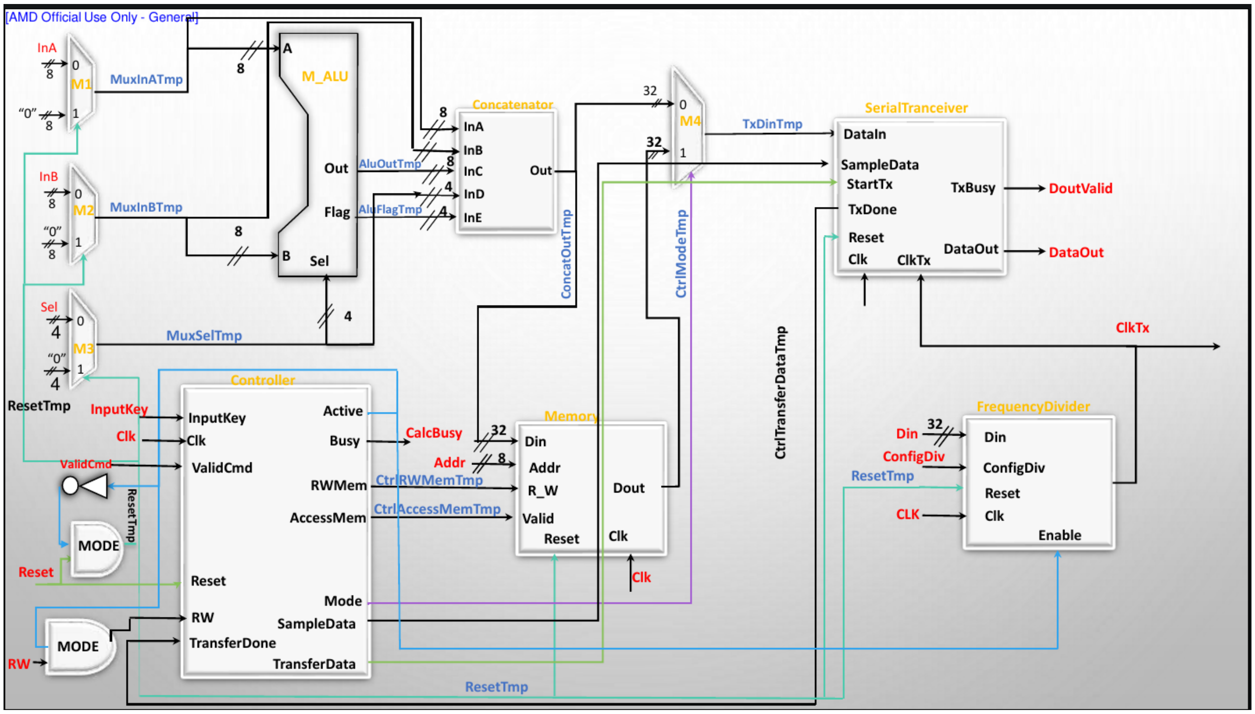
Binary Calculator Documentation

AMD Autumn-Practice-2023

Chirica Andrei

SCHEMATIC



Inputs:

- **InA**: first input for ALU
- **InB**: second input for ALU
- **Sel**: used to give ALU which operation should be performed
- **InputKey**: used to insert the secret sequence(1-0-1-0) and if the sequence is correct it sets the Mode wire
- **Clk**: the global clock signal of the circuit
- **ValidCmd**: InputKey is taken in consideration and operations are executed whether ValidCmd is on or off
- **Reset**: asynchronously resets the circuit
- **RW**: command to whether read(0) or write(1) in memory
- **ADDR:** the address to which will be written or read to/from memory
- **Din**: gives the grade of division to generate the clock for a serial transmission (Note: should be a divisor of 32 to work properly)
- **ConfigDiv**: the configuration coming from Din is saved whether ConfigDiv is 0 / 1

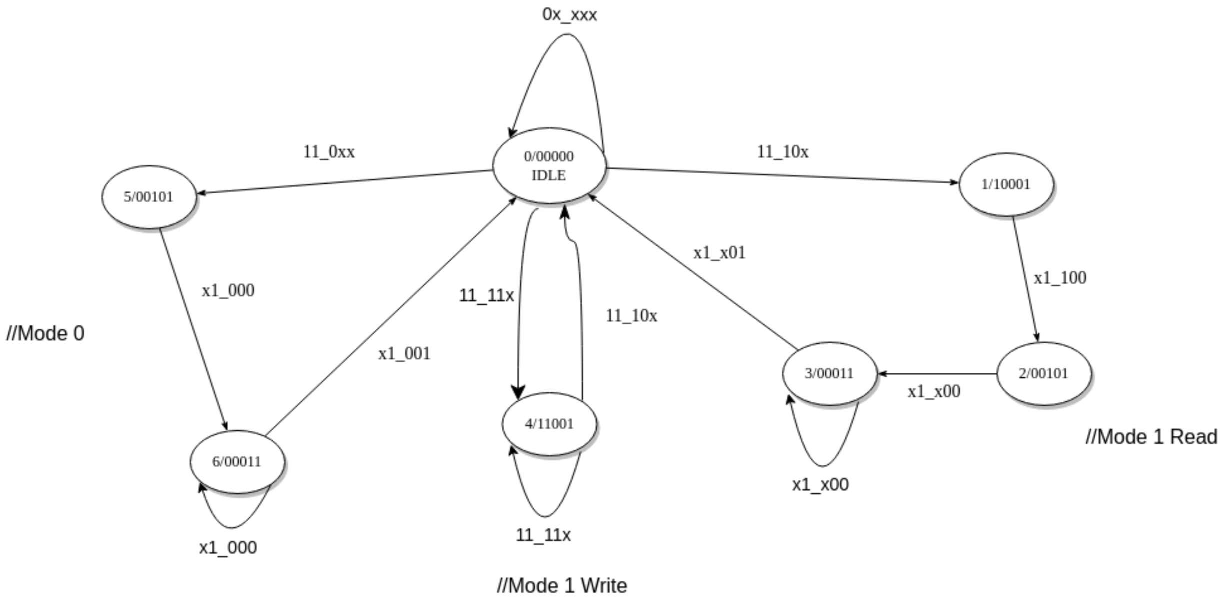
Outputs:

- **DoutValid**: whether DataOut should be taken in consideration
- **DataOut**: serial output of the calculator; The number of bits that can be transmitted can be variable
- **ClkTx**: the clock generated by the Frequency_Divider for the serial output
- **CalcBusy**: indicates whether the calculator is busy executing a command or not

CONTROLLER FSM

IN = {ValidCmd, Active, Mode, RW, TxDone}

OUT = {AccesMem, RWMem, SampleData, TxData, Busy}



//Mode 1 Read

Represented in the right portion of the graph. When it reaches the third state it will stay there until the TxDone signal is 1. After that it changes the state to IDLE.

//Mode 1 Write

Represented by the fourth state of the graph. It goes there when the RW input is set to 1 and it stays there until RW is set to 0. (Caution! If the other inputs will not be changed on the next clock when it arrives in the IDLE state it will go into the //MODE 1 READ state).

//Mode 0

It can be seen in the left part of the graph. Since it doesn't use memory the only operation is to write the data to the serial output.

KNOWN BUGS

1. When transmitting a result to the serial output whether in //Mode 1 Read or //Mode 0, DataOut should be taken in consideration a clock after the StartTx input is set to 1. Otherwise the result will be erroneous.
2. After a complete serial transmission the i variable inside the Serial Transceiver may go into an oscillating state between the maximum and maximum-1 value. It does not affect the output results and when a new command for a transfer is received is behaving normally.(It happened when the length of DataOut was 8 and the calculator was put into an IDLE state after the end of the transmission).

TESTS DONE

1. Operation for ALU: 7 + 8; Resul = 15; Flags = 4'b0000;
2. //MODE 1 WRITE at Address 4 - Worked
3. //MODE 1 READ from Address 4 - Worked with the length of the Serial Transceiver's DataOut 1 and 8 bits
4. //MODE 0 Worked with the length of the Serial Transceiver's DataOut 1 and 8 bits