

# ECE337 Project Final Presentation

## Handwritten Digit Recognizer using Neural Networks

TA: Anirudh Sivakumar

Wednesday (11:30), Team 3: David Pimley, Dustin Andree, Vadim Nikiforov, Chan Weng Yan

Purdue University, Electrical and Computer Engineering Dept.



# Introduction

## Project Features:

- Handwritten digit detection (MNIST Database)
- Interfacing with an external training system
- Error code system for improper usage
- Parallelization and pipelining to improve performance

# MNIST Digit Examples



# Introduction

## Motivation for Project:

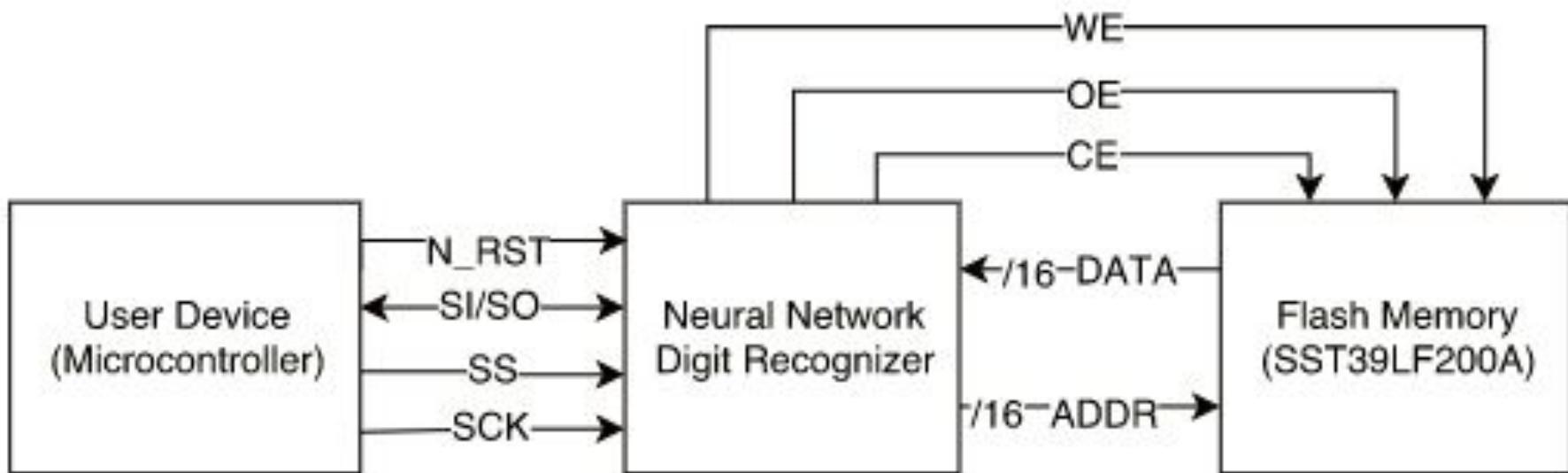
- Data Interpretation
  - Checks
  - Addresses
  - Signs
- High Accuracy with use of Neural Nets / Machine Learning
- Dedicated Chip for Higher Performance

# Introduction

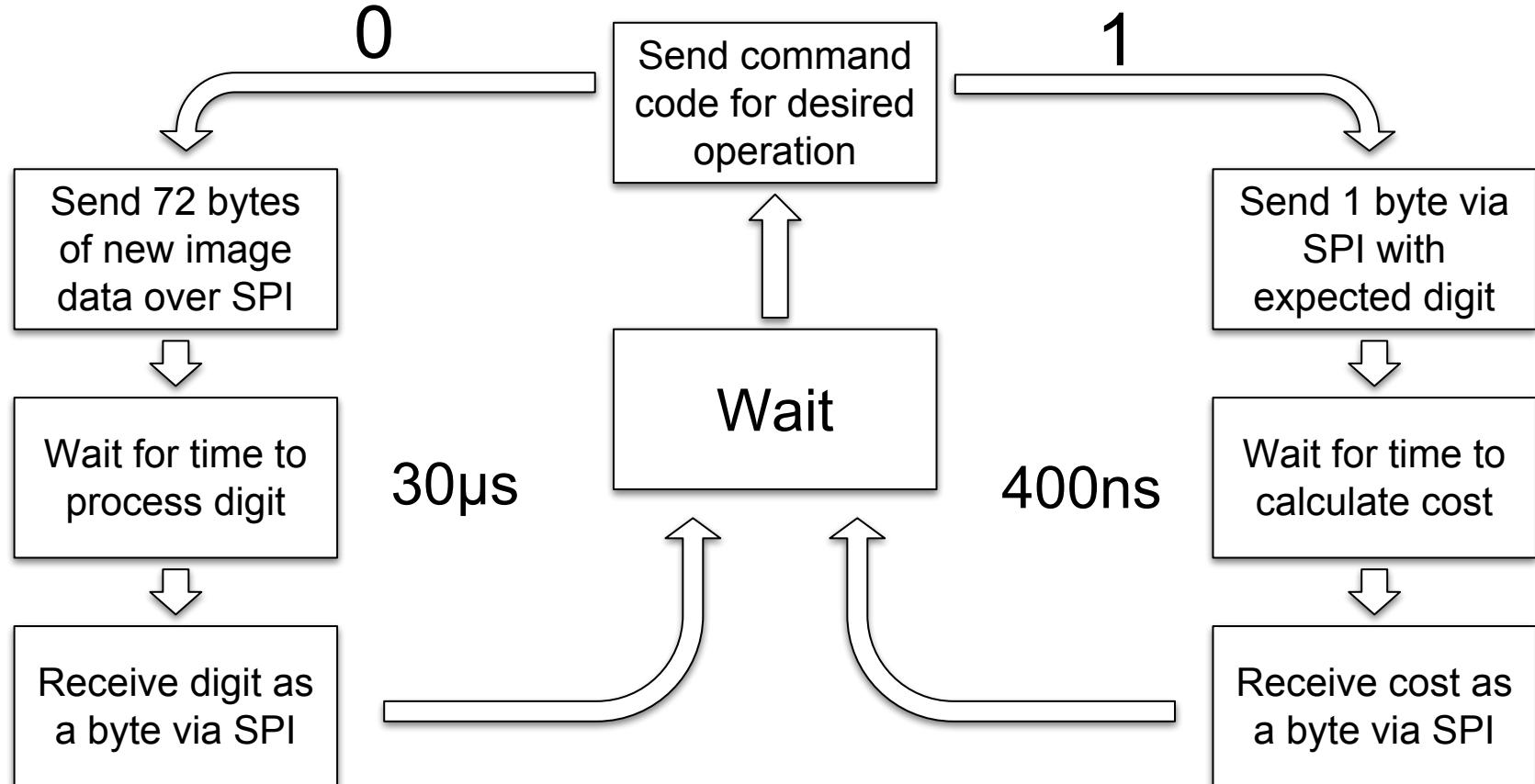
## ASIC Implementation:

- Use of Parallelization in Neural Nets
- Pipelined ALU
- ALU made for Neural Net Calculations
- Low Power

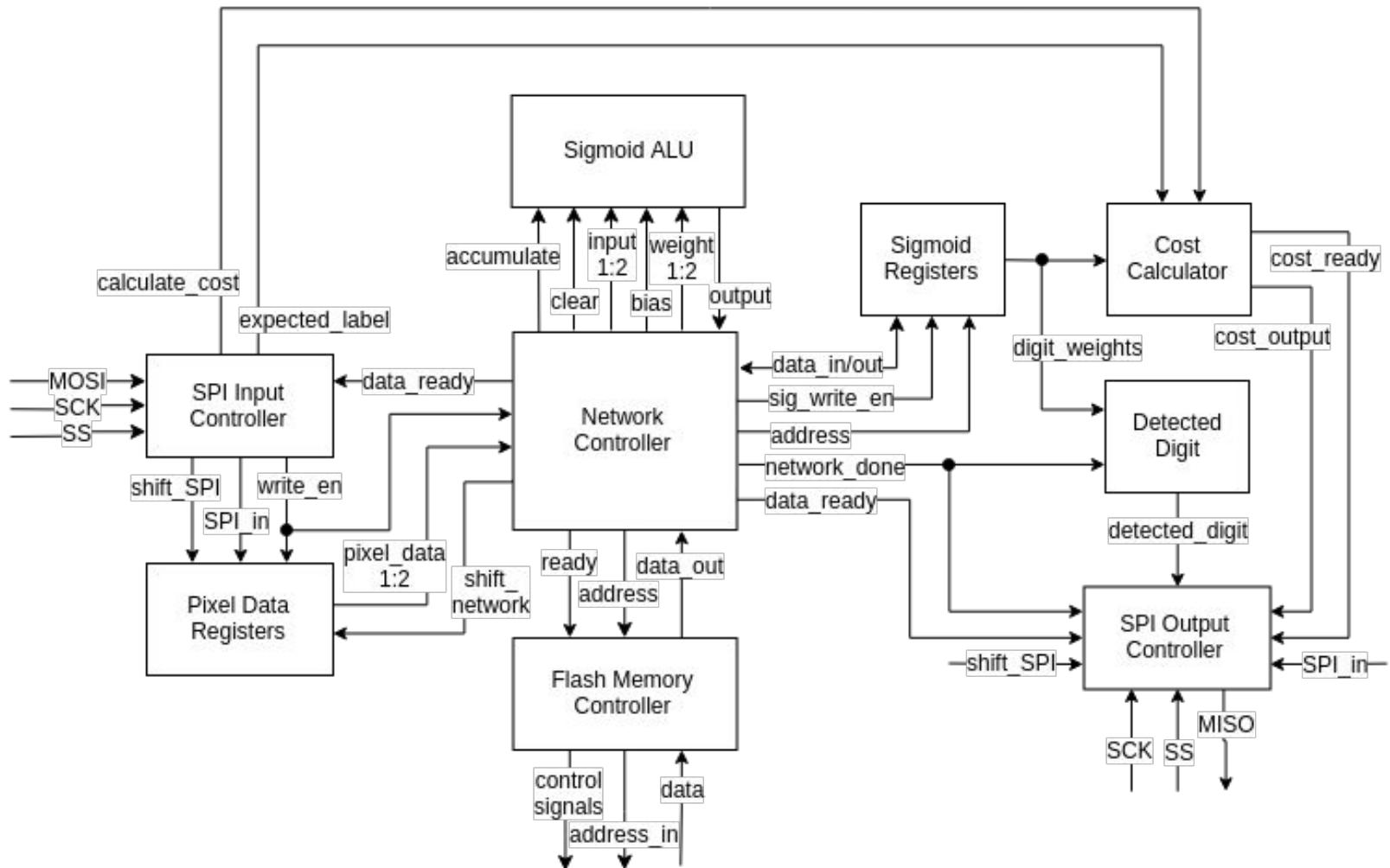
# Top Level Interfaces



# Top Level Usage Diagram



# Top Level Block Diagram



# Successful Design Decisions

**4 Bit Sigmoid and Pixel Data (Percentages)**

Width of Image (Pixels)

Width of Image (Pixels)	2	4	6	8	10	12	14	16	18	20
Number of Hidden Neurons	39.00	36.08	81.40	81.39	88.73	88.61	89.67	88.40	90.49	91.41
28	18.27	41.87	82.60	87.68	86.40	88.84	89.72	88.18	88.47	91.37
26	32.27	37.36	82.05	87.48	89.21	87.69	88.75	90.59	87.51	90.65
24	38.42	37.38	61.95	85.30	87.44	87.71	85.80	88.83	90.13	91.67
22	27.34	66.50	76.00	84.26	87.72	88.01	89.38	86.80	89.55	91.19
20	40.90	64.69	78.13	84.59	87.57	73.15	87.36	89.84	89.25	86.79
18	34.96	49.66	82.72	78.96	88.25	88.03	88.47	87.07	89.77	90.43
16	35.93	47.09	68.83	83.05	85.73	89.23	87.46	88.33	90.58	88.75
14	30.47	66.94	79.37	85.39	86.83	74.29	87.76	87.02	87.52	88.75
12	30.64	47.36	80.55	80.10	80.79	82.07	79.39	83.63	84.80	79.09
10	30.77	45.78	71.91	66.13	66.27	73.80	77.23	77.41	78.23	80.81
8	26.72	36.69	54.79	51.36	60.21	54.08	56.71	48.91	60.90	65.22
6	28.14	37.17	32.11	31.13	33.43	32.30	38.98	34.44	35.38	38.10
4	23.31	20.02	21.02	23.14	23.19	20.20	19.36	26.42	22.93	19.45
2										

# Significant Design Decisions

## Variation of Image Input

- Use of 12 x 12 Pixel Input Images / 8 Hidden Neurons
  - Reduced ALU Load
  - Reduced Number of Simulated Flip Flops
  - Medium Efficiency
  - Medium-High Accuracy

# Significant Design Decisions

## Variation of Binary Precision

- Use of Lower Precision Fixed Point Numbers
  - Allows for Higher Clock Speed
  - Simplifies Arithmetic in ALU
  - Reduces Area

# Results for Fixed Success Criteria

Success Criteria	Status
Test Bench Exists for all Top-Level components	<u>Passed</u> , Every top level block has a test bench
Entire design synthesizes completely without latches or warnings	<u>Passed</u> , design is latch free and no warnings are generated when synthesized
Source and mapped version have the same behavior	<u>Passed</u> , source and mapped versions operate with the same functionality on the same test sets.
IC Layout passes geometry and connectivity tests.	<u>Passed</u> , Layout passes with zero errors in geometry and connectivity tests.
Entire design compiles with targets for area, pin count, throughput and clock frequency	<u>Passed</u> Area: target = 3.2 X 3.2, Final = 2.7 X 2.7 Pin Count: target = 48, Final = 48 Throughput: Target = 10,000 images/sec, Final = 11,753 images/sec Clock frequency: Target = 200 MHZ Final = 200 MHZ

# Results for Design Specific Success Criteria

Success Criteria	Status
Design recognizes more than 60% of all digits.	<u>Passed</u> , 88.76% accuracy with a 10000 test image vector.
Design outputs the correct value for the cost function.	<u>Passed</u> , outputs the proper cost over 100 randomized sets of sigmoid confidences and label inputs.
Design is able to propagate a set of weights and biases through a sigmoid neuron.	<u>Passed</u> , the design is able to recognize digits showing that the sigmoid neurons have weights and biases propagated correctly.
Design is able to process more than 10,000 images a second.	<u>Passed</u> , final throughput = 11753 images/sec
Design is able to properly load a 12 x 12 pixel grayscale image through SPI.	<u>Passed</u> , design can use SPI to send in a 12 x 12 image
Design is able to interface with a SST39LF200A flash memory chip.	<u>Passed</u> , design can get data from a simulated flash memory chip

# Successful Rapid Digit Detection

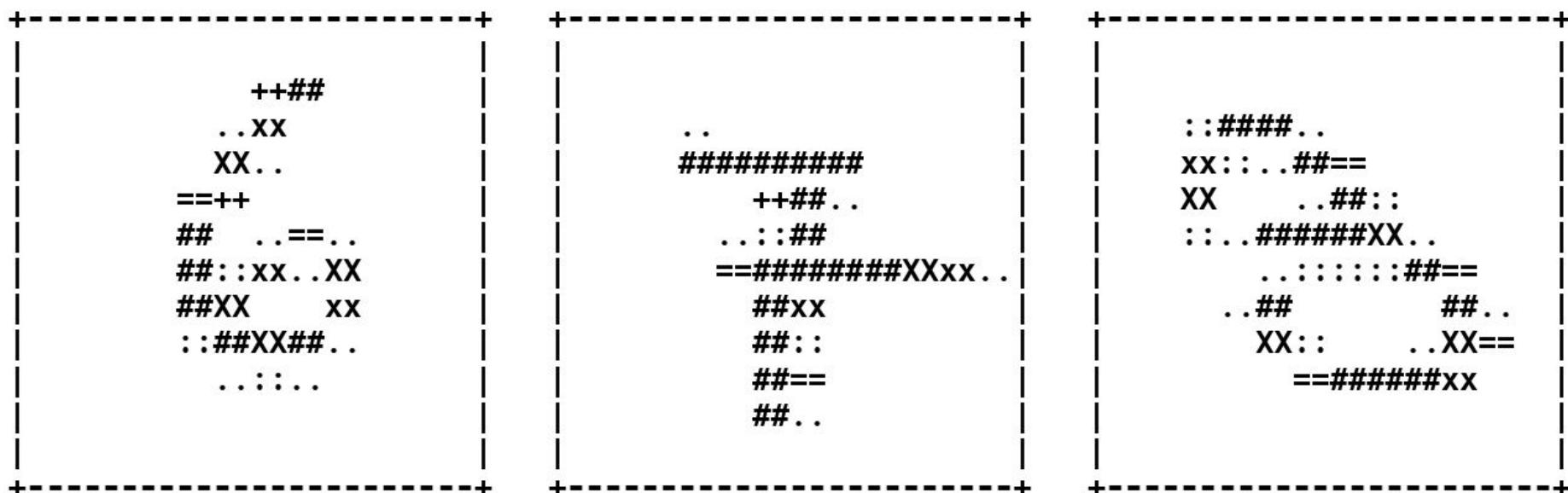
**Design is able to process more than 10,000 images a second.**

```
** Info: Num tested: 10000, Num correct: 8876
Time: 850750040 ns Scope: tb_digit_recognizer_final File:
```

**Final Throughput:**  $10000 / 850.0750040 \text{ ms}$   
 $= 11763 \text{ images / sec}$

# Successful Digit Recognition

```
** Info: Num tested:          10000,  Num correct:      8876
Time: 850750040 ns  Scope: tb_digit_recognizer_final File:
```

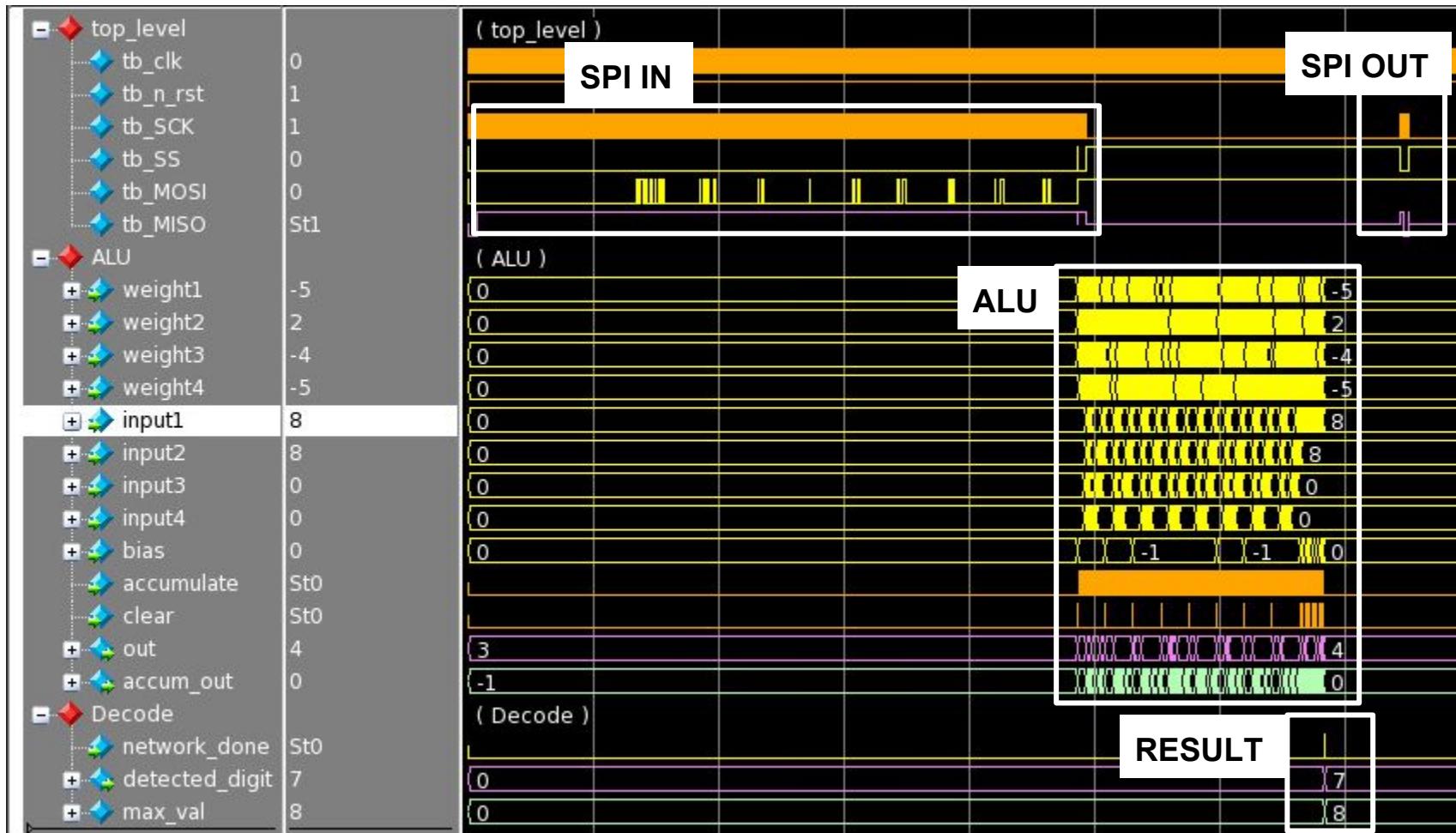


Result: 6  
Expected: 6

Result: 7  
Expected: 7

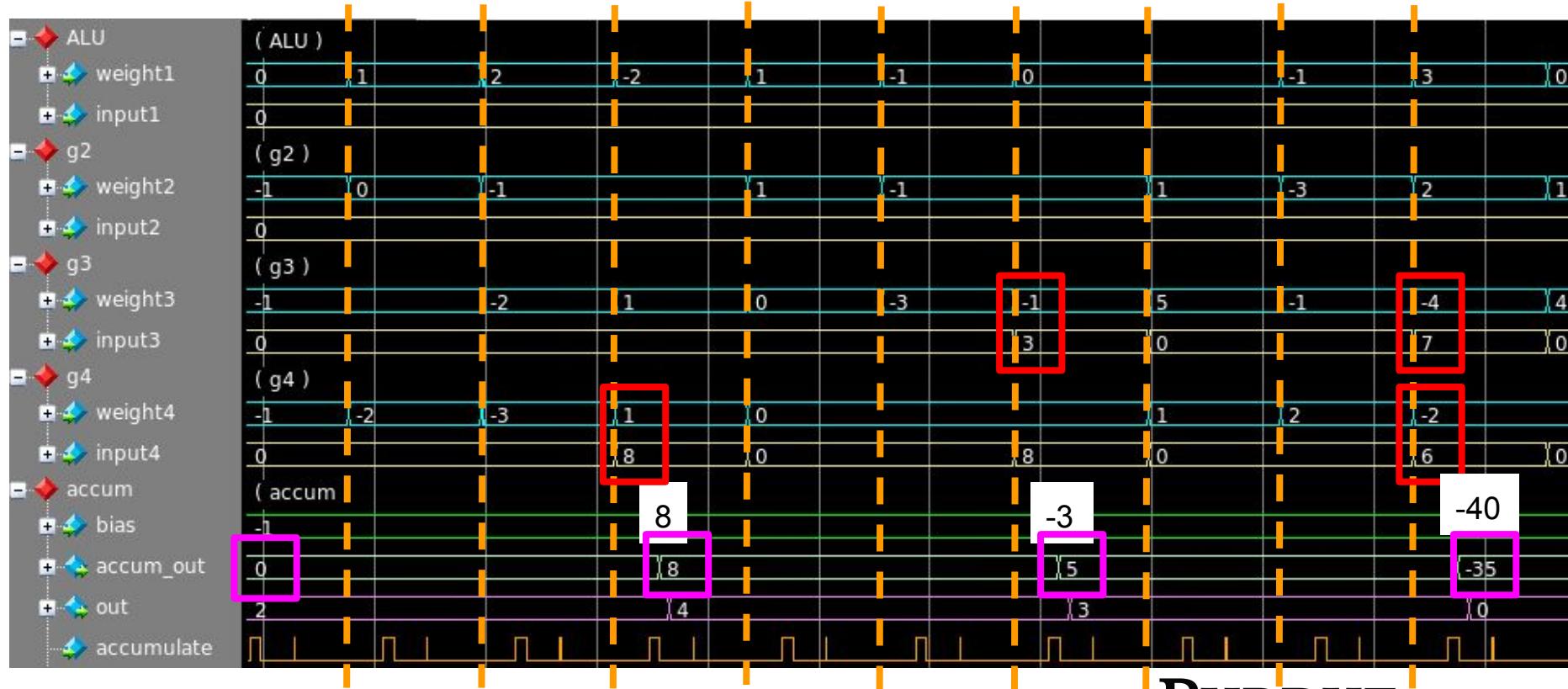
Result: 2  
Expected: 3

# Successful Digit Recognition



# Successful Weight / Bias Propagation

Design is able to propagate a set of weights and biases through a sigmoid neuron.



# Correct Cost Functionality

**Design outputs the correct value for the cost function.**

- States {SUB, SQ, ADD, STO} for each image
- Accumulation of squared errors
- L\_INPUT to increment index of input vectors
- Cost output in XXXX.XXXX fixed point binary

**LABEL\_INPUTS:** ['0000', '0000', '1000 : (1.00)', '0000', '0000', '0000', '0000', '0000', '0000']

**SIGMOID\_CONF:** ['0000', '0000', '0110 : (0.75)', '0000', '0000', '0000', '0000', '0000', '0000']

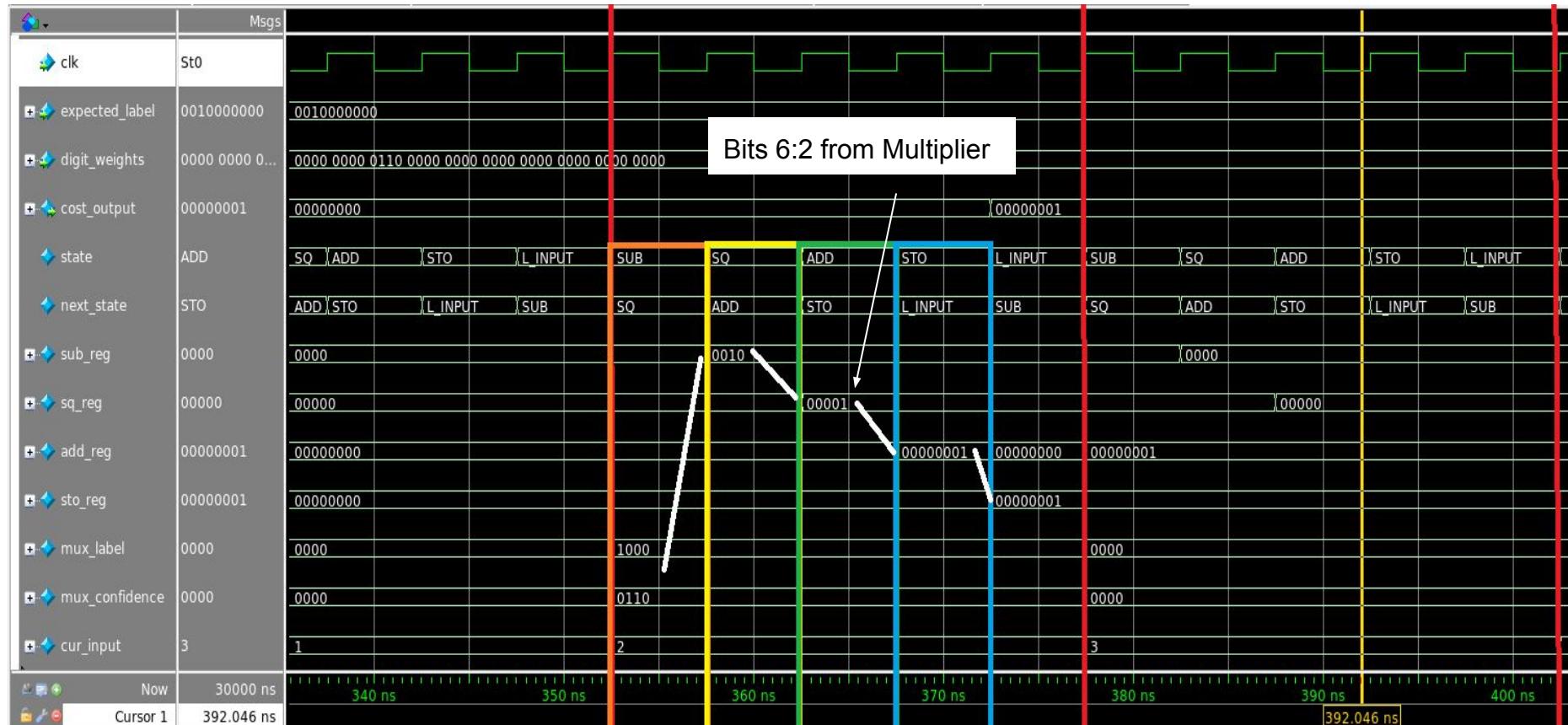
**SUB\_MATRIX :** ['0000', '0000', '0010 : (0.25)', '0000', '0000', '0000', '0000', '0000', '0000']

**SQ\_MATRIX :** ['00000000', '00000000', '00000001 : (0.0625)', '00000000', '00000000',  
'00000000', '00000000', '00000000', '00000000', '00000000']

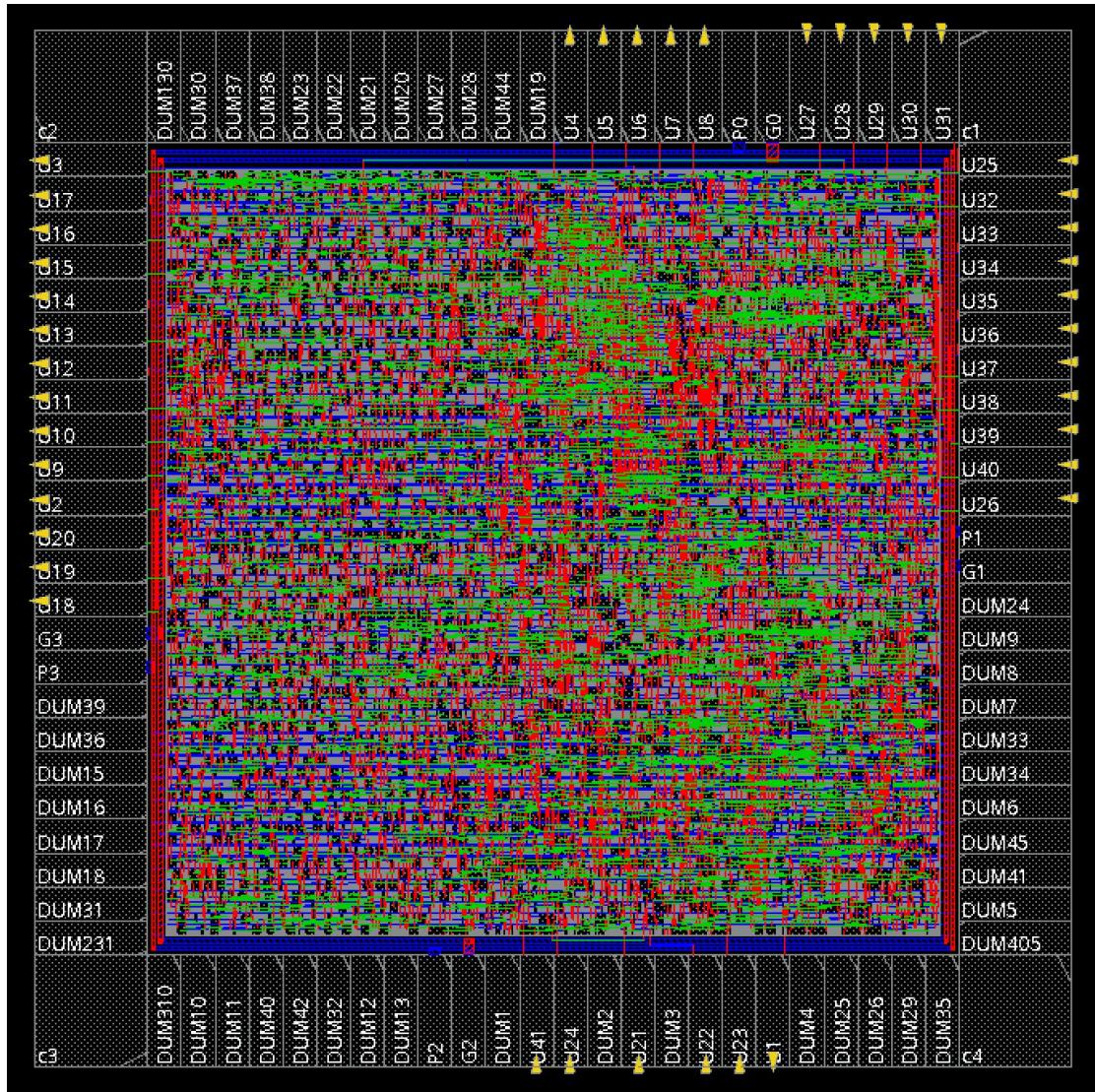
**COST\_OUTPUT : 00000001 = 0.0625**

# Wave Output of Successful Cost Function

Design outputs the correct value for the cost function.

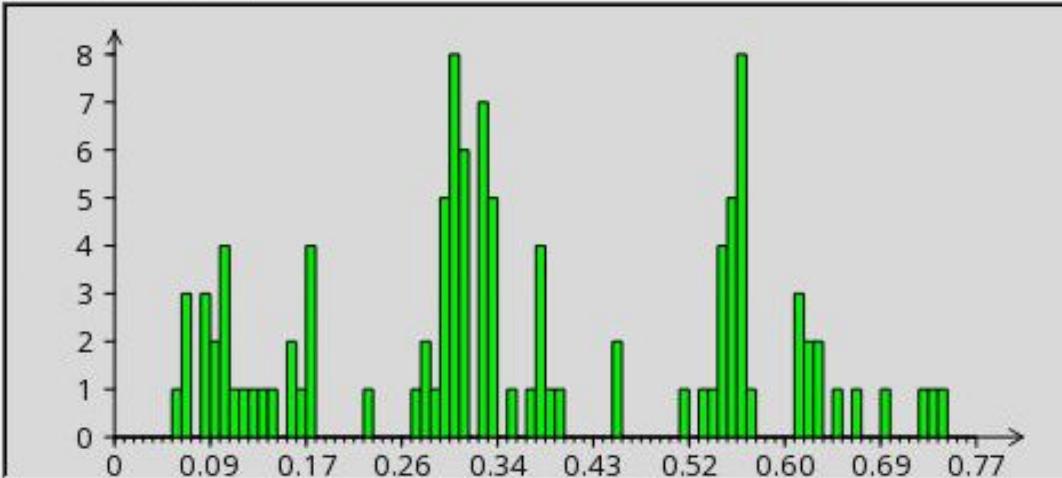


# Successful Chip Layout



# Successful Timing Analysis

Path Histogram



Category Summary

Name:	all
Total Path:	105
Passing Path:	105
Failing Path:	0
WNS:	0.0000
TNS:	0.0000



Target Clock rate: **5.00ns** 200MHz

Mapped Longest Critical Path: **3.29ns** 300MHz

Longest Predicted Critical Path: **4.40ns** (8-bit Multiplier)

# Successful Geometry and Connectivity

```
***** End: VERIFY CONNECTIVITY *****
Verification Complete : 0 Viols. 0 Wrngs.
(CPU Time: 0:00:00.3  MEM: 0.000M)
```

```
***** End: VERIFY CONNECTIVITY *****
Verification Complete : 0 Viols. 0 Wrngs.
(CPU Time: 0:00:00.4  MEM: 0.000M)
```

Verification Complete : 0 Viols. 0 Wrngs.

```
*****End: VERIFY GEOMETRY*****
*** verify geometry (CPU: 0:00:02.0  MEM: 1.0M)
```

# Conclusion

- Challenges

- Bit Precision
- Creating test scenarios
- Debugging Network Controller

# Conclusion

- **Different Design Approach**
  - FIFO Block for Network Controller
  - Multiple Communication Standards (SPI, USB, UART)
  - Floating Point Arithmetic
  - Back Propagation Block

# Conclusion

- **Improvements to Existing Design**
  - Larger Test Images (28 x 28)
  - Increased Number of Hidden Neurons
  - Higher Clock Rates for SPI

# References

- [1] “Mnist for ml beginners,” 6 2017. [Online]. Available: [https://www.tensorflow.org/versions/r1.1/get\\_started/mnist/beginners](https://www.tensorflow.org/versions/r1.1/get_started/mnist/beginners)
- [2] *2 Mbit/4 Mbit/8 Mbit (x16) Multi-Purpose Flash*, Silicon Storage Technology, Inc., 4 2011.
- [3] M. A. Nielsen, *Neural Networks and Deep Learning*. Determination Press, 2018. [Online]. Available: <http://neuralnetworksanddeeplearning.com/>
- [4] “Design success criteria and design budgeting,” Purdue University, 3 2017.



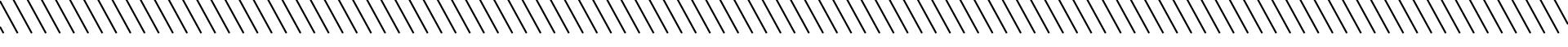
**Questions?  
Comments?  
Suggestions?**

**WE ARE PURDUE. WHAT WE MAKE MOVES THE WORLD FORWARD.**

**PURDUE**  
UNIVERSITY



# Additional Figures



**WE ARE PURDUE. WHAT WE MAKE MOVES THE WORLD FORWARD.**



# Fixed Success Criteria

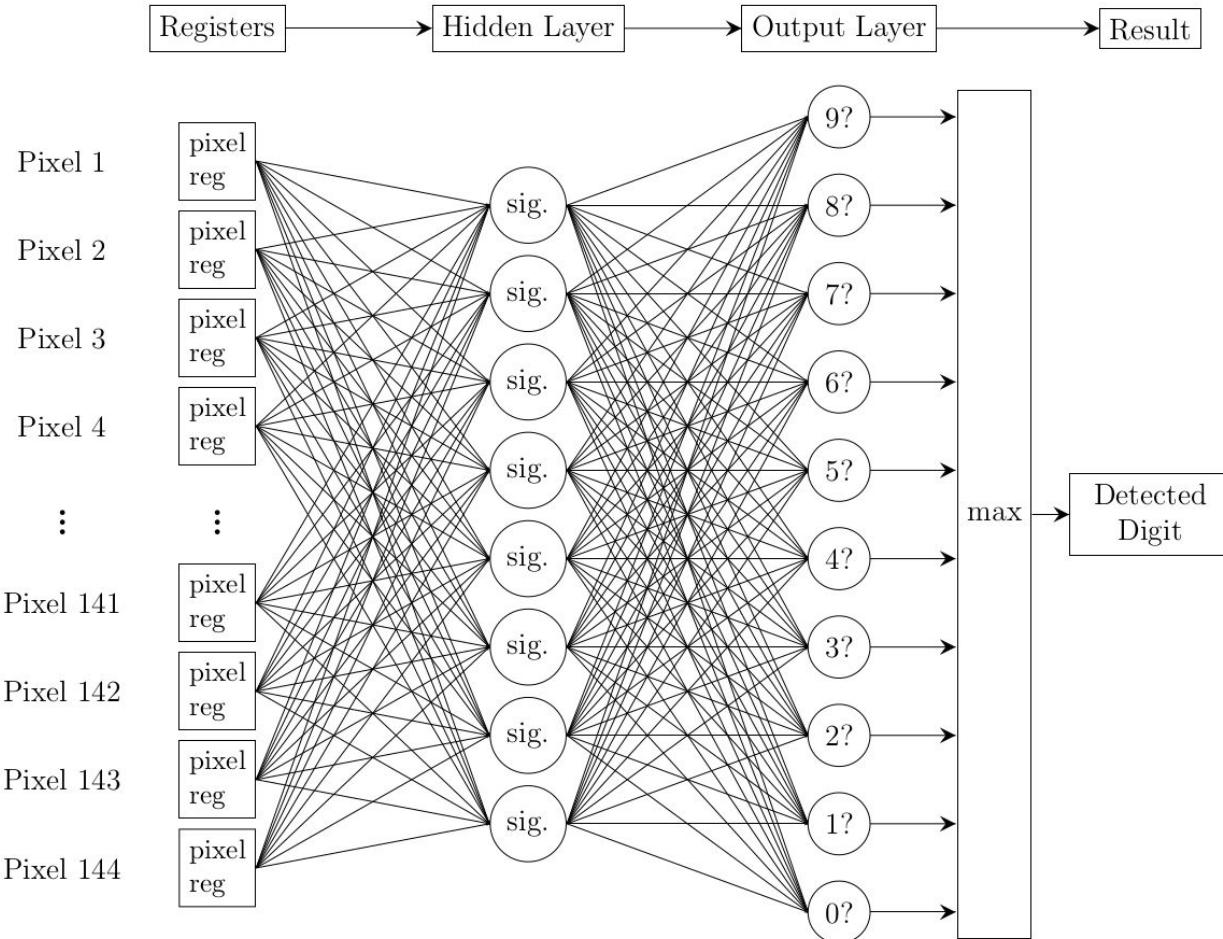
## 2.1 Fixed Criteria

1. (2 points) Test benches exist for all top-level components and the entire design. The test benches for the entire design can be demonstrated or documented to cover all of the functional requirements given in the design specific success criteria.
2. (4 points) Entire design synthesizes completely, without any inferred latches, timing arcs, and sensitivity list warnings.
3. (2 points) Source and mapped version of the complete design behave the same for all test cases. The mapped version simulates without timing errors except at time zero.
4. (2 points) A complete IC layout is produced that passes all geometry and connectivity checks.
5. (2 points) The entire design complies with targets for area, pin count, throughput (if applicable), and clock rate. The final targets for these parameters will be determined by course staff based on your design review. Failure to reach any of the targets will result a score of 1 out of 2 provided that you are within 50% on area, 10% on pin count, and 25% on throughput. Doing worse in any category will result in a score of 0 out of 2.

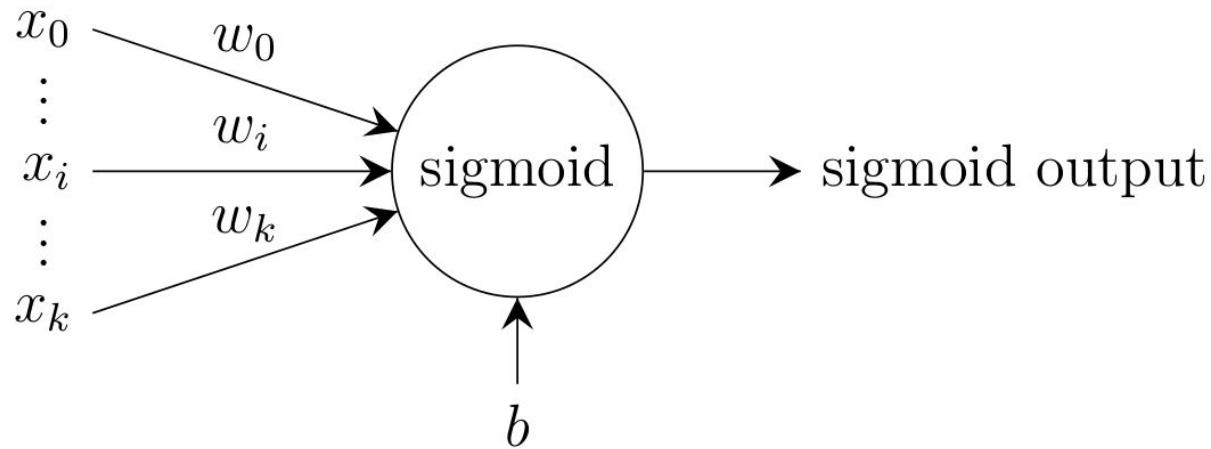
# Design Specific Success Criteria

## 2.2 Design Specific Success Criteria

1. (1 point) Demonstrate by simulation of Verilog test benches that the complete design is able to detect at least 60% of digits.
2. (1 point) Demonstrate by simulation of Verilog test benches that the complete design is able to output a proper cost output for use in external training modules.
3. (2 points) Demonstrate by simulation of Verilog test benches that the complete design is able to correctly propagate a set of weights and biases through a sigmoid neuron.
4. (1 points) Demonstrate by simulation of Verilog test benches that the complete design is able to process 10,000 images per second.
5. (2 points) Demonstrate by simulation of Verilog test benches that the complete design is able to properly load in a  $12 \times 12$  grayscale image through SPI, and output the specified error codes over SPI.
6. (1 points) Demonstrate by simulation of Verilog test benches that the complete design is able to correctly interface with a SST39LF200A flash memory chip by using specified timing parameters as a reference.

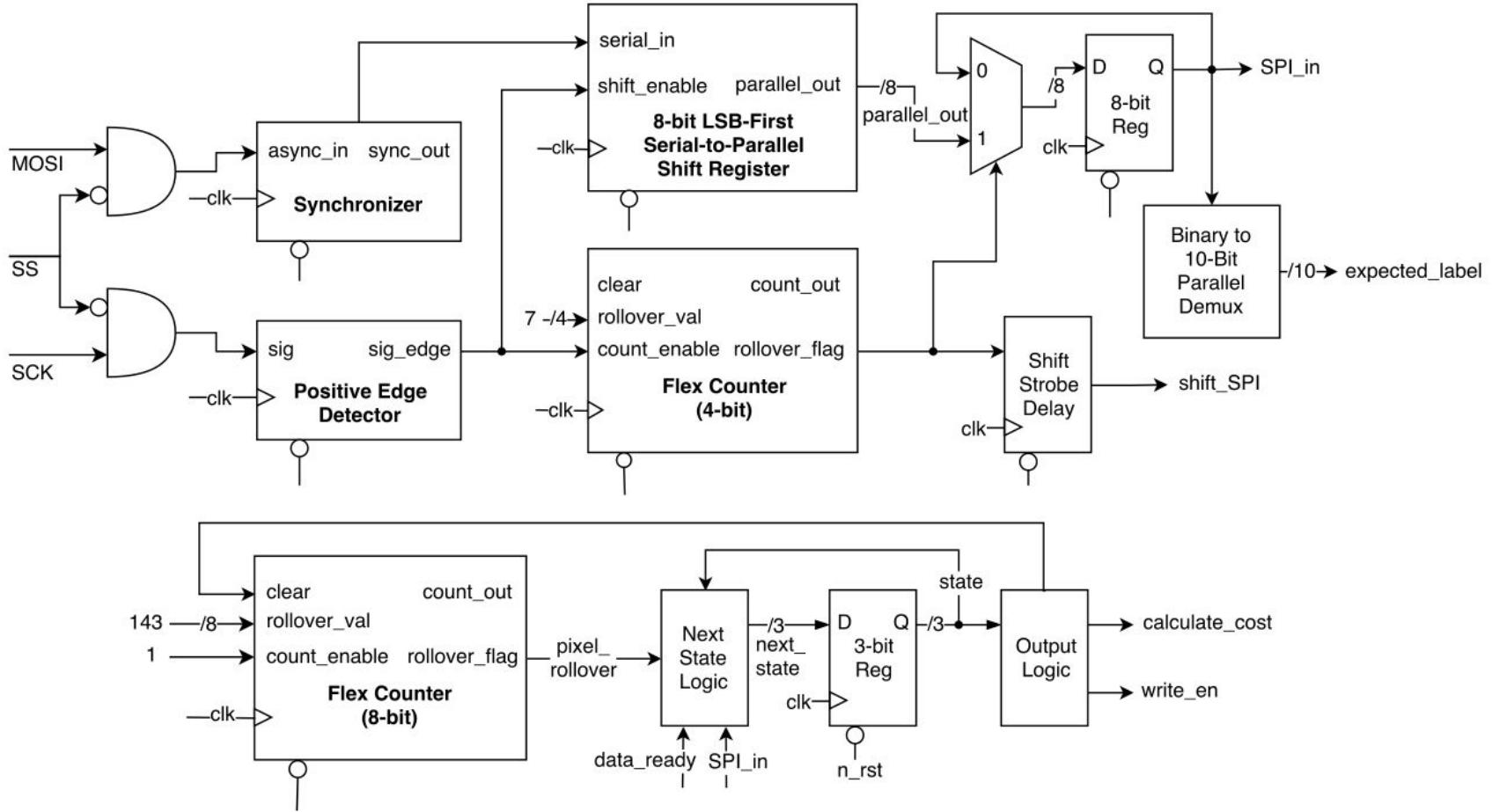


**High-level network structure with input, hidden, and output layer.**

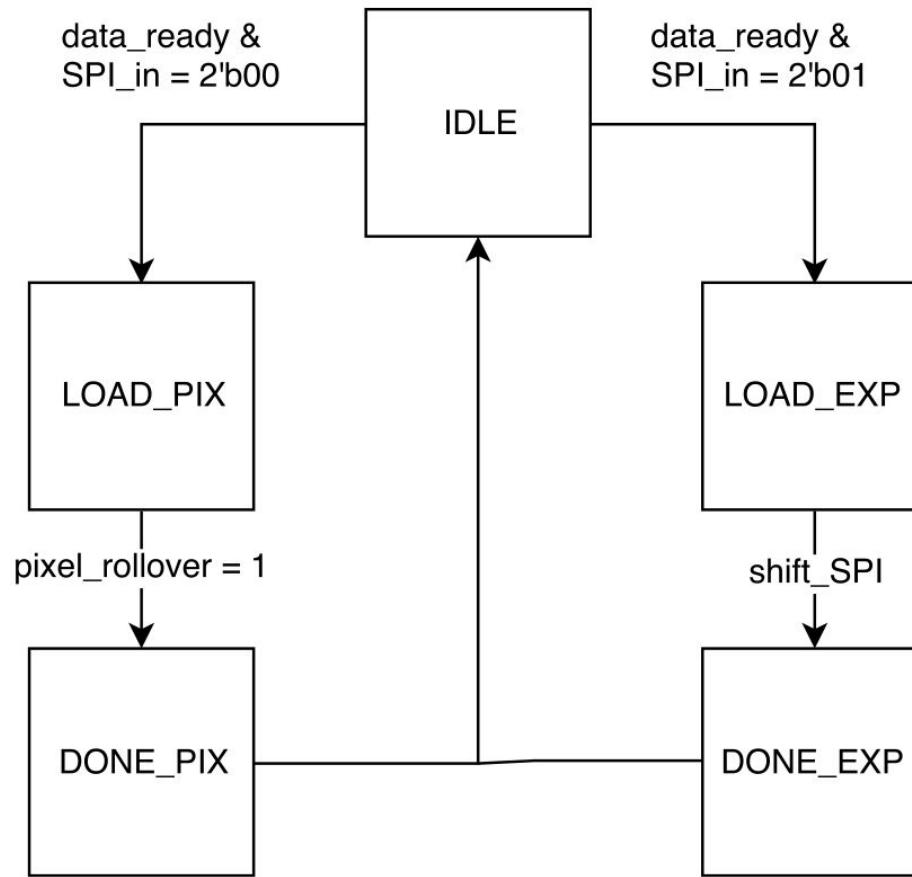


$$z = \sum_i w_i x_i + b \qquad \sigma(z) \equiv \frac{1}{1 + e^{-z}}$$

**Brief overview of a sigmoid neuron**



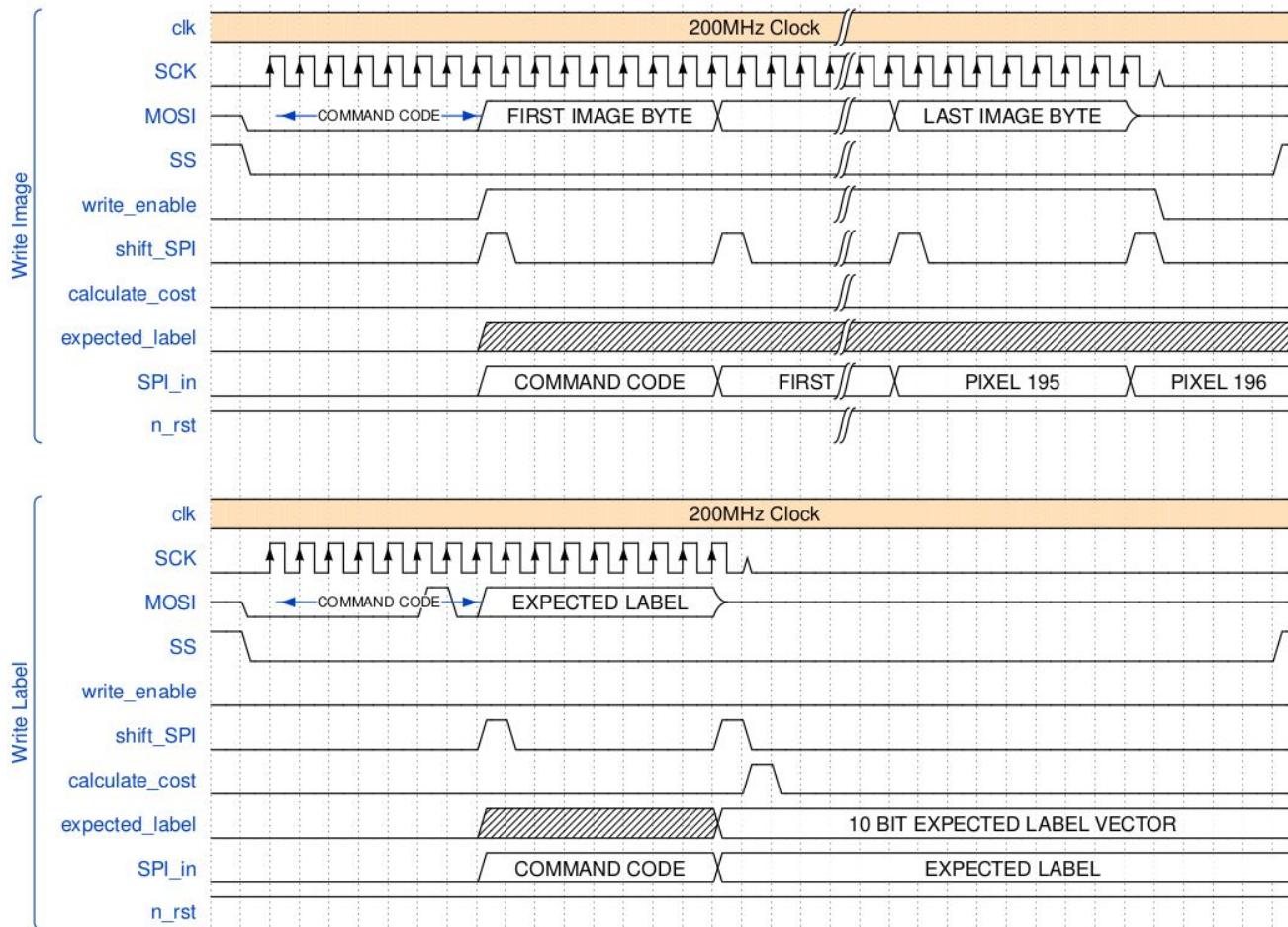
## SPI Input Controller block diagram



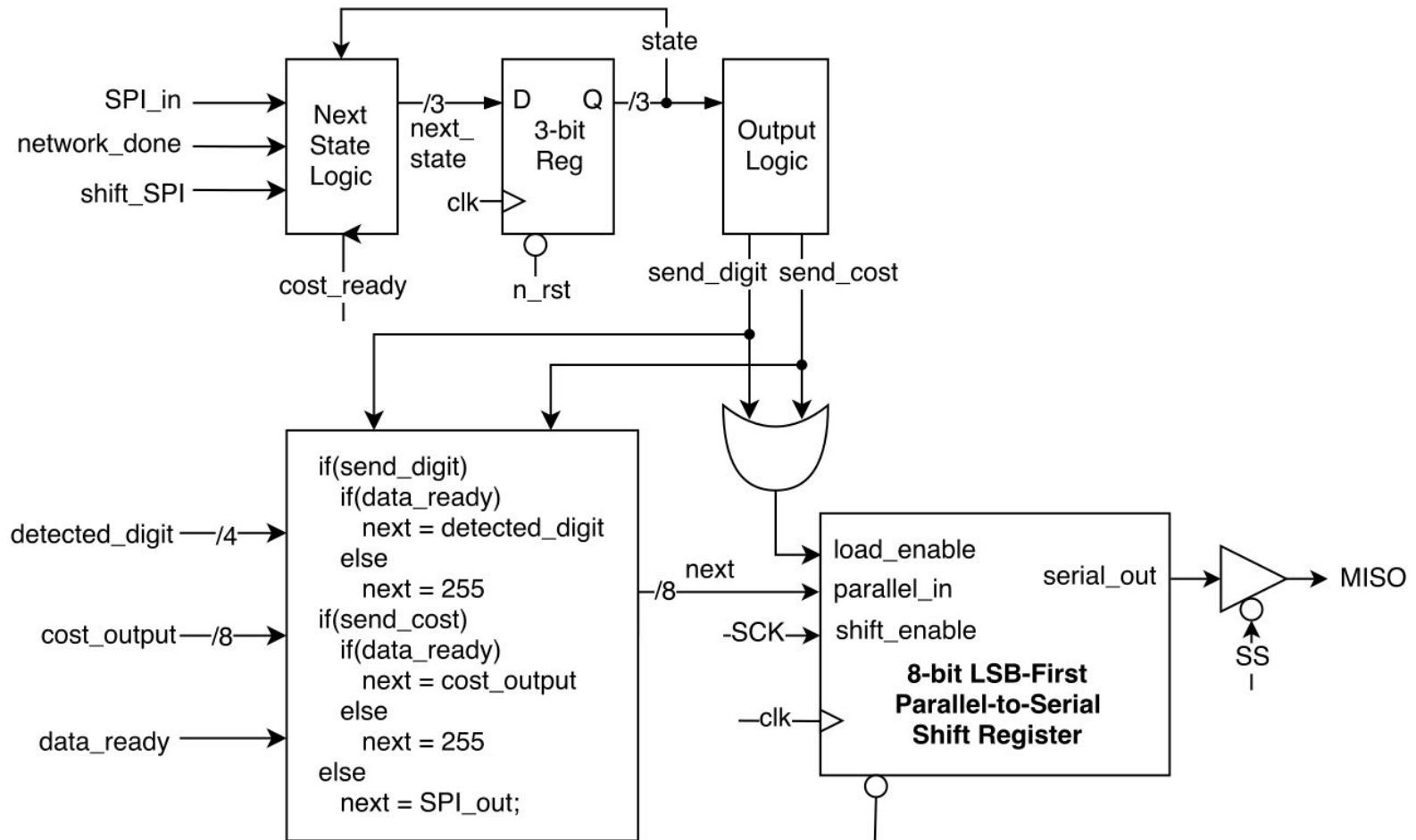
SPI Input Controller state diagram

state	is_idle	calculate_cost	write_enable_pixel
IDLE	1	0	0
LOAD_PIX	0	0	1
DONE_PIX	0	0	0
LOAD_EXP	0	0	0
DONE_EXP	0	1	0

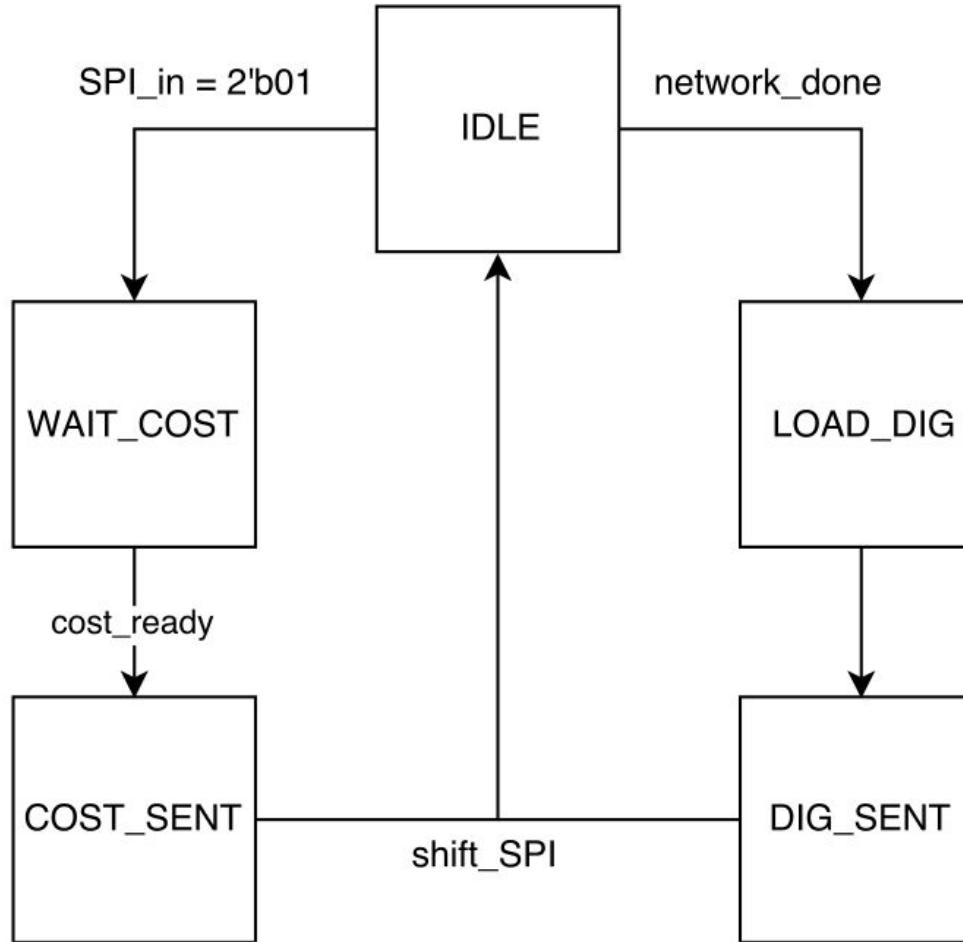
## SPI Input Controller output logic



## SPI Input Controller timing diagram



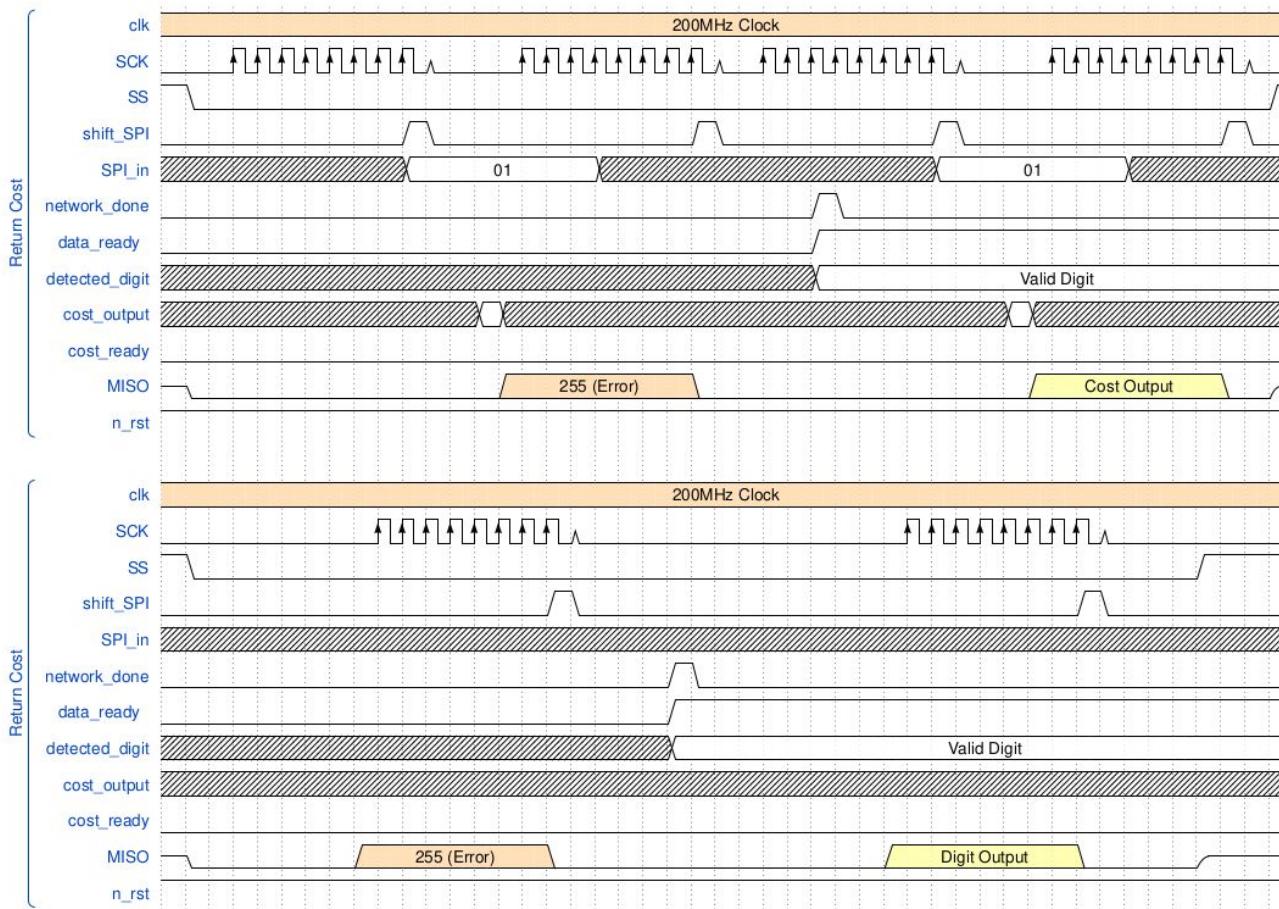
## SPI Output Controller block diagram



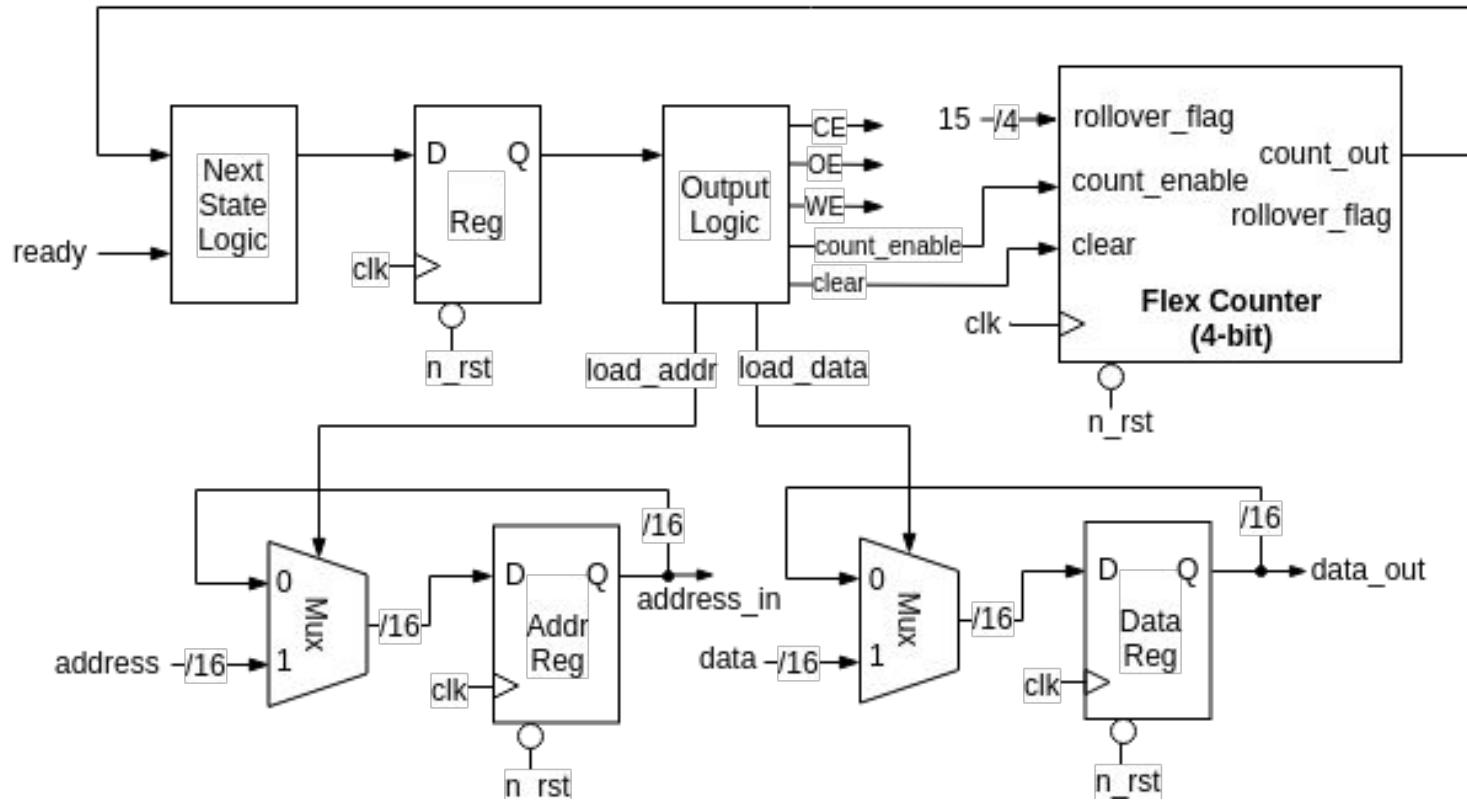
**SPI Output Controller state diagram**

state	send_digit	send_cost
IDLE	1	0
WAIT_COST	0	1
COST_SENT	0	1
LOAD_DIG	1	0
DIG_SENT	0	0

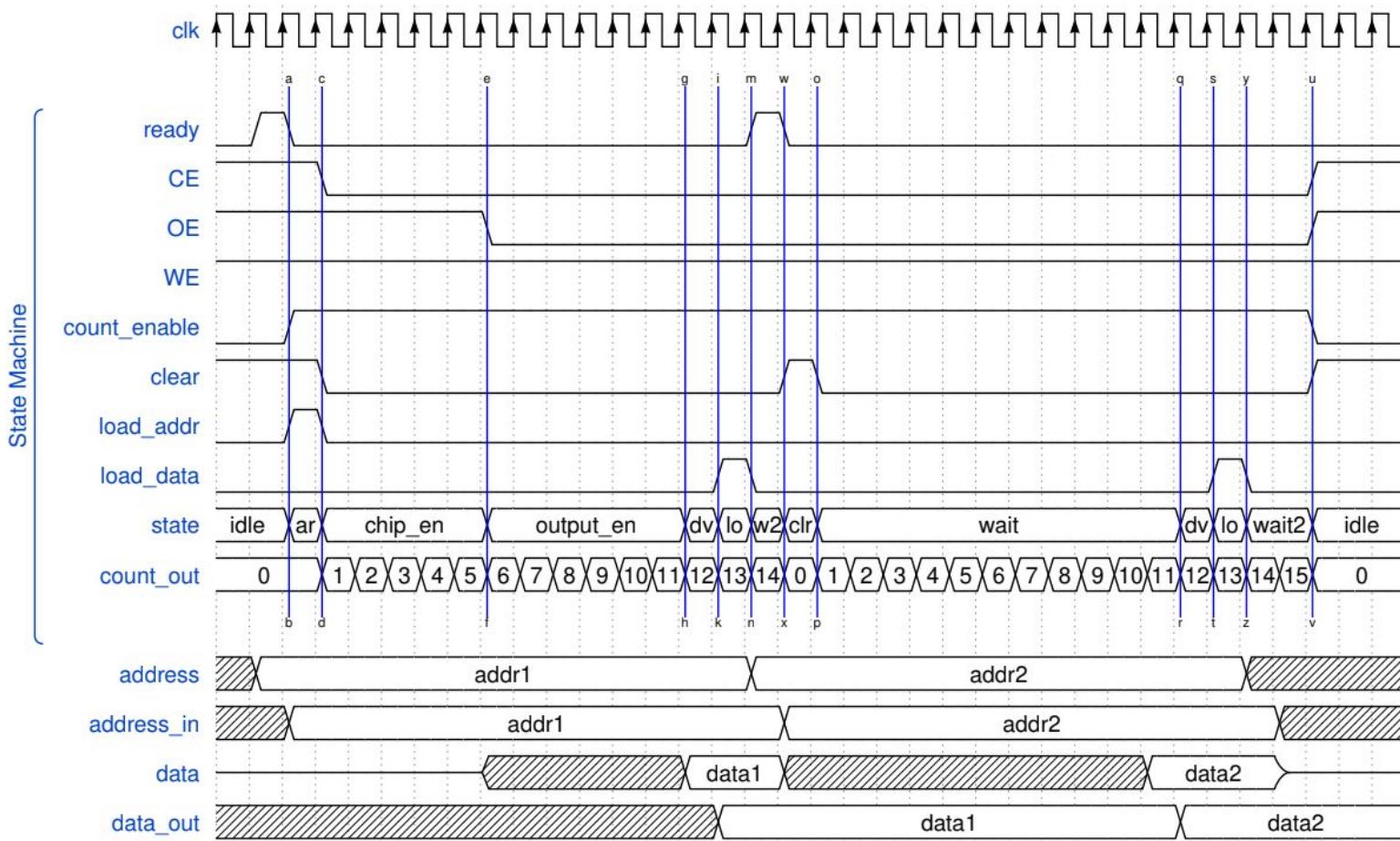
## SPI Output Controller output logic



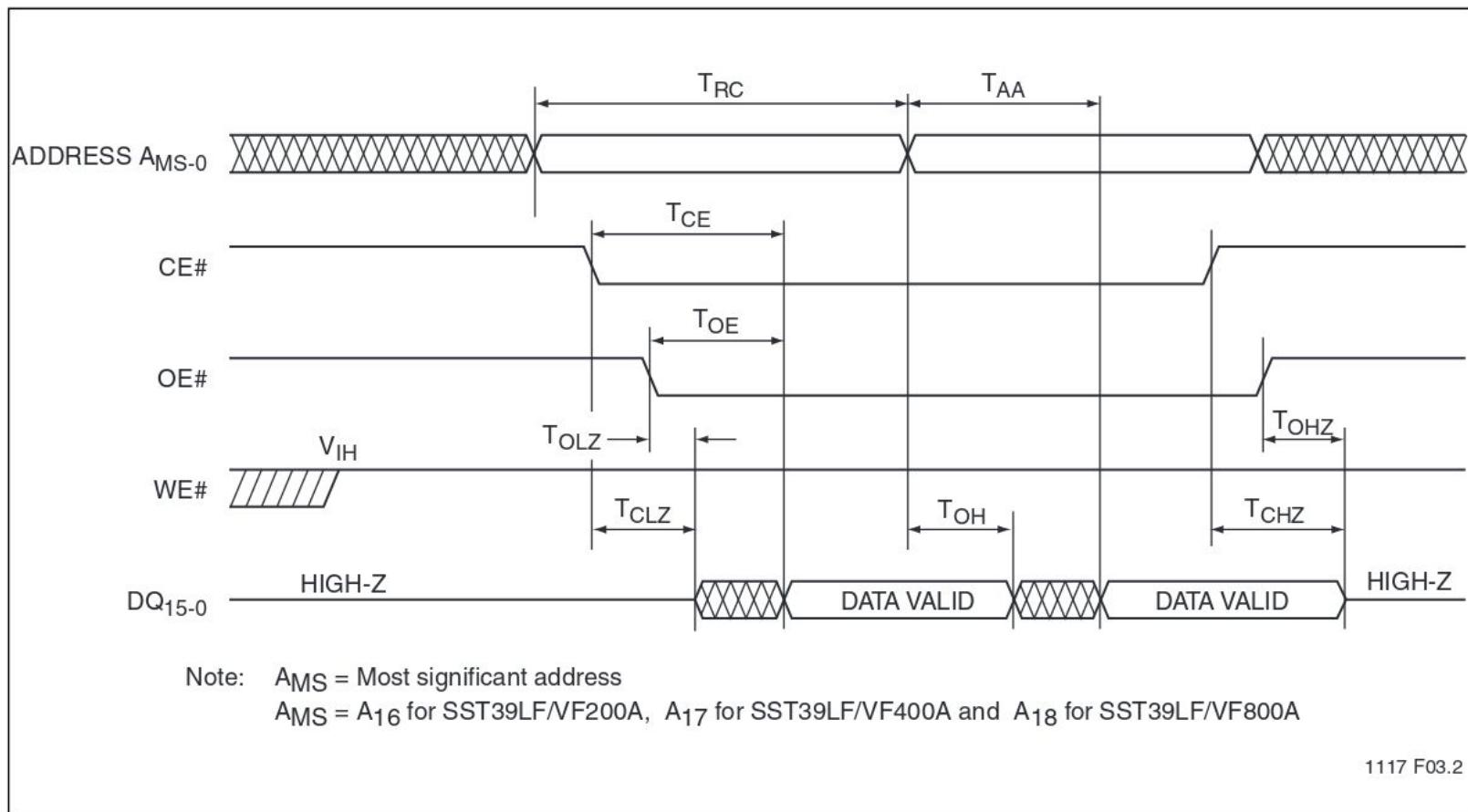
## SPI Output Controller timing diagram



## Flash Memory Controller RTL Diagram



## Flash Memory Controller timing diagram



**Figure 5:** Read Cycle Timing Diagram

**Timing diagram for flash memory [SST39LF200A] interfacing with neural network.**

**Device allows for 55ns access time per word of data**

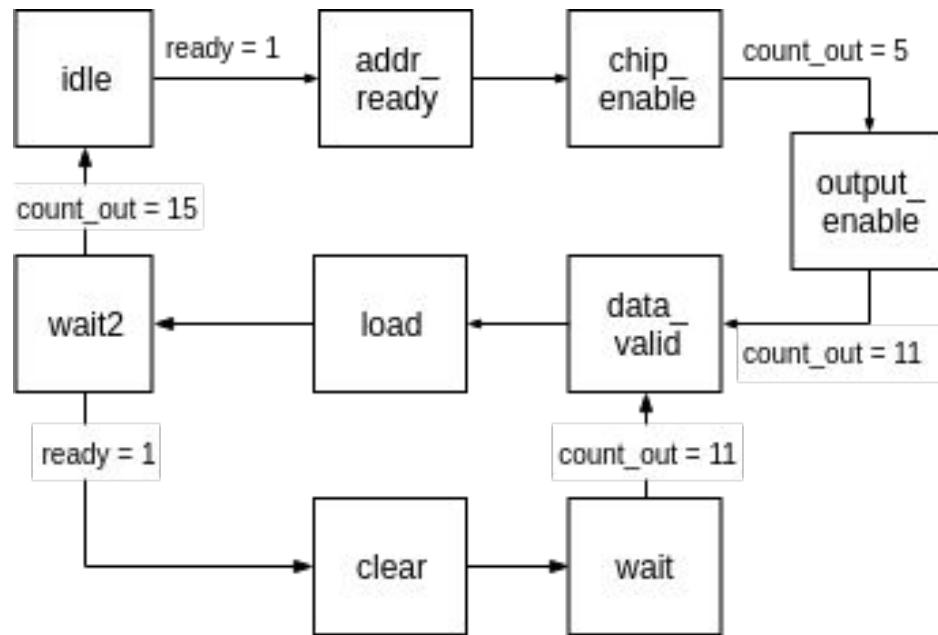
**Table 15:** Read Cycle Timing Parameters  $V_{DD} = 3.0\text{-}3.6V$

Symbol	Parameter	SST39LF200A/400A/800A-55		Units
		Min	Max	
$T_{RC}$	Read Cycle Time	55		ns
$T_{CE}$	Chip Enable Access Time		55	ns
$T_{AA}$	Address Access Time		55	ns
$T_{OE}$	Output Enable Access Time		30	ns
$T_{CLZ}^1$	CE# Low to Active Output	0		ns
$T_{OLZ}^1$	OE# Low to Active Output	0		ns
$T_{CHZ}^1$	CE# High to High-Z Output		15	ns
$T_{OHZ}^1$	OE# High to High-Z Output		15	ns
$T_{OH}^1$	Output Hold from Address Change	0		ns

T15.7 25001

1. This parameter is measured only for initial qualification and after a design or process change that could affect this parameter.

Timing parameters for flash memory [SST39LF200A] interfacing with neural network.



## Flash Memory Controller State Diagram Diagram

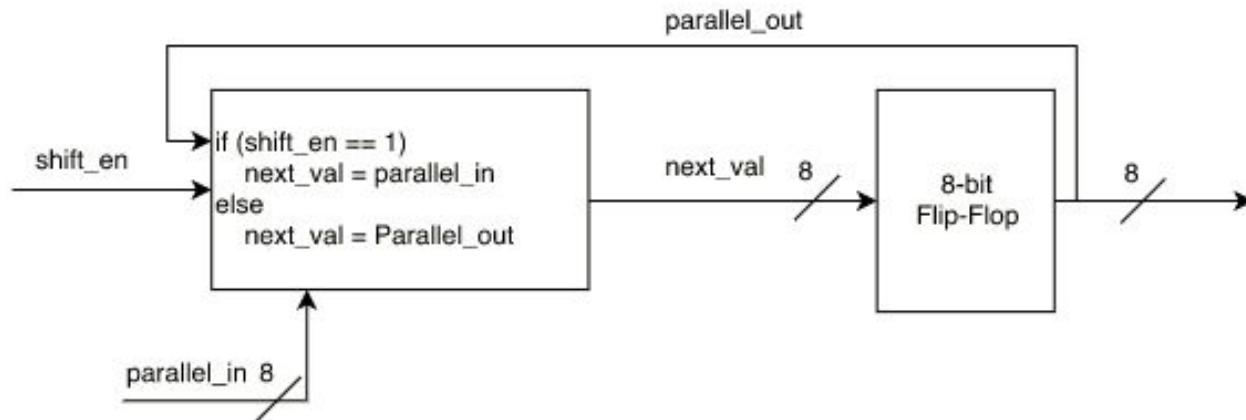


Figure 18: PTP shift register RTL diagram

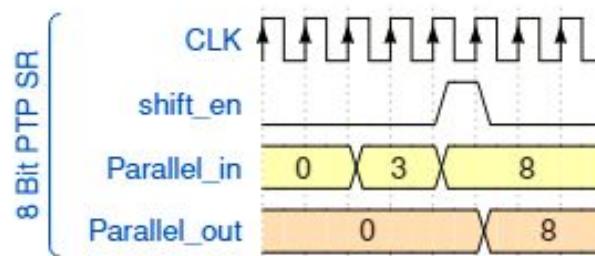


Figure 19: PTP shift register timing diagram

## PTPSR Diagrams

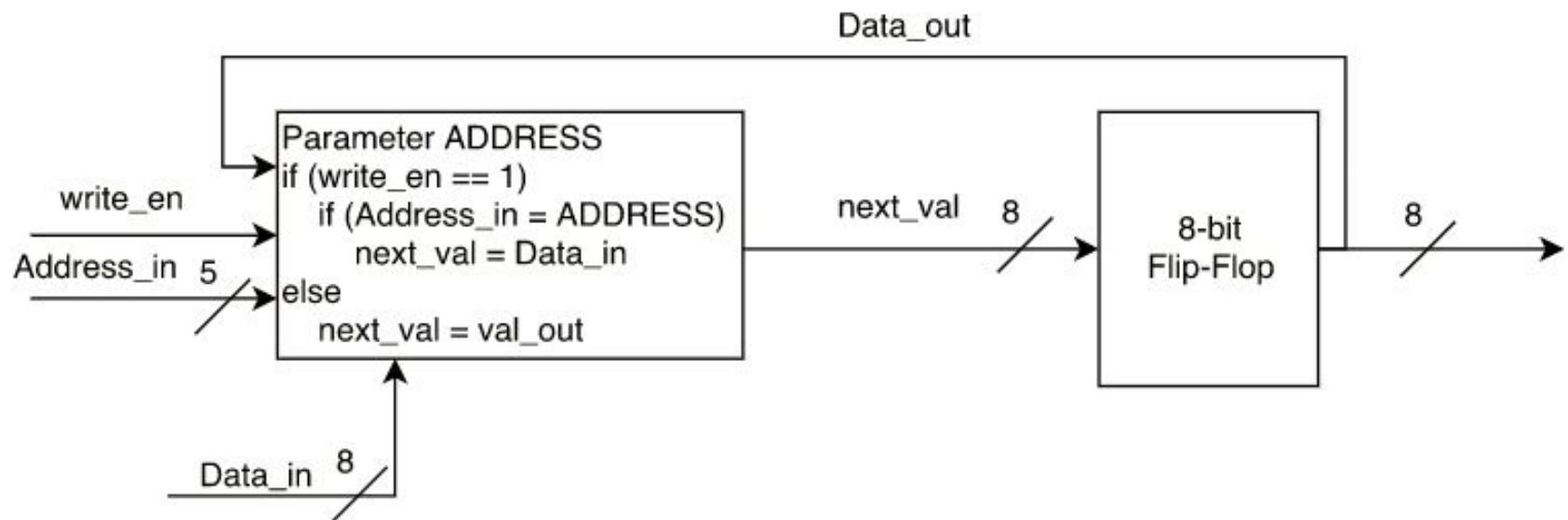


Figure 20: Addressable Register RTL diagram

## Addressable Register RTL

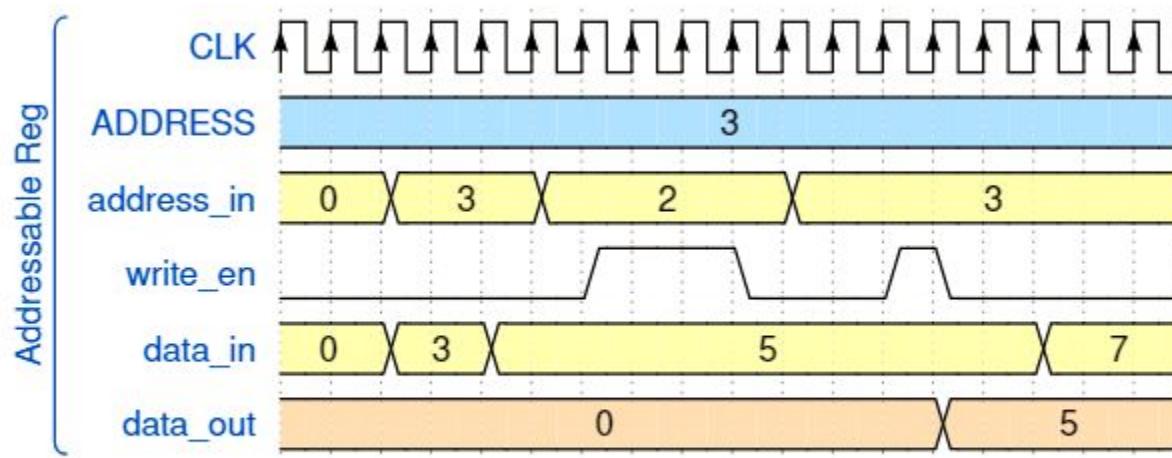


Figure 21: Addressable Register timing diagram

## Addressable Register timing diagram

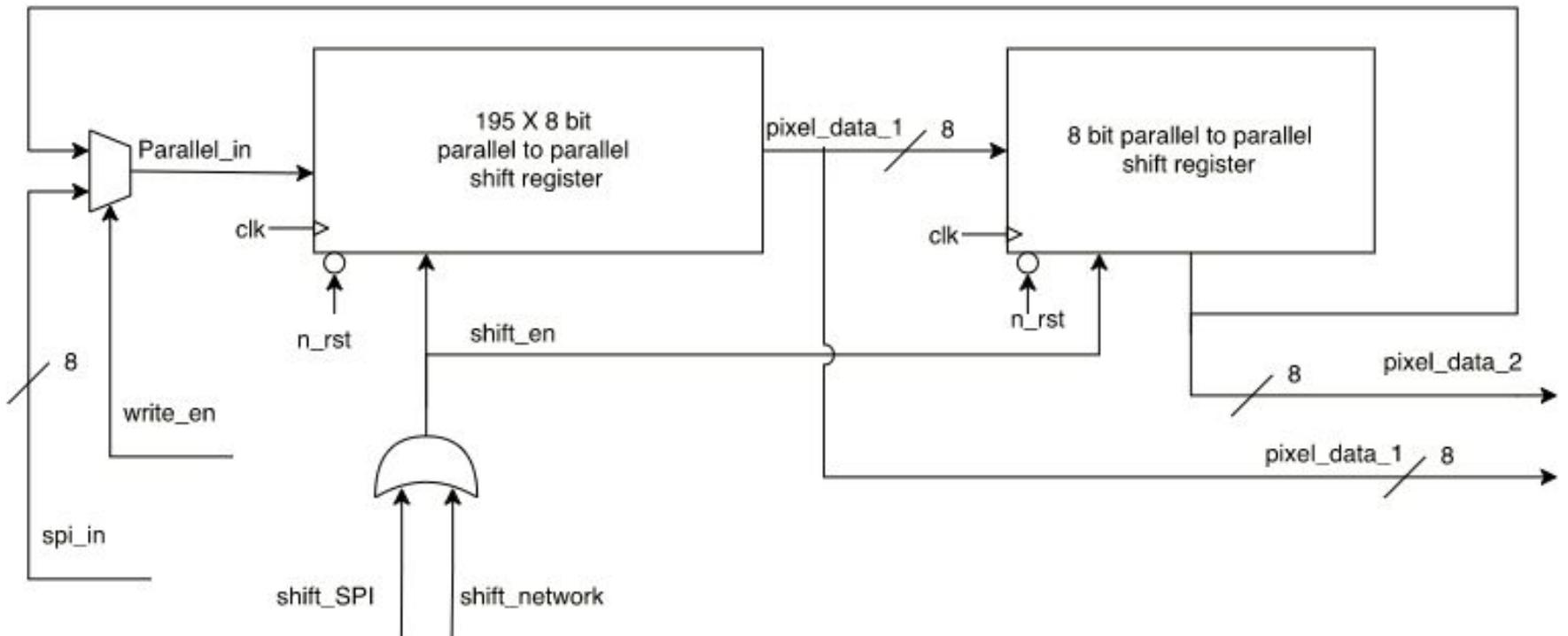


Figure 28: Pixel Data Register RTL diagram

## Pixel Data Register RTL

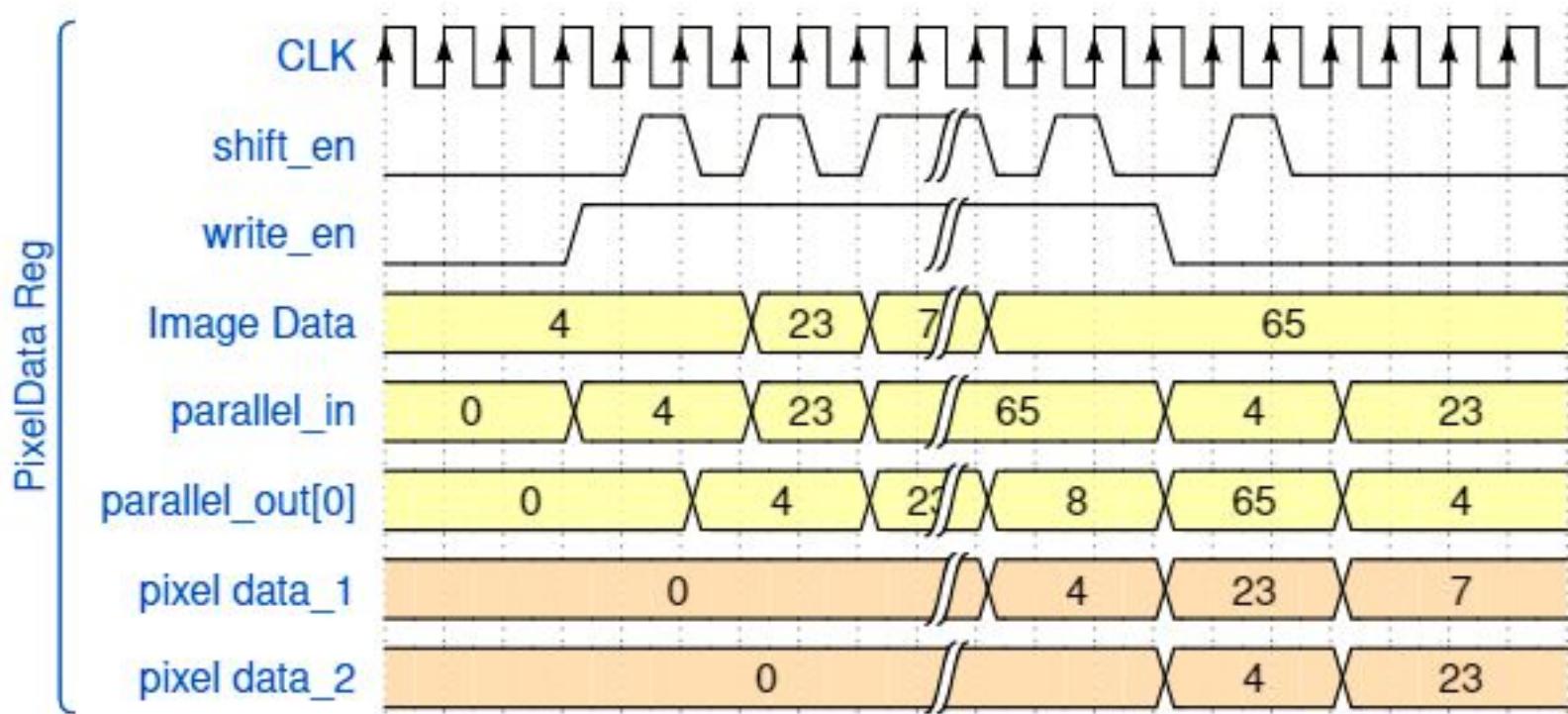
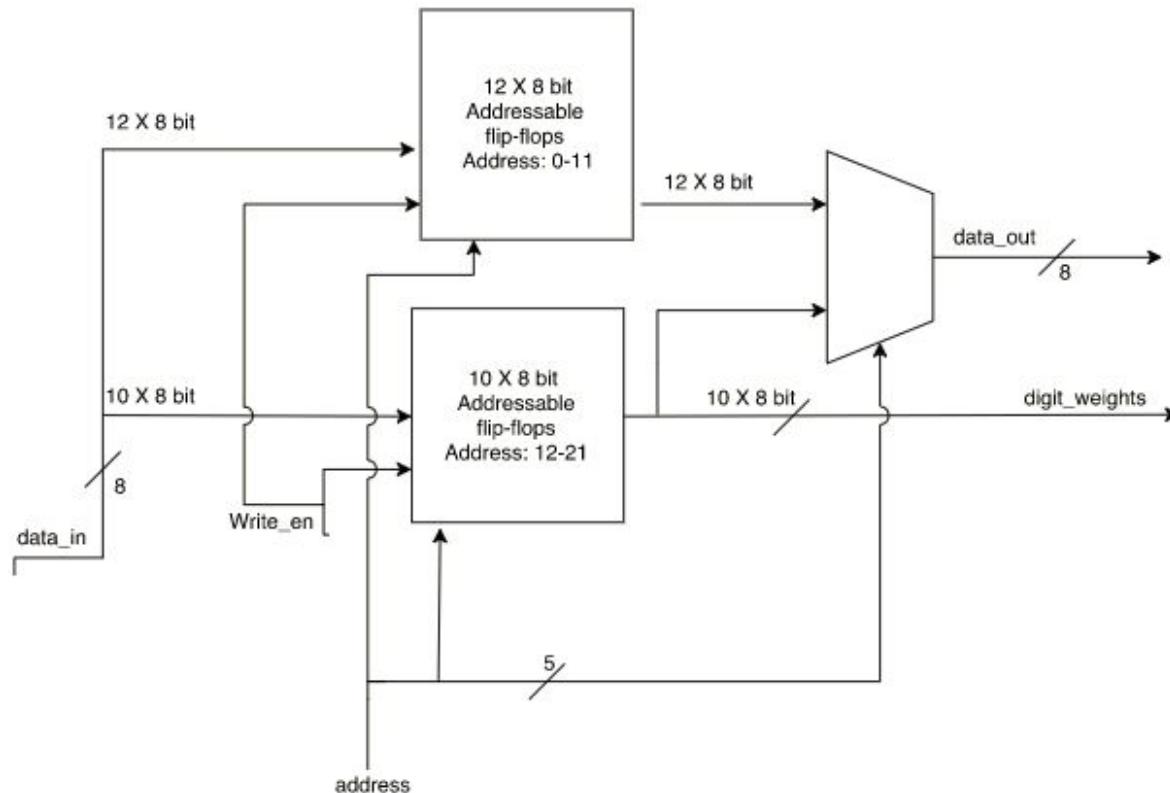


Figure 29: Pixel Data Register timing diagram

## Pixel Data Register timing diagram



*Figure 30: Sigmoid Register RTL diagram*

## Sigmoid Register RTL

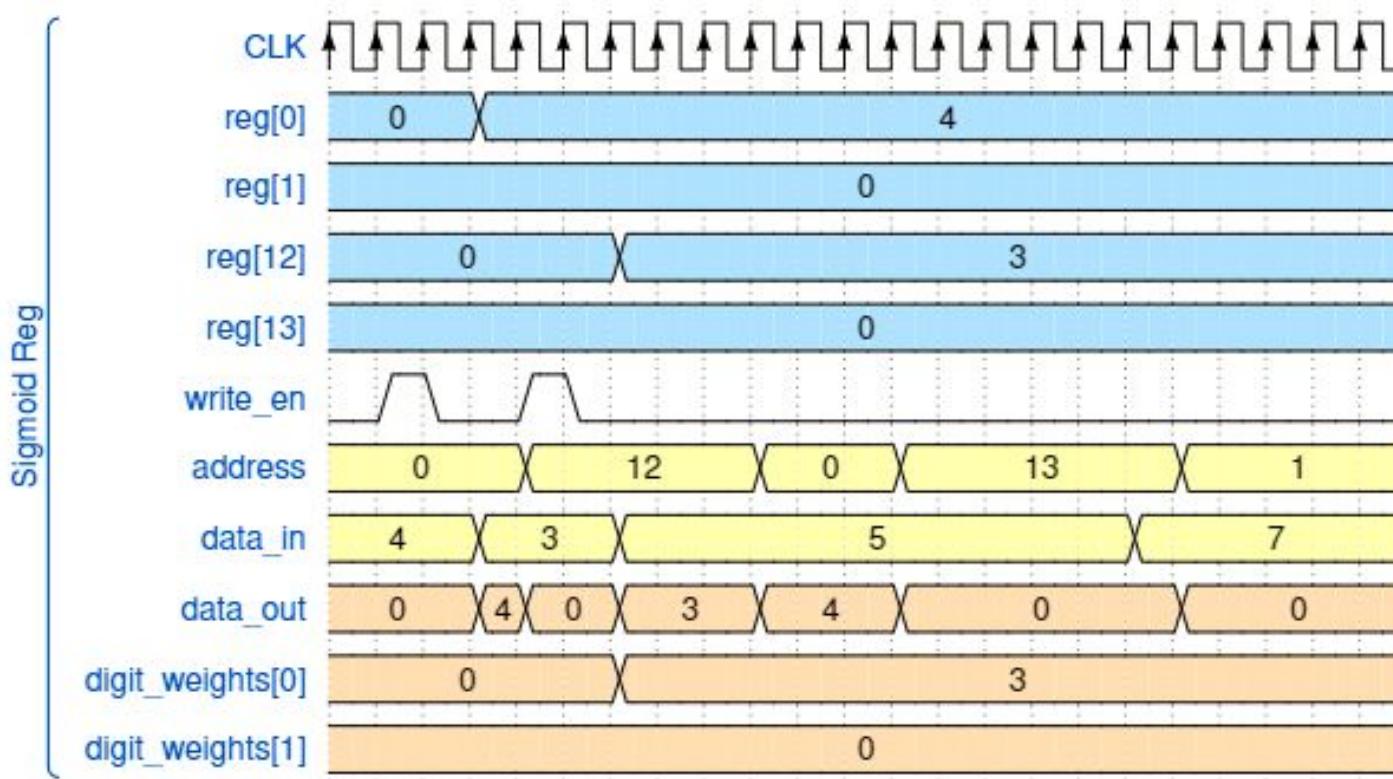


Figure 31: Sigmoid Register Timing diagram

## Sigmoid Register Timing Diagram

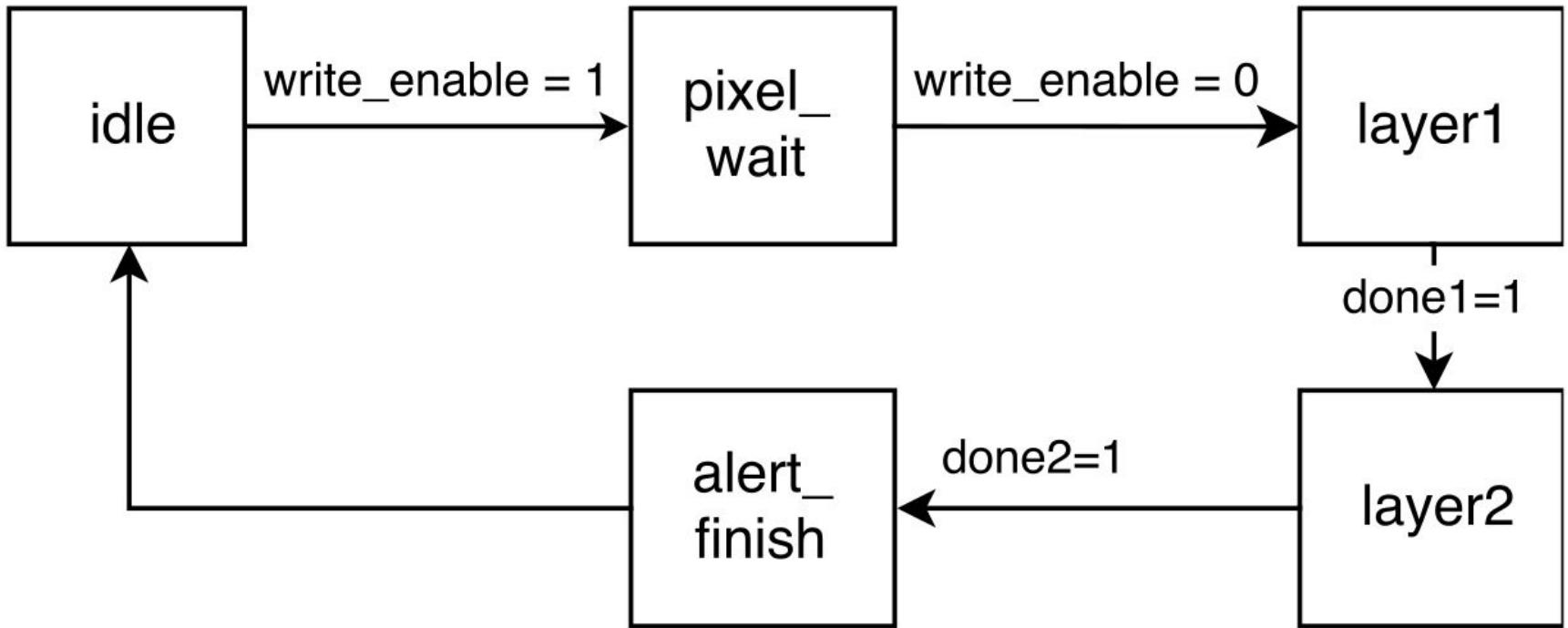


Figure 32: Network Controller top level state diagram

Network Controller top level state diagram

State	Input Rollover Val	Neuron Rollover Val	network_done	data_ready
idle	0	0	0	1
pixel_wait	0	0	0	0
layer1	35	7	0	0
layer2	1	9	0	0
alert_finish	0	0	1	0

(a) Network Controller top level Output Logic

## Network Controller top level output logic

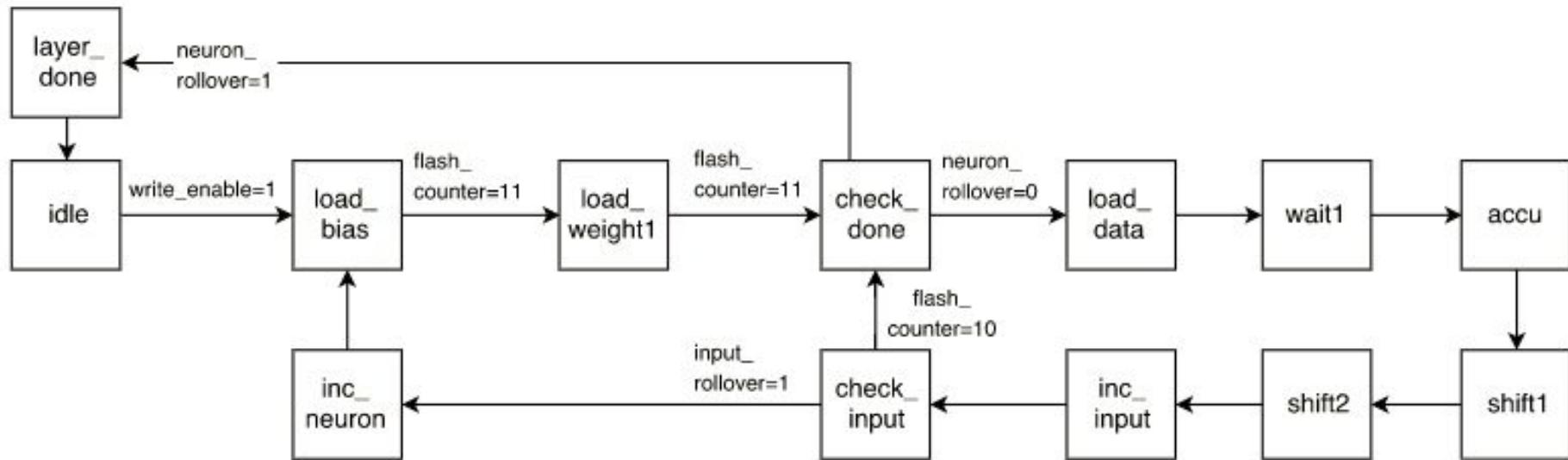


Figure 33: Network Controller layer 1 state diagram

## Network Controller Layer 1 state diagram

State	input_en	weight_en	bias_en	shift	sig_write	ready	accum.	clear
idle	0	0	0	0	0	0	1	0
load_bias	0	0	1	0	0	1	1	1
load_weight1	0	1	0	0	0	1	1	0
check_done	0	0	0	0	0	0	1	0
load_data	1	0	0	0	0	1	1	0
wait1	0	0	0	0	0	0	1	0
accu	0	0	0	0	0	0	0	0
shift1	0	0	0	1	0	0	1	0
shift2	0	0	0	1	0	0	1	0
inc_input	0	0	0	0	0	0	1	0
check_input	0	0	0	0	0	0	1	0
inc_neuron	0	0	0	0	1	0	1	0
layer_done	0	0	0	0	0	0	1	0

(b) Network Controller layer 1 Output Logic

## Network controller Layer 1 output logic

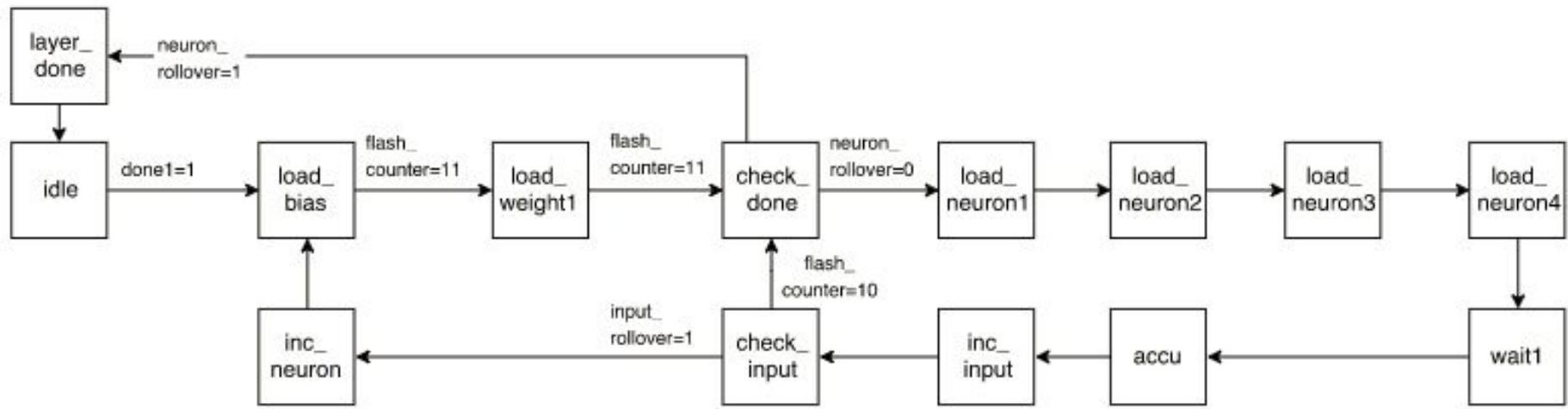


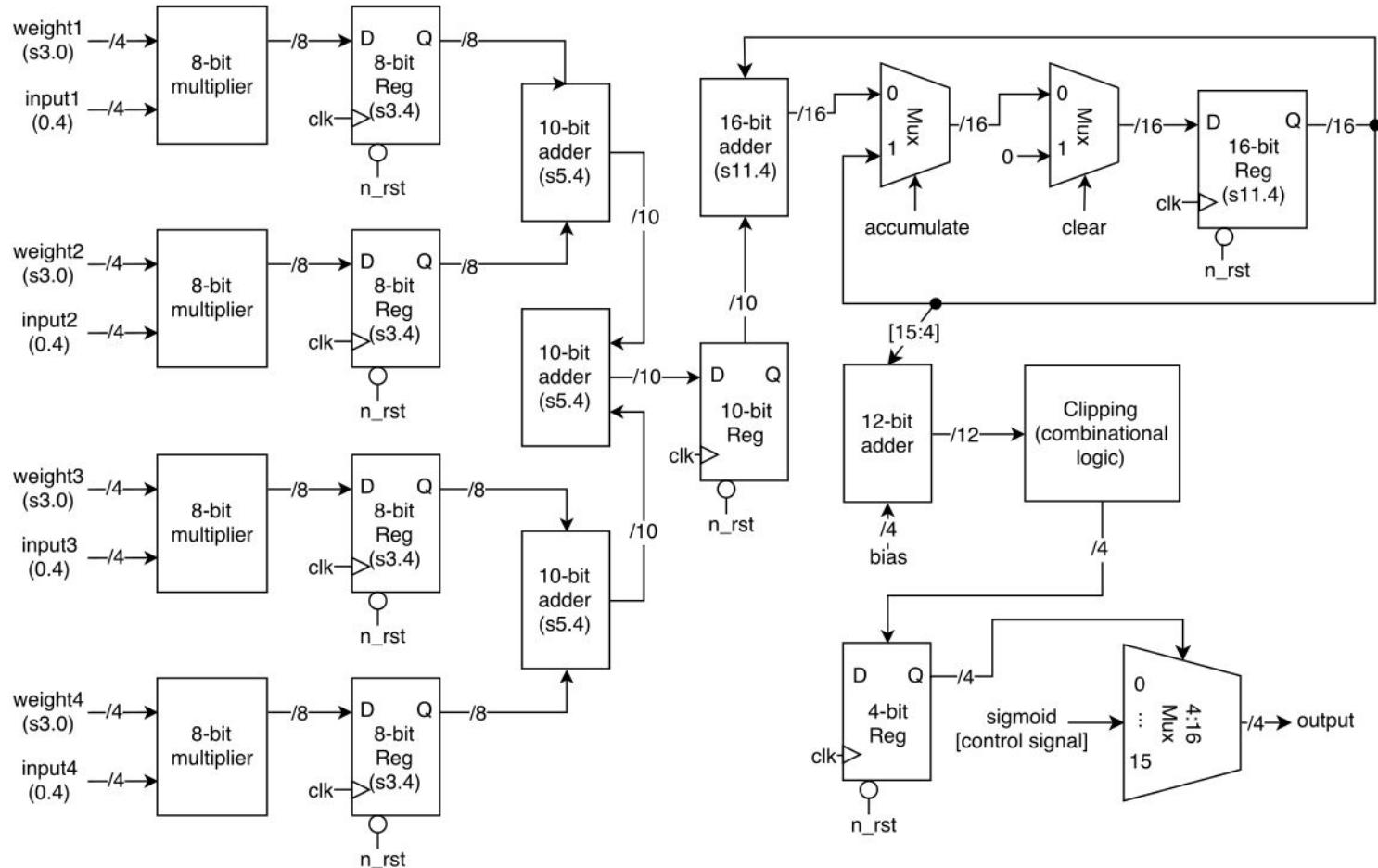
Figure 34: Network Controller layer 2 state diagram

## Network Controller Layer 2 state diagram

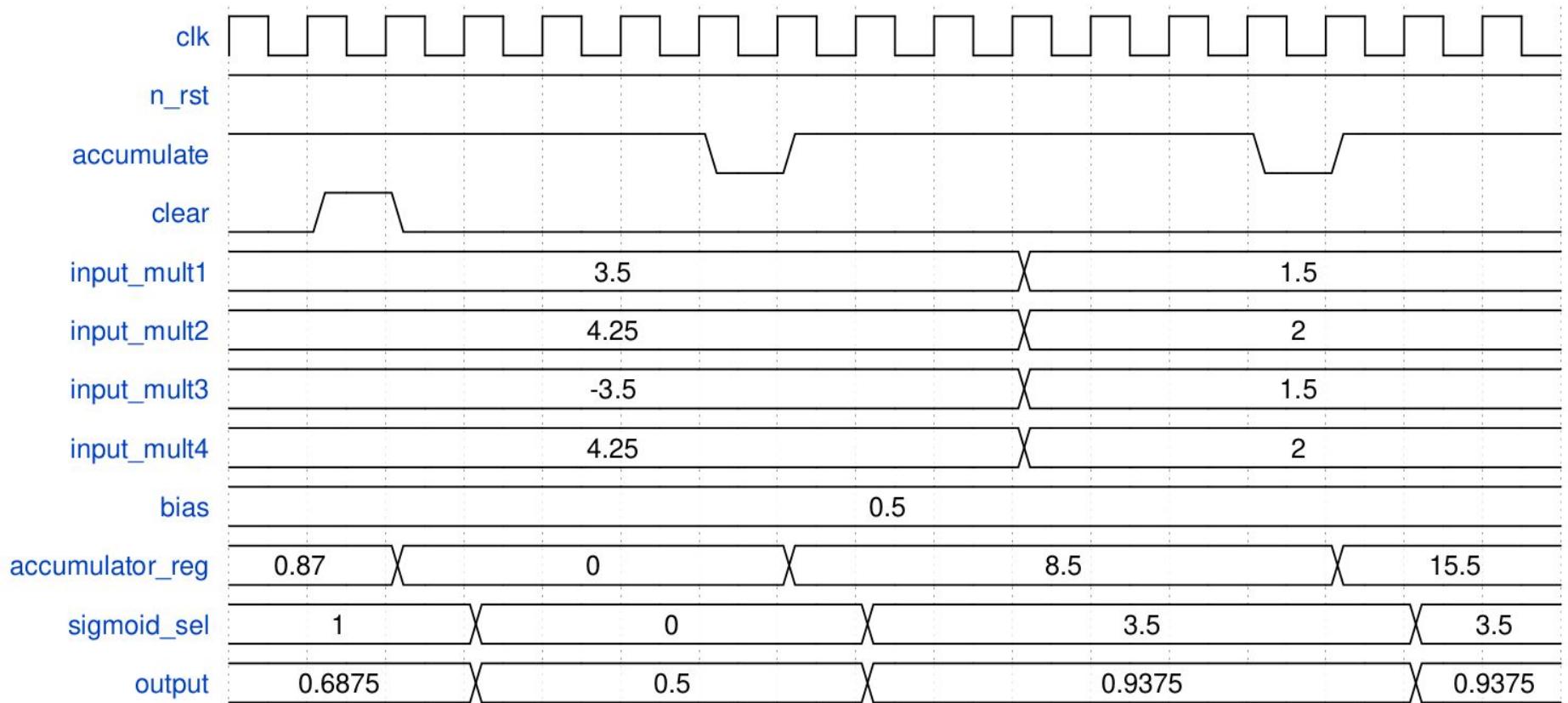
State	input_en	weight_en	bias_en	sig_write	ready	accum.	clear
idle	0	0	0	0	0	1	0
load_bias	0	0	1	0	1	1	1
load_weight1	0	1	0	0	1	1	0
check_done	0	0	0	0	0	1	0
load_neuron1	1	0	0	0	0	1	0
load_neuron2	1	0	0	0	0	1	0
wait1	0	0	0	0	0	1	0
accu	0	0	0	0	0	0	0
inc_input	0	0	0	0	0	1	0
check_input	0	0	0	0	0	1	0
inc_neuron	0	0	0	1	0	1	0
layer_done	0	0	0	0	0	1	0

(c) Network Controller layer 2 Output Logic

## Network controller Layer 2 output logic



## Sigmoid ALU block diagram



## Sigmoid ALU timing diagram

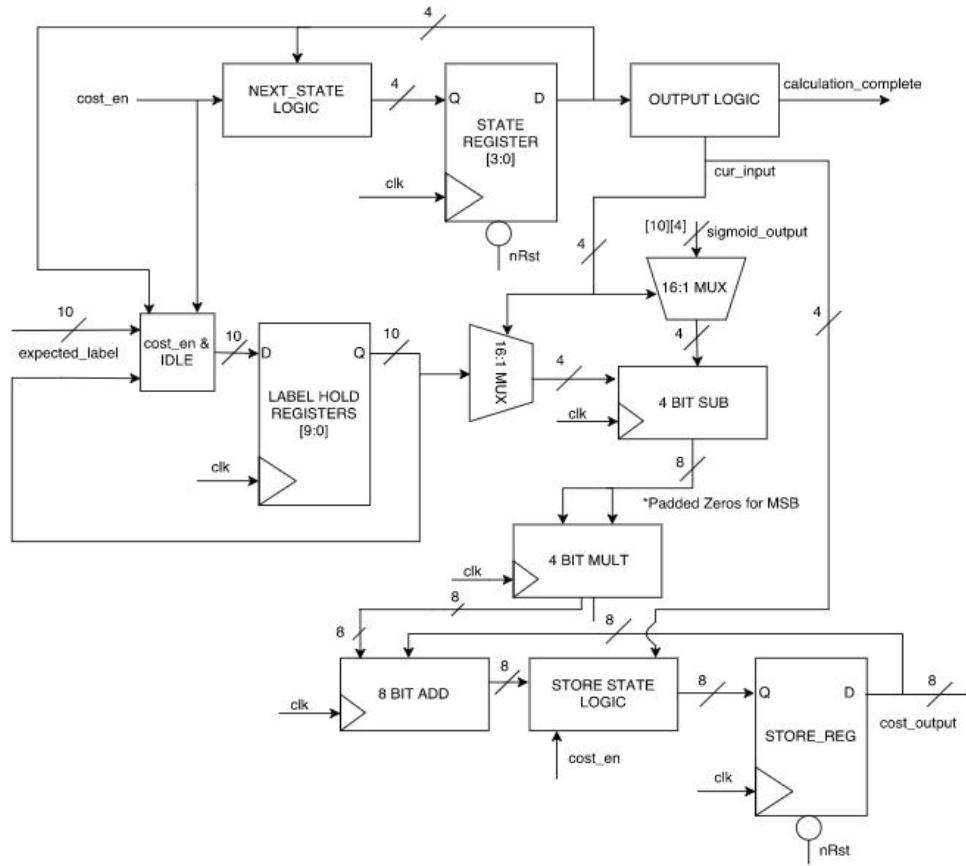
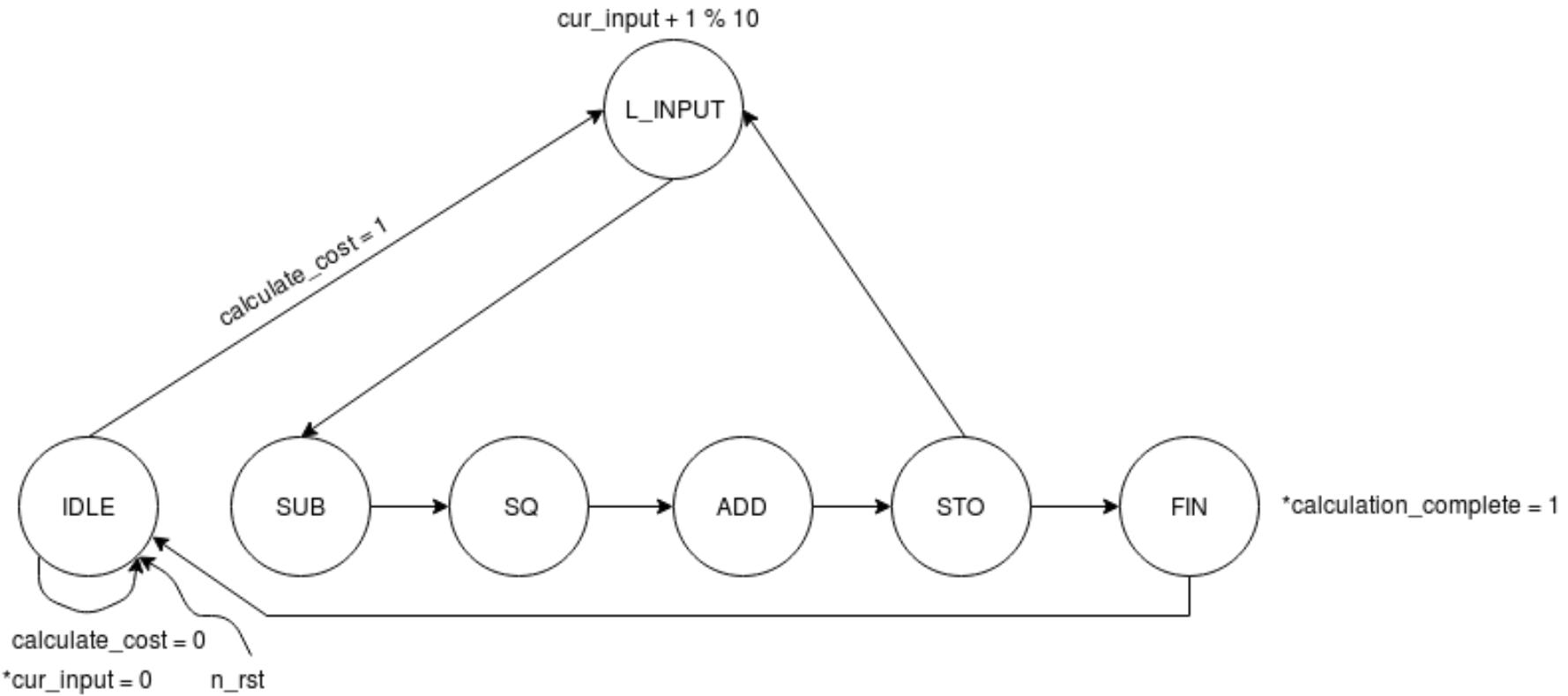
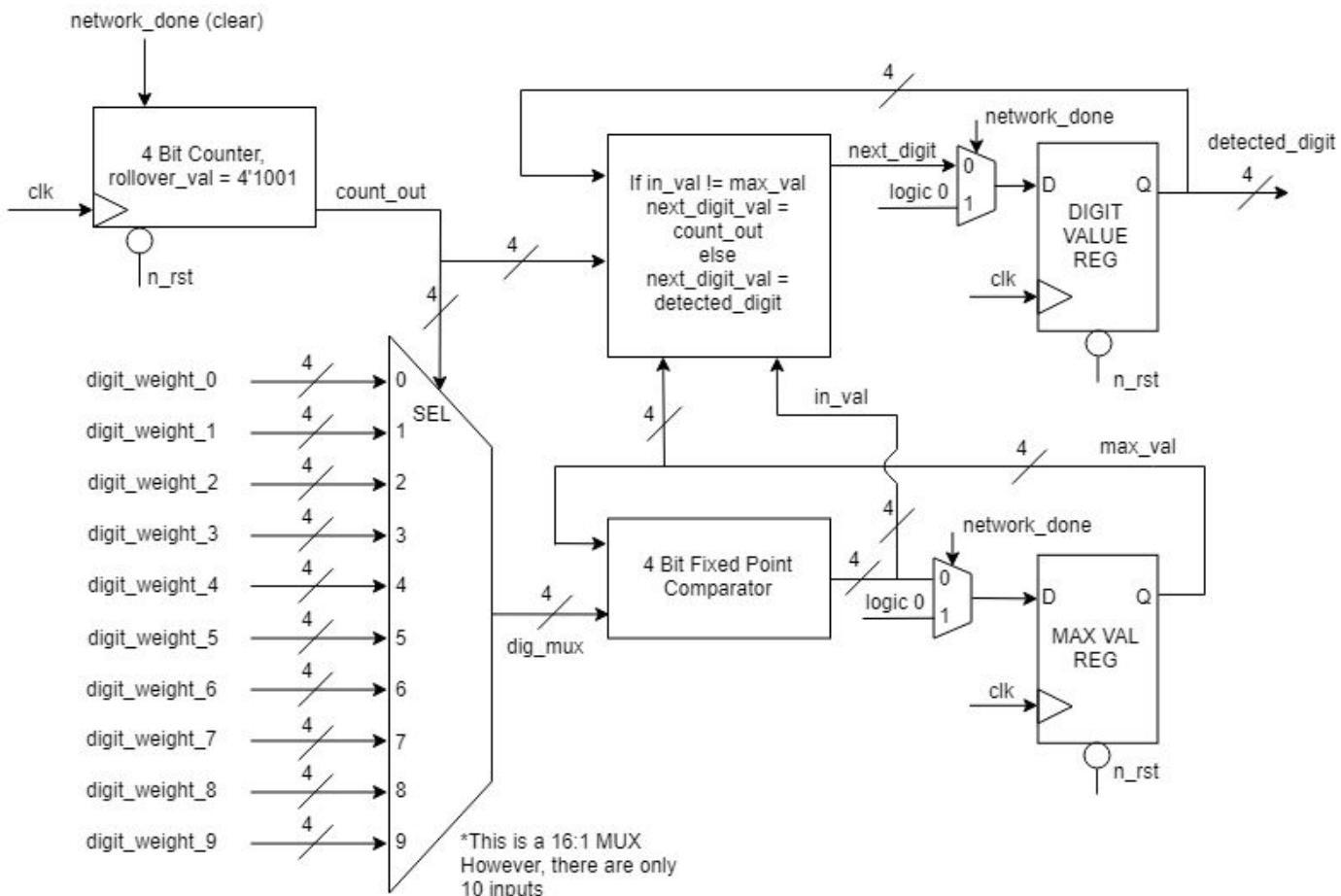


Figure 40: RTL Diagram for Implementing the Cost Function Block

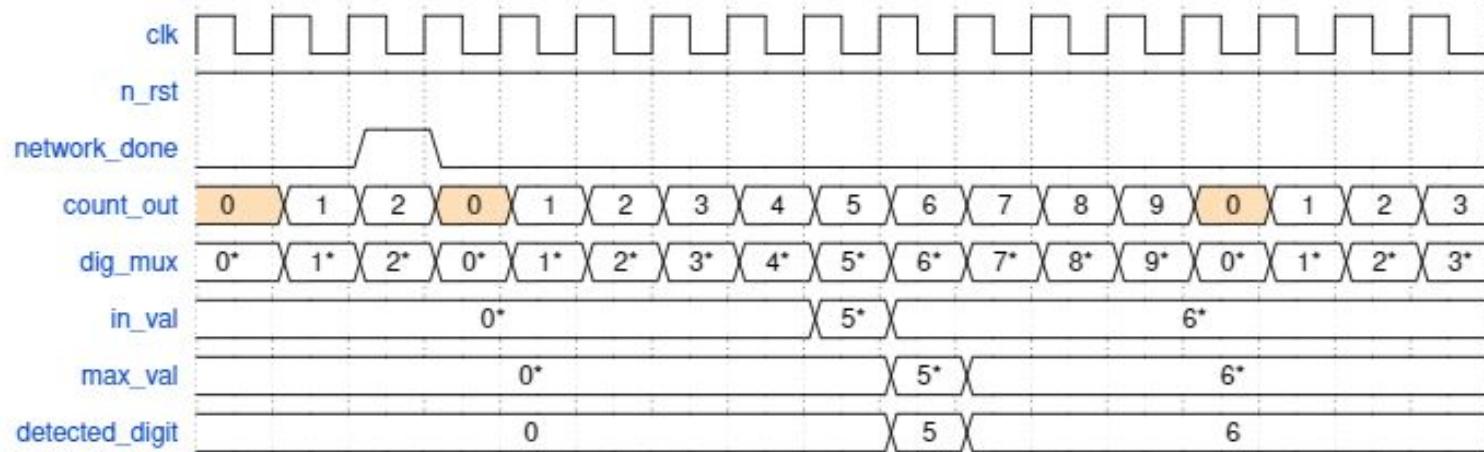
## Cost Calculator RTL Diagram



## Cost Calculator State Diagram #1



## Digit Decode RTL Diagram



example diagram where input vector is [0, 0, 0, 0, 0, 0.1, 0.8, 0.1, 0, 0], X\* represents the value at index X

*Figure 44: Timing Diagram for Implementing the Detected Digit Block*

## Digit Decode Timing Diagram