The Design of a Bitcoin Hash Processor

ECE 111 Final Project

Winter Quarter 2019

Date: 3/12/19

Team Members: Benjamin Chang, Vi Phung

**Introduction**

Bitcoin is a cryptocurrency, a type of digital currency. It operates without a central bank or a single administrator. Bitcoin can be sent from user to user on the peer-to-peer bitcoin network without the need for intermediaries.

Blockchain is a public ledger that record all bitcoin transactions. It contains a chain of blocks, each block has a hash of previous block up to the genesis block of the chain. The blockchain maintained by a network of communicating nodes running bitcoin software. The network nodes could validate transaction and add to their ledger and notify other nodes. A new block is created about every 10 minutes which gets added into the blockchain and published to all nodes without requiring central oversight. It prevents double spending of a particular bitcoin after it has been used. The blockchain enables users to see if bitcoins exist.

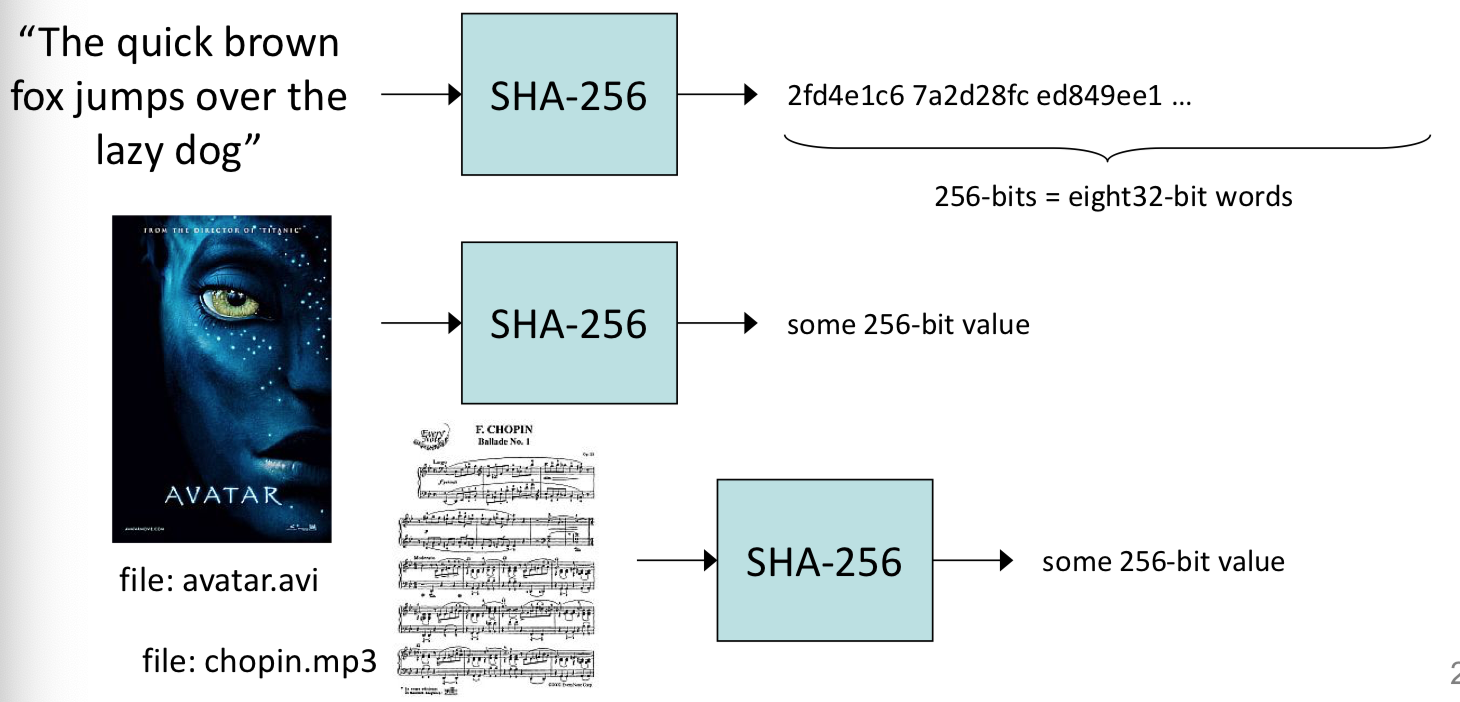
Bitcoin miners help to keep the Bitcoin network secure by approving transactions. Blockchain is secure because each block contains a SHA-256 cyptographic which is connected to previous blocks. To be able to link to the network, a new block must have “proof-of-work”. The “proof-of-work” is a piece of data which is difficult (costly and time consuming) to produce but easy for others to verify. It is a random process with low probability. In addition, it requires many trials and errors on average before a valid “proof-of-work” is generated. Therefore, blockchain is a very secured and unalterable implementation by requiring “proof-of-work” for each new block.

In bitcoin, “proof-of-work” requires miners to look for a number called a “nonce”. This process is done using the SHA256 algorithm. The algorithm’s goal is to compute a unique hash value, “nouce”, for any input “message”, where a “message” can be anything. When the block content is hashed along with this “nonce”, the result will come out numerically smaller than the network’s difficulty target. It is extremely difficult to generate a “nonce”.

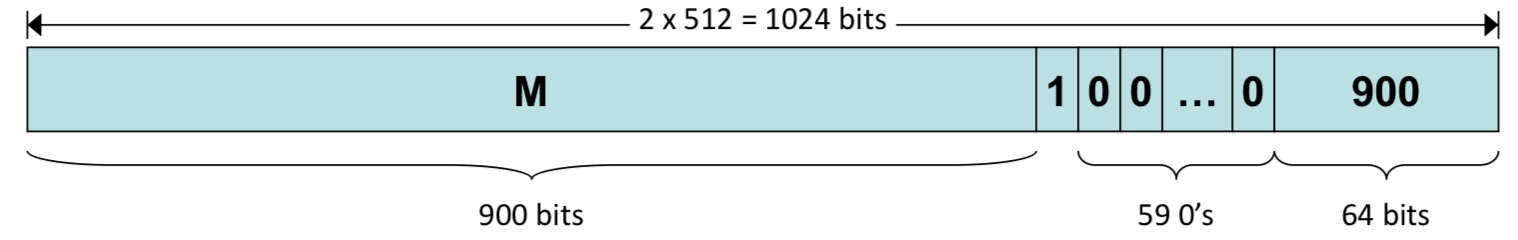
In our final project, we work on a “Bitcoin-Hash Process” that will compute different cryptographic hashes for different value of “nonces”. More details will be shown later in the report.

**Description of the SHA-256 Algorithm**

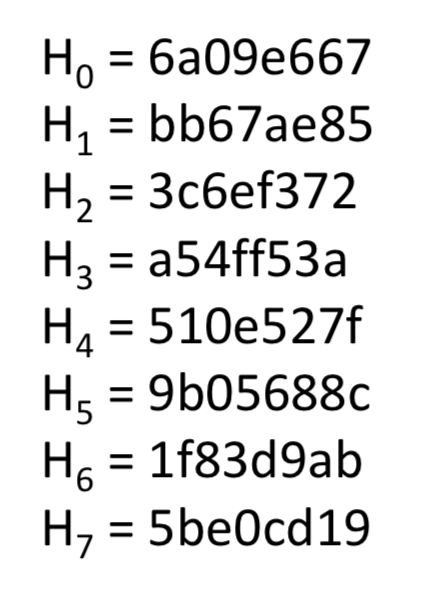
SHA stands for Secure Hash Algorithm. The ultimate goal is to compute a unique hash value for any input “message”. In this project. “Message” could be anything. It is called SHA-256 Algorithm since it returns a 256-bit hash value.

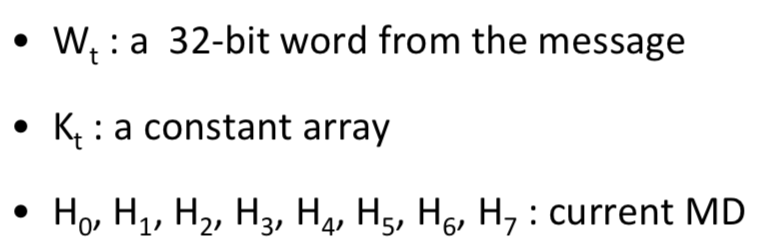
****

SHA-256 has 5 steps total. A message M that needs to run SHA-256 will be fed into the algorithm. In step 1, we need to add a single “1” to the end of the message. Then the rest of the message is padded with “0” until the message equals to 448 mod 512 (modular arithmetic). In step 2, we assign the length of the message as unsigned value at the last 64 bit, left over after the 448 modular arithmetic. In the diagram below, it is an example of where to add 1, 0 and message’s length need to be. The example’s message has 900 bits.

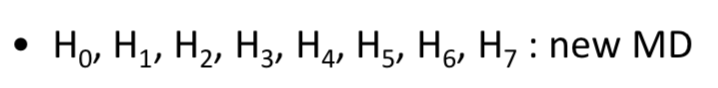


In step 3, we prepare buffer initiation to initialize message digest to eight 32-bit words.



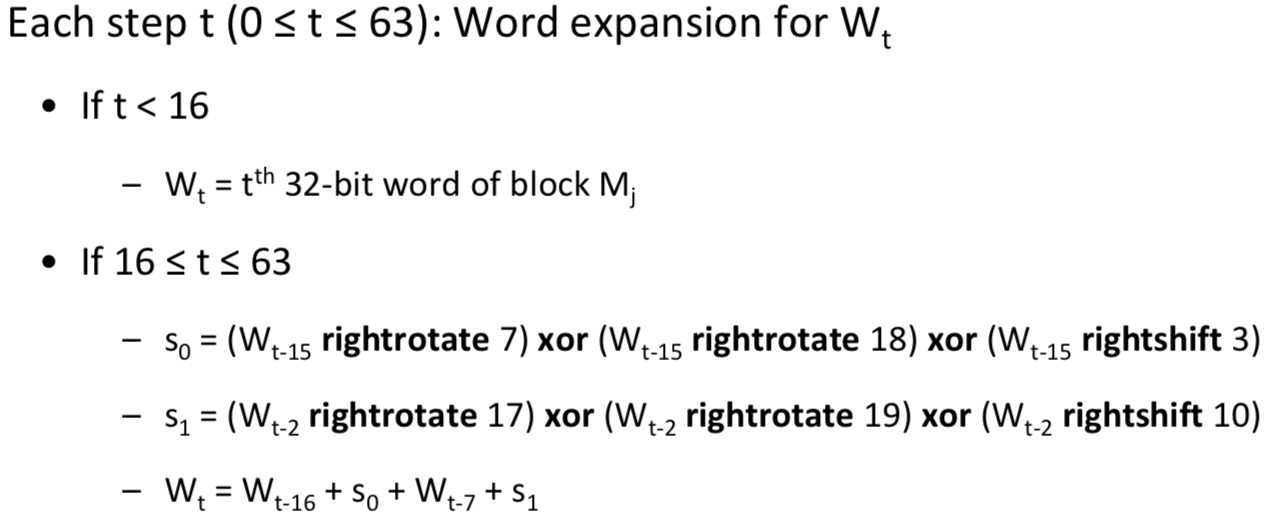
In step 4, we process the message M using the algorithm. At first, the message M is divided into 512-bits blocks, named ; and processed in order, one after the other. The input will have: 

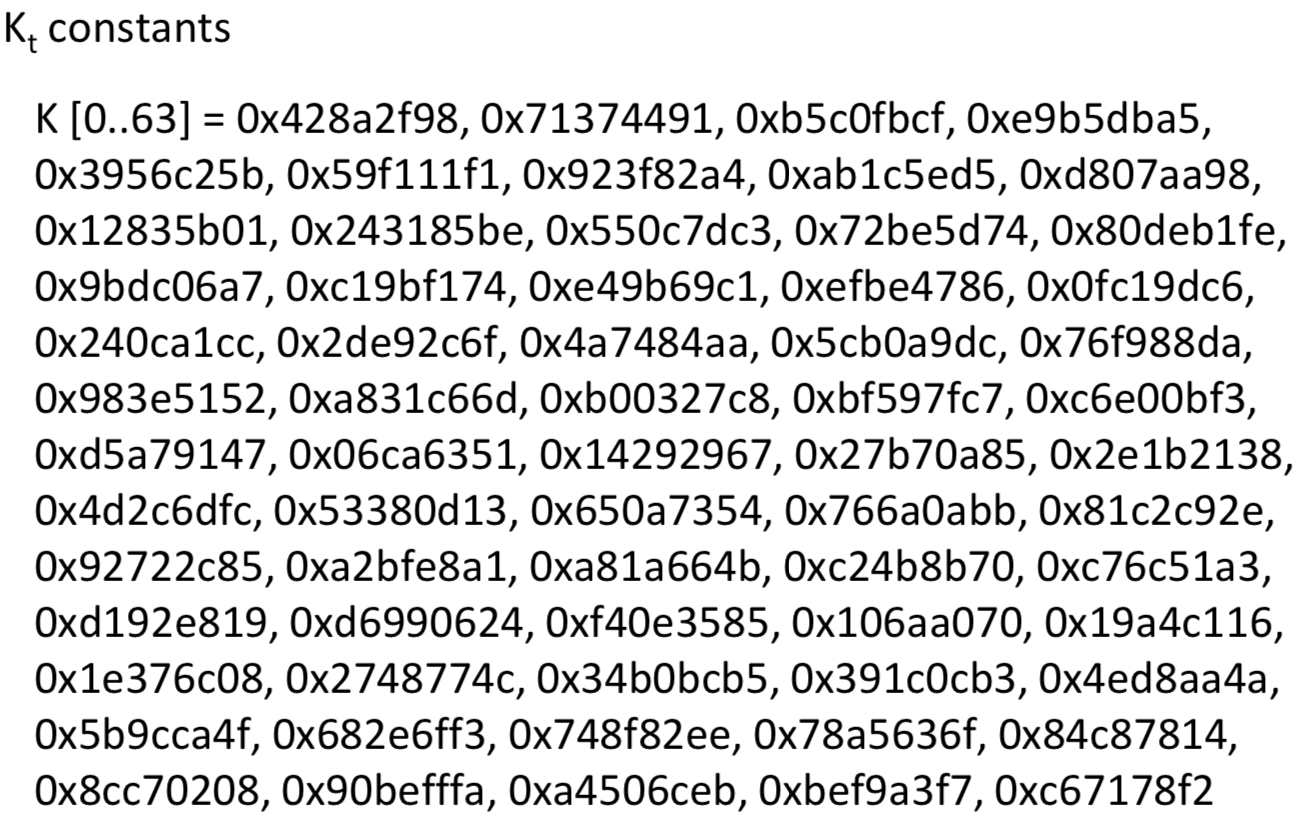
The output will have:



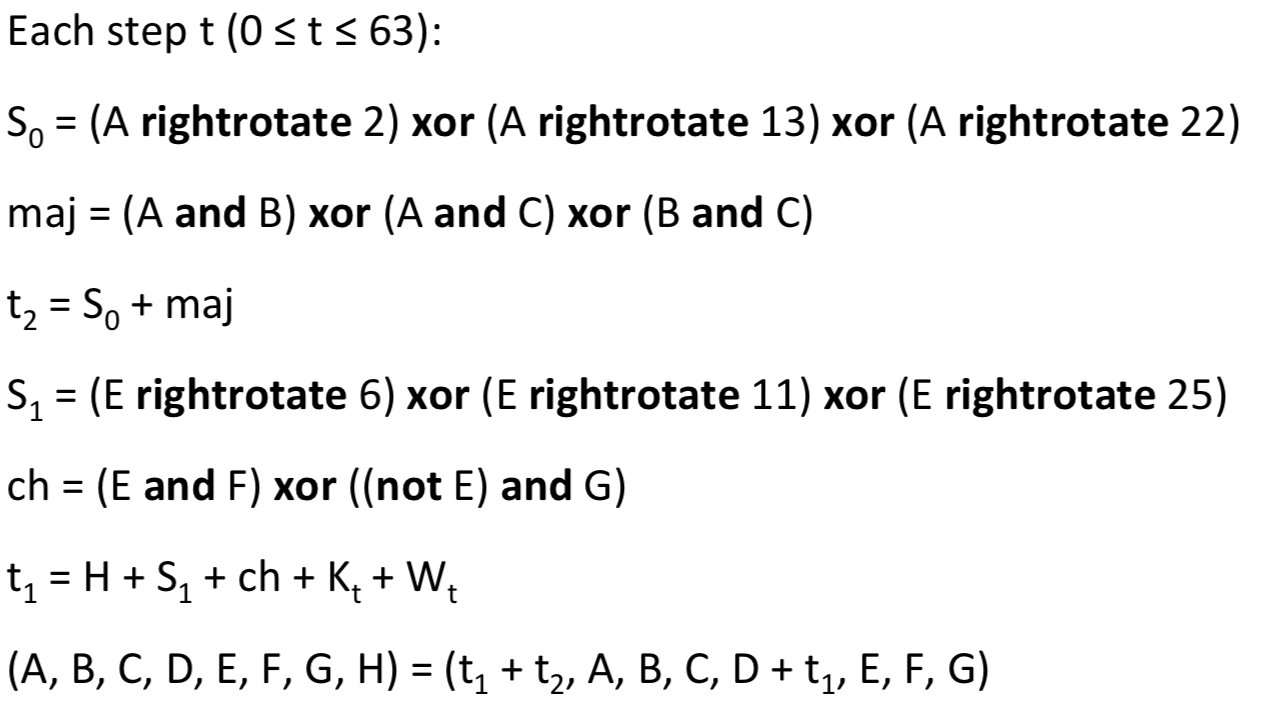
Before processing each , we need to initialize:

There is 64 processing round of 512-bits block. Each round will compute as follows:

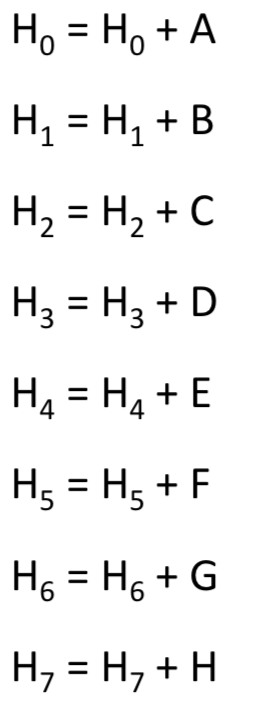


Before continuing to the next step, we need to initialize the constants. 

Now we are ready for the next part of step 4.



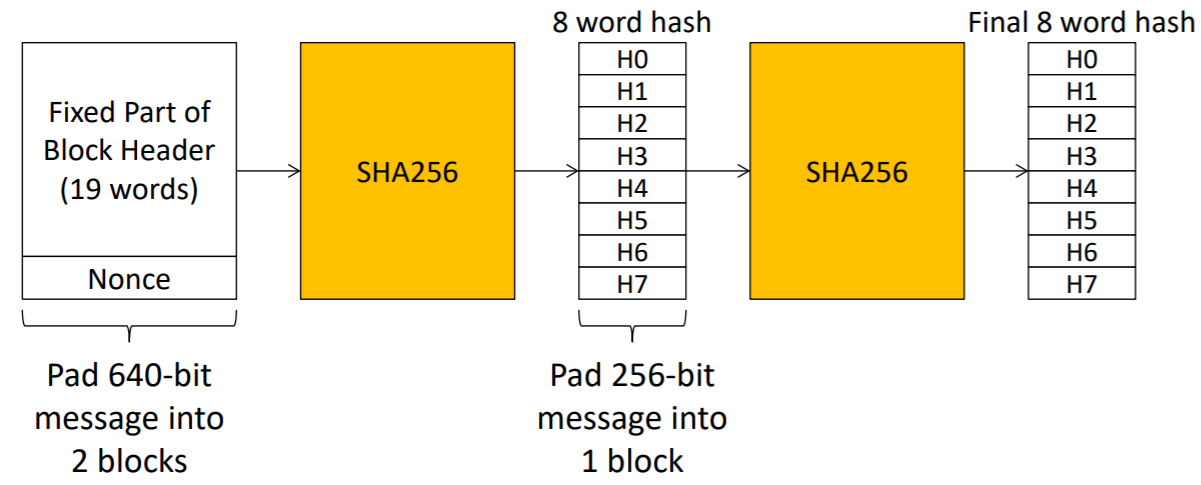
Lastly, after all 64 rounds have ran and each has processed. We can run the last part of step 4.



Step 5 will output all the results after processing all with the 256-bit hash of the message M available in: .

**Description of the Bitcoin-Hash Processor Final Project**

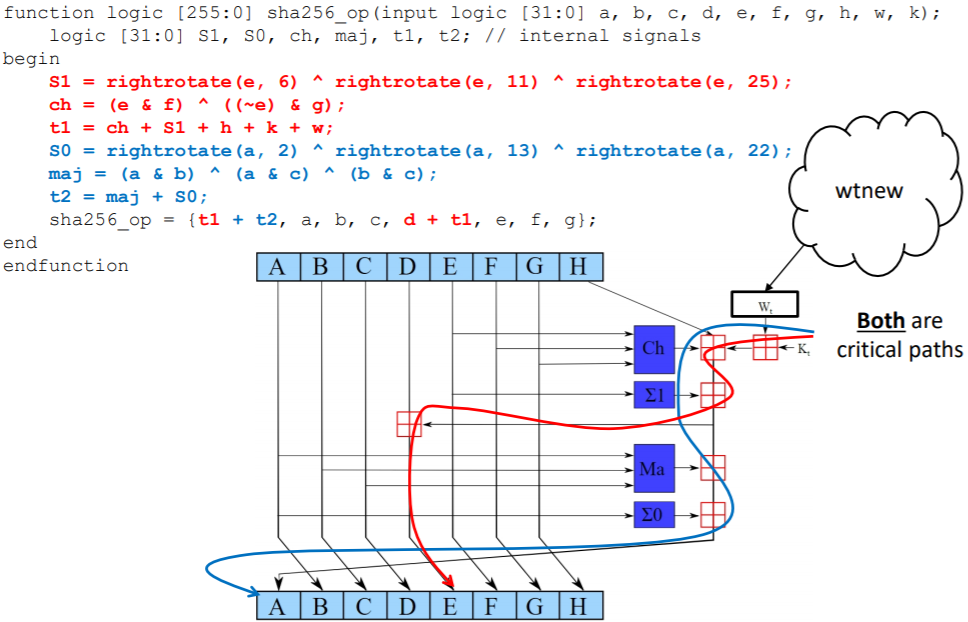
The final project is to design a bitcoin-hash processor utilizing the SHA-256 Algorithm. We need to compute final hashes for 16 nonces, (nonces = 0, 1, … 15) without checking if any is less than target. The result of the first sha256 operation is computed by processing the fixed 19 words of the block header. The input message is divided into 512-bit blocks and each block is run with the sha256 hash operation 2 times total with the output of the first computation fed into the second computation. With this particular project, there will always be only 2 blocks.



**Design Details**

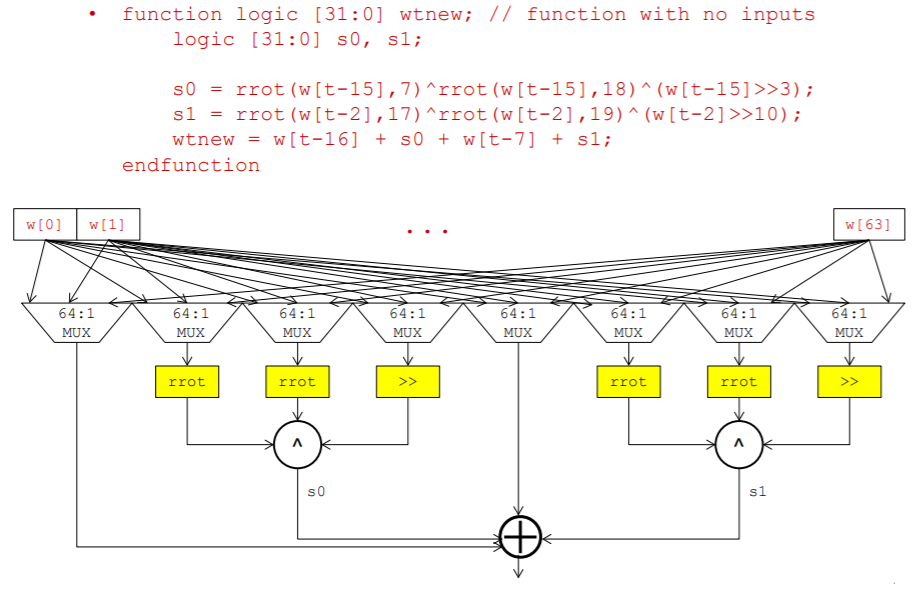
Initially, we created a preliminary implementation of the Bitcoin Hashing project which was optimized for area\*delay. This was used as the basis for our final project where we continued to optimize the original version and created an additional delay optimized version. Throughout this process, we met with Professor Bill Lin and TAs to seek guidance.

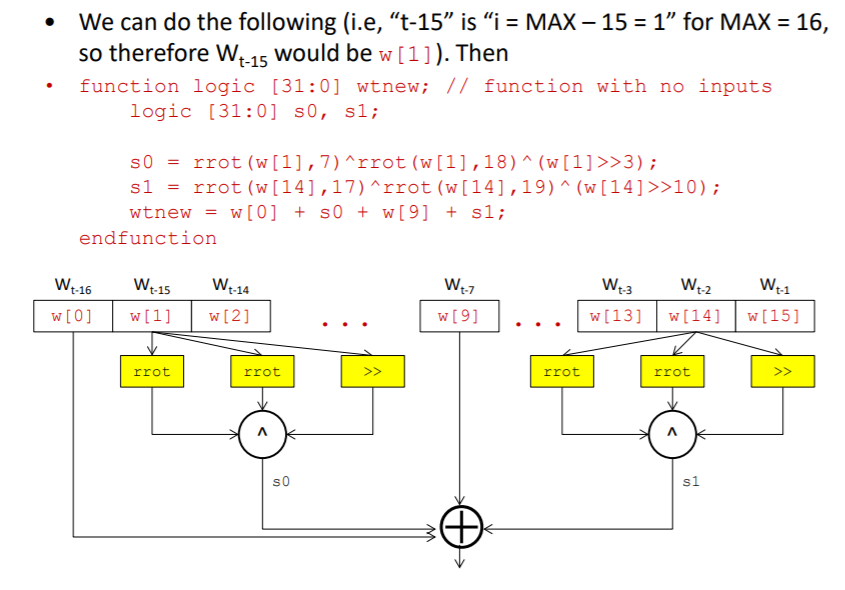
We will discuss a few of the key optimizations strategies we used to increase our design’s efficiency. One of the functions used is the sha256\_op which is shown below.



This function contains a critical path in its computation of t1 and t2 due to the time the device needs to wait for all of the summed variables to be available. We utilized a method to decrease this critical path by precomputing “”. These are the only added variables whose values were previously available for computation. Specifically, the values of ‘k’ and ‘w’ were already known. In addition, the value of ‘h’ could be precomputed one cycle earlier as it is merely ‘g’ from the previous cycle. As a result, by precomputing “”, the delay of our design was reduced significantly.

Another optimization we used was having 16 of 32-bit registers to represent ‘w’ rather than 64 of 32-bit registers. Figures showing the difference between these two methods are shown below. The thought-process behind this is that the newest ‘w’ computed can always be placed in w[15] with all the previously stored values shifted left by 1. This method of storing the ‘w’ values is efficient since only the previous 16 of ‘w’ values are needed every time the sha256 hash operation is computed. This also reduces the number of MUXs required as each new ‘w’ does not have to have a MUX to assign where to store it. Additionally, the number of MUXs is also reduced as they are no longer required in areas of our design like the ‘wt\_new’ function where previous ‘w’ values need to be accessed. This optimization primarily helped with decreasing the area of our design.



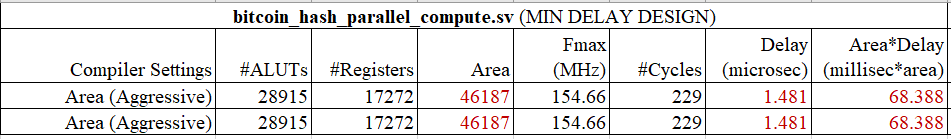


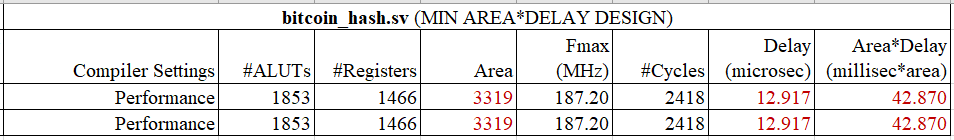
We also optimized our design by noticing that the initial computation of the sha256 hash for the 1st block for the 1st hash is the same for all nonce values and therefore only needs to be computed once. By implementing this into our design, we reduced the number of cycles our design needed to run.

In our final key optimization, we pipelined how we read from memory. This is due to the memory model of our device in which an extra cycle is required to run before data read from memory becomes available. This reduced the number of cycles our design needed to run.

**Summary of Results**

We turned in 2 bitcoin hashing designs. One optimized for decreased delay, and the other for decreased area\*delay. Our results are summarized in the tables below:





**Positive/Negative Impact Considerations**

With the emergence of Bitcoin and other cryptocurrency technologies in today’s world, it is important to consider the effects - both positive and negative - that this new form of currency will have on society. As a decentralized form of currency, Bitcoin can positively impact people living in 3rd world countries, especially minorities in those countries like women, by giving them a way to have access to power. Bitcoin is also positive in that it enables currency transactions to be sped up significantly by reducing transaction delays which it accomplishes by being able to bypass laws and intermediaries which transactions of conventional forms of currency need to go through.

However, there are many negative aspects to Bitcoins and other cryptocurrencies technologies. For one, they encourage illegal black markets which take advantage of the decentralised nature of cryptocurrency databases. These databases keep track of a person’s bitcoins and since they are decentralised, there is no way to recover stolen or laundered currency as only rules which users of a network agree to are enforced which are usually just basic network rules (Hern). It can also be said that the existence of cryptocurrencies such as Bitcoins results in the waste of a lot of energy to mine the currency, but with little positive societal impact to show for it.

Blockchains also have both positives and negatives. Blockchains have a positive impact in that they provide increased transparency and secureness in ledgering currency transactions. They are transparent in that they are available for everyone to access, making each of their transactions easily verifiable by third parties, and secure in that their data is verified each time it is passed to the next node in the chain. Blockchains also speed up transaction time as there is no need for a trusted third party like a bank to facilitate a transaction since the protocol and algorithm used by cryptocurrencies can be trusted to execute the way it is supposed to without human error.

However, there are downsides to blockchains as well. For one, blockchains have to store the entire history of all transactions and thus require more and more memory to store and more time to move around which will eventually make the chain less practical for the typical cryptocurrency user. Blockchains can also have a negative impact on society as if the chain for a cryptocurrency ever needs to be shut down, it won’t be possible due to the chain being stored at each network node (computer with the software to mine the cryptocurrency). Also, with blockchains, transactions are irreversible which means there is more incentive for theft and fraud with the use of a blockchain which can lead to more online crime.

**References**

Hern, Alex. “Bitcoin and Cryptocurrencies – What Digital Money Really Means for Our Future.” *The Guardian*, Guardian News and Media, 29 Jan. 2018, [www.theguardian.com/technology/2018/jan/29/cryptocurrencies-bitcoin-blockchain-what-they-really-mean-for-our-future](http://www.theguardian.com/technology/2018/jan/29/cryptocurrencies-bitcoin-blockchain-what-they-really-mean-for-our-future).

Lin, Bill (2019). *Advanced Digital Design Project* ECE 111 Advanced Digital Design Project Winter 2019 University of California, San Diego website: <http://cwcserv.ucsd.edu/~billlin/classes/ECE111/schedule.php>

Meruem “Pros and Cons of Blockchain Technology - Cryptolume Crypto Academy.” *Cryptolume*, 5 Sept. 2018, cryptolume.co/pros-and-cons-of-blockchain-technology/.

“Proof of Work.” *Proof of Work - Bitcoin Wiki*, en.bitcoin.it/wiki/Proof\_of\_work.