

## **SWAPPING OF NUMBERS**

**EXP NO: 15**

### **AIM:**

To compute swapping of numbers using 8085 processor.

### **ALGORITHM:**

- 1)  
Load a 8-bit number from memory location into accumulator.
- 2)  
Move value of accumulator into register H.
- 3)  
Load a 8-bit number from next memory location into accumulator.
- 4)

Move value of accumulator into register D.

5)

Exchange both the registers pairs.

6)

Halt

**PROGRAM:**

LDA 2001

MOV B,A

LDA 2002

STA 2001

MOV A,B

STA 2002

HLT

**INPUT:**

Address (Hex)	Address	Data
07D1	2001	20
07D2	2002	7

## OUTPUT:

The screenshot displays the GNUSim8085 - 8085 Microprocessor Simulator interface. The main window is divided into several sections:

- Registers:** A table showing the state of various registers. The Accumulator (A) contains the value 07. Other registers like BC, DE, HL, PSW, PC, SP, and Int-Reg are also listed with their current values.
- Flag:** A section showing the status of various flags (S, Z, AC, P, C) and their current states.
- Assembly Code:** A central area displaying the assembly program being executed. The code includes instructions like `<Program title>`, `jmp start`, `:data`, `:code`, `start: nop`, `lda 2001`, `mov B,A`, `lda 2002`, `sta 2001`, `mov A,B`, `sta 2002`, `hlt`, and `hlt`.
- Memory:** A table on the right side showing the memory contents. It lists addresses (Hex) and data values. The memory at address 2001 contains the value 20, and at address 2002, it contains the value 7. Other memory locations are shown with zero values.
- Decimal - Hex Conversion:** A section for converting between decimal and hexadecimal values.
- I/O Ports:** A section for managing input and output ports.
- Assembler Message:** A log at the bottom right showing the message "Program assembled successfully".

The simulator is currently in an "Idle" state, as indicated by the status bar at the bottom.

## RESULT: Thus

the program was executed successfully using 8085 processor simulator.