобразования, сильно зависит от числа модулей, как это видно из графика. Время округления, сравнения чисел также сильно зависят от числа модулей.

Поэтому эффективность применения высокоточных вычислений в модулярной арифметики будет тем выше, чем меньше в задаче операций преобразования результатов из модулярной системы счисления, округлений, сравнений.