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

Exit