Mine Cryptocurrencies Sooner, Faster, and Cheaper with Achronix Speedcore Embedded FPGAs (WP014)

January 14, 2019 **White Paper**

Abstract

New cryptocurrencies such as Monero introduce ASIC-resistance and memory-hardness to prevent ASICs from being built that give some operators a competitive mining advantage over others who do not have access to the same technology. This white paper discusses the relevant background and presents a solution based on Achronix Speedcore™ embedded FPGAs (eFPGAs), enabling users to regain a highly profitable advantage over competing solutions.

Introduction

Cryptocurrency mining is the process of computing a new cryptocurrency unit based on all the previously found ones. The concept of cryptocurrency is nearly universally recognized by the publicity of the original cryptocurrency, Bitcoin. Cryptocurrencies were supposed to be a broadly democratic currency vehicle not controlled by any one entity, such as banks, governments, or small groups of companies. Much of a cryptocurrency's acceptance and trustworthiness is based on that proposition. However, with Bitcoin, that is not how it unfolded. Instead, Bitcoin quickly became virtually monopolized by a small number of entities located in an even smaller number of geographies. This outcome is the result of particular properties of how Bitcoin mining works:

- Rewards are given to early adopters who build the fastest mining hardware in ASICs on the latest technology nodes, leaving all other contenders at a huge disadvantage.
- Bitcoin is designed to be computationally intensive, requiring huge amounts of power. Entities with access to the lowest cost power have a huge advantage.

The extent to which these unforeseen developments affect the validity of Bitcoin as a truly democratized, trusted cryptocurrency has been a topic of debate among experts. In response, developers have devised several new cryptocurrencies, which, by design, are mathematically proven to be impervious to the techniques that allowed Bitcoin to become largely monopolized.

This white paper explores the underlying fundamentals of, and the best solutions to implement mining for these new forms of cryptocurrency. While it does not go into mathematical details of blockchain technologies, a high-level understanding of the economics of cryptocurrency mining is reviewed.

The Value of a Currency

For something to be considered a currency, society demands that it be a fungible (interchangeable), durable, portable, recognizable, and a store of value. Cryptocurrencies are perfectly durable, provably recognizable (mathematically), and perfectly portable. Fungibility is currently provided primarily through currency exchanges. Store of value is achieved by limiting the supply. Minting the cryptocurrency through protocol-dictated (low) inflation rates creates built-in scarcity.

Scarcity drives value, whether it is diamonds, gold, the US dollar, or cryptocurrency. A guaranteed scarcity is a required safeguard against undue inflation or deflation of the value, which makes cryptocurrency suitable for use as a currency.

Bitcoin's scarcity is mathematically guaranteed by its protocol that only allows a fixed number of Bitcoins to be awarded per block mined. The amount of energy required to mine a Bitcoin depends on a difficulty-adjusted algorithm. As more miners compete for the block reward, the difficulty increases, making the Bitcoin network consume more energy overall. This energy is paid for by the miners and is the primary expense of the Bitcoin network.

The Bitcoin mining process involves execution of a complex, specified algorithm that searches for a specific number in a very large number space. Executing this algorithm consumes a certain amount of electrical energy per search, hence the efficiency of a Bitcoin algorithm is usually measured in dollars per gigahash per second ($/Gh/sec), where the cost is largely driven by power consumption.

Specifically, the search is for a valid number that mathematically seals a block with users' signed transactions. The miners award themselves some (fractional) Bitcoins that did not previously exist as a reward for performing this service for users. By finding a valid number, the miner has proven he has done the work to search for it. All mining-based cryptocurrency algorithms operate on similar "proof of work" principles as the key mechanism to add blocks to the chain. Since the valid solution is extremely hard to find, sustained contention from other miners is statistically impossible.

The first-generation execution engines used to run the Bitcoin mining algorithm were CPUs and then later (GP) GPUs. These processors are software-programmable, general-purpose machines which are not particularly efficient at executing the Bitcoin mining algorithm. Moreover, they consume a relatively large amount of energy while executing the algorithm. As Bitcoin's popularity grew and the competition to find new Bitcoins increased, operators of Bitcoin miners started to convert these software programs into hardware implementations to speed their search, and to reduce the amount of electrical power required.

For this reason, operators of Bitcoin cryptocurrency miners started using FPGAs instead of software running on CPUs and GPUs. FPGAs are semiconductor devices that can be programmed by end users to implement digital functions, which, for example, can implement computations directly in hardware rather than by executing the instructions of a program. Computations implemented in hardware are much more efficient than software running on a CPU for the equivalent calculation. In addition, a hardware implementation enables much higher degrees of parallelism, further increasing computational performance. Therefore, FPGAs can execute the Bitcoin algorithm much more quickly and more efficiently than CPUs and GPUs.

However, because the Bitcoin algorithm is unchanging, the most efficient implementation is to build it as hard-wired, fixed-function logic circuit. This fact has lead to the migration of FPGA-based implementations to ASIC-based ones as reprogrammability is not needed for the Bitcoin algorithm.

Figure 1: ***Cryptocurrency Mining ASICs are Small Enough to Fit in a Thumb Drive***

To further increase computational density, miners began packing as many individual ASICs as possible, often a hundred or more, into small form-factor mining rigs.

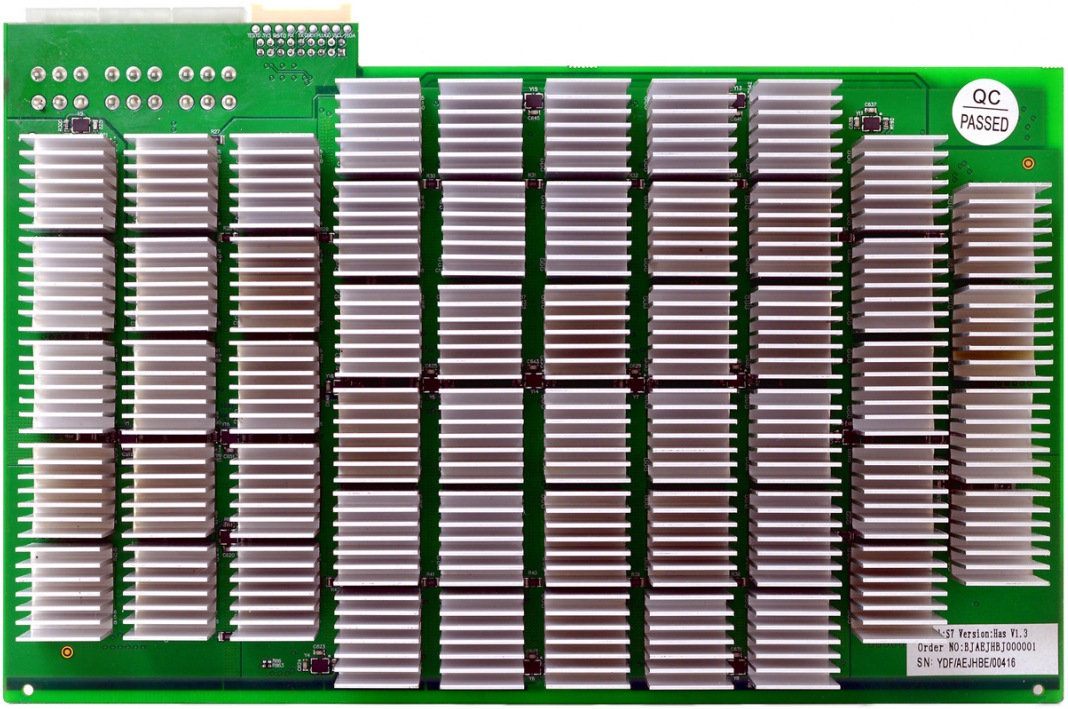

Figure 2: ***PCB from a Bitcoin Mining Rig with 63 ASICs. A Rig May Contain Multiple Such Boards***

Figure 3: ***Bitcoin Mining Rig Enclosure. Most Popular Mining Rigs Have a Similar Form Factor***

Hundreds of thousands of these rigs operate in dedicated data centers around the globe near the world's lowest cost electrical power. Those locations currently include places such as Inner Mongolia in China, where generators fueled by mountains of inexpensive coal produce large amounts of relatively low cost electricity. Other places include Iceland with abundant geothermal energy sources, and Venezuela thanks to energy subsidies.

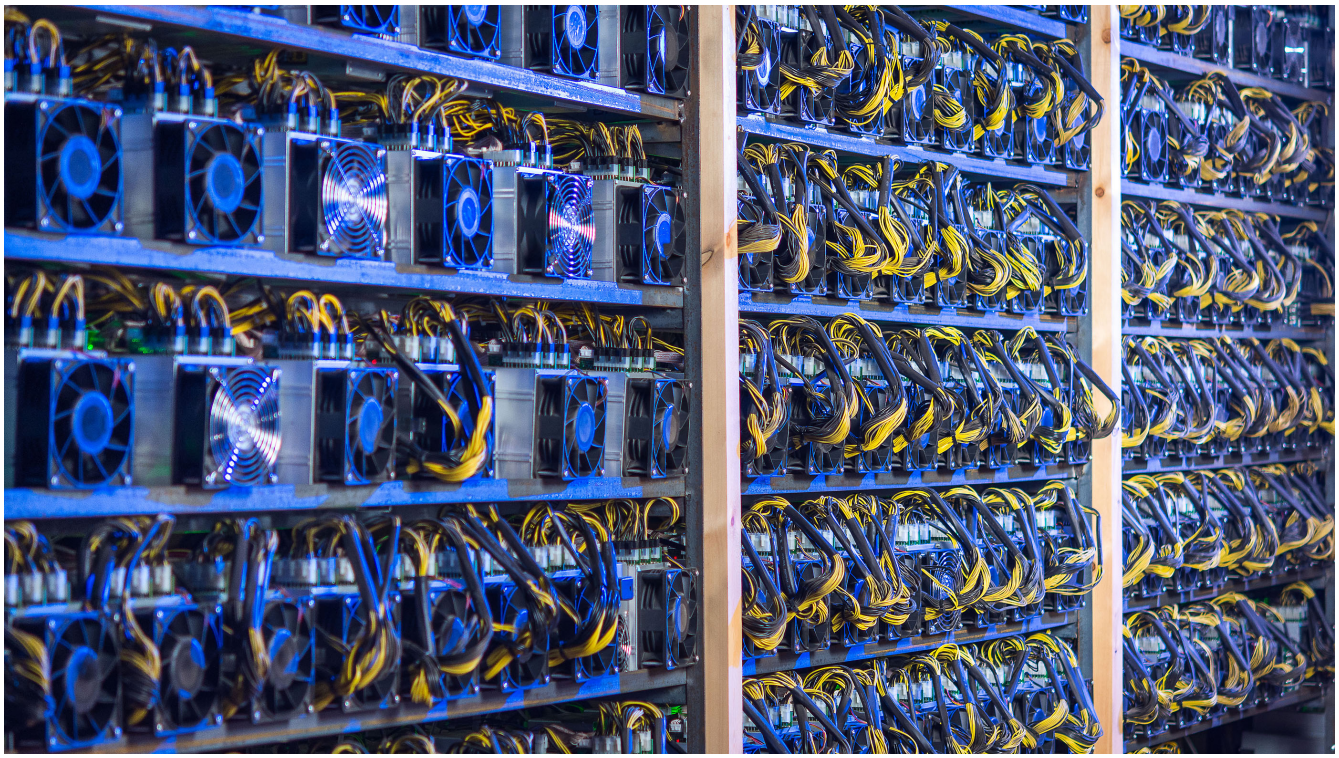

Figure 4: ***A Single Rack Consists of Dozens of Air Cooled Mining Rigs***

The Cryptocurrency Community is Developing Alternatives to Bitcoin

Bitcoin has lost much of its allure due to the concentration of control of the world's Bitcoin mining resources by a few players in a few locations. In response, the larger, global cryptocurrency community has started to develop alternative cryptocurrencies using lessons learned from the Bitcoin experience.

New post-Bitcoin cryptocurrency algorithms are being created to ensure that no single entity can dominate the currency. The most important such technique is called "ASIC resistance", making it impractical to build an ASIC implementation. The prevailing method to make a cryptocurrency ASIC resistant is to create a framework in which a mining algorithm can be changed to a new algorithm whenever necessary, say every six months. Such a change is called a blockchain fork. Upon each fork, all existing hardware platforms executing the old algorithm immediately become completely and permanently worthless unless they can be reprogrammed to execute the new algorithm.

Examples of recent forks are Siacoin, Monero, or Bitcoin Gold forking their blockchain on purpose to invalidate the ASIC miners that were released. There are also cryptocurrencies that choose algorithms that are purposely designed to defeat attempts to add sufficient flexibility to ASICs, or others that require proof of memory or storage to ensure ASICs cannot accelerate the hash rate.

The new algorithm is not known or disclosed upfront, nor can it be anticipated beforehand whether it might be executable by some semi-flexible compute architecture that could be implemented with an ASIC. Because it takes at least six months, usually more, to develop and fabricate new fixed-algorithm ASICs designed to execute the new algorithms, such ASICs will be useless as soon as they are deployed. With CPUs and GPUs, the new algorithm can be implemented and deployed quickly, thus ASIC resistance creates a more level playing field for all players.

As an example, the figure below shows the Global Monero hash rate over time, showing when the fork happened from Cryptonight V0 to V7 in April 2018. Upon this fork, all non-programmable mining hardware became worthless. The next fork happened six months later as scheduled.

Figure 5: ***Global Monero Hash Rate Plotted Over Time. The fork from Cryptonight V0 to V7 in April 2018 caused a sharp drop because all non-programmable mining hardware became worthless. The next fork happened six months later as scheduled. Source:*** *bitinfocharts.com*

ASIC-Resistant Cryptocurrencies and FPGAs

With mining ASICs invalidated, FPGA technology has again become the preferred solution. Like an ASIC, an FPGA is also an efficient miner compared to CPUs and GPUs. FPGAs are much more flexible than ASICs and the designs running on FPGAs can quickly be adjusted to accommodate changes in cryptocurrency mining algorithms. When a new mining algorithm is announced for a specific cryptocurrency, a data center of FPGA-based mining rigs can be reconfigured literally overnight to use the new algorithm at the same speed and with similar electrical efficiency.

The compelling advantage of FPGAs is a much faster time to market, compared with ASICs, for the initial implementation, while still allowing for efficiency gains later by means of rolling out optimized FPGA bitstream updates to mining rigs. This gives users of FPGA-based miners a major profit advantage over non-programmable implementations.

And, just as importantly, the programmable hardware allows users to switch, at any time, to mine the currently most profitable coin. Coin hopping algorithms are very common in the mining space. Exploiting the flexibility of reprogrammable mining hardware this way can significantly add to mining profits.

These are all powerful arguments in favor of using programmable hardware in the form of FPGA technology as implementation vehicles for cryptocurrency mining algorithms.

Memory-Hard Cryptocurrencies and FPGAs

While ASIC resistance is a method to counteract monopolization of cryptocurrencies by the few entities that can afford to build ASIC solutions, it does not ensure the necessary scarcity of a currency. The developers of cryptocurrency algorithms have devised an additional method to fundamentally ensure scarcity called "memory hardness", which can be combined with ASIC resistance. The most prominent and promising examples of these new cryptocurrencies are Monero/XMR and Ethereum/ETH. These new cryptocurrency algorithms are both ASIC resistant and memory hard.

One way to make a mining algorithm ASIC resistant is to require more hardware than can fit in an ASIC, which is the case with algorithms requiring several gigabytes of memory, e.g., the Equihash algorithm used for Ethereum.

Memory hardness is incorporated into cryptocurrency mining algorithms to prevent the use of shortcuts in executing the algorithm. A memory-hard cryptocurrency mining algorithm requires that the mining process must read a value stored in a memory location, use that value according to the steps specified by the current algorithm for that specific cryptocurrency, and then use the result as the address for the next memory transaction. The value of the memory addresses, therefore, cannot be determined *a priori*. Different memory-hard algorithms ensure this feature in somewhat different ways.

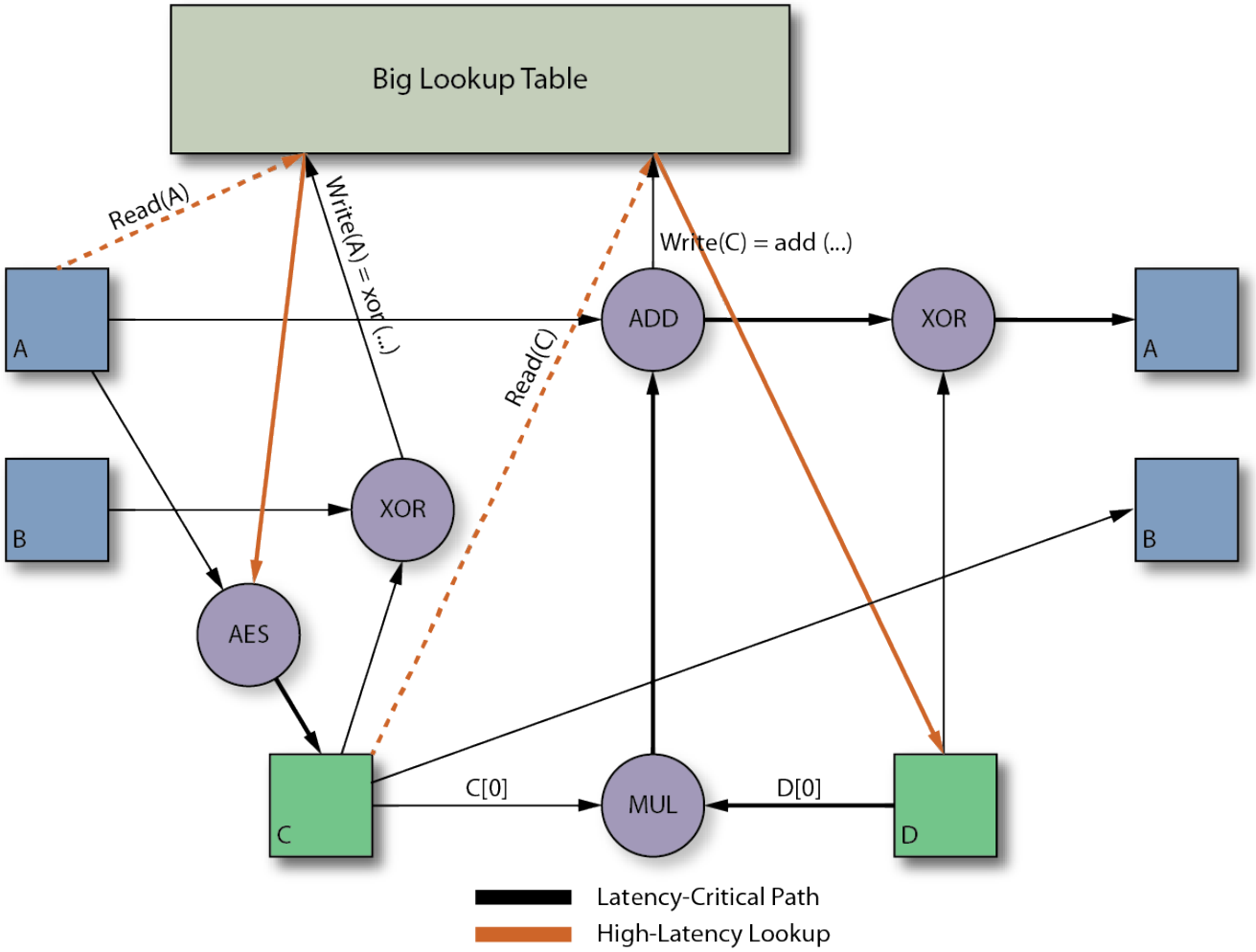

Figure 6: ***Latency-critical path of the main loop of the Cryptonight hash algorithm showing memory access dependencies in the main loop of the Cryptonight hash algorithm. The main loop is iterated many times. Each access depends on the previous access. In this case the lookup table size is 2 MB. The computation functions are relatively simple and do not require much dynamic power .***

Memory-hard algorithms create high difficulty proof-of-work routines by leveraging the fundamental lower bound on the speed of memory transactions that are rooted in the laws of physics. Therefore, their execution time cannot be shortened below a certain minimum. The algorithm also guarantees that all transactions must take place sequentially, so no implementation can accelerate this process by means of parallelization.

The number of memory transactions required may be a million or more, and each of the ensuing intermediate results, and therefore the final result, cannot be pre-calculated.

Memory-hard cryptocurrency algorithms require millions of sequential memory transactions to execute the complete algorithm, but execution of the algorithm is not computationally intensive. Therefore, the performance of the miner almost entirely depends on memory bandwidth and transaction latency, i.e., the shortest memory cycle time wins. Memory hardness, therefore, drives the architecture of a superior cryptocurrency miner.

How the ideal hardware platform is architected depends on the algorithm, since execution performance of different algorithms depends in different ways on memory size and access patterns. The size of the memory array to be traversed by the algorithm dictates the architecture of the platform. For example, the Ethereum cryptocurrency algorithms require memory sizes of several gigabytes, whereas Monero requires memories that are three orders of magnitude smaller. These differences result in very different memory architectures.

Algorithms with smaller memory size requirements can, in theory, be implemented using external memory, but are not as efficient for a variety of reasons, primarily because bandwidth and latency of on-die memory are better by orders of magnitude than off-chip memory. Power consumption and density on the PCB are other key factors adding to both initial and operational cost of implementations attempting to use off-die memories. Recall that mining farms deploy many thousands of mining machines, hence achieving the best power efficiency and highest compute performance per unit of volume are critical.

CPUs and GPUs are a Poor Fit

These cryptocurrency algorithms are designed to use memory spaces that are too large to fit into the second-level caches on most of today's microprocessors, which forces these processors to go off chip for these memory accesses. External memory devices such as DDR4 memories are simply too slow, too expensive, too power-hungry and take up way too much space for competitive execution of these algorithms. For example, SDRAMs are block-oriented memories, and therefore, are very inefficient for the fine-grained transactions performed by cryptocurrency algorithms.

Thus PC/server CPUs and GPUs are poor implementation vehicles for cryptocurrency mining rigs that need to execute memory-hard algorithms compared to architectures using on-die memories. Furthermore, GPUs are poorly equipped to make sparse traverses through a large memory space, which is yet another reason that they are inefficient engines for memory-hard cryptocurrency mining.

General-Purpose FPGAs are not the Answer Either

Large, high-end FPGAs do have a considerable amount of on-die memory in the form of many medium and large embedded memory blocks. This fact would seem to make stand-alone FPGAs the implementation of choice for memory-hard cryptocurrency algorithms. However, off-the-shelf FPGAs are designed for general-purpose applications and are not at all designed for cryptocurrency mining. The amount of on-die memory in general-purpose FPGAs remains the main factor severely limiting the number of mining processes that can be running in parallel on these devices. FPGA memory blocks are evenly, relatively sparsely distributed across an FPGA core, but these sparse embedded memory blocks are individually too small to be used for the memory-hard cryptocurrency mining spaces. As a result, these smaller, embedded FPGA memories must be combined into larger memories, drastically slowing overall performance and disqualifying general-purpose FPGAs for these algorithms.

Furthermore, general-purpose FPGAs have literally become "programmable piles of parts," a term recently coined by EE Journal's Bryon Moyer in his article "Programmable Pile of Parts: Big FPGAs Go Off the Rails" to describe the latest, large FPGAs from major FPGA vendors. Cryptocurrency mining algorithms do not require many of the on-chip functions, including almost all of the available hardened IP (PCIe controllers, Ethernet MACs, SerDes ports, etc.). All of those unneeded component parts become "dark silicon" in cryptocurrency mining applications, meaning that a lot of expensive silicon in these FPGAs will be wasted, while still adding to static leakage power and cost. On the PCB, any off-the-shelf FPGA large enough to accommodate meaningful mining capacity will take up a lot of board space, and requires many accompanying components varying from complex power regulators to decoupling capacitors. This large footprint is unacceptable in the context of cryptocurrency mining where the highest density is an absolute must, given the vast scale of mining farms.

Energy efficiency is as important as performance. Good efficiency is required to achieve both a quicker path to ROI on each mining rig and a long tail on the life of the miner being profitable. General-purpose FPGAs are not optimized to be energy efficient when executing cryptocurrency mining workloads.

Finally, hardware implementations of cryptocurrency algorithms are, due to their structure, inherently difficult to route on general-purpose FPGAs. This routing difficulty is due not only to the overhead from the combining of memory blocks into large memories, but also to the random logic implementing the hashing components of the algorithms.

The Best Implementation Technology for Cryptocurrency

The best implementation technology for these new cryptocurrency miners are ASICs with embedded FPGAs (eFPGAs). This solution allows the mining rig developer to put exactly the right amount of required resources on chip to implement the target algorithm(s) while enabling quick reconfiguration in response to algorithm changes.

Combining ASIC and eFPGA technology creates the perfect semiconductor vehicle for realizing new cryptocurrency mining architectures. The eFPGA permits hardware reconfiguration of the cryptocurrency mining engine whenever there are changes in the underlying algorithm, which means that the chip need not be re-spun to accommodate hash algorithm changes. It merely needs to be reprogrammed, for example, by downloading a new bitstream image file remotely into flash memory.

Embedded FPGAs also enable tight integration with multiple, correctly-sized, on-die memories immediately adjacent to the algorithm execution hardware. This use of on-die memories, compared to external memories (as is the case with GPUs and CPUs) gives this solution a tremendous power, performance and area (PPA) advantage. With modern semiconductor technology nodes, it is now possible to integrate several megabytes of on-die SRAM memory. In addition, the fact that this is an ASIC enables the most compact memory configuration to be selected, which is, for example, much more PPA-efficient than what can be achieved with standard FPGAs.

The figure below shows an example architecture of a cryptocurrency mining ASIC that combines an Achronix Speedcore eFPGA with six on-die memories located around the periphery of the eFPGA. The ASIC only has a few slow I/O pins and does not need any additional chips to execute the mining workloads.

This particular design incorporates sufficient resources for six cryptocurrency mining engines that can all operate in parallel. For efficiency, speed, and profit, it is imperative to maximize the number of on-chip mining engines that can operate in parallel.

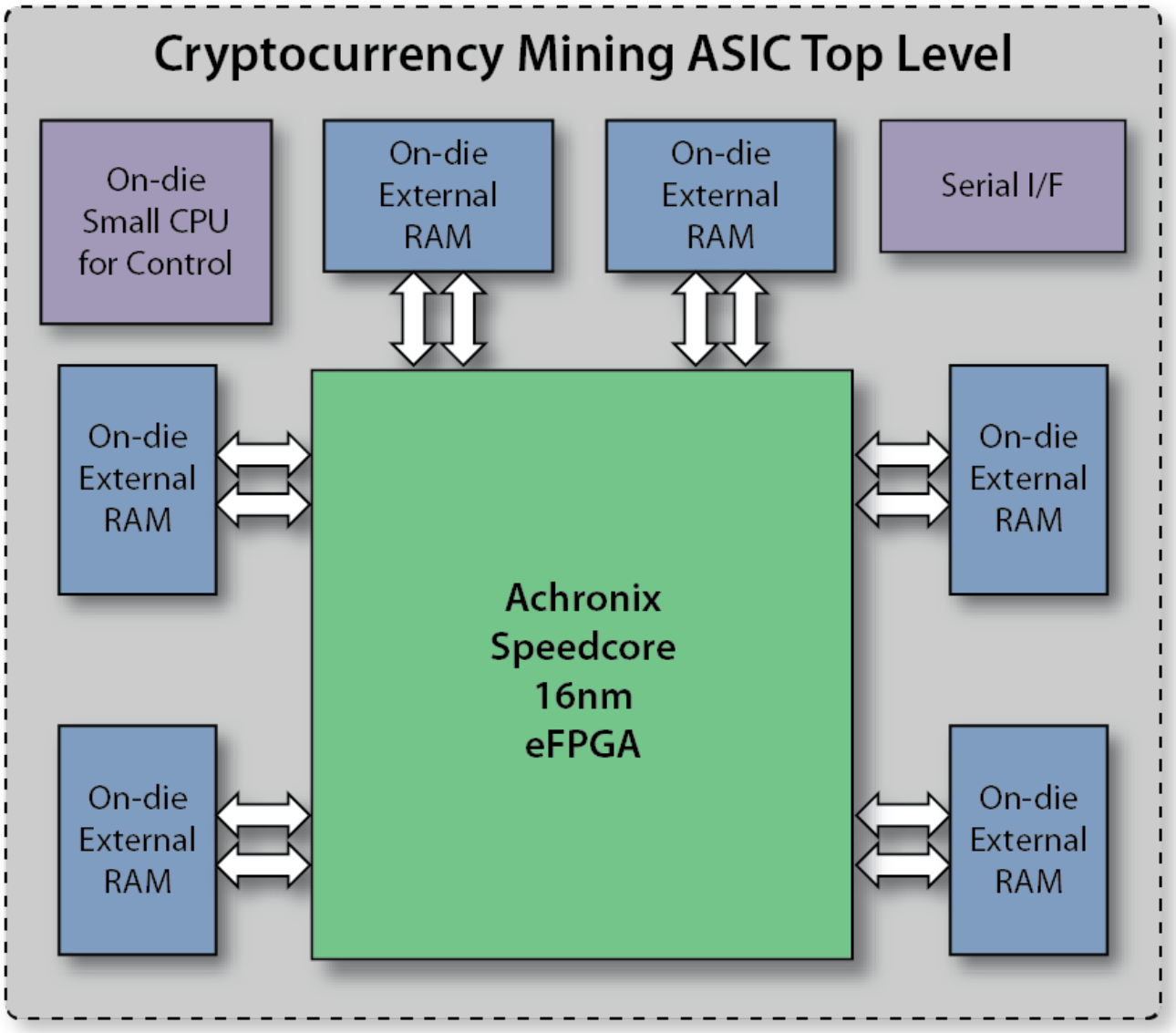

Figure 7: ***A Cryptocurrency mining ASIC with an eFPGA and large on-die memories for best performance, power efficiency and mining rig density (not to scale). The ASIC has only a few slow I/O pins and does not need any additional devices to execute the mining workloads***

Achronix's Speedcore eFPGA arrays are customizable, which means they are sized to be the optimal implementation for an application domain, while still offering a very high degree of flexibility to accommodate significant changes to the hashing algorithms. It is important that the programmable resources in the Speedcore fabric instantiated into the ASIC be tailored to the target cryptocurrency algorithm(s) to minimize fabrication costs. A typical Speedcore eFPGA for this application will contain some number of lookup-tables (LUTs) to supply the programmable logic needed to execute the algorithm (replicated six times in this example).

Connecting the eFPGA with the adjacent six memory arrays is straightforward. Moreover, a relatively simple means to connect to the outside world is provided by means of a serial interface. The Speedcore eFPGA array may also be designed to contain some number of DSP blocks and smaller memory blocks within the eFPGA array, as several operations in the hashing algorithms include wide integer arithmetic. The location of the columns of the memories and DSP blocks inside the Speedcore fabric is also optimized for best performance with this class of algorithms.

Given the inherently difficult-to-route characteristic of the hashing algorithms used in this context, the eFPGA implementation must maximally exploit all available routing layers in the ASIC's metal stack to ensure routing closure when compiling a mining algorithm into the FPGA's bitstream. FPGA routing architectures that have not been optimized for these algorithms, or that utilize only a limited number of routing layers, will exhibit great difficulty dealing with the inherent routing congestion.

Finally, it is possible to add custom hardware blocks to the Speedcore array. That means that if there is a fixed-function block used often in the target algorithm(s), it may be beneficial to harden that block and then to embed several such custom blocks inside of the Speedcore eFPGA array. This is an important benefit of Achronix's Speedcore eFPGA technology. For example, it may be advantageous to create a custom RTL block based on a SHA-256 hashing algorithm if it is required by the overall cryptocurrency mining algorithm. ROM tables used in the AES encryption algorithm, which have not changed in decades, are another excellent candidate for custom blocks. Any block that is unlikely to change is a good candidate for a custom block in the Speedcore eFPGA.

In short, you can use the Speedcore eFPGA array's easy customizability to design precisely the FPGA needed for the target class(es) of cryptocurrency algorithm(s) with no dark silicon. This is an extremely important economic driver when contemplating the fabrication and deployment of hundreds of thousands or millions of cryptocurrency mining engines.

Monero Miner Case Study

This section summarizes the results of an internal study on how to implement a complete hardware mining solution for Monero based on the architecture described in the previous section. Monero cryptocurrency is in the class of ASIC-resistant, memory-hard algorithms. Monero has been, and is expected to continue to be, one of the most popular and enduring cryptocurrencies in its class. The goal of the proposed implementation is to achieve the highest mining profit for this class of cryptocurrencies

Depending on the number of on-die memories and the selected size of the Speedcore instance, different trade-offs between die size and performance can be made to determine the most profitable sweet spot. Monero hash rates ranging from 130 to over 2,000 hashes per second per device can be achieved. In this study, a configuration exceeding 1,000 Monero hashes per second was found to be the optimal point between size and mining performance. This study also included the initialization and finalization logic that precedes and follows the memory-hard main iteration loop in the algorithm. Like the main iteration loop, this logic may also be changed due to a fork, so it needs to be implemented with programmable logic as well.

Estimated average power consumption is well below 10 Watt per 1000 hashes per second — a fraction of what other solutions can achieve.

The figure below shows the logic placement in the ACE GUI demonstrating how major logic hierarchies were placed in the Speedcore instance. The algorithm's memory initialization and finalization logic (shown in yellow, green and orange) is also implemented with programmable logic. In terms of resource usage it dominates the main loops' hashing logic (shown in blue). This implementation outperforms dedicated, non-programmable 1000 hashes/sec miner ASICs. This figure does not show the on-die scratchpad memories.

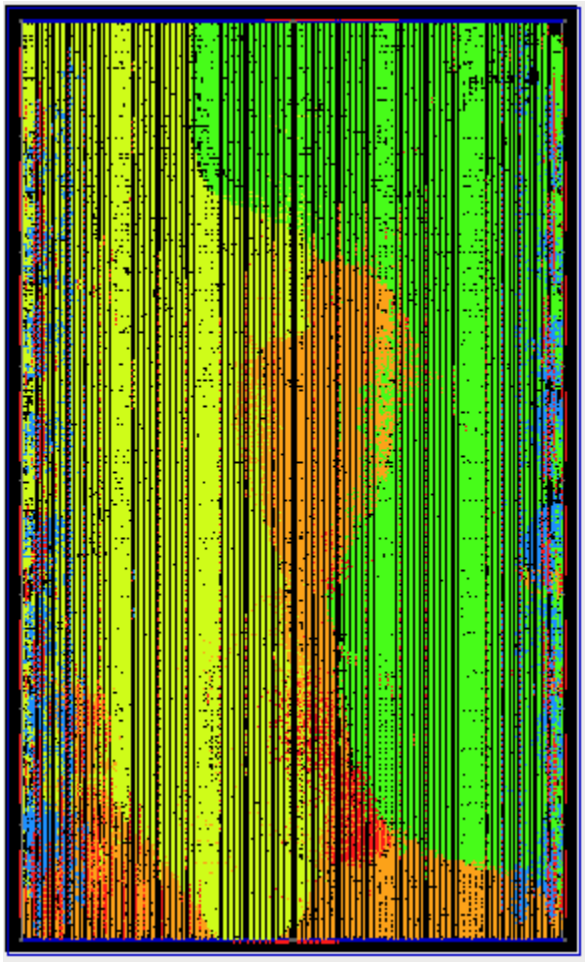

Figure 8: ***Achronix Speedcore-based Implementation of a >1000 hash per second miner ASIC. On-die scratchpad memories are not shown.***

Importantly, this Speedcore-enabled miner is entirely self-contained in one chip with a single power supply. No external components such as memory devices are needed, and only a very few low-speed PCB board traces are needed, keeping the PCB cost lower than with any other solution. Low power consumption permits passive cooling with a small, low-profile heat sink. This compactness, in combination with low power consumption, is imperative when building very high density mining rigs with a customary form factor containing 250 such chips, achieving over 250,000 hashes per second per rig.

For comparison, commercially available high-end Monero mining rigs built using CPUs and multiple GPU cards typically yield 5,000 hashes per second per rig, consume circa 1,000 Watts at said rate, have a form factor equivalent to 10 custom Bitcoin rigs, and cost over $2,000 per rig. A diverse offering of such rigs can be found online, differing in terms of number of GPU cards, performance of the GPUs, form factors, actual availability and cost. The rig chosen to represent GPU-based rigs was the most competitive one that could be found in terms of hashing and form factor efficiency.

The cost of the Speedcore eFPGA-based ASIC depends on its production volume, yields, etc. In the case of the GPU-based rig chosen, the die size of each GPU chip is circa 230 mm^2 which is significantly larger than the die size of a Speedcore eFPGA-based ASIC solution with comparable hashing performance (the technology nodes are similar between the ASIC and the GPU). Again notice that the ASIC has all memories on-die, while the GPU-based solutions require costly external memories which are not even taken into account in this cost and area comparison.

This study also attempts to make a comparison with a hypothetical standalone FPGA-based Monero mining solution.

Table 1: ***Key Characteristics of Speedcore-based eFPGA, a typical High-End Multiple-GPU-based, and a hypothetical High-End Standalone FPGA-based Monero Mining Rig***

Monero Mining Rig Technology	Single Speedcore eFPGA-based ASIC Rig with 250 Devices per Rig	High-End Commercially Available 6-GPU + 1-CPU-based Rig	Standalone 5-high-end FPGA-based + 1 CPU based Rig
Mining Chip Performance (kH/sec)	1.0	0.83	0.3 (estimated)
Number of Chips used for Mining per Rig	250	6	5
Rig Performance (kH/sec)	250	5	1.5
Rig Power Usage (Watt)	2,500	750	unknown, likely similar to GPU-based
Form Factor Ratio per Rig (volume normalized to a single Speedcore eFPGA-based ASIC Rig)	1	10	10 (estimated)
Price per Rig	(see text)	$2,300 (est.)	unknown, likely much higher than GPU-based

Table 2: ***Relative Comparison Between a Speedcore-based, and a Multiple-GPU-based Monero Mining Rig***

Monero Mining Rig Technology	Speedcore eFPGA-based ASIC Rigs (scaled to same form factor volume as GPU /CPU-based Rig)	High-End Commercially Available 6-GPU + 1-CPU-based Rig	Speedcore Advantage
Rig Mining Power – Hashing Performance Efficiency (Watt/(kH/sec))	10	150	15×
Performance – Density Efficiency ((kH/sec)/volume)	2,500	5	500×

In this study, only standard Speedcore eFPGA features were used, such as 20-kb BRAMs, DSP64s, LRAMs and so forth. With custom blocks, further density, power and performance advantages could be achieved. For example, FIR-filter support functions such as pre-adders and filter coefficient register files could be removed from the DSP64 block, since these are unlikely useful in cryptocurrency hash functions. Another example of customization is to add certain wide-logic functions, which are often used in hash functions, in hard form as tiles in custom columns.

The physical size of the Speedcore-based mining ASIC is dominated by the on-die memories and the algorithms' initialization and finalization logic. As these hashing algorithms are not computationally demanding, the amount of logic resources required by the actual inner loop of the algorithm is relatively small.

While this case study focused on Monero, similar cryptocurrency algorithms can be mapped to this solution as well. With those algorithms, similar advantages are achievable with the same Speedcore instance.

Although standalone FPGA-based rigs could, in theory, be built, these are not commercially available. The estimated hash rate of even the largest, costliest of such standalone FPGA-based boards delivers less than one third of the performance of the eFPGA-based miner ASIC. This estimate is based on the large difference in on-die memory size alone, which is much larger in case of the Speedcore eFPGA-based solution. In addition, such rigs would consume much more power, cost much more, and require much more space than a Speedcore eFPGA-based ASIC.

The up-front cost of a standalone FPGA-based rig is estimated to be much higher than a GPU-based one, since the latter is a commodity product and the former is not. At the same time, the estimated mining performance of even a high-end standalone FPGA is less than that of a commodity GPU. Space-wise, standalone FPGA-based solutions would be similar to GPU-based solutions, because in both cases, the devices that execute the mining algorithm would be on a separate circuit board. Both standalone FPGA and GPU solutions are not self-contained like the Speedcore eFPGA-based ASIC solution.

Power-consumption-wise, an standalone FPGA based solution is expected to be better than a GPU-based one, but still worse than an eFPGA-based one.

GPU-based mining systems have been productized, and standalone FPGA based ones not, so pricing, power usage and hashing performance of GPU-based systems are widely published, while no corresponding information could be found about any hypothetical standalone FPGA-based mining product. Due to the lack of specific information, no quantitative relative comparison could be made between standalone FPGA-based rigs. However, due to the aforementioned relative comparisons, standalone FPGA-based solutions are expected to be similar or less efficient than GPU-based ones, and, therefore, much less efficient than standalone FPGA ones. Finally, the standalone FPGA-based solution discussed here would use on-die memories like the eFPGA-based solution, but unlike the GPU-based one. An alternative approach might be to use external memories with the standalone FPGA. Given how the hashing algorithm interacts with the memories, these external memories would need to be HBM memories, driving up cost of the standalone FPGA solution even further, although it might eliminate the performance disadvantage of such a standalone FPGA solution versus GPU-based ones.

Large Memory-Hard Cryptocurrency Algorithms

There is a different class of cryptocurrency algorithms that requires much larger amounts of memory, on order of several gigabytes. Obviously these memories cannot be integrated as memory on the same die as the FPGA core. As with the smaller-memory algorithms, latency and bandwidth requirements determine what the best architecture should look like, but with off-die memory instead. For superior mining performance, again the memory architecture and implementation must be optimized heavily and specifically, which now involves external memory interfaces, bus protocols and memory devices. Going off-chip introduces inherently much larger latency, hence interleaving techniques are required to hide that latency. Exactly what the memory subsystem architecture for this class of algorithms looks like is beyond the scope of this white paper.

Conclusion

When it comes to cryptocurrency mining rigs, time to market is critical. The first mining rigs to be placed online will be the first to start mining currency. ASICs that incorporate eFPGA technology in the form of a custom Achronix Speedcore eFPGA block will eliminate the risk that the silicon solution will need to be re-spun and replaced every six months to accommodate the algorithmic changes.

An ASIC with a Speedcore eFPGA has the following advantages over a GPU-based or stand-alone FPGA based solution:

- Much greater performance
- Much lower power consumption
- Much better density
- Much lower cost
- No dependency on any one silicon vendor

All these benefits combined give the Speedcore eFPGA-based ASIC a mining performance/cost ratio benefit of several orders of magnitude over GPU-based solutions. Speedcore-eFPGA based solutions will have an even larger advantage over standalone FPGAs. The net result is that extremely profitable mining rigs can be built with a Speedcore-based ASIC solution.

For more information about Speedcore technology and how Achronix can help in developing a winning cryptocurrency mining engine contact Achronix today.

About the Author

Raymond Nijssen is the Vice President and Chief Technologist at Achronix. He has over 20 years of experience in the FPGA and EDA industries in various technical and management positions. Mr. Nijssen joined Achronix as Chief Software Architect to manage the software development group, define the foundations and algorithms of the software system, and architect key aspects of the company's FPGA architectures. In his current role, he is responsible for the productization of the company's current products and R&D for new technologies for future products. Prior to joining Achronix, Mr. Nijssen was at Tabula where he was responsible for placement and timing analysis of a time-multiplexed FPGA technology. Prior to Tabula, he was one of the first engineers at Magma Design Automation and held multiple leadership positions in charge of routing and placement, data models and customer deployment of Magma's Blast Plan Pro hierarchy hierarchical virtual prototyping and floorplanning products for very large ASIC designs. Mr. Nijssen received his MSEE degree from Eindhoven University of Technology in The Netherlands, and after that followed its postgraduate program studying EDA for VLSI. He holds several patents related to place-and-route and asynchronous circuit technologies.

Achronix Semiconductor Corporation

2903 Bunker Hill Lane
Santa Clara, CA 95054
USA

Website: www.achronix.com
E-mail : info@achronix.com