Practical 01

Assembly Language

THIS IS A PROCTORED PRACTICAL

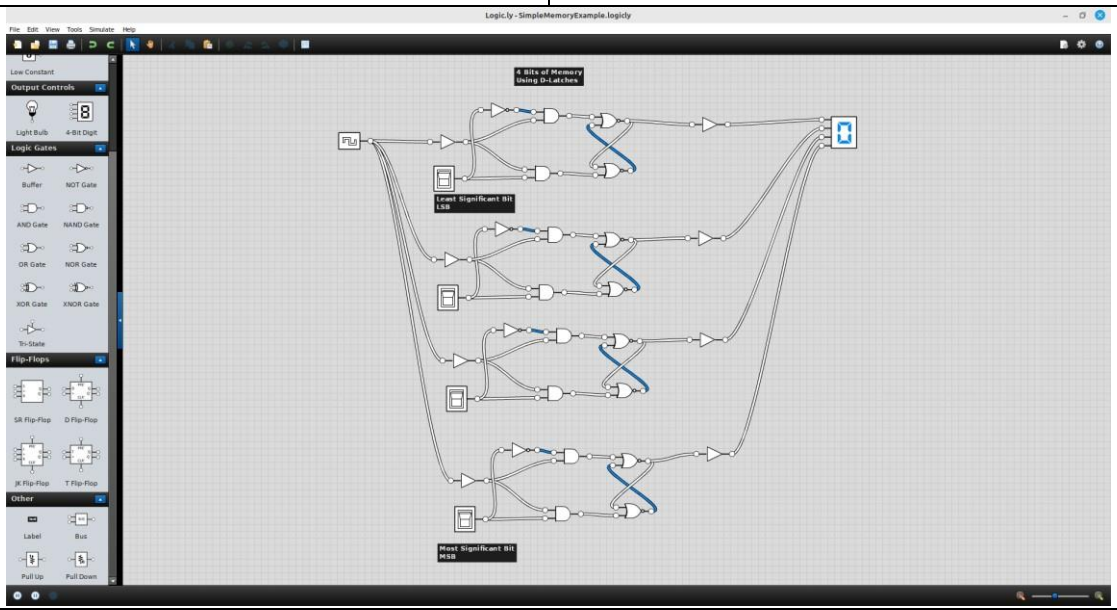
YOU MUST SHARE YOUR SCREEN SO YOUR PARTICIPATION IN THIS PRACTICAL CAN FULLY INVIGILATED

1. Create a Github repository "Assembly_and_C"
2. Create a sub directory PRACTICAL_01
3. Add Github link to CA Spreadsheet
   e.g https://STUDENTID.github.com/Assembly_and_c/PRACTICAL_01
4. Invite Lab Supervisors including MakeMuddyGames as a collaborators
5. Go to designated group to complete practical
6. Upload completed Practical files to Github repository

NOTE: Use of EASy68K editor and emulator allowed, use of internet allowed, use of slide deck(s) allowed. Installer located here http://www.easy68k.com/

Create a unique file ***e.g. part1.X68*** for each practical section below.

Objective Understand and utilise Basic Memory concepts, BINARY, HEX and Literals**:**

1	Create a new 68K project and name the file ***part1.logicly*** Create the following circuit using logic.ly	Store the following Decimal Values as Binary (Bits LSB to MSB) 0 to 15
		

2	Create a new 68K project and name the file ***part2.X68***
	
	Edit compile and execute the code across, examine and note contents of data registers and memory. Identify the memory location of `$3000` and its contents.
		`MOVE.L #%00001111,D1`
		`MOVE.B D1,D2`
		`MOVE.B D1,$2000`
		`MOVE.B $2000,D2`
		`MOVE.B $2000,$3000`

3	Create a new 68K project and name the file ***part3.X68***	`ORG $1000`
		`START:`
		`MOVE.B #$64,D1`
		`LEA text, A1`

Practical 01

Assembly Language

	Edit compile and execute the code across and observe the output.	MOVE #14,D0
		TRAP #15
		
		MOVE #3,D0
		TRAP #15
		
		SIMHALT
		
		text dc.b 'Data Register: ',0
		
		END START

4	Create a new 68K project and name the file ***part4.X68*** Edit compile and execute the code across and observe the input and output.	

```
      ORG    $1000
START:
    LEA text, A1
    MOVE #4,D0
    TRAP #15
    MOVE #14,D0
    TRAP #15
    MOVE #3,D0
    TRAP #15

SIMHALT
text  dc.b    'Data Register: ',0

          END    START
```

Demonstrate completed assembly files at the end of the LAB and ensure it has been checked

Student Name	**Brandon Jaroszczak**	**Student Number**	**C00296052**
Date	**13/1/2025**	**Checked**	