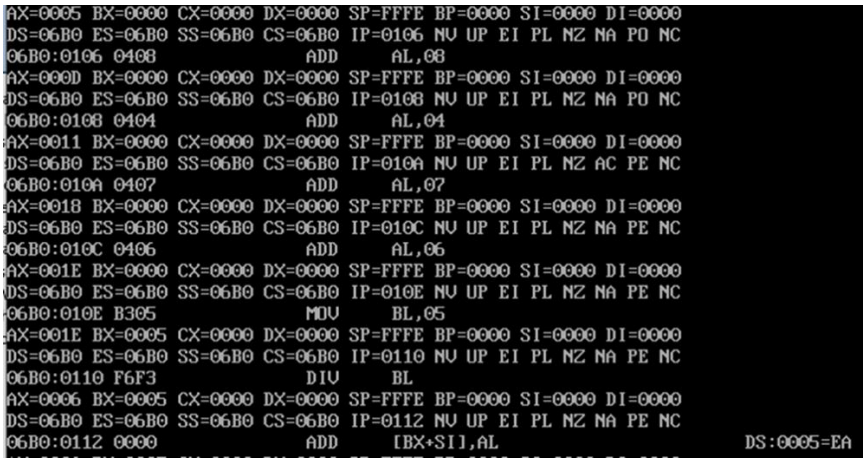
Jakob Michael Palomo
2ITG

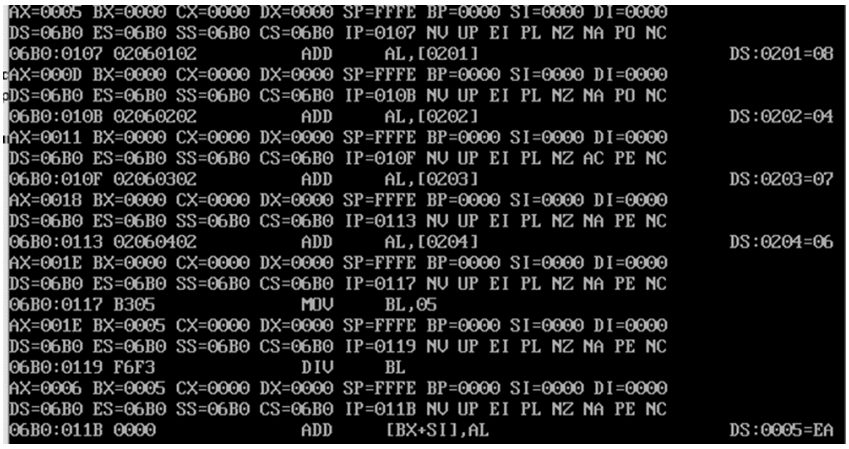
Post Report 9

After doing experiment 9, I learned that Assembly language programming involves writing low-level code that is executed directly by the CPU. In assembly programming, data can be stored in registers or memory locations, and two commonly used methods for this are direct addressing and memory addressing.

Direct addressing is where storing data directly in registers, such as the al, bl, and ah registers, located within the CPU. It is known for its speed and efficiency, but registers have limited storage capacity, which makes it difficult to store large amounts of data.



Memory addressing on the other hand involves storing data in memory locations that are accessed by the CPU. It provides more flexibility and ease of modification but is generally less efficient due to the need for additional commands to load and store data.



Jakob Michael Palomo
2ITG

The choice between the two methods depends on the specific requirements of the program. If speed and efficiency are crucial, direct addressing may be the best option. If flexibility and ease of modification are more important, memory addressing may be preferred. Understanding the differences between these two methods is essential for writing efficient and effective assembly language programs. Programmers need to consider the requirements of their program to decide which method to use for optimized performance.