

ISTANBUL TECHNICAL UNIVERSITY
COMPUTER ENGINEERING DEPARTMENT

BLG 222E
COMPUTER ORGANIZATION
PROJECT REPORT

PROJECT NO : 1

GROUP NO : G71

GROUP MEMBERS:

150220901 : MOHAMAD CHAHADEH

150190068 : ÖZGÜR SEFEROĞLU

150200913 : FITNETE GUNI

SPRING 2023

Contents

0	Introduction	2
0.1	Project Parts	2
0.2	Task Distribution	2
1	Part 1	3
1.1	Implementation	3
1.2	Simulation	4
2	Part 2	4
2.1	Part 2a	4

0 Introduction

0.1 Project Parts

- Part 1: n-bit register
- Part 2: Register Files
 - Part 2a: 16-bit IR Register
 - Part 2b: Register File (RF)
 - Part 2c: Address Register File (ARF)
- Part 3: 8-bit ALU
- Part 4: Whole System Integration

0.2 Task Distribution

1. ÖZGÜR SEFEROĞLU: Part 3, Part 4
2. MOHAMAD CHAHADAH: Part 1, Part 2a, Part 2b, Part 4
3. FITNETE GUNI: Part 2c, Part 4

1 Part 1

Implementing an n-bit Register, controlled using a 2-bit FunSel signal and an Enable signal. Figure 1 Shows the diagram and characteristic equation of Part 1.

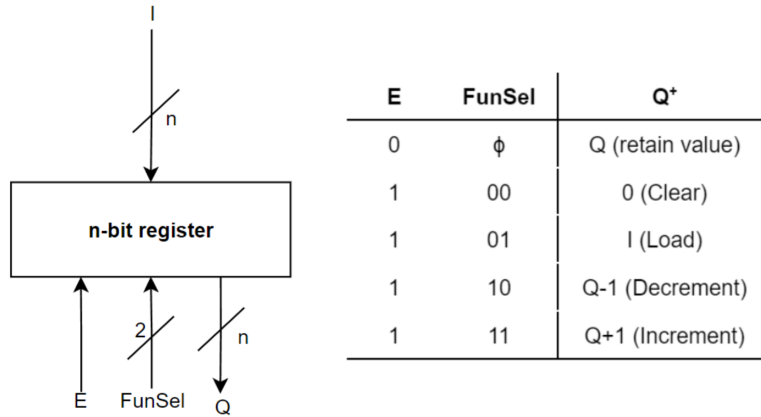


Figure 1: Diagram and characteristic equation of Part 1

1.1 Implementation

the module implemented takes one parameter n, which is the number of bits of the register, a clock signal, two control inputs being FunSel and Enable, and one output of n-bits. the module is implemented by use the always block which executes on every positive edge of the clock, an if statement to check the enable signal, and a case block for the various inputs of FunSel. the following is the code for implementing the module.

```

module part1 #(parameter n = 4)
(input clk,
input [1:0] FunSel,
input [n-1:0] data_in,
input enable,
output reg [n-1:0] data_out);

wire [n-1:0] zero = 0;
always @(posedge clk)
begin
    if (enable == 0)
        data_out <= data_out;
    else
        case (FunSel)

```

```

        2'b00: data_out <= zero;
        2'b01: data_out <= data_in;
        2'b10: data_out <= data_out - 1;
        2'b11: data_out <= data_out + 1;
    endcase

end
endmodule

```

1.2 Simulation

Figure 2 Shows a simulation of the module implemented earlier of $n = 4$.

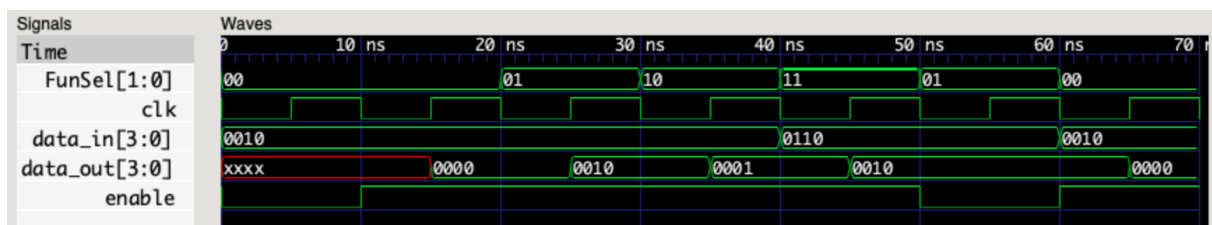


Figure 2: Simulation of Part 1, n-bit Register.

2 Part 2

2.1 Part 2a