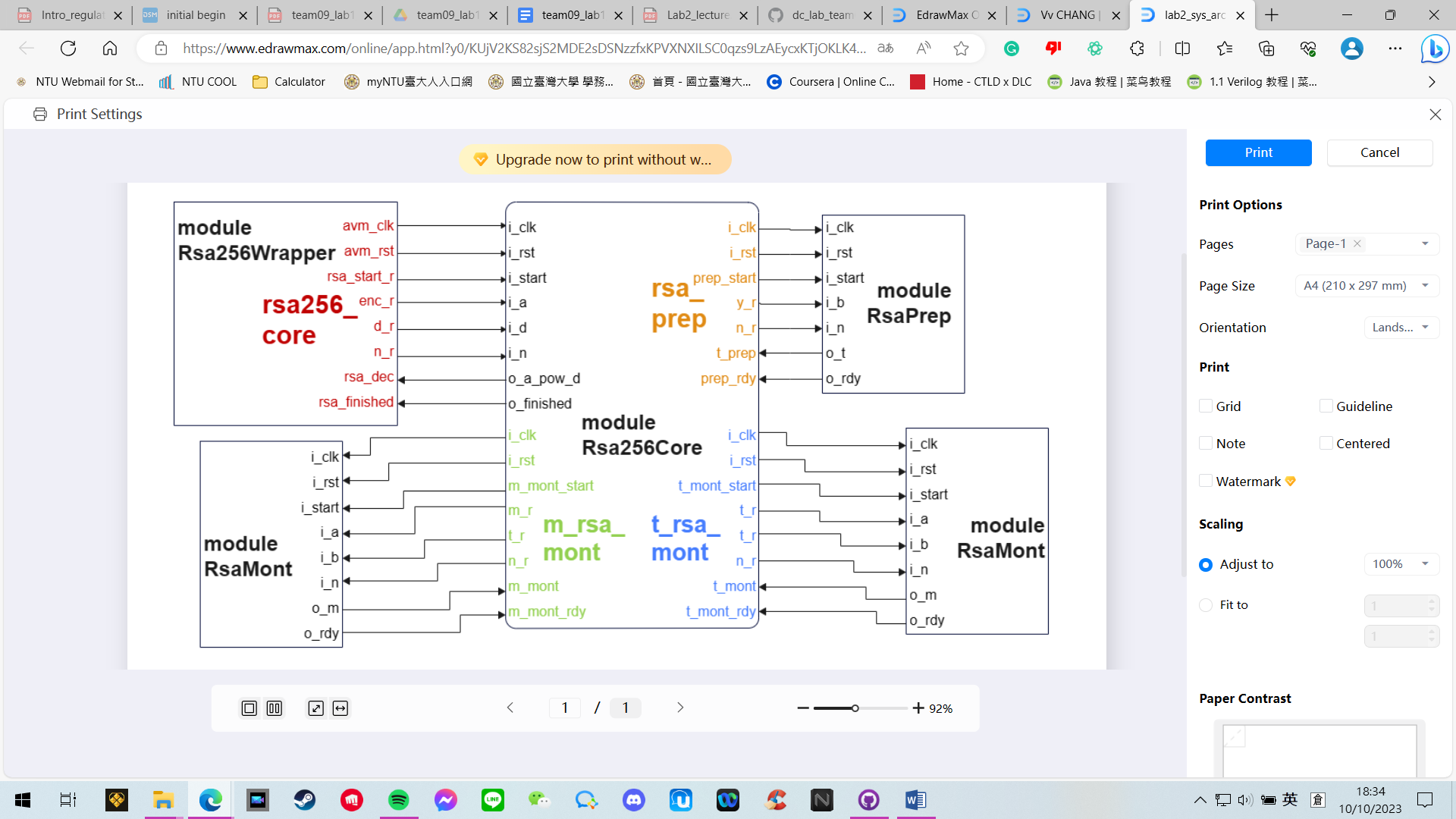
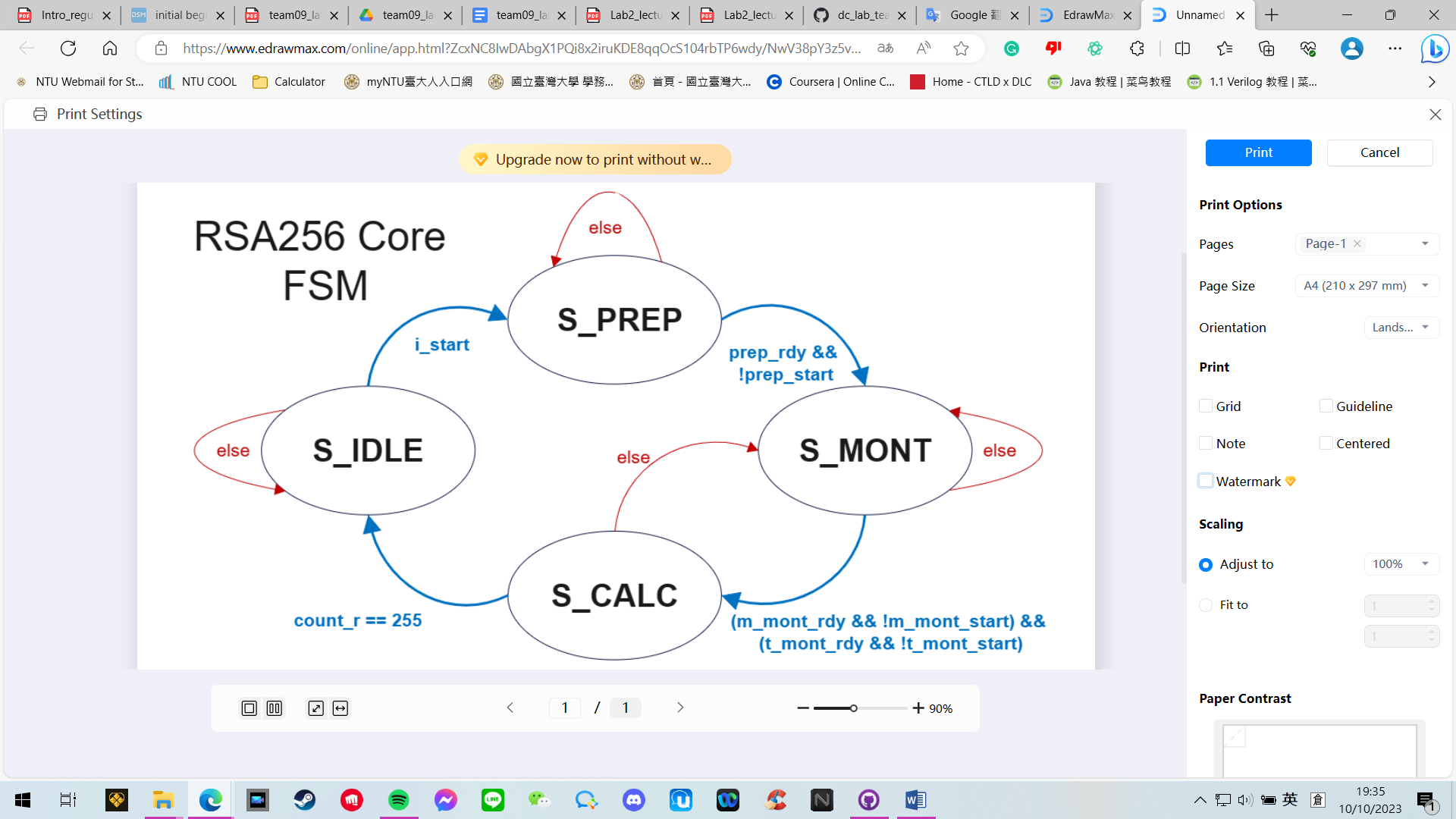
Team09\_Lab2\_Report

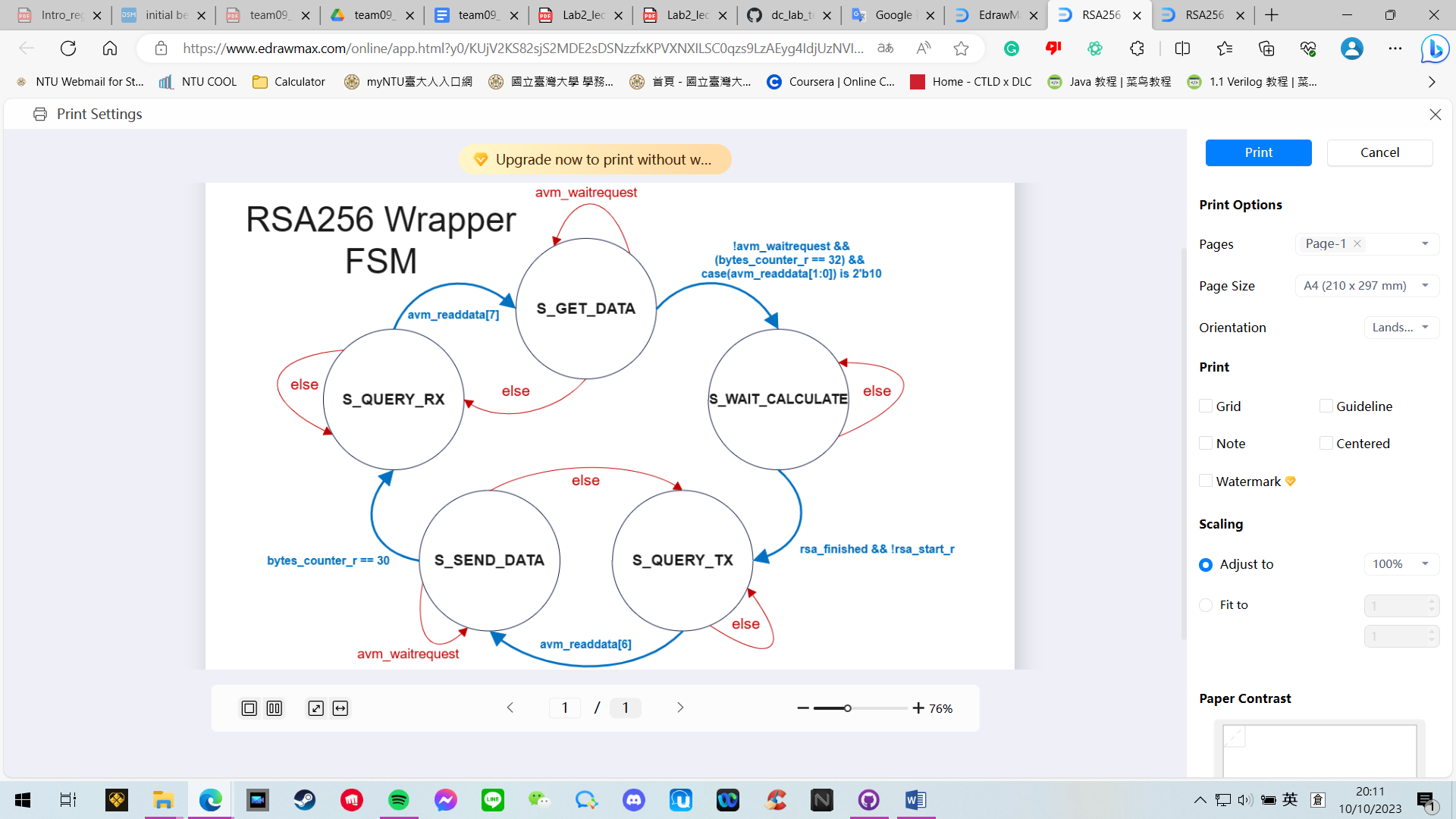
File Structure:

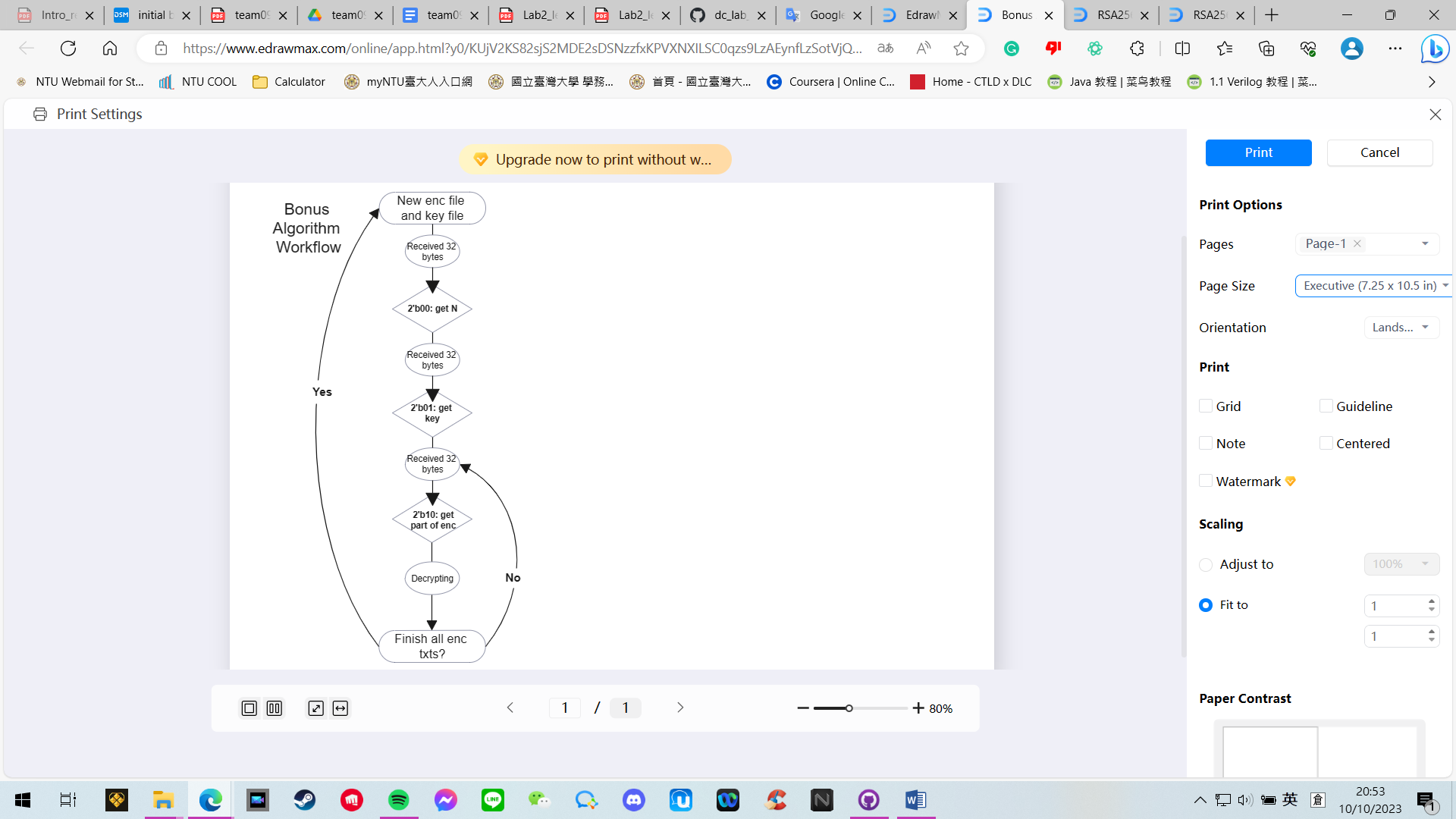
* team09\_lab2/team09\_lab2\_report : This file contains information about the source code in the directory and the instructions of lab1.
* team09\_lab2/src/Rsa256Core.sv : We have implemented RSA256 with Montgomery Algorithm of lab2 in this file.
* team09\_lab2/src/Rsa256Wrapper.sv : We have implemented RSA256 Wrapper with RS232 Protocol of lab2 in this file.
* team09\_lab2/src/DE2\_115 : For the files in this folder, we have only modified DE2\_115.sv to replace the module with the rsa\_qsys module we generated.

System Architecture:

Hardware Scheduling:





**Bonus**

We have implemented the method that multiple ciphertexts can be decrypted continuously without reset.

Python rs232.py:

* We added one byte after each 32 bytes segment
* Add s.write(b”0x00”) after key[0:32]
* Add s.write(b”0x01”) after key[32:64]
* Add s.write(b”0x02”) after every iteration of enc[i:i+32]

Module Rsa256Wrapper.sv:

* At the state S\_GET\_DATA, if the bytes\_counter reaches 32 times, we determine which cases of the last byte that is received in avm\_readata since this byte is intentionally inserted every 32 bytes in rs232.py by us.
* For case 2'b00, this means that the 32 bytes we received previously were N, so we save it in n\_w.
* For case 2'b01, this means that the 32 bytes we received previously were key, so we save it in d\_w.
* For case 2'b02, this means that the 32 bytes we received previously were ciphertext, so we save it in enc\_w.

Fitter Summary:

Timing Analyzer:

Problem & Solution:

1. Simulation Abort : "Protocol not met %m"
   1. 解
2. Simulation Fxxx : "Simulation time too long."
   1. 解
3. Timing assignment not met
   1. 解
4. 其他:
   1. 其他:

Conclusion & Suggestion: