CNN FPGA System Architecture Review

System Overview

- Multi-model CNN implementation (10 models)
- 16x12 image input processing
- Target frequency: 73MHz (non-pipelined) / 51MHz (pipelined)
- 16 parallel compute engines

Key Components & Architecture

1. Processing Core:
    - 16 Parallel Compute Engines
    - Each engine: 7 DSP blocks (3x 9x9 multipliers, 4x 19x8 multipliers) total of 112 DSP to run one colum at once.
    - The result of the MultiMultiplierEngine will be ready by the next CLk
    - ReLU activation implementation
    - Fully connected layer processing

2. Memory Architecture:
    - CONV_W_MEM: Convolution weights [15:0] × N
    - CONV_B_MEM: Convolution bias [15:0] × N
    - WFC_MEM: FC weights [127:0] × 3 × N
    - BFC_MEM: FC bias [44:0] × N
    - Relu_out_FIFO [18:0] X128 X3 X N
    - PIxel_MEM int8[15:0] * MAX_W

3. Data Management:
    - Triple buffer design for continuous processing of each line
    - Optimized memory controller
    - Efficient address management
    - Pipeline-optimized data flow
    - Correcly only the 2 line buffers and the FIFO need EN

Critical Parameters

- T_RAM: RAM request load time
- N: number of models
- T_find_winner: Winner calculation time
- MAX_W: the maximun lenght that an image can be we from correct testing the longest image was 350

Expected Performance for testing MultiMultiplierEngine

- Processing speed: Up to 73MHz
- Parallel model execution
- Continuous data flow capability

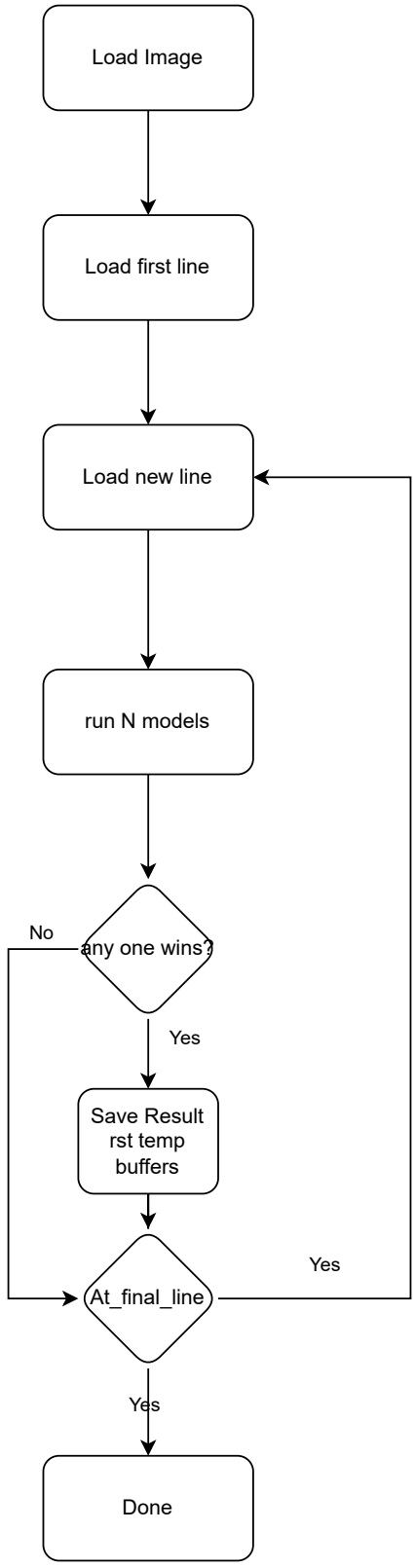
Implementation Considerations

1. Resource Requirements:
    - DSP blocks: 112 (7 × 16)
    - Memory banks: 6 main types

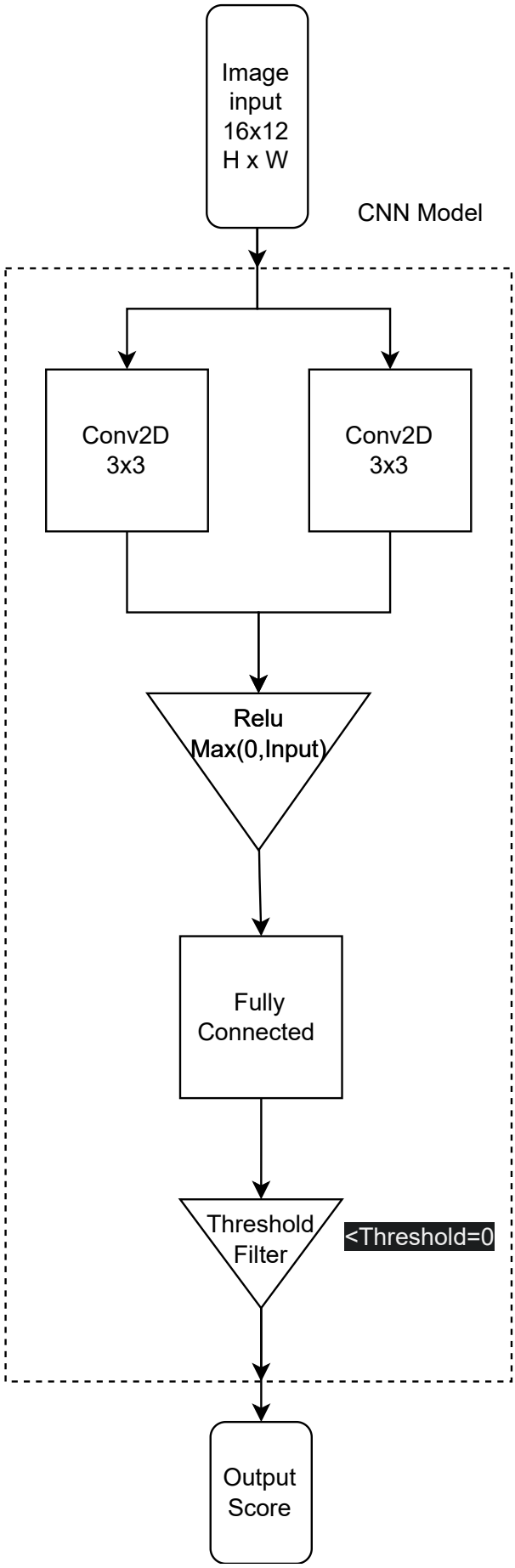
2. Critical Paths:
    - Memory access timing
    - Compute engine pipeline
    - Winner determination logic

3. Potential Challenges:
    - Memory bandwidth optimization
    - Pipeline synchronization
    - Resource balancing

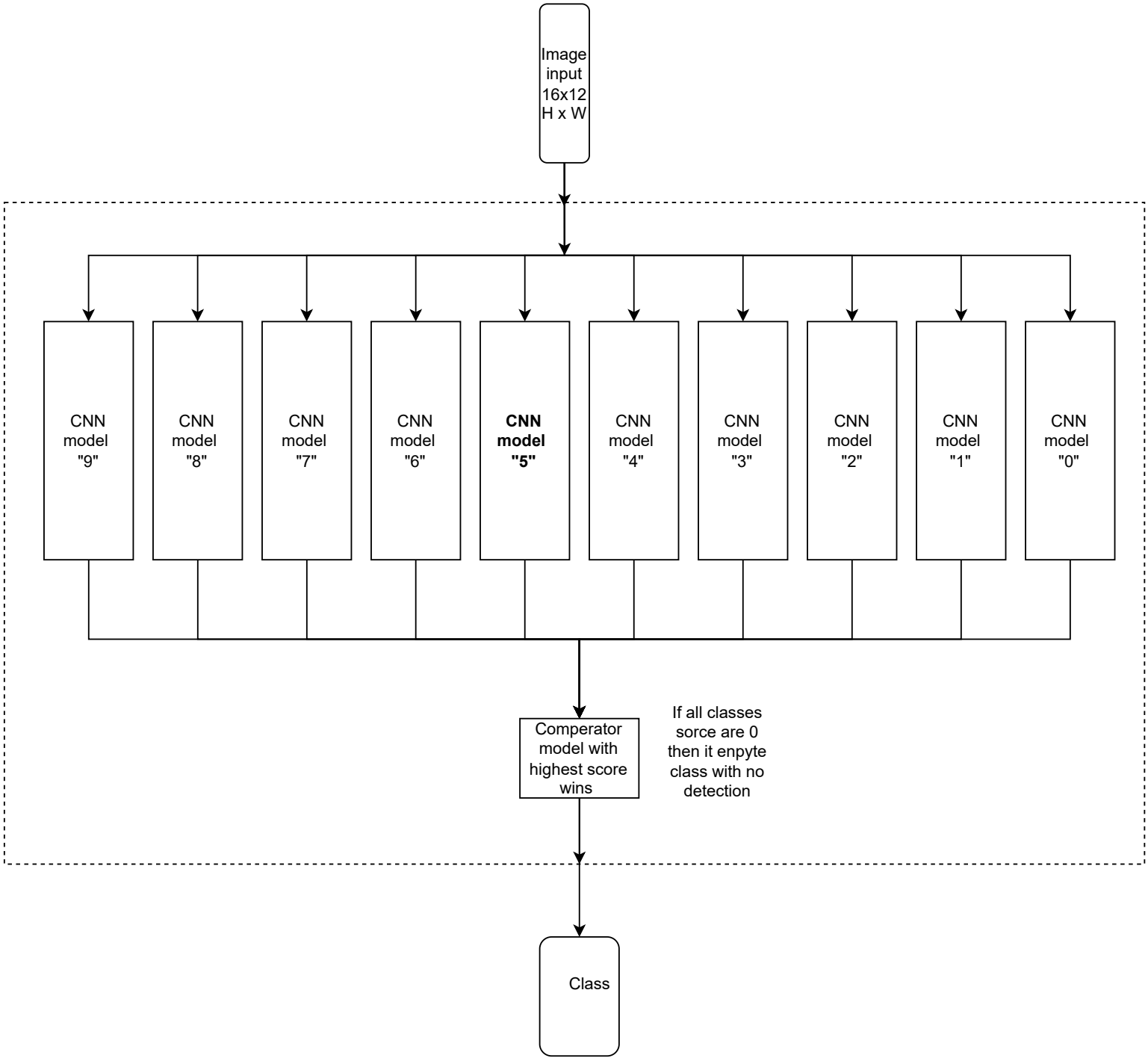
System WorkFlow



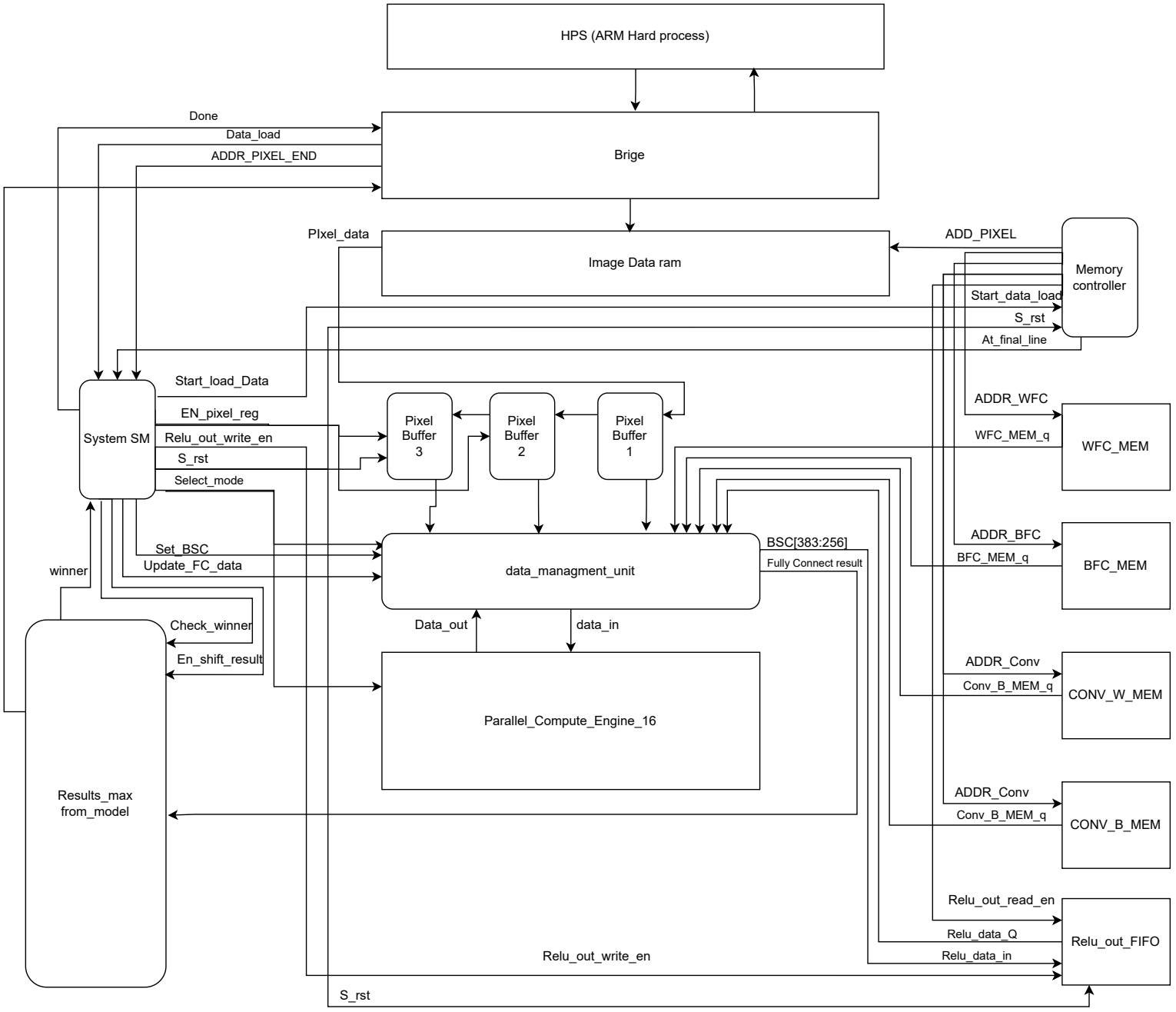
Single CNN netwok



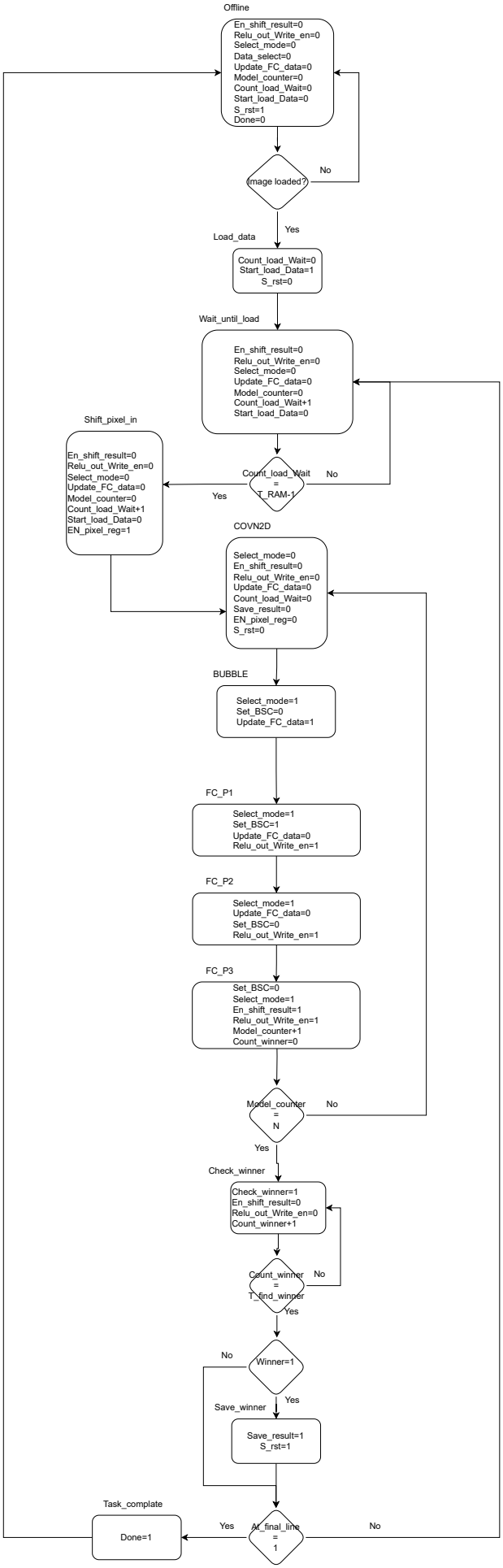
Full system CNN workflow



System planed Architecture

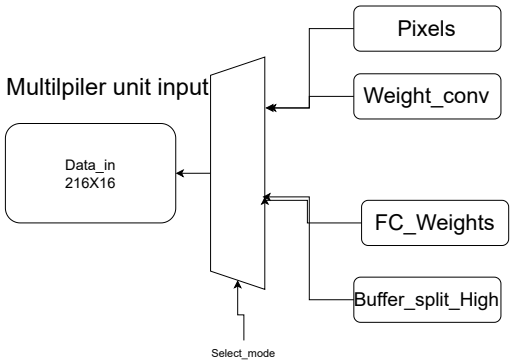


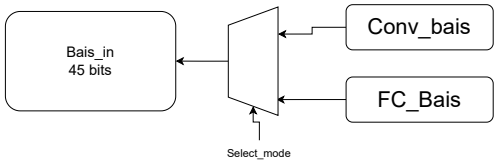
Main system SM

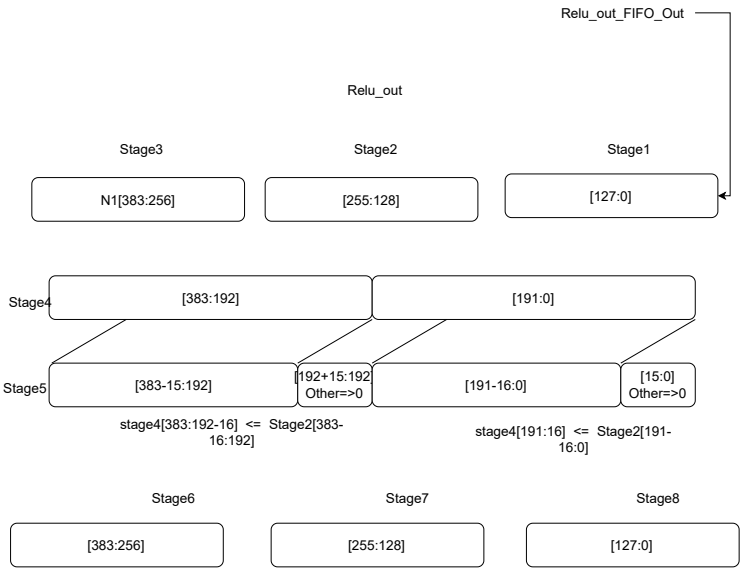


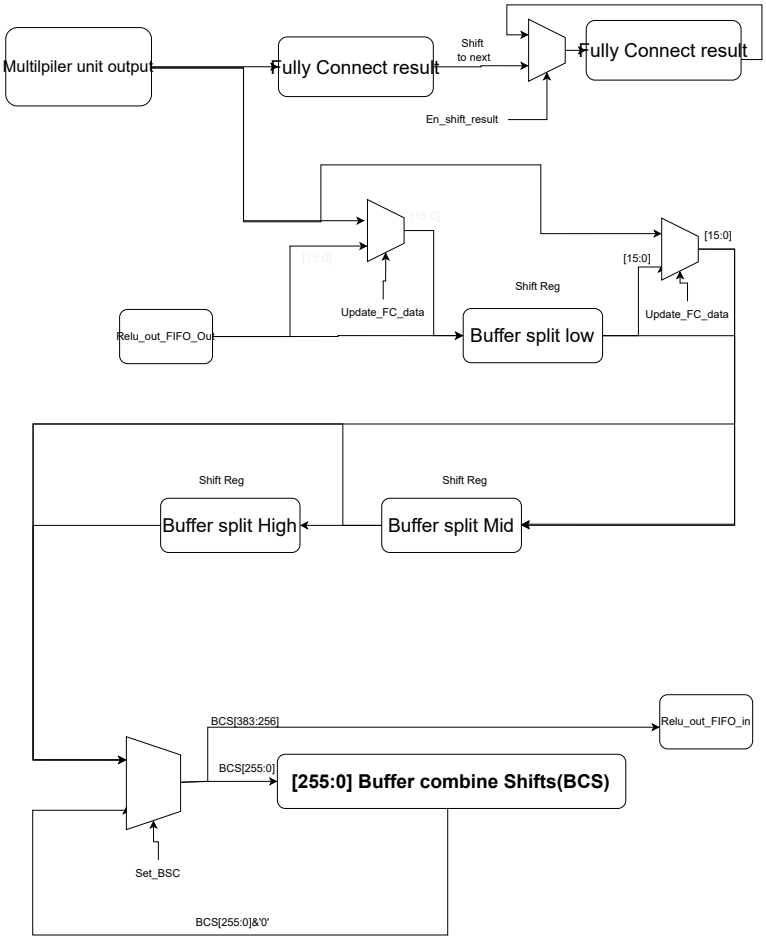
Data managment

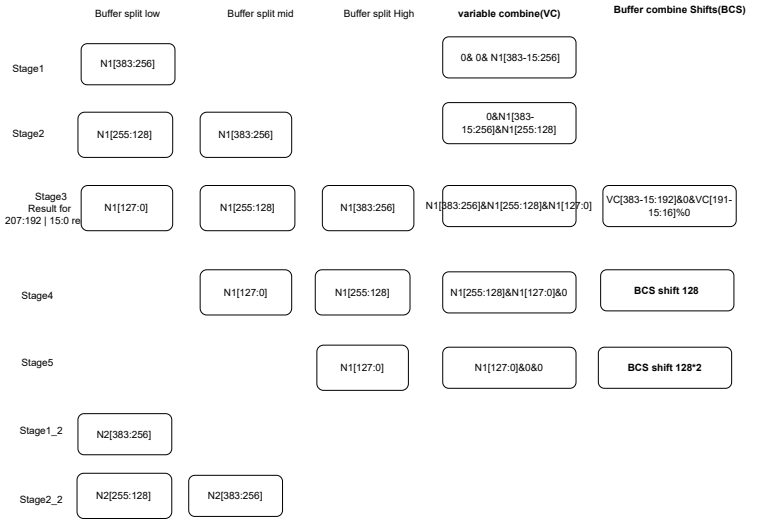
Input buffer load from
Memory controller and manages by Main system SM



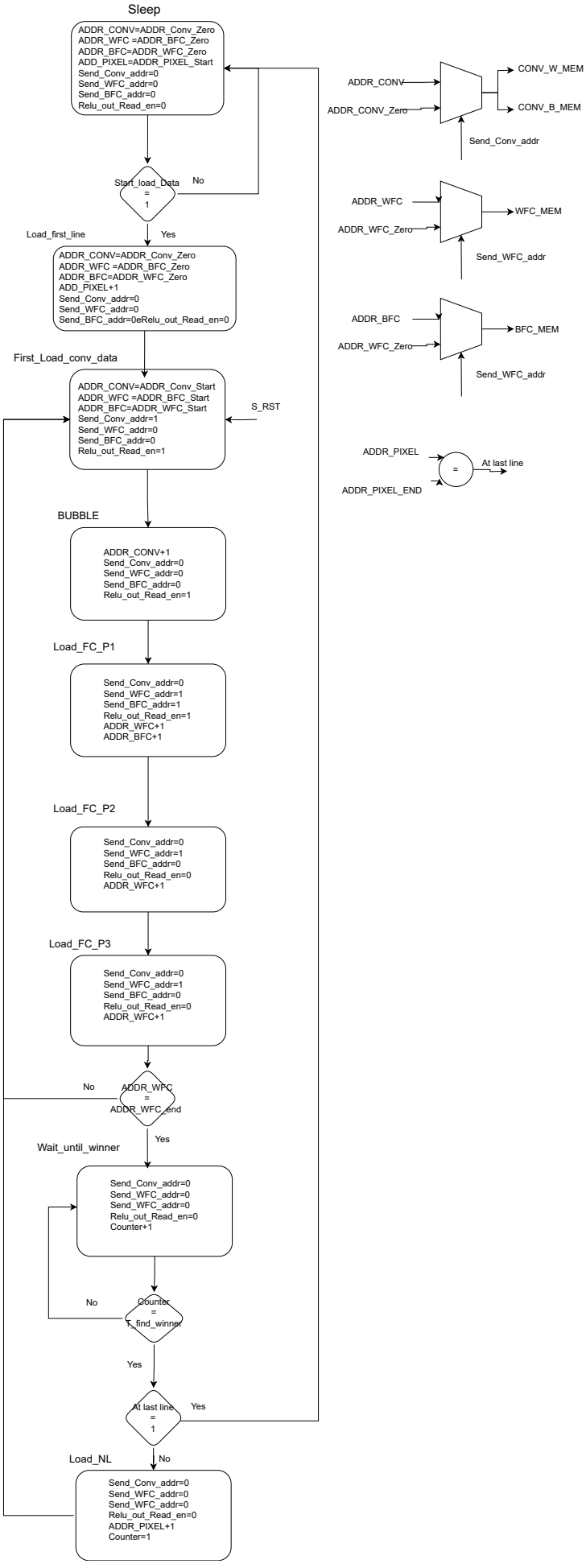




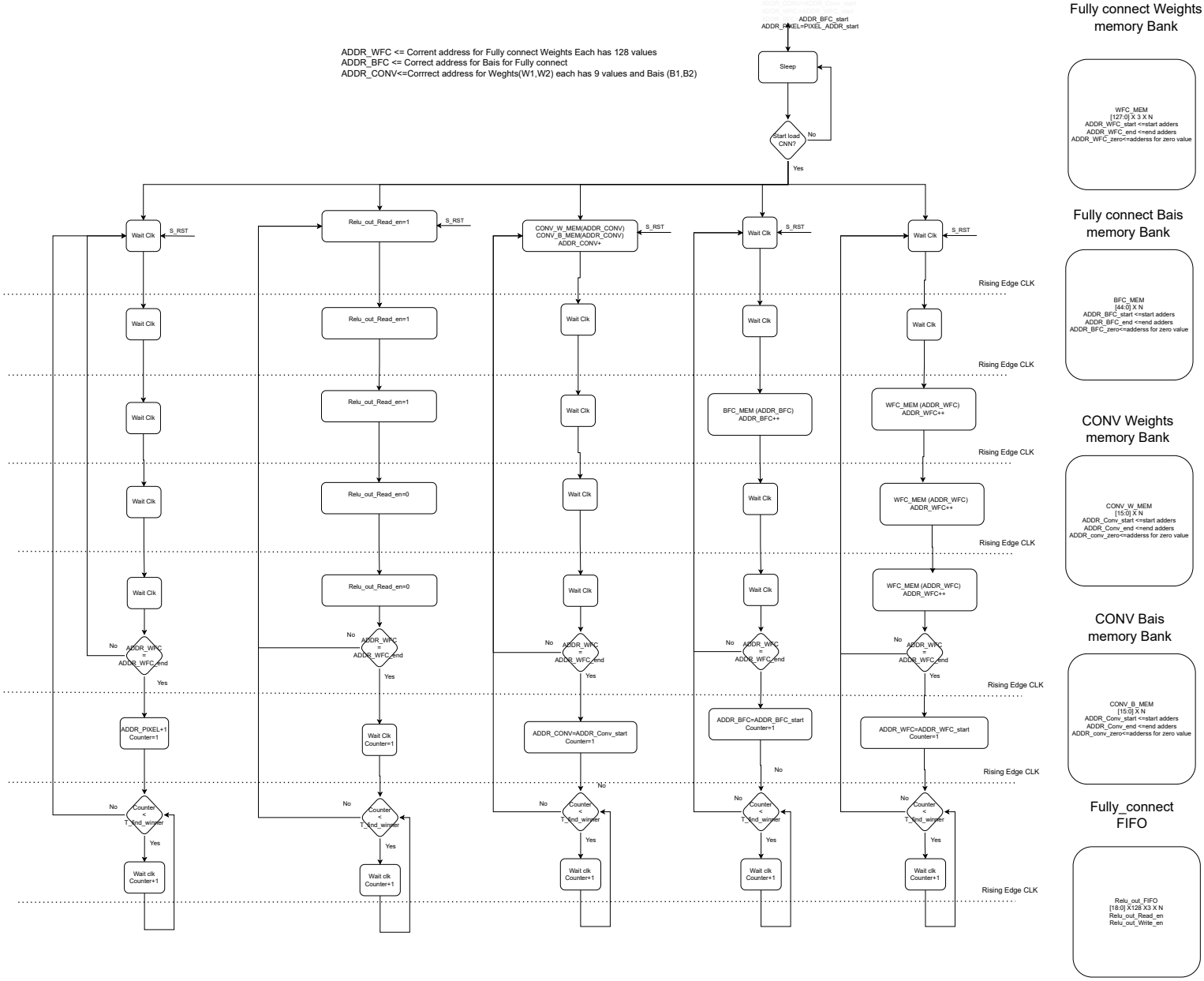




Memory Controller SM



Memory controller work flow



Fully connect Weights memory Bank

WFC_MEM
[127:0] X 3 X N
ADDR_WFC_start <=start adders
ADDR_WFC_end <=end adders
ADDR_WFC_zero<=adderss for zero value

Fully connect Bais memory Bank

BFC_MEM
[44:0] X N
ADDR_BFC_start <=start adders
ADDR_BFC_end <=end adders
ADDR_BFC_zero<=adderss for zero value

CONV Weights memory Bank

CONV_W_MEM
[15:0] X N
ADDR_Conv_start <=start adders
ADDR_Conv_end <=end adders
ADDR_conv_zero<=adderss for zero value

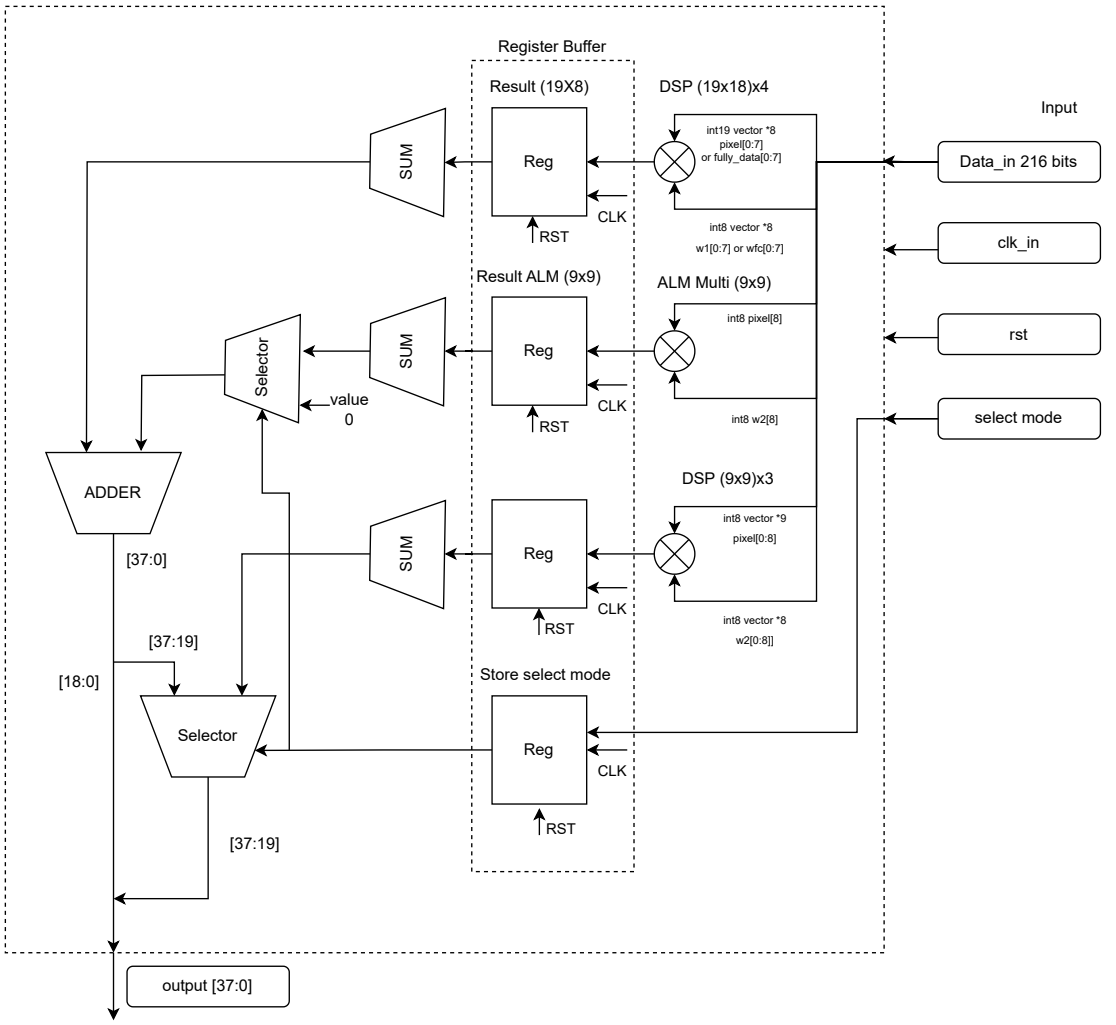
CONV Bais memory Bank

CONV_B_MEM
[15:0] X N
ADDR_Conv_start <=start adders
ADDR_Conv_end <=end adders
ADDR_conv_zero<=adderss for zero value

Fully_connect FIFO

Relu_out_FIFO
[18:0] X128 X3 X N
Relu_out_Read_en
Relu_out_Write_en

MultiMultiplierEngine
(created and tested)



Block operetion:

When Mode 0 selected:

We preform Sum of eight(data(int19)*weight(int8))

Input split :

$pixel_{int8}$ [0:8] 72 bits

$weight1_{int8}$[0:8] 72 bits

$weight2_{int8}$[0:8] 72 bits

When Mode 1 selected:

We preform Sum of Nine(pixel(int8)*weight1(int8)) and the sum of Nine(pixel(int8)*weight2(int8))

Input split :

$fully\_data_{int19}$[0:7] 152 bits

$fully\_weight_{int8}$[0:7] 64 bits

7 DSP per block:

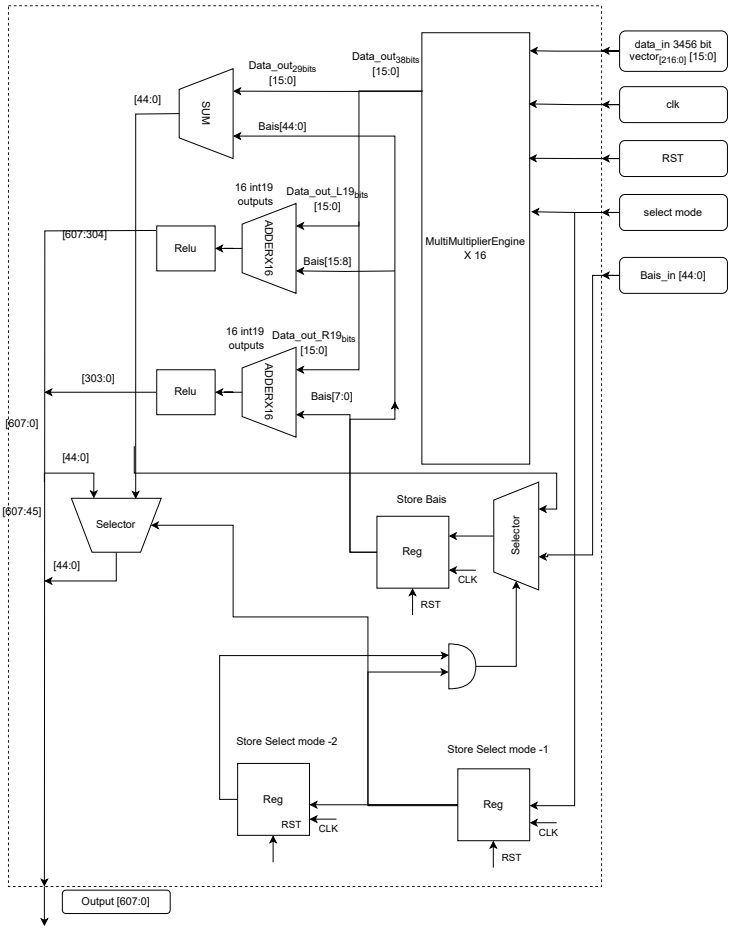
3 configure to run tree 9x9 multilper each

4 configure to run two 19x8 multilper each

Pipline configuration

The Block can run with speed up to 73Mhz without Pipline up to 51Mhz

Parallel_Compute_Engine_16

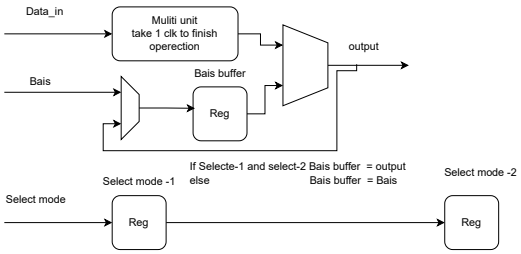


Block operetion:

FIrst we run all the data_in vector in 16 MultiMultiplierEngine units.

When Mode 0 selected:
We add the biases to the outputs of the 16 MultiMultiplierEngine units for total of 32 different int19 results also doing this step we add selector that will set zero to the negative(Relu layer).
Input split :
Bais for the left results <=Bais_in [15:8]
Bais for the left results <=Bais_in [7:0]

When Mode 1 selected:
We sum the outputs of the 16 MultiMultiplierEngine units with the input bais as it is.



Pipline								
Clk	Data_in	**Bais In**	**Select mode**	**Mult unit Buffer**	**Select mode -1**	**Select mode -2**	**Bais Buffer**	**Output**
1	**Conv(1)**	**B1,B2**	**0**					
2	**0**	**0**	**1**	**Multi Result(Conv(1))**	**0**		**B1,B2**	
3	**date for(383,256)**	**Bais**	**1**	**0**	**1**	**0**	**0**	**Res(191:175,15:0)**
4	**date for(255,128)**	**Dont care**	**1**	**Multi Result(date for(383,256))**	**1**	**1**	**Bais**	**0**
5	**date for(127,0)**	**Dont care**	**1**	**Multi Result(date for(255,128))**	**1**	**1**	**Sum(383:256)+bais**	**Sum(383:256)+bais**
6	**Conv(2)**	**B1,B2**	**0**	**Multi Result(date for(127,0))**	**1**	**1**	**Sum(383:256)+ Sum(255:128)+bais**	**Sum(383:256)+ Sum(255:128)+bais**
7	**0**	**1**	**1**	**Multi Result(Conv(1))**	**0**	**1**	**B1,B2**	**Fully correct result**