

ROI 计算

| | | | |
|-------------------------------|-------------|-------------|------------|
| 减半时间/年 | 4 | 2 | 1 |
| ROI - 90% Lock | 44% | 84% | 167% |
| ROI - 70% Lock | 57% | 108% | 214% |
| ROI - 50% Lock | 80% | 151% | 300% |
| Block Reward | 38.02570538 | 71.57779836 | 142.598571 |
| 1 st Year Reward | 1e+8 | 1.88e+8 | 3.75e+8 |
| 30% Reward/100M Validator MAN | 30% | 56% | 113% |

表格假设区块奖励持续年限为 16 年，到期截止，横排为假设的减半时间。

由于所有区块奖励方式均需要抵押金，故可以计算时假设所有抵押金获利机会均等，以 ICO 初始数量作为分母，估算第一年在不同锁定情况下的年度回报。

ROI=年度发币数量/锁币数量

考虑到 AI 计算/POW 受算力影响极大，故实际偏差可能较大

若考虑特定情况，比如验证者区块奖励占比 30%，验证节点总抵押金为 1 亿：
即可计算如下：

ROI=验证者年度奖励数量/验证者预估总抵押金

例如：

假设减半时间为 4 年,16 年发完的话一共分四个阶段,年度奖励数量 (x) 计算如下：

$$4*(X+0.5X+0.25X+0.125X)=7.5*10^8$$

可得 $X=1*10^8$

验证者区块奖励占比 30%，验证节点总抵押金为 1 亿

$$ROI=10^8*30\%/10^8=30\%$$

附简易代码:

```
period_4y=(365*3+366)*24*60*5
coins=7.5*10**8
ico=2.5*10**8
mul4=sum(0.5**x for x in range(4))
block_4y=coins/(period_4y*mul4)
print block_4y

period_2y=period_4y/2
mul2=sum(0.5**x for x in range(8))
block_2y=coins/(period_2y*mul2)

print block_2y

period_1y=period_4y/4
mul1=sum(0.5**x for x in range(16))
block_1y=coins/(period_1y*mul1)

print block_1y

for mul in [mul4*4, mul2*2, mul1]:
    print coins/mul
    for p in [0.9,0.7,0.5]:
        roi=100*(coins/mul)/(p*ico)
        print p,roi

reward= 7.5*10**8*0.3
validator=1*10**8
for mul in [mul4*4, mul2*2, mul1]:
    roi=100*(reward/mul)/validator
    print roi
```