```
Entry
\mathbf{X}
              = v0
    v0 1
    v1 1
              := v1
             := 0
   phi_mul := 0
    exitcond := (i == 32)
         := i + 1
   i 1
   next mul := phi mul + 0x9e3779b9
    next_reg := next_mul
Y
              := v1 1 << 4
   tmp
   tmp1 := tmp + k0_read
   tmp2 := v1_1 >> 5
   tmp3 := tmp2 + k1\_read
    tmp4
         = v1 1 + next reg
    next reg2 := next reg
    tmp5 := tmp3 xor tmp4
        := tmp5 xor tmp1
   tmp6
    v0 2
              := tmp6 + v0 1
   tmp7
              = v0 2 << 4
    tmp8
         := tmp7 + k2 read
        = v0 2 >> 5
   tmp9
   tmp10 := tmp9 + k3_read
   tmp11 := v0_2 + next_reg2
   tmp12 := tmp11 xor tmp8
    tmp13
              := tmp12 + tmp10
    v1 2
              := tmp13 + v1 1
             = v0 2
    v0_1
         = v1 2
    v1 1
                                                    S_{pre}
X
              :=i 1
    phi mul := next mul
    exitcond := (i == 32)
              := i + 1
    i 1
   next mul := phi mul + 0x9e3779b9
    next reg := next mul
Y
              := v1 1 << 4
    tmp
    tmp1
              := tmp + k0 read
                                                   S_{loop}
              = v1 1 >> 5
    tmp2
        := tmp2 + k1 read
    tmp3
    tmp4
         = v1 1 + next reg
    next_reg2 := next_reg
Z
   tmp5
              := tmp3 xor tmp4
    tmp6
              := tmp5 xor tmp1
    v0 2
              := tmp6 + v0 1
              = v0 2 << 4
    tmp7
    tmp8
              := tmp7 + k2 read
              = v0 2 >> 5
    tmp9
   tmp10
              := tmp9 + k3 read
    tmp11
              := v0 2 + next reg2
    tmp12
              := tmp11 xor tmp8
    tmp13
              := tmp12 + tmp10
    v1 2
              := tmp13 + v1 1
   v0 1
              = v0 2
    v1 1
              := v1 2
              :=i 1
    phi_mul
              := next mul
              := (i == 32)
    exitcond
                                                   SpreExit
              := i + 1
    i 1
                         Exit
```