Mining Miscellanea

Saravanan Vijayakumaran

Department of Electrical Engineering
Indian Institute of Technology Bombay

January 30, 2026

Finding and Distributing Mining Nonces

Bitcoin Mining

Block Header =

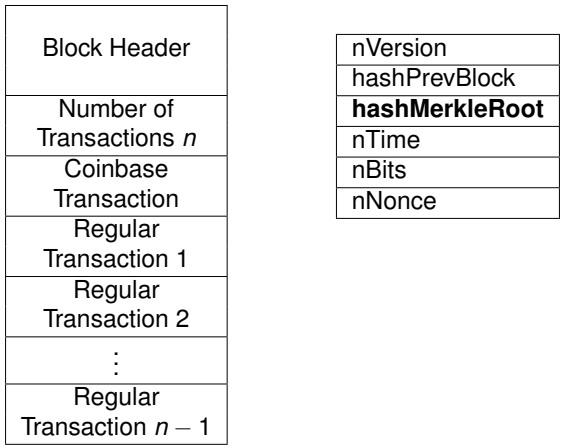
nVersion	4 bytes
hashPrevBlock	32 bytes
hashMerkleRoot	32 bytes
nTime	4 bytes
nBits	4 bytes
nNonce	4 bytes

- A $3000 mining rig can perform 200 TH/s
- A 4-byte nNonce field means $2^{32} \approx 4 \times 10^9$ possibilities
- **What should a miner do if all the 2^{32} nNonce values fail threshold test?**
  - Changing hashPrevBlock and nBits fields invalidates block
  - Change bits in the nVersion field?
  - Change timestamp to change nTime field?
  - Change transactions to change hashMerkleRoot field?

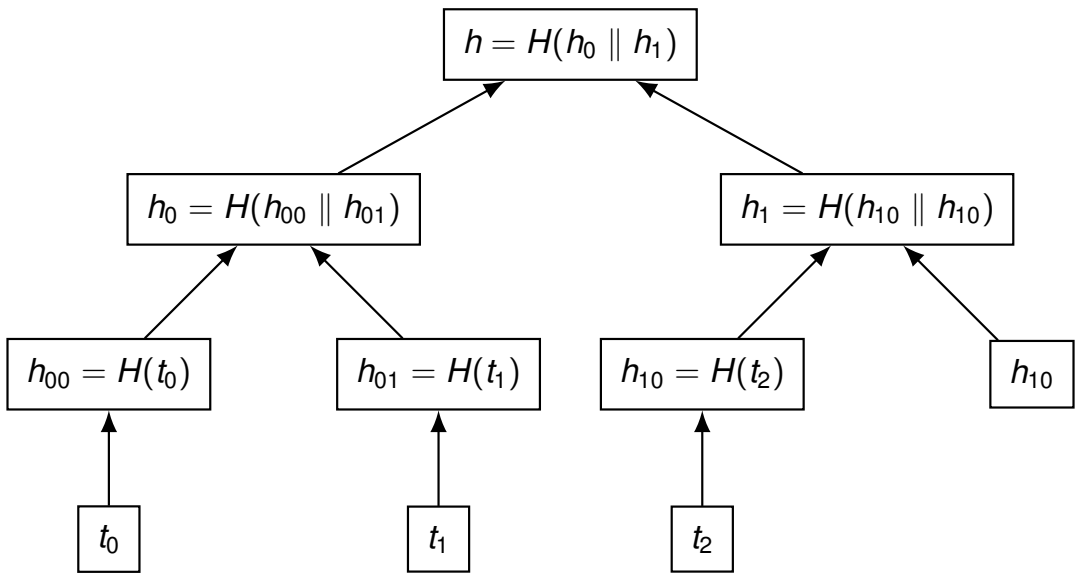
Modifying nVersion and nTime

- nVersion
  - Three bits of the 32-bit nVersion are set to 001
  - Remaining 29 bits are used by miners to signal support for soft forks
  - Changing the signaling bits can interfere with protocol upgrades
  - Some miners still do it (see block 541,604)
- nTime
  - Timestamps can be changed only by increments of a second
  - In block at height N, the nTime value needs to be greater than median of nTime values of blocks $N-1, N-2, \ldots, N-11$
  - A node rejects a block if the nTime field specifies a time which exceeds its network-adjusted time by more than 2 hours
  - Miners cannot risk invalidating their mined blocks by modifying nTime indiscriminately

Transaction Merkle Root

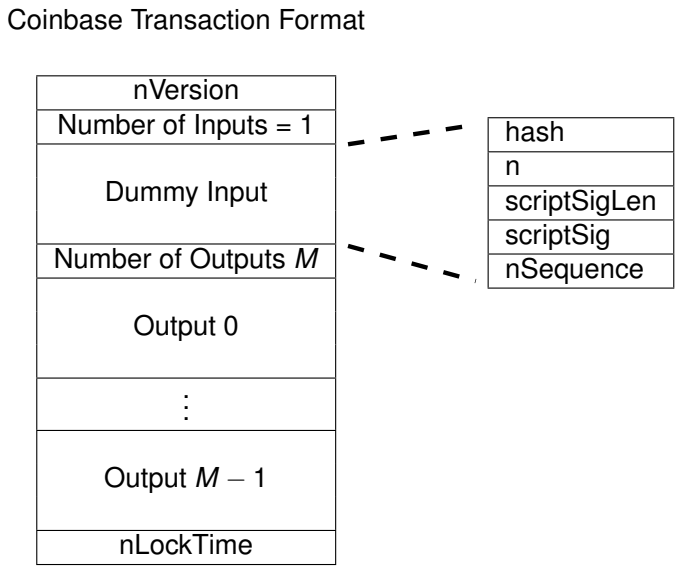


- hashMerkleRoot contains root hash of transaction Merkle tree
- Modifying any transaction or the transaction order will modify the root hash



The Extra Nonce Solution

- Although coinbase transaction do not unlock previous outputs, they contain a dummy input



- Dummy input fields
  - hash is set to all zeros (0x000...000)
  - n is set to 0xFFFFFFFF
  - scriptSig field can be at most 100 bytes long; also called coinbase field
  - Since March 2013, the first 4 bytes of scriptSig encode the block height
  - The remaining scriptSig space is used as an **extra nonce** by miners

Genesis Block Coinbase Field

- Satoshi put the following text in the genesis block coinbase field

  The Times 03/Jan/2009 Chancellor on brink of second bailout for banks

THE TIMES

Max 5C, min -5C | Saturday January 3 2009 timesonline.co.uk No 69523 | £1.50

Eat Out from £5

More than 900 great restaurants, including four **Gordon Ramsay** favourites from £15

Start collecting tokens today. Pullout Inside

Israel prepares to send tanks and troops into Gaza

Israel allowed foreigners to flee the Gaza Strip as it prepared for a ground offensive. At least 430 Palestinians were killed in a week of airstrikes News, page 3

Chancellor on brink of second bailout for banks

Billions may be needed as lending squeeze tightens

Francis Elliott Deputy Political Editor
Gary Duncan Economics Editor

Alistair Darling has been forced to consider a second bailout for banks as the lending drought worsens.

The Chancellor will decide within weeks whether to pump billions more into the economy as evidence mounts that the £37billion part-nationalisation last year has failed to keep credit flowing. Options include cash injections, offering banks cheaper state guarantees to raise money privately or buying up "toxic assets", The Times has learnt.

The Bank of England revealed yesterday that, despite intense pressure, the banks curbed lending in the final quarter of last year and plan even tighter restrictions in the coming months. Its findings will alarm the Treasury.

The Bank is expected to take yet more aggressive action this week by cutting the base rate from its current level of 2 per cent. Doing so would reduce the cost of borrowing but have little effect on the availability of loans.

Whitehall sources said that ministers planned to "keep the banks on the boil" but accepted that they need more help to restore lending levels. Formally, the Treasury plans to focus on state-backed gurantees to encourage private finance, but a number of interventions are on the table, including further injections of taxpayers' cash.

Under one option, a "bad bank" would be created to dispose of bad debts. The Treasury would take bad loans off the hands of troubled banks, perhaps swapping them for government bonds. The toxic assets, blamed for poisoning the financial system, would be parked in a state vehicle or "bad bank" that would manage them and attempt to dispose of them while "detoxifying" the mainstream banking system.

The idea would mirror the initial proposal by Henry Paulson, the US Treasury Secretary, to underpin the American banking system by buying

Continued on page 6, col 1
Leading article, page 2

99p

Pub chain cuts the price of a pint from £1.69 to 1989 levels
Business, page 47

Michael Sheen Frost, Nixon and me
Magazine

Working mums So that's how she does it
Body&Soul

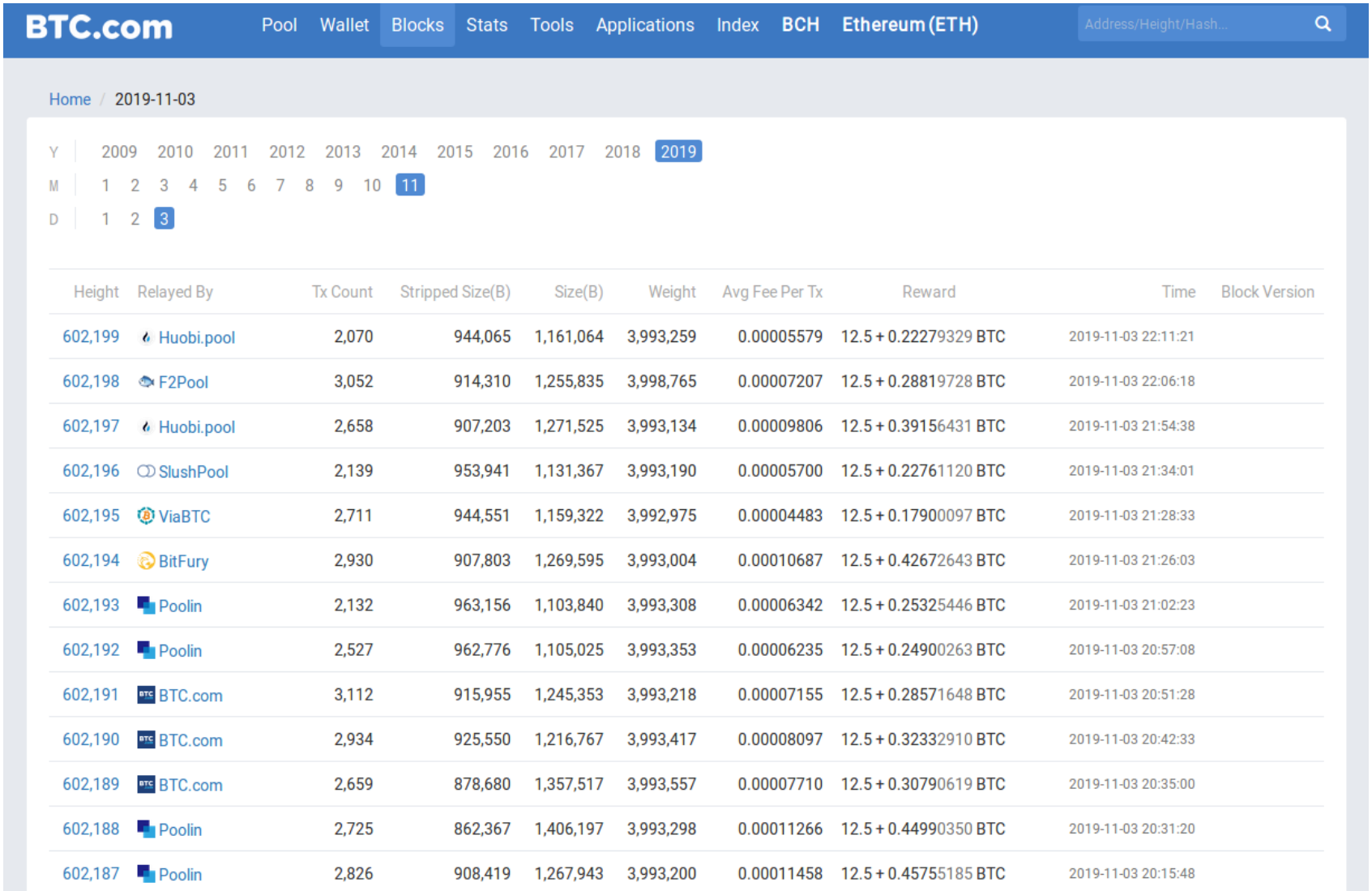
Detox in style The best spas on the planet
Travel

Salmon Rushdie I Won't Marry Again
Pages 22, 23

Giant Killing? Guide to the FA Cup Third Round
Sport

Coinbase Markers

- Miners identify themselves in the coinbase field



Height	Relayed By	Tx Count	Stripped Size(B)	Size(B)	Weight	Avg Fee Per Tx	Reward	Time	Block Version
602,199	Huobi.pool	2,070	944,065	1,161,064	3,993,259	0.00005579	12.5 + 0.22279329 BTC	2019-11-03 22:11:21	
602,198	F2Pool	3,052	914,310	1,255,835	3,998,765	0.00007207	12.5 + 0.28819728 BTC	2019-11-03 22:06:18	
602,197	Huobi.pool	2,658	907,203	1,271,525	3,993,134	0.00009806	12.5 + 0.39156431 BTC	2019-11-03 21:54:38	
602,196	SlushPool	2,139	953,941	1,131,367	3,993,190	0.00005700	12.5 + 0.22761120 BTC	2019-11-03 21:34:01	
602,195	ViaBTC	2,711	944,551	1,159,322	3,992,975	0.00004483	12.5 + 0.17900097 BTC	2019-11-03 21:28:33	
602,194	BitFury	2,930	907,803	1,269,595	3,993,004	0.00010687	12.5 + 0.42672643 BTC	2019-11-03 21:26:03	
602,193	Poolin	2,132	963,156	1,103,840	3,993,308	0.00006342	12.5 + 0.25325446 BTC	2019-11-03 21:02:23	
602,192	Poolin	2,527	962,776	1,105,025	3,993,353	0.00006235	12.5 + 0.24900263 BTC	2019-11-03 20:57:08	
602,191	BTC.com	3,112	915,955	1,245,353	3,993,218	0.00007155	12.5 + 0.28571648 BTC	2019-11-03 20:51:28	
602,190	BTC.com	2,934	925,550	1,216,767	3,993,417	0.00008097	12.5 + 0.32332910 BTC	2019-11-03 20:42:33	
602,189	BTC.com	2,659	878,680	1,357,517	3,993,557	0.00007710	12.5 + 0.30790619 BTC	2019-11-03 20:35:00	
602,188	Poolin	2,725	862,367	1,406,197	3,993,298	0.00011266	12.5 + 0.44990350 BTC	2019-11-03 20:31:20	
602,187	Poolin	2,826	908,419	1,267,943	3,993,200	0.00011458	12.5 + 0.45755185 BTC	2019-11-03 20:15:48	

Source: `https://explorer.btc.com/btc/blocks`

Block Distribution

- The percentage of blocks mined by each miner can be calculated from coinbase markers

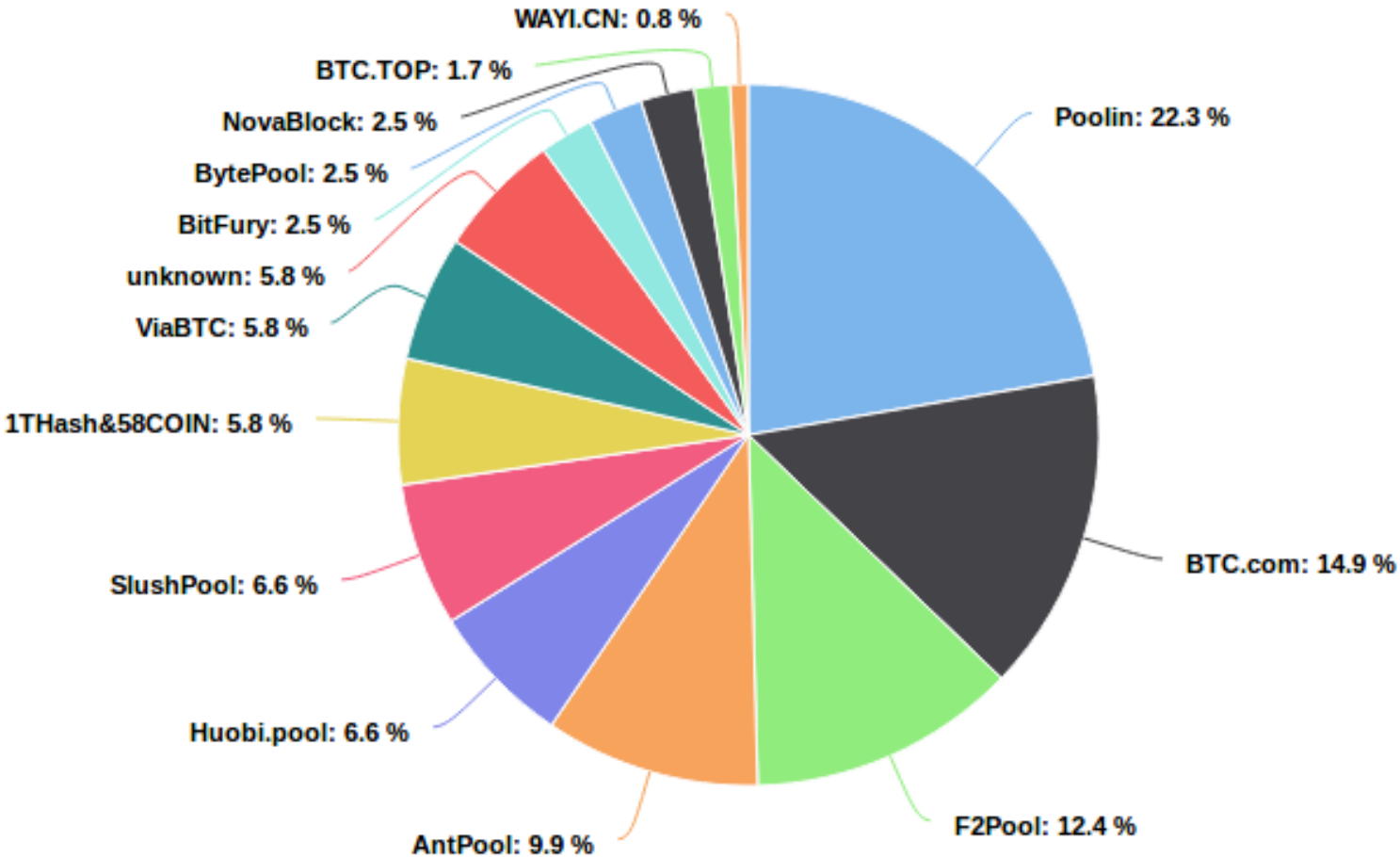


Image credit: `https://explorer.btc.com/btc/insights-pools`

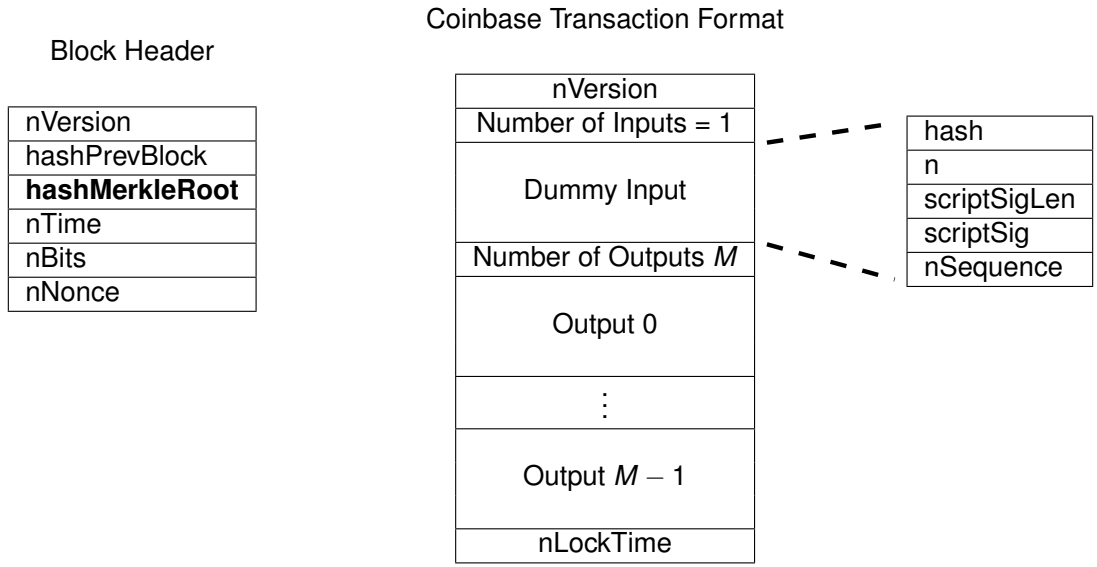
Mining Pools

- The network hashrate is 1000 Exahashes/s = 1000×10^{18} hashes/s
- A $4000 mining rig can perform 200 TH/s
- The probability of an individual rig owner winning a block is low
- Rig owners join mining pools
- Mining pool operation
  - Pool owner “distributes” the mining search space among the pool miners (participants)
  - When a pool miner finds a hash starting with 32 zeros, it submits the block header to the pool as proof of its efforts. This is called a **share**.
  - If one of the pool miners finds a valid block, the block reward is distributed to all pool miners proportional to the number of submitted shares
  - Pool takes a portion of the block reward as coordination fee

Distributing Search Space

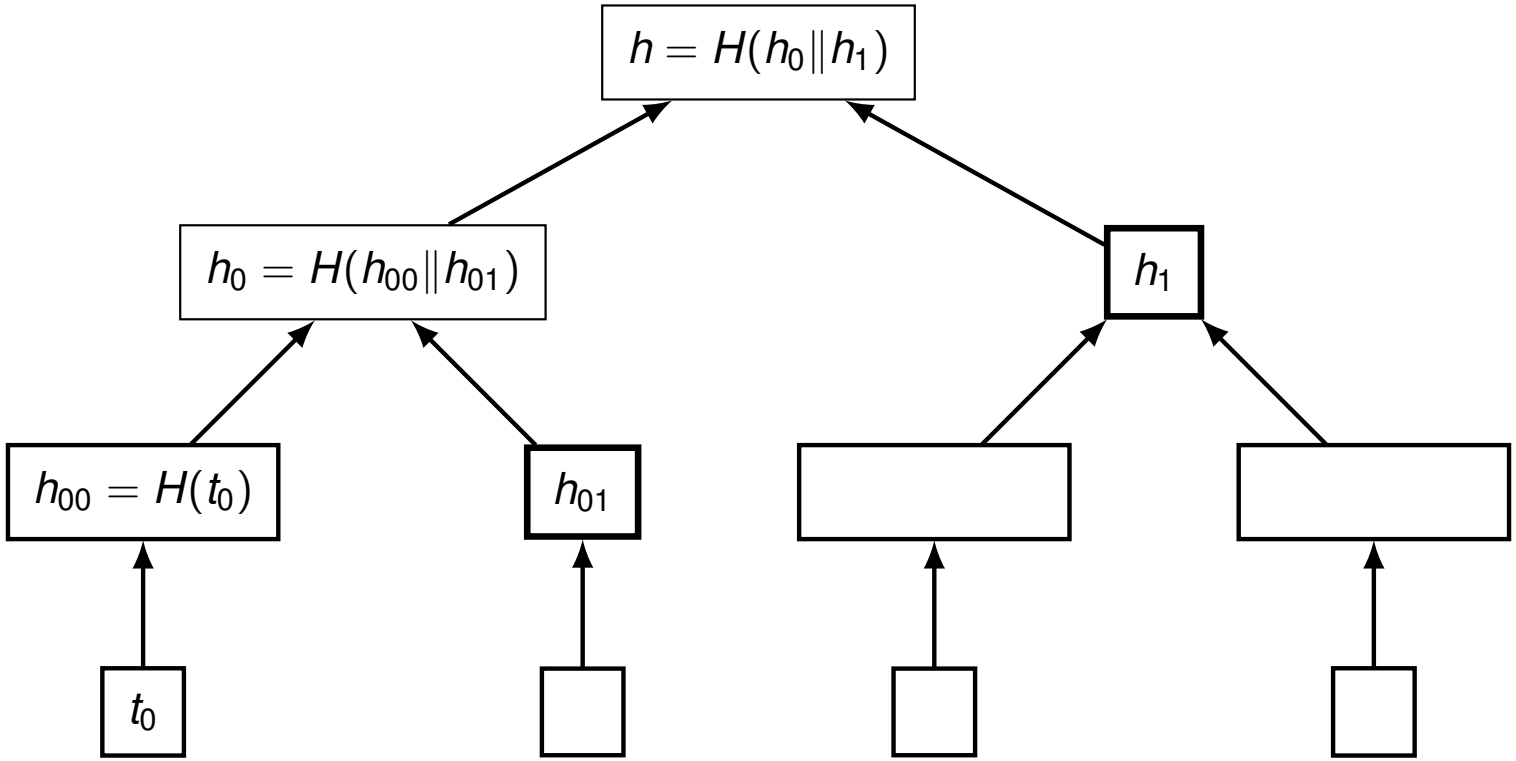
- Pool owner can distribute search space by having a different extra nonce for each pool miner
- Rolling of extra nonce by pool owner for every pool miner does not scale
  - Pool owner recomputes hashMerkleRoot for every extra nonce change
  - Pool miners only change nNonce and nTime (assuming nVersion is not changed)
- Instead, extra nonce is split into two parts
  - ExtraNonce1 is used to distribute search space
  - ExtraNonce2 is changed by the individual pool miners

Transaction Merkle Root



- Pool owner sends each pool miner the following
  - nVersion, hashPrevBlock, nTime, nBits fields of block header
  - Coinbase1 = Part of the coinbase transaction before extra nonce
  - ExtraNonce1 = Miner-specific extra nonce
  - ExtraNonce2_size = The number of bytes in ExtraNonce2 the miner can change
  - Coinbase2 = Part of the coinbase transaction after extra nonce
  - Merkle_branch = List of hashes used to calculate hashMerkleRoot

Merkle Branch



- Every time ExtraNonce2 is changed, the hashMerkleRoot has to be recalculated
- Instead of sending all the transactions, only necessary hashes are sent

AsicBoost

SHA-256

- SHA = Secure Hash Algorithm, 256-bit output length
- Accepts bit strings of length upto $2^{64} - 1$
- Output calculation has two stages
  - Preprocessing
  - Hash Computation
- Preprocessing
  1. A 256-bit state variable $H^{(0)}$ is set to

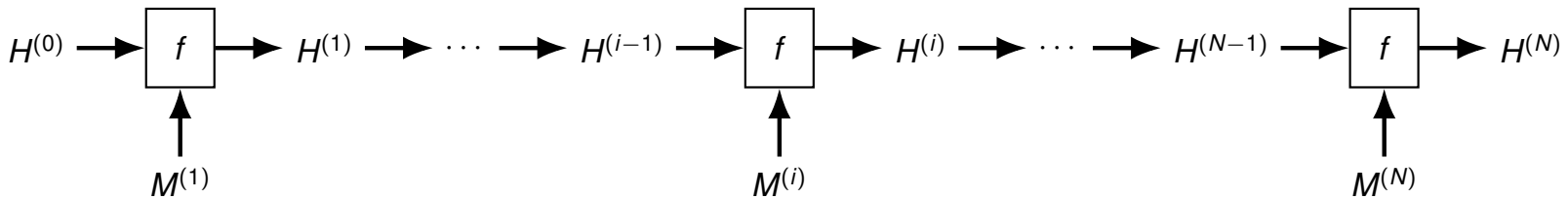
$$H_0^{(0)} = \texttt{0x6A09E667}, \quad H_1^{(0)} = \texttt{0xBB67AE85},$$
$$H_2^{(0)} = \texttt{0x3C6EF372}, \quad H_3^{(0)} = \texttt{0xA54FF53A},$$
$$H_4^{(0)} = \texttt{0x510E527F}, \quad H_5^{(0)} = \texttt{0x9B05688C},$$
$$H_6^{(0)} = \texttt{0x1F83D9AB}, \quad H_7^{(0)} = \texttt{0x5BE0CD19}.$$

  2. The input M is padded to a length which is a multiple of 512

SHA-256 Hash Computation

1. Padded input is split into N 512-bit blocks $M^{(1)}, M^{(2)}, \ldots, M^{(N)}$
2. Given $H^{(i-1)}$, the next $H^{(i)}$ is calculated using a function f

$$H^{(i)} = f(M^{(i)}, H^{(i-1)}), \quad 1 \leq i \leq N.$$



3. f is called a *compression function*
4. $H^{(N)}$ is the output of SHA-256 for input M

SHA-256 Compression Function Building Blocks

- U, V, W are 32-bit words
- $U \wedge V$, $U \vee V$, $U \oplus V$ denote bitwise AND, OR, XOR
- $U + V$ denotes integer sum modulo 2^{32}
- $\neg U$ denotes bitwise complement
- For $1 \leq n \leq 32$, the shift right and rotate right operations

$$\mathrm{SHR}^n(U) = \underbrace{000\cdots000}_{n \text{ zeros}} u_0 u_1 \cdots u_{30-n} u_{31-n},$$

$$\mathrm{ROTR}^n(U) = u_{31-n+1} u_{31-n+2} \cdots u_{30} u_{31} u_0 u_1 \cdots u_{30-n} u_{31-n},$$

- Bitwise choice and majority functions

$$\mathrm{Ch}(U, V, W) = (U \wedge V) \oplus (\neg U \wedge W),$$
$$\mathrm{Maj}(U, V, W) = (U \wedge V) \oplus (U \wedge W) \oplus (V \wedge W),$$

- Let

$$\Sigma_0(U) = \mathrm{ROTR}^2(U) \oplus \mathrm{ROTR}^{13}(U) \oplus \mathrm{ROTR}^{22}(U)$$
$$\Sigma_1(U) = \mathrm{ROTR}^6(U) \oplus \mathrm{ROTR}^{11}(U) \oplus \mathrm{ROTR}^{25}(U)$$
$$\sigma_0(U) = \mathrm{ROTR}^7(U) \oplus \mathrm{ROTR}^{18}(U) \oplus \mathrm{SHR}^3(U)$$
$$\sigma_1(U) = \mathrm{ROTR}^{17}(U) \oplus \mathrm{ROTR}^{19}(U) \oplus \mathrm{SHR}^{10}(U)$$

SHA-256 Compression Function Calculation

- Maintains internal state of 64 32-bit words $\{W_j \mid j = 0, 1, \ldots, 63\}$
- Also uses 64 constant 32-bit words $K_0, K_1, \ldots, K_{63}$ derived from the first 64 prime numbers $2, 3, 5, \ldots, 307, 311$
- $f(M^{(i)}, H^{(i-1)})$ proceeds as follows
  1. Internal state initialization

$$W_j = \begin{cases} M_j^{(i)} & 0 \leq j \leq 15, \\ \sigma_1(W_{j-2}) + W_{j-7} + \sigma_0(W_{j-15}) + W_{j-16} & 16 \leq j \leq 63. \end{cases}$$

  2. Initialize eight 32-bit words

$$(A, B, C, D, E, F, G, H) = \left(H_0^{(i-1)}, H_1^{(i-1)}, \ldots, H_6^{(i-1)}, H_7^{(i-1)}\right).$$

  3. For $j = 0, 1, \ldots, 63$, iteratively update $A, B, \ldots, H$

$$\begin{aligned} T_1 &= H + \Sigma_1(E) + \mathrm{Ch}(E, F, G) + K_j + W_j \\ T_2 &= \Sigma_0(A) + \mathrm{Maj}(A, B, C) \\ (A, B, C, D, E, F, G, H) &= (T_1 + T_2, A, B, C, D + T_1, E, F, G) \end{aligned}$$

  4. Calculate $H^{(i)}$ from $H^{(i-1)}$

$$(H_0^{(i)}, H_1^{(i)}, \ldots, H_7^{(i)}) = \left(A + H_0^{(i-1)}, B + H_1^{(i-1)}, \ldots, H + H_7^{(i-1)}\right).$$

AsicBoost

- A method to speedup Bitcoin mining by a factor of 20%
- Proposed by Timo Hanke and Sergio Demian Lerner
- Exploits the fact that SHA256 operates on 64 byte chunks
- The Bitcoin block header is 80 bytes long

<table>
<tr><td colspan="3">Chunk 1</td><td colspan="5">Chunk 2</td></tr>
<tr><td colspan="7">Block header</td><td rowspan="3">Padding</td></tr>
<tr><td colspan="6">Block header candidate</td><td rowspan="3">Nonce</td></tr>
<tr><td rowspan="2">Version</td><td rowspan="2">Previous hash</td><td colspan="2">Merkle root</td><td rowspan="2">Time stamp</td><td rowspan="2">Bits (difficulty)</td></tr>
<tr><td>Head</td><td>Tail</td></tr>
<tr><td>4 bytes</td><td>32 bytes</td><td>28 bytes</td><td>4 bytes</td><td>4 bytes</td><td>4 bytes</td><td>4 bytes</td><td>48 bytes</td></tr>
<tr><td colspan="3"></td><td colspan="3">Message[2]</td><td></td></tr>
</table>

Image source: `https://arxiv.org/abs/1604.00575`

- If two transaction Merkle roots collide in the last 4 bytes, some of the SHA-256 work in the second chunk can be reused
- Recall that the internal state initialization (W_j calculation) does not depend on the previous hash $H^{(i-1)}$

Bitcoin Header Double SHA256 Calculation

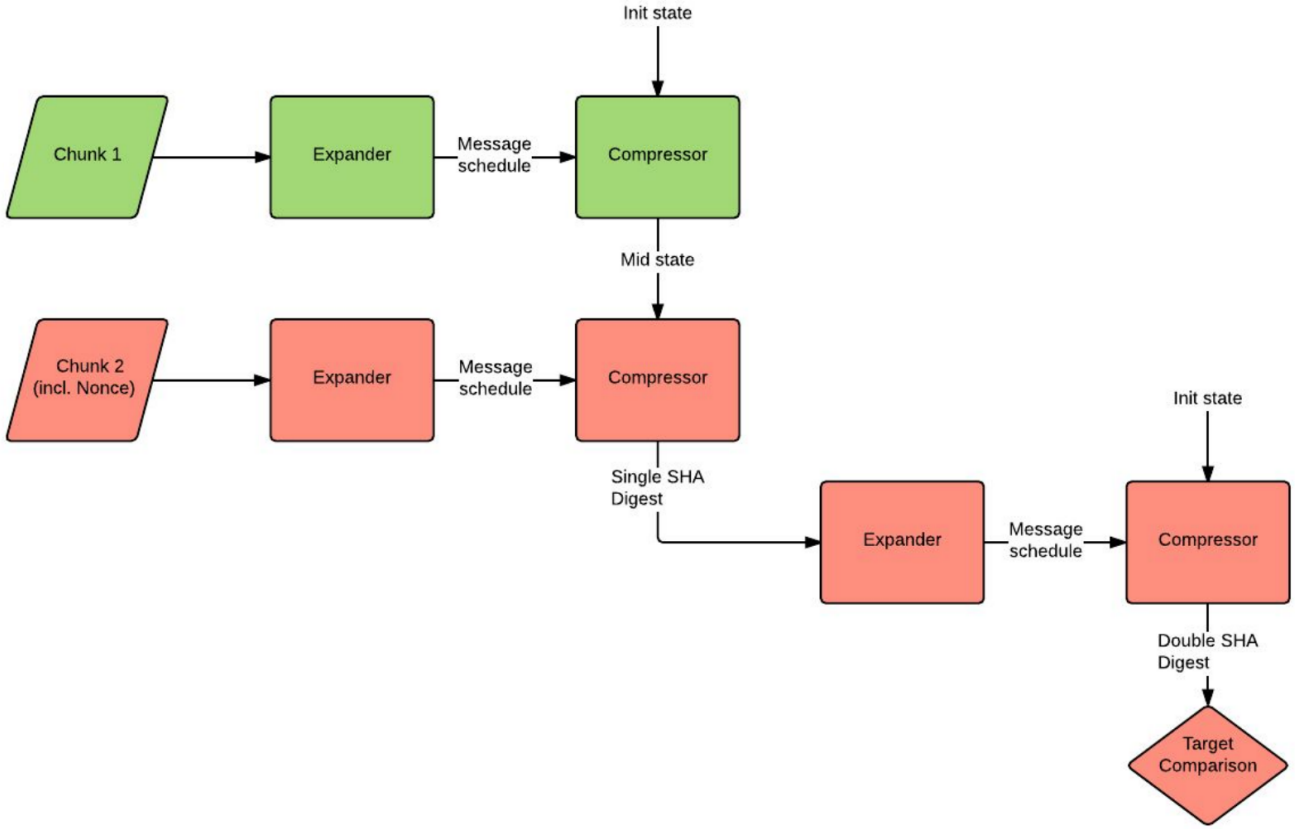


Image source: `https://arxiv.org/abs/1604.00575`

- In the above figure, the green blocks represent computation that can be reused

Bitcoin Mining Loop

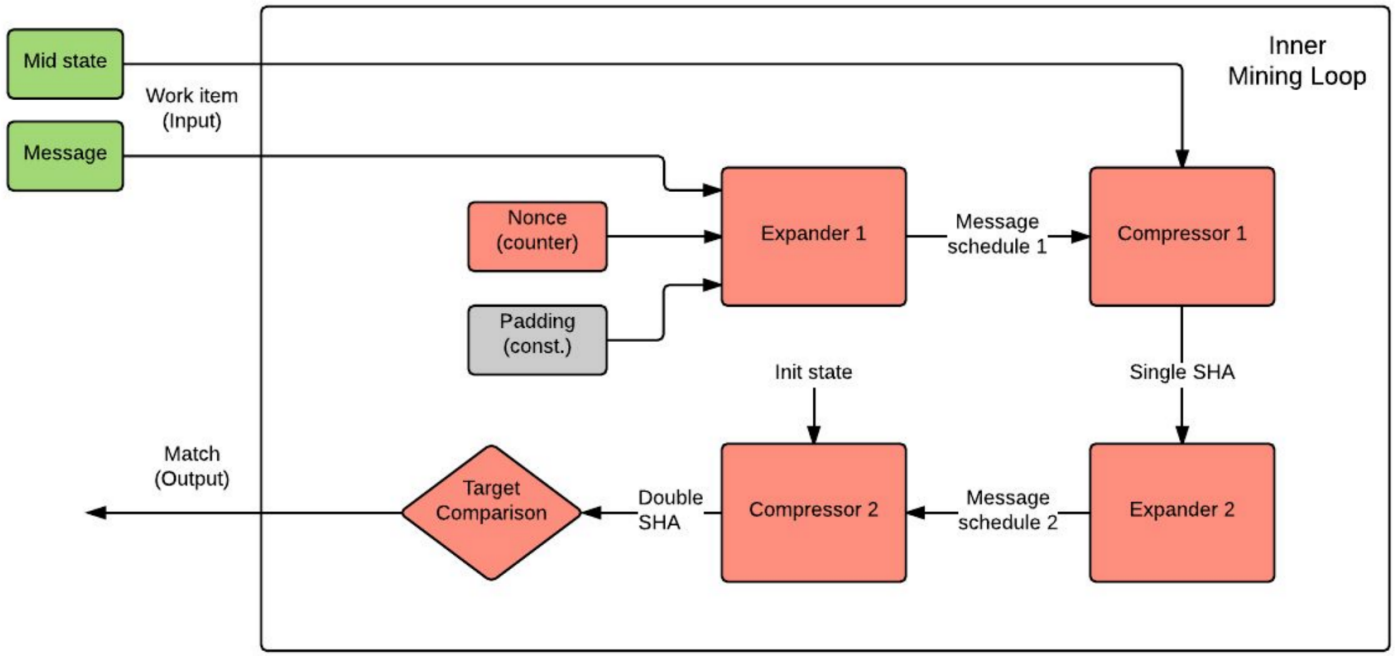


Image source: `https://arxiv.org/abs/1604.00575`

- In the above figure, the components shown in red are updated more frequently

AsicBoost Loop

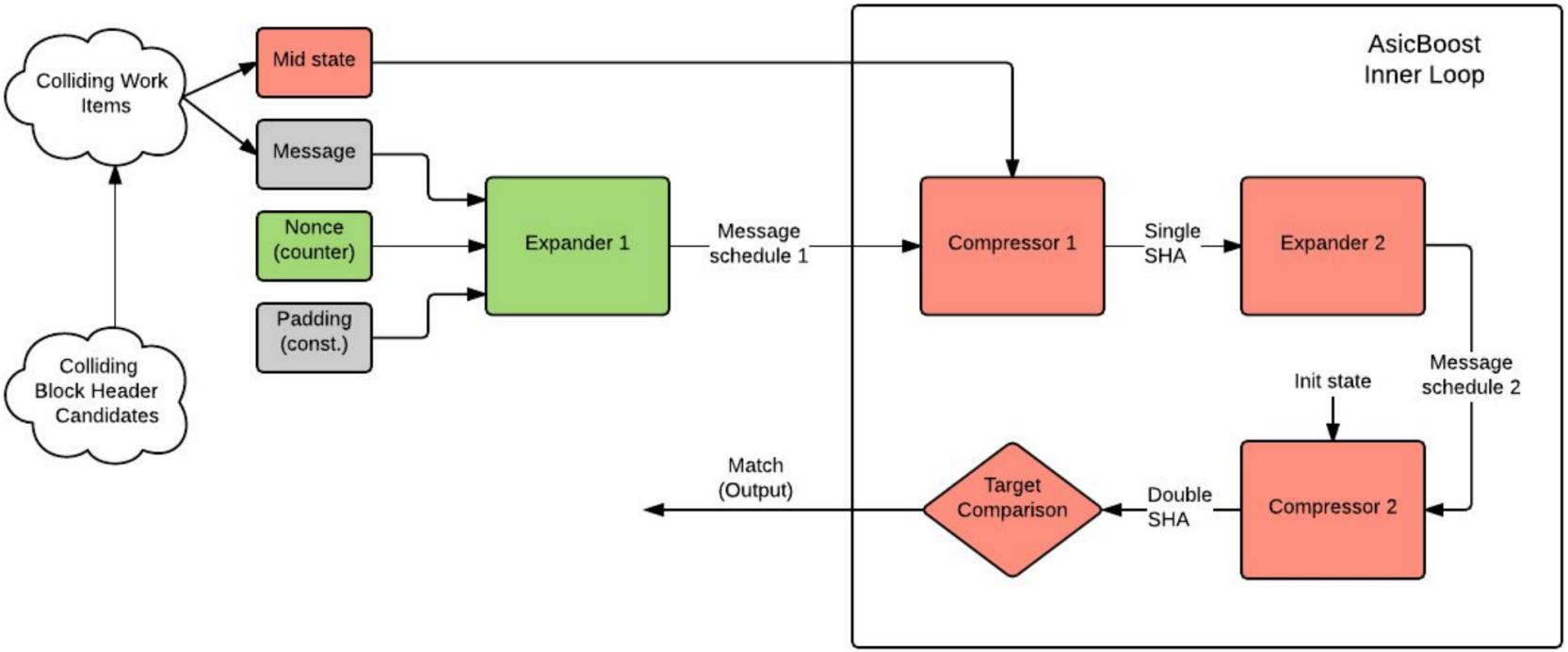


Image source: `https://arxiv.org/abs/1604.00575`

- In the above figure, the grey and green blocks represent computation that can be reused
- If two transaction Merkle roots coincide in the last 4 bytes, then the output of Expander 1 can be reused

AsicBoost Duo Core

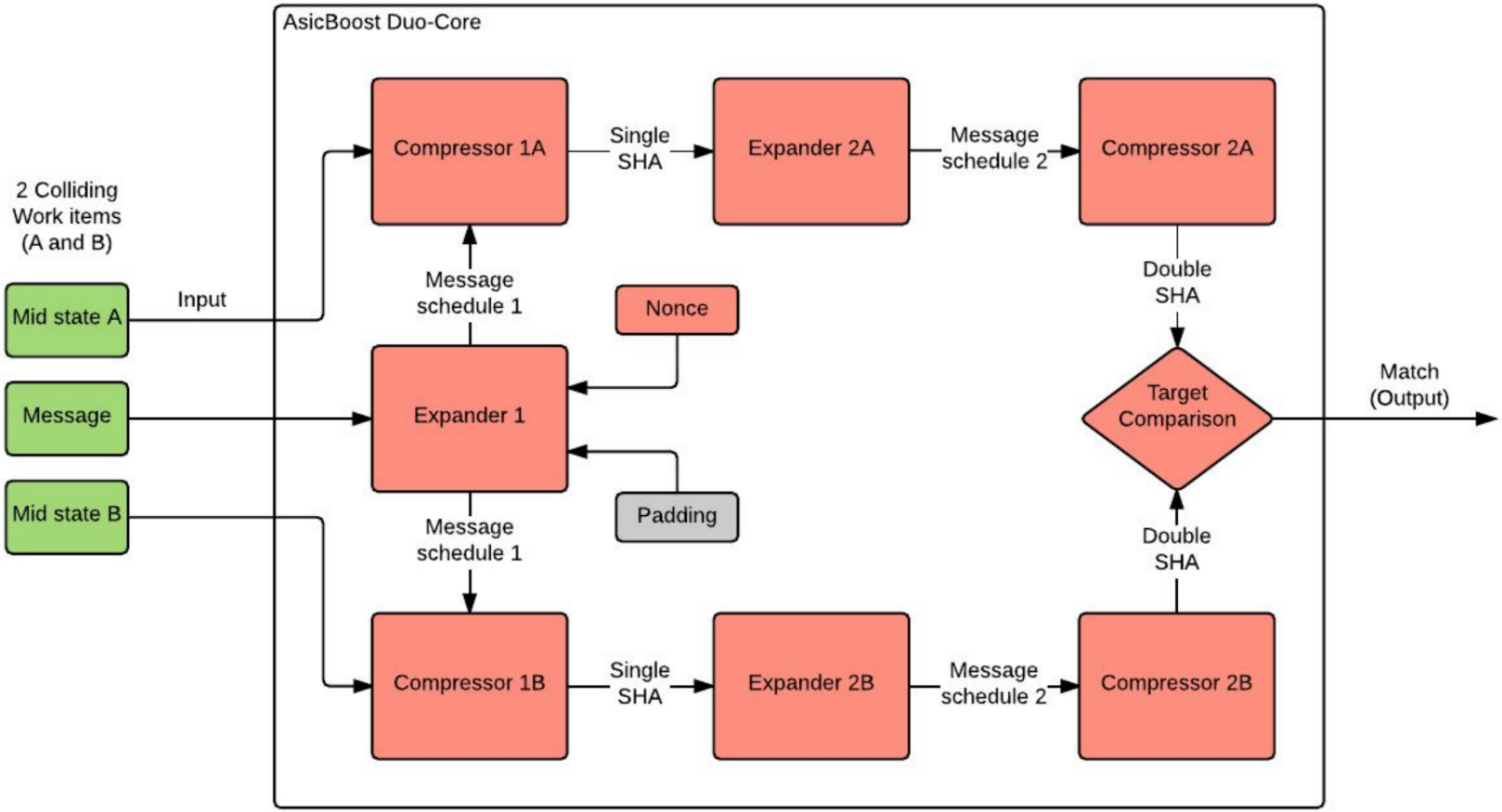


Image source: `https://arxiv.org/abs/1604.00575`

References

- Sections 4.2, 4.3, 5.3 of *An Introduction to Bitcoin*, S. Vijayakumaran, `www.ee.iitb.ac.in/~sarva/bitcoin.html`
- BIP 34: Block v2, Height in Coinbase `https://github.com/bitcoin/bips/blob/master/bip-0034.mediawiki`
- Bitcoin Genesis Block `https://en.bitcoin.it/wiki/Genesis_block`
- Bitcoin Blocks with Coinbase Markers `https://btc.com/block`
- Bitcoin Block Distribution `https://btc.com/stats/pool`
- Bitmain Mining Rigs `https://shop.bitmain.com/`
- Slushpool Documentation `https://slushpool.com/help/hashrate-proof/`
- Hardening Stratum, the Bitcoin Pool Mining Protocol `https://arxiv.org/abs/1703.06545`
- AsicBoost `https://arxiv.org/abs/1604.00575`