(B2) **Polynomial Evaluator Accelerator**

By: Jonathan Chu, Joshua Rosado, and Bennett Von Ancken

University of Maryland Department of Electrical & Computer Engineering

A. James Clark School of Engineering

Problem Definition

- Development of a hardware accelerator to handle polynomial arithmetic operations
- Specification asked for commands to store polynomials, evaluate such polynomials for a set value, evaluate a polynomial with a sequence of inputs, and system reset
- Usage of Lightweight Dataflow Actors to cause the design to be portable and efficient in gate level utilization

Design Calculations & Analysis

16-bit commands are sent for the operations specified about above, so we decided to carefully split these bits in this manner:

Operation	Address	Numerical Argument
8-bits	3-bits	5-bits

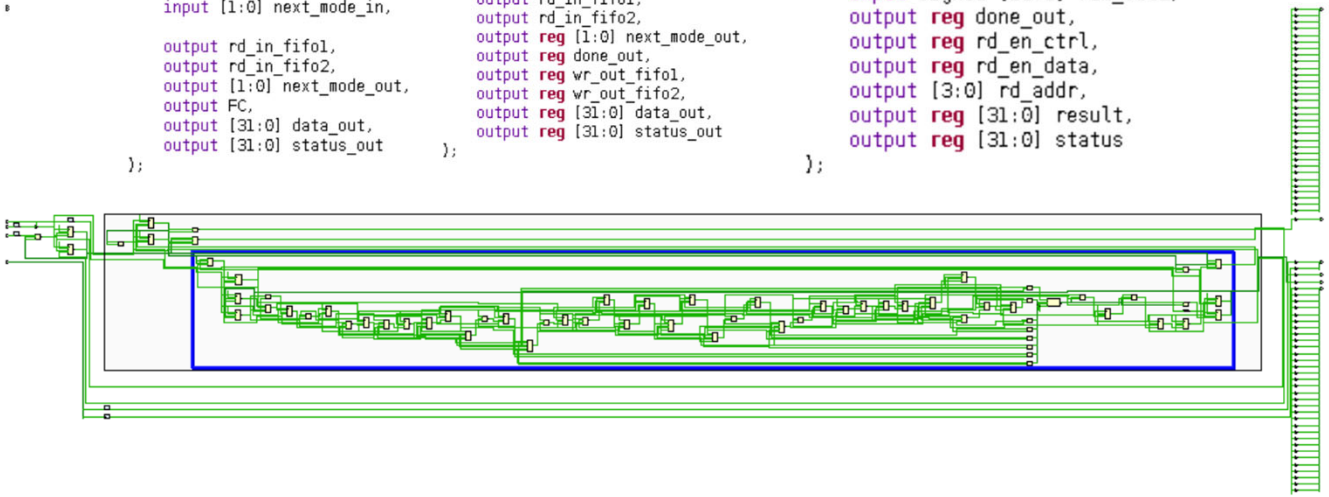
Creation of a software prototype and multiple hardware designs to carefully understand how the system would work, and which design would be the most efficient and practical for this task.

Final Design

```
module invoke_FSM_1_v2
(
    input clk, rst,
    input [15:0] control_fifo,
    input [15:0] data_fifo,

    input invoke,
    input [1:0] next_mode_in,

    output rd_in_fifo1,
    output rd_in_fifo2,
    output [1:0] next_mode_out,
    output FC,
    output [31:0] data_out,
    output [31:0] status_out
);
```

```
module firing_state_FSM2 (
    input clk, rst,
    input [15:0] data_in_fifo1,
    input signed [15:0] data_in_fifo2,
    input start_in,
    input [1:0] next_mode_in,

    output rd_in_fifo1,
    output rd_in_fifo2,
    output reg [1:0] next_mode_out,
    output reg done_out,
    output reg wr_out_fifo1,
    output reg wr_out_fifo2,
    output reg [31:0] data_out,
    output reg [31:0] status_out
);
```

```
module operate_mode_FSM_3_v2
(
    input clk, rst,
    input start_in,
    input [15:0] ram_out1,
    input signed [15:0] ram_out2,
    output reg done_out,
    output reg rd_en_ctrl,
    output reg rd_en_data,
    output [3:0] rd_addr,
    output reg [31:0] result,
    output reg [31:0] status
);
```



Prototype & Test Results