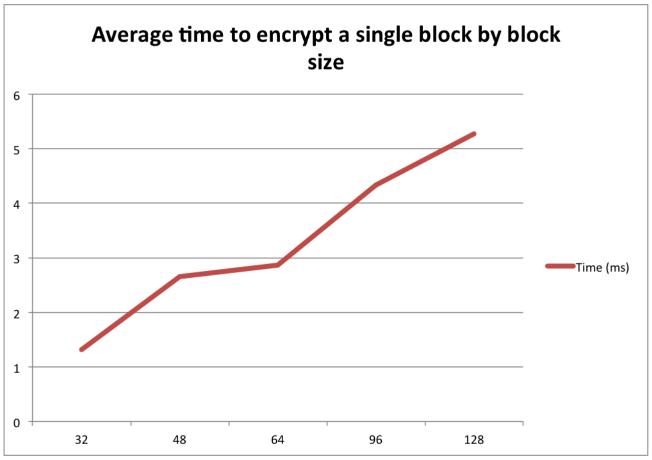
Discussion: Friday 11:30AM
Team: 6 [Smooth Operators]
Name: Cam Herringshaw, Zijie Ku, Jonah Scheinerman
Uniqname: camdroid, kuzijie, jonahsch

Due to the nature of the SPECK family of block cipher, our team implemented the cipher using C++ with various block size from 32 up to 128 bits. The SPECK round function, which is used to scramble the plaintext to cipher text, is broken down into two smaller parts, namely feistel_add and feistel_xor based on Figure 4.2 on the specification, for more efficient implementation. Such approach is based on a branch-and-bound algorithm which is based on the differential characteristic of the round function. By doing so, we are able to achieve the optimal data throughput for both encryption and decryption processes. However, due to nature of such block cipher, the randomness of encryption is limited, as the pattern in the plaintext is also perceivable in the ciphertext. Hence, the SPECK family is vulnerable to Boomerang and Rectangle.

Below is a table of average times to encrypt a block of a given size:

Block Size	Time(ms)
32	1.31805
48	2.65163
64	2.86323
96	4.33653
128	5.26806

As a graph:


Average time to encrypt a single block by block size
Time (ms)