# **LCD Interfacing with 8086 Microprocessor (Proteus Simulation)**

## **Introduction**

This report details the step-by-step process of interfacing a **16x2 LCD** with the **Intel 8086 microprocessor** in **Proteus simulation**. The project involves using an **ADC0804** for analog-to-digital conversion and a **74HC373 latch** for data handling. The objective is to display sensor-based pressure data on the LCD using an 8-bit mode communication method.

## **Components Used (Proteus Simulation)**

1. **Intel 8086 Microprocessor (Virtual Component in Proteus)**
2. **16x2 LCD Display**
3. **ADC0804 (Analog-to-Digital Converter)**
4. **74HC373 Latch**
5. **Virtual Power Supply (5V and 12V in Proteus)**
6. **Clock Generator**
7. **Push Buttons and Potentiometer (for Testing Input Values in Simulation)**
8. **Resistors and Capacitors (as needed in Proteus)**
9. **Proteus Simulation Software**
10. **EMU8086 or MASM for Assembly Code Simulation**

## **Step-by-Step Implementation in Proteus**

### **Step 1: Setting Up the Proteus Environment**

* Open **Proteus 8 Professional** and create a new project.
* Add the following components from the Proteus library:
  + **8086 Microprocessor**
  + **16x2 LCD (LM016L or Equivalent)**
  + **ADC0804**
  + **74HC373 Latch**
  + **Clock Generator** for **8086 Timing Requirements**
  + **Power Supply (5V and 12V)**

### **Step 2: Circuit Design in Proteus**

* **8086 Microprocessor** sends control and data signals to the **74HC373 latch**.
* The **74HC373 latch** acts as a buffer between the **8086 and LCD**.
* The **ADC0804** converts the analog sensor data (such as pressure) into digital form before sending it to the LCD.
* The **oscillator circuit** is configured to provide the necessary clock pulses to **8086 and ADC0804**.
* The **connections for the LCD**:
  + **RS (Register Select)** → Connected to a control pin from **8086**.
  + **RW (Read/Write)** → Connected to **ground** (for write-only operation).
  + **E (Enable Pin)** → Controlled by **8086** to send data.
  + **D0-D7 (Data Pins)** → Connected through the **74HC373 latch**.

### **Step 3: Configuring ADC0804 for Analog-to-Digital Conversion**

* The ADC receives an **analog input from a potentiometer (simulating a sensor)**.
* The **converted digital output** is sent to the **8086 microprocessor**.
* The 8086 reads the ADC values and processes them for display on the LCD.

### **Step 4: Writing Assembly Code for LCD and ADC Control**

* **Initialize the LCD** by sending required command instructions.
* **Read ADC0804 data** and store it in **8086 registers**.
* **Send processed data** to the LCD for display.
* **Loop continuously** to update values in real-time.

### **Step 5: Running the Simulation in Proteus**

1. **Load the Assembly Code into the 8086 Emulator (EMU8086 or MASM)**.
2. **Run the Code** and observe the signals in **Proteus Logic Analyzer**.
3. **Check the LCD Display** for proper data output.
4. **Adjust the Potentiometer** in Proteus to simulate different sensor values and verify LCD updates.

### **Step 6: Debugging and Testing**

* Ensure **power connections** are correctly set in Proteus.
* Verify **8086 to latch to LCD data flow**.
* Test ADC0804 output in Proteus using a virtual **Logic Analyzer**.
* Debug any **assembly code errors** by checking data movement in registers.

### **Step 7: Observations and Results**

* The LCD successfully displays **real-time ADC values**.
* The ADC0804 converts analog signals effectively and sends them to the **8086 for processing**.
* The **Proteus simulation works correctly**, proving that the LCD interfacing with **8086 is functional**.

## **Conclusion**

This project successfully demonstrates **LCD interfacing with the 8086 microprocessor in Proteus** using an **ADC0804 and 74HC373 latch**. The implementation allows real-time **analog-to-digital data conversion and display**, making it a useful setup for **simulating industrial applications** where microprocessor-controlled displays are required.