chaincode

BIP42

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

GetBlockValue() prior to BIP42

```
int64_t GetBlockValue(int nHeight, int64_t nFees)

int64_t nSubsidy = 50 * COIN;

// Subsidy is cut in half every 210,000 blocks

// which will occur approximately every 4 years.

nSubsidy >>= (nHeight / 210000);

return nSubsidy + nFees;

}
```

on GitHub

GetBlockValue() prior to BIP42

```
int64_t nSubsidy = 50 * COIN;

nSubsidy >>= (nHeight / 210000);

// int64_t 64 bit signed integer

// >>= right shift and assign

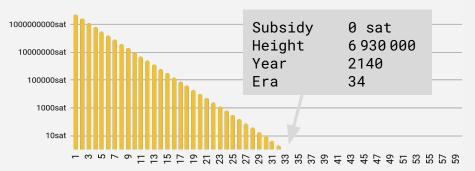
Line 3: assigning 50 * 100 000 000 to nSubsidy

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



Subsidy visualized for reward eras

Block subsidy prior to BIP42





The Bug



cppreference.com on bitwise shift operators:

If the value of the right operand is greater or equal to the number of bits in the promoted left operand, the behavior is undefined.

On all currently supported platforms the reward era cycle repeats itself every 64 halvings.

PR #3842: Implementation of BIP 42

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

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?