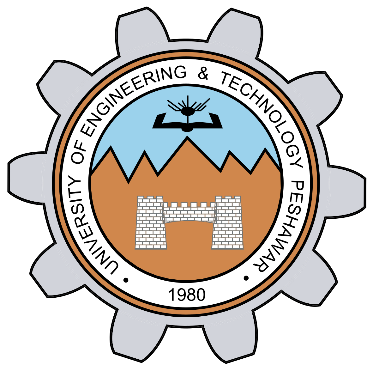
**Lab:1**



**MBSD Lab**

**Spring 2023**

**Submitted by:**

**Maaz Habib**

**Reg no:**

**20Pwcse1952**

“On my honour, as a student of University of Engineering and Technology Peshawar, I have neither nor received unauthorized assistance on this academic work”.

**Submitted to:**

**Dr: Amaad Khalil**

**Title: Introduction to 8051 Microcontroller & Installation of Software (Proteus and Keil)**

**Abstract:**

This lab report aims to give you a clear understanding of the 8051 microcontroller and how to set up software tools like Proteus and Keil. The 8051 microcontroller is a popular and widely used component in embedded systems. To work with this microcontroller effectively, it's important to have the right software tools for programming and simulating its behaviour. This report will explain the basic concepts of the 8051 microcontrollers, such as its structure, features, and uses. Additionally, it will provide simple, step-by-step instructions on how to install Proteus and Keil software, which are commonly used for designing, simulating, and fixing projects based on the 8051 microcontrollers.

**1. Introduction:**

The 8051 microcontroller is a small computer chip that was created by Intel in 1980. It's called an 8-bit microcontroller because it can process 8 bits of data at a time. People really like using the 8051 because it's easy to understand, can be used in many ways, and you can find it almost everywhere. It's used in lots of things like controlling machines in factories, making homes smart and automated, building robots, and even in cars and other vehicles.

**2. 8051 Microcontroller Architecture:**

The 8051 microcontroller has a few important parts that work together. These parts include the Central Processing Unit (CPU), Random Access Memory (RAM), Read-Only Memory (ROM), Serial Communication Interface (UART), Input/Output (I/O) ports, and Timers/Counters. The CPU is like the brain of the microcontroller. It follows instructions and carries out tasks stored in the program memory. The RAM is like a temporary storage space that holds data while the microcontroller is running. The ROM stores the program code and important data that doesn't change. The UART allows the microcontroller to talk to other devices using a serial connection. It's like a communication link. The I/O ports are like the microcontroller's hands and eyes. They let it connect to buttons, sensors, displays, and other external things. The Timers/Counters are used to measure time and count events. They help the microcontroller do things at specific times or keep track of how many times something has happened.

**3. Proteus Software Installation:**

Proteus is a software tool for designing and simulating electronic circuits. It allows users to create virtual prototypes of microcontroller-based systems and simulate their behaviour. Here are the steps to install Proteus:

**Step 1:** Download the Proteus software installation package from the official website.

**Step 2:** Run the installation package and follow the on-screen instructions to install Proteus**.**

**Step 3:** Once the installationis complete, launch Proteus from the desktop shortcut or Start menu.

**Step 4:** Upon launching Proteus, create a new project and select the appropriate microcontroller model, such as AT89C51.

**Step 5:** Design the circuit by adding components from the Proteus library and interconnecting them as required.

**Step 6:** Write the program code in the integrated development environment (IDE) provided by Proteus or import an existing program.

**Step 7:** Simulate the circuit by running the program and observing the behaviour of the microcontroller and connected peripherals.

**4. Keil Software Installation:**

Keil is a popular Integrated Development Environment (IDE) for developing software for various microcontrollers, including the 8051. It provides a comprehensive toolchain for writing, compiling, and debugging embedded applications. Here are the steps to install Keil:

**Step 1:** Download the Keil software installation package from the official website.

**Step 2:** Run the installation package and follow the on-screen instructions to install Keil.

**Step 3:** Once the installation is complete, launch Keil from the desktop shortcut or Start menu.

**Step 4:** Create a new project and select the appropriate microcontroller model, such as AT89C51.

**Step 5:** Write the program code in the editor provided by Keil.

**Step 6:** Configure the build settings, including the memory model and compiler options.

**Step 7:** Compile the program code