# Computer Architecture Final Project Proposal

*Brenna Manning, Meg McCauley, Griffin Tschurwald, Selina Wang*

Implementation of AES Advanced Encryption Standard algorithm using verilog and potentially FPGA. The AES (Advanced Encryption Standard) is an algorithm used to encrypt data using a key.

We plan to convert text to hex using ASCII, then use AES to further encrypt the message. With the proper key, this message could be decrypted back to the hex numbers and then converted back to plain text. There are three different types of encoding algorithms: hashing algorithms, symmetric-key algorithms and asymmetric key algorithms. With AES, we plan to use a symmetric-key algorithm.

We also plan to implement this verilog code on our FPGA board, and test the speed vs a AES encryption Python library we found. We will also look at where the Verilog can be optimized for speed and try to tune it for performance.

## Deliverables

*MVP*

- Inputting a text string and encoding it into hex (compare premade Python library binascii and code written by our team)
- Come up with a good cipher (cryptographic algorithm) to encrypt our data
- Create block diagram of how the encryption works and how all the pieces fit together
- Accurately portray how information travels through the encryption process.
- Implement AES data encryption in high level verilog module
- Implement decryption of encrypted information in high level verilog module.

*Planned*

- Optimize for speed - Our encryption and decryption run faster than python?

*Potential Stretch Goals*

- Some form of this functional on FPGA board.
- Some sort of GUI with encryption and decryption options
    - Encrypt: takes user text input and key input
    - Decrypt: takes key input - if correct, returns decrypted text, if not returns results of decryption with incorrect key (will be nonsense)
- Explore non-symmetric-key algorithms: asymmetric, hashing, etc.
- Modules communicating with python program

**More References:**

AES Federal Doc. (2001) - http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
Explanation of AES Standard - http://searchsecurity.techtarget.com/definition/Advanced-Encryption-Standard
AES Wikipedia - https://en.wikipedia.org/wiki/Advanced_Encryption_Standard
ASCII Conversion - http://www.asciitable.com/
Python cryptography library - https://pypi.python.org/pypi/pycrypto
Another Python library - https://docs.python.org/2/library/binascii.html
Other algorithms: asymmetric, hash, etc - http://research.ijcaonline.org/volume44/number16/pxc3878380.pdf

**Work Plan:**

| **Task** | **Estimated Time** | **Due Date** |
|---|---|---|
| Work Plan and finish proposal | 1 hour | Dec 1 |
| Research AES and outline our verilog module | 2 hours | Dec 3 |
| Block Diagram of Encrypt | 2 hours | Dec 5 |
| Block Diagram of Decrypt | 2 hours | Dec 5 |
| Writing Encryption Verilog | 3 hours | Dec 7 |
| Midpoint Check In | | |
| Writing Decryption Verilog | 3 hours | Dec 7 |
| Test Bench | 3 hours | Dec 9 |
| Wiring everything together and debug stuff | 3 hours | Dec 11 |
| $$$ MVP Achieved $$$ | | |
| Load and get working on FPGA | 3 hours | Dec 14 |
| Speed test against python cryptography library | 2 hours | Dec 14 |
| Optimize Verilog for speed | 2 hours | Dec 14 |
| Win CompArch | | |

**Scheduled Meetings**

| | |
|---|---|
| Thursday 6:30-8:30 *2hrs* | Griffin Meg (Library) |
| Friday 7-8:30 *3.5hrs* | Brenna Griffin Selina (Library) |
| Saturday 12-2 *5.5hrs* | Brenna Selina (Library) |
| Saturday 6-8 *7.5hrs* | Everyone (Library) |
| Sunday 9-11:30 *10hrs* | Everyone (EH4) |
| Monday 1:30-3:00 *11.5* | Griffin Meg (Library) |
| Monday 3:30-5:00 *13* | Brenna Selina (Library) |
| Tuesday 3:30-5:30 *15* | Brenna Meg Selina<3 (Library) |
| Tuesday 8:00-10:00 *17* | Tentatively Brenna Meg Selina (EH3) |
| Wednesday 1:00-3:00 *19* | Brenna Meg Selina (Library) |
| Thursday 3:30-5:00 *20.5* | Brenna Selina (Library) |
| Thursday 7:00-9:00 *22.5* | Griffin Meg |
| Friday 1:00-3:30 *25* | Brenna Griffin Meg |
| Sunday 9:00-11:00 *27* | errybody |
| Monday 12:00-2:00 *29* | Everyone <3 |
| Tuesday DEMO! | |