

C and PIC24 Assembly Language Programming
5th Laboratory Report for ECE 383
Microcomputers

Submitted by
Patrick Brooks
Lauren Brackin
11650957

The University of Alabama
Tuscaloosa, AL 35487

3/5/21

Abstract:

The main objectives of this lab are to introduce students to students to basic C language programming and equivalent PIC24 assembly language. They will get a glimpse into the ecosystem. Students will carry these skills with them throughout their careers. Task one starts by having students build a C program and record flags. Task two instructs students to create an “&” bitwise program. Task three is where students will translate C program from task 2 to PIC24 assembly. Task four requires students to translate PIC24 assembly code to C language.

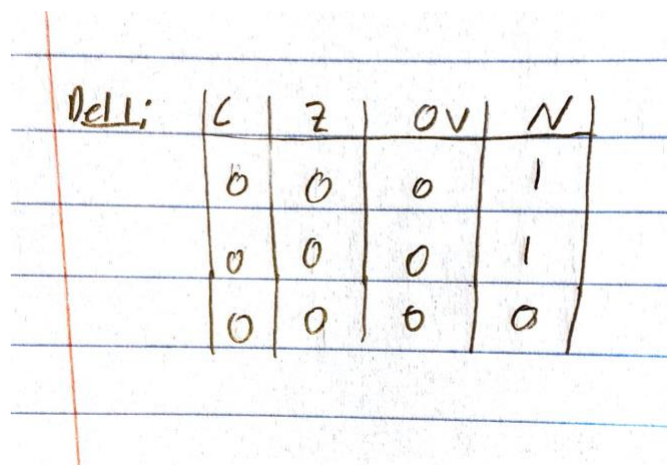
Introduction:

The objective of the lab is to introduce students to MPLAB and coding with PIC24 assembly. Task one starts by having students build a C program and record flags. In task two instructs students to create an “&” bitwise program. Task three is where students will translate C program from task 2 to PIC24 assembly. Task four requires students to translate PIC24 assembly code to C language.

Procedure/Results:

Task 1: Basic Arithmetic Operations

- a. Use Project->Project Wizard for the creation of an MPLAB project.
 - i. Use Configure → Select Device → PIC24HJ128GP502
 - ii. Active Tool suite → Microchip C30 Toolsuite
 - iii. New Project File → Destination
 - iv. Skip the addition of existing files to the project
 - v. View → Program Mem, Register Files, Special Function Registers
 - vi. Debugger → Select Tool → MPLAB Sim



Handwritten table showing register values for C, Z, OV, and N flags.

Reg:	C	Z	OV	N
	0	0	0	1
	0	0	0	1
	0	0	0	0

Figure 1. Deliverable 1

Deliverable 2: Type casting is the act of changing the variable type in order for the compiler to perform arithmetic operations with other variables. It is used when u16_d is made as an assignment. Typecasting is useful because it allows for arithmetic operations to happen with different sized variables.

Figure 2. Deliverable 2

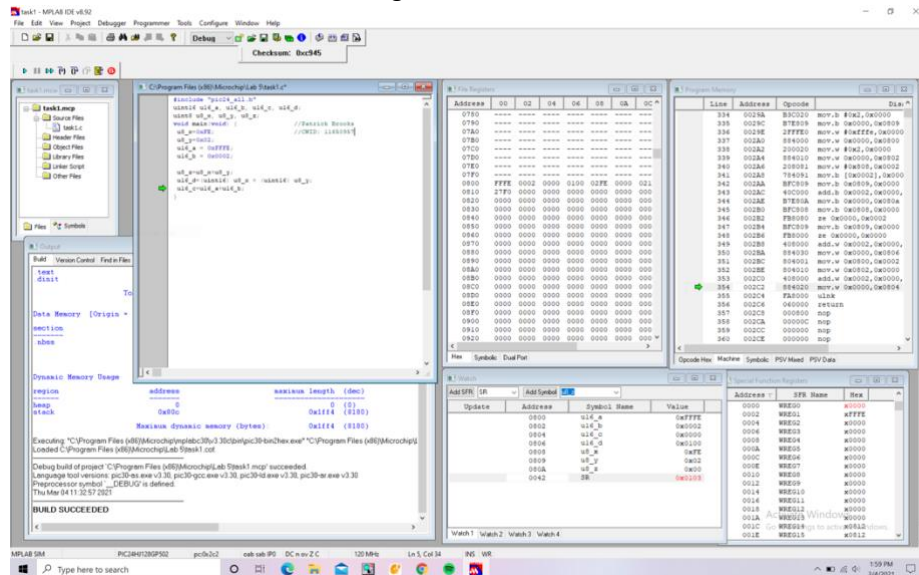


Figure 3. Deliverable 3

Task 2: C Program check_val

a. Create “task2.mcp” project

i. Create your own C program to do the tasks mapped out in the lab instructions

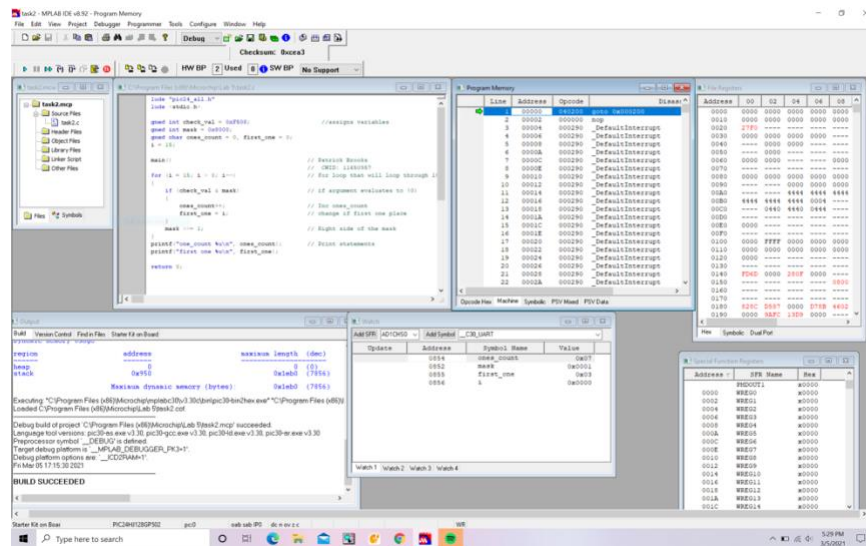


Figure 4. Deliverable 4

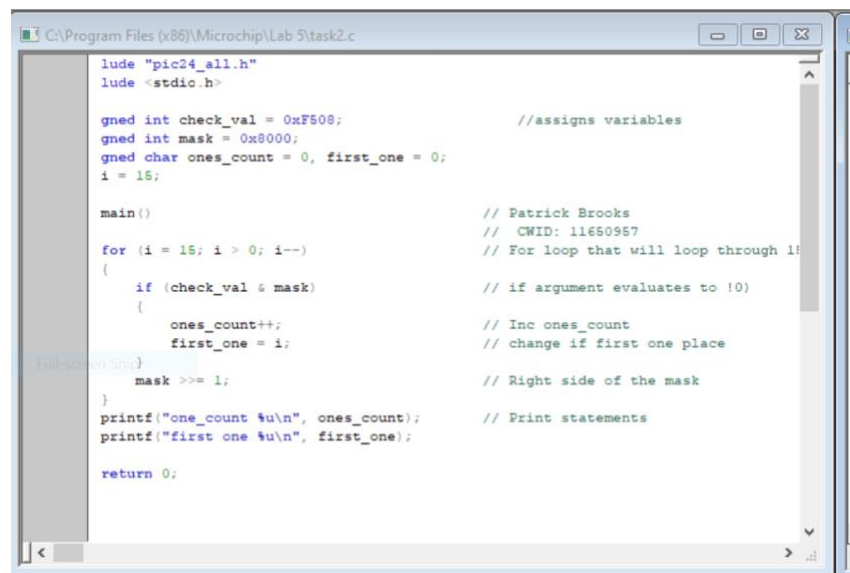


Figure 5. Deliverable 5

https://www.youtube.com/playlist?list=PLLXkv2lvJPoMffuA0Y12svi_HbA_r_3uF

Figure 6. Deliverable 6

Task 3: Assembly Language Program check_val

a. Create “task3.mcp” project

i. Translate C program from task 2 to an assembly program

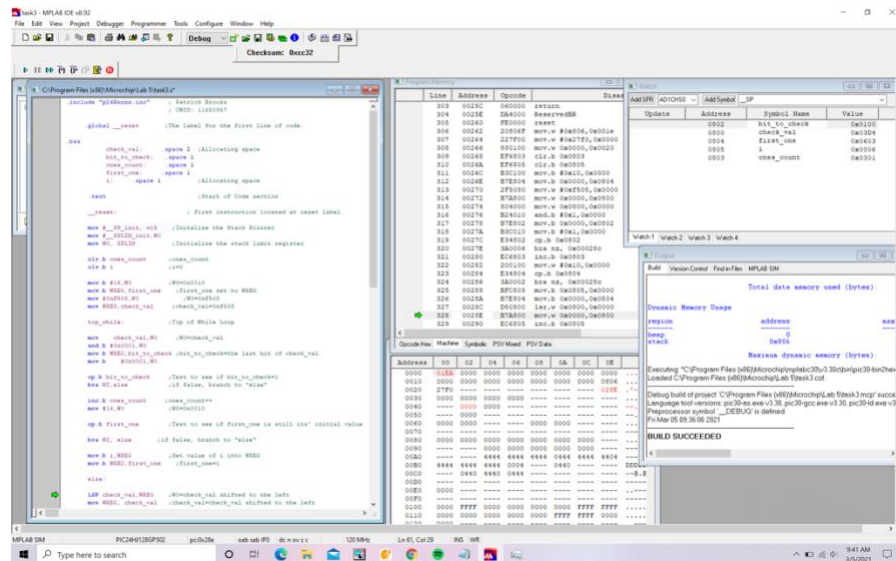


Figure 7. Deliverable 7

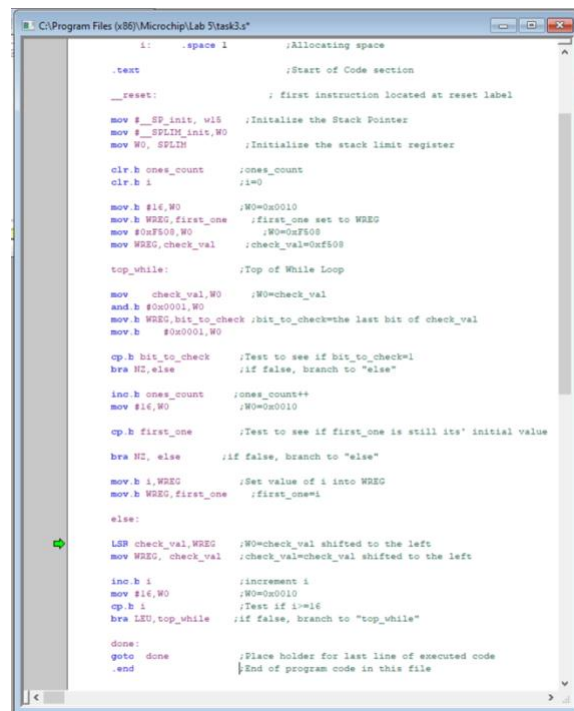


Figure 8. Deliverable 8

https://www.youtube.com/playlist?list=PLLXkv2lvJPoMffuA0Y12svi_HbA_r_3uF

Figure 9. Deliverable 9

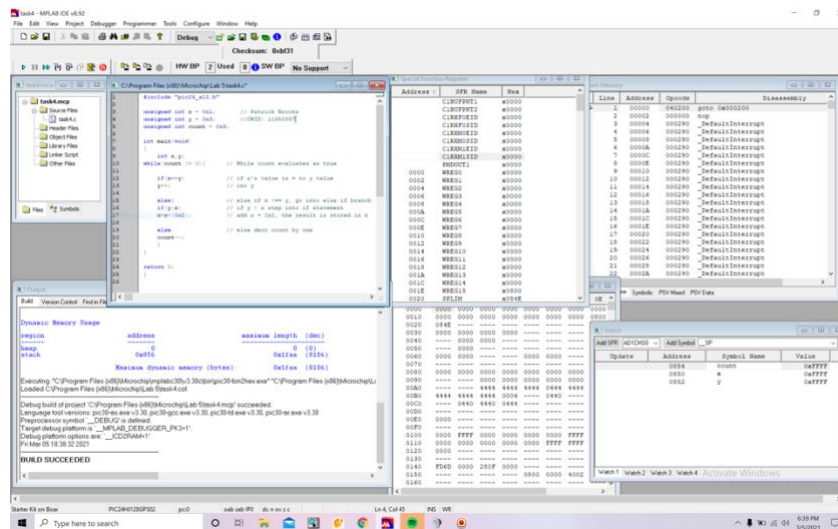


Figure 10. Deliverable 10

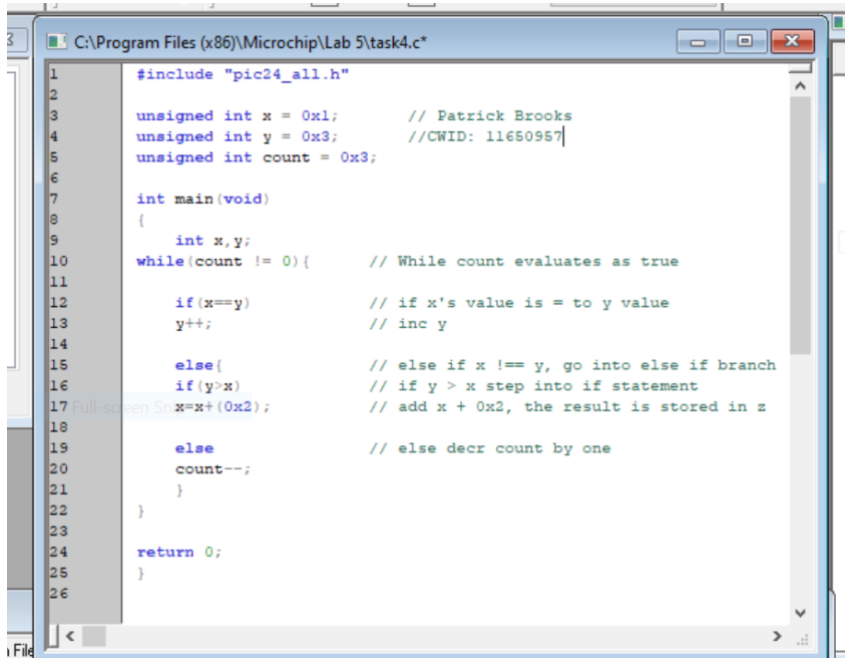


Figure 11. Deliverable 11

https://www.youtube.com/playlist?list=PLlXkv2lvJPoMffuA0Y12svi_HbA_r_3uF

Figure 12. Deliverable 12

Conclusion:

After completion of the lab students learned how to use MPLAB to create programs with C and translate them to assembly and vice versa. As well as revisiting useful addition and subtraction concepts, branch statements, and bitwise operators. Students also deepened their understanding of the lab report format which will be used throughout their academic endeavors.