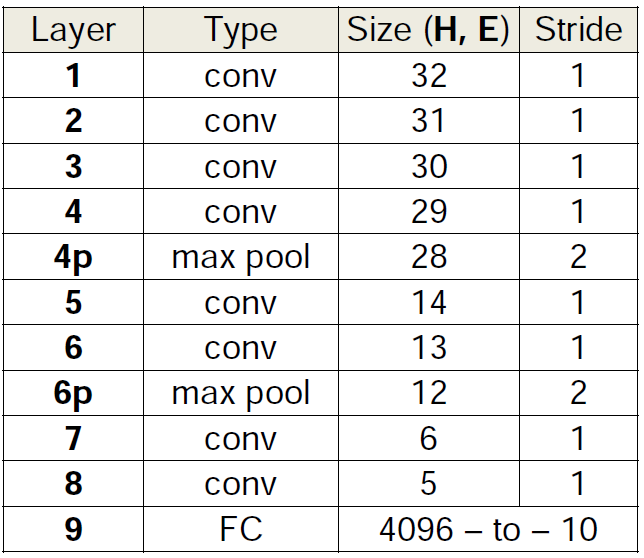
亮点：

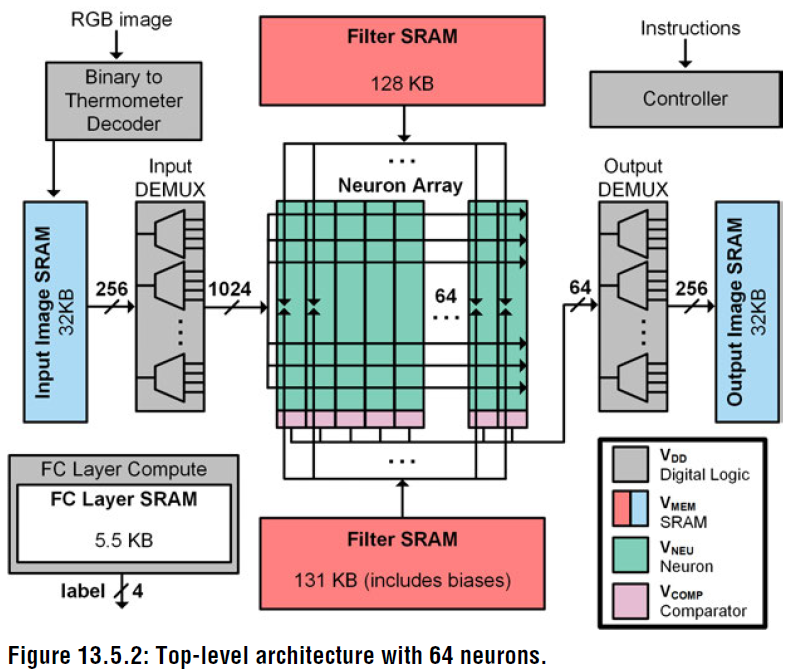
1. 使用binary network来降低了数据量
2. 使用数模混合的方式即switched-capacitors来替代传统的数字加法器，从而减少了硬件开销
3. All on chip memory，减少了从片外存储中读取数据所带来的硬件资源消耗

网络结构：



其中，对于每一层网络的filter的大小时钟固定为2x2x256x256。

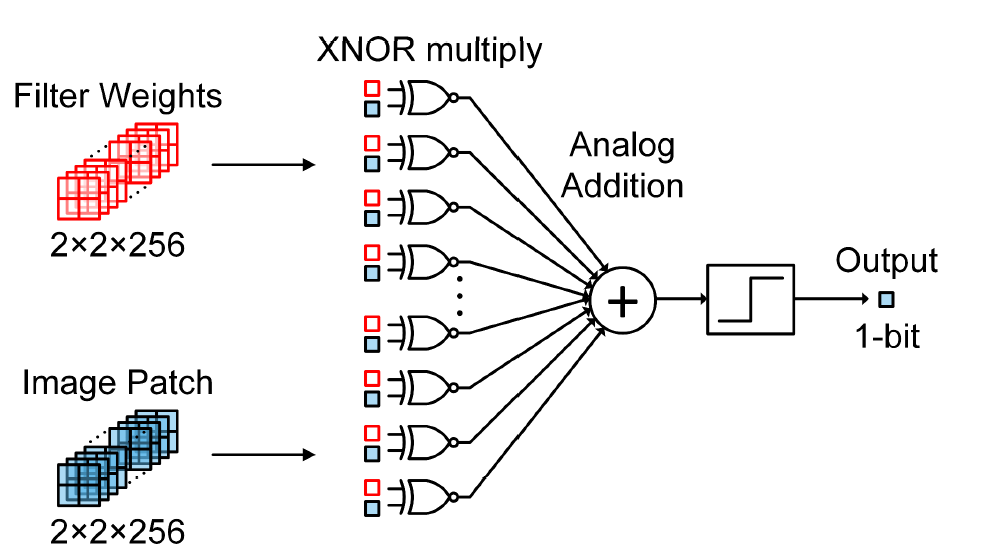
DLA 整体架构



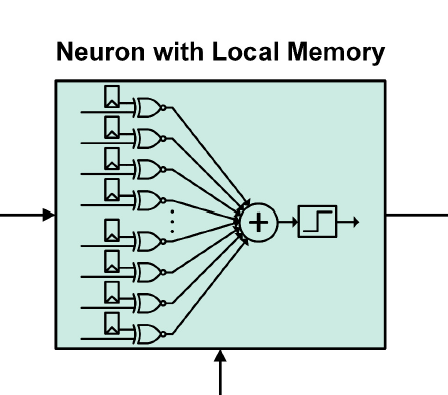
该款使用switched-capacitor neuron来代替传统意义上的加法器，每个neuron能够支持1024个输入，总共有64个neuron。其中1024对应于一个set的filter 共2x2x256个element。使用这种方式一是能够充分利用硬件的并行性，并且不会产生psum，减少了psum访存的功耗。对于输入输出图像的SRAM是可以进行交换的，也就是数据流向可以完全反向，这样减少了存储器的用量。对于输入DEMUX而言，能够将输入图像派遣到对应的neuron中的对应输入端。因为filter的大小是2x2，所以这里的DeMUX是1-to-4。所以实际上假设在初始阶段load input image需要4个时钟的latency。之后需要2个时钟latency等到demux将所有数据派遣完毕之后才可开始计算。对于output DEMUX而言，由于共有64个neuron，也就是一次能够产生64个output channel数据，而每个数据之间channel相差4，所以需要用demux重新分配。

并行性以及reuse：

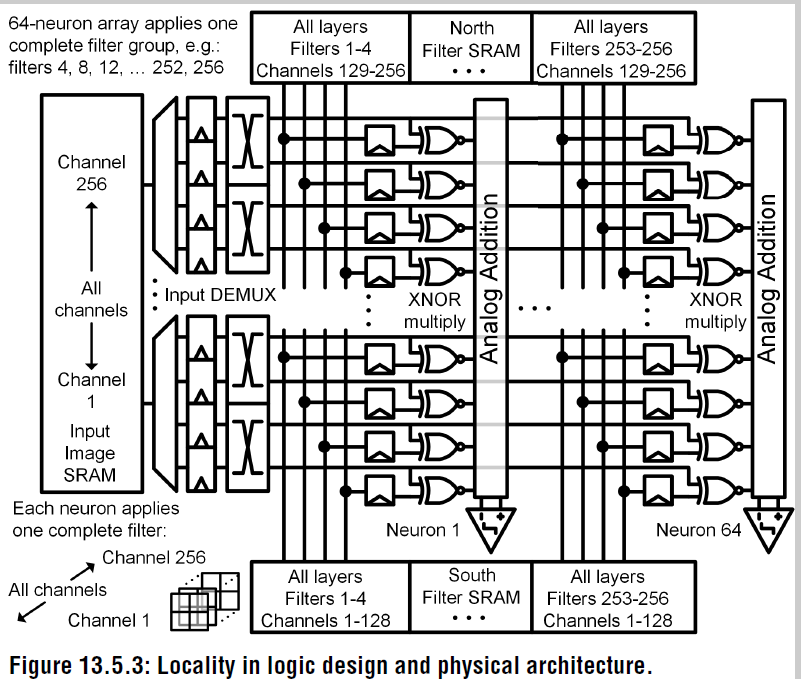
考虑到数据的相关性，以及期望减少weght的搬运次数，采用如下的并行方式和数据从利用。



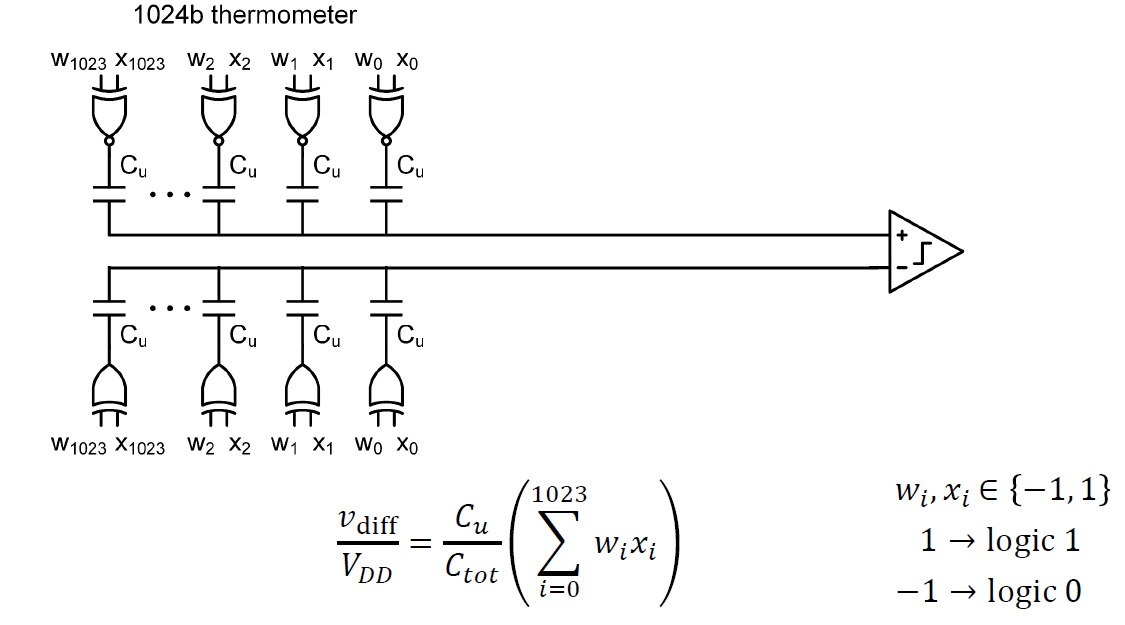
也就是一次处理完一个output channel的数据。而为了使用weight stationary的dataflow，采用如下结构



上图说明了其内部更为细致的结构：



由图可以看到，在每个1-4 demux之后还有两个2-2 的crossbar，这样做的目的是，由于weight是按照固定顺序排列好放在寄存器中的，但是考虑到相邻的两个feature patch的数据存在两个数据的overlap，所以需要使用2-2 crossbar来交换顺序，使得feature-weight pair 能够匹配。例如初始的feature 按照从左到右，从上到下的顺序标记为0，1，2，3.当移动到下一个feature patch的时候，需要使用的就会变成4，1，5，3。所以为了保证顺序的正确使用crossbar交换之后就会变成1，4，3，5.

Swithed capacitor neuron adder 原理介绍：

首先对于二值化的网络，以上网络为例说明，那么其和应该为

,

但是在实际电路中并不存在-1，只有logic 1 和 logic 0. 于是我们使用logic 1来代替1，logic 0来代替零。所以从表达式的等价性上来说乘法就变成了同或，或者说是异或非。即

但是这样的逻辑存在这一个问题就是，在实际数学计算过程中，是存在加法(1)和减法(-1)的，而在数字电路中只有加法(1)和不变(0)，会产生误差，于是我们使用差分输入的方式即可保证逻辑表达式和数学表达式的等价：

实际网络 1 -1

同或逻辑 1 0

异或逻辑 0 1

差分输入 1 -1

从而保证了功能正确性。

如果仅从数学的角度上出发的话，由电荷守恒定理，有：

如果从电路的角度出发那么有：

上两式相减即可得到正确的结果。