



AZ Money

Arizonas Microcurrency

What is AZ Money?

- A sidechain of the bitcoin blockchain
- A local microcurrency designed for Arizona
- A Proof-of-Work consensus protocol
- SHA 256
- Launched on Feb 14, 2023 – Arizona's Birthday
- Total of 7.6 million coins in a 5-year distribution phase.



A small group of unelected officials running the global economy is an impossible task.

AZ Money is the local response to Bitcoin's global one.

purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a central authority. Digital signatures provide part of the solution, but the main problem is that if a trusted third party is still required to prevent double-spending,

double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into a proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As proof, a majority of the network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave at any time. The network accepts the longest proof-of-work chain as proof of what happened while they were gone.

The diagram illustrates the linking of two blocks in a blockchain. The first block on the left contains a 'Prev Hash' field and a 'Nonce' field. An arrow points from the 'Prev Hash' field of the first block to the 'Prev Hash' field of the second block on the right. Both blocks also contain 'Tx' (transaction) fields.

The proof-of-work also solves the problem of determining the majority decision making. If the majority of nodes are based on one-IP-address, then, it could be subverted by anyone able to allocate many IPs. Proof-of-work is essentially one CPU-one vote. The majority decision is represented by the longest chain which has the greatest proof-of-work effort invested in it. If a majority of CPU power is controlled by honest nodes, the honest chain will grow the fastest and outpace any competing chains. To modify a past block, an attacker would have to redo the proof-of-work of the block and all blocks after it and then catch up with and surpass the work of the honest nodes. We will show later that the probability of a successful attack is exponentially smaller as blocks are added. To compensate for increasing hardware speed and varying interest in running nodes over time, the proof-of-work difficulty is determined by a moving average targeting an average number of blocks per hour. If they're generated too fast, the difficulty increases.

- 1) New transactions are broadcast to all nodes.
- 2) Each node collects new transactions into a block.
- 3) Each node works on finding a difficult proof-of-work for its block.
- 4) When a node finds a proof-of-work, it broadcasts the block to all nodes.

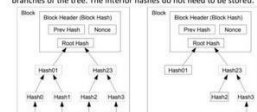
- 1) New transactions are broadcast to all nodes.
- 2) Each node collects new transactions into a block.
- 3) Each node works on finding a difficult proof-of-work for its block.
- 4) When a node finds a proof-of-work, it broadcasts the block to all nodes.
- 5) Nodes accept the block only if all transactions in it are valid and not already spent.
- 6) Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.

Nodes always consider the longest chain to be the correct one and will keep working on extending it. If two nodes broadcast different versions of the next block simultaneously, some nodes may receive one or the other first. In that case, they work on the first one they received, but save the other branch in case it becomes longer. The tie will be broken when the next proof-of-work is found and one branch becomes longer; the nodes that were working on the other branch will then switch to the longer one. New transactions broadcast do not necessarily need to reach all nodes. As long as they reach many nodes, they will get into a block before long. Blocks that are not included in a block are dropped immediately. If a node receives a block, it will request it when it receives the next block and realize it missed one.

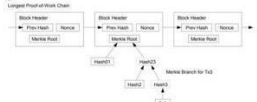
By convention, the first transaction in a block is a special transaction that starts a new coin owned by the creator of the block. This addition of an incentive for nodes to support the network, and provides a way to initially distribute coins into circulation, since there is no central authority to issue them. The steady addition of a constant amount of new coins is analogous to gold miners expending resources to add gold to circulation. In our case, it is CPU time and electricity that is generated. This incentive was also funded via

input value, the difference is a transaction fee that is added to the incentive value of the block containing the transaction. Once the predetermined number of coins have entered circulation, the incentive can transition entirely to transaction fees and does not completely inflate free. The incentive may help encourage nodes to stay honest. If a greedy attacker is able to assemble more CPU power than all the honest nodes, he would have to choose between using it to defraud people by stealing back his payments, or using it to generate new coins. He ought to find it more profitable to play by the rules and generate new coins that favour him with more new coins than anyone else. Even if he combined, then to undermine the system and the validity of his own wealth.

Once the latest transaction in a coin is buried under enough blocks the spent transactions before it can be discarded to save disk space. To facilitate this without breaking the block's hash, transactions are hashed in a Merkle Tree [7][2][5], with only the root included in the block's hash. Old blocks can then be compacted by stubbing off branches of the tree. The interior hashes do not need to be stored.

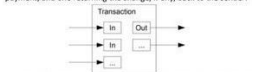


Merkle branch linking the transaction to the block it's timestamped in. He can't check the transaction for himself, but by linking it to a place in the chain, he can see that a network node has accepted it, and blocks added after it further confirm the network has accepted it.



As such, the verification is reliable as long as honest nodes control the network, but is more vulnerable if the network is overpowered by an attacker. While network nodes can verify transactions for themselves, the simplified method can be fooled by an attacker's fabricated transactions for as long as the attacker can continue to overpower the network. One strategy to protect against this would be to accept alerts from network nodes when they detect an invalid block, prompting the user's software to download the full block and alerted transactions to confirm the inconsistency. Businesses that receive frequent payments will probably still want to run their own nodes for more independent security and quicker verification.

Although it would be possible to handle coins individually, it would be unwieldy to make a separate transaction for every cent in a transfer. To allow value to be split and combined, transactions contain multiple inputs and outputs. Normally there will be either a single input from a larger previous transaction or multiple inputs combining smaller amounts, and at most two outputs: one for the payment, and one returning the change. If any, back to the sender.



It should be noted that fan-out, where a transaction depends on several transactions, and those transactions depend on many more, is not a problem here. There is never the need to extract a complete standalone copy of a transaction's history.

The traditional banking model achieves a level of privacy by limiting access to information to the parties involved and the trusted third party. The necessity to announce all transactions publicly precludes this method, but privacy can still be maintained by breaking the flow of information in another place: by keeping public keys anonymous. The public can see that someone is sending an amount to someone else, but without information linking the transaction to anyone. This is accomplished by using a pseudonym, or a public key, to identify the sender and receiver of the transaction.

anyone. This is similar to the level of information released by stock exchanges, where the time and size of individual trades, the "tape", is made public, but without telling who the parties were.

```

graph LR
    subgraph Current_Model [Current Model]
        Id1[Identities] --> Trans1[Transactions]
        Trans1 --> TTP[Trusted Third Party]
        TTP --> C[Counterparty]
        C --> P1[Public]
    end
    subgraph New_Privacy_Model [New Privacy Model]
        Id2[Identities] --> Trans2[Transactions]
        Trans2 --> P2[Public]
    end

```

As an additional firewall, a new key pair should be used for each transaction to keep them from being linked to a common owner. Some linking is still unavoidable with multi-input transactions, which necessarily reveal that their inputs were owned by the same owner. The risk is that if the owner of a key is revealed, linking could reveal other transactions that belonged to the same owner.

11. Calculations

We consider the scenario of an attacker trying to generate an alternate chain faster than the honest chain. Even if this is accomplished, it does not throw the system open to arbitrary changes, such as creating value out of thin air or taking money that never belonged to the attacker. Nodes are not going to accept an

The sender can't change the transaction. We assume the sender is an attacker who wants to make the recipient believe he paid him for a while, then switch it to pay back to himself after some time has passed. The receiver will be alerted when that happens, but the sender hopes it will be too late. The receiver generates a new keypair and gives the public key to the sender shortly before receiving this. This prevents the sender from preparing a chain of blocks ahead of time by working on it continuously until he is lucky enough to generate enough ahead, then executing the transaction at that moment. Once the transaction is sent, the dishonest sender starts working in secret on a parallel chain containing an alternate version of his transaction. The recipient waits until the transaction has been confirmed by a number of blocks before he links it to his. He doesn't assume an exact amount of progress the attacker has made, but assuming the honest blocks took the average expected time per block, the attacker's potential progress will be a Poisson distribution with expected value:

$$\lambda = \frac{zq}{p}$$

To get the probability the attacker could still catch up now, we multiply the Poisson density for each amount of progress he could have made by the probability he could catch up from that point:

$$\frac{f(x)}{1} \leq \frac{f(y)}{1} \quad \text{if } x \leq y$$
$$1 - \frac{1}{n} \sum_{k=0}^{n-1} \left(1 - \left(\frac{n-k}{n} \right)^{n-1} \right)$$

```
Converting to C code...
#include <math.h>
```

```
double AttackerSuccessProbability(double q, int z)
{
    double p = 1.0 - q;
    double lambda = z * (q / p);
    double sum = 1.0;
    int i, k;
    for (k = 0; k <= z; k++)
    {
        double poisson = exp(-lambda);
        for (i = 1; i <= k; i++)
            poisson *= lambda / i;
        sum += poisson * (1 - pow(q / p, z - k));
    }
    return sum;
}
```

Running some results, we can see the probability drop of exponentially with z . $q=0.1$ $z=0$ $P=1.00000000$ $z=1$ $P=0.2045873$ $z=2$ $P=0.0509779$ $z=3$ $P=0.0131722$ $z=4$ $P=0.0034552$ $z=5$ $P=0.000913$ $z=6$ $P=0.0002428$ $z=7$ $P=0.0000647$ $z=8$ $P=0.0000173$ $z=9$ $P=0.0000046$ $z=10$ $P=0.0000012$ $q=0.3$ $z=0$ $P=1.00000000$ $z=1$ $P=0.1773523$ $z=2$ $P=0.0416605$ $z=3$ $P=0.0101008$ $z=4$ $P=0.0024804$ $z=5$ $P=0.0006132$ $z=6$ $P=0.0001522$ $z=7$ $P=0.0000379$ $z=8$ $P=0.0000095$ $z=9$ $P=0.0000024$ $z=10$ $P=0.0000006$

12. Conclusion

We have proposed a new system for electronic transactions of controlled goods. The system was developed using the framework of combinatorial game theory. It is based on the use of digital signatures, which provides strong control of ownership, but is incomplete without a way to prevent double-spending. To solve this, we proposed a peer-to-peer network using a distributed ledger to record a public history of transactions that quickly becomes computationally impractical for an attacker to change. Dishonest nodes control a majority of CPU power. The network is robust in its unstructured simplicity. Nodes work all at once without any central authority or need for trust. Nodes can leave and join the network at will, accepting the proof-of-work chain as proof of what has been agreed upon. Nodes can be paid in a variety of ways, expressing their acceptance of valid blocks by working on extending the chain and rejecting invalid blocks by refusing to work on them. Any needed rules and incentives can be enforced with this consensus mechanism.

References

- [1] W. Dai, "b-money," <http://www.weidai.com/bmoney.txt>, 1998.
- [2] H. Massias, X.S. Avila, and J.-J. Quisquater, "Design of a secure timestamping service with minimal trust requirements," in 20th Symposium on Information Theory in the Benelux, May 1999.
- [3] S. Haber, W.S. Stornetta, "How to time-stamp a digital document," in *Journal of Cryptology*, vol. 3, no. 2, pages 99-111, 1991.
- [4] D. Bayer, S. Haber, W.S. Stornetta, "Improving the efficiency of



Microcurrencies

The Beginning of a Complete Banking and Governing System Overhaul

There is a natural global trend where power is undergoing constant consolidation and centralization. This direction poses a serious threat to Peace and Liberty everywhere! Launching microcurrencies is a powerful way to buck that trend! A microcurrency is decentralized money distributed within a local region with the eventual integration into local law and technologically implemented as a Bitcoin sidechain. It only takes a few good (wo)men to kick off a microcurrency in their respective area which could then set in motion a strong chain reaction for that area to eventually become completely sovereign with its own banking and governing system. Relatively small sovereign territories, with hundreds of thousands to several million people formed with the launching of a microcurrency, are referred to as microstates throughout this paper. As more and more microstates form and assert their sovereignty, the concentration of global power will reverse, with federal and world governments reduced into a multitude of independent coalitions that are strictly and narrowly defined.

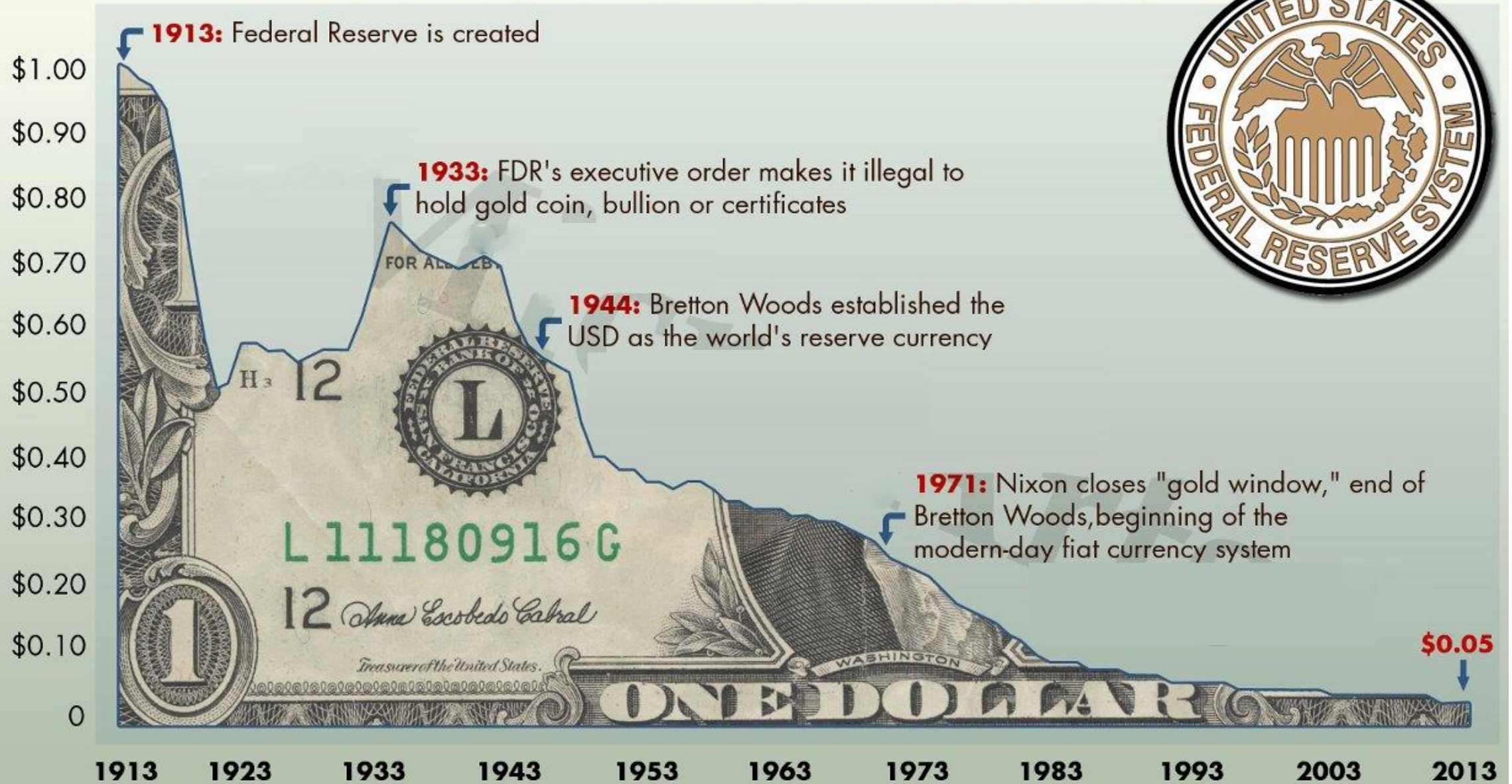
Ask about an AZ Money Mining Contract and Start Earning Arizona's Native Microcurrency Today!

1. The AZ Money Coin symbol is SAGZ which is short for Saguaro.
2. To help distribute the SAGZ we sell mining contracts shown in these images here.
3. These Mining contracts include a public/private wallet keypair so your mining rewards (SAGZ) go the wallet associated with your card.



HOW THE **FED** KILLED THE DOLLAR

Purchasing Power of the U.S. Dollar (1913-2013)



Source: U.S. Bureau of Labor Statistics



Over the past 15 years, Bitcoin has demonstrated incredible resilience, proving time and again that it is not a fleeting trend, but rather a groundbreaking innovation that has reshaped the way we think about money, finance, energy and the very foundations of trust. In an ever-evolving financial world, Bitcoin has not only survived but thrived, becoming the best-performing asset in the history of modern finance while leading the way for other cryptocurrencies.





-
- As digital assets ascend and take root, it will become obvious that the private digital storage of wealth is necessary for the dignity and safety of each person.



In essence, a blockchain is a trustworthy and decentralized way to record and verify digital information such as transactions. Blockchains can be used for various purposes beyond cryptocurrencies, such as supply chain tracking, voting systems, tokenization of real-world assets, and more, where transparency and security are paramount.



AZ Money

Arizonas Microcurrency

Peer-to-Peer
decentralized
finance is a
cornerstone for
a prosperous
society.





Bitcoin price appreciation far exceeded every other asset class over the last decade.

CREATIVE PLANNING		Asset Class Total Returns Since 2011 (Data via YCharts as of 10/31/23)													@CharlieBilello	
ETF	Asset Class	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023 YTD	2011-23 Cumulative	2011-23 Annualized
N/A	Bitcoin (\$BTC)	1473%	186%	5507%	-58%	35%	125%	1331%	-73%	95%	301%	66%	-65.5%	107.9%	11455951%	147.9%
QQQ	US Nasdaq 100	3.4%	18.1%	36.6%	19.2%	9.5%	7.1%	32.7%	-0.1%	39.0%	48.6%	27.4%	-32.6%	32.3%	625.2%	16.7%
IWF	US Growth	2.3%	15.2%	33.1%	12.8%	5.5%	7.0%	30.0%	-1.7%	35.9%	38.3%	27.4%	-29.3%	23.0%	431.6%	13.9%
SPY	US Large Caps	1.9%	16.0%	32.2%	13.5%	1.2%	12.0%	21.7%	-4.5%	31.2%	18.4%	28.7%	-18.2%	10.6%	322.6%	11.9%
GLD	Gold	9.6%	6.6%	-28.3%	-2.2%	-10.7%	8.0%	12.8%	-1.9%	17.9%	24.8%	-4.2%	-0.8%	8.5%	32.7%	2.2%
BIL	US Cash	0.0%	0.0%	-0.1%	-0.1%	-0.1%	0.1%	0.7%	1.7%	2.2%	0.4%	-0.1%	1.4%	4.0%	10.4%	0.8%
EFA	EAFE Stocks	-12.2%	18.8%	21.4%	-6.2%	-1.0%	1.4%	25.1%	-13.8%	22.0%	7.6%	11.5%	-14.4%	3.8%	67.2%	4.1%
HYG	High Yield Bonds	6.8%	11.7%	5.8%	1.9%	-5.0%	13.4%	6.1%	-2.0%	14.1%	4.5%	3.8%	-11.0%	3.1%	63.1%	3.9%
CWB	Convertible Bonds	-7.7%	15.9%	20.5%	7.7%	-0.8%	10.6%	15.7%	-2.0%	22.4%	53.4%	2.2%	-20.8%	2.2%	168.1%	8.0%
EMB	EM Bonds (USD)	7.7%	16.9%	-7.8%	6.1%	1.0%	9.3%	10.3%	-5.5%	15.5%	5.4%	-2.2%	-18.6%	-0.4%	36.6%	2.5%
DBC	Commodities	-2.6%	3.5%	-7.6%	-28.1%	-27.6%	18.6%	4.9%	-11.6%	11.8%	-7.8%	41.4%	19.3%	-0.6%	-7.9%	-0.6%
TIP	TIPS	13.3%	6.4%	-8.5%	3.6%	-1.8%	4.7%	2.9%	-1.4%	8.3%	10.8%	5.7%	-12.2%	-1.4%	31.0%	2.1%
MDY	US Mid Caps	-2.1%	17.8%	33.1%	9.4%	-2.5%	20.5%	15.9%	-11.3%	25.8%	13.5%	24.5%	-13.3%	-1.4%	208.2%	9.2%
PFF	Preferred Stocks	-2.0%	17.8%	-1.0%	14.1%	4.3%	1.3%	8.1%	-4.7%	15.9%	7.9%	7.2%	-18.2%	-1.7%	52.9%	3.4%
IWD	US Value	0.1%	17.5%	32.1%	13.2%	-4.0%	17.3%	13.5%	-8.5%	26.1%	2.7%	25.0%	-7.7%	-1.9%	201.2%	9.0%
BND	US Total Bond Market	7.7%	3.9%	-2.1%	5.8%	0.6%	2.5%	3.6%	-0.1%	8.8%	7.7%	-1.9%	-13.1%	-2.4%	20.5%	1.5%
EEM	EM Stocks	-18.8%	19.1%	-3.7%	-3.9%	-16.2%	10.9%	37.3%	-15.3%	18.2%	17.0%	-3.6%	-20.6%	-2.4%	0.0%	0.0%
LQD	Investment Grade Bonds	9.7%	10.6%	-2.0%	8.2%	-1.3%	6.2%	7.1%	-3.8%	17.4%	11.0%	-1.8%	-17.9%	-3.0%	41.4%	2.7%
IWM	US Small Caps	-4.4%	16.7%	38.7%	5.0%	-4.5%	21.6%	14.6%	-11.1%	25.4%	20.0%	14.5%	-20.5%	-4.6%	151.4%	7.4%
VNQ	US REITs	8.6%	17.6%	2.3%	30.4%	2.4%	8.6%	4.9%	-6.0%	28.9%	-4.7%	40.5%	-26.2%	-8.9%	116.9%	6.2%
TLT	Long Duration Treasuries	34.0%	2.6%	-13.4%	27.3%	-1.8%	1.2%	9.2%	-1.6%	14.1%	18.2%	-4.6%	-31.2%	-14.0%	23.1%	1.6%
Highest Return		BTC	BTC	BTC	VNQ	BTC	BTC	BTC	BIL	BTC	BTC	BTC	DBC	BTC	BTC	BTC
Lowest Return		EEM	BIL	GLD	BTC	DBC	BIL	BIL	BTC	BIL	DBC	TLT	BTC	TLT	DBC	DBC
% of Asset Classes Positive		62%	95%	52%	71%	38%	100%	100%	5%	100%	90%	67%	10%	43%	90%	90%



@21JCLP

When we look at the data dating back to 2011, the numbers tell an astonishing story. The cumulative return on Bitcoin (BTC) stands at an incredible 11,207,985 %, with an annualized growth rate of 147.5 %.

This performance is undeniably superior to all other financial assets, and it is a testament to the resilience and ingenuity behind this groundbreaking technology.

What is a blockchain?

+-----+	+-----+	+-----+
Block 1	Block 2	Block 3
Transactions:	Transactions:	Transactions:
- A -> B: 10	- B -> C: 5	- C -> D: 3
- C -> D: 2	- D -> A: 4	- A -> B: 7
Hash: 0x123...	Hash: 0x456...	Hash: 0x789...
Prev. Hash: N/A	--> Prev. Hash: 0x123...	--> Prev. Hash: 0x456...
+-----+	+-----+	+-----+



- A blockchain is like a digital ledger or record book that keeps track of transactions in a secure and unchangeable way over the course of time. It's a system of storing and transferring data that is distributed across a network of computers. Each block in the chain contains a record of transactions, a cryptographic hash of the previous block, and a timestamp. The blocks are linked together by the hashes, forming a chain that is resistant to tampering and fraud. One way of visualizing this is by imagining a chain of blocks, with the genesis block being the first block in the chain, and where each block contains a list of transactions.

Ask about an AZ Money Mining Contract and Start Earning Arizona's Native Microcurrency Today!

1. The AZ Money Coin symbol is SAGZ which is short for Saguaro.
2. To help distribute the SAGZ we sell mining contracts shown in these images here.
3. These Mining contracts include a public/private wallet keypair so your mining rewards (SAGZ) go the wallet associated with your card.



What is AZ Money?

- A sidechain of the bitcoin blockchain
- A local microcurrency designed for Arizona
- A Proof-of-Work consensus protocol
- SHA 256
- Launched on Feb 14, 2023 – Arizona's Birthday
- Total of 7.6 million coins in a 5-year distribution phase.



AZ Money uses the same mining process as Bitcoin with some minor differences

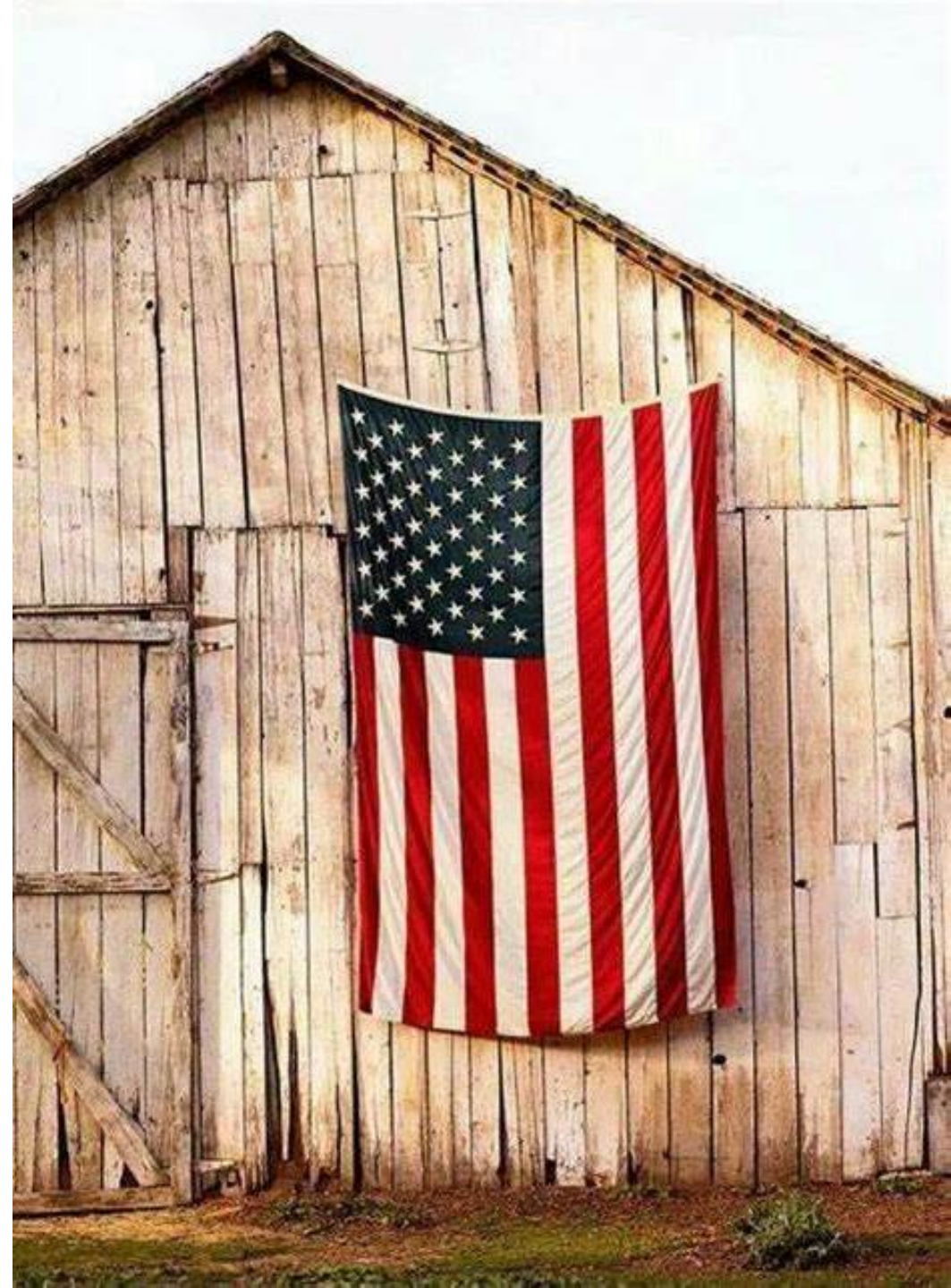


Facts	Bitcoin	AZ Money
Supply	21,000,000 total coins	7,637,625 total coins
Consensus Mechanism	POW (proof of work)	POW (proof of work)
Genesis Block	January 3, 2009	February 14, 2023
Fully Distributed By	Year 2140 (Estimated)	Feb 14, 2028 (Estimated)
Divisible by	100,000,000 satoshis (\$ATS) = 1 BTC	100,000,000 saguaros (SAGZ) = 1 AZ Money Coin
Current block reward	6.25	15
Block time	10 minutes	2 minutes
Halvings	Every 4 years	Yearly
Block size	Weighted Limit (1-4 MB)	Weighted Limit (Legacy = 100KB)
Hashrate (1 EH/s is equal to 1,000,000 GH/s, 1 TH/s is equal to 1,000 GH/s).	As of August 2023, the current Bitcoin hashrate is 467.04 EH/s	78 TH/s (0.000078 EH/s)

Just as silver and gold have distinct roles in traditional financial portfolios, AZ Money and Bitcoin offer diverse opportunities within the cryptocurrency space. While Bitcoin serves as a digital store of value and a secure asset for long-term investment, AZ Money provides a local entry point for those looking to engage with blockchain technology. These digital counterparts together contribute to the evolving landscape of digital finance, offering options that cater to a wide spectrum of investors and use cases.

AZ Money was launched to help inspire a local community to adopt an interdependent monetary, banking, and governmental system that honors and protects the fullness of individual liberty without exception. We must join arms in changing the way we think about banking that is why it's important to share this with your network of friends and family.

Banks are an essential service. The solution to this conundrum starts by embracing Liberty where anyone can open and operate financial services WITHOUT permission, then hold those that have operated such services dishonestly accountable! This simple change moves much of the responsibility to ensure honest banking from the government back to the people.



Ask about an AZ Money Mining Contract and Start Earning Arizona's Native Microcurrency Today!

1. The AZ Money Coin symbol is SAGZ which is short for Saguaro.
2. To help distribute the SAGZ we sell mining contracts shown in these images here.
3. These Mining contracts include a public/private wallet keypair so your mining rewards (SAGZ) go the wallet associated with your card.



The Money of Freedom

Bitcoin is often seen as a symbol of global financial freedom because it operates on a decentralized network, free from government control and centralized institutions. AZ Money is a digital currency that empowers individuals on a local level. Our goal is to provide people with the ability to transact within Arizona without the need for traditional banks. AZ Money has fixed supply and is resistant to inflation, protecting one's wealth from the erosion caused by fiat currency devaluation. All these attributes that make Bitcoin a symbol of global financial sovereignty and autonomy for individuals seeking to take control of their own financial destinies are should be applied at a local and state level to help bring Bitcoin adoption to the state of Arizona.

A PEER-TO-PEER ELECTRONIC CASH SYSTEM

purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a central institution. Digital signatures provide part of the solution, but the main missing piece is a method for preventing double-spending. To solve this, a peer-to-peer network is needed where each participant maintains a copy of the entire transaction history. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave at any time. The network is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain of blocks, and nodes that are not cooperating to attack the network will accept the longest proof-of-work chain as proof of what happened while they were gone.

The double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into a proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As more nodes join the network, the proof of work itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave at any time. The network is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain of blocks, and nodes that are not cooperating to attack the network will accept the longest proof-of-work chain as proof of what happened while they were gone.



The proof-of-work also solves the problem of determining representation in majority decision making. If the majority were based on one-IP-address-one-vote, it could be subverted by anyone able to allocate many IPs. Proof-of-work is essentially one-CPU-one-vote. The majority decision is represented by the longest chain, which has the greatest proof-of-work effort invested in it. If a majority of CPU power is controlled by honest nodes, the honest chain will grow the fastest and outpace any competing chain. To modify a past block, an attacker would have to redo the proof-of-work of the block and all blocks after it and then catch up with and surpass the work of the honest nodes. We will show later that the probability of a slower attacker catching up diminishes exponentially as subsequent blocks are added. To compensate for increasing hardware speed and varying interest in running nodes over time, the proof-of-work difficulty is determined by a moving average targeting an average number of blocks per hour. If the target time is not reached, the difficulty increases.

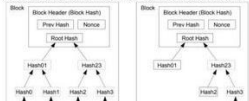
- ### 5. Network
- The steps to run the network are as follows:
- 1) New transactions are broadcast to all nodes.
 - 2) Each node collects new transactions into a block.
 - 3) Each node works on finding a difficult proof-of-work for its block.
 - 4) When a node finds a proof-of-work, it broadcasts the block to all nodes.
 - 5) Nodes accept the block only if all transactions in it are valid and not already spent.
 - 6) Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.
- Nodes always consider the longest chain to be the correct one and will keep working on extending it. If two nodes broadcast different versions of the next block simultaneously, some nodes may receive one or the other first. In that case, they work on the first one they received, but save the other branch in case it becomes longer. The tie will be broken when the next proof-of-work is found and one branch becomes longer; the nodes that were working on the other branch will then switch to the longer one. New transaction broadcasts do not necessarily need to reach all nodes. As long as they reach many nodes, they will get into a block before long. Block broadcasts are also tolerant of dropped messages. If a node does not receive a block, it will request it when it receives the next block and realizes it missed one.

6. Incentive

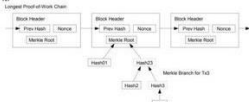
By convention, the first transaction in a block is a special transaction that starts a new coin owned by the creator of the block. This adds an incentive for nodes to support the network, and provides a way to initially distribute coins into circulation, since there is no central authority to issue them. The steady addition of a constant amount of new coins is analogous to gold miners expending resources to add gold to circulation. In our case, it is CPU time and electricity that is expended. This incentive can be funded with transaction fees. If the output value of a transaction is less than its input value, the difference is a transaction fee that is added to the incentive value of the block containing the transaction. Once a predetermined number of coins have entered circulation, the incentive can transition entirely to transaction fees and be used to fund operations. The incentive may help encourage nodes to stay honest. If a greedy miner is able to assemble more CPU power than all the honest nodes, he would have to choose between using it to defraud people by stealing back his payments, or using it to generate new coins. He ought to find it more profitable to play by the rules, such rules that favour him with more new coins than everyone else combined, than to undermine the system and the validity of his own wealth.

7. Reclaiming Disk Space

Once the latest transaction in a coin is buried under enough blocks, the spent transactions before it can be discarded to save disk space. To facilitate this without breaking the block's hash, transactions are hashed in a Merkle Tree [7][25], with only the root included in the block's hash. Old blocks can then be compacted by stubbing off branches of the tree. The interior hashes do not need to be stored.



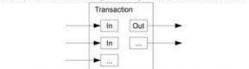
Merkle branch linking the transaction to the block it's timestamped in. He can't check the transaction for himself, but by linking it to a place in the chain, he can see that a network node has accepted it, and blocks added after it further confirm the network has accepted it.



As such, the verification is reliable as long as honest nodes control the network, but is more vulnerable if the network is overpowered by an attacker. While network nodes can verify transactions for themselves, the unmodified method can be fooled by an attacker's fabricated transactions as long as the attacker can continue to overpower the network. One strategy to protect against this would be to accept alerts from network nodes when they detect an invalid block, prompting the user's software to download the full block and alerted transactions to confirm the inconsistency. Businesses that receive frequent payments will probably want to run their own nodes for more independent security and quicker verification.

9. Combining and Splitting Value

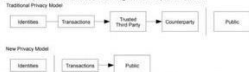
Although it would be possible to handle coins individually, it would be unwise to make a separate transaction for every cent in a transfer. To allow value to be split and combined, transactions contain multiple inputs and outputs. Normally there will be either a single input from a larger previous transaction or multiple inputs combining smaller amounts, and at most two outputs: one for the payment, and one returning the change, if any, back to the sender.



It should be noted that fan-out, where a transaction depends on several transactions, and those transactions depend on many more, is not a problem here. There is never the need to extract a complete standalone copy of a transaction's history.

10. Privacy

The traditional banking model achieves a level of privacy by limiting access to information to the parties involved and the trusted third party. The necessity to announce all transactions publicly precludes this method, but privacy can still be maintained by breaking the flow of information in another place: by keeping public keys anonymous. The public can see that someone is sending an amount to someone else, but without information linking the transaction to anyone. This is similar to the level of information released by stock exchanges, where the time and size of individual trades, the "tape", is made public, but without telling who the parties were.



As an additional firewall, a new key pair should be used for each transaction to keep them from being linked to a common owner. Some linking is still unavoidable with multi-input transactions, which necessarily reveal that the inputs were owned by the same owner. The risk is that if the owner of a key is revealed, linking could reveal other transactions that belonged to the same owner.

11. Calculations

We consider the scenario of an attacker trying to generate an alternate chain faster than the honest chain. Even if this is accomplished, it does not throw the system open to arbitrary changes, such as creating value out of thin air or taking money that never belonged to the attacker. Nodes are not going to accept an invalid transaction as payment, and honest nodes will never accept a block containing them. An attacker can only try to change one of his own transactions to take back money he recently spent. The race between the honest chain and an attacker chain can be characterized as a Binomial Random Walk. The success event is the honest chain being extended by one block, increasing its lead by +1, and the failure event is the attacker's chain being extended by one block, reducing the gap by -1. The probability of an attacker catching up from a given deficit is analogous to a Gambler's Ruin problem. Suppose a gambler with unlimited credit starts at a deficit and plays potentially an infinite number of trials to try to reach break-even. We would calculate the probability he ever reaches

the sender can't change the transaction. We assume the sender is an attacker who wants to make the recipient believe he paid him for a while, then switch it to pay back to himself after some time has passed. The receiver will be alerted when that happens, but the sender hopes it will be too late. The receiver generates a new key pair and gives the public key to the sender shortly before signing. Once the transaction is sent, the dishonest sender starts working in secret on a parallel chain containing an alternate version of his transaction. The recipient waits until the transaction has been added to a block and a blocks have been linked after it. He doesn't know the exact amount of progress the attacker has made, but assuming the honest blocks took the average expected time per block, the attacker's potential progress will be a Poisson distribution with expected value:

$$\lambda = \frac{2^k}{2^k - 1}$$

To get the probability the attacker could still catch up now, we multiply the Poisson density at each amount of progress he could have made by the probability he would catch up from that point:

$$\sum_{k=0}^{\infty} \frac{e^{-\lambda} \lambda^k}{k!} \cdot \frac{2^k}{2^k - 1}$$

Rearranging to avoid summing the infinite tail of the distribution...

$$1 - \sum_{k=0}^{\infty} \frac{e^{-\lambda} \lambda^k}{k!} \cdot \frac{2^k}{2^k - 1}$$

Converting to code...

```
#include <math.h>
double AttackerSuccessProbability(double q, int z)
{
    double p = 1.0 - q;
    double lambda = z * (q/p);
    double sum = 1.0;
    int k;
    for (k = 0; k <= z; k++)
    {
        double poisson = exp(-lambda);
        for (j = 1; j <= k; j++)
            poisson *= lambda / j;
        sum += poisson * (1 - pow(q/p, z - k));
    }
    return sum;
}
```

Running some results, we can see the probability drop off exponentially with z. q=0.1 z=0 P=1.0000000 z=1 P=0.3045873 z=2 P=0.0597779 z=3 P=0.0131772 z=4 P=0.003652 z=5 P=0.00137 z=6 P=0.000428 z=7 P=0.000147 z=8 P=0.0000713 z=9 P=0.000046 z=10 P=0.000022 z=11 P=0.000010 z=12 P=0.0000046 z=13 P=0.0000022 z=14 P=0.0000010 z=15 P=0.00000046 z=16 P=0.00000022 z=17 P=0.00000010 z=18 P=0.000000046 z=19 P=0.000000022 z=20 P=0.000000010

Solving for P less than 0.05... P < 0.001 q=0.10 z=5 q=0.15 z=0.20 z=1 q=0.25 z=15 q=0.30 z=24 q=0.35 z=41 q=0.40 z=89 q=0.45 z=144

12. Conclusion

We have proposed a system for electronic transactions without relying on trust. We started with the usual framework of coins made from digital signatures, which provides strong control of ownership, but is incomplete without a way to prevent double-spending. To solve this, we proposed a peer-to-peer network using proof-of-work to record a public history of transactions that quickly becomes computationally impractical for an attacker to change if honest nodes control a majority of CPU power. The network is robust in its unstructured simplicity. Nodes work all at once with little coordination. They do not need to be identified, since messages are not routed to any particular place and only need to be delivered on a best effort basis. Nodes can leave and rejoin the network at will, accepting the proof-of-work chain as proof of what happened while they were gone. They vote with their CPU power, expressing their acceptance of valid blocks by working on extending them and rejecting invalid blocks by refusing to work on them. Any needed rules and incentives can be enforced with this consensus mechanism.

References

[1] W. Dai, "B-money," <http://www.world.com/bitmoney.htm>, 1998.
[2] M. Maslins, X.S. Avila, and J.-J. Quisquater, "Design of a secure timestamping service with minimal trust requirements," in 20th Symposium on Information Theory in the Benelux, May 1995.
[3] S. Haber, W.S. Stornetta, "How to time-stamp a digital document," in Journal of Cryptology, vol. 3, no. 2, pages 99-111, 1991.
[4] D. Bayer, S. Haber, W.S. Stornetta, "Improving the efficiency and



Secure, fast, immutable settlement is better because...



Fiat money is just an exploit:



It's arbitrage between the security and final settlement of physical money, and the insecurity and reversibility of its virtual impersonators.



Bitcoin is simply a technological solution to end this arbitrage. And by extension, demonetize the massive and horribly destructive political industry & rent seeking financial structure that has been globally constructed around it.



If we focus only on the activity of villains, we develop a blindspot for what heroes build.

Contrary to crypto skeptics' views, the merits of decentralization and personal autonomy align with fundamental American rights: free speech, privacy and due process.





Crypto's unique benefits play a role in strengthening core American rights. We urge crypto's antagonists to take a closer look at the technology's role in enhancing — not undermining — some of our vital Constitutional provisions.

When in the course of human events, it becomes necessary for one people to dissolve the political bands which have con-

[illegible]

Wilton, Gainsath,
Lymann Hall,
Geo. Walston.

Geo Hooper
 Joseph Hewes
 John Pison

Edmond R. Hodge Jr.

Thos Lloyd Jones
Arthur Middleton

John Hancock

Samuel Chace
Wm. Paro
Thos. Stone
Chas. Smith

George Washington
Richard Henry
The Jefferson

Robt Morris
Benjamin Bush

Benj. Franklin
John Morton

Geo. Lymer
Scrib.

- The First Amendment protects our right to freedom of expression, whether through art, words, music, computer code or associations. This right is strengthened when individuals can own their own expressive content. And blockchain technology makes ownership possible online.
- Blockchain technology allow individuals to create and retain ownership over their entire social media presence, so that no one platform can censor their content or delete their data. Global blockchain networks allow people a protected avenue of political speech without fear of persecution.
- These Web3 ideals in action stand in stark contrast to traditional financial systems, which are often controlled by those with the largest financial stakes, plagued by censorship, and tend not to be concerned with innovation or the free exchange of ideas.



- Although computer code came into existence less than a century ago, it has become an important way in which developers express ideas and develop new technologies and systems. Source code is the language in which developers and designers share ideas about science and engineering across the globe and is even considered by some courts to be protected speech.
- While the “right to code” is not found in our founding documents (how could it be?), it should be considered, in the popular imagination, as something similar to a composer’s score or sharing the spoken or written word, an issue Americans are passionate about. Rather than trying to treat code as a novel entity, crypto’s open source nature should be heralded as a test of the durability of America’s free speech protections, evolving the idea for the 21st century.

Ask about an AZ Money Mining Contract and Start Earning Arizona's Native Microcurrency Today!

1. The AZ Money Coin symbol is SAGZ which is short for Saguaro.
2. To help distribute the SAGZ we sell mining contracts shown in these images here.
3. These Mining contracts include a public/private wallet keypair so your mining rewards (SAGZ) go the wallet associated with your card.





- Moving away from centralization
- The Fourth Amendment safeguards our privacy and security by protecting us from unreasonable searches and seizures. Open blockchain networks enhance privacy and ownership over financial and personal information in many ways.





- Born out of the wreckage of the 2008 financial crisis — where we learned the destructive impact of centralized power in the financial system — open blockchain networks and digital assets reduce the reliance on risky centralized intermediaries.
- These networks ensure there are little to no meaningful barriers to entry for those who wish to participate in the global economy, evening the playing field for everyone with an internet connection. These networks also allow individuals to avoid the risk of their intermediaries closing or preventing access to their accounts.





Exxon is dealing with greenhouse gas emissions by ... mining crypto?

The oil and gas giant is piloting an effort to use methane gas to mine crypto, and expansion could be around the corner.



Exxon is using natural gas to mine bitcoin. Because of course it is.
Image: American Public Power Association/Unsplash

Exxon Weighs Taking Gas-to-Bitcoin Pilot to Four Countries

- The oil giant launched its Bakken crypto pilot in January 2021
- Miners are pushing to use 'flared' gas to power operations

