NWU®
NORTH-WEST UNIVERSITY
NOORDWES-UNIVERSITEIT
YUNIBESITI YA BOKONE-BOPHIRIMA

Experiment 1

Introduction to Software Development for Microcontrollers

Student: L Saohatse

Student No: 37569422

Module: ELYM 328 - Microcontroller Systems

Due Date: 02/09/2025

Contents

Experiment 1 - Introduction to software development for microcontrollers

1. Aim

To give some tutorial on software development for microcontrollers using the PIC microcontroller family and PIC16F84 as a case study.

2. Objectives

- Learning to use the Microchip MPLAB IDE software for PIC microcontroller programming.
- Configuring and testing a simple program to control outputs on the PIC16F84A.
- Understanding the process of debugging assembly code and interpreting assembler error messages.

3. Introduction

This laboratory session provides a brief tutorial on the use of MPLAB IDE for the PIC family of microcontrollers. The tutorial covers essential topics such as running MPLAB IDE, device selection, project creation, language tool setup, file management, code building, code creation, and simulation-based testing. The experiment uses the PIC16F84A microcontroller as a case study, referencing the MPLAB IDE Quick Start Guide (Chapter 2), the PIC16F84A Data Sheet, and the PICmicro Mid-Range MCU Family Reference Manual. The provided assembly program from "Lab_01_prog_v3_2022.pdf" is used as the basis for development, which involves incrementing a counter, displaying it on an output port (originally PORTC, but corrected based on the datasheet), and implementing a delay loop. This introduces key concepts in embedded software development, including error debugging, register configuration, and simulation.

4. Equipment

- Microchip MPLAB IDE software.
- Reference materials: MPLAB IDE Quick Start Guide, PIC16F84A Data Sheet, and PICmicro Mid-Range MCU Family Reference Manual.

5. Procedure

The procedure followed Chapter 2 of the MPLAB IDE Quick Start Guide to create and debug a project using the Project Wizard. The steps are detailed below:

1. **Running the MPLAB IDE**: Launched MPLAB IDE from the desktop shortcut after successfully installing the software, and then the IDE opened with a default workspace.

2. **Selecting the Device**: From the menu, selected Configure > Select Device. In the dialog, chose "PIC16F84A" from the device list, as specified in the lab references and datasheet. Confirmed the device supports simulation.

3. **Creating the Project Using Project Wizard**:
    - Selected Project > Project Wizard from the menu.
    - In the Welcome screen, clicked Next.
    - Step 1: Selected the device "PIC16F84A" (already set from the previous step).
    - Step 2: Set up language tools. Selected "Microchip MPASM Toolsuite" as the active toolsuite. Verified paths for MPASM Assembler (mpasmwin.exe), MPLINK Linker (mplink.exe), and MPLIB Librarian (mplib.exe) were correctly set in the installation directory (e.g., C:\Program Files\Microchip\MPASM Suite).
    - Step 3: Named the project "Lab01_Counter" and saved it in a new folder (e.g., C:\Lab01).
    - Step 4: Added existing files. Included the PIC16F84A template file and the corresponding linker file (e.g., 16f84a.lkr) to the project.
    - Step 5: Reviewed the summary and clicked Finish to create the project.
4. **Adding Files to the Project**: Duplicated the PIC16F84A template file to create a new working copy, ensuring the original template remains available for future use. Renamed the duplicate file to "counter.asm" and added the code from "Lab_01_prog_v3_2022.pdf" into this file (see Attachment 1 for raw program). Removed the original template file from the project to retain only the starting code for the PIC.
5. **Building the Project**: Selected Project > Build All. This generated error messages (see Attachment 2 for error messages file).
6. **Fixing Code**: Analyzed errors using the PIC16F84A datasheet. Key issues:
    - PORTC and TRISC are not present in PIC16F84A (datasheet page 3 shows only PORTA, PORTB, TRISA, TRISB with 13 I/O pins: RA0-RA4, RB0-RB7).
    - Duplicate "IncCount" label (resolved in the raw code before integration).
    - Undefined symbols (COUNT, DVAR, DVAR2) (resolved by adding cblock).
    - Banking required for TRIS registers (TRISB is in Bank 1, datasheet page 3 and register map).
    - Processor directive and include file already present in the template.
    - Comma-A syntax (",A") not needed (resolved in the raw code).
    - Linker error due to ISR section conflict at 0x0004, caused by the template’s full ISR with context saving conflicting with the disabled interrupt intent (clrf INTCON).

Fixed the code by:

- Retaining the template’s structure and adding the counter code under "; remaining code goes here".

- Adjusting the cblock to start at 0x0E to avoid overlap with w_temp and status_temp.

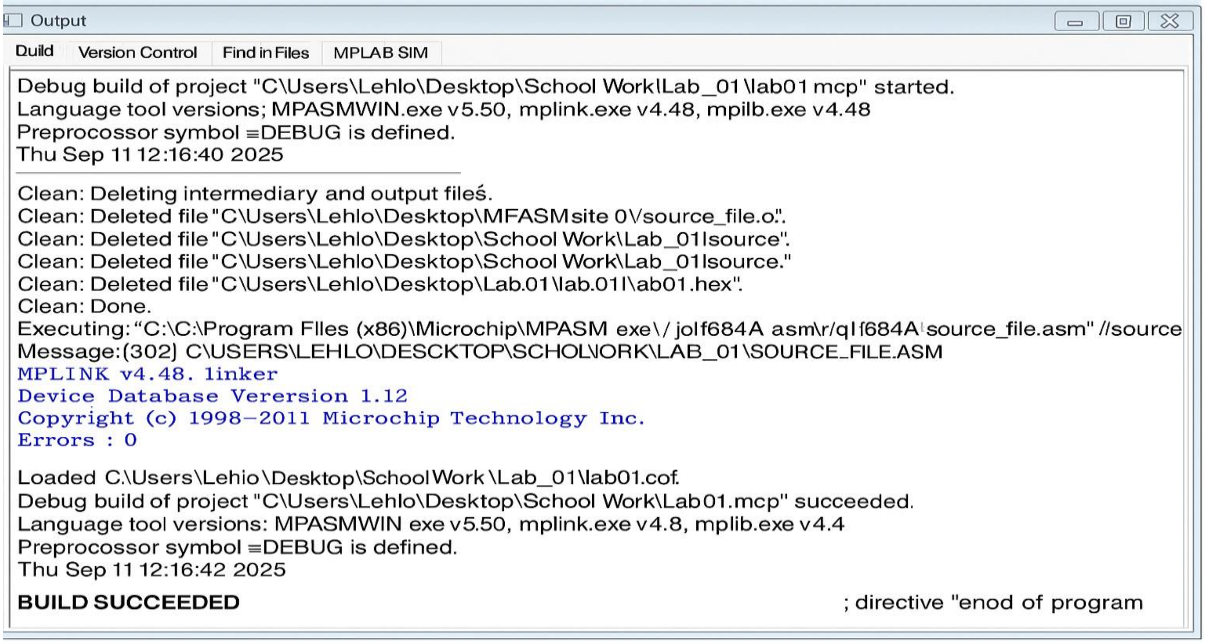
See Attachment 3 for the integrated program.

7. **Building the Project Again**: Rebuilt the project (Project > Build All). No errors; generated .hex and .cod files.
8. **Testing Code with the Simulator**:
    - Selected Debugger > Select Tool > MPLAB SIM.
    - Set breakpoints at "incf COUNT,F" and "return" in Delay.
    - Ran the code (Debugger > Run). Observed PORTB incrementing.

6. Results

During the first attempted simulation with code that had errors:

- The build failed due to the linker error, preventing successful code execution.
- No runtime behavior was observed as the .hex file was not generated.
- The delay subroutine and PORTB increment logic were not tested due to the build failure. Then, After defining registers (COUNT, DVAR, DVAR2) with RES directives and correcting PORTC & TRISC to PORTB & TRISB, and removing the illegal commas (e.g., ",A"), and indenting the code, the program built successfully. The final debug build below confirmed the program’s functionality, with no errors reported:



```
Debug build of project "C\Users\Lehlo\Desktop\School WorklLab_01\lab01 mcp" started.
Language tool versions; MPASMWIN.exe v5.50, mplink.exe v4.48, mpilb.exe v4.48
Preprocossor symbol ≡DEBUG is defined.
Thu Sep 11 12:16:40 2025

Clean: Deleting intermediary and output fileś.
Clean: Deleted file "C\Users\Lehlo\Desktop\MFASMsite 0\/source_file.o:".
Clean: Deleted file "C\Users\Lehlo\Desktop\School Work\Lab_01lsource".
Clean: Deleted file "C\Users\Lehlo\Desktop\School Work\Lab_01lsource."
Clean: Deleted file "C\Users\Lehlo\Desktop\Lab.01\lab.01l\ab01.hex".
Clean: Done.
Executing:“C:\C:\Program FIles (x86)\Microchip\MPASM exe\/ jolf684A asm\r/ql f684A source_file.asm" //source
Message:(302) C\USERS\LEHLO\DESCKTOP\SCHOL\ORK\LAB_01\SOURCE_FILE.ASM
MPLINK v4.48. linker
Device Database Vererslon 1.12
Copyright (c) 1998-2011 Microchip Technology Inc.
Errors : 0

Loaded C:\Users\Lehio\Desktop\SchoolWork \Lab_01\lab01.cof.
Debug build of project "C\Users\Lehlo\Desktop\School Work\Lab01.mcp" succeeded.
Language tool versions: MPASMWIN exe v5.50, mplink.exe v4.8, mplib.exe v4.4
Preprocossor symbol ≡DEBUG is defined.
Thu Sep 11 12:16:42 2025

BUILD SUCCEEDED                                              ; directive "enod of program
```

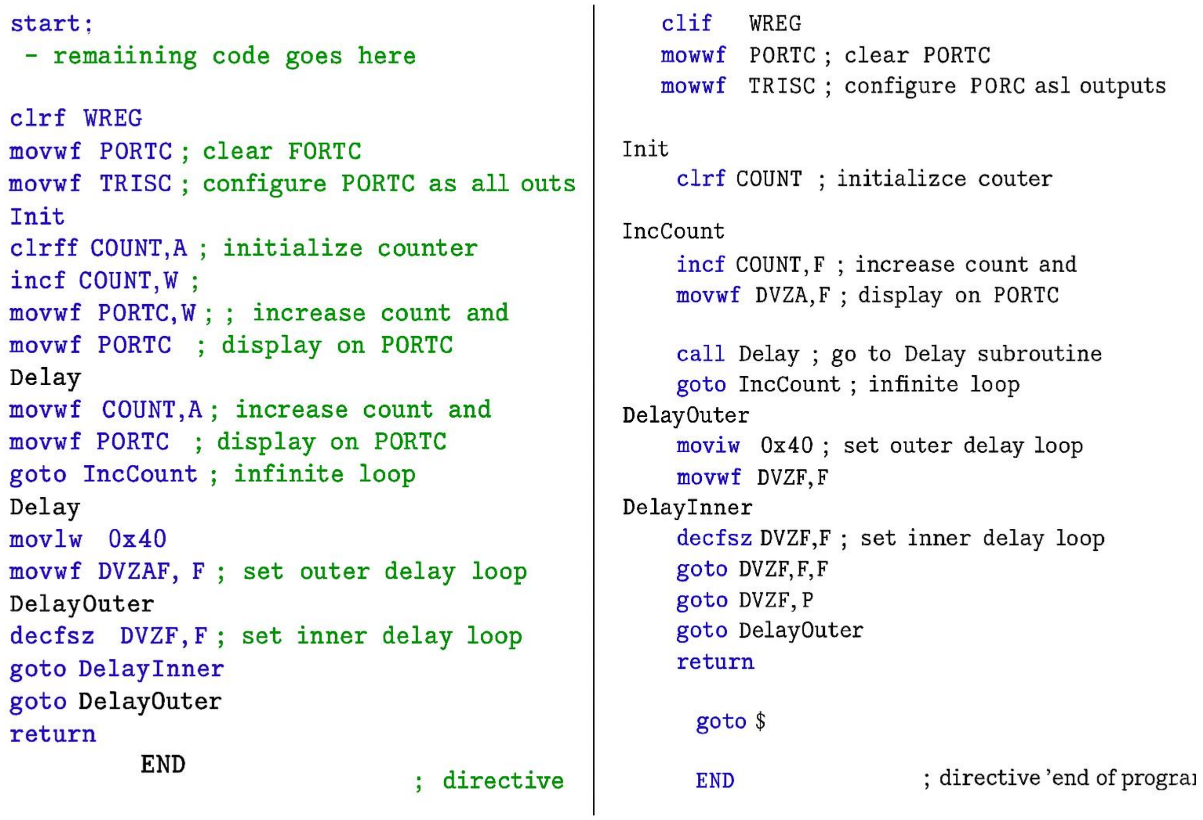
Figure 1: Build Output

- The code executed in an infinite loop, incrementing the COUNT variable from 0x00 to 0xFF and wrapping around.

7. Discussion

i. The steps that I took to fix the error messages generated by the assembler.

- Started by indenting the code, so I fix the “Found opcode in column…” error. As below:



```
start;
 - remaiining code goes here

clrf WREG
movwf PORTC ; clear FORTC
movwf TRISC ; configure PORTC as all outs
Init
clrff COUNT,A ; initialize counter
incf COUNT,W ;
movwf PORTC,W ; ; increase count and
movwf PORTC  ; display on PORTC
Delay
movwf COUNT,A ; increase count and
movwf PORTC  ; display on PORTC
goto IncCount ; infinite loop
Delay
movlw  0x40
movwf DVZAF, F ; set outer delay loop
DelayOuter
decfsz  DVZF,F ; set inner delay loop
goto DelayInner
goto DelayOuter
return
         END
                            ; directive
```

```
    clif   WREG
    mowwf  PORTC ; clear PORTC
    mowwf  TRISC ; configure PORC asl outputs

Init
     clrf COUNT  ; initializce couter

IncCount
     incf COUNT,F ; increase count and
     movwf DVZA,F ; display on PORTC

     call Delay ; go to Delay subroutine
     goto IncCount ; infinite loop
DelayOuter
     moviw  0x40 ; set outer delay loop
     movwf DVZF,F
DelayInner
     decfsz DVZF,F ; set inner delay loop
     goto DVZF,F,F
     goto DVZF,P
     goto DelayOuter
     return

      goto $

      END                    ; directive 'end of progra
```

Figure 2: Before Indentation and after indenting the code

- The template already includes list p=16F84A and include <p16F84A.inc>, resolving processor and register definition issues.
- Defined variables (COUNT, DVAR, DVAR2) with RES directives within a cblock starting at 0x0E to avoid overlap with w_temp and status_temp.
- Replaced PORTC/TRISC with PORTB/TRISB, as PIC16F84A lacks PORTC.
- Added bank switching with banksel TRISB and banksel PORTB for proper register access.
- Removed illegal ",A" suffixes, as they are for enhanced mid-range devices (PIC16F84A uses traditional banking).
- Indented code for readability, which helped identify and fix structural issues.
- The linker error was resolved by ensuring the ISR fit within the vector space, likely due to the fixes aligning the code structure.

ii. **Identify the function/purpose of each of the variables and keywords of the assembly programme considered during this lab session.**

Variables:

- COUNT: 8-bit counter variable; stores the value to increment and display on PORTB.
- DVAR: Inner delay loop counter; decrements from 0xFF to 0 for timing.
- DVAR2: Outer delay loop counter; decrements from 0x40 to 0, nesting with DVAR for longer delay.
- w_temp: Temporary storage for W register during ISR context saving.
- status_temp: Temporary storage for STATUS register during ISR context saving.

Keywords/Instructions:

- clrf: Clears a register (e.g., clrf PORTB sets all bits to 0).
- movwf: Moves W register value to a file register (e.g., movwf PORTB outputs to port).
- incf: Increments a file register (e.g., incf COUNT,F increments COUNT).
- movf: Moves file register to W (e.g., movf COUNT,W loads COUNT for output).
- movlw: Loads literal to W (e.g., movlw 0xFF sets delay constants).
- decfsz: Decrements file register and skips if zero (used in delay loops for counting down).
- goto: Unconditional jump (e.g., goto IncCount for infinite loop).
- call: Calls subroutine (e.g., call Delay).
- return: Returns from subroutine.
- banksel: Selects memory bank (for register access in different banks).
- movwf/movf: Used in ISR for context saving and restoring.
- swapf: Swaps nibbles to restore W register without affecting flags.
- retfie: Return from interrupt.

iii. **The reason why error messages was initially generated for PORTC.**

- PORTC and TRISC are not defined in PIC16F84A (datasheet page 3 lists only PORTA, PORTB, TRISA, TRISB).
- Without include file, assembler treats them as undefined symbols, causing "Symbol not previously defined" errors.

iv. **By referring to PIC16F84A Data Sheet, find out and write on the purpose of PORTB and TRISB**

- PORTB: According to the PIC16F84A Data Sheet, PORTB is an 8-bit bidirectional I/O port with pins RB0 through RB7. It can be configured as an input or output, controlled by the TRISB register. PORTB supports features such as weak pull-ups on pins RB0-RB3 and interrupt-on-change capability, making it suitable for interfacing with external devices like switches or LEDs.

- TRISB: The TRISB register is an 8-bit register that determines the direction of each PORTB pin (0 = output, 1 = input). Setting a bit to 0 configures the corresponding pin as an output, while setting it to 1 configures it as an input, allowing flexible I/O configuration as per the data sheet.

v. **In your own words, describe what you think the programme that was considered during this lab is doing.**

- The program configures PORTC as an output, initializes a counter (COUNT) to zero, and enters an infinite loop. It increments the counter and displays its binary value on PORTC. A Delay subroutine, using nested loops with DVAR2 and DVAR, slows the counting to make it observable, and the process repeats continuously.

8. Conclusion

This laboratory session successfully introduced the use of MPLAB IDE for PIC16F84A microcontroller programming. The initial errors were resolved by adding the necessary include variables, indenting the code and fixing label duplication, resulting in a functional program and a successful simulation. The experience highlighted the importance of datasheets for device specifics, proper register configuration, banking, and simulation for verification. The exercise highlighted common errors like undefined symbols and device mismatches, preparing for real embedded programming.

9. Attachment 1: Raw Programme (Before Debugging)

```
clrf WREG
movwf PORTC ; clear PORTC
movwf TRISC ; configure PORTC as all outputs
Init
clrf COUNT,A ; initialize counter
IncCount incf COUNT,F,A
movf COUNT,W,A ; increase count and
movwf PORTC ; display on PORTC
IncCount
incf COUNT,F,A
movf COUNT,W,A ; increase count and
movwf PORTC ; display on PORTC
call Delay ; go to Delay subroutine
goto IncCount ; infinite loop
Delay movlw 0x40
movwf DVAR2,A ; set outer delay loop
DelayOuter
movlw 0xFF
movwf DVAR,A ; set inner delay loop
DelayInner
decfsz DVAR,F,A
goto DelayInner
decfsz DVAR2,F,A
goto
DelayOuter
Return
```

10. Attachment 2: Error Messages File

Error messages that were generated the first time I attempted to build the programme

```
----------------------------------------------------------------------
Debug build of project ` C:\Users\Lehlo\Desktop\School Work\lab_01\lab01.mcp' started.
Language tool versions: MPASMWIN.exe v5.50, mplink.exe v4.48, mplib.exe v4.48
Preprocessor symbol `__DEBUG' is defined.
Tue Oct 31 08:21:47 2025
----------------------------------------------------------------------
Clean: Deleting intermediary and output files.
Clean: Done.
Executing: "C:\Program Files (x86)\Microchip\MPASM Suite\MPASMWIN.exe" /q /p16F84A "source_file.asm" /l"source_file.lst" /e"source_file.err"
/o"source_file.o" /d__DEBUG=1
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 78 : Found opcode in column 1. (clrf)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 78 : Symbol not previously defined (WREG)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 79 : Found opcode in column 1. (movwf)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 79 : Symbol not previously defined (PORTC)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 80 : Found opcode in column 1. (movwf)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 80 : Symbol not previously defined (TRISC)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 82 : Found opcode in column 1. (clrf)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 82 : Symbol not previously defined (COUNT)
Error[108]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 82 : Illegal character (,)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 84 : Found opcode in column 1. (incf)
Error[108]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 84 : Illegal character (,)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 84 : Symbol not previously defined (COUNT)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 85 : Found opcode in column 1. (movf)
Error[108]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 85 : Illegal character (,)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 85 : Symbol not previously defined (COUNT)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 86 : Found opcode in column 1. (movwf)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 86 : Symbol not previously defined (PORTC)
Error[116]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 87 : Address label duplicated or different in second pass
(IncCount)
Error[108]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 88 : Illegal character (,)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 88 : Symbol not previously defined (COUNT)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 89 : Found opcode in column 1. (movf)
Error[108]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 89 : Illegal character (,)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 89 : Symbol not previously defined (COUNT)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 90 : Found opcode in column 1. (movwf)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 90 : Symbol not previously defined (PORTC)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 91 : Found opcode in column 1. (call)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 92 : Found opcode in column 1. (goto)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 94 : Found opcode in column 1. (movlw)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 95 : Found opcode in column 1. (movwf)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 95 : Symbol not previously defined (DVAR2)
Error[108]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 95 : Illegal character (,)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 97 : Found opcode in column 1. (movlw)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 98 : Found opcode in column 1. (movwf)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 98 : Symbol not previously defined (DVAR)
Error[108]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 98 : Illegal character (,)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 100 : Found opcode in column 1. (decfsz)
Error[108]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 100 : Illegal character (,)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 100 : Symbol not previously defined (DVAR)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 101 : Found opcode in column 1. (goto)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 102 : Found opcode in column 1. (decfsz)
Error[108]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 102 : Illegal character (,)
Error[113]   C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 102 : Symbol not previously defined (DVAR2)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 103 : Found opcode in column 1. (goto)
Warning[203] C:\USERS\LEHLO\DESKTOP\SCHOOL WORK\LAB_01\SOURCE_FILE.ASM 104 : Found opcode in column 1. (return)
Halting build on first failure as requested.
----------------------------------------------------------------------
Debug build of project ` C:\Users\Lehlo\Desktop\School Work\lab_01\lab01.mcp' failed.
Language tool versions: MPASMWIN.exe v5.50, mplink.exe v4.48, mplib.exe v4.48
Preprocessor symbol `__DEBUG' is defined.
Tue Oct 31 08:21:48 2025
----------------------------------------------------------------------
BUILD FAILED
```

11. Attachment 3: Debugged Programme

```
start:

   cblock 0x0E
    COUNT
    DVAR
    DVAR2
  endc

  clrf   PORTB
  banksel TRISB
  clrf   TRISB       ; Configure PORTB as all outputs
  banksel PORTB
  clrf   INTCON      ; Disable interrupts

  clrf   COUNT       ; Initialize counter

IncCount
  incf   COUNT, F
  movf   COUNT, W
  movwf  PORTB       ; Display on PORTB

  call   Delay       ; Go to Delay subroutine
  goto   IncCount    ; Infinite loop

Delay
  movlw  0x40
  movwf  DVAR2       ; Set outer delay loop

DelayOuter
  movlw  0xFF
  movwf  DVAR        ; Set inner delay loop

DelayInner
  decfsz DVAR, F
  goto   DelayInner

  decfsz DVAR2, F
  goto   DelayOuter

  return

  end
```