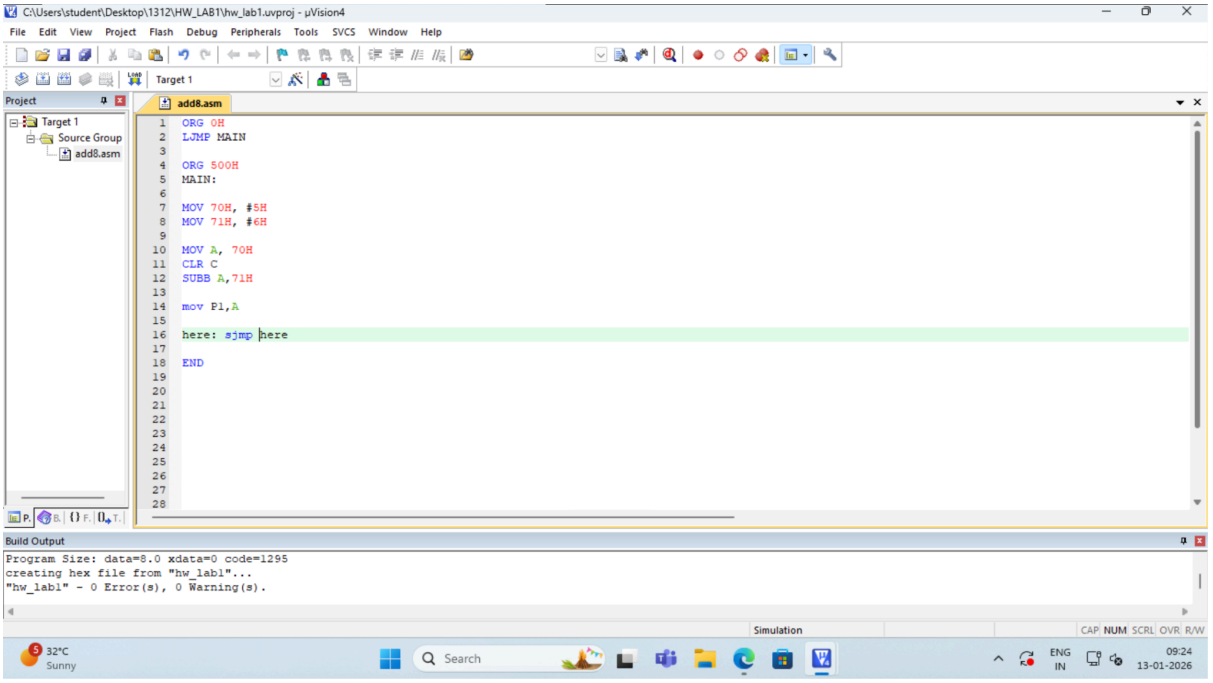
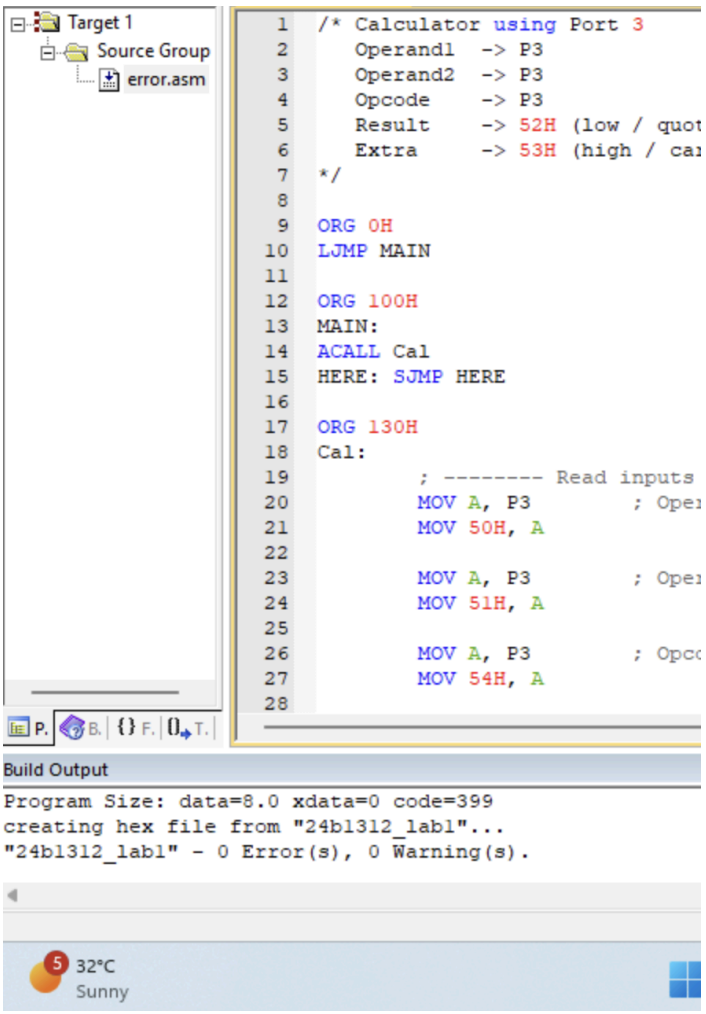
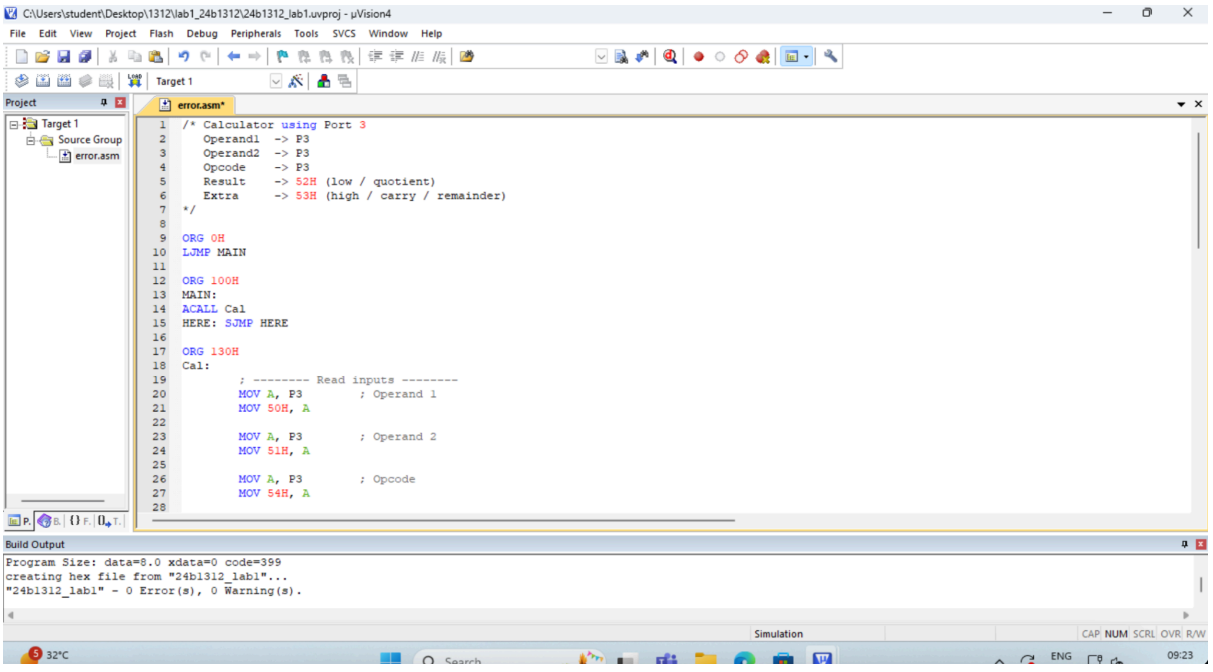
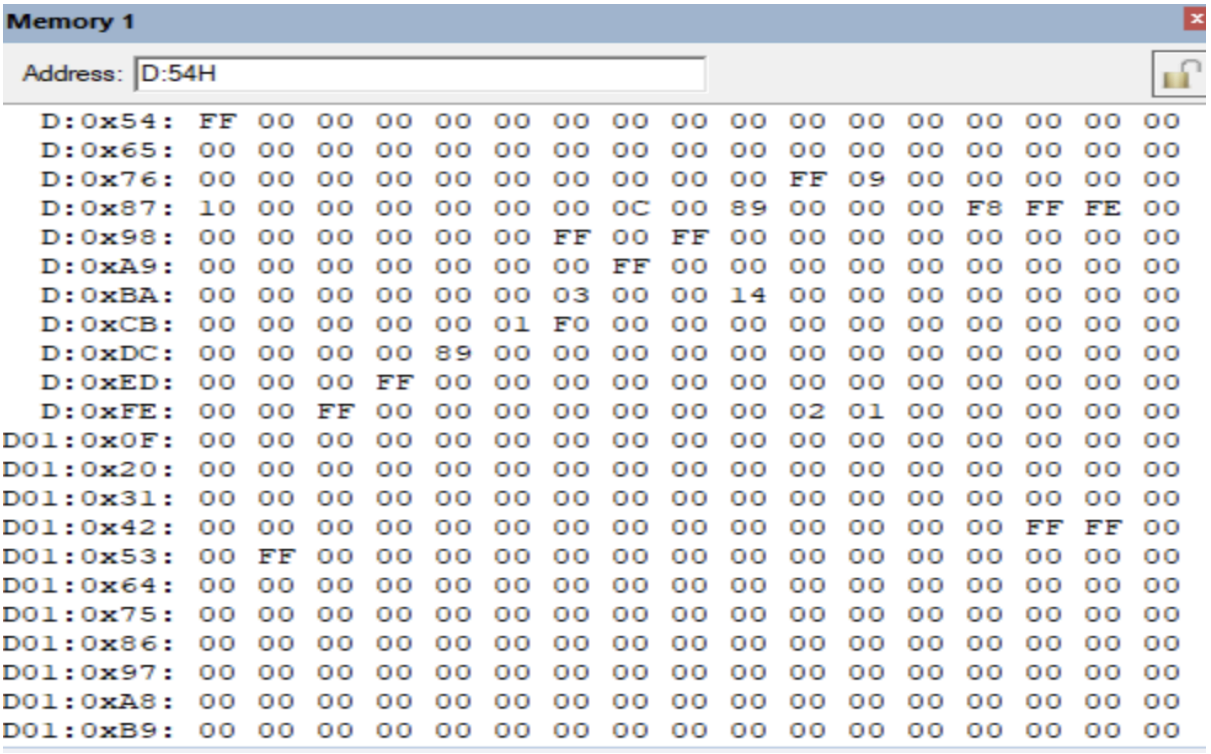
24B1312 , Supriya Anand Mishra
Microprocessors Laboratory
Electrical Eng, IITB
LAB 1

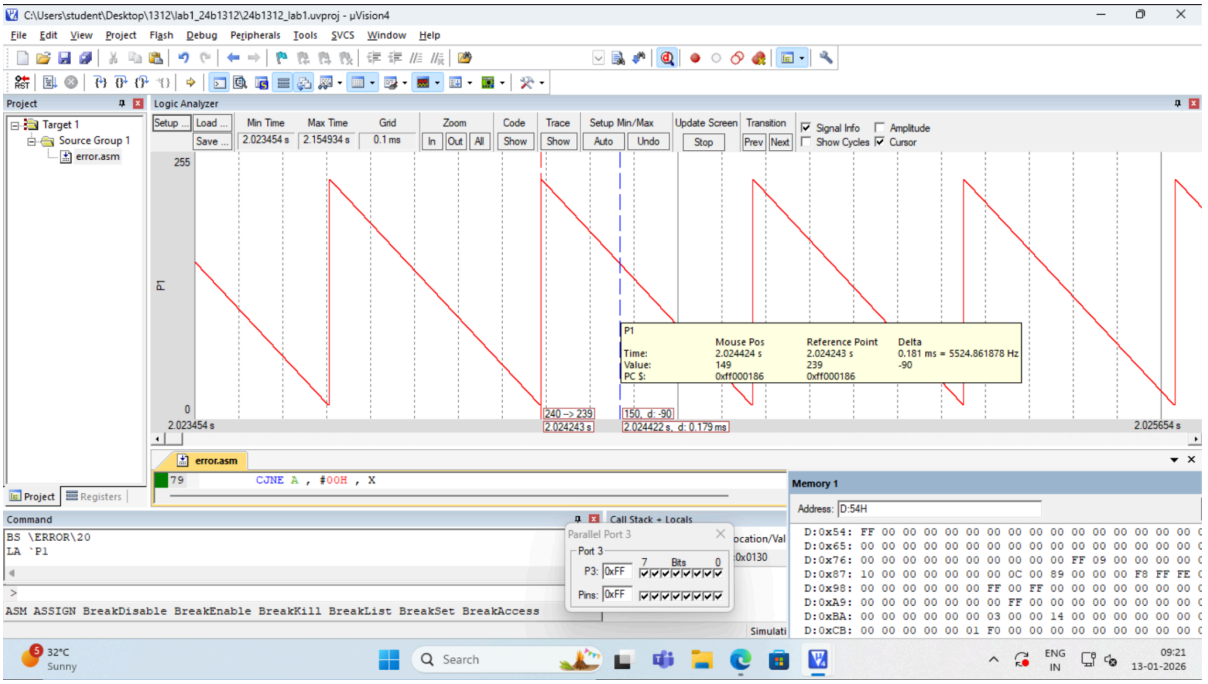


Short HW Q


Target 1
Source Group
error.asm
/* Calculator using Port 3
Operand1 -> P3
Operand2 -> P3
Opcode -> P3
Result -> 52H (low / quot
Extra -> 53H (high / ca
*/
ORG 0H
LJMP MAIN
ORG 100H
MAIN:
ACALL Cal
HERE: SJMP HERE
ORG 130H
Cal:
; -------- Read inputs
MOV A, P3 ; Oper
MOV 50H, A
MOV A, P3 ; Oper
MOV 51H, A
MOV A, P3 ; Opc
MOV 54H, A
Build Output
Program Size: data=8.0 xdata=0 code=399
creating hex file from "24b1312_lab1"...
"24b1312_lab1" - 0 Error(s), 0 Warning(s).
32°C
Sunny


C:\Users\student\Desktop\1312\lab1_24b1312\24b1312_lab1.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Project
Target 1
Source Group
error.asm
error.asm*
/* Calculator using Port 3
Operand1 -> P3
Operand2 -> P3
Opcode -> P3
Result -> 52H (low / quotient)
Extra -> 53H (high / carry / remainder)
*/
ORG 0H
LJMP MAIN
ORG 100H
MAIN:
ACALL Cal
HERE: SJMP HERE
ORG 130H
Cal:
; -------- Read inputs --------
MOV A, P3 ; Operand 1
MOV 50H, A
MOV A, P3 ; Operand 2
MOV 51H, A
MOV A, P3 ; Opcode
MOV 54H, A
Build Output
Program Size: data=8.0 xdata=0 code=399
creating hex file from "24b1312_lab1"...
"24b1312_lab1" - 0 Error(s), 0 Warning(s).
Simulation
32°C
ENG
09:23


Memory 1
Address: D:54H
D:0x54: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x65: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x76: 00 00 00 00 00 00 00 00 00 00 FF 09 00 00 00 00
D:0x87: 10 00 00 00 00 00 00 0C 00 89 00 00 00 F8 FF FE
D:0x98: 00 00 00 00 00 00 FF 00 FF 00 00 00 00 00 00 00
D:0xA9: 00 00 00 00 00 00 00 FF 00 00 00 00 00 00 00 00
D:0xBA: 00 00 00 00 00 00 03 00 00 14 00 00 00 00 00 00
D:0xCB: 00 00 00 00 00 01 F0 00 00 00 00 00 00 00 00 00
D:0xDC: 00 00 00 00 89 00 00 00 00 00 00 00 00 00 00 00
D:0xED: 00 00 00 FF 00 00 00 00 00 00 00 00 00 00 00 00
D:0xFE: 00 00 FF 00 00 00 00 00 00 00 02 01 00 00 00 00
D01:0x0F: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01:0x20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01:0x31: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01:0x42: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF FF
D01:0x53: 00 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01:0x64: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01:0x75: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01:0x86: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01:0x97: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01:0xA8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01:0xB9: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00


C:\Users\student\Desktop\1312\lab1_24b1312\24b1312_lab1.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Project
Target 1
Source Group 1
error.asm
Logic Analyzer
Setup ... Load ... Save ...
Min Time 2.023454 s
Max Time 2.154934 s
Grid 0.1 ms
Zoom In Out All
Code Show
Trace Show
Setup Min/Max Auto Undo
Update Screen Stop
Transition Prev Next
Signal Info
Amplitude
Show Cycles
Cursor
255
P1
0
P1
Time: 2.024424 s
Value: 149
PC $: 0xff000186
Mouse Pos
Reference Point 2.024243 s 239 0xff000186
Delta 0.181 ms = 5524.861878 Hz -90
240 --> 239
2.024243 s
150, d: -90
2.024422 s, d: 0.179 ms
2.023454 s
2.025654 s
error.asm
79 CJNE A , #00H , X
Project
Registers
Command
BS \ERROR\20
LA `P1
ASM ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess
Call Stack + Locals
Parallel Port 3
Port 3
P3: 0xFF
Pins: 0xFF
Bits 7 0
Memory 1
Address: D:54H
D:0x54: FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x65: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D:0x76: 00 00 00 00 00 00 00 00 00 00 FF 09 00 00 00 00
D:0x87: 10 00 00 00 00 00 00 0C 00 89 00 00 00 F8 FF FE
D:0x98: 00 00 00 00 00 00 FF 00 FF 00 00 00 00 00 00 00
D:0xA9: 00 00 00 00 00 00 00 FF 00 00 00 00 00 00 00 00
D:0xBA: 00 00 00 00 00 00 03 00 00 14 00 00 00 00 00 00
D:0xCB: 00 00 00 00 00 01 F0 00 00 00 00 00 00 00 00 00
32°C Sunny
Search
ENG IN
09:21 13-01-2026

Aim

1. To identify and correct all syntactical and logical errors in a given 8051 assembly program (Error code.asm) using Keil μVision.
2. To debug the corrected program by providing inputs through Port 3, observe how the operands and opcode are stored in internal memory, and analyze the output waveform generated on Port 1 using the Logic Analyzer.

Apparatus / Software Used

- Keil μVision IDE
- 8051 Microcontroller (Simulator)
- Logic Analyzer (Keil built-in)
- Peripheral Input Window (Port 3)

Procedure

Part 1: Error Identification and Debugging

Part 2: Debugging with Inputs and Logic Analyzer

Observations

1. Initial compilation resulted in multiple errors due to incorrect instruction usage, reserved words, and improper addressing modes.
2. After correcting the errors, the program compiled successfully without any warnings.
3. Operand 1, Operand 2, and Opcode values provided through **Port 3** were correctly read by the program.
4. The values from Port 3 were stored sequentially in the designated internal memory locations.
5. The program executed the operation based on the opcode provided.
6. A periodic waveform was observed on **Port 1**.
7. The frequency of the waveform on Port 1 was successfully measured using the Logic Analyzer.

Result

- The given erroneous assembly program was successfully debugged and executed.

- Inputs provided through Port 3 were correctly processed and stored in internal memory.
- The output waveform was generated on Port 1 and its frequency was successfully determined using the Logic Analyzer.

Conclusion

This experiment helped in understanding the importance of correct syntax, addressing modes, and instruction usage in 8051 assembly programming. It also demonstrated how Keil's debugging tools such as breakpoints, peripheral input windows, and logic analyzer can be effectively used to trace program execution, provide real-time inputs, and analyze output signals.

What I Learned

- How to identify and correct syntax and logical errors in 8051 assembly programs.
- The importance of proper label placement and spacing in assembly language.
- How to use breakpoints for controlled execution during debugging.
- How to provide inputs to a program through I/O ports using Keil's peripheral windows.
- How to observe internal memory changes during program execution.
- How to use the Logic Analyzer to analyze digital waveforms and measure frequency.
- The practical workflow of building, debugging, and testing embedded programs.