

## Lab sheet 02

### Title:

Introduction to Keil IDE

### Aims:

- To understand how Keil IDE works
- Create Sample assembly Language project

### Tasks:

- Experience with Keil IDE
- Run the project using Keil IDE
- Move values between registers
- Simple Addition

### Activities:

Copying a Value Between Registers Using Assembly Language

MOV R0, #100

MOV R1, R0

The screenshot shows the µVision IDE interface with two windows open:

- Top Window (Assembly View):** Displays the assembly code for Lab02.s. The code includes directives like `area output,code,readonly`, `MOV R0,#100`, `MOV Z1,Z0`, and `STOP B STOP`.
- Bottom Window (Registers View):** Shows the CPU register state. The PC register is highlighted and shows the address of the current instruction. Other registers (R0, R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14) are listed with their current values.
- Memory View:** A table titled "Call Stack + Locals" shows the value of the variable `__asm_0x0` at address `0x00000000`.
- System Status:** The status bar at the bottom indicates the system is running at 28°C with a partly cloudy weather forecast, and the time is 12:29 PM on 12/29/2025.

## Discussion:

This code demonstrates how to transfer a value between registers.

The value 100 is first entered into register R0 using `MOV R0, #100`.

#100 is an immediate value in this case.

The value from R0 is then copied into R1 via MOV R1, R0.

Both registers now hold the same value.

Final outcome

R0 = 100.

R1 = 100.

In assembly language, this technique is used to transfer or copy data between registers.

Reference:

Keil Software