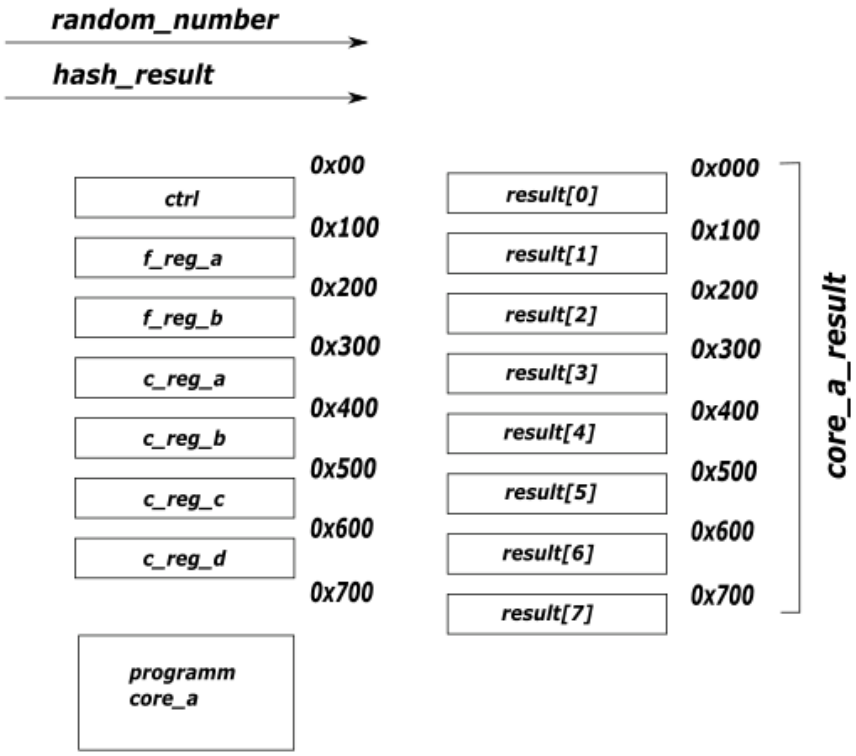
Сопроцессор эллиптической криптографии.

Регистровая модель. Краткая справка.

Сопроцессор предназначен для ускорения вычислений преобразований на эллиптических кривых. Ускорение вычислений достигается за счёт применения блоков аппаратных ускорителей задач модулярной арифметики и оптимизации набора инструкции сопроцессора для ускорения вычисления задачи формирования и проверки цифровой подписи.

Пользователю предоставляется на доступ 14 регистров и память программ (условно на рисунке). Левая группа регистров доступна только на запись, правая только на чтение.



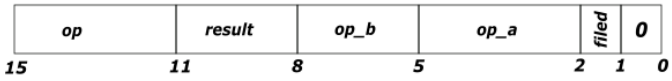
Для произведения вычислений необходимо загрузить программу, произвести загрузку регистров, и передать команду на запуск вычислений (возможен старт с конкретного адреса программы). После произведения вычислений результаты доступны по регистрам result[i].

Регистры f_reg_a, f_reg_b – хранят значения конечного поля, в котором происходят вычисления. Для оптимизации алгоритма расчёта подписи происходит быстрое переключение на работу в одном из конечных полей. Эти регистры не доступны, для внутреннего ядра и определяют только поля в котором производиться вычисления.

Для расчётов также используются два аппаратных числа (результат хеширования и случайное число) доступные при обработке со стороны вычислительного ядра.

Формат инструкций.

В качестве формата инструкций используется примитивная ISA (instructions set architecture). Инструкции, соответственно разделены на вычислительные инструкции и инструкции условного перехода.



execute

branches

Описание формата вычислительных инструкций в таблице ниже:

Поле	Описание	Примечание
field	Выбирает конечное поле для вычислений: 0 – в качестве поля f_reg_a 1 – в качестве поля f_reg_b	
Op_a	Операнд А. В качестве операнда принимаются следующие аргументы: 0000 - аппаратная единица 0001 - аппаратный нуль 0010 – аппаратный вход a (функция хеширования) 0011 – аппаратный вход b (случайное число) 0100 – core_reg_a 0101 – core_reg_b 0110 – core_reg_c 0111 – core_reg_d 1000 – result[0] 1001– result[1] 1010– result[2] 1011– result[3] 1100– result[4] 1101– result[5] 1110– result[6] 1111– result[7]	Аппаратные 1 и 0 используются для операций с бесконечными точками
Op_b	Операнд А. В качестве операнда принимаются следующие аргументы: 000 – result[0] 001– result[1] 010– result[2] 011– result[3] 100– result[4] 101– result[5] 110– result[6] 111– result[7]	
Result	Адрес регистра для записи результата: 000 – result[0] 001– result[1] 010– result[2] 011– result[3] 100– result[4] 101– result[5] 110– result[6] 111– result[7]	

Op	Код операции Лёгкие операции: 0000: result = op_a-1 0010: result = op_a+1 0100: result = op_a>>1 0110: result = (a+b) mod q 1000: result = (a-b) mod q 1010: result = op_a 1100: result = op_a == op_b ? 1110: result = op_b is odd/even Тяжёлые операции: 0001: result = (op_a*op_b) mod q 0011: result = (op_a) mod q 0101: result = (op_a^-1) mod q 1001: result = request new random num	

Описание формата инструкций условного перехода в таблице ниже:

Поле	Описание	Примечание
branch	Переход: 0001 - nop 0011 – ret 0101 – jump to addr 1101 – jump to addr if compare_flag == true, else jump to next addr	
Addr	Адрес перехода	