Bitcoin Mining and Improvements
Lublin, Poland 22 Sept 2014

Nicolas T. Courtois
Our Works on Bitcoin

- blog.bettercrypto.com


⇒ Section 2.6: Analysis of Bitcoin From The Point of View of Investors


- Poster: http://www.nicolascourtois.com/bitcoin/POSTER_100x_Secrypt2014_v1.0.pdf
Introducing Bitcoin
Crypto Currencies

Bitcoin In A Nutshell

• bitocoins are cryptographic tokens
  – stored by people on their PCs or mobile phones

• ownership is achieved through digital signatures:
  – you have a certain cryptographic key, you have the money.
  – publicly verifiable, only one entity can sign

• consensus-driven, a distributed system which has no central authority
  – but I will not claim it is decentralized, this is simply not true!
  – a major innovation is that financial transactions CAN be executed and policed without trusted authorities. Bitcoin is a sort of financial cooperative or a distributed business.

• based on self-interest:
  – a group of some 100 K people called bitcoin miners own the bitcoin “infrastructure” which has costed about 0.5-1 billion dollars (estimation)
  – they make money from newly created bitcoins and fees
  – at the same time they approve and check the transactions.
  – a distributed electronic notary system
Poland = 3rd Place Worldwide

Two Key Concepts

- Initially money are attributed through **Proof Of Work (POW)** to one public key A:
  - To earn bitcoins one has to “work” (hashing) and consume energy (pay for electricity).
  - In order to cheat one needs to work even much more (be more powerful than the whole network, for a short while).

- Money transfer from public key A to public key B:
  - Like signing a transfer in front of one notary which confirms the signature.
  - Multiple confirmations: another notary will re-confirm it, then another, etc…
  - We do NOT need to assume that ALL these notaries are honest.
    - At the end it becomes too costly to cheat.
Miracle Of Bitcoin

Removes two pillars of money:

- “trust”
  => P2P self-regulation
  <= self-interest?

- legal/government protection and policing
  => anarchy!
Citations

Bitcoin is:

• **Wild West** of our time [Anderson-Rosenberg]
Play Money?

A distinction play vs. real money has almost disappeared recently.
Crypto Currencies

Bitcoin=Freedom

A payment system in which
• it is THE PAYER who initiates the transaction
• controls the amount being paid
• money and payments are stored outside of the banking system [erodes the dominant position of banks]
• money cannot be confiscated [cf. Cyprus banks].
• it challenges fractional reserve banking [new!] and forces finance to become more “transparent”

“Troubled” bitcoin [The Economist May 2014] is certainly is here to stay

=> but now must face all sorts of competition and technical reforms [our work]
In Practice

![Image of a cryptocurrency transaction on a mobile device]

- Address: Lke9VyQHixh
- Balance: 249.9 mBTC (~USD 184.43)
- Status: Receiving 25 mBTC
- 1 BTC = USD 738.00 (Bitstamp)
P2P Payment
Bt is guide to, decentralized, no central authority, one ASIC one vote, => no third party risk [no need to trust the banker!]
• Knows no limits, borders, laws, etc…
  • Computers connected into a P2P network…
  • Every transaction can be downloaded by anyone…

Nicolas T. Courtois 2009-2014
Network Properties

Satoshi original idea [cf. Sect. 5 in his paper]:
– everybody participates equally
The Reality is VERY Different!

In violation of the original idea of Satoshi Bitcoin network has now 3 sorts of VERY DIFFERENT ENTITIES

- only “rich people” are mining
  - 100,000 people maybe
- some “full nodes”: they trust no one
  - 5,000 only
- wallet-only nodes
  - millions but not very active
*Panic – May 2014*

- # active nodes $<<$ # miners
- 6K $<<$ 100K

www.coindesk.com/bitcoin-nodes-need/

**Waning support**

Looking at a 60-day chart of bitcoin nodes shows that the number has gone down significantly. It went from 10,000 reachable nodes in early March to below 8,000 at the beginning of May.

Source: Bitnodes
Digital Currency
Digital Currency

=>PK-based Currency,
an important modern application of Digital Signatures!
Main Problem:

This capability can be “spent twice”.

Avoiding this “Double Spending” is the main problem when designing a digital currency system.

NOT yet solved in a satisfactory way, instability, slow transactions, more about this later.

Crypto
**Crypto Citations**

About Bitcoin:
• The accuracy of past transactions is guaranteed by cryptography,
SHA256

- SHA-256 hash function
- provides integrity of everything [hard to modify]
Block Chain
**Bitcoin Mining**

- **Minting:** creation of new currency.
- **Confirmation + re-confirmation** of older transactions

Random Oracle – like mechanism

**Ownership:**
- “policed by majority of miners”:
- only the owner can transfer

[a part of] 25 BTC produced.

[Diagram showing minting process with inputs like miner’s public key, random number generator (RNG), data from previous transactions, and hash output requiring 66 zeros at the start.]
Block Chain

Def: A transaction database shared by everyone.

Also a ledger.

Every transaction since ever is public.

Each bitcoin “piece” is a union of things uniquely traced to their origin in time

(cf. same as for several banknotes due to SN)
Hash Power => Security???

Sams writes: "The amount of capital collectively burned hashing fixes the capital outlay required of an attacker […] to have a meaningful chance of orchestrating a successful double-spend attack […]"

NO THIS IS MISTAKEN
(see our paper)
Crypto Currencies

Crazy Hash Power Increase

Nearly doubled every month… 1000x in 1 year.

Thm:

\[ 1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \ldots = 2 \]

the total income is only twice the income for the first month.
Bitcoin Address

To: 1K2CcfWY5sBL2x8eQWXpcmjPCgoXdi36
Amount: 1.0 BTC
SEND
Ledger-Based Currency

A “Bitcoin Address” = a sort of equivalent of a bank account.
Bitcoin Ownership

Amounts of money are attributed to public keys. Owner of a certain “Attribution to PK” can at any moment transfer it to some other PK (== another address).
Transfer

To: 1K2CcfWYw5sBL2xSeQWXpcmjPCgoXdi36
Amount: 1.0 BTC
SEND
Bitcoin Transfer

Transactions have multiple inputs and multiple outputs.

Input Bitcoin Addresses
0.2 BTC 1.3 BTC

Transaction Signed by All Owners with their SK

1.0 BTC 0.499 BTC + Fees 0.001 BTC

Output Bitcoin Addresses
Bitcoin
Mining
Money Out of Thin Air
Crypto Currencies

Bitcoin vs. Klondike

2012-2014
>100,000 miners
maybe ½ - ¾???? were victims of scams and paid for miners were not delivered in reasonable time

1896-1899
100,000 miners, 4,000 struck gold
Bitcoin Mining

- Minting: creation of new currency.
  Creation of “money”
  + re-confirmation
  of older transactions

data from previous transactions

HASH

Nicolas T. Courtois 2009-2014
Def: The bitcoin transaction database shared by everyone.
Ownership:
- “policed by miners”:

`HAS`H

Miner’s public key

Data from previous transactions

RNG

Must start with 66 zeros
Crypto Currencies

Bitcoin Mining

• Minting: creation of new currency. Creation+re-confirmation of older transactions

Random Oracle – like mechanism

Means: treat as a DETERMINISTIC black box which answers at random.

YES it is…
However now I’m going to show it isn’t.

Nicolas T. Courtois 2009-2014
Bitcoin Mining

- Minting: creation of new currency. Creation+re-confirmation of older transactions

Random Oracle – like mechanism

Means: treat as a DETERMINISTIC black box which answers at random.

YES it is,

However now I’m going to show it isn’t.

Marginal improvement (a constant factor).
Five Generations of Miners

1. CPU Mining

Example:
Core i5 2600K, 17.3 Mh/s, 8 threads, 75W

CPU = about 4000 W / Gh/s
Four Generations
Crypto Currencies

Four Generations of Miners

2. GPU Mining

Example:
NVIDIA Quadro NVS 3100M, 16 cores, 3.6 Mh/s, 14W

CPU = about 4000 W / Gh/s, in this case
GPU = about 4000 W / Gh/s, in this case

Who said GPU was better than CPU?
Not always.
Four Generations of Miners

3. FPGA Mining

Example:
ModMiner Quad, 4 FPGA chips, 800 Mh/s, 40W

CPU, GPU = about 4000 W / Gh/s
FPGA = about 50 W / Gh/s, in this case
Four Generations of Miners

3. FPGA Mining

Example:
ModMiner Quad, 4 FPGA chips, 800 Mh/s, 40W

CPU, GPU = about 4000 W / Gh/s
FPGA = about 50 W / Gh/s

100x less energy.
Five Generations of Miners

FPGA: 100x less energy.

Still much less with ASIC:
  Good points: asynchronous logic, arbitrary gates, etc..
  Drawback: hard to update!

Another 10 – 100 times improvement.
(100x is cheating:
  I was comparing one 28 nm ASIC
to one 45 nm FPGA)
Five Generations of Miners

4. ASIC Miners

CPU, GPU = about 4000 W / Gh/s
FPGA = about 50 W / Gh/s
ASIC = now down to 0.35 W / Gh/s

Overall we have improved the efficiency 10,000 times since Satoshi started mining in early 2009…
Like 1000% per year improvement.
Hash Rate - Doubled Nearly Every Month!

1000x in 1Y

Source: blockchain.info
Crypto Currencies

Recently Still +60% Every Month

Source: blockchain.info

Hash Rate GH/s 244,669,273

Nicolas T. Courtois 2009-2014
5. Quantum Miners?

Business Law:

Every technology improved by 30%, 67%, each year? why not 1000% ???

Nicolas T. Courtois 2009-2014
Crypto Currencies

Butterfly Labs and their angry customers

Promised 1 W per GH/s, delivered 3.2 W to customers
Better Miners: less nm

KNC vs. BitFury vs. Butterfly

52 20 nm  28 nm  65 nm
ASICS Comparison

By power / Gh/s

0.35 W  low power mode

1 W

cf. https://en.bitcoin.it/wiki/Mining_hardware_comparison
Criminal Scams

See [bitcoinscammers.com](http://bitcoinscammers.com)
Immoral Business Practice

I do not know a single company which is totally honest.

KNC and Cointerra has been the most honest IMHO, but worked mostly with pre-orders.

=> huge problem

Class Action Lawsuit: CoinTerra Seeks Out-of-Court Settlement

I'm preparing a class action lawsuit against @kncminer for failure to refund, non-response & I'm alleging fraud. Anyone want to be included?
Miners for Cash

Available since April 2014.
Quickly falling prices.

Before:
it was IMPOSSIBLE for miners to evaluate the profitability of their investments.

Waiting for 6 months is like getting…. 50 TIMES smaller return, like 2% of the original expected income for a miner…
Miners and Poland

Carlson/MegaBigPower.com has a Polish investor: a Poland-based scientific research center BioInfoBank

MegaBigPower.com also run a pool: 12 PH/s as of 9/2014, 100K$/day

THE MINE  megabigpower.com/themine

MegaBigPower’s mining operation is divided between locations in Poland and the United States. In the USA, we have nearly two petahash of bitcoin mining set up in eastern Washington State. The location was
New Miners

Cointerra Q1 2015:
4.5 TH/s, 1300 W, 2500 USD, 16nm, 14 M$ investment?

⇒ 0.29 W per Gh/s
Total Cost? About 1.0 Billion USD

Quick estimation of the cost of hardware as of April 2014:
Current hash rate 40,000 Th/s (April 2014)
Assume most people use Neptune first generation which costed 3500 USD for 0.25 Th/s of hash power (better devices exist frankly just in pre-orders, well for a majority of people).

So current hash rate might have costed 40,000 x 4 x 3,500 USD, so maybe 600 M dollars in hash equipment.

However probably most people still use miners NOT as good as Neptune, then probably this is 2 times more... So maybe it is already more than 1 billion today.

600 M / 100 K people = 6000 USD typical investment?
Bitcoin
And Hash Functions
Crypto Currencies

Mining Overview

hashed data from previous transactions

<table>
<thead>
<tr>
<th>ver=2</th>
<th>hashPrevBlock</th>
<th>hashMerkleRoot</th>
<th>timestamp</th>
<th>target</th>
<th>nonce</th>
<th>padding+len</th>
</tr>
</thead>
<tbody>
<tr>
<td>32 bits</td>
<td>256 bits</td>
<td>256 bits</td>
<td>32 bits</td>
<td>32 bits</td>
<td>32 bits</td>
<td>384 bits</td>
</tr>
</tbody>
</table>

Goal: find a valid pair (merkle_root, nonce) which gives 60 bits at 0 in H2

CISO Problem: Constrained Input Small Output
Crypto Currencies

Mining Internals

hashed data from previous transactions

ver=2
32 bits

hashPrevBlock
256 bits

time
32 bits

target
32 bits

nonce
32 bits

padding+len
384 bits

IV
256 bits

H0

H1
256 bits

padding+len
256 bits

000...
H2
256 bits

A_t, B_t, C_t, D_t, E_t, F_t, G_t, H_t

Ch() → ∑_t

Maj() → ∑_t

64R

256 bits

512 bits

512 bits

256 bits

256 bits
Bitcoin
Hash Functions
And Block Ciphers (♥️)
Crypto Currencies

SHA-256 Compression Function

\[ 256 \]

IV

\[ 512 \]

\[ M \]

message expansion

block cipher

Davies-Meyer

\( f(M, IV) \)

\[ 32 \times 64 \]

64 rounds

cf. Pieprzyk, Matusiewicz et al.
Fact:

The process of BitCoin Mining is no different than a brute force attack on a block cipher:

- Apply the same box many times, with different keys…
- Here the block cipher is a part of a hash function but it does NOT matter.
  - 98% of computational effort is evaluating this block cipher box with various keys and various inputs
  - Like a random oracle.
Davies-Meyer

Transforms a block cipher into a hash function. In SHA-256 we have:
block size=256, 64 rounds, key size=256 expanded 4x.
***One Round of SHA-256

cf. Pieprzyk, Matusiewicz et al.
Optimising Mining
(39% gain w.r.t. best ASIC)
Like Generation 4.1.
Hashing Block of 300+ Bits

cf. Pieprzyk, Matusiewicz et al.
Crypto Currencies

Hashing Block of 300+ Bits

cf. Pieprzyk, Matusiewicz et al.
Padding

Message to be hashed:
- 640 bits

Padding + len:
- 384 bits

IV:
- 256 bits

H0:
- 256 bits

H1:
- 256 bits
<table>
<thead>
<tr>
<th>Field</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>version</td>
<td>32 bits</td>
<td>Version of the Bitcoin software version creating this block</td>
</tr>
<tr>
<td>hashPrevBlock</td>
<td>256 bits</td>
<td>Hash of the previous block considered as valid in the Bitcoin network (most of the time there is only one candidate)</td>
</tr>
<tr>
<td>hashMerkleRoot</td>
<td>256 bits</td>
<td>Here a set of recent yet unconfirmed Bitcoin transactions are hashed into one single value on 256 bits = the Merkle Root</td>
</tr>
<tr>
<td>timestamp</td>
<td>32 bits</td>
<td>Current timestamp in seconds since 1970-01-01 00:00 UTC</td>
</tr>
<tr>
<td>target</td>
<td>32 bits</td>
<td>The current Target represented in a compact 32 bit format</td>
</tr>
<tr>
<td>nonce</td>
<td>32 bits</td>
<td>Nonce chosen by the miner, typically goes from 0x00000000 to 0xFFFFFFFF until the CISO puzzle is solved</td>
</tr>
<tr>
<td>padding + len</td>
<td>384 bits</td>
<td>standard fixed SHA256 padding on 384 bits for Len=640 bits</td>
</tr>
</tbody>
</table>
Crypto Currencies

Hashing Internals

hashed data from previous transactions

ver=2
32 bits

hashPrevBlock
256 bits

hashMerkleRoot
256 bits

timest.
32 bits

target
32 bits

nonce
32 bits

padding+len
384 bits

IV
256 bits

512 bits

H0

H1
256 bits

padding+len
256 bits

000...
H2
256 bits
Improvement 1 – Amortized Cost(H0)=0

- **ver=2**: 32 bits
- **hashPrevBlock**: 256 bits
- **hashMerkleRoot**: 256 bits
- **timest.**: 32 bits
- **target**: 32 bits
- **nonce**: 32 bits
- **padding+len**: 384 bits

- **IV**: 256 bits
  - 64R
- **H0**: 256 bits
- **H1**: 256 bits
- **padding+len**: 256 bits

- **IV**: 256 bits
  - 64R

- **H2**: 256 bits
  - 000...
## Improvement 2 – Gains 3 Rounds At the End

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>t=59:</td>
<td>B6AE8FFF</td>
<td>FFBE70472</td>
<td>C062D46F</td>
<td>FCD18878</td>
<td>B21BADD3D</td>
<td>6D83BFC6</td>
<td>7E44008E</td>
</tr>
<tr>
<td>t=60:</td>
<td>B85E2CE9</td>
<td>B6AE8FFF</td>
<td>FFBE70472</td>
<td>C062D46F</td>
<td>961F4894</td>
<td>B21BADD3D</td>
<td>6D83BFC6</td>
</tr>
<tr>
<td>t=61:</td>
<td>04D24D6C</td>
<td>B85E2CE9</td>
<td>B6AE8FFF</td>
<td>FFBE70472</td>
<td>948D25B6</td>
<td>961F4894</td>
<td>B21BADD3D</td>
</tr>
<tr>
<td>t=62:</td>
<td>D39A2165</td>
<td>04D24D6C</td>
<td>B85E2CE9</td>
<td>B6AE8FFF</td>
<td>FB121210</td>
<td>948D25B6</td>
<td>961F4894</td>
</tr>
<tr>
<td>t=63:</td>
<td>506E3058</td>
<td>D39A2165</td>
<td>04D24D6C</td>
<td>B85E2CE9</td>
<td>5E5F0F24</td>
<td>FB121210</td>
<td>948D25B6</td>
</tr>
</tbody>
</table>
**Improvement 3**

Gains 3 Rounds At the Beginning

they do NOT depend on the nonce
### Improvement 4

**Incremental Computation**

<table>
<thead>
<tr>
<th>Round t</th>
<th>32 bit W₁</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>XXXXXXXX</td>
<td>last 32 Bits of hashMerkleRoot</td>
</tr>
<tr>
<td>1</td>
<td>XXXXXXXX</td>
<td>timestamp</td>
</tr>
<tr>
<td>2</td>
<td>XXXXXXXX</td>
<td>target</td>
</tr>
<tr>
<td>3</td>
<td>XXXXXXXX</td>
<td>nonce (00000000 to FFFFFFFF)</td>
</tr>
<tr>
<td>4</td>
<td>0x80000000</td>
<td>padding starts</td>
</tr>
<tr>
<td>5</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0x00000000</td>
<td>padding ends</td>
</tr>
<tr>
<td>14</td>
<td>0x00000000</td>
<td>length H</td>
</tr>
<tr>
<td>15</td>
<td>0x000000280</td>
<td>length L</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Round t</th>
<th>32 bit W₁</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>XXXXXXXX</td>
<td>H₁₀</td>
</tr>
<tr>
<td>1</td>
<td>XXXXXXXX</td>
<td>H₁₁</td>
</tr>
<tr>
<td>2</td>
<td>XXXXXXXX</td>
<td>H₁₂</td>
</tr>
<tr>
<td>3</td>
<td>XXXXXXXX</td>
<td>H₁₃</td>
</tr>
<tr>
<td>4</td>
<td>XXXXXXXX</td>
<td>H₁₄</td>
</tr>
<tr>
<td>5</td>
<td>XXXXXXXX</td>
<td>H₁₅</td>
</tr>
<tr>
<td>6</td>
<td>XXXXXXXX</td>
<td>H₁₆</td>
</tr>
<tr>
<td>7</td>
<td>XXXXXXXX</td>
<td>H₁₇</td>
</tr>
<tr>
<td>8</td>
<td>0x80000000</td>
<td>Padding Starts</td>
</tr>
<tr>
<td>9</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0x00000000</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0x00000000</td>
<td>Padding Ends</td>
</tr>
<tr>
<td>14</td>
<td>0x00000000</td>
<td>length H</td>
</tr>
<tr>
<td>15</td>
<td>0x000000100</td>
<td>length L</td>
</tr>
</tbody>
</table>
Crypto Currencies

**Improvement 4 - contd**

- **Incremental Computation**

  2 increments instead of 200 gates.
### Improvement 5

- Gains
- 18 Additions
- \( \approx 3600 \) gates

<table>
<thead>
<tr>
<th>Round t</th>
<th>32 bit ( W_t )</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>XXXXXXXX</td>
<td>last 32 Bits of hashMerkleRoot</td>
</tr>
<tr>
<td>1</td>
<td>XXXXXXXX</td>
<td>timestamp</td>
</tr>
<tr>
<td>2</td>
<td>XXXXXXXX</td>
<td>target</td>
</tr>
<tr>
<td>3</td>
<td>XXXXXXXX</td>
<td>nonce (00000000 to FFFFFFFF)</td>
</tr>
<tr>
<td>4</td>
<td>0x000000000</td>
<td>padding starts</td>
</tr>
<tr>
<td>5</td>
<td>0x000000000</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0x0000000000</td>
<td>padding ends</td>
</tr>
<tr>
<td>14</td>
<td>0x0000000000</td>
<td>length ( H )</td>
</tr>
<tr>
<td>15</td>
<td>0x0000000280</td>
<td>length ( L )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Round t</th>
<th>32 bit ( W_t )</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>XXXXXXXX</td>
<td>( H_1^0 )</td>
</tr>
<tr>
<td>1</td>
<td>XXXXXXXX</td>
<td>( H_1^1 )</td>
</tr>
<tr>
<td>2</td>
<td>XXXXXXXX</td>
<td>( H_1^2 )</td>
</tr>
<tr>
<td>3</td>
<td>XXXXXXXX</td>
<td>( H_1^3 )</td>
</tr>
<tr>
<td>4</td>
<td>XXXXXXXX</td>
<td>( H_1^4 )</td>
</tr>
<tr>
<td>5</td>
<td>XXXXXXXX</td>
<td>( H_1^5 )</td>
</tr>
<tr>
<td>6</td>
<td>XXXXXXXX</td>
<td>( H_1^6 )</td>
</tr>
<tr>
<td>7</td>
<td>XXXXXXXX</td>
<td>( H_1^7 )</td>
</tr>
<tr>
<td>8</td>
<td>0x000000000</td>
<td>Padding Starts</td>
</tr>
<tr>
<td>9</td>
<td>0x000000000</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0x0000000000</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0x0000000000</td>
<td>Padding Ends</td>
</tr>
<tr>
<td>14</td>
<td>0x0000000000</td>
<td>length ( H )</td>
</tr>
<tr>
<td>15</td>
<td>0x0000000100</td>
<td>length ( L )</td>
</tr>
</tbody>
</table>
**Improvement 6**

**Saving**

2 More Additions

≈ 400 gates

with Hard Coding

AND SAVE LIKE HALF of the next addition!

(addition with a constant = cheaper, depends on the constant)
Improvement X

Classical trick: Carry Save Adders.

\[ a + b + c \]

Cost = 2 adders

\[ a + b + c \]

Cost = \(1 + \varepsilon\) adders

Nicolas T. Courtois 2009-2014
Whole Round

Only two full adders.
For $0 \leq t \leq 15,$

$$W_t = M_t$$

For $16 \leq t \leq 63,$

$$W_t = \sigma_1(W_{t-2}) \oplus W_{t-7} \oplus \sigma_0(W_{t-15}) \oplus W_{t-16}$$

$\Rightarrow$ just copy for 16 R

non-trivial part

$$\sigma_0(x) = \text{ROTR}^7(x) \oplus \text{ROTR}^{18}(x) \oplus \text{SHR}^3(x)$$

$$\sigma_1(x) = \text{ROTR}^{17}(x) \oplus \text{ROTR}^{19}(x) \oplus \text{SHR}^{10}(x)$$
\[ \sigma_0(x) = \text{ROTR}^7(x) \oplus \text{ROTR}^{18}(x) \oplus \text{SHR}^3(x) \]

\[ \sigma_3(x) = \text{ROTR}^{17}(x) \oplus \text{ROTR}^{19}(x) \oplus \text{SHR}^{19}(x) \]

\[ \text{Ch}(X, Y, Z) = (X \land Y) \oplus (\neg X \land Z) \]

\[ \text{Maj}(X, Y, Z) = (X \land Y) \oplus (X \land Z) \oplus (Y \land Z) \]

\[ \Sigma_0(x) = \text{ROTR}^4(x) \oplus \text{ROTR}^{13}(x) \oplus \text{ROTR}^{13}(x) \]

\[ \Sigma_1(x) = \text{ROTR}^6(x) \oplus \text{ROTR}^{21}(x) \oplus \text{ROTR}^{21}(x) \]

Message Schedule
Improvement 7 - Fact:

Some early values do NOT yet depend on the nonce. In H1 computation only (left column).

\[ W_{16} = \sigma_1(W_{14}) \boxplus W_9 \boxplus \sigma_0(W_1) \boxplus W_0 \]

\[ W_{17} = \sigma_1(W_{15}) \boxplus W_{10} \boxplus \sigma_0(W_2) \boxplus W_1 \]
Improvement 7 – 3 more

2 more 32-bit additions are saved by hard coding,
and more for the next addition
(again, adding a constant, depends on the constant, average cost maybe saving another 1? addition).

Some 600 extra gates saved.
Improvement 8 – 1 More Incremental

We have:

\[ W_{19} = \sigma_1(W_{17}) \oplus W_{11} \oplus \sigma_0(W_4) \oplus W_3. \]

\( W_3 \) is the nonce which is incremented by 1.

| \( W_0 \) | 0xffffffff | 0xffffffff | 0xffffffff | 0xffffffff | 0xffffffff |
| \( W_1 \) | 0xffffffff | 0xffffffff | 0xffffffff | 0xffffffff | 0xffffffff |
| \( W_2 \) | 0xffffffff | 0xffffffff | 0xffffffff | 0xffffffff | 0xffffffff |
| \( W_3 \) | 0x00000000 | 0x000000001 | 0x000000002 | 0x000000003 | 0x000000004 |
| \( W_{19} \) | 0x1108b759 | 0x1108b75a | 0x1108b75b | 0x1108b75c | 0x1108b75d |

Table 9: Code Execution Results for \( W_{19} \) with Different Nonces
Improvement X2

Also use Carry Save Adders in message scheduling. Only 1 full adder in each of (only) 48-3 values which need still to be computed.
Fact 12.1 (Hash Speed). The amortized average cost of trying one output $H_2$ to see if it is likely to have 60 or more leading zeros is at most about 1.89 computations of the compression function of SHA-256 instead of 3.0, which represents an improvement by 39%.
Future – Dan Kaminsky
Earlier he said that he has no stakes in ‘this game’. Then at minute 40 he claims that the current Bitcoin Proof of Work function based on SHA-256 will not survive “the year” (to be replaced before end of 2013). He says that assigns zero percent probability that “we” will continue with the present POW function”. Back to CPU mining.

https://www.youtube.com/watch?v=si-2niFDgtI
Crypto Currencies

SHA-256 to be phased out?

https://www.youtube.com/watch?v=si-2niFDgtd

HOWEVER

[claimed by Courtois just afterwards]:

NOBODY OWNS BITCOIN

We claim the contrary: any attempt to change the POW is close to impossible to enforce

I WAS RIGHT, it has NOT been changed.
Too much money at stake.