

BIP42

A finite monetary supply for Bitcoin
(Pieter Wuille, April 1st 2014)

GetBlockValue() prior to BIP42

```
1  int64_t GetBlockValue(int nHeight, int64_t nFees)
2  {
3      int64_t nSubsidy = 50 * COIN;
4
5      // Subsidy is cut in half every 210,000 blocks
6      // which will occur approximately every 4 years.
7      nSubsidy >>= (nHeight / 210000);
8
9      return nSubsidy + nFees;
10 }
```

[on GitHub](#)

GetBlockValue() prior to BIP42

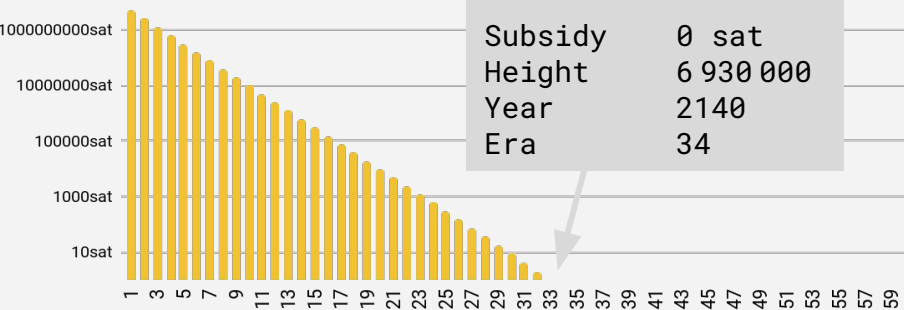
```
3     int64_t nSubsidy = 50 * COIN;           // int64_t 64 bit signed integer
[...]  
7     nSubsidy >>= (nHeight / 210000);     // >>= right shift and assign
```

Line 3: assigning `50 * 100 000 000` to `nSubsidy`.

Line 7: right shift `nSubsidy` by `nHeight / 210000` and assign back to `nSubsidy`.

Subsidy visualized for reward eras

Block subsidy prior to BIP42



PR #3842: Implementation of BIP 42

```
1  int64_t GetBlockValue(int nHeight, int64_t nFees)
2  {
3      int64_t nSubsidy = 50 * COIN;
4      int halvings = nHeight / 210000;
5
6      // Force block reward to zero when right shift is undefined.
7      if (halvings >= 64) // softforks in April 2262
8          return nFees;
9
10     // Subsidy is cut in half every 210,000 blocks
11     // which will occur approximately every 4 years.
12     nSubsidy >>= halvings;
13     return nSubsidy + nFees;
14 }
```

Closing thoughts

- BIP 42 is meant as an April fools joke (created on April 1st 2014)
- PR #3842: `Fix for GetBlockValue() after block 13,440,000` merged in April 2014
- Softforks in the year 2262
- `GetBlockValue()` is now `GetBlockSubsidy()`



Thank you and questions?