MULTIPROGRAMMABLE ALU

NAME: Lakshmi Sai Bhargav Vemparala

ROLL NO.: 2025102061

TABLE NO.: 13

GROUP: G17

EXPERIMENT 5: Programmable 1-bit ALU

Aim:

To design, assemble, and test a programmable 1-bit Arithmetic and Logic Unit (ALU) capable of performing 8 different arithmetic and logic functions using multiplexers and XOR gates, and to verify its functionality against the theoretical function table.

Components Required:

1. Digital Test Kit / Breadboard
2. Connecting Wires
3. IC 74LS151 (8:1 Multiplexer)
4. IC 74LS157 (Quad 2:1 Multiplexer)
5. IC 74LS86 (Quad 2-input XOR gate)
6. LEDs for output visualization

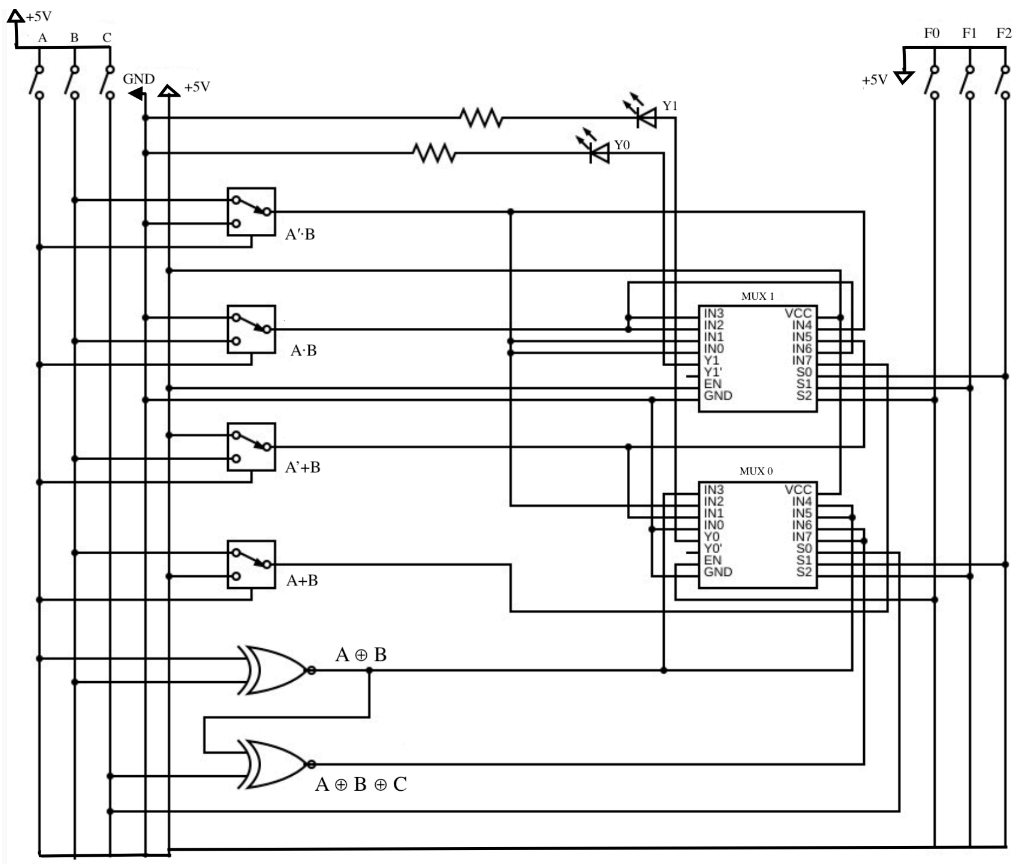
Circuit Implementation:

The ALU is designed to perform 8 functions on 1-bit inputs A, B, and C, controlled by the function select inputs F2, F1, and F0.

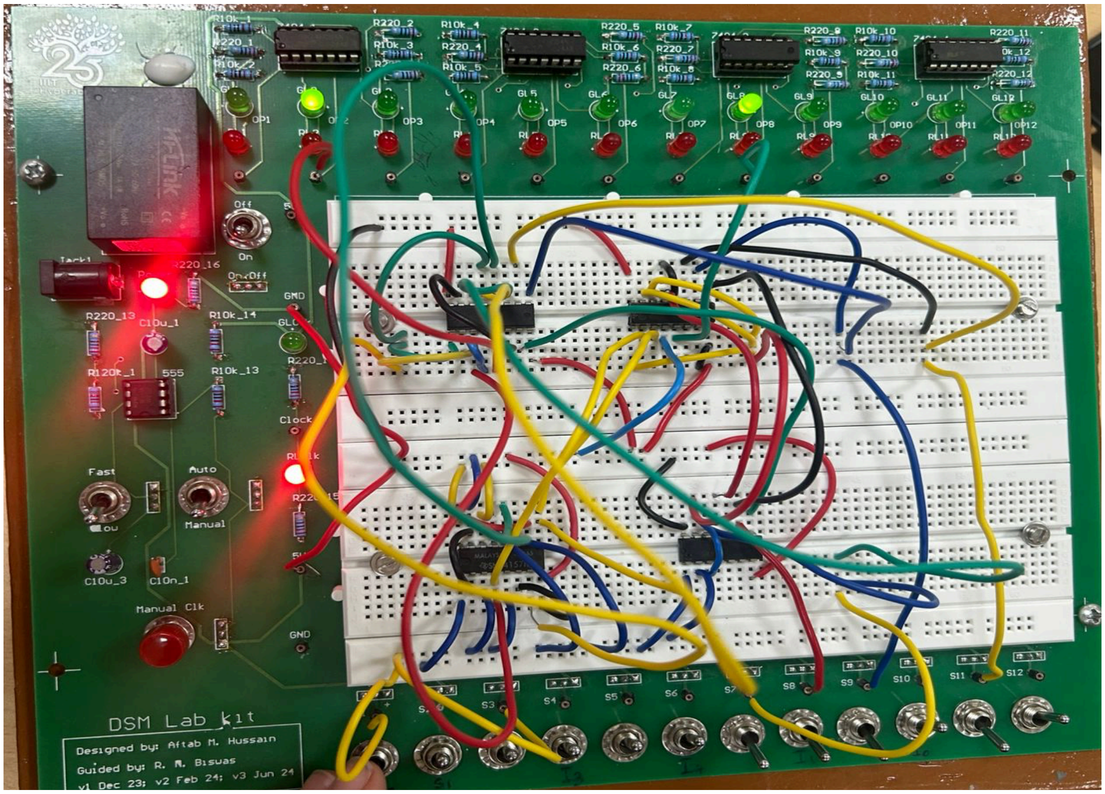
- **MUX0** generates the output **Y0** (logical or arithmetic result).
- **MUX1** generates the output **Y1** (carry/borrow output), which is enabled only for arithmetic functions when F2 = 1.
- Boolean functions such as $A \cdot B$, $A' \cdot B$, $A + B$, $A' + B$, and $A \oplus B$ are realized using the quad 2-input multiplexer and XOR gate.

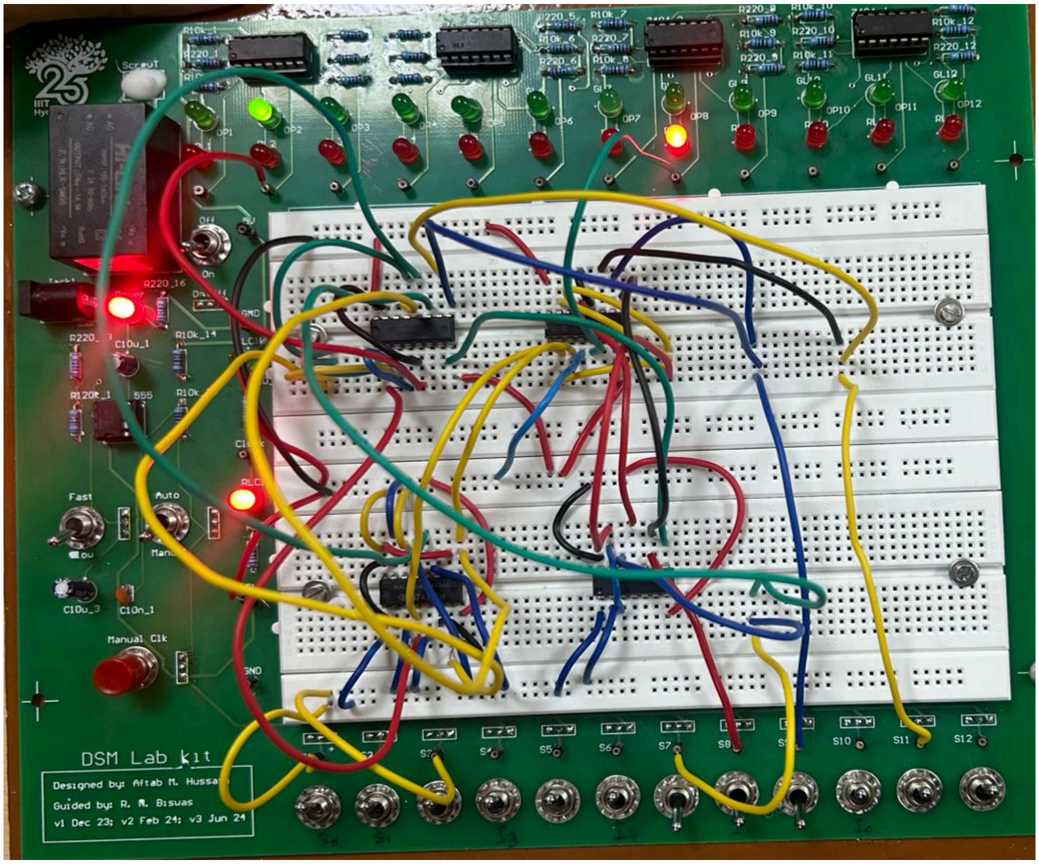
F2 F1 F0	ALU Function	Y1	Y0
000	0 (Zero)	–	0
001	A OR B	–	A + B
010	A AND B	–	A · B
011	A EX-OR B	–	A ⊕ B
100	A PLUS B	CARRY	SUM
101	A MINUS B	BORROW	DIFFERENCE
110	A PLUS B PLUS C	CARRY	SUM
111	A MINUS B MINUS C	BORROW	DIFFERENCE

Reference Circuit:



Visual Circuit:





Procedure:

1. Test each multiplexer individually by applying all inputs and recording outputs.
2. Verify all XOR gates by checking outputs for every input combination.
3. Connect Y0 and Y1 to MUX0 and MUX1, ensuring MUX1 is enabled only for F2 = 1.
4. Implement A·B, A’·B, A + B, A’ + B using 2-input MUXs and verify outputs.
5. Generate A XOR B and (A XOR B) XOR C using XOR gates and record results.
6. Assemble the full ALU with MUX0, MUX1, and F2’ enable, then test all function select combinations.
7. Compare recorded Y0 and Y1 outputs with expected ALU functions to confirm correctness.

Observation:

For logic functions (F2 = 0), only Y0 is active, giving correct logical results (AND, OR, XOR).

For arithmetic functions (F2 = 1), both Y1 (carry/borrow) and Y0 (sum/difference) were obtained.

Outputs matched exactly with the ALU function table.

Result and Analysis:

The designed programmable 1-bit ALU successfully performed 8 different operations:

Logical: OR, AND, XOR, Zero.

Arithmetic: Addition, Subtraction, Addition with Carry, Subtraction with Borrow.

Thus, the practical results matched the theoretical ALU function table, confirming the correct operation of the circuit.

Circuit Simulation:

https://www.tinkercad.com/things/9nN8BUR2F8N-alu?sharecode=FDDuvasWMvXM5xK0QbdYOKayfUcBKtbhUfKGCKuyWLQ