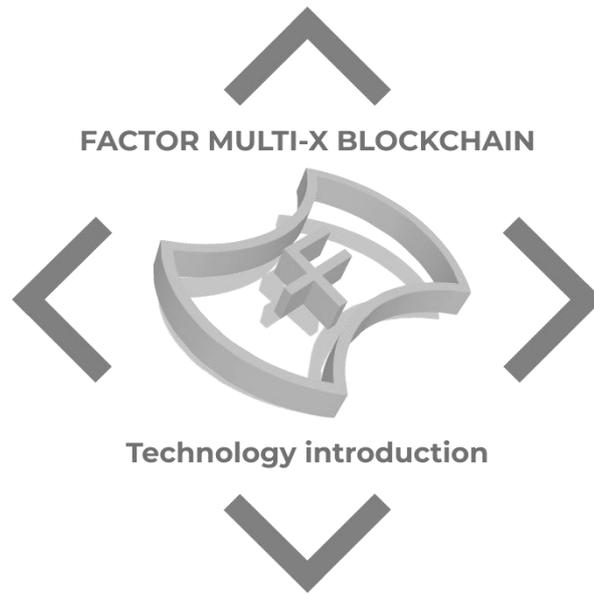


Factor Whitepaper



Version 0.1

Date updated: 06/05/2019

Factor Multi-X Blockchain
Company

Introduction

From the past to the present days, 'technology' has developed under various forms. The Internet, which we commonly use today, started as a military technology, only later, business applications were found. Technology is the driving force that makes many things possible, but the real impact it has on many people's lives starts from the moment the technology is exploited and spread to people. In this context, blockchains are the same. Currently, there are many layers of interest and enthusiasm regarding the block chain field. Many of them are considered to be ahead of their time. It is clear that the philosophy of individual sovereignty that excludes central governing institutions and authorizes all participants, can become a reality thanks to IT technology. These technologies are making contribution and rewards of participants possible, and the business applications using this technologies are the new paradigms developed by block chains companies.

However, looking at the current phenomenon from the impact and potential change it has in our lives, we can't help but acknowledge that the current stage of development of the blockchain is still "early." This is because despite the continuous development of blockchain technology, new blockchain developers still face many problems increasing running fees and many other problems that are related to the existing blockchains technology. Also, there are still many technical problems to be solved, and many blockchains are being proposed and developed at the level of idea or concept.

Factor Blockchain has been developed and researched to address these problems. Factor address challenges such as current Oracle problems, TPS speeds, scalability of blockchain, and enables the development of blockchain applications for the next generation of networks coming in a near future. Factor Blockchain's development is aiming at integrating existing blockchain systems, and rather than being limited to the purpose of connectivity with each coin, it can be said that the existing blockchain and Factor Blockchain are being connected. If the current block chains are reborn (as a phoenix) as a single block chain, it will be a breakthrough to solve a variety of problems like TPS speeds, scalability, and compatibility that may arise when blockchains are applied to real ecosystems.

The ultimate goal of Factor Blockchain is to ensure that the use of blockchain-applied technology is not simply for finance, but for innovative uses that will yield high speed, compatibility, scalability and limitless application value in the industrial ecosystem that the coming Fourth Industrial Revolution will bring about. Furthermore, the Factor Blockchain technology team is confident that it will be an innovative opportunity to change the foundation of the economic ecosystem in various areas such as smartphones, home appliances, cars, smart homes, weather observation and trade.

Factor MX Blockchain will not only connect existing block chains, but will also take the lead in developing next-generation networks, and will not stop ongoing development and efforts for blockchain life, as well as to address existing technical difficulties, excessive costs, inconsistencies in programming development languages, and incompatibility and difficulty of use with existing programs. Based on the versatility of the blockchain, Factor MX Blockchain will support various collaborative scenarios of trust, and will continue to increase the quantity of modules and protocols in accordance with the requirements of the new scenarios and application.

Blockchain is redefined through Factor MX Blockchain, and the future we will encounter will be realized by Factor MX Blockchain.

Index

1 Structure and Design

- 1 - 1 Frame Work Design
- 1 - 2 Distributed Ledger Application
- 1 - 3 Contract Models
- 1 - 4 MerkleTree Storage Model
- 1 - 5 Node System
- 1 - 6 Core Protocol
- 1 - 7 User Authority Management Protocol
- 1 - 8 Distributed Data Exchange Protocol
- 1 - 9 Application Framework
- 1 - 10 History Management Module
- 1 - 11 Databases

2 Improvements and Connectivity

- 2 - 1 Ethereum
- 2 - 2 Neo
- 2 - 3 Eos
- 2 - 4 significance of Connectivity

3 Speed

- 3 - 1 MX node
- 3 - 2 Bootstrap
- 3 - 3 Seed node
- 3 - 4 Masternode Speed
- 3 - 5 Speed Hash

4 Security and Stability

- 4 - 1 Secp256R1 key
- 4 - 2 Private Send
- 4 - 3 MerkleTree Application Security
- 4 - 4 Master Node Network Attack Security
- 4 - 5 Oracle section Security
- 4 - 6 Security Hash

5 Scalability

- 5 - 1 Hyperledger
- 5 - 2 Graphene
- 5 - 3 Finance
- 5 - 4 IOT
- 5 - 5 Public Institutions
- 5 - 6 Logistics, Distribution
- 5 - 7 Manufacturing and Production

6 Dapp

- 6 - 1 Factor Dapp
- 6 - 2 Factor ToolKit
- 6 - 3 Supported Languages and Programs
(Oracle section)

7 Games

- 7 - 1 Unreal Engine
- 7 - 2 Unity Engine
- 7 - 3 Havoc Engine
- 7 - 4 Cry Engine
- 7 - 5 Jupiter Engine
- 7 - 6 Gamebryo Engine
- 7 - 7 Source Engine

8 Security Solutions

- 8 - 1 Factor Vaccine
- 8 - 2 Factor Online Security
- 8 - 3 Factor Business Solution

9 Glossary

10 Conclusion

11 References

1. Structure and Design

1 - 1 Framework Design

The factor blockchain supports a distributed ledger ecosystem, and a system supporting multiple platforms by applying the MX (Multi-X) block chain. It increases speed, scalability, and compatibility through over 26-factor hashes function within MX block chain. It connects them all within one block (patented technology) connecting each hashes together. Therefore, it is compatible with all block chains, and offers a wide range of capabilities with excellent scalability. This enables applications to be used on a variety of platforms and provides excellent speeds through the use of MXnode, Seed node, Masternode, Normal node, the spread method of Pos, and acceleration through hash rate power of POW.

In addition, the MX node system is a solution to attempts to speed up the existing system, just like choosing a 21BP system was (for EOS system). It has a security and speed-up role, and is used to form a blockchain. Using these features, DAPP users can develop software that is supportive and compatible at a low cost and high efficiency. Factor's Bootstrap technology is designed to pre-download the existing distributed ledger, after checking a SHA256 Checksum of Bootstrap in a highly secure way to get the most out of your speed without the need for a full synchronization process. In addition to supporting the next-generation of networks coming in the future, this application allows the blockchain to perform the future innovative role of bringing one blockchain that unifies all.

- 1) Prevent transactional replay attacks on forks that do not contain valid blocks.
- 2) Support more than 26 patented hash algorithms, high speed, security and scalability.
- 3) Inform the network that certain users and stakes are in a particular fork.
- 4) Enable specific users to use the POS consensus algorithm.
- 5) Enable specific users to use POW consensus algorithm.
- 6) MX Seed node technology has the ability to import and connect certain specified nodes first.
- 7) MX Masternode Technology has the ability to maintain and enhance the network's connectivity, security, and speed.
- 8) The consensus algorithm operates in a way that is compatible with POS, POW and is connected with the Oracle session. The Oracle session works in conjunction with POS, POW and various consensus algorithms in a compatible manner.
- 9) Dapp users are divided into existing users and new users, who can use the features described above, and users of other existing platforms can connect Dapp using the Factor's network.

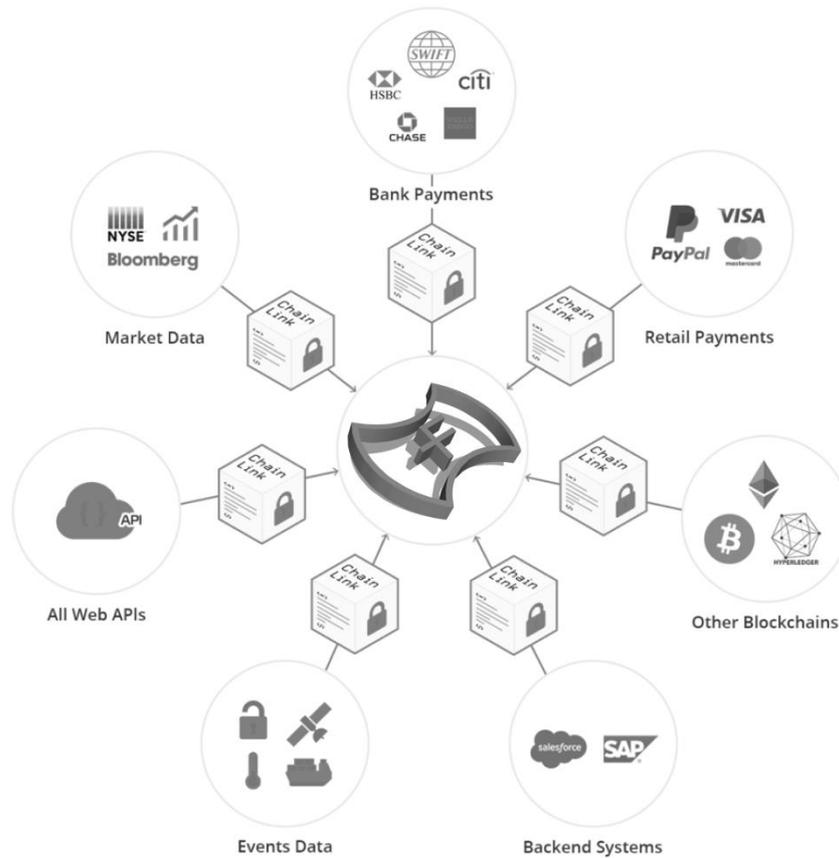
1 – 2 Distributed Ledger Application

Using the distributed ledger framework, you can combine the various programs that are in the Oracle session with a distributed ledger.

Therefore, it is useful to apply Mongolia DB, MySQL, database management system (for example), because they are frequently used in the current databases deployed by companies around the world.

Since it is possible to store a variety of information in the distributed ledger (not only applying existing smart contract) but also other kinds of information can be stored on the ledger.

The MX Block Chain Distributed ledger operates through the following process, and the Smart Contract is created with the information.



1 - 3 Contract Model

The contract model of Factor Blockchain is based on the contract model of the existing Ethereum with the application of node.js, java, though an Oracle session. However, the smart contract model in Factor Blockchain supports more than this. It supports programs such as Node.js, JAVA, go, C ++, Python, etc. and there are plans to support more programs in the future.

One of the best examples is the way that financial institutions (Visa, Paypal, Mastercard, HSBC, CITI, and America Express) create Oracle sessions that can be applied and connected to existing block chains.

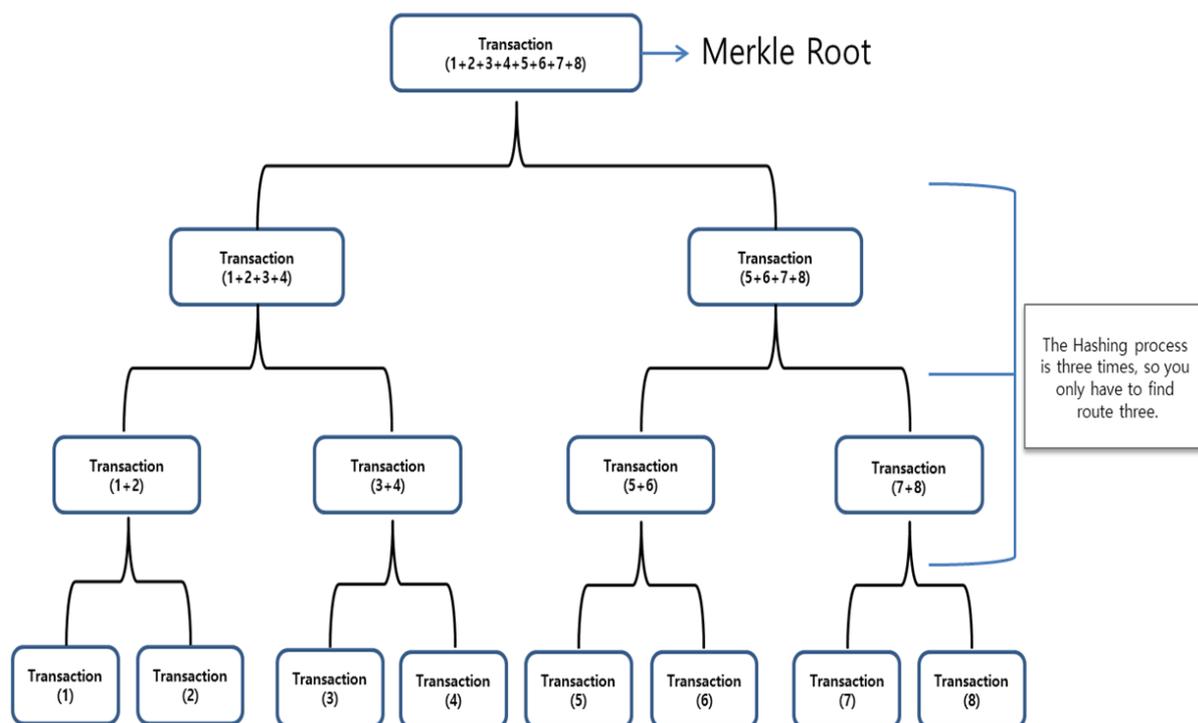
However, smart contracts cannot connect with key external resources such as off-chained data and APIs. Therefore, the current method is designed to operate middleware or on various servers in the middle format.

However, this causes problems such as speed degradation, server error, compatibility problems, and it is very ineffective to use with existing block chain.

Factor Blockchains, however, do not rely solely on intermediate middleware, but are designed to be compatible with existing programs without middleware due to the design and performance of Factor Blockchain itself.

1 - 4 MerkleTree Storage Model

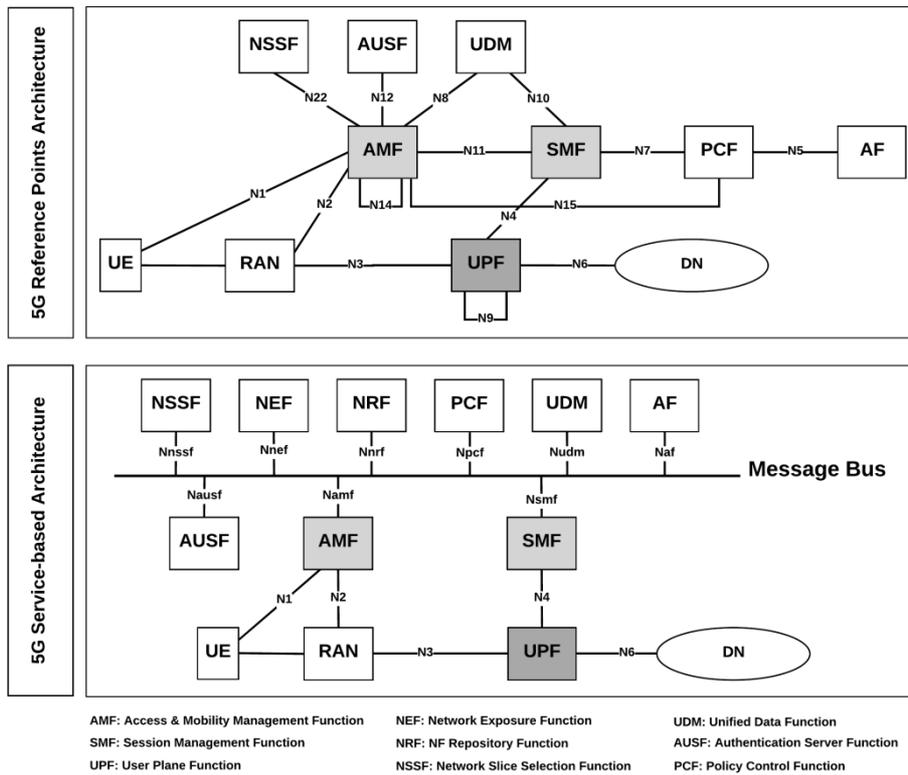
The Factor's Merkle Tree storage model is also known as a binary tree, which receives binary values from an existing merkle tree. First, refer to the following figure.



To find the value of transaction No. 5 in a transaction list (1 + 2 + 3 + 4 + 5 + 6 + 7 + 8) (in the figure), you can find the value of the hash data by locating the values through the process as shown in the picture above. The path to finding a particular transaction is simple even when the volume of transaction increases exponentially. The traditional SHA-256 method requires hashing the interaction of 2 hashes to form a single hash value of 32-bit. But in the Factor Blockchain, each of the various hash values is formed in the Merkle tree through a connection and alignment process. This means that if the existing method supports a single hash function (for example SHA-256) then the Factor block chain provides more diverse hash functions. You can also find unique number (Nonce) and TimeStamp values for each block, not just a specific transaction route, even in the validation process in the Merkle tree. Using these additional functions, the Factor block chain provides better features than to check Merkle trees.

1 - 5 Node System

Factor's system use a technologies called "Masternode". That uses POW and POS together in parallel for different tasks. The current existing blockchain are using Prove of Stake and Prove of work but separately, never both together. Factor blockchain use the two-technology at the same time. So factor node system compare to old blockchains (always depending on the hash rate) overcomes this dependency. So the factor Masternode consensus speed, connectivity, security , increases. This means the factor Patented MX data node spreading method and the seed node technology are combined. Whereas existing blockchains Masternode are simply used for network maintenance. Even without full masternode Factor node can contribute to network connectivity, by using the normal node (Wallet). Below is a picture of the core reference structure of the 5G network.



The figure above shows the structure of the next generation of mobile Network, the 5G network. (it does not show how the 5G network works).

When a user uses a smart phone with a Factor protocol application, the node mentioned above will continue to accelerate to the highest speed possible without generating Masternode and making it contributed. This is why the speed will gradually increase.

This will be the cornerstone to overcome the slowing down of existing nodes as the number of nodes increase and it will be able to support the constantly changing network environment,

1 - 6 Core Protocol

Factor Core protocol is written in C language and supports Windows, Linux, Mac, Raspberry Pie, Android, IOS, and more. This is possible because Factor's core protocol supports different systems architecture is able to support multiple programming languages that can be executed on different operating system.

The factor's core protocol is formed through a rather complex process.

1 – 7 User Authority Management Protocol

The MX blockchain is divided into two types of distributed ledger: open and closed.

Closed distributed ledgers are designed to address the security and confidentiality aspects of the blockchain required by businesses applications.

On the other hand, open distributed ledgers allow access to open distributed ones, according to user access management settings. The purpose of this application is to ensure that a variety of companies, countries, groups, and individuals can use it for their own purposes.

1 - 8 Distributed Data Exchange Protocol

, provider, agent, and owner can be operated independently. In other words, the Factor block chain has been designed to be fast, secure, and easy to use by people with the right authorities (access rights) over these diverse data.

1 – 9 Application Frame Work

For users who use Dapp (Decentralized application), Factor has Dapp Toolkit and One click Dapp. This has the advantage of reducing the inconvenience of compiling and version management in the Solidity language, thereby reducing the hassle of constantly deleting, modifying, and changing compilations and languages.

In addition, you can support languages that exist on various platforms with a single toolkit, such as Qtum, eos, neo, etc., rather than just supporting the Ethereum Solidity language.

One click Dapp makes it easy to add and modify even if you do not know the features and programming language. Therefore, these Dapps are connected in the Factor blockchain and support the same language on different blockchain platforms and support compatible network.

Factor supply developers with a language compatible with different platforms, and network. Other platform have a problem of cost, speed, compatibility and scalability but this can be solved by Factor because of Factor Toolkit and Fator One click Daap.

1 - 10 History Management Module

Distributed applications running on a blockchain allow connection with many kinds of technologies.

However, the information, records, certification systems, medical systems, etc. contained in the blockchain will need modules to manage and control the information.

For example, Factor Blockchain can be used with MRIs if the user has access to medical information from a distributed ledger of the blockchain. The MRI (medical device) machine discovers information about the human body and then transmits it to the computer. Normally, the user would only be able to draw conclusions from what is contained in the information received from the MRI machine. If you have access to the information available in the Blockchain distributed ledger book, you can not only access the data from the MRI machine, but also compare it with other medical information available and interlock it in real time to other forms of medical treatment equipment. The treatment procedure is then available immediately.

Furthermore, DNA analysis, identity authentication, and a whole variety of blood analysis can be completed, saving enormous amounts of time and money. This is only one example of the incredibly wide range of real-life application that Factor Blockchain can have.

1 - 11 Data Base

In the past, databases were only what information a single user collected and gathered, such as government agencies and libraries.

By applying this idea, various search engines such as Google, Naver, and Yahoo have been created and have raised the quality of our lives. However, there are various problems regarding the quality, quantity, and accuracy of the information provided in these search engines. Information is often given to the user dependent on advertising revenue, popularity of the search terms, and much of this information is able to be edited by anyone. This can lead to misinformation being spread rapidly and on a large scale.

However, it is possible to solve these problems through the consensus algorithms of blockchains, accurate information selecting processes created through algorithms, non-individual knowledge, etc. This information system selecting process is more accurate, and can also be used by artificial intelligence. This is the only solution to ensure accuracy of information, as well as saving time and expense, and can be used by real people in their daily life. Factor Blockchain has been designed with the aim of extending the concept and specification of the database mentioned above to apply all of the data of the blockchain user and enable it to be used with existing programs (Mongo DB, Mysql, MSsql).

2 Connectivity

2 - 1 Ethereum

Currently, Ethereum is trying to achieve scalability by adding side blockchain, plasma network, and Leiden network, and is trying to increase speed through sharding, and is trying to solve various problems through it. But there are many gaps in speed and scalability.

However, when connecting to the Ethereum session of Factor Dapp the results shows that the technology of the Factor supports, diverse development language and expand the scope of the supported programs in the Oracle session, and even increase speed through the linking of 26 or more hash functions.

Just like with Ethereum, many other coins using the existing Ethereum session can also be connected to the DAPP of Factor Blockchain. In general, the higher the number of nodes is, the slower the speed to update them is. But with factor the situation is different because the higher the number of nodes is, the faster the speed becomes, due to Factor Blockchain's patented MX-Node data spreading technology (node update technology).



2 - 2 Eos Connectivity

EOS, uses a coin called RAM to pay for the processing fees of its decentralized computing. It is a key factor in running Dapps and maintaining the 21 BPs overall network speed. If you connect to the EOS module of the Factor Blockchain, you can decrease the running fees of running the 21 BPs. And it can also increase the speed and efficiency in the case of Ethereum. We were able to convert the method used for 21 BPs from the high-cost, low-efficiency to low-cost, and high-efficiency.

In terms of connectivity of the hash function, since of Factor's smart contract supports various languages including Solidity, when they are used in connection with the Oracle session, it is compatible with block chains that use versions of Solidity languages such as Factor modules.

Thus, numerous tokens based on EOS will also be able to run on Factor blockchain by connecting to the DAPP in the Factor blockchain.

2 - 3 Neo Connectivity

During Neo ICO, it took 25 minutes for the Neo-blockchain to go from block number 1816381 to block number 1816382. Which is a very long time. Neo use the DBFT consensus algorithm which stops running when $(n-1) / 3$ of the nodes are corrupted. This situation was caused by overloading the node that are responsible for the transfer of the transactions. Connecting the Neo base with Factor Blockchain Dapp not only compensates for the weaknesses of the consensus algorithm above, but also increases the transmission speed and connectivity between each of the MX nodes. It can also overcome the problem of DBFT consensus algorithm and supports many programs in Oracle session through the function of Neo solidity smart contract. Therefore, it is possible to overcome the defects of Neo with a simple connection to Mx Blockchain.

2 - 4 Significance of Connectivity

Because MX blockchain can connect all of the existing coins/token using Factor Dapps, it interconnects the separated blockchains making all the different functions available..

As a result, it will not only be available in many industries during the fourth industrial revolution, such as IOT, self-driving cars and artificial intelligence (AI), but will also greatly help industrialization by enhancing scalability as well as reducing costs for speeding up things such as Lightning Blockchain and Plasma Network.

3. Speed

3 - 1 Speed hash function

GPU and CPU hash function are included.

Some of Factor hash function are designed to increase the speed. they are divided into A, B, and C. This is the foundation for the 32-bit and 64-bit operating systems used by Apple, Windows and Linux and supports the performance of computers with GPU, CPU multi-threading processing.

Hash function A

Hash function A has a 1024 bit state and operates on input blocks of 512 bit. The processing of input block consists of three steps:

- XOR in half the input block state.
- Apply a sequence (encryption function) without a 42 round key to that state. This consists of 42 iterations:
- Divide the input into 4 bit blocks out of 256 and map each via one of the two 4 bit S-boxes. The selection is made with a round dependent key schedule of 256 bit, and each input block is combined with the key bit. The results are mapped through the 5 → 4 bit S-box.
- Mix adjacent 4 bit blocks using the separation code of maximum distance for GF (24).
- Set the 4 bit block to be adjacent to the different blocks in the next round.
- XOR the input block.

The resulting digest is the first 224, 256, 384, or 512 bit of the final value of the 1024 bit. SSE2 command sets are suitable for implementing bit-division and provide a speed of 16.8 cycles per byte.

Hash function B

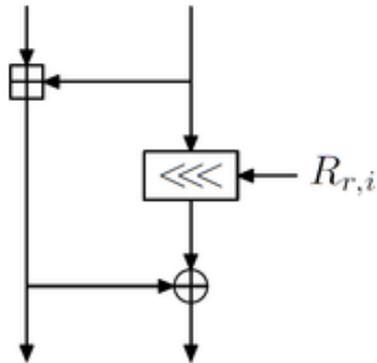
The internal state the B hash function is twice of the output size and uses a powerful message extension and a modified field in the compression function. This operating mode is safe from regular attacks. The most important component of the B hash function is the message extension, which is designed to provide a minimum length. This prevents differential decryption of the B hash function. The B hash function features small scales of parallel processing within the compression function, which can be used to create an efficient implementation using vector commands. These features are available in many popular architectures (SSE on x86 , AltiVec on PowerPC , and lwMMXt on ARM).

OS mode :	32 bit	64 bit
Core2 에서 의 16 번째시 속도		
16 번째시- 256	12 cpb	11 cpb
16 번째시- 512	13 cpb	12 cpb

Hash function C

Hash function C supports size of internal state and size of random output of 256, 512 and 1024 bits.

When Intel Core 2 Duo requires 6.1 high cycles per byte for output size of 64-bit mode, the core of threefish function in hash function C is based on the MIX function. The MIX function converts two 64-bit words by adding another constant and XOR. The UBI Chaining mode combines values of input chain, with input string of an arbitrary length and generates a output with a fixed size. The threefish function of hash C uses a combination of nonlinear addition and exclusive logic sum. This feature is optimized for 64-bit processors and the C hash function application defines optional features such as random hash, parallel tree hashing, stream cipher, personalization, and key-inducing functions.



3 - 2 Seed node

The Seednode allows the user to find and connect to a first list of nodes when the Client module is used for the first time. However, if the connected seednodes happen to be slower, then seednodes won't be used, but will be connected to the network that manage the nodes, and then synchronized.

MX Seednode technology uses these principles to calculate the speed of every node and to sequentially maintain the connection of the node with a speed priority. This is characterized by a constant increase in speed if the nodes in Factor are added. As the nodes in Factor are added, the connections to the slower nodes are suspended and the nodes are synchronized faster. Even though a slower node is added to the network, the overall network connectivity increases as more interconnected nodes are added.

3 - 3 Masternode

Traditional Masternode have worked in a lot of systemd that pays for coin when it contributes to network maintenance. But the MX Masternode can do more than that.

1. Accelerate Transaction
2. Network Attack Defense
3. Help Seednode connect spread-wise
4. Enable Private Send
5. Governance Functions
6. Low-Power, High-Efficiency Network Contribution Functions

3 - 4 MX node

The Consensus in MX node, has the following for creating and operating POS, Masternode POS.

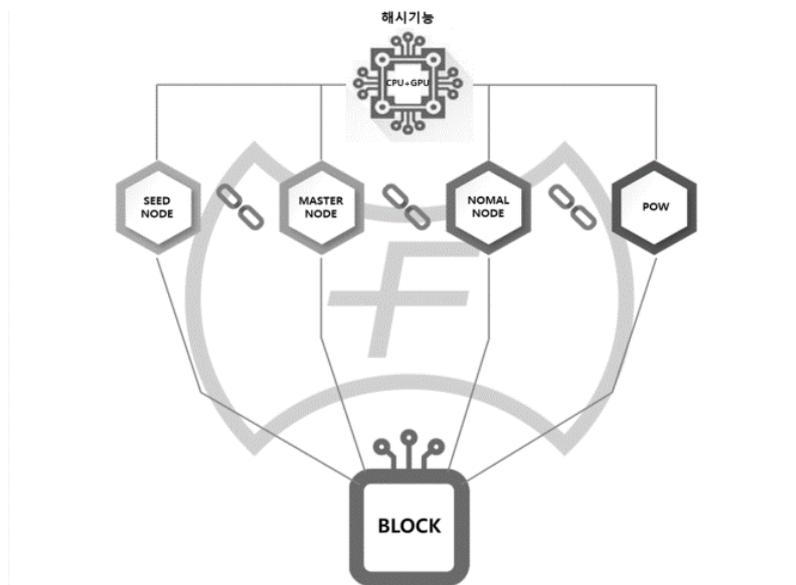
- ① **GPU - CPU** Factor uses CPU for hash rate, but also supports GPU multiple-threading acceleration too. For that reason the speed of factor data spreading node system is improving.

- ② **Seednode** - In the data spreading illustration below [Figure 1], [2], When a node to connect to another node, the connection list is automatically bringing the synchronization process so it automatically starts working. The program compares the POW speed of the Master node and of the Normal node and connect (update) them based on speed priority.
- ③ **Masternode** - In the node data spreading illustration below [Figure 1] below, [2], when a node is connected, the MX masternode plays a connection management role, which contributes to improve the the speed of the connection during synchronization process.
- ④ **POW** - A node running the POW algorithm in figure [2], as shown in [Figure 1] below, contributes to the acceleration method at connection time. This improves transaction speed according to the power of the hashrate, which increases gradually as more miners are added.
- ⑤ **Normal node** - In a illustration [Figure 1] below, [2], if a user has a high connection speed when creating a node, it can contribute to the increase in speed when importing block information during network synchronization.

Picture [1] spread method



Picture [2] MX Node system



Description [spreading method]

In data spreading as shown in Figure [1], there are four types of nodes settings : Seednode, Masternode, Normalnode, and POWnode. The 4 nodes in Figure [2] are shown in Figure [1] the fastest node is selected by the system.

First of all, the low-power GPU acceleration of the MX blockchain is used to accelerate synchronisation of the Factor's Node system. Seednode's function is used to get a particular node list and automatically connect at the time of synchronization, and to have all nodes connected at the same time as they are connected to each other in a ranking-priority manner. After this connection, the Mastnode secures connectivity between all nodes. It is connected primarily based on a speed priority and through the ability to maintain inter-node connectivity. Finally, the POW accelerates the speed so that all nodes can be connected to the highest speed node. Therefore, Factor's MX node has the characteristics of increasing the network speed proportionally to the number of additional high speed nodes added to the network. This means that as the number of nodes increases MXnode has the ability to deliver high-speed for a lower price than the node design used by EOS 21BP.

3 - 5 Bootstrap

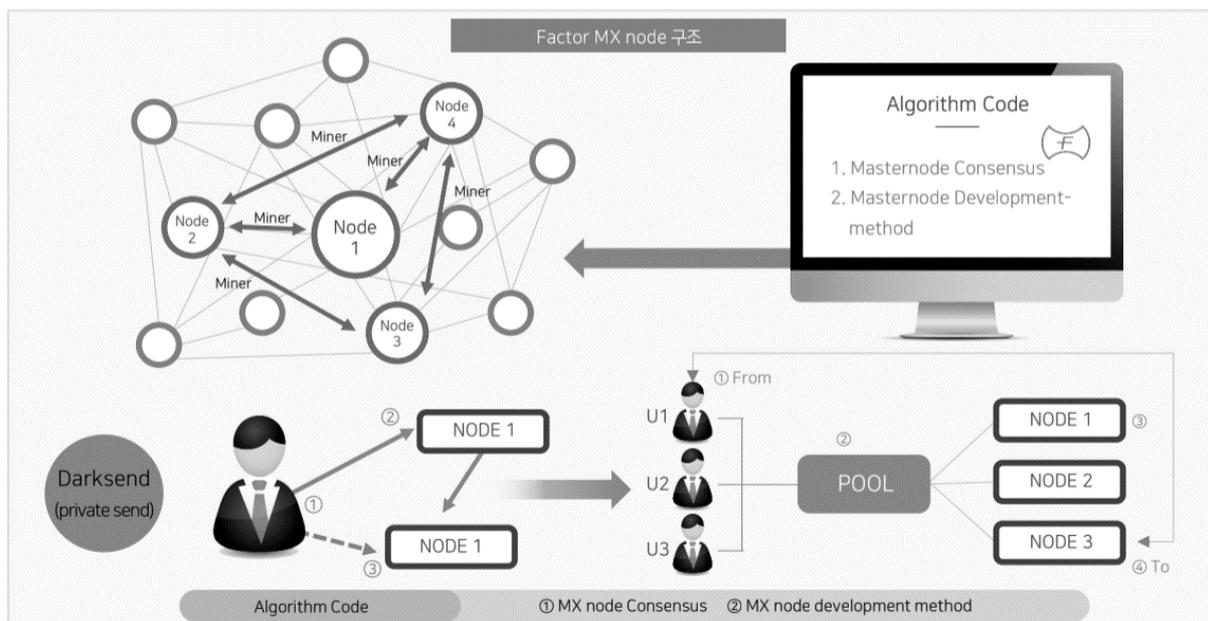
Existing distributed ledgers can be downloaded and applied in advance. Bootstrap has the advantage of reducing synchronizing time since it allows you to skip the process of obtaining blockchain information by compressing previously existing blockchain data. MX Bootstrap technology provides the advantage to check for falsified files using the SHA-256 function provided by MX Blockchain.

4. Security or safety

4 - 1 Secp256R1 key

The traditional blockchain method of Secp256k1key uses the ECDSA (Elliptic Curve Digital Signature Algorithm) cryptographic algorithm that belongs to the ELC (Elliptic Curve Cryptography) method. In the existing system, secp256k1 curve is used as a parameter of ECDSA, and secp256k1 curve is used as one of the constants for creating the elliptic curve, which is established as a standard.

ECDSA is the encryption method, and secp256k1key is the aggregation of constants that will be substituted for this number of encryption. Therefore, this type of RSA encryption technology is used in various real-life situations, including WIFI password formation, ECC in JavaScript, Windows operating system CD-key, SSH-key, etc. But the problem here is that the value of keys is divided into secret keys (key that can be encrypted) and public keys. Using this encryption method if quantum computers appear, ECC and RSA cryptography cannot be used in the future because of their ability to predict and decode the secret key value. Therefore, MX blockchain uses a more secure **secp256r1** method to protect against the computational power of quantum computers.



4 - 2 Private Send

The principle of the Private Send feature is simple. This function ensures anonymity by sending transactions across multiple people and making it unclear who sent them. When the user enters the amount to be remitted in the wallet to proceed with the transaction and press the Private Send button, your wallet divides the amount entered into smaller units (unit: 0.01, 0.1, 1, 10). Your wallet is then sent with messages to nodes (masternodes) with special configuration settings on the network. At this point, the message values sent to the masternodes are entered and the masternodes randomly exchange transactions with each other and mix. Each masternode mixes the information entered by the user, sends the contents to the masternode user's wallet and also sends to itself the amount of money that they want to transfer, which is

sent to an address formed in a random format. To make the funds completely untraceable, the wallet repeats these processes to any number of times the amount can be divided, with each time the process being known as a "round". There are as many rounds completed as possible before the transaction is formed. In addition, the mixing process takes place out of the user's sight, and in the masternode received will not contain any personal information. Therefore, it is impossible to track the masternodes that have been mixed.

4 - 3 Merkle Tree Application Security

In the previously described structure of the process, the former block + current Merkle Tree + Time Stamp, and Nonce values are added to form one block in which the blockheader is configured. The MX Block's Block Time Stamp and Nonce values are applied to provide greater security of the MX Block structure. The timestamp has the ability to convert the current time value to UNIX value for recording. Here, the time stamp does not just record the time of the block, but it also contains a unique code within the Time Stamp which Factor can use to verify activity. Thus, the ability to check and inspect the unique statements of any falsified blocks is an advantage, and the conversion of the specified nonce value reduces the time required to examine the data in the structure of the Merkle Tree. Therefore, the MX blockchain does not rely solely on the existing Merkle tree structure, but also allows for an increase in speed and security of inspecting any potential falsified blocks.

4 - 4 Masternode Network DDOS Attack Security

High security levels are provided by maintaining a masternode network. For network attacks such as DDOS attacks, network traffic attacks, and so on. If a hacker makes a DDOS attack on one node, through a traffic distribution system, a traffic rejection system, and a governance system the node is able to handle the situation. Even if it attacks the entire node, the node is able to respond with firewalls on the master node, specific port connections.

The node is able to prevent a DNS server traffic attacks, high traffic capacity, and modification of the network system with its a governance system.

4 - 5 Oracle Session Security

The environment outside the blockchain is called the Oracle session. This is to make sure that the programs that exist in the Oracle session are connected to the blockchain, and that the commands run on the blockchain work in a real external environment. When programs such as JAVA, NodeJS, MongoDB, and MySQL that exist in the Oracle session are connected, they are tested for security and operability inside the Factor Blockchain. Therefore, when different types of programs are connected to the MX blockchain, they are linked through the filtering process between each program. Programs that have been confirmed to be safe when MXblock is linked will be officially supported by the Dapp Toolkit in the future.

4 - 6 Security Hash functions

The Security Hash functions D, E of Factor Blockchain functions as follows.

- Hash function D

The block conversion f in D hash function- f [1600] for SHA-3 is a sequence using **XOR**, **AND**, and **NOT** operations and is designed for easy implementation in software and hardware. This is defined for the word size of 2 and the $w = 2 \cdot l$ bit. Major SHA-3 submissions use 64-bit words, $l = 6$. An array of $5 \times 5 \times w$ bits in which the status can be considered. Using the Little Endian bit numbering convention and row key indexing, let's say $a[i][j][k]$ is the bit $(5i + j \times w + k)$ of the input, i.e. the leading selection means the J column, and k is the bit. The index math is performed by the module in the first two dimensions and the third by the module in the w . The primary block replacement function consists of five steps, $12 + 2 \cdot l$.

Security against Attack from Quantum Computers

Typical computers (grover algorithm) require 2^d for classic brute-force attacks, while Quantum computers can perform structured preimage attacks of $\leq 2^{d/2}$. Structured preimage attacks mean a second preimage attack and a collision attack. The quantum computer also performs $3 \sqrt{2^D} = 2^{D/3}$, which destroys the crash tolerance in accordance with the specified attack date. Maximum strength is $c/2$, and provide the following high-level boundaries for SHA-3 quantum security:

Example	Bitwise security strength			
	Collision (Brassard et al.)	Collision(Bernstein)	Pre-image	두 번째 이미지 Second image
SHA3-224 (M)	74/3/3	112	112	112
SHA3-256 (M)	85 1/3	128	128	128
SHA3-384 (M)	128 자	192	192	192
SHA3-512 (M)	170/3/3	256	256	256

SHAKE128 (M, d)	$\lceil d/3,128 \rceil$	$\lceil d/2,128 \rceil$	$\geq \min(d/2, 128)$	$\lceil d/2,128 \rceil$
SHAKE256 (M, d)	$\lceil d/3,256 \rceil$	$\lceil d/2,256 \rceil$	$\geq \min(d/2, 256)$	$\lceil d/2,256 \rceil$

● Hash function E

In addition to E hash function -128 the original E hash function is considered unsafe because the 128-bit result is too small and also because of design weaknesses (originally in the case of hash E). The 256- and 320-bit versions of the E-Hash provide the same level of security as the E-Hash-128 and E-Hash-160 respectively. It is designed for applications that have sufficient security but require longer hash results.

The EHASI-160 (also known as the RIPE message digest) is typically displayed in hexadecimal digits. The following shows a 43 byte ASCII input and its E hash-160 hash.

```
E hash -160("The quick brown fox jumps over the lazy dog") =
37f332f68db77bd9d7edd4969571ad671cf9dd3b
```

The EHASI-160 works with the desired "Avalanche effect" of the cryptographic hash function (for example, changing d to c to become a completely different hash).

```
E hash-160("The quick brown fox jumps over the lazy cog") =
132072df690933835eb8b6ad0b77e7b6f14acad7
```

The hash value for a zero-length string is:

```
E hash-160("") =
9c1185a5c5e9fc54612808977ee8f548b2258d31
```

5. Scalability

5 - 1 Factor Ledger Module

The Factor Ledger Module offers a wide range of settings to design business application. The functions currently available have various uses in real life such as animal tracking, bonds, auto auctions, digital property, fund opening, trade credit, gamse, food management, identity certificates, exchanges, car life cycle managements, car maintenance cycle management.... Factor Ledger makes it easier for real-life users to build the system they need for their businesses even if they don't know much about blockchain. It also uses blockchain technology to provide cost savings, high technical skills and a diverse network of BA models, and has high efficiency benefits at low cost. This can be applied to existing programs for a whole variety of applications.

5 - 2 Graphene

The Factor Graphene module is a cost-effective blockchain solution for exchange of distributed assets, industry-required performance and scalability, active account acceptance, wages payments, compensation programs, user assets, stakeholder-approved project funding, convertible assets and delegated Proof of stake (DPOS) consensus. User-asset (UIA) systems allow intangible assets to be incorporated into tokens, and assets to be divided assets. In addition, if you want to use the blockchain for company usage , the access can be restricted to only authorized users, thanks to active user accounts management used in a variety of business application (BAs). Graphene modules use these functions to enable the overall function, operation, and management of a company in real life. Therefore, factor can solve the difficulties of traditional blockchains for business applications.

5 - 3 Finance

- Distributed ledger of Swift System.
- IBM Mainframe (integrated bank computer system).
- HP UNIX (next generation system).
- HP Superdome (Integrated Next Generation System).
- IBM UNIX (WINS).
- Factor Dapp and Toolkit used in the financial sector.

5 - 4 IOT

AMQP(Advanced Message Queuing Protocol)

CoAP(Constrained Application Protocol)

DDS(Data Distribution Service)

JSON-LD(JavaScript Object Notation for Linked Data)

MQ Telemetry Transport

Near-field Communication

Supervisory Control and Data Acquisition

6LoWPAN, HomeKit, IoTivity, LoRaWAN, Zigbee

5 - 5 Public Institutions

For civil documents, electronic votes, and identification document, the Factor Blockchain System can be used to reduce overall costs and improve security.

5 - 6 Logistics, Distribution

CRM, SCM, ERP, Logistic System

QR code can be searched easily in a distributed database designed for the distribution and logistics industry.

At any time all users have the same distributed database.

5 - 7 Manufacturing, Production

Blockchain technology will be applied to robots used in manufacturing and production. This means that the robot's language functions can be applied to the blockchain distributed book.

6.Dapp(Ecosystem)

6 - 1 Factor Dapp

- ① Summary and Overview of Ecosystem
- ② Technical ecosystem
- ③ Governance ecosystem
- ④ Application ecosystem
- ⑤ Operating methods and how to participate

6 - 2 Factor ToolKit

Factoremix and Oneclick Dapp are supported.

Factoremix provides integrated support for the languages used in the factor's Ethereum, Neo, and Eos modules.

In addition, Oneclick Dapp allows the user to be able to create a Dapp just by pressing a button.

So a Dapp application can be created even if the user is not familiar with blockchain development languages.

Since the Oracle session will be added in the future, various programs will become blockchain compatible by adding the Factor ToolKit. It would allow every developer to connect to Factor blockchain providing them with greater scalability and famous development tools.

6 - 3 Supporting Languages and Programs (Oracle section)

① Go Language

Developed by Google and used to connect to the current blockchain, the language design and Go's syntax are largely similar to C. Code blocks are enclosed in brackets and have common control structures, including "for", "switch", and "if". Unlike C, a semicolon at the end of a line is an option, not a requirement. A variable declaration is an option that is created differently. Factor has command line conversion it is explicit : "Go" and "Select" keyword are used to deal with parallel programming.

The new types include map, Unicode string, array slice, and channel for internal thread communication, and Go is designed to be compiled quickly even on basic hardware. This allows files to be compiled from a low-spec computer. In order to confirm the files are not counterfeit and not infected by a virus. The user can use a pure client file generated by the compiler. So Go language is used as garbage collection.

And Go's structural rules related to parallel processing (channel and optional channel inputs) are taken from the CSP of Tony Hoare. Class inheritance, Generic, assertion, method overload, and pointer operation are not included in the Go, among the features in C++, JAVA. As a development language for Go, you can supplement your deficiencies with the languages of other supported Oracle sessions.

② JAVA

Object-Oriented Programming (OOP), which is used to identify the units of several independent units, or "objects," that is, from what you see as a list of commands. Each guest user sends and receives messages, data, and functions to facilitate program changes. Therefore, it is mainly used for large-scale software development and is easy to maintain and repair. While intuitive code analysis is possible, creating too many object is not required for actual development. Actually it

prevent the source code to work well in the program. In Java, the javac.exe program acts as a compilation function, and Java can run on any operating system that has Java Runtime Environment (JRE). Starting with JAVA version 8, Lambda Expressions is supported, easier to filter, map and aggregate elements in the collection, and more concise. Functional programming C++ has the ability to remove objects that are created in memory that require developers to write their own code, but that are not used automatically. There are many different applications that can be implemented and multi-threaded easily.

③ Node.js

We use the V8 engine because of the traditional JavaStrip problem. This will continue to be updated and feature-added by Google, and we're using the DRIVEN method of events.

This is the way it works, only when you send data to the input device when you trigger the event. Because the web server "connects" only to the events that occur, you can minimize resources, and most web servers will continue to run until the event occurs, resulting in increased latency and memory consumption. In addition, non-blocking I/O is used. The traditional approach takes over all the memory when a Read/Write event occurs and all of the user processes are stopped, and all of the memory is used for that event, but Non-blocking I/O converts the memory as soon as the event starts so that it can be ready for other tasks. This results in faster speeds than traditional syncs and less memory autonomy. It supports single threads and has the advantage of having the same language as the client and server.

④ C, C++ Language

C language is very simple, it can be used starting from the smallest bit of memory, the smallest unit of software configuration, to memory management, and to the advanced concept of (Object-Oriented programming) OOP. In fact, the OS APIs that are at the very bottom are almost all C-language platforms nowadays, and most of the infrastructure software are written in C, providing a link to other languages.

If you look at the steps from the low level, the machine code will vary depending on the machine, and there are several versions of the assembler language, such as Intel / AT&T...

But in the end there is a commonality in the C language.

In addition, if you go to the upper part of C language, it is divided into various parts such as C ++ / Java / C # / Objective-C / Python.

The structure can be represented as a double cone type in which two cones are attached to each other vertically at the apex..

This alone is enough to understand the importance of C language. Therefore, even if you do not actually use C language but use other high-level languages, you can use it interlocked. In this case, a large number of languages provide FFI with C language.

7. Game

7 - 1 Unreal Engine

7 - 2 Unity Engine

7 - 3 Havoc Engine

7 - 4 Cry Engine

7 - 5 Jupiter Engine

7 - 6 Gamebryo Engine

7 - 7 Source Engine

8. Security Solution

8 - 1 Factor Vaccine

8 - 2 Factor Online Security

8 - 3 Factor Business Solution

9. Glossary

MX(Multi X) Block chain

MX Distributed ledger

ECDSA(Elliptic Curve Digital Signature Algorithm)

A cryptographic algorithm used for digital signature purposes belonging to the Elliptic Curve Cryptography (ECC) method. ECC was proposed in 1985 as an alternative to RSA cryptography. Longer cryptographic keys enhance security, but because of reduced cryptographic speed, the use of ECC instead of increasing the length of the RSA encryption key is on the rise.

secp256k1

Use secp256k1 curve as a parameter for ECDSA. secp256k1 curve is established as a standard and is a constant set for elliptic curve. This is called "Elliptic curve 256-bit domain parameter." It consists of SEC (Standard for Efficient Cryptography) + p (Parameter p over F_p) + 256 (Field Size p) + k (Koblitz Curve variant) + 1 (sequence number).

Secp256r1

Use secp256r1 random curve as parameter of ECDSA. Like secp256r1, it uses a curve that uses r(Random Parameter) instead of k.

MX Node

MX nodes maintain network connectivity, which solves the problem caused by a small number of mining concentrated phenomena, which can create a more robust and secure distributed blockchain through the interaction of the mining server and nodes.

Merkletree

Factor MX blocks form an interactive hashtree, or Merkle tree, and if you know the hash value of one of the child nodes even when you're only trying to verify a portion of the data, it has the ability to validate data on all of the child nodes, and the blocks that are created are constantly linked to the previous blocks, creating a distributed volume, and therefore a secure blockchain.

Hash function

A hash function is a function that maps data of a fixed length to any data of any length. The fixed length value of the hash function is called the hash value. This value is also called hash code, sum, checksum, etc.

Because hash functions are usually implemented with algorithms that are not very complex, they consume relatively less system resources such as CPU and memory. The same output value is guaranteed for the same input value, which is distributed as evenly as possible over an even range. There is also a hash function that allows you to have different output values for the same input by entering a separate value from the original that creates a hash value for a special purpose.

Because the hash function is often narrower in the output than the input range, there are rare cases where the same value is output, even though the input is different. The detailed principle is to use the pigeonhole principle. This is known as a 'collision'. In principle, the hash function should not be able to calculate the collision intentionally, except for these unavoidable conflicts.

With these characteristics, a hash function designed for a variety of purposes is available and is very useful in a variety of areas, including:

- Data structure
- hash table (or hash map)
- Hash set
- Bloom filter
- Cache
- Search for duplicated records
- Search for similar records
- Search for similar partial strings
- Geometric hash
- Tamper detection/error detection

Lately, the default library often includes a hash function, so you can extract and use the hash value directly without having to implement it. However, older languages need to be addressed by installing or implementing an extended library. For Python, a hash function must be executed in order to put a class into a dictionary, and a comparison function (cmp) also must be implemented with the hash. If a hash function is not implemented, replace the address value of the object with the hash value.

Famous hash algorithms include Message-Digest Algorithm (MD) and Secure Hash Algorithm (SHA). Each algorithm improves the hash function due to serious hash collisions, etc. and is passed in MDn, SHA-n format in the order announced. However, SHA-2 is an exception and SHA-256 and SHA-512 are called SHA-2 families together. As of 2014, the latest versions tend to use MD6, SHA-3, or the default hash function, which is usually included in the library provided by your language.

ECC (Elliptic Curve Cryptosystem Technology)

ECC is a cryptographic technique using the mathematical properties of the "Elliptic Curve" or, "Elliptic Curve over Finite Field." ECC was proposed in 1985 as an alternative to RSA cryptography. Longer cryptographic keys (indicated in bits) enhance security (which requires a lot of time to decrypt), but because of slower cryptography, RSA is used to enhance security and ECC is used instead of increasing the encryption key length. In other words, ECC indicates the same cryptographic performance (time spent decrypting) with a small number of encryption keys. For example, 3072-bit RSA and 256-bit ECC have the same cryptographic performance. A smaller bit count improves cryptographic performance because operations can be processed faster. In other ways, even though the CPU performance of the crypto-operating device is low, you can maintain it.

10. Conclusion

As we have seen in this white paper, we will summarize the technology of Factor MX blockchain as follows.

Below.

- 1) Factor MX blockchain is a secp256r1 method in the main block and has developed and applied more than 26 new hash algorithm functions. If a new technology, quantum computer, is available, it is designed to be updated.
- 2) Factor MX Blockchain technology is designed to be compatible as well as connected to existing coins.
- 3) Factor MX blockchain also has a half-amount function, such as bitcoin, but its high hash performance reduces power consumption to one hundredth of bitcoin and is mined.
- 4) It is compatible with the Pow and Pos functions and is designed to allow masternode mining so that anyone can mine.
- 5) The node selection method of MX blockchain is a patented spread dispensing system and is designed as a priority node selection method, which can be significantly faster than the conventional node selection method.
- 6) Because quantum mechanics is installed in the distributed ledger of blockchain, Oracle section has excellent security.
- 7) MX blockchain is a self-developed main block chain that has a DAPP, which allows for numerous scalability.
- 8) MX blockchain is designed to develop toolkits with Factor-DApp so that not only new developers but also users can easily access to and utilize them.
- 9) Factor MX blockchain has 7 kinds of games and is connected to the MX blockchain using Factor DApp.
- 10) The Factor research team will continue to work hard to develop Factor MX blockchain so that it can be used in all industrial systems by combining MX blockchain and artificial intelligence(AI) to increase productivity, reduce costs, and become a safe industrial society.

11. Exemption Clause

The content presented in this white paper is for informational purposes only and you should not rely upon the statement in it. We do not take any legal responsibility arising from the information set forth in this white paper. In particular, the "development technology" specified in this white paper is subject to change and is not related to any statement about performance of coins and profit from it. There are no regulatory bodies that oversee or approve the information in this white paper. Therefore, any necessary legal actions shall not be taken in accordance with regulatory requirements or jurisdictional rules. The applicable laws and regulatory requirements in the publication, distribution, or dissemination of this white paper are also not bound by the rules. The information in the white paper is subject to change. The Factor Blockchain studies steadily, so any changes to this white paper can be found in the "Modified Versions" section.