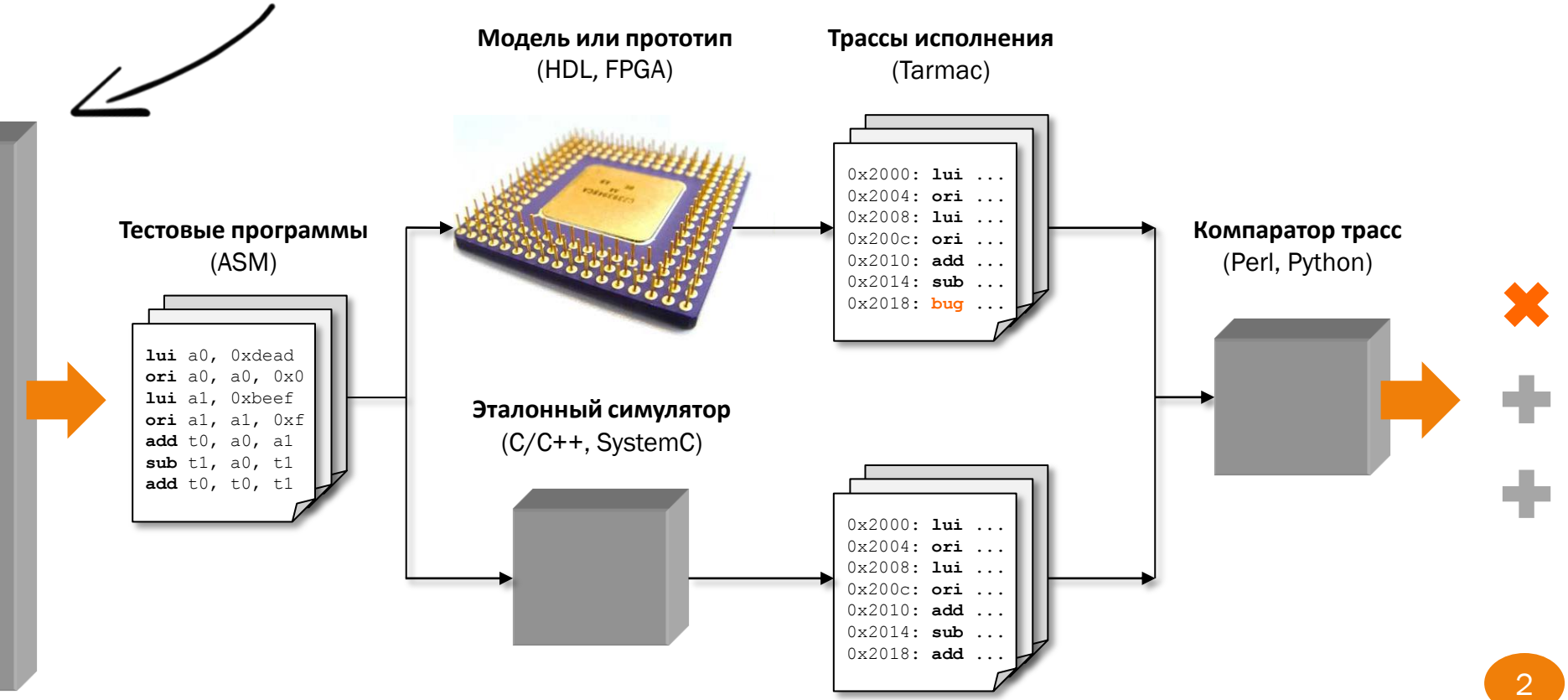
XII международная конференция
CEE-SECR / РАЗРАБОТКА ПО
28 - 29 октября, Москва

Автоматизированная разработка генераторов тестовых программ для микропроцессоров на примере MIPS

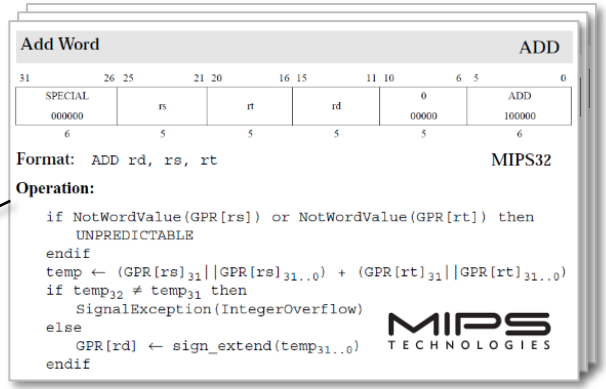
Александр Камкин

Институт системного программирования РАН

Это генератор тестовых программ



Как его разработать?



Руководство по системе команд

Параметры генерации (шаблон)

- Последовательности команд
- Ограничения на данные
- Распределения вероятностей
- и т.д. и т.п.

```
TEST_PROGRAM:
 addiu   $sp,$sp,-24
 sw      $fp,16($sp)
 move    $fp,$sp
 sw      $4,24($fp)
 sw      $5,28($fp)
 sw      $0,12($fp)
 j       LABEL_2
 nop
LABEL_3:
 lw      $2,12($fp)
 addiu   $2,$2, 1
 sw      $2,8($fp)
 j       LABEL_4
 nop
LABEL_5:
 lw      $2,12($fp)
 sll     $2,$2,3
 move    $3,$2
 lw      $2,24($fp)
 addu    $2,$3,$2
 ldc1    $f2,0($2)
 lw      $2,8($fp)
 sll     $2,$2,3
 move    $3,$2
 lw      $2,24($fp)
 addu    $2,$3,$2
 ldc1    $f0,0($2)
 c.lt.d  $f0,$f2
 nop
 bc1t    LABEL_8
 nop
 j       LABEL_6
 nop
```

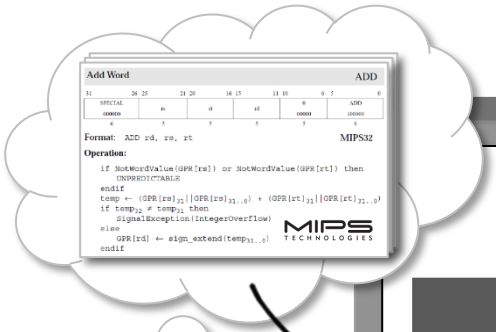
```
$2,12($fp)
$2,$2,3
$3,$2
$2,24($fp)
$2,$3,$2
$f0,0($2)
$f0,0($fp)
$2,12($fp)
$2,$2,3
$3,$2
$2,24($fp)
$4,$3,$2
$2,8($fp)
$2,$2,3
$3,$2
$2,24($fp)
$2,$3,$2
$f0,0($2)
$f0,0($4)
$2,8($fp)
$2,$2,3
$3,$2
$2,24($fp)
$2,$3,$2
$f0,0($fp)
$f0,0($2)

$2,8($fp)
$2,$2,1
$2,8($fp)
```

```
$2,8($fp)
$3,28($fp)
$2,$2,$3
$2,$0,$L5

$2,12($fp)
$2,$2,1
$2,12($fp)

$3,12($fp)
$2,28($fp)
$2,$2,-1
$2,$3,$2
$2,$0,$L3

$sp,$fp
$fp,16($sp)
$sp,$sp,24
$31
```

Если только синтаксис

- Нарушение предусловий: **`UNPREDICTABLE`**
- Обращения к **неинициализированным** данным
- Переходы назад могут вызвать **зацикливания**
- **Низкая вероятность** возникновения **corner-cases**
- **Невозможность** создания **self-checking** тестов

Нужно учитывать семантику



Генератор на базе MicroTESK

Транслятор

Модель процессора

Спецификации

Тестовые шаблоны

Ядро генератора

Тестовые программы

```
lui a0, 0xdead
ori a0, a0, 0x0
lui a1, 0xbeef
ori a1, a1, 0xf
add t0, a0, a1
sub t1, a0, t1
add t0, t0, t1
```

Спецификации на nML

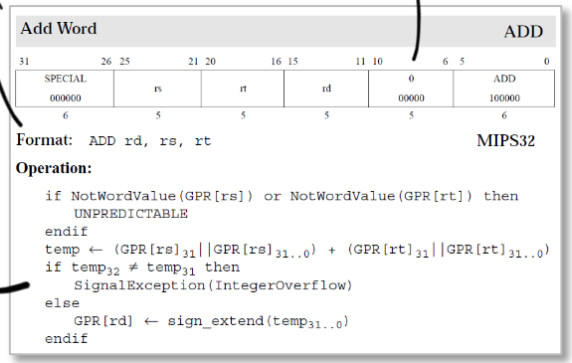
```
mode R (i: card(5)) = GPR[i]
  syntax = format("r%d", i)
  image  = format("%5s", i)
  action = {
    ...
  }
```

Режимы доступа

```
type DWORD = card(64)

reg GPR [32, DWORD]

op ADD(rd: R, rs: R, rt: R)
  image  = format("0000%5s%5s%5s0000100000", rs.image, rt.image, rd.image)
  syntax = format("add %s, %s, %s", rd.syntax, rs.syntax, rt.syntax)
  action = {
    if (NotWordValue(rs) || NotWordValue(rt)) then
      unpredictable;
    endif;
    temp = rs<31>::rs<31..0> + rs<31>::rt<31..0>;
    if (temp<32> != temp<31>) then
      exception("IntegerOverflow");
    else
      rd = sign_extend(DWORD, temp<31..0>);
    endif;
  }
```

Регистры

Команды



Спецификация

Шаблоны на Ruby

Всего $2 \times 2 = 4$ варианта

```
class MyTemplate < Template
  def initialize ... end
  def pre        ... end
  def post       ... end

  def run
    block(:combinator => 'product') {
      iterate {
        add t0, t1, t2
          do situation('Normal') end
        add t0, t1, t2
          do situation('IntegerOverflow') end
      }
      iterate {
        sub t3, t4, t5
          do situation('Normal') end
        sub t3, t4, t5
          do situation('IntegerOverflow') end
      }
    }.run
  end
end
```

Пролог / эпилог

Ассемблер

Всякие “фишки”

Инициализирующий код

```
lui r9,  0x858d
ori r9,  r9,  0xc02e
lui r10, 0x58d2
ori r10, r10, 0x7219
lui r12, 0x898a
ori r12, r12, 0x49b7
lui r13, 0x0f31
ori r13, r13, 0xc65a
```

Тестовое воздействие

```
add r8,  r9,  r10
sub r11, r12, r13
```

Заключение

- Высокая автоматизация на базе спецификаций
- Гибкая настройка на разные архитектуры
- Расширяемость
- Поддержка MIPS
- Open source
- Применяется в реальных проектах

Архитектура	MIPS™
Число команд	250 из 320
Объем nML спецификаций	4.5 KLOC
Трудоемкость	4 чел.-мес.

Будущее MicroTESK

- **Online**-генерация тестовых программ
- Генерация **симуляторов**
- Оценка **тестового покрытия**
- Настраиваемая **компиляция**
- **Верификация ПО** с учетом целевой архитектуры

http://www.isprasopen.ru

MicroTESK

Verification Technology for Microprocessors

```
_start:     .org  0x50000
            nop
            adr   x0, page_table
            msr   ttbr0_el1, x0
            movz  x0, #0x8082, lsl #16
            movk  x0, #0x3219, lsl #0
            msr   tcr_el1, x0
            isb
            movz  x0, #0x30c5, lsl #16
            movk  x0, #0x0831, lsl #0
            msr   sctlr_el1, x0
            isb
            movz  x0, #0x0020, lsl #16
            movk  x0, #0x4000, lsl #0
            ldar  x1, [x0], #01
```

http://forge.ispras.ru/projects/microtesk

microtesk-support@ispras.ru