

# Phase 2 VLSI

## Team 1 - Control Unit

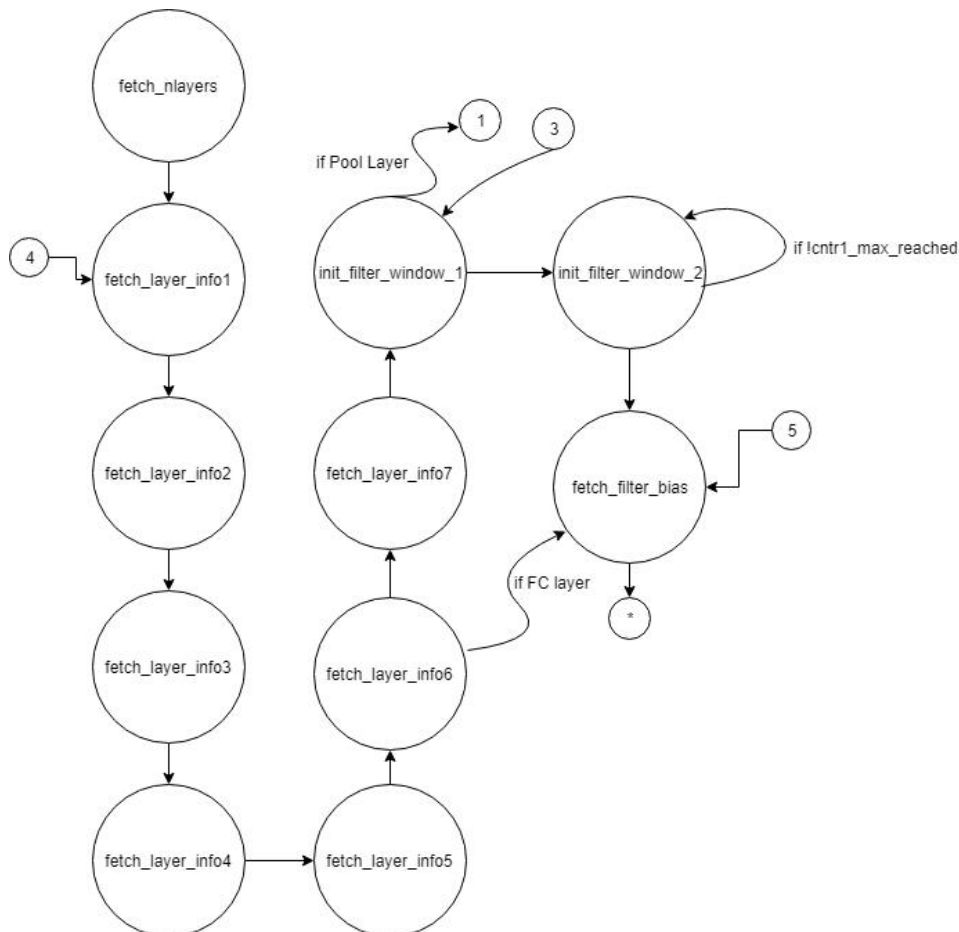
Ahmed Khaled, Maryam Shalaby, Zeinab Rabie, Omnia Zakaria

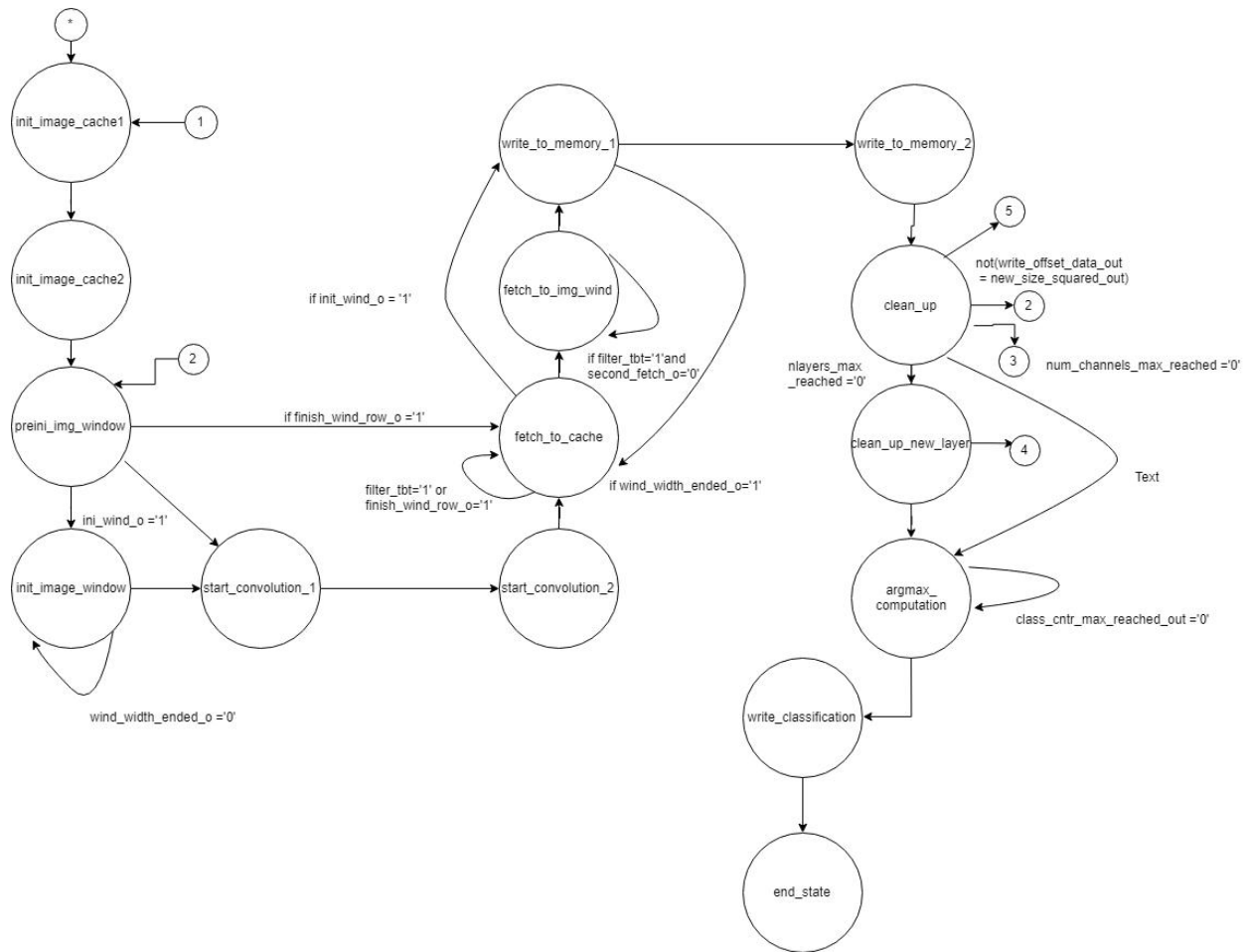
### Design Flow Reports

The sequence of states was modified to accommodate the fact that each input channel has a corresponding filter channel. Therefore, after finishing an input channel, we fetch a new filter channel. The phase 1 document describes each state. The new states are:

- States For reading layer information from memory.
- States for handling the logic of layers and channels' counters.
- Some states needed to be divided into multiple sub-states to avoid the problem of reading and writing to memory or the change of control signals.

#### The modified state diagram:





## Entity Description:

Most of the description is present in the Phase 1 doc, attached. Please see it. The main objective of the control unit is to connect between the I/O Block and the Computation Block.

The control unit is divided into two main modules: **The Image Cache** and **The Controller**.

- **The Image Cache:** The Cache is a block of 28 vertical queues, each queue consists of 5 registers, each one 16 bits, representing a pixel in the image. The input to the image cache is read pixel by pixel, through column inputs, shifting the queue contents vertically with upper most row getting discarded. Thus, imitating a vertical slider on the input image. A decoder chooses the columns to insert a pixel into, inserting into all columns shifts the cache an entire row up.
- **The Controller:** It's responsible for:
  - Reading layers' info from memory and loading them into the appropriate register for use later.
  - Reading the image and loading it into the Image Cache.
  - Reading the filter and loading it into the filter window.
  - Handling the flow of data between image cache and image window.
  - Sending signals to the Computation block to start when data is ready.
  - Saving the results produced by the computation block and saving it into memory.
  - Preparing data needed for the argmax unit to start working.

- Saves the final result of the classification process into memory.

## I/O Description:

- The Interface with I/O block:
  - IO\_ready\_in: an input signal to indicate that all needed data are loaded into memory in an agreed on format.
- Interface with Memory:
  - Mem\_addr\_out: address of data to be read or written.
  - Mem\_read: to read from memory location with the address specified in Mem\_addr\_out.
  - Mem.write: to write into memory location with the address specified in Mem\_addr\_out.
  - Mem\_data\_in: data read.
  - Mem\_data\_out: data to be written.
- Interface with Filter window:
  - Filter\_data\_out: data to be loaded into the filter window.
  - Filter\_ready\_out: to load filter data into the filter window.
- Interface with Image window:
  - Wind\_en: to load data into image window.
  - Wind\_col\_in: the column shifted from the cache to the window.
    - For the sake of the synthesis of our control block, we added the image window inside the whole control block.
- Interface with Computation block:
  - Comp\_unit\_ready: to indicate that data is ready for computation block to work.
  - Comp\_unit\_data1\_out: to be loaded as initial value to the computation block accumulator.
  - Comp\_unit\_data2\_out: to be loaded as initial value to the computation block accumulator in case of a 3x3 filter.
  - Comp\_unit\_data1\_in: to be written to memory.
  - Comp\_unit\_data2\_in: to be written to memory in case of a 3x3 filter.
  - Argmax\_ready: to indicate that data is ready for ArgMax unit to work.
  - Argmax\_data\_out: data to be sent from memory to the ArgMax unit to operate on.
  - Argmax\_data\_in: result of the ArgMax Unit.