Aditya Raj 19BCS1706-- Final Practical

Algorightm:

1. Load 00 in a register C (for borrow)
2. Load two 8-bit number from memory into registers
3. Move one number to accumulator
4. Subtract the second number with accumulator
5. If borrow is not equal to 1, go to step 7
6. Increment register for borrow by 1
7. Store accumulator content in memory
8. Move content of register into accumulator
9. Store content of accumulator in other memory location
10. Stop

Code:

Memory	Mnemonics	Operands	Comment	
2000	**MVI**	**C, 00**	**[C]<-00**	
2002	**LHLD**	**2500**	**[H-L] <- [2500]**	
2005	**MOV**	**A,H**	**[A] <- [H]**	
2006	**SUB**	**L**	**[A] <- [A] – [L]**	
2007	**JNC**	**200B**	**Jump if no borrow**	
200A	**INR**	**C**	**[C] <- [C] + 1**	
200B	**STA**	**2502**	**[A] -> [2502], Result**	
200E	**MOV**	**A,C**	**[A]<-[C]**	
2010	**STA**	**2503**	**[A]->[2503],Borrow**	
2013	**HLT**		**Stop**	

Screenshots of simulation: