Российский процессор Эльбрус

Плюсы:

1. Это не калька.

То есть это не скопировано с зарубежного аналога и не выдается за свое со словами: «Вау, смотрите, вот наше импорто замещение»

У него архитектура Е2К (Эльбрус 2000)

В ее основе лежит VLIW (Very Long Instruction Word – очень длинное командное слово)

Как правило, есть 2 доминирующие архитектуры в мире: CISC(x86) и RISC(ARM)

(Оговорка на самом деле их до ебени матери)

VLIW

Это очень длинное командное слово

Предположим, что вы VLIW я вам даю инструкцию: записать в тетрадке что-то, выделить красным цветом, выделит…

Фиксированная длина слова

Идея VLIW базируется на том, что задача эффективного планирования параллельного выполнения нескольких команд возлагается на «разумный» компилятор. Такой компилятор вначале исследует исходную программу с целью обнаружить все команды, которые могут быть выполнены одновременно, причем так, чтобы это не приводило к возникновению конфликтов. В процессе анализа компилятор может даже частично имитировать выполнение рассматриваемой программы. Наследующем этапе компилятор пытается объединить такие команды в пакеты, каждый из которых рассматривается так одна сверхдлинная команда.

Например,

Я знаю только русский язык

Вы знаете только французский.

Я хочу вас угостить холсом, но вы не понимаете чего я от вас хочу. Тогда я обращаюсь к умному компилятору (Страбыкин Дмитрий Алексеевич) и говорю, что вот я хочу раздать холс и передаю ему. Он в свое время делит холс на ваше количество, говорит вам на французском, что вот вас угощают холсом и раздает вам его.

Регистры предикатов – по сути память, в которой хранится либо 0, либо 1

Это нужно для условия

Псевдокод

If (Меня слушают)

Then Good\_locking;

Else {Write(‘А следовала бы’);}

Эльбрус загружает обе части условия в память. И как только выполняется условие «Меня слушают» он записывает в регистр предикатов 1. И когда надо определиться какой код надо выполнять, он не думает, а считывает значение из регистра предиката и выполняет то, что ему нужно

Плюсы: Нет простоя процессора в прыжке

Он может готовиться к прыжку параллельно с исполнением других команд.

Если в обычном случае процессор не знает, что дальше, то у нас же все уже известно при помощи компилятора.

В интел и АМД есть бекдоры

Регистры в которых хранятся данные, которые потом используются процессором у Эльбруса разделяются на 3 стека



Если вы адекватно написали прогу и адекватно использовали файла подкачки, то будет вам счастье

Режим безопасных Вычислений

Эльбрус умеет компилировать приложения в защищенном режиме. И в таком режиме прога не может вылезти за предел тех данных, с которыми ей явно разрешили работать.

То есть если вы сказали проге работать с массивом на 256 символов(бит), а ей попытались подсунуть 257, то она просто упадет. Выдаст ошибку и откажется работать. И это хорошо тем, что взломать вас будет сложнее Не получится подать одних данных под видом других.

Так как в любой проге есть ошибки, стоит понимать, что защищенный режим только для узкой категории людей.

Оптимизация кода под интел – оптимизация кода и под Эльбрус.

Так почему же это работает? Нам процессорИнтел впихивает свои продукты под другим неймингом?

А дело в том, что компилятор Эльбруса LCC может находить для низкоуровнего кода для процессора Интел находить аналогичные команды для процессора Эльбрус и вставлять их при компиляции.

Причем пишите на ассемблере одного процессора, а в итоге он работает на ассемблере другого процессора.

Как эльбрус может использовать avx инструкции, которые под процессор интел. А мы с вами знаем, что для того, что бы эти avx работали нужны 256 бит регистры. А в Эльбрусе они 128 бит.

И тут нам опять приходит на помощь компилятор LCC. Он может имитировать наличие регистров под AVX он задействует 2 своих регистра, выдавая их за один.

Профилирование

Если вы четко знаете с какими данными будете работать, то вы можете натренировать ее.

Берете программу, компилируете особенным способом. После чего программа, то есть компилятор автоматически генерирует данные и пытается понять с каким типом данных работаете, что это за аднные