```
0x2d3
--------
assert(0 == msg.value)
$s2 = c[0x4]
$s3 = c[0x24]
$s6 = c[0x44]
$s4 = $s6
assert(0 == (0xff & (s[0x0] >> 0xa0)))
$s7 = ad_mask & s[0x2]
m[$m] = 0x6352211e << 0xe0
$s10 = 0x4 + $m
m[$s10] = $s2
assert(extcodesize($s7))
assert(call(msg.gas, $s7, 0x0, $m, (0x20 + $s10) - $m, $m, 0x20))
$s5 = m[$m]
assert(msg.sender == (ad_mask & $s5))
$s7 = ad_mask & s[0x2]
m[$m] = 0x2972b0f0 << 0xe0
$s11 = 0x4 + $m
m[$s11] = ad_mask & self
$s12 = 0x20 + $s11
m[$s12] = $s2
assert(extcodesize($s7))
assert(call(msg.gas, $s7, 0x0, $m, (0x20 + $s12) - $m, $m, 0x20))
assert(m[$m])
assert($s3 > 0x0)
$s8 = block.timestamp
$s11 = 0x3c + $s8
assert(0 == ($s11 < $s8))
assert($s6 > $s11)
m[$m] = block.timestamp
$s12 = 0x20 + $m
m[$s12] = (ad_mask & $s5) << 0x60
$s12 = 0x14 + $s12
m[$s12] = $s2
$s12 = 0x20 + $s12
m[$s12] = $s3
$s6 = sha3($m, (0x20 + $s12) - $m)
$s7 = $m
$m = 0x80 + $m
m[$s7] = $s6
$s8 = 0x20 + $s7
m[$s8] = ad_mask & $s5
$s8 = 0x20 + $s8
m[$s8] = $s3
m[0x20 + $s8] = $s4
m[0x0] = $s2
m[0x20] = 0x3
$s8 = sha3(0x0, 0x40)
s[$s8] = m[$s7]
$s10 = 0x1 + $s8
s[$s10] = (ad_mask & m[0x20 + $s7]) | (0xffffffffffffffffffffffff0000000000000000000000000000000000000000 & s[$s10])
s[0x2 + $s8] = m[0x40 + $s7]
s[0x3 + $s8] = m[0x60 + $s7]
if (s[0x5] > 0x0){
  $s7 = ad_mask & s[0x1]
  m[$m] = 0x23b872dd << 0xe0
  $s12 = 0x4 + $m
  m[$s12] = ad_mask & msg.sender
  $s13 = 0x20 + $s12
  m[$s13] = ad_mask & (ad_mask & s[0x0])
  $s13 = 0x20 + $s13
  m[$s13] = s[0x5]
  assert(extcodesize($s7))
  assert(call(msg.gas, $s7, 0x0, $m, (0x20 + $s13) - $m, $m, 0x20))
  assert(m[$m])
}
m[$m] = $s6
$s14 = 0x20 + $m
m[$s14] = $s3
$s14 = 0x20 + $s14
m[$s14] = $s4
log3($m, (0x20 + $s14) - $m, 0x9493ae82b9872af74473effb9d302efba34e0df360a99cc5e577cd3f28e3cab2, $s2, ad_mask & $s5)
stop()
```