# HYCON WHITEPAPER v1.3

INFINITY PROJECT

한국어

# 요약

본 백서는 인피니트 프로젝트의 비전 개요를 기술한다. 인피니트 프로젝트는 1) 하이콘(HYCON) 코인, 2) 기업 맞춤형 블록체인 솔루션을 위한 오픈소스 인피니티 플랫폼, 3) 탈중앙화된 암호화폐 거래소 플랫폼이라는 세 단계로 발표될 예정이다. 그러나 본 백서의 주요 목적은 하이콘, 즉 SPECTRE 프로토콜을 사용함으로써 보안수준을 유지하면서도 빠른 트랜잭션 속도를 보장하는 암호화폐에 대한 세부적인 분석을 제공하는 것이다. 이하에서는 현존하는 여러 암호화폐의 난제와 한계점을 살펴보고 하이콘이 제안하는 해결책에 대해서 기술한다. 이밖에 SPECTRE 에 대한 개요와 인피니티 프로젝트에 도입하게 된 배경 및 하이콘의 기술사양에 대해서도 설명한다.

# 들어가며

"현금, 수천년동안 가장 다용도로 지속적으로 사용해왔던 인류의 기술 중 하나. 이제 향후 15 년간 저물어가다 마침내 0 과 1 이라는 디지털 흐름에 편입될 것이다." – **이코노미스트(2007)**

오늘날 온라인 및 모바일 뱅킹의 시대에서 돈은 손으로 쥘 수 있고 눈에 보이는 형태에서 인터넷상을 떠도는 디지털 숫자의 형태로 변모하고 있다. 이런 상황에서 암호화폐라는 이름으로 일련의 암호화된 디지털 코드 형태로만 존재하는 새로운 화폐가 등장한 것은 자연스러운 현상이다. 이 디지털 화폐 혁명은 여전히 신원미상으로 추정되는 사토시 나카모토(Satoshi Nakamoto)가 비트코인 기술백서[19]를 2008 년에 공개하면서 시작되었다.

하루가 멀다하고 새롭게 출현하는 암호화폐의 공통점은 바로 블록체인이라는 기술적 아키텍처를 근본으로 한다는 점이다.

블록체인은 최초의 블록(genesis block, 시초블록)부터 현재의 블록까지 시스템 내에서 이루어지는 모든 거래 정보를 기록하고 유지하는 '공유된 공공원장'이다. 이 공공원장은 연결리스트 또는 사슬처럼 블록이 연결된 구조로 만들어졌고 각 블록은 특정 시간 동안 네트워크 전파에 의해 검증된 특정 개수의 트랜잭션 내역을 포함하고 있다. 인피니티 프로젝트에서 소개하는 새로운 암호화폐 하이콘은 기존 블록체인 기술이 직면한 도전과제를 해결하고자 고안되었다. 본 백서에서 다룰 내용은 다음과 같다.

- 블록체인 기술의 개발 현황 및 해결해야 할 문제
- 인피니트 프로젝트의 목표
- 블록체인의 한계를 극복하기 위해 하이콘이 제안하는 방법
- 하이콘의 기술 사양

# 기존 블록체인 기술에 대한 논의

이 논의를 위해 비트코인과 이더리움을 중점적으로 다룰 것이다. 블록체인 기술을 도입한 구현체 중 현재까지 가장 많이 사용되고 연구된 암호화폐이기 때문이다. 블록체인 기술을 살펴볼 때 유용한 참고문헌 중 하나는 일리-휴오모(Yli-Huomo et al.)[31]의 연구이다. 이 연구는 블록체인에 대한 최근의 연구들과 블록체인 기술 기반 시스템에 내재된 한계에 대해 종합적으로 요약해 놓았다. 비록 비트코인 블록체인에 주안점을 두는 연구이지만 결과는 본 백서 논의에 대체로 적용할 수 있다. 논의에서의 주요 기준은 스완(Swan)[29]의 연구를 참조했으며 이 연구 역시 본 백서에 적용할 수 있다.

일리-휴오모와 스완의 연구에서 주목하는 현 블록체인 시스템의 7 가지 한계는 다음과 같다.

- 처리량
- 지연 시간(latency)
- 크기 및 대역폭
- 보안
- 자원 낭비
- 사용성
- 버전관리, 하드포크, 다중체인

## 처리량

대표적인 블록체인 기반 암호화폐인 비트코인의 경우 트랜잭션 확정에 10 분 이상 걸린다. 현재 트랜잭션은 초당 약 4 건씩 발생하고, 최대 처리량은 초당 7 건에 이른다. 이더리움은 초당 트랜잭션 10 건 이상을 처리할 수 있으며 비트코인 네트워크와 비교했을 때 확정 속도가 약 10 배 빠르다. 그러나 이런 블록체인의 처리량을 비자(VISA) 네트워크와 비교해보면 현재 블록체인의 한계를 알 수 있다. 비자 네트워크는 수초만에 트랜잭션을 검증할 수 있고 초당 평균 2,000 건의 거래를 처리할 수 있으며 최대 처리량은 초당 65,000 건에 이른다. 이러한 수치에서 한 눈에 알 수 있듯 비자와 같이 전통적 중앙화 결제 네트워크와 비교했을 때, 현재 가장 널리 사용되는 블록체인 네트워크들의 처리량은 현격히 적다.

블록체인 처리량을 제한하는 주된 요소 중 하나는 노드간 지연시간이다. 비트코인에 채택될 라이트닝 네트워크(Lightning network)[22]나 이더리움 블록체인의 마이크로 버전에서

사용중인 레이든 네트워크(Raiden network)[23] 등 지연시간 문제를 해결하기 위한 시도는 있었으나 아직 실행가능한 장기 해결책에 대한 합의에까지는 이르지 못한 상태이다.

## 지연 시간

앞서 언급했듯 노드간 지연시간으로 네트워크의 최대 처리량에 제한이 생기기 때문에 지연시간과 처리량은 제한요인으로서 서로 밀접한 관계가 있다. 노드간 지연시간이 늘어나면 유효기간이 지난 블록 상에서 채굴을 하게 될 리스크가 증가한다. 비트코인 네트워크를 예로 들면, 한 노드에서 생성된 하나의 블록이 네트워크 상 50%의 노드에 전파되는데 평균 2 초 미만, 90%의 노드에 전파되는데는 평균 약 13 초가 소요된다(2017 년 4 월 기준)[4]. 이더리움의 경우, 50%의 노드에 전파되는데 평균 1 초 미만, 90%의 노드 전파에는 약 10 초가 소요된다[11].

비트코인의 경우, 블록 하나를 채굴하는데 소요되는 시간(약 10 분)이 네트워크에 전파되는 시간(90% 노드 전파에 약 13 초)에 비해 길기 때문에 노드간의 지연시간이 큰 제한요인으로 작용하지 않는다. 그러나 이더리움의 경우 블록 생성 간격이 짧기 때문에 지연시간이 비트코인보다 짧아도 문제가 될 수 있다. 그래서 이더리움은 고스트(GHOST; Greedy Heaviest Observed Subtree) 프로토콜에 기반한 알고리즘을 통해 채굴자로 하여금 긴 지연시간 또는 짧은 블록 생성 간격으로 인해 생긴 병렬체인이 아닌 가장 긴 체인에서 채굴하도록 인센티브를 주는 방식을 사용한다.

## 크기 및 대역폭

크기 및 대역폭 관련하여 논의할 때 반드시 고려해야 할 두 가지 사항이 있다. 바로 블록의 체인구조 전체를 표현하는 물리적 데이터와 네트워크 상에서 전파되는 개별 블록의 크기이다. 새로운 블록을 채굴하고 블록체인 네트워크 전체와 상호 작용할 수 있는 풀 노드(full node)[1]가 공공원장 전체의 로컬 복사본을 유지해야 한다면, 이 로컬 복사본 유지에 필요한 저장용량이 블록체인을 이루고 있는 블록의 개수에 비례해야 함은 명백하다. 따라서 블록의 갯수 증가로 블록체인이 비대해지면 나중에는 소수의 노드만이 블록 처리에 관여할 수 있게 되어 더 심각한 중앙화로 이어질 위험이 있다. 또한 블록의 크기로 인해 트랜잭션 양이 제한되는 상황에서 트랜잭션 양이 사용가능한 대역폭을 초과하기 시작하면 채굴 비용이 큰 폭으로 상승할 수 있으며, 트랜잭션의 양을 늘리기 위해 블록의 크기를 키우거나 블록 생성 간격을 좁히는 등 핵심 프로토콜을 수정해야 할 수도 있다. 이러한 상황이 되면 프로토콜 업그레이드를 진행하기 위해 일반적으로 바람직하지 않다고 여겨지는 하드포크를 진행해야 한다.

[1] 모든 트랜잭션 정보를 가지고 있는 노드로 라이트 노드(light node)와 대비됨

## 보안

작업증명(Proof of Work; PoW)을 이용한 블록체인 기술의 큰 장점은 해킹하기가 매우 어렵다는 것이다. 임의의 부정한 사용자 한 명이 블록체인에 이미 기록된 트랜잭션 정보 하나를 조작하기 위해서는 해당 정보가 포함되어 있는 블록과 그 블록 이후 생성된 모든 블록에 대한 작업증명을 모두 다시 해야 한다. 이러한 방식의 공격이 성공적으로 수행되기 위해 필요한 연산자원은 전체 네트워크의 해싱파워[2]의 51%에 이른다. 그래서 "51% 공격"이라는 이름이 붙여졌다. 그러나 51%의 해싱파워가 있다면 정상적인 방식으로 채굴하는 것이 부정하게 행동하는 것보다 이득이 클 것이기 때문에 이런 식의 공격이 일어날 가능성은 적다.

두 번째로 가능한 공격 유형은 시빌공격(Sybil attack)이다. 시빌공격이란 부정한 사용자 한 명이 네트워크 상에서 채굴에 사용할 부정 계정을 다수 생성하고 해당 계정들에 유리한 방향으로 네트워크를 교란시키는 공격이다. 비트코인과 같이 작업증명 기반의 시스템에서는 블록 채굴을 위해 얼마나 많은 해싱파워를 사용하는가에 따라 네트워크에 행사할 수 있는 영향력이 달라진다. 한 시빌공격자가 채굴자 두 명으로 가장한다 해도 해싱파워 역시 둘로 분산되어야 하기 때문에 결국 공격자가 얻을 수 있는 이득은 없다.

그러나 블록체인 네트워크 상의 사용자 자금을 공격할 수 있는 방법이 없는 것은 아니다. 사용자들은 주로 중앙화된 거래소에 개인키를 보관하는데 이 거래소가 해킹당하면 공격자에게 사용자의 지갑, 더 나아가 사용자의 암호화폐에까지 접근을 허락하게 된다.

블록체인 업계의 또 다른 보안 위험요소는 코딩오류가 있는 스마트계약(smart contract)을 이행할 때 발견된다. 2016 년 6 월 17 일, 이제는 DAO 사태로 잘 알려진 스마트 계약을 대상으로 한 공격이 발생했다. 공격자는 스마트 계약 코드상의 작은 결점을 이용하여 5 천만~6 천만 달러 상당의 이더를 획득하는 스마트 계약을 실행할 수 있었다. 이 사건으로 인해 결국 논란이 되는 하드포크가 실시되어 이더리움 네트워크가 두 갈래로 갈라지며 이더리움 클래식이 탄생했다.

## 자원 낭비

비트코인 블록체인은 전기에너지, 더 나아가 환경에 막대한 영향을 미친다. 현재 트랜잭션 한 건을 검증하는데 필요한 전력은 249 kWh 이며 비트코인 블록체인에 새 블록을 지속적으로 추가하기 위해 연간 32 TWh 이상 소비하는 것으로 추정한다. 이더리움의 경우 비트코인에 비해서는 적지만 여전히 막대한 전력을 소모하며 결과적으로 환경에 큰 영향을 미친다[7]. 실제 비트코인과 이더리움 각각의 네트워크 유지에 소모되는 전력을 합하면 뉴질랜드의 연간 전력

[2] 채굴 작업에서 요구되는 연산능력, 즉 연산자원과 같은 의미. 해싱파워는 채굴 성공 확률과 비례함

소비량에 이를 정도이다. 현재 작업증명 방식의 블록체인에서 벗어나려는 양상이 나타나고 있으며, 특히 이더리움의 지분증명(PoS; Proof of Stake)을 향한 움직임이 두드러진다.

## 사용성

비트코인 블록체인에서는 트랜잭션이 블록에 담겨 약 10 분마다 발표되며, 트랜잭션의 검증을 위해 각 트랜잭션 발표 후 보통 50 분 가량을 대기한다. 실생활을 예로 들면 가게에서 물건을 고른 후 계산이 처리될 때까지 한 시간 동안 줄을 서서 기다리는 것과 비슷하다. 이런 방식을 실생활에 실시간으로 적용한다는 것은 명백히 용납하기 어렵다.

비트코인을 비롯한 현재 이용가능한 암호화폐 대다수에 관련하여 다소 우려가 되는 두 번째 부분은 익명성 또는 가명성이라는 개념이다. 트랜잭션은 공개적으로 이루어지고 블록체인 상의 모든 참여자에 의해 공유된다. 그러나 비밀 보장이 필요한 트랜잭션의 경우 이러한 공개성이 항상 바람직한 것은 아니다. 누구나 해당 데이터를 검토할 수 있어 알고리즘을 통해 참여자의 개인적인 트랜잭션 내역으로부터 데이터를 추출할 수 있기 때문이다. 부연 설명을 위해 실생활의 경우를 예로 들어보자. 한 참여자가 비트코인을 어머니에게 송금한다. 이때 트랜잭션 데이터를 기반으로 다음과 같은 사항을 확인할 수 있다.

- 참여자 및 어머니의 비트코인 주소를 통해 주고 받은 전체 비트코인 수량
- 참여자 및 어머니의 비트코인 주소 잔고
- 참여자 및 어머니와 비트코인을 주고 받은 여타 참여자들의 비트코인 주소

이렇듯 비트코인 주소 하나를 한 참여자로 특정할 수 있다면 비트코인을 주고 받은 참여자의 거래 내역을 서로 확인할 수 있으며, 이들이 무엇을 샀는지, 어떤 도박을 했는지, 심지어 누가 *익명의* 후원을 받았는지도 파악할 수 있다. 미국 FBI 에서 여러 번 증명한 바 있듯, 현재 비트코인은 진정한 의미에서의 익명은 아니다. 참여자에게 있어 비트코인의 가장 큰 단점 중 하나가 금융투명성일 수도 있다. 이때문에 연구자들은 지캐시(Zcash)에도 장착된 개인정보 보호용 영지식 암호화 기법인 zk-SNARKS[24] 등 여러 솔루션을 개발하여 문제를 해결하려 노력 중이다. zk-SNARKS 는 이더리움의 메트로폴리스(Metropolis) 업그레이드 중 비잔티움(Byzantine) 단계에도 이용된 바 있다.

## 버전관리, 하드포크, 다중체인

블록체인 상 포크[3]의 제일 큰 문제는 "합의나 보안의 손상"이다. 이해를 돕기 위해 극단적인 예를 들어보자.

- 온 우주에서 이용 가능한 컴퓨팅 파워의 100%를 사용하는 심각하게 비대해진 블록체인이 단 한 개 존재하는 경우
- 이와 대조적으로 서로 경합하는 체인 100 개가 온 우주에서 이용 가능한 컴퓨팅 파워의 1% 씩을 균일하게 나누어 갖는 경우

첫 번째 경우에서 "51% 공격"에 성공하기 위해 정상 노드가 유지하고 있는 체인을 장악하려면 실제로 온 우주에서 이용 가능한 컴퓨팅 파워의 51%가 필요하다. 그러나 컴퓨팅 파워가 분열되어 있는 두 번째 경우에는 온 우주에서 이용 가능한 컴퓨팅 파워의 0.51% 만 있어도 각 체인을 손상시킬 수 있다.

블록체인은 정상 노드의 컴퓨팅 파워를 모두 합한 수치가 악의적인 노드의 컴퓨팅 파워를 모두 합한 수치보다 높도록 만들어 시스템을 유지하는 합의에 의존하고 있다. 포크로 인해 각 체인이 끊기고 컴퓨팅 파워가 감소하면 공격 성공을 위해 요구되는 자원의 관점에서 더 적은 엔트리 포인트만으로도 공격에 성공할 공산이 크다.

하드포크는 프로토콜 합의의 손상에서 비롯된 또 하나의 달갑지 않은 결과이다. 주어진 블록체인 생태계 내 투자자 사이에서 발생하는 이념적인 차이로 인해 블록체인이 분기 또는 포크될 수 있다. 일례로 확장성(scaling) 문제와 신속하고 저렴한 전자현금으로 활용되지 못하는 문제 때문에 비트코인에서 분리된 비트코인 캐시와 앞서 언급한 바와 같이 블록체인의 불변성이라는 철학적 기조에 따라 이더리움에서 포크된 이더리움 클래식을 들 수 있다. 그렇지만 하드포크가 항상 논쟁을 불러일으키지는 않는다. 하드포크는 종종 블록체인 시스템에서 핵심적인 프로토콜의 변경으로 인해 발생하기도 하는데 이더리움의 2017 메트로폴리스 업그레이드를 예로 들 수 있다. 하드포크 이후에도 원래 블록체인에 작용하던 전체 해싱파워는 그대로 유지될 수도 있으나, 이념적 하드포크의 경우 각 체인은 보안성이 떨어지고 공격에 더욱 취약해진 채 서로 경합하는 두 개의 체인으로 분리될 것이다.

[3] 포크란 개발자가 프로젝트의 소스코드를 복사해서 독립적으로 구별·분리된 부분을 만들고 거기에서 수정 작업을 하는 것으로 수정 정도에 따라 소프트포크와 하드포크가 있음

# 인피니티 프로젝트 – 핵심 목표

인피니티 프로젝트를 시작하며 다음과 같은 두 가지 핵심적인 질문을 했다.

- ✓ 현재 기존 암호화폐의 한계점을 고려할 때, 시장의 니즈와 요구사항은 무엇이며 어떻게 해결책을 제시할 수 있는가?
- ✓ 암호화폐가 경제 전반에 더욱 광범위하고 폭넓게 채택 및 통합되기 위해서는 어떤 특성이 필요한가?

이러한 두 질문을 염두에 두고 각 프로젝트의 강점과 약점을 밝혀내기 위해 비트코인, 이더리움, 여러 유망한 알트코인 등 현존하는 블록체인을 철저하게 분석했다. 그러나 상기의 질문에 완벽한 답을 제시하는 프로젝트는 찾을 수 없었다.

그래서 인피니티 프로젝트 팀은 목표를 달성하기 위해 일반 대중이 실생활에서 사용할 수 있는 새로운 기술과 알고리즘을 연구하기 시작했다. 동시에 인피니티 프로젝트의 기본 프레임워크를 설계하기 시작했고 다음과 같은 5 가지 핵심 목표를 설정했다.

**인피니티 프로젝트 핵심 목표**

1. 암호화폐에 대한 시장의 실질적 수요 확인
2. 유연한 암호화폐 개발
3. 사용자 중심 블록체인 플랫폼 개발
4. 지속가능한 혁신을 촉진할 수 있는 시스템 환경 개발
5. 탈중앙화된 방식으로 암호화폐 거래할 수 있는 방법 연구

## 핵심 목표 1 - 시장 수요 확인

많은 블록체인 프로젝트가 최근 대대적인 관심과 인정을 받고 있지만 아직까지 어떤 암호화폐도 전 세계적인 규모로 전자 상거래 분야에 진출하지는 못했다. 더 정확히 말하자면 대다수의 암호화폐 프로젝트와 실제 적용 사례 사이에는 여전히 뚜렷한 격차가 존재한다. 현재 극소수의 온라인 판매자와 소수의 기타 서비스에서만 암호화폐를 수용하거나 채택한 상황이라 비트코인을 포함한 현재 이용 가능한 암호화폐를 표준 전자화폐로 선정하여 자유롭게 사용하기가 불가능하다.

이 문제를 해결하고 실제 적용 사례를 늘리며 채택 과정을 촉진하기 위해 해당 분야 및 커뮤니티의 전문가, 개발자와 협업하면 도움이 된다. 모든 사용자가 혜택을 볼 수 있는 성공적이고 시장 친화적인 암호화폐 개발을 주도하기 위해서이다.

인피티니 프로젝트팀의 두 가지 핵심 질문들 중 하나인 "시장이 원하는 사용자 중심 화폐란 무엇인가?"에 답하기 위해서 먼저 시장과 개발 양쪽의 입장에서 바람직한 해결 방법을 찾는데 필요한 핵심 블록체인 기술이 무엇인지 정의를 내려야 한다. 이에 따라 인피니티 프로젝트 팀은 새로운 암호화폐를 개발할 때 가장 우선시 해야 할 핵심 성공요인(KSF; key success factor)은 "시장이 필요로 하는 실질적인 해결책을 제공한다는 전제 하에 암호화폐를 설계하고 도입하는 것"이라는 결론을 도출했다.

## 핵심 목표 2 - 유연한 화폐

인피니티 프로젝트 팀은 기존의 많은 암호화 프로젝트에서 볼 수 있는 획일적인 통화 개발이라는 전통적인 관점에서 벗어나 다양한 통화 모델을 수용할 수 있는 유연한 플랫폼의 개념을 도입하고자 했다.

이는 하이콘(HYCON; Hyper-Connected Coin)의 탄생으로 이어졌다. 하이콘은 빠르고 비용이 저렴하며 확장 가능하고 안전하도록 초기부터 설계되었기 때문에 실제 상황에서 다양하게 활용할 수 있다.

하이콘이 기반을 두고 있는 인피니티 블록체인은 호환성이 있는 모듈식 구조로 설계되었기 때문에 구체적인 니즈에 따라 기반 기술을 쉽게 적용하고 수정할 수 있다.

## 핵심 목표 3 - 사용자 중심 플랫폼

비트코인이 불을 지핀 패러다임 전환의 가장 중요한 부분 중 하나는 안전하고 탈중앙화된 전자 거래 촉진이라 할 수 있다. 비트코인은 모두에게 열려 있으며 한때 비현실적으로 여겨졌던 은행 없는 결제라는 개념도 실제로 구현할 수 있는 가능성을 열었다.

그러나 대다수 암호화폐가 개념적인 수준에서 실제 UI 와 UX 까지 활용하는 데까지 많은 학습이 필요하다는 점이 암호화폐의 광범위한 채택에 주요 장애물이 되고 있다. 인피니티 프로젝트는 더욱 간단하고 사용자 친화적인 플랫폼을 제공하고 직관적인 사용자 월렛과 거래소 플랫폼 인터페이스를 도입하여 진입장벽을 낮추고자 한다. 궁극적인 목표는 더 많은 사람들이 패러다임 변화를 일으킬 정도로 강력한 블록체인 기술을 제대로 활용할 수 있도록 하는 것이다.

## 핵심 목표 4 - 유연한 혁신

인피니트 프로젝트 개발 중 고려했던 가장 중요한 측면 하나는 어떻게 하면 더 많은 사람, 사업, 정부, NGO 로 하여금 블록체인 기술을 활용할 수 있게 도울까 하는 점이었다. 그래서 인피니티 프로젝트 팀은 기존의 다양한 블록체인, 플랫폼, 암호화폐에 대한 연구를 기반으로 발전시킨 유연한 블록체인 개념, 즉 인티니티 플랫폼의 구현을 연구해 왔다. 하이콘은 인피니티 플랫폼을 구성하게 되지만 플랫폼의 유일한 구성 요소는 아니다.

인피니트 플랫폼 연구의 목표는 직관적으로 사용할 수 있고 누구나 다양한 방식으로 활용할 수 있는 플랫폼을 개발하는 것이다. 인피니트 플랫폼은 다음과 같은 사례에 적용할 수 있다.

- 빠르고 저렴하게 가치를 교환하는 수단인 하이콘에 기반한 안전한 암호화폐 도입
- 정보 보안을 강화하고 효율적인 데이터 저장 및 전송을 용이하게 하는 탈중앙화된 기업 장부 생성
- 암호화를 통해 보안이 강화된 거래소

인피니티 플랫폼의 잠재적 적용 사례와 혁신은 무궁무진하다. 인피니티 플랫폼은 사용자가 필요로 하는 블록체인 솔루션을 구축할 수 있도록 유연하게 조정될 수 있다.

## 핵심 목표 5 – 안전하고 탈중앙화된 거래소

인피니티 프로젝트에서는 사용자가 서로 다른 암호화폐를 탈중앙화된 방식으로 거래할 수 있는 방법을 활발하게 연구하고 있다. 현재 거래소에서는 암호화폐를 적은 비용으로 빠르게 거래하기 위하여 중앙화된 방식에 의존하고 있으나 이러한 중앙화된 방식으로 인해 사용자는 명목화폐와 암호화폐 자산을 거래소에 위탁해야 한다.

안타깝게도 거래소를 거쳐가는 막대한 거래량에도 불구, 해당 거래소에서 사용되는 소스코드는 공개적으로 검토할 수 없는 경우가 많다. 전 세계적으로 거래소에 보관된 사용자의 암호화폐가 악성 사용자에게 도난당하는 사건이 여러 번 발생했다. 지금처럼 사용자 자금과 정보가 중앙화된 방식으로 거래소에 보관된다면 해당 거래소는 지속적으로 공격의 목표가 될 공산이 크다.

인피니티 프로젝트의 향후 연구 과제로, 하이콘이 진정한 거래 수단이 되도록 하기 위하여 아토믹 스왑(atomic swap)[4]의 개념을 하이콘에 접목시키고자 한다. 하이콘은 여러 다른 암호화폐를 거래하는 수단이 될 수 있고 트랜잭션 수수료는 네트워크를 보호하는

[4] 거래소나 제 3 자의 개입 없이 서로 다른 블록체인 사이에서 암호화폐 토큰을 교환하는 것

채굴자들에게 배분될 것이다. 아토믹 스왑을 통해 다른 암호화폐의 지급 증명이 완료될 때까지 하이콘을 에스크로에 예탁할 수 있고 이를 통해 하이콘과 다른 암호화폐 사이의 무신뢰(trustless) P2P 거래도 용이해지게 된다.

# 하이콘 기술 사양

| 특성 | 사양 |
|---|---|
| 해시 함수 | Cryptonight & Blake 2b |
| 합의 프로토콜 | 스펙터(SPECTRE) |
| 체인 구조 | 방향성 비순환 그래프(DAG) |
| 블록 속도 | 1000 ms |
| 채굴 방법 | 작업 증명(PoW) |

## 제네시스 블록

하이콘의 제네시스 블록은 한국 시간 기준 2018 년 1 월 4 일 새벽 3 시 15 분(GMT+9)에 생성되었다. 깃허브(GitHub) 내 하이콘 팀 저장소[36]에서 열람 가능하다. 제네시스 블록 관련 추가 정보는 부록 A 에 수록되어 있다.

## 해시 알고리즘

본 백서의 첫 번째 버전에서는 하이콘 시스템의 유일한 해시 함수로 Blake 2b 를 언급한 바 있다. 그러나 최근 ASIC[5] 기술의 발달에 따라 채굴에 Blake 2b 사용을 중단하는 대신 ASIC 에 대한 저항력이 있는 해시 알고리즘인 Cryptonight 를 사용하기로 결정했다. 또 다른 암호화폐인 모네로(Monero)도 Cryptonight 를 사용한다. Cryptonight 가 흥미로운 이유는 해시 작업에서 의사난수를 사용하여 메모리 읽기/쓰기를 하기 때문이다. 이 때문에 Cryptonight 는 CPU 나 GPU 를 불문하고 대략 비슷한 결과물을 산출하며 표준 ASIC 아키텍처와는 호환되지 않는다.

[5] Application-specific integrated circuit 의 약자로 암호화폐 채굴에 특화된 주문형 반도체를 뜻함.

향후에는 채굴 자원의 중앙화를 막기 위해 모네로가 세운 선례와 같이 정기적으로 해시 알고리즘을 수정하여 채굴 기간 동안 ASIC 저항력을 유지할 예정이다. [43]

# 합의 알고리즘 - SPECTRE 프로토콜

비트코인 블록체인 상에서 합의 알고리즘으로 사용되었던 나카모토 프로토콜과 대조적으로, 하이콘은 합의를 유지하기 위해 스펙터라는 프로토콜을 사용한다[26]. 스펙터는 블록 간의 순서를 정하기 위해 블록 사이에 투표 알고리즘을 적용함으로써 블록체인을 방향성 비순환 그래프(directed acyclic graph; DAG)의 형태로 일반화한다. 이를테면 블록 x 내의 트랜잭션이 블록 y 내 트랜잭션보다 먼저 블록체인에 추가되거나 반대로 블록 y 내의 트랜잭션이 블록 x 내 트랜잭션보다 먼저 추가되는 식이다. 본 백서에서 스펙터 프로토콜에 대해 모두 설명할 수는 없으나 투표 규칙의 기본 개요는 아래에 기술한다.

## 투표 규칙

스펙터의 투표 규칙 논의를 위해 투표 과정을 시각적으로 다룰 필요가 있다. 한 가지 주목해야 하는 사실은 어떤 투표도 노드 사이에서 전달되지 않으며 각 노드가 명시적으로 투표를 행사할 필요도 없다는 점이다. 오히려 투표는 블록에서 이루어지며 투표 방식은 DAG 구조를 보면 알 수 있다.

스펙터 투표 과정에 사용되는 기준은 다음과 같다. 주목해야 할 중요한 용어는 $past(x)$ 와 $future(x)$인데, $past(x)$ 는 $x$ 로부터 도달 가능한 블록을, $future(x)$는 선행 블록 $x$ 를 참조하는 블록을 가리킨다. 예를 들어, 만약 $x$ 가 $past(y)$에 속한다면 $y$ 는 $future(x)$ 에 속한다고 표현할 수 있다. 즉,

$$y \in future(x) \Leftrightarrow x \in past(y)$$

또 한 가지 주목할 것은 $virtual(G)$로 표기되는 가상 블록의 past 는 DAG 전체라는 것이다.

$z$ 라는 블록이 블록 $x$, 블록 $y$ 에 투표할 때 아래와 같은 규칙이 적용된다.

1. $z$ 가 $future(x)$에 속하지만 $future(y)$에 속하지 않는다면 $z$ 는 블록 $x$ 에 투표한다.
2. $z$ 가 $future(x)$및 $future(y)$에 속한다면 z 와 past 가 동일한 가상 블록의 투표에 근거하여 z 의 투표가 재귀적으로 결정된다.
3. $z$ 가 $future(x)$와 $future(y)$에 둘 다 속하지 않는다면 $z$ 의 투표는 $future(z)$에 속한 모든 블록의 다수결 투표에 의해 결정된다.

4. $z$가 $past(Virtual(G))$를 갖는 가상 블록이라면, 즉 $past(z)$ = DAG 전체라면, $z$의 투표는 DAG의 다수결 투표에 의해 결정된다.

5. $z = x$ 혹은 $z = y$인 경우, 만일 $y$가 $future(x)$에 속하지 않거나 $x$가 $future(y)$에 속하지 않는다면 $z$는 스스로에게 투표한다.

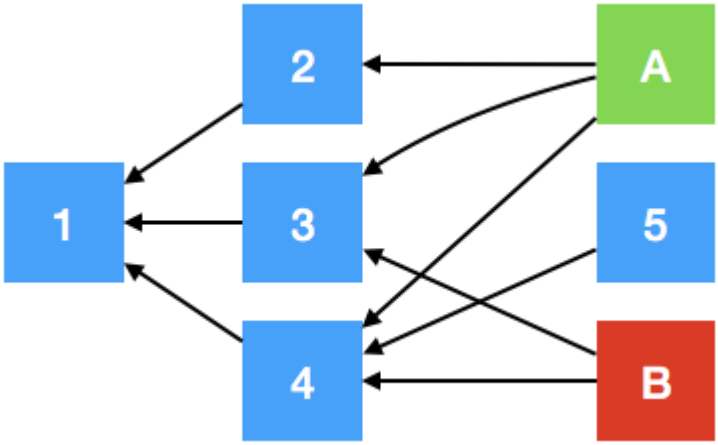
## SPECTRE 프로토콜을 DAG의 예시에 응용하기

스펙터의 응용을 잘 설명하기 위해 작동 상태인 스펙터 프로토콜의 예를 단계별로 살펴보고 투표 과정의 상태를 단편적으로 제공한다. 다음 예는 스펙터 백서를 직접 인용했다[25].

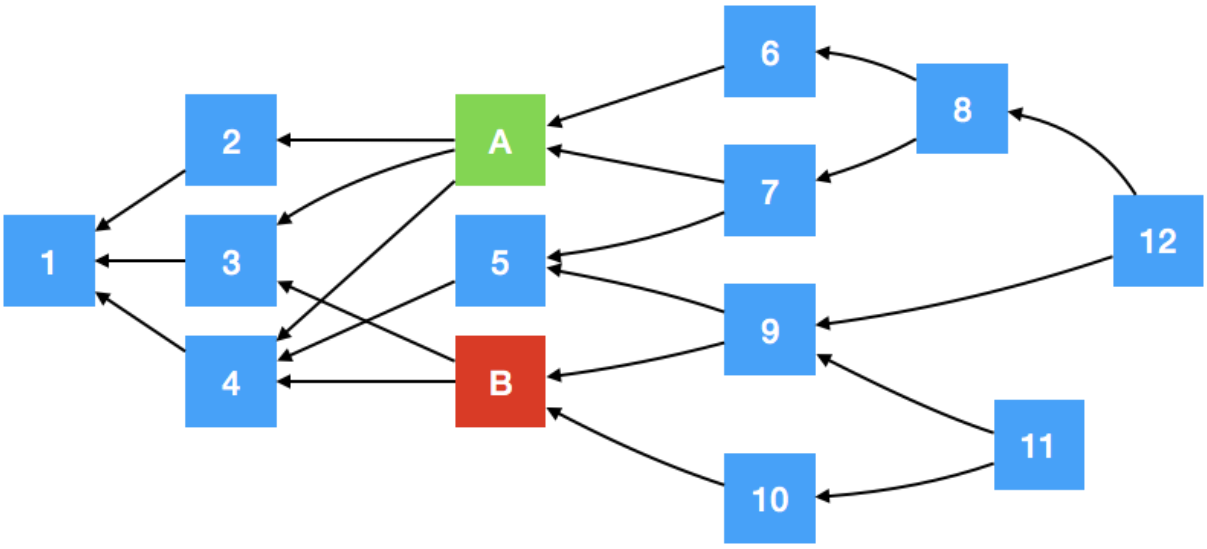
### 사례 - 이중지불

간단한 예를 들어보자. 블록 A는 트랜색션 t1을, 블록 B는 t1과 충돌하는 트랜잭션 t2를 담고 있다. 이는 악성 충돌일 수도 있고 단순히 노드 간 지연시간으로 트랜잭션이 두 번 발생하여 채굴자 두 명이 동일한 트랜잭션 수수료를 받게 된 경우일 수도 있다. DAG의 구조에 따라 이러한 이중지불을 다른 방식으로 해결할 수 있는데, 이는 두 블록의 past와 future가 다르기 때문이다.

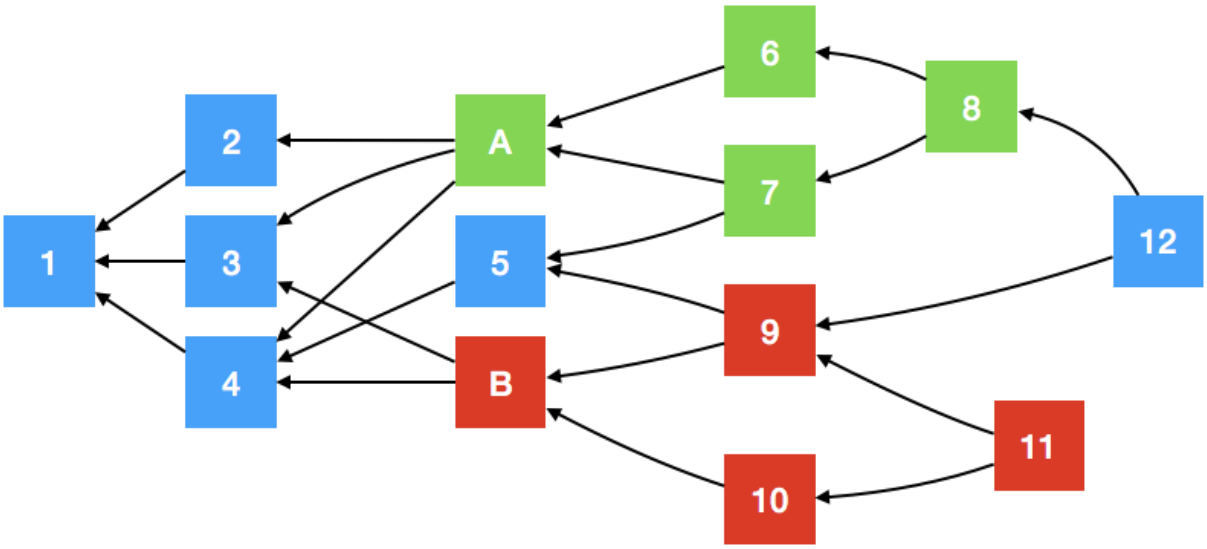
초기에 블록 A와 B가 블록 5와 거의 동시에 생성되는 경우 다음과 같은 양상을 띤다.



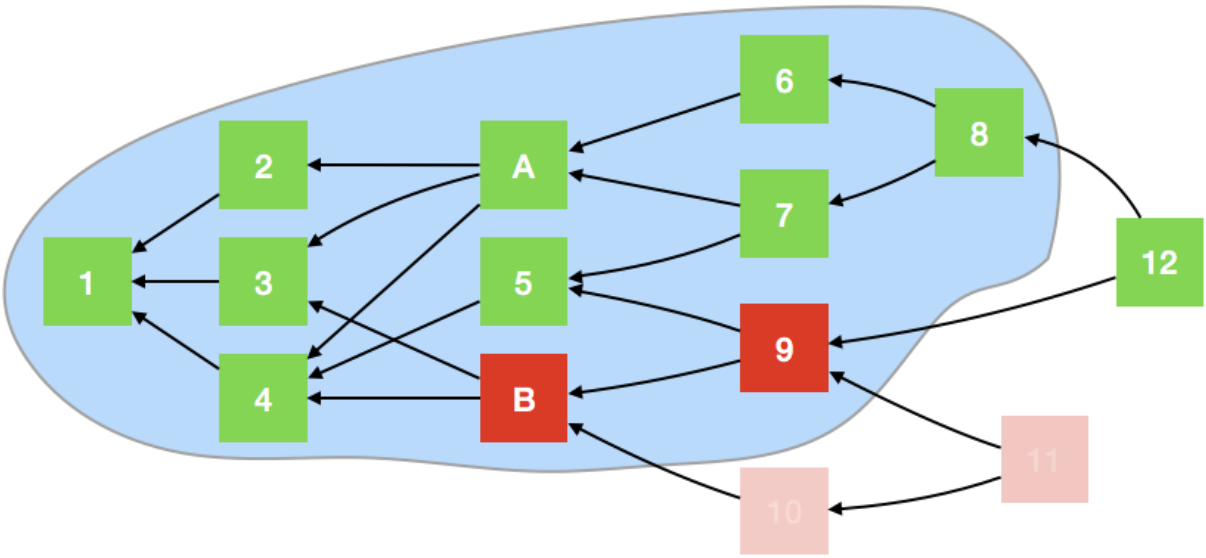

이 단계에서는 시스템에서 이중지불을 인식하지 못한다. 충돌하는 두 블록을 모두 참조하는 후속 블록이 생성되지 않았기 때문이다. 그러나 DAG의 구조가 발전하고 블록이 추가되면 이중지불이 발견되고 DAG의 구조가 분석되어 어느 블록이 선행하는지를 결정한다.




위 그림에서 블록 12 는 생성된 블록 중 A 와 B 사이의 이중 지불을 감지하는 첫 블록이다. 앞서 언급한 규칙에 따라 투표 수는 다음과 같이 계산될 수 있다. 블록 6, 7, 8 모두 블록 B 가 past 에 속하지 않기 때문에 블록 A 에 투표한다. 동일한 이유로 블록 9, 10, 11 은 블록 B 에 투표한다.



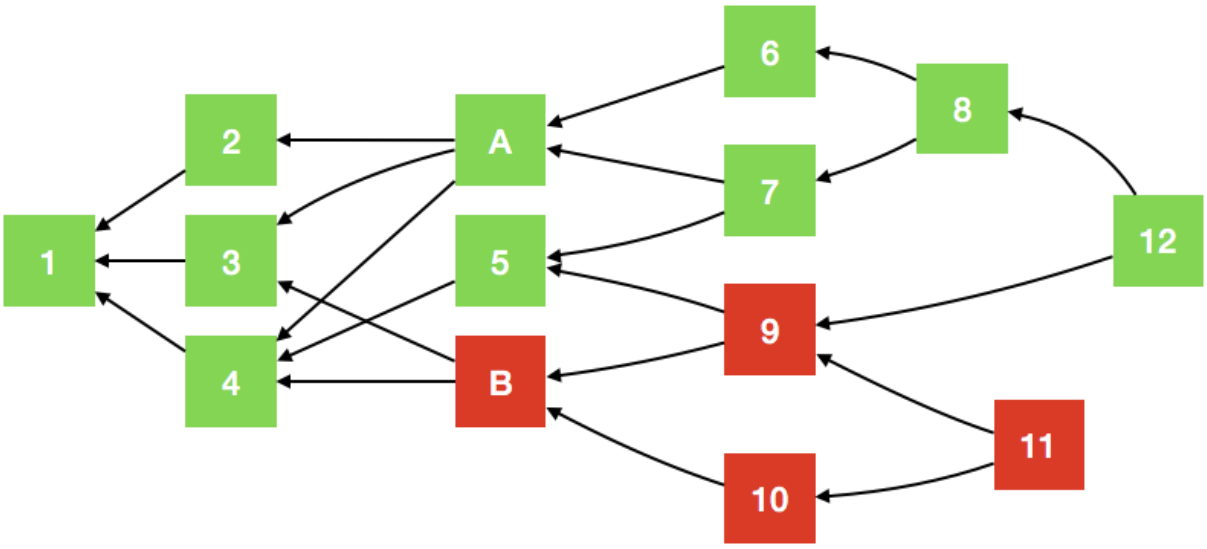

블록 12 는 $past(12)$의 재귀 호출을 기반으로 투표한다. 블록 10 과 11 은 $past(12)$에 속하지 않으므로 블록 12 의 투표를 결정할 때 고려되지 않는다. 블록 12 의 투표에 고려되는 블록 영역은 아래 그림에 나타나 있다.




블록 1 부터 5 까지는 $future(A)$ 또는 $future(B)$ 에 속하지 않으므로 $future(1)$ 부터 $future(5)$까지의 블록 대다수와 동일한 투표를 하게 된다. 이 재귀 투표의 경우, $future(1)$부터 $future(5)$까지의 블록들이 블록 A 에 투표를 더 많이 했으므로 블록 1 부터 5 는 블록 A 에 투표를 하게 된다. 블록 12 의 과거, 즉 $past(12)$에서는 블록 A 에 9 표, 블록 B 에 2 표를 투표했으므로 블록 12 는 블록 A 에 투표한다. 만일 투표수가 동점이 되면 블록 12 가 결정적으로 그 동점을 깨서 투표에 참여하는 노드 모두 블록 12 의 투표에 동의하도록 한다. 블록 12 의 투표를 결정하는 블록은 모두 $past(12)$에 속한 블록이므로 블록 12 의 표는 절대 바뀌지 않는다.

이후 DAG 에서의 투표 과정은 남은 블록의 future 블록을 기반으로 진행한다. 블록 12 의 투표가 확정되면 블록 5 는 블록 A 에 투표하게 되는데, 이는 블록 7, 8, 12 에 의한 투표가 블록 9 와 11 에 의한 투표수를 넘어서기 때문이다.

블록 4 는 블록 A, 5, 6, 7, 8, 12 에서 A 에 대한 투표를(6 표), 블록 B, 9, 10, 11 에서 B 에 대해 투표를(4 표) 확인한 후 블록 A 에 투표한다. 블록 3, 2, 1 도 마찬가지로 모두 블록 A 에 투표한다. 이러한 투표 과정을 거쳐 최종적으로 블록 A 에 10 표, 블록 B 에 4 표라는 결과가 도출된다.




스펙터의 흥미로운 특성은, 특히 위와 같이 단순한 사례의 경우, 여타 블록체인 기술처럼 가장 긴 체인을 선택한다는 것이다. 블록 A 를 통과하여 블록 1 부터 블록 12 까지 도달하는 경로와 B 를 통과하는 경로를 보면, 1→A→12 경로가 1→B→12 경로보다 길다는 것을 알 수 있다. 즉, 가장 긴 체인이 살아남는 것이다.

## DAG vs. 블록체인

하이콘에 대해 논의할 때 DAG 구조와 관련된 질문을 흔히 접하게 된다. 하이콘의 블록 생성 속도를 1,000 ms 로 설정한 결과 하이콘에 DAG 구조를 적용하게 되었다. 블록체인의 처리량을 높일 방법을 모색하던 중 분산 네트워크 상 시스템 지연시간으로 인해 블록체인이 의도와는 달리 포크될 수 있다는 사실이 명확해졌다. 이러한 포크를 회피하기보다는 포크와 이로 인해 생성되는 구조(제대로 인정받지 못하고 지나치게 강조되는 DAG 구조)를 수용하기로 했다. 선형적인 블록체인보다 DAG 구조가 유리한 이유는 블록 생성 간격을 줄여 트랜잭션 확정 속도를 높여주기 때문이다. 기존 블록체인 상 새로 채굴된 블록은 이전 블록의 해시값을 참조하여 체인 끝에 연결되는 반면, DAG 에 추가될 새 블록은 DAG 의 끝 부분을 참조한다. 이에 따라 체인이 분기될 위험을 감수하지 않고도 여러 블록이 서로 다른 노드에서 동시에 생성될 수 있다. 여러 선행 블록이 존재 가능하므로 새 블록이 동시에 추가될 수 있으며 채굴자는 자신의 블록이 고아 블록(orphaned block)이 될 걱정 없이 채굴 보상을 거두어들일 수 있다. 문제는 노드가 다른 곳에 발표된 트랜잭션을 다시 발표하여 이중지불을 유발할 때 발생한다. 그러나 스펙터를 사용하면 고아 블록이 생기지 않는 선에서 어느 트랜잭션을 거부할지에 대한 합의를 이끌어낼 수 있다.

### 인피니티 스펙터 구현

스펙터에서의 투표 과정에 자원이 상당히 소모되므로 구현은 신중하게 관리될 필요가 있다. 개발의 편의를 위해 초기 프로토타입은 파이썬으로 작성되었지만 인피니티 스펙터 구현의 최종 버전은 데이터 구조 및 메모리 관리에 대한 완전한 제어가 유지되어 더 나은 성능을 제공할 수 있도록 C, C++ 또는 Rust 와 같은 언어로 작성될 것이다.

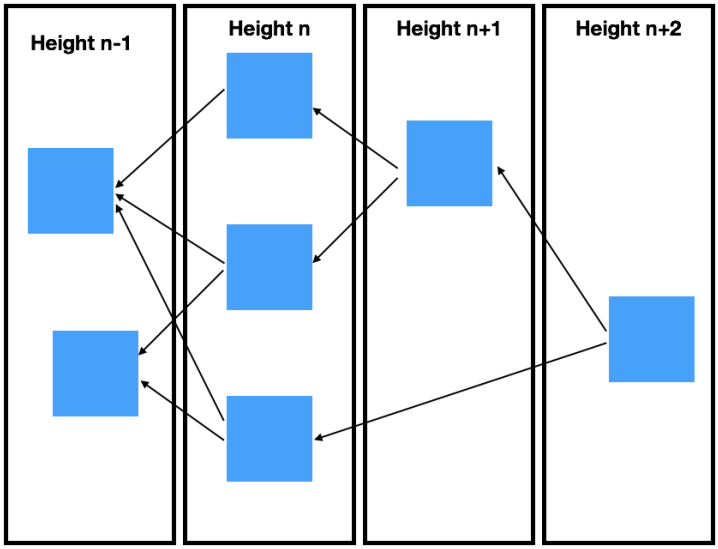
## 블록 "높이"와 체인 연결

DAG 구조를 간단히 살펴본 것에서 알 수 있듯이 비트코인이나 이더리움에서 사용되는 전통적인 블록 높이라는 개념의 의미를 약간 수정해야 할 필요가 생겼다. 비트코인이나 이더리움 블록체인에서 블록 높이란 제네시스 블록에 체인으로 연결된 블록의 수를 의미한다. 하이콘에서 블록의 높이는 좀더 일반적인 표현으로 제네시스 블록에서부터 현재 블록까지 생성된 DAG 층의 수를 나타낸다. 따라서 계산법도 상당히 간단하다. 새로운 블록의 높이는 그 블록의 가장 높은 부모 블록의 높이에 1 을 더한 값이다.

P 라는 부모 블록에 연결된 임의의 새로운 블록을 B 라고 하면

$$Height(B) = max(Height(p)) + 1;\ p \in P$$

이 상황을 그림으로 설명하면 아래와 같다. 새롭게 생성된 블록은 아직 참조되지 않은 이전 블록 중에서 가장 높은 블록을 참조하고 블록의 높이는 이전에 참조된 블록의 높이보다 1 높은 값으로 설정된다.



## 네트워크 인프라 – Node.js, Typescript

시스템 아키텍처에 이 셋업을 사용하면 Node.js 의 내장 기능으로 비동기 작업을 지원할 수 있다는 장점이 있다. Node.js 를 통해 크로스 플랫폼의 "논블로킹 이벤트 I/O"가 가능하고 개별 구성 요소는 정상 작동 흐름 이외의 작동 결과를 기다릴 수 있다. 네트워크로부터의 메시지 수신이나 사용자의 입력과 같은 특정한 상황이 발생할 때만 촉발 및 실행되며 대기 시간 동안 다른 코드가 실행될 수 있다. [20]

Typescript 를 사용하기로 결정한 이유는 이 프로그램이 본질적으로는 Javascript 인 코드에 강력한 타입 체킹을 수행할 수 있기 때문이다. 타입 형식을 지원하는 Javascript 를 사용하여 명시적으로 정의된 타입 덕분에 디버깅 프로세스가 더 간단해지면서도 Node.js 가 제공하는 비동기성을 활용하여 플랫폼을 구축할 수 있었다. Typescript 파일은 실행하기 전에 컴파일 되어야 하기 때문에 Javascript 응용 프로그램에서 디버깅할 때 으레 그렇듯 복잡하게 얽혀있는 콜백을 통해 추적하기보다 컴파일 단계에서 구문 오류나 타입 에러를 보다 쉽게 발견할 수 있다.

## 직렬화 – 프로토콜 버퍼

블록체인 시스템에는 네트워크 상에 항상 수많은 메시지가 떠돌아다니는데 노드 소프트웨어가 이러한 데이터를 일관성 있고 올바른 방법으로 복호화할 수 있어야 한다는 점이 중요하다. 구글이 개발한 프로토콜 버퍼를[14] 사용하여 서로 다른 플랫폼 사이에서 일관성 있게 메시지를 정의할 수 있고 인피니티 블록체인을 구성하는 노드를 다양한 프로그래밍 언어로 개발할 수도 있다. 직렬화 레이어는 언어의 구속을 받지 않아 크로스 플랫폼 응용 프로그램에 매우 유용하다. 프로토콜 버퍼는 전·후방 호환성을 지원하기 때문에 업데이트를 해도 하드포크보다 소프트포크만 일어날 확률이 높아진다. 또한 프로토콜 버퍼는 서드파티 소프트웨어와의 호환성을 높여주어 다른 개발자들이 하이콘 네트워크와 상호작용 할 수 있도록 한다.

## 채굴

### 개요

현존하는 대다수의 암호화폐와 같이 블록을 하나 생성하려면 작업 증명이 필요하다. 채굴자는 DAG 끝부분의 해시, 블록에 포함될 트랜잭션의 머클 루트, 현재의 난이도를 초과하는 해시값을 계산할 때까지 변하는 넌스(nonce)값에 근거하여 다음 블록의 해시를 계산하게 된다. 스펙터 창시자들은 본 프로토콜을 사용하여 초당 10 블록을 생성할 수 있다고 주장하는데 하이콘은 초기 목표를 초당 1 블록 생성으로 설정할 것이다. 현재의 프로토타입은 작업증명 기반이지만 같은 작업 증명 기반의 비트코인과 이더리움 네크워크를 유지하기 위해 막대한 양의 전기를 낭비하고 있음을 충분히 인지하고 있기에 다른 대안도 고려하고 있다. 잘 알려지지

않은 방법 중 우리가 고려하고 있는 방법은 수용능력 증명(PoC; Proof of Capacity)으로도 잘 알려진 저장공간 증명(Proof of Space)이다[32]. 이 방법은 채굴자가 많은 양의 데이터를 미리 계산하여 저장한 후 현재의 난이도를 만족하는 해답을 파일 속에서 찾도록 한다. 매우 적은 전기에너지를 사용하고 효율적인 방법이라는 사실이 버스트코인(Burst coin)과 스페이스민트(Space mint)를 통해 증명되었다.

## 세부적 채굴 과정

채굴 과정의 첫 단계는 블록 헤더 내용의 암호화 및 해시 처리로, 본 단계는 채굴 과정의 결과에 따라 변하지 않는다. 그 내용에는 이전 블록에 대한 레퍼런스, 블록에 포함될 트랜잭션의 머클 루트, 블록 목표 난이도, 블록 타임스탬프, 이 블록의 트랜잭션 이후 현재 월드 스테이트를 가리킬 머클-패트리샤 트리의 루트이다(자세한 정보는 월렛 & 계정 부분 참고).

| 블록 헤더 구성(이전 해시) |
| --- |
| **이전 블록: 32 Byte 해시의 배열** |
| **머클 루트: 32 Byte 해시** |
| **난이도: 4 Bytes** |
| **타임스탬프: 8 Bytes** |
| **스테이트 루트: 32 Byte 해시** |

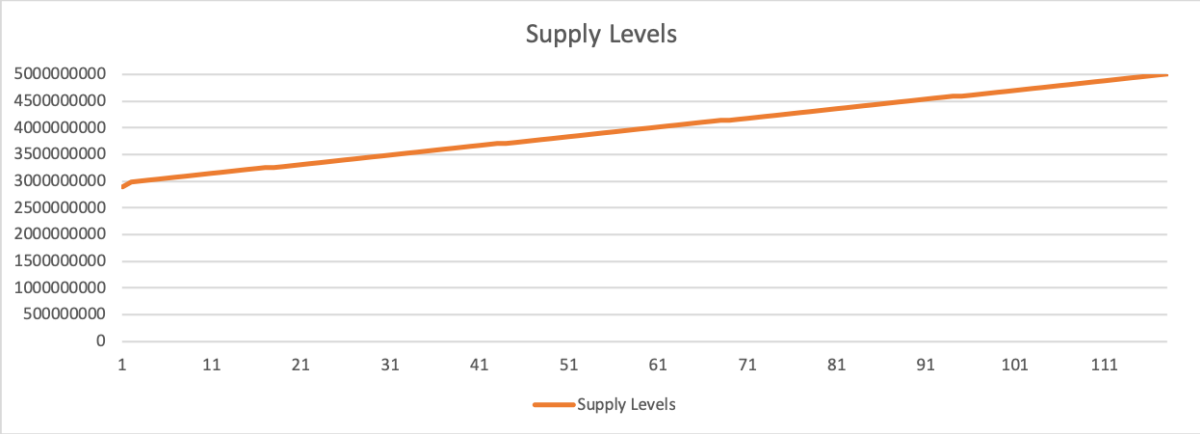
본 데이터는 GPU 또는 CPU 채굴자에게 변하지 않는 이전 블록의 해시값을 제공하기 위해 Blake 2b 64 Byte 버전으로 해시 처리된다. 특히 GPU 채굴에서 이 단계가 필요한 이유는 하이콘은 다수의 부모 블록에 연결될 수 있어 각 블록 헤더의 길이가 변할 수 있기 때문이다. GPU 채굴 소프트웨어는 데이터 구조가 정해진 길이일 때 가장 잘 작동하므로 사전 해시 작업이 필요하다. 64 Byte 헤더를 사전 해시 처리한 값은 Cryptonight 해시 알고리즘을 사용하여 해시 처리를 시도할 때마다 1 씩 증가하는 8 Byte 넌스값과 결합된다. 이 결합된 값은 함께 해시 처리되어 블록을 나타내는 32 Byte 해시값을 리턴한다. 이 해시값은 블록 헤더에 명시된 난이도와 비교되고 적절한 난이도 한계치에 도달하면 리턴된 넌스값이 블록 헤더에 포함되고 발행된다.

### 스트라텀(Stratum) 통합 & XMRig

하이콘은 스트라텀 프로토콜 및 XMRig 의 수정 버전을 사용하여 GPU 채굴을 지원한다[39].

### 채굴 보상

새로운 블록 생성에 대한 작업 증명을 성공적으로 완료하면 채굴자는 보상으로 하이콘을 받는다. 하이콘 채굴에는 상당한 시간이 소요될 예정이다. 초기 채굴 보상은 블록 당 240 하이콘이었으나, 고스프 프로토콜(GHOST Protocol) 업데이트 후 120 하이콘으로 감소하였으며 현재는 블록 당 12 하이콘으로 산정되어 있다.




## 월렛 & 계정

### 하이콘 월렛 GUI

하이콘 소프트웨어가 구동하는 풀 노드는 웹 형태의 로컬 호스트 GUI 를 사용하여 월렛 운영, 트랜잭션, 블록체인 사용을 가능하게 한다. GUI 는 가볍고 고성능인 인터페이스를 제공하기 위해 리액트(React)로 개발되었다.

### 하이콘 월렛

하이콘 월렛은 트랜잭션 서명을 위해 secp256k 와 같은 업계 표준의 타원곡선 암호 방식을 활용하고[33] 서드파티 월렛 제공 업체와의 통합을 수월하게 하기 위해 BIP 39 에 명시된 대로 월렛 리커버리를 위한 연상 기호를 적용한다[40]. BIP 32 와 BIP44 에 명시된 대로 HD(계층적 결정성) 월렛을 사용하기 위한 준비도 마쳤다. [41][42]

### 하이콘 주소

하이콘 주소는 연결된 공용키를 32 바이트 Blake2b 로 해시 처리한 결과에서 가져온 20 바이트로 생성한다. 가독성을 위해 주소의 첫 문자는 대문자 H 로 시작하고 Base58 스트링의 결과값으로 구성한다. 스트링의 마지막 4 글자는 주소의 체크섬이다. 체크섬은 3 단계로 계산된다. 먼저 주소의 32 바이트 blake2b 해시값을 계산한 후 그 결과값을 Base58 스트링으로 인코딩한다. 마지막으로 스트링의 처음 4 글자를 가지고 주소를 나타내는 스트링에 덧붙인다. 이런 방식으로 체크섬을 사용하면 잘못 입력된 주소가 사용될 가능성을 최소화할 수 있다.

### 계정 & 잔고

하이콘 참여자의 지출액과 잔고를 기록하기 위해 회계 모델을 도입해야 한다. 하이콘이 채택하기로 한 모델은 이더리움 황서(yellow paper)에 기술되어 있고[34] 이더리움이 사용하는 모델이 기반을 두고 있는 머클 패트리샤 트리(Merkle Patricia Trie) 데이터 구조이다[35]. 일련의 트랜잭션이 발행되면 모든 트랜잭션 이력을 하나의 해시값으로 저장하고 있는 월드 스테이트가 업데이트 되고 그 값은 각 블록에 저장된다. 이 월드 스테이트 값은 모든 하이콘 계정의 계정 데이터를 나타내는 머클-패트리샤 루트의 해시값이다.

계정 데이터에는 특정 하이콘 계정의 잔고, 그 특정 계정을 참조하는 가장 최근 블록에 대한 레퍼런스, 그 특정 계정에서 시작된 트랜잭션 수를 나타내는 넌스값이 저장되어 있다. 넌스값은 리플레이 공격에서 데이터를 보호하는 데 사용된다. 이전 블록에 대한 레퍼런스는 트랜잭션 이력을 더욱 빨리 검색할 수 있게 해주고 스펙터 알고리즘에서 이중지불을 쉽게 추적할 수 있도록 해주는 최적화된 기능이다.

## 동기화

하이콘은 네트워크에 처음으로 동기화할 때 헤더 우선 접근(헤더를 먼저 전송받는 방식)을 사용한다. 첫 번째 동기화 시도와 후속 시도 이후에 연결된 참가자들에게 특정 블록의 높이보다 큰 블록의 높이(현재 로컬 데이터베이스에 저장된 최대 블록 높이)와 헤더의 개수를 묻는 메시지가 발송된다. 그 헤더를 받은 블록은 검증을 거치고 만약 해당 블록이 로컬 데이터베이스에 없으면 연결된 피어는 블록의 전체 데이터를 요청한다. 전송된 블록은 다시 한 번 검증을 거치고 그 결과 블록이 유효하면 데이터베이스에 추가된다. 블록은 부모 블록이 검증을 받아 포함될 때만 데이터베이스에 추가되기 때문에 이 절차는 반드시 순차적이다.

# 결론

본 백서의 출발점이 된 기존 암호화폐의 한계를 살펴보는 작업은 인피니티 프로젝트 전체의 근간을 이루고 있다. 인피니티 프로젝트의 비전은 빠르고 안전하며 손쉽게 확장 가능한 사용자 중심의 블록체인과 암호화폐 생태계를 제공하여 더 많은 사람들이 사용할 수 있도록 하는 것이다. 인피니티 프로젝트 팀은 스펙터 프로토콜과 Blake 2b 해시 알고리즘을 결합하여 안전하고 편리한 신규 암호화폐를 제안했다. 암호화폐 하이콘과 인피니티 프로젝트는 본 백서에서 기술한 방법을 활용하여 글로벌 암호화폐 지형에 가치 있고 차별화된 암호화폐를 제공할 것이다.

# REFERENCES


[1] Blake2.net. (2017). BLAKE2. [online] Available at: https://blake2.net/ [Accessed 16 Oct. 2017].

[2] CoinDesk. (2016). Understanding The DAO Attack - CoinDesk. [online] Available at: https://www.coindesk.com/understanding-dao-hack-journalists/ [Accessed 20 Nov. 2017].

[3] Croman, K., Decker, C., Eyal, I., Gencer, A.E., Juels, A., Kosba, A., Miller, A., Saxena, P., Shi, E., Sirer, E.G. and Song, D., 2016, February. On scaling decentralized blockchains. In International Conference on Financial Cryptography and Data Security (pp. 106-125). Springer Berlin Heidelberg.

[4] Decker, C. (2017). BitcoinStats. [online] Bitcoinstats.com. Available at: http://bitcoinstats.com/network/propagation/ [Accessed 10 Nov. 2017].

[5] Decker, C. and Wattenhofer, R., 2013, September. Information propagation in the bitcoin network. In Peer-to-Peer Computing (P2P), 2013 IEEE Thirteenth International Conference on (pp. 1-10). IEEE.

[6] digiconomist.net. (2017). Bitcoin Energy Consumption. [online] Available at: https://digiconomist.net/bitcoin-energy-consumption [Accessed 16 Nov. 2017].

[7] Digiconomist. (2017). *Ethereum Energy Consumption Index (beta) - Digiconomist*. [online] Available at: https://digiconomist.net/ethereum-energy-consumption [Accessed 8 Dec. 2017].

[8] The Economist. (2007). The end of the cash era. [online] Available at: http://www.economist.com/node/8702890 [Accessed 27 Sep. 2017].

[9] Ethereum Blog. (2014). Toward a 12-second Block Time - Ethereum Blog. [online] Available at: https://blog.ethereum.org/2014/07/11/toward-a-12-second-block-time/ [Accessed 27 Sep. 2017].

[10] Etherscan.io. (2017). Ethereum Average BlockSize Chart . [online] Available at: https://etherscan.io/chart/blocksize [Accessed 16 Nov. 2017].

[11] Ethstats.net. (2017). Ethereum Network Status. [online] Available at: https://ethstats.net/ [Accessed 16 Nov. 2017].

[12] Goland.org. (2017). How to make block chains strongly consistent – Stuff Yaron Finds Interesting. [online] Available at: http://www.goland.org/why_block_chains_are_strongly_consistent/ [Accessed 27 Sep. 2017].

[13] Goland.org. (2017). The block chain and the CAP Theorem – Stuff Yaron Finds Interesting. [online] Available at: http://www.goland.org/blockchain_and_cap/ [Accessed 27 Sep. 2017].

[14] Google Developers. (2017). Protocol Buffers | Google Developers. [online] Available at: https://developers.google.com/protocol-buffers/ [Accessed 20 Oct. 2017].

[15] James-Lubin, K. (2015). Blockchain scalability. [online] O'Reilly Media. Available at: https://www.oreilly.com/ideas/blockchain-scalability [Accessed 16 Nov. 2017].

[16] Koteska, B., Karafilovski, E. and Mishev, A. (2017), Blockchain Implementation Quality Challenges: A Literature Review : Proceedings of the SQAMIA 2017: 6th Workshop of Software Quality, Analysis, Monitoring, Improvement, and Applications, Belgrade, Serbia, 11-13.9.2017.

[17] Malanov.A, (2017). Six main disadvantages of Bitcoin and the blockchain. [online] Kaspersky.com. Available at: https://www.kaspersky.com/blog/bitcoin-blockchain-issues/18019/ [Accessed 16 Nov. 2017].

[18] Motherboard. (2017). One Bitcoin Transaction Now Uses as Much Energy as Your House in a Week. [online] Available at: https://motherboard.vice.com/en_us/article/ywbbpm/bitcoin-mining-electricity-consumption-ethereum-energy-climate-change [Accessed 20 Nov. 2017].

[19] Nakamoto, S., 2008. Bitcoin: A peer-to-peer electronic cash system.

[20] The NodeSource Blog - Node.js Tutorials, Guides, and Updates. (2014). Why Asynchronous?. [online] Available at: http://nodesource.com/blog/why-asynchronous/ [Accessed 16 Nov. 2017].

[21] Park, J.H. and Park, J.H., (2017). Blockchain Security in Cloud Computing: Use Cases, Challenges, and Solutions. Symmetry, 9(8), p.164.

[22] Poon, J. and Dryja, T.. (2016). The Bitcoin Lightning.network [online] Available at: https://lightning.network/lightning-network-paper.pdf.

[23] Raiden-network.readthedocs.io. (2017). Raiden Specification — Raiden Network 0.2.0 documentation. [online] Available at: https://raiden-network.readthedocs.io/en/stable/spec.html [Accessed 7 Dec. 2017].

[24] Reitwiessner, C. (2017). zkSnarks in a Nutshell [online] Available at: http://chriseth.github.io/notes/articles/zksnarks/zksnarks.pdf [Accessed 23 Nov. 2017].

[25] Sirer, E.G. and Song, D., 2016, February. On scaling decentralized blockchains. In International Conference on Financial Cryptography and Data Security (pp. 106-125). Springer Berlin Heidelberg.

[26] Sompolinsky, Y., Lewenberg, Y. and Zohar, A., 2016. SPECTRE: A Fast and Scalable Cryptocurrency Protocol. IACR Cryptology ePrint Archive, 2016, p.1159.

[27] Sompolinsky, Y. and Zohar, A., 2015, January. Secure high-rate transaction processing in bitcoin. In International Conference on Financial Cryptography and Data Security (pp. 507-527). Springer, Berlin, Heidelberg.

[28] Son, M. (2017). Bitcoin's Rise Happened in Shadows of Finance. Now Banks Want In. [online] Bloomberg.com. Available at: https://www.bloomberg.com/news/articles/2017-10-05/bitcoin-s-rise-happened-in-shadows-of-finance-now-banks-want-in [Accessed 7 Dec. 2017].

[29] Swan, M., 2015. Blockchain: Blueprint for a new economy. " O'Reilly Media, Inc.".

[30] VISA (2017). Visa Inc. Facts & Figures . [online] Available at: https://usa.visa.com/dam/VCOM/global/about-visa/documents/visa-facts-figures-jan-2017.pdf [Accessed 20 Nov. 2017].

[31] Yli-Huumo, J., Ko, D., Choi, S., Park, S. and Smolander, K. (2016). Where Is Current Research on Blockchain Technology?—A Systematic Review. PLOS ONE, 11(10), p.e0163477.

[32] Dziembowski, Stefan; Faust, Sebastian; Kolmogorov, Vladimir; Pietrzak, Krzysztof (2015). "Proofs of Space". 9216: 585–605.
Available at: https://eprint.iacr.org/2013/796.pdf

[33] Secg.org. (2010). Standards For Efficient Cryptography 2, [online] Available at: http://www.secg.org/sec2-v2.pdf [Accessed 20 Jan. 2018].

[34] Wood, G., 2014. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, *151*.

[35] Ethereum. (2018). *ethereum/wiki*. [online] Available at: https://github.com/ethereum/wiki/wiki/Patricia-Tree [Accessed 22 Jan. 2018].

[36] Team Hycon. (2018). *Team-Hycon/Genesis-View*. [online] Available at: https://github.com/Team-Hycon/Genesis-View [Accessed 22 Jan. 2018].

[37] Cryptonight Hash Function Cryptonote.org. (2018). [online] Available at: https://cryptonote.org/cns/cns008.txt [Accessed 2 Feb. 2018].

[38] JustCryptoNews. (2018). *Sia Coin - BitMain Antminer A3 Blake (2b) ASIC Miner Announced*. [online] Available at: https://www.justcryptonews.com/340/sia-coin-bitmain-antminer-a3-blake-2b-asic-miner-announced [Accessed 2 Feb. 2018].

[39] Monero (XMR) CPU Miner - GitHub. (2018). *xmrig/xmrig*. [online] Available at: https://github.com/xmrig/xmrig [Accessed 6 Feb. 2018].

[40] Palatinus, M., Rusnak, P., Voisine, A., Bowe, S.. *bitcoin/bips/bip-0039*. [online] Available at: https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki [Accessed 6 Feb. 2018].

[41] BIP32. (2012). bitcoin/bips. [online] Available at: https://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki.

[42] BIP44 (2014). bitcoin/bips. [online] Available at: https://github.com/bitcoin/bips/blob/master/bip-0044.mediawiki.

[43] The Monero Project. (2018). Monero: A Scheduled Network Upgrade is Planned for April 6. [online] Available at: https://getmonero.org/2018/03/28/a-scheduled-protocol-upgrade-is-planned-for-April-6-2018-03-28.html.

# 부록

# 부록 A – 제네시스 블록

## 논의

하이콘의 제네시스 블록은 비트코인 제네시스 블록 발행 9 주년에 맞춰 2018 년 1 월 4 일 오전 3:15(GMT+9)에 생성되었다. 제네시스 블록에 저장된 정보의 요약본은 아래와 같다. 팀 하이콘 깃허브 저장소에서 다수의 프로그래밍 언어로 개발된 복호화 소프트웨어와 제네시스 블록을 확인할 수 있다.

하이콘의 제네시스 블록에는 헤더 1 개와 트랜잭션 6 개가 포함되어 있고 트랜잭션 6 개는 초기에 할당될 하이콘의 생성을 나타낸다. 하이콘과 예산 할당에 대한 자세한 사항은 부록 B 를 참고하면 된다.

블록 헤더에는 채굴 난이도, 트랜잭션의 머클 트리 루트, 월드 스테이트의 루트, 블록의 타임스탬프 정보가 들어가 있다.

블록에 저장된 나머지 데이터는 향후 토큰 배포에 사용될 계정에 로딩되어 있는 트랜잭션으로 구성된다. 제네시스 블록에 담긴 트랜잭션을 이런 방식으로 암호화하는 것이 하이콘 ICO(암호화폐 공개)를 준비하는 과정에서 가장 투명한 방법이라고 판단했다. 이렇게 되면 해당 계정의 모든 자금을 처음부터 추적할 수 있기 때문이다.

# 하이콘 제네시스 블록의 내용

복호화된 하이콘 제네시스 블록의 내용은 아래와 같다.

**블록 헤더**

난이도: 0 Merkle Root:
cff5f8a5381ce41e26bf3f5f7b658dcef0d4935dfd791460614feb894ff36457

스테이트 루트:
e08408cb5bf38fb2652676af953d169c7997dd2af88299163b9a389b9d6a3ed4

타임스탬프: Thu Jan 04 03:15:05 KST 2018

**트랜잭션**

ICO 계정
Account Address(raw): 9565e92e694ef206abe21d65d3a93996682d41f7
Amount: 2,000,000,000 HYCON

에어드롭 계정
Account address(raw): fa7042154efb88d06c198ef106ca31aed57e6875
Amount: 400,000,000 HYCON

Team 계정
Account address(raw): 8bab45e2f5c79c00d539ae1a65dbd1f8fd416ca7
Amount: 500,000,000 HYCON

개발 비용
Account address(raw): 7a7b31e5aced4889a75d1042a6f1204d2a889af8
Amount: 500,000,000 HYCON

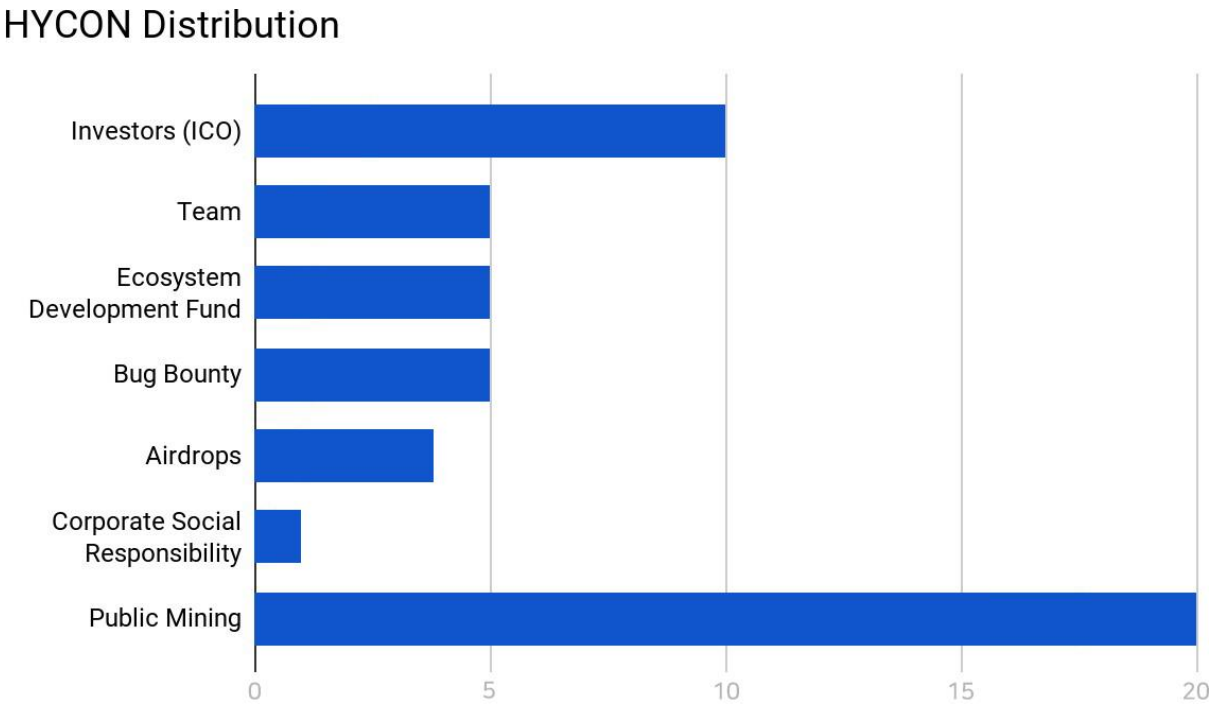
버그 현상금
Account address(raw): 571a6e4554afbb09ee7da1ae20c18dbca9fabc46
Amount: 500,000,000 HYCON

기업의 사회적 책임(CSR)
Account address(raw): 11d8046e6cd88f9e580b84a0b10c7c452f0030fc
Amount: 100,000,000 HYCON

# 부록 B – 코인 분배율 & 예산 할당

## 코인 분배율

발행될 총 하이콘 수량은 50 억 개로 코인 할당 방법은 아래와 같다.




위 그래프에서 볼 수 있듯이 2,000,000,000 HYC 하이콘이 공개적으로 채굴된다. 나머지 3,000,000,000 HYC 는 제네시스 블록과 함께 생성되었고 네트워크 공개 시 코인 배포를 담당할 하이콘 계정에 할당한 상태다.

이 3,000,000,000 HYC 는 6 개 부분으로 나누어진다. 그 중 가장 많은 부분인 1,000,000,000 HYC 는 ICO 참여자 및 사전 투자자들에게 할당한다. 500,000,000 HYC 는 각각 하이콘 팀원들, 기반 생태계 개발 기금, 버그 현상금 지급에 할당한다. 100,000,000 HYC 는 기업의 사회적 책임 부문에 들어가고 나머지 400,000,000 HYC 는 이벤트 또는 향후 결정될 방법을 통해 배포한다.

# 예산 할당

하이콘 ICO 예산 할당의 최우선 사항은 재능 있는 개발팀을 구축하고 장기적으로 프로젝트의 미래 전망을 세우는 것이다. 따라서 펀드레이징 금액의 70%는 하이콘과 인피니티 프로젝트(인피니티 플랫폼과 인피니티 분산 거래소 등)의 미래 연구·개발에 투자된다. 하이콘과 인피니티 프로젝트는 별개로 진행되지만 전체 프로젝트 진행의 유동성을 보장할 펀드레이징 행사는 하이콘 ICO 가 유일하다는 점이 중요하다.