

Implementing Digit Classification with Convolutional Neural Network

IC Design Lab Final Presentation

Group 1: Yun-Sheng Chang, Yi-Hao Huang

Outline

- ❖ Check List
- ❖ Design and Implementation
- ❖ Synthesis and Optimization
- ❖ P&R Result

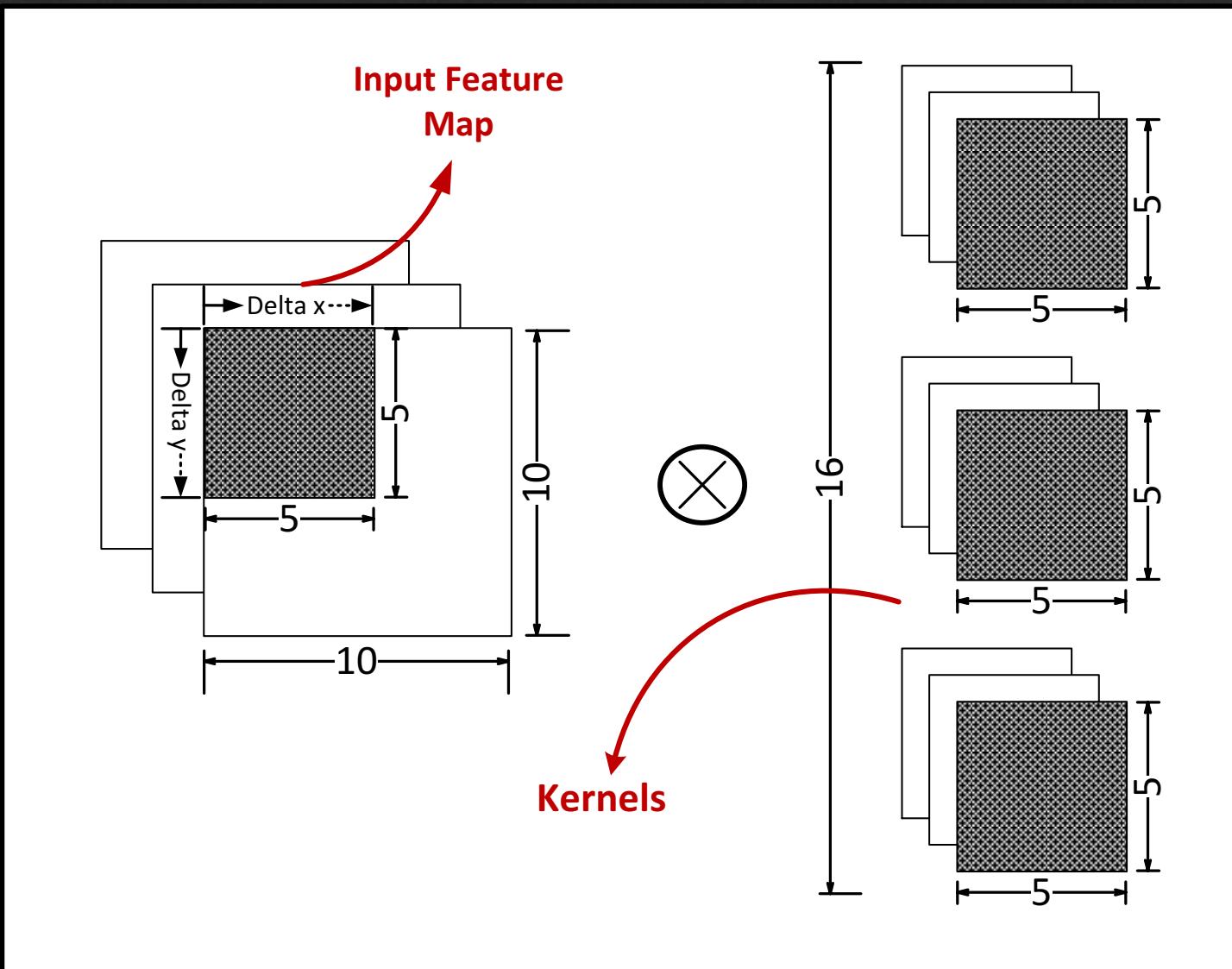
Check List

- ❖ Pre-simulation: done
 - ❖ pass 10 testbenches
 - ❖ cycle count: 935145
- ❖ Synthesis: done
 - ❖ area: 508797 μm^2
 - ❖ frequency: 83.3 MHz
 - ❖ power: 8110 μW
- ❖ P&R: done
 - ❖ area: 673510 μm^2
 - ❖ power: 12304 μW
- ❖ Post-simulation and Time-based Power Analysis: undone

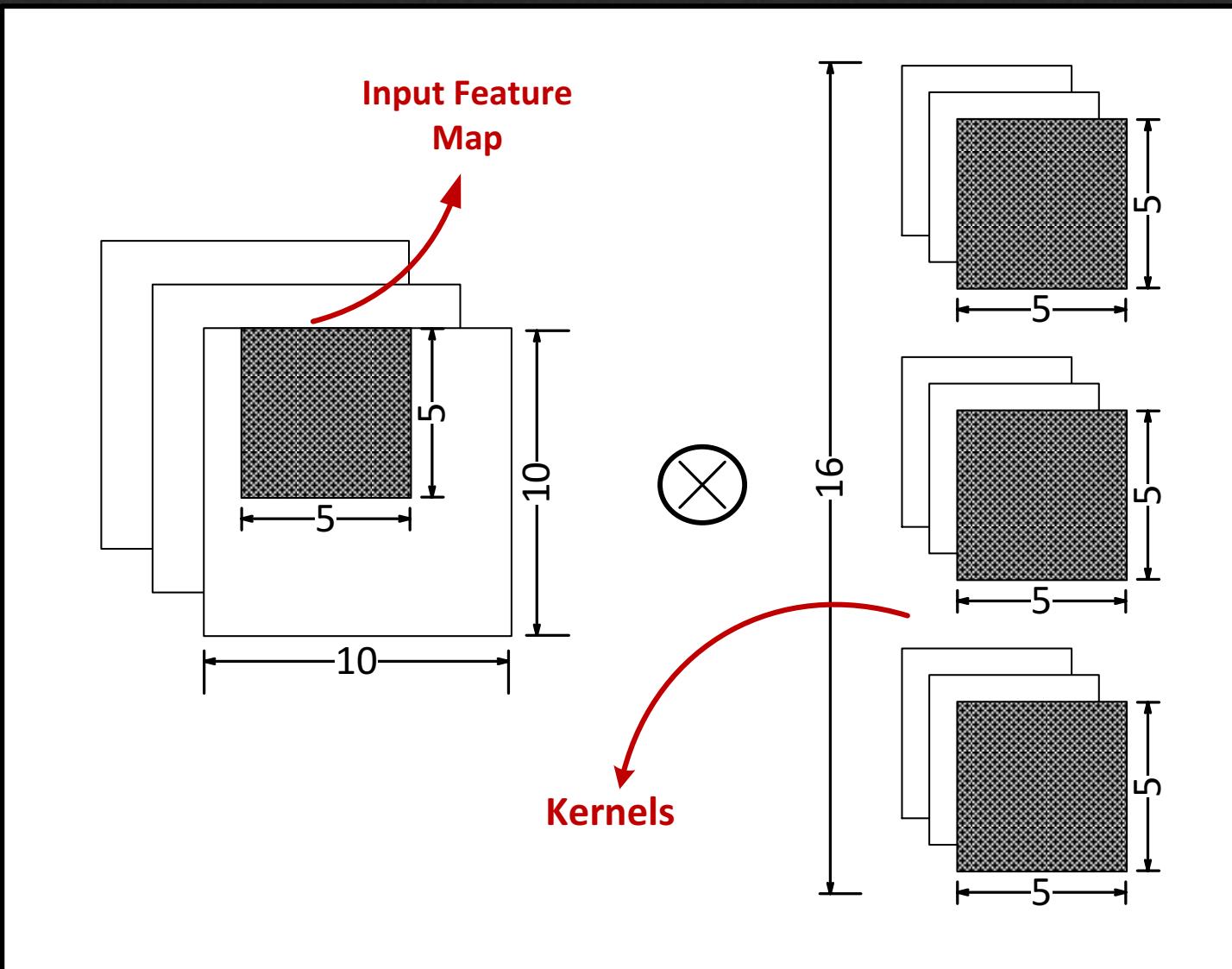
Design and Implementation

- ❖ Convolution layer
 - ❖ IDLE
 - ❖ LD_PARAM
 - ❖ LD_KNL
 - ❖ LD_IFMAP_FULL
 - ❖ LD_IFMAP_PART
 - ❖ CONV
 - ❖ DONE

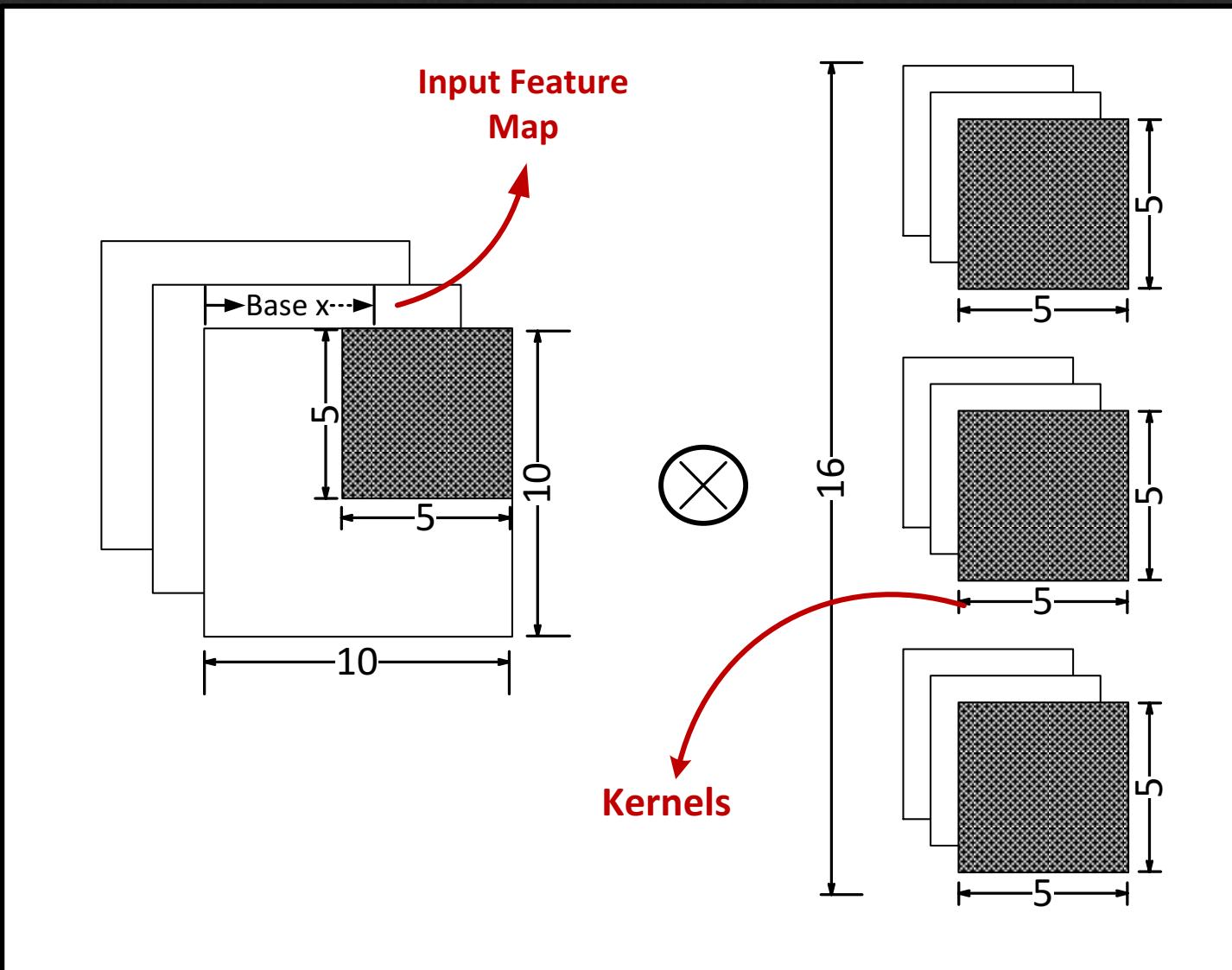
CONV



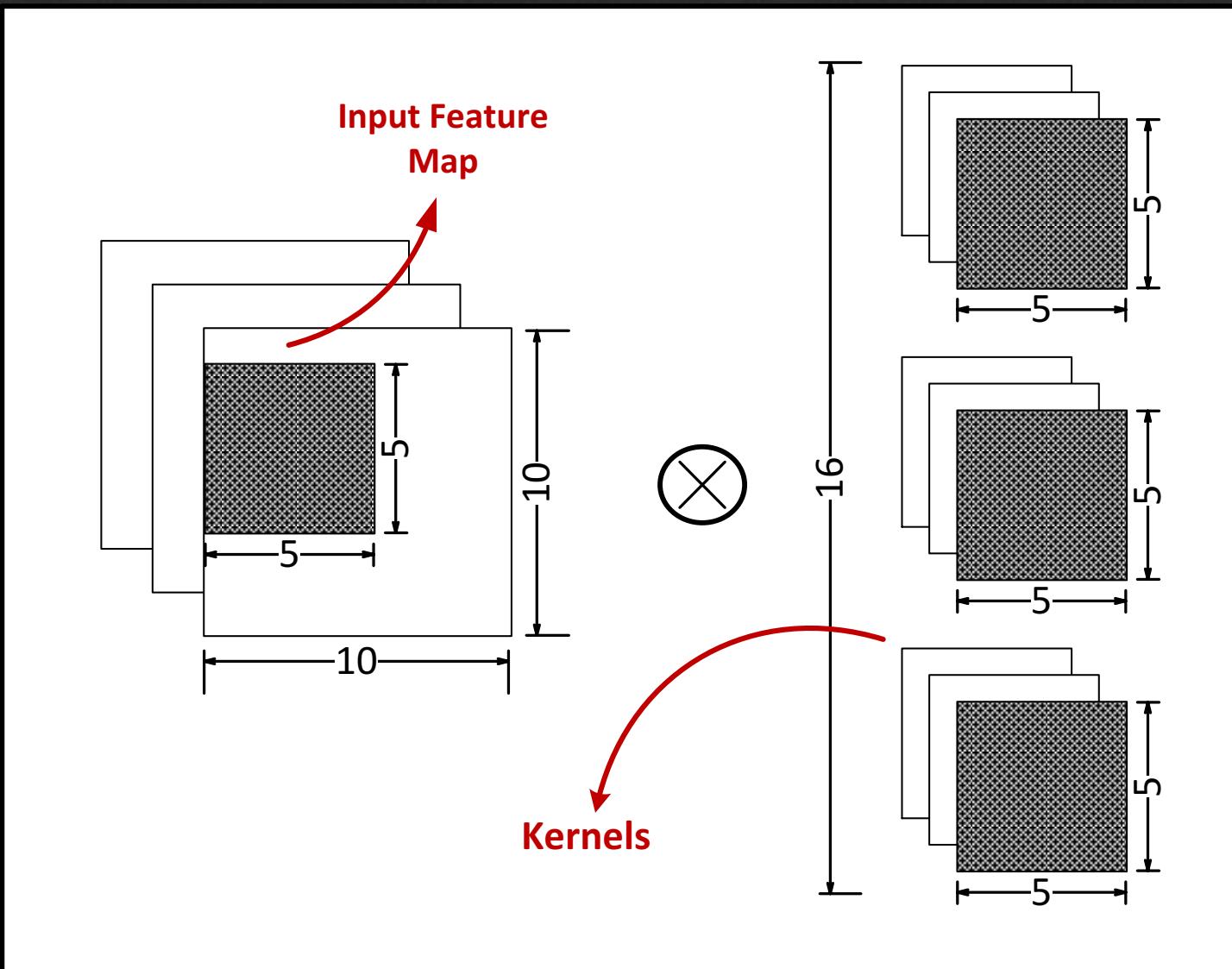
CONV



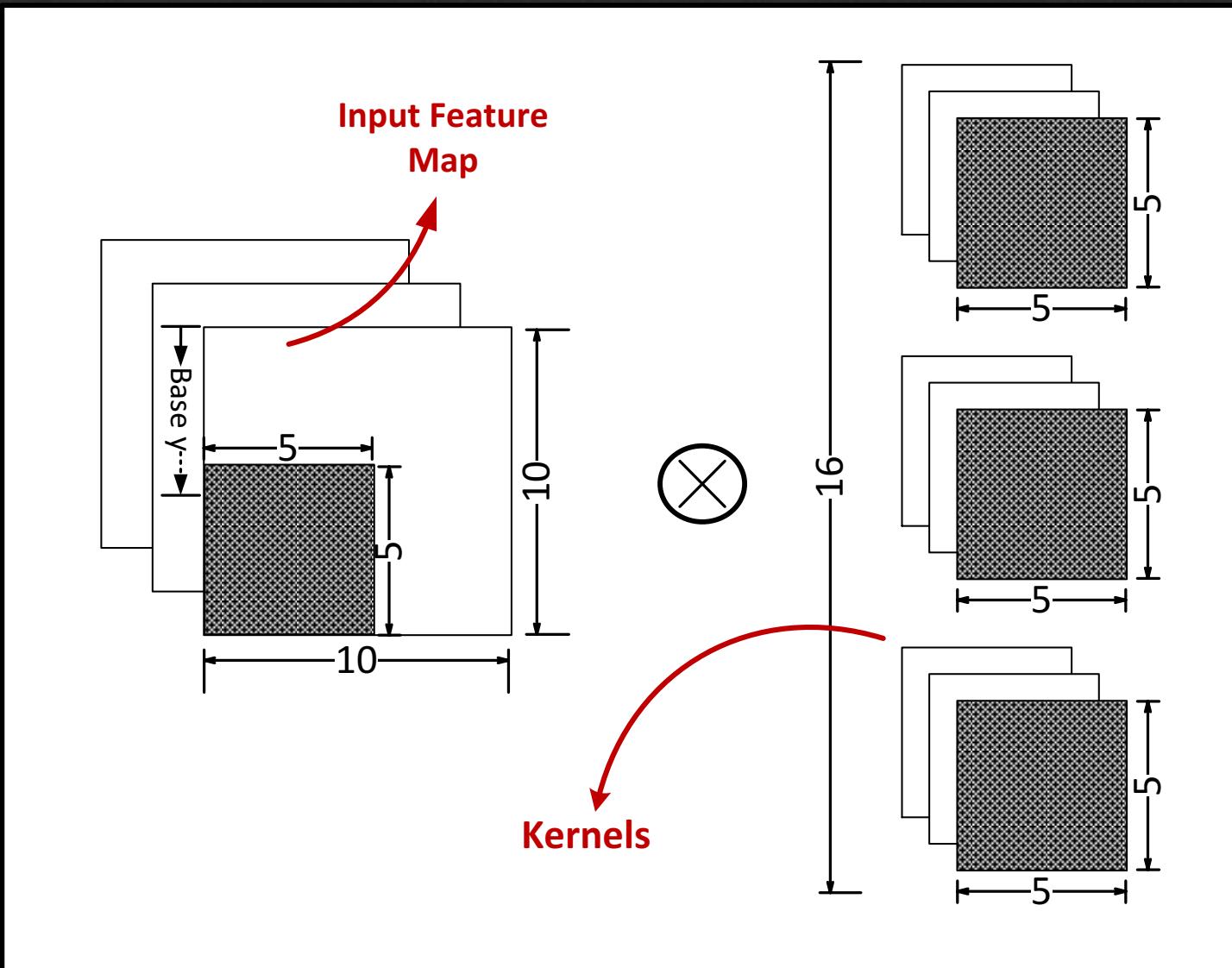
CONV



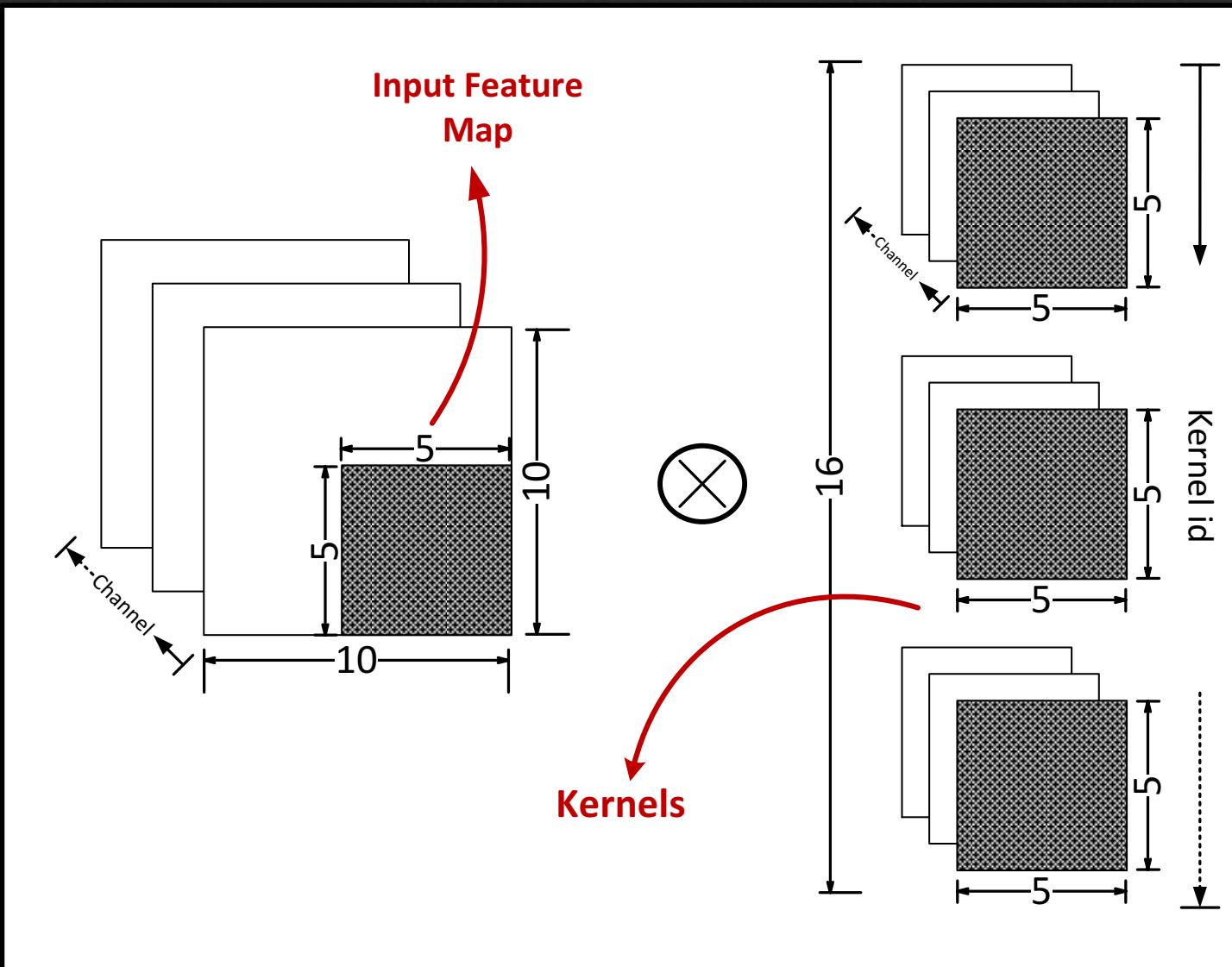
CONV



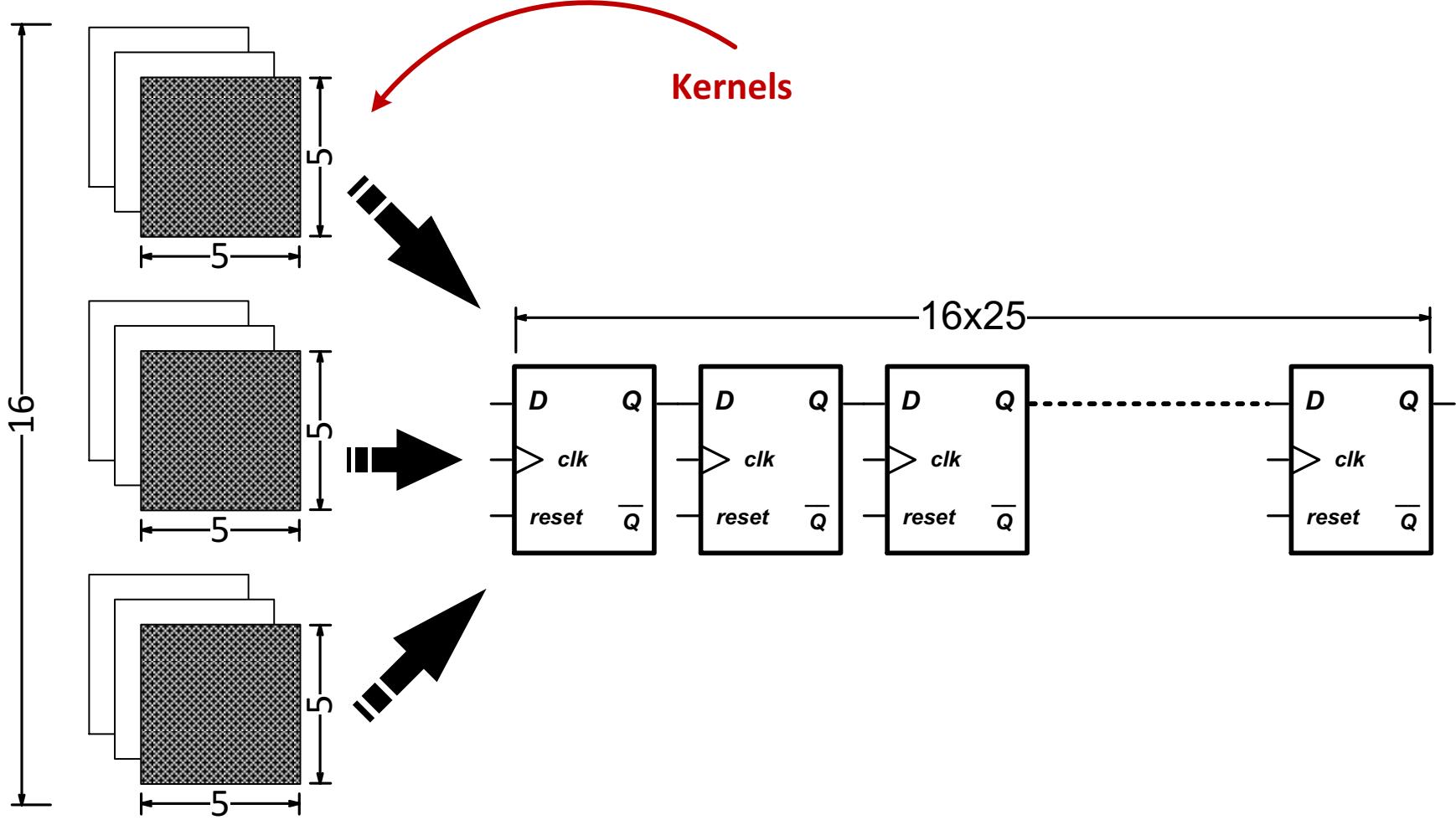
CONV



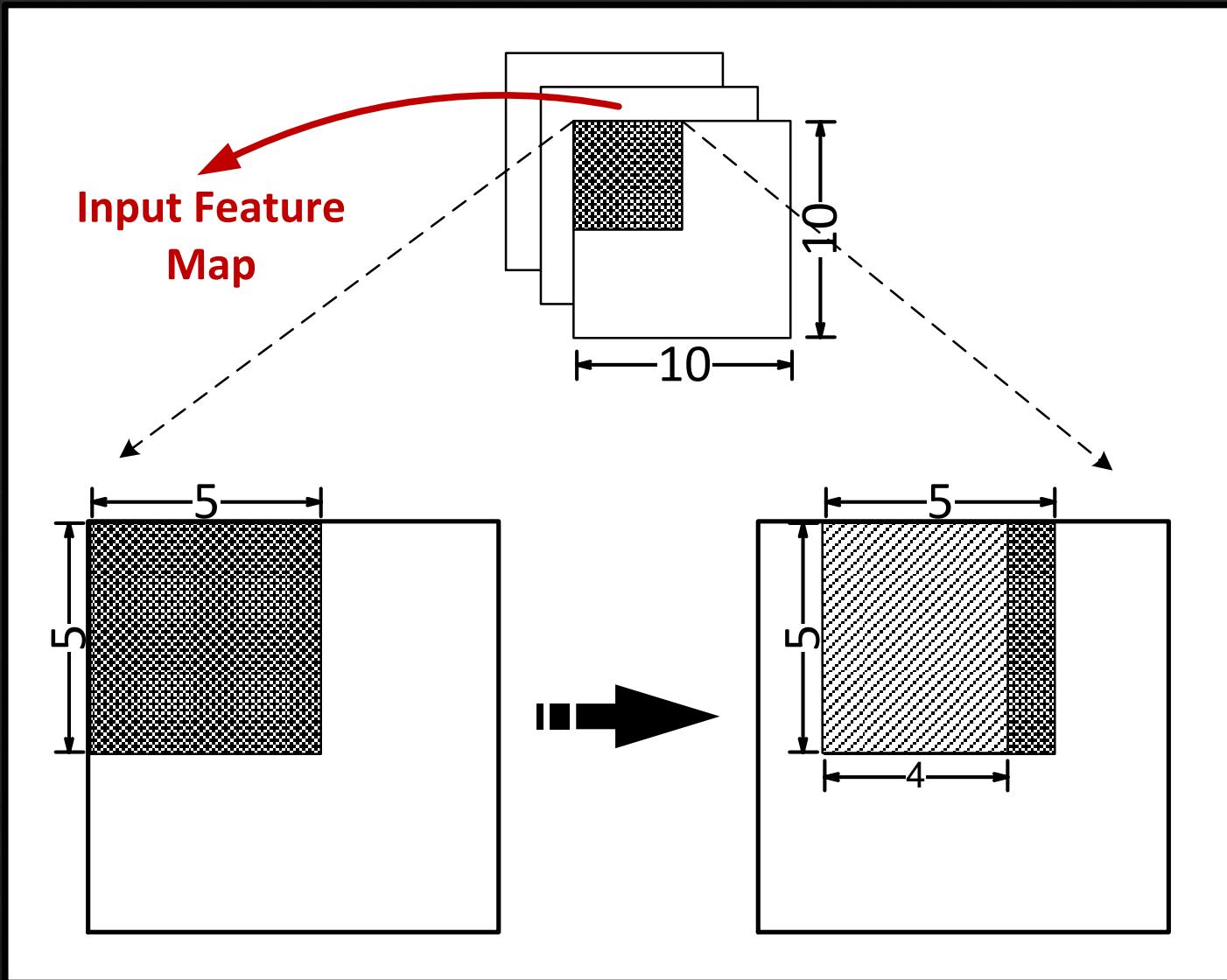
CONV



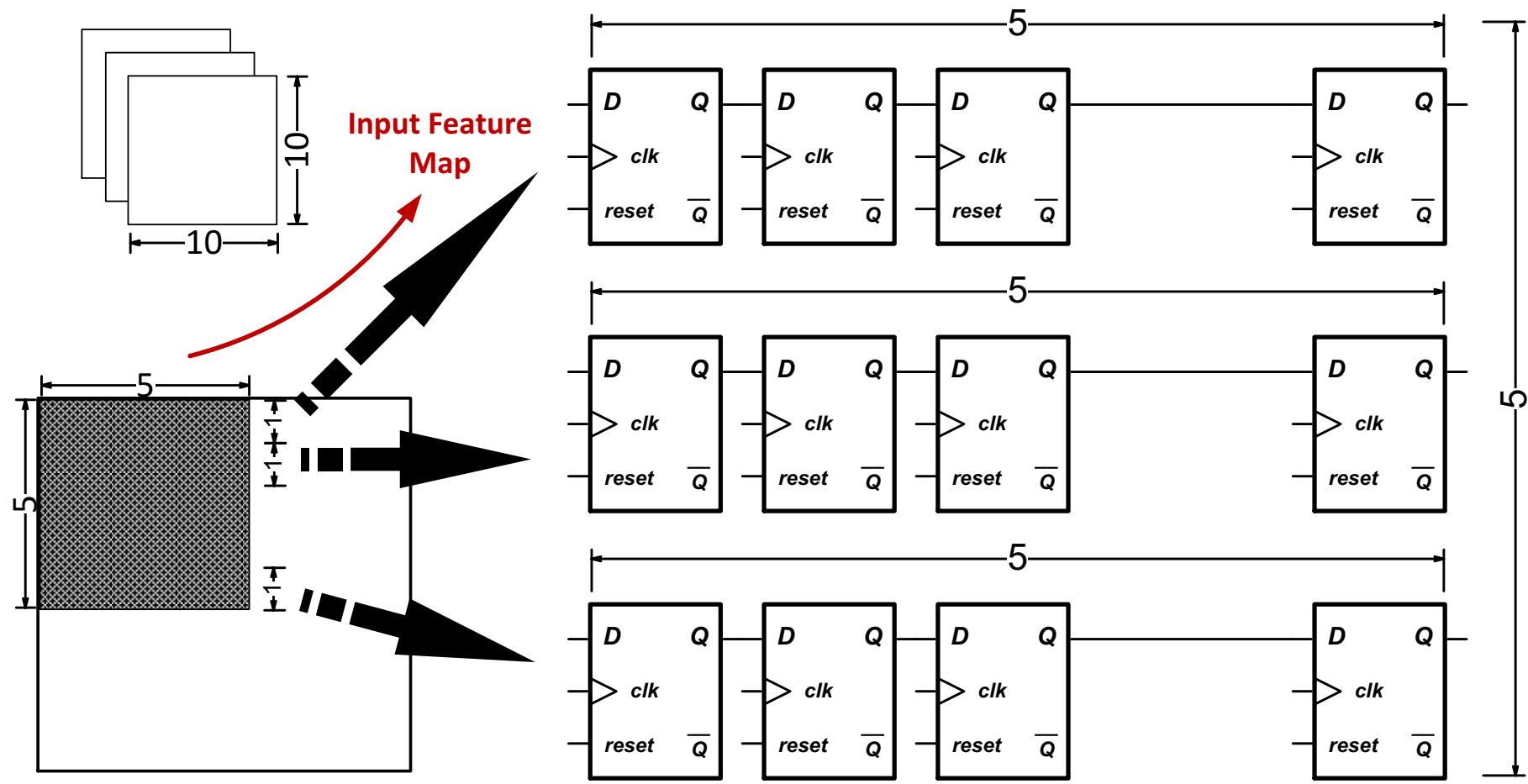
LD_KNL



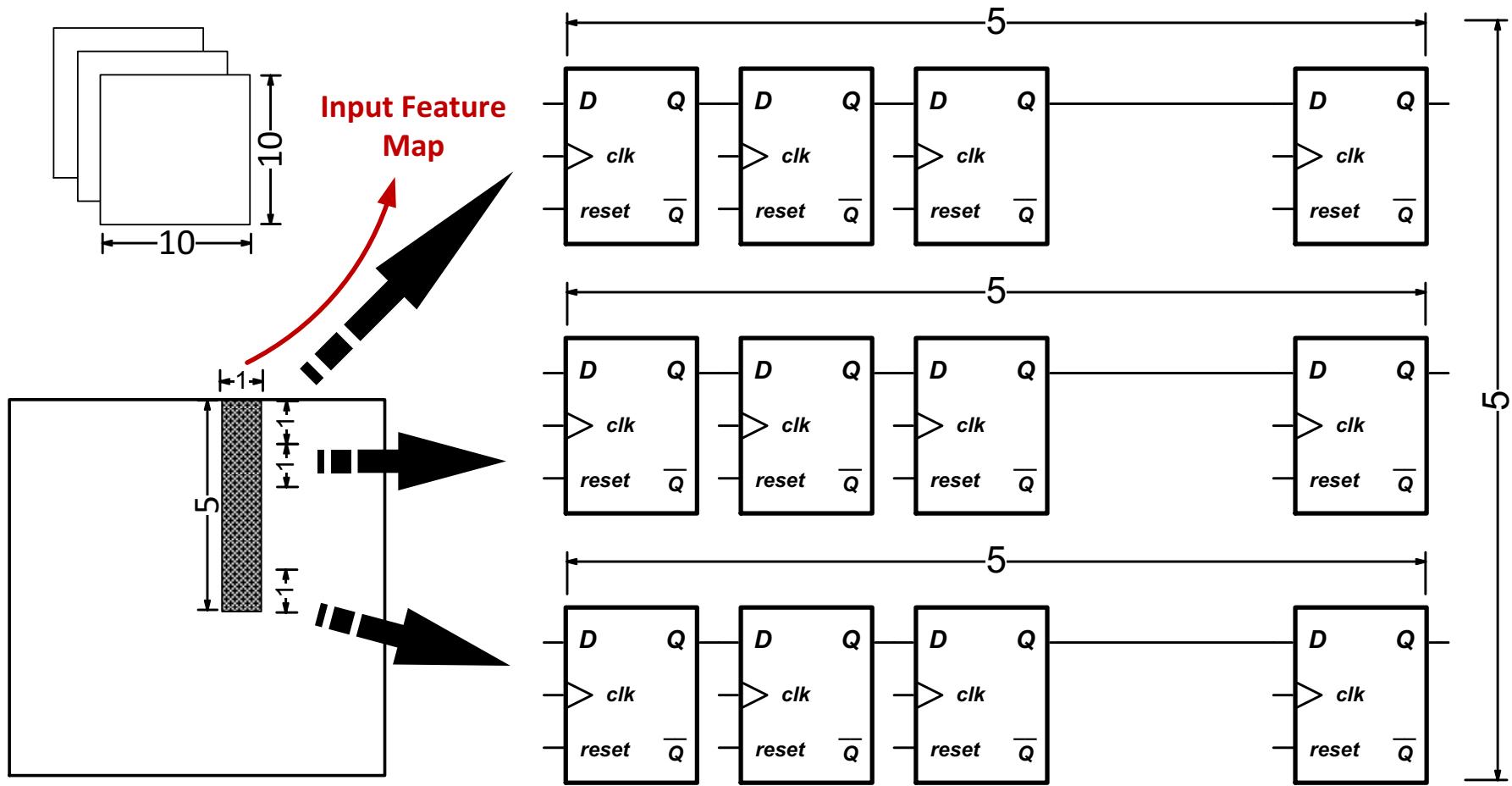
LD_IFMAP



LD_IFMAP_FULL



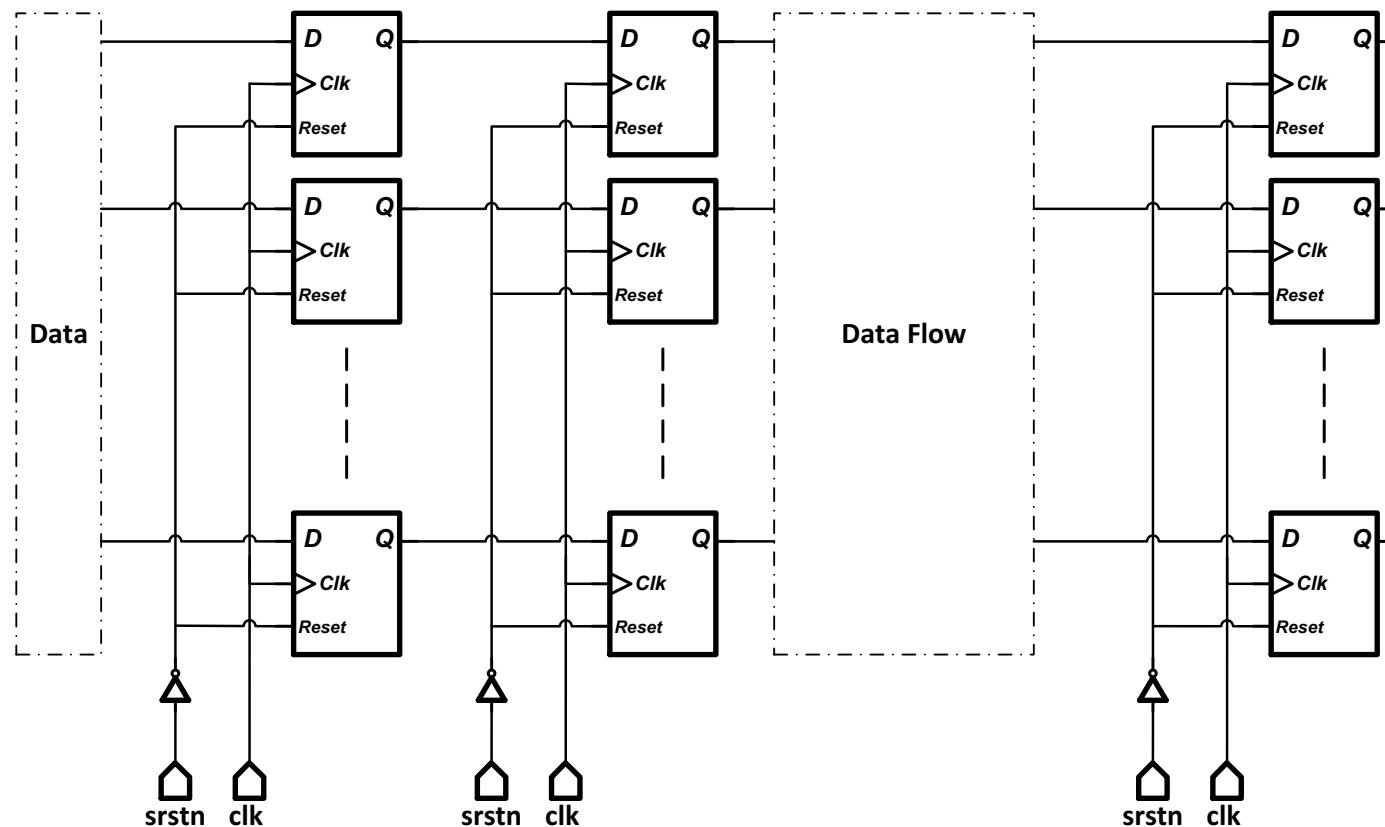
LD_IFMAP_PART



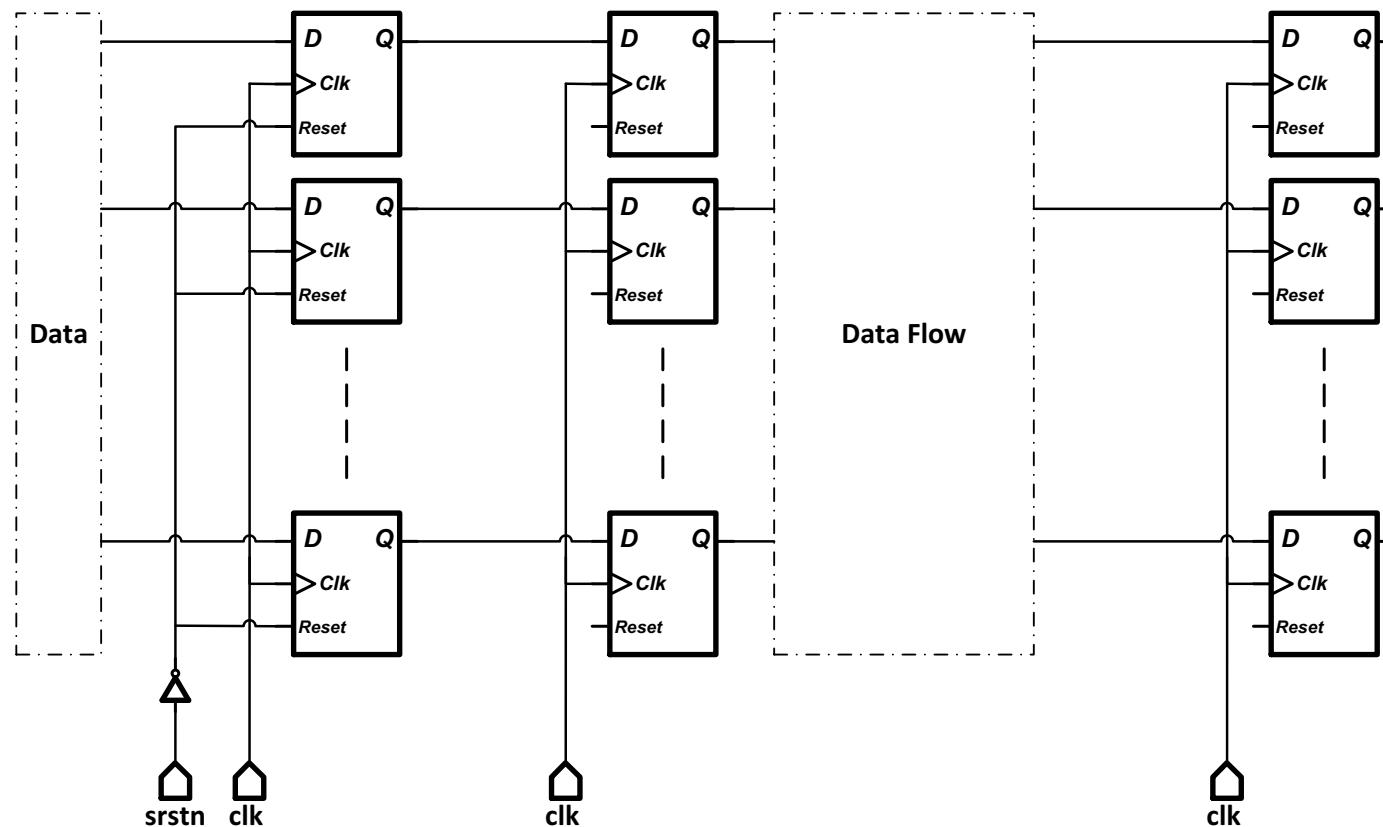
Optimization & Synthesis

- ❖ Change good structure, reduce area & shorten critical path
 - ❖ remove reset for follower flip-flops
 - ❖ build address table for address decoder MUXs
 - ❖ ring flip-flops structure to remove 9-bits address decoder

Non Reset Follower-Flip-Flops

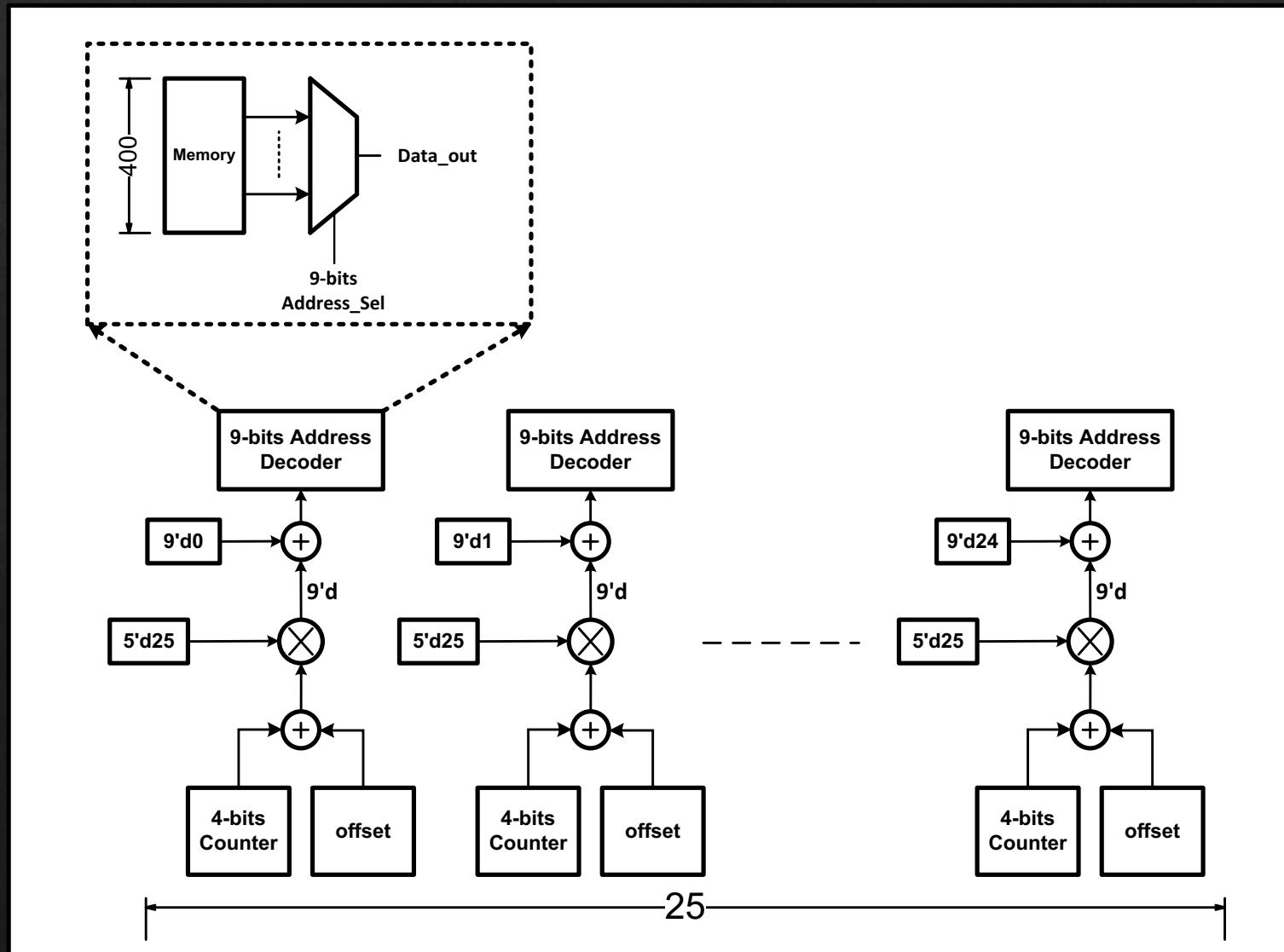


Non Reset Follower-Flip-Flops

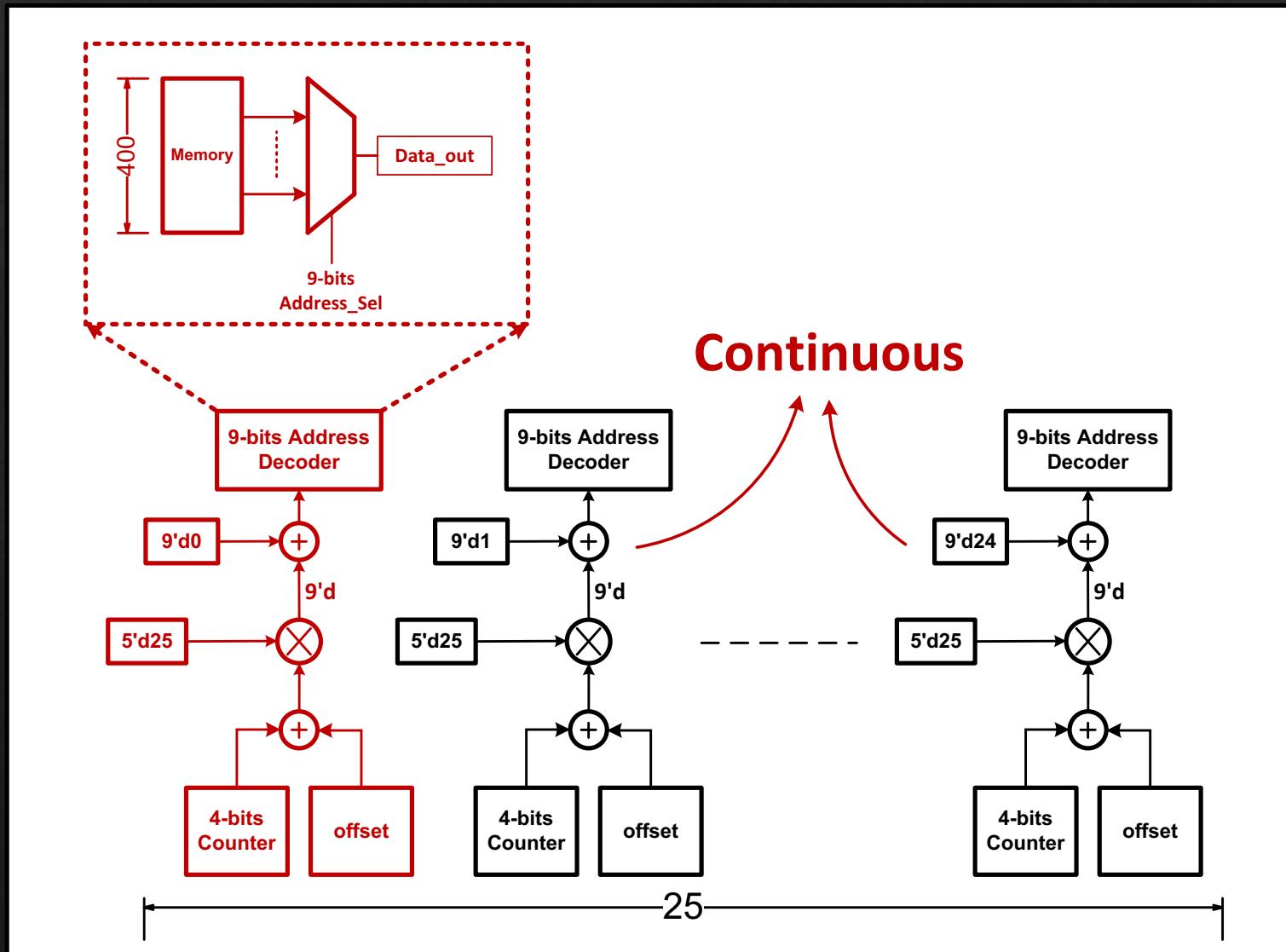


area decreased by 15w um²

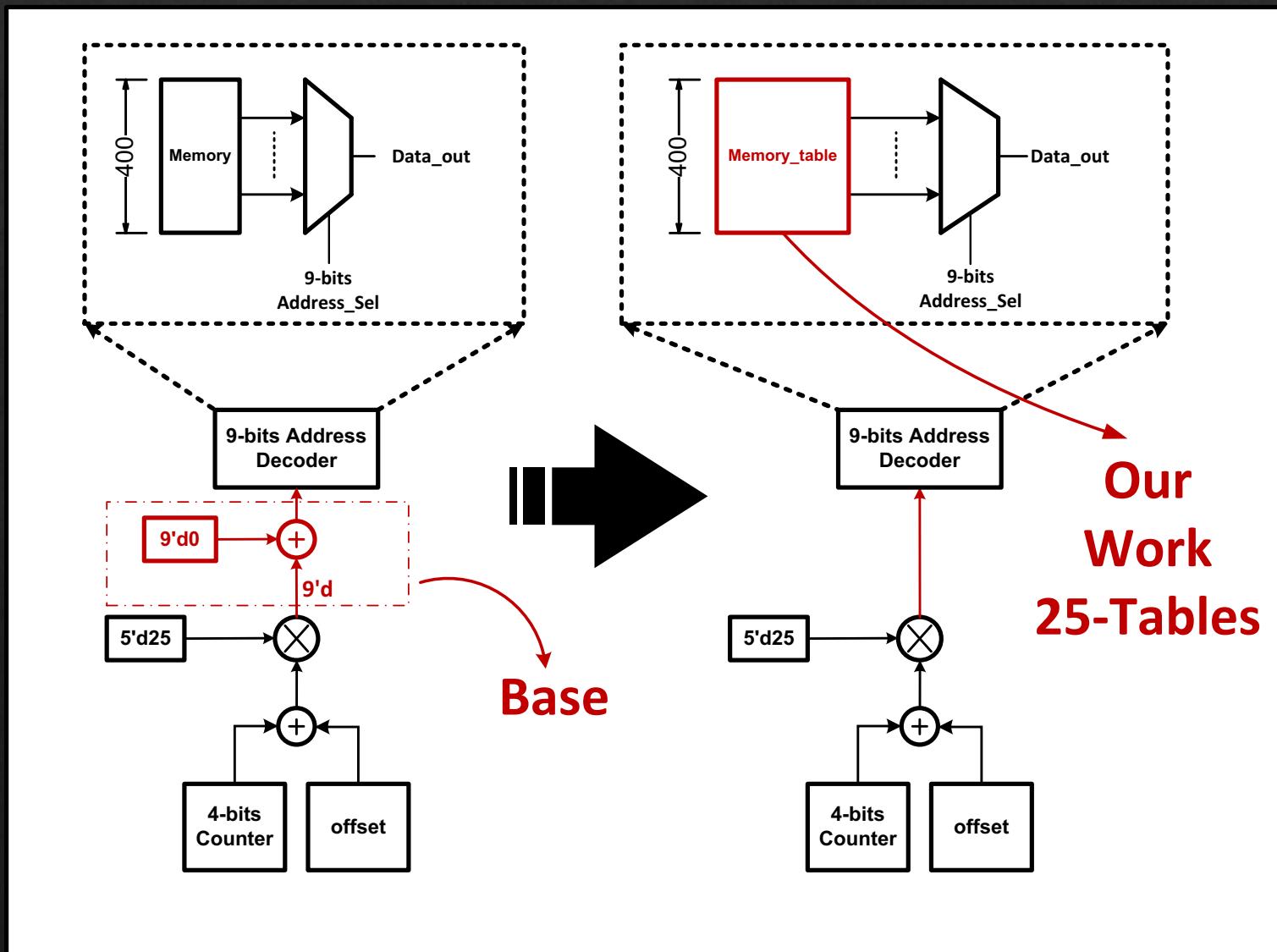
Build Address Table



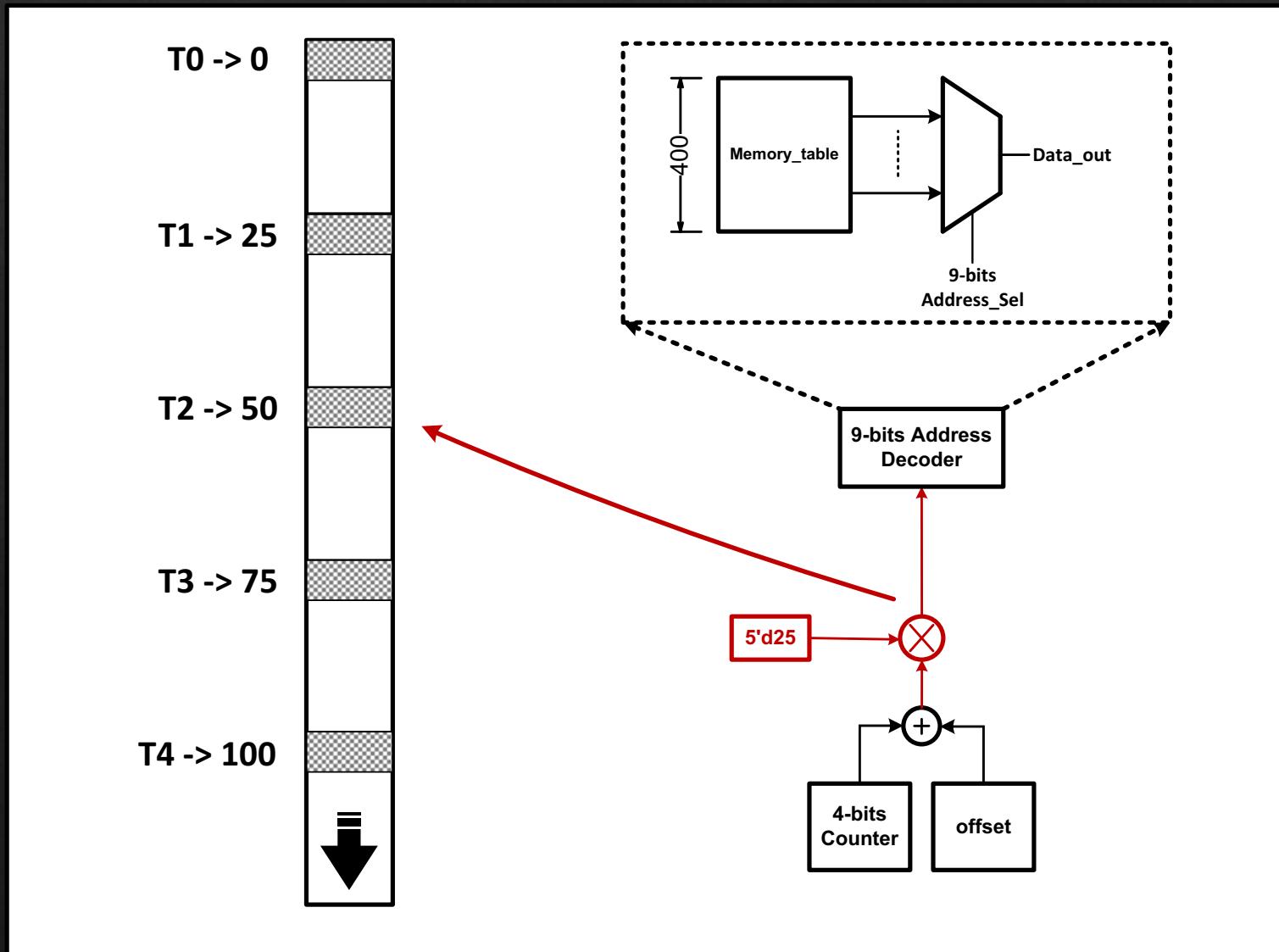
Build Address Table



Build Address Table

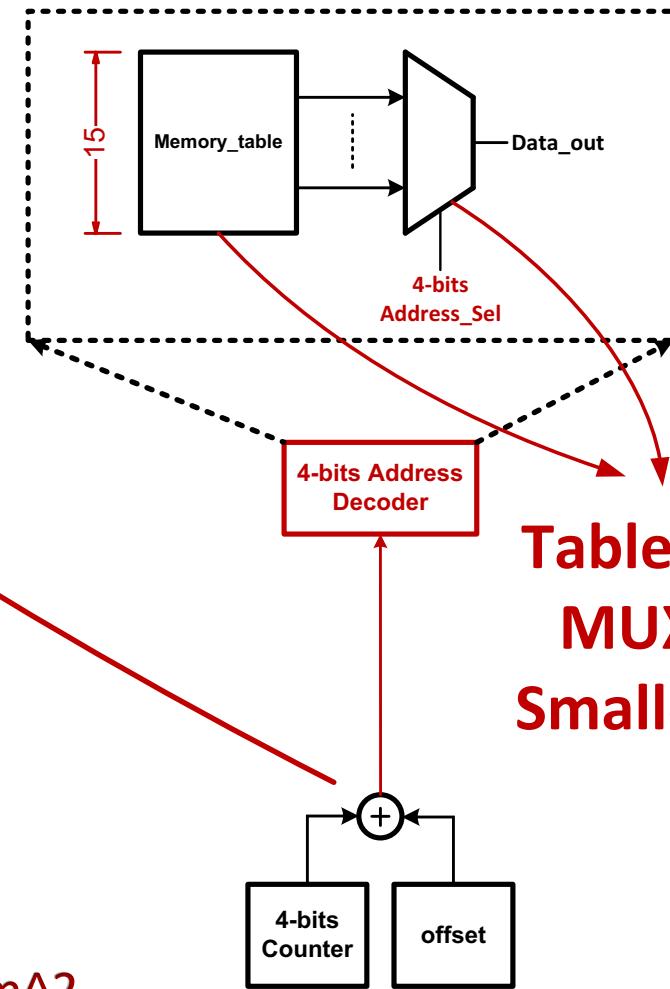


Build Address Table



Build Address Table

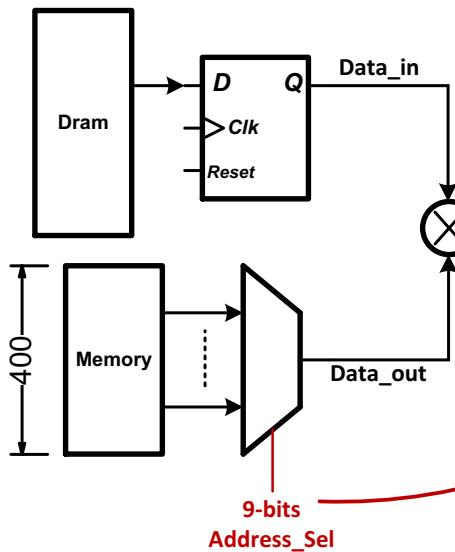
T0 -> 0	
T1 -> 1	
T2 -> 2	
T3 -> 3	
T4 -> 4	
T5 -> 5	
T6 -> 6	
T7 -> 7	
T8 -> 8	
T9 -> 9	
T10 -> 10	
T11 -> 11	
T12 -> 12	
T13 -> 13	
T14 -> 14	
T15 -> 15	



**Tables &
MUXs
Smaller!!**

area decreased by 170w um²

Ring-Flip-Flops

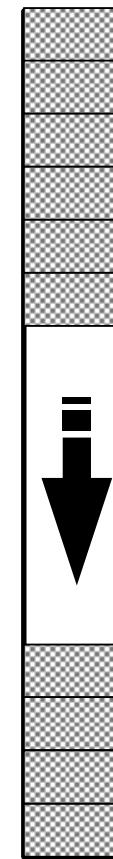


Perform 400
times
With 9-bits
address_sel

Continuous
increasing

T0 -> 0
T1 -> 1
T2 -> 2
T3 -> 3
T4 -> 4
T5 -> 5

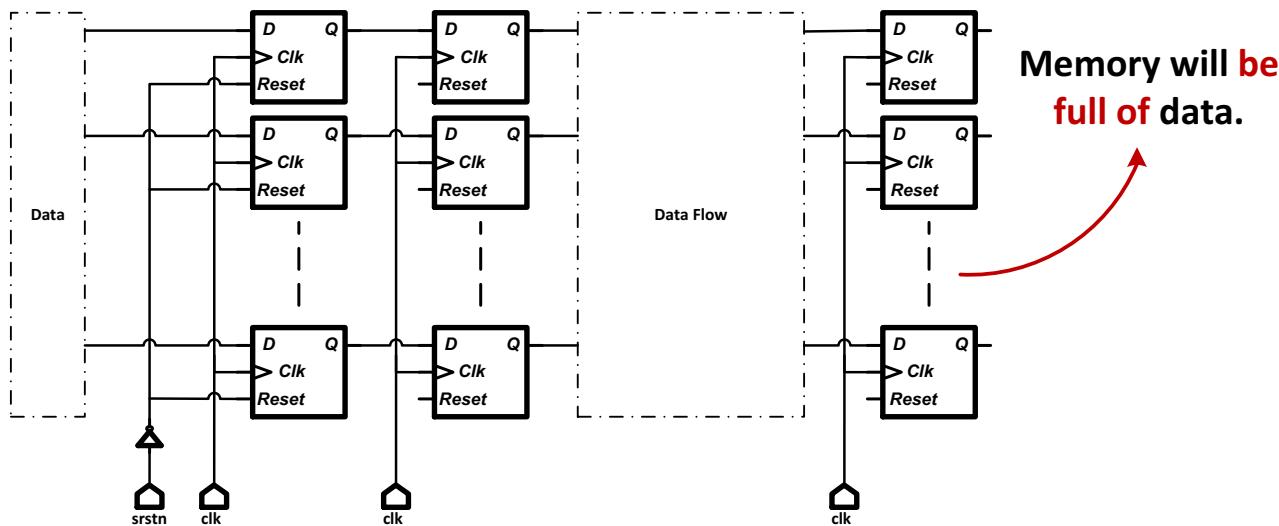
T396 -> 396
T397 -> 397
T398 -> 398
T399 -> 399



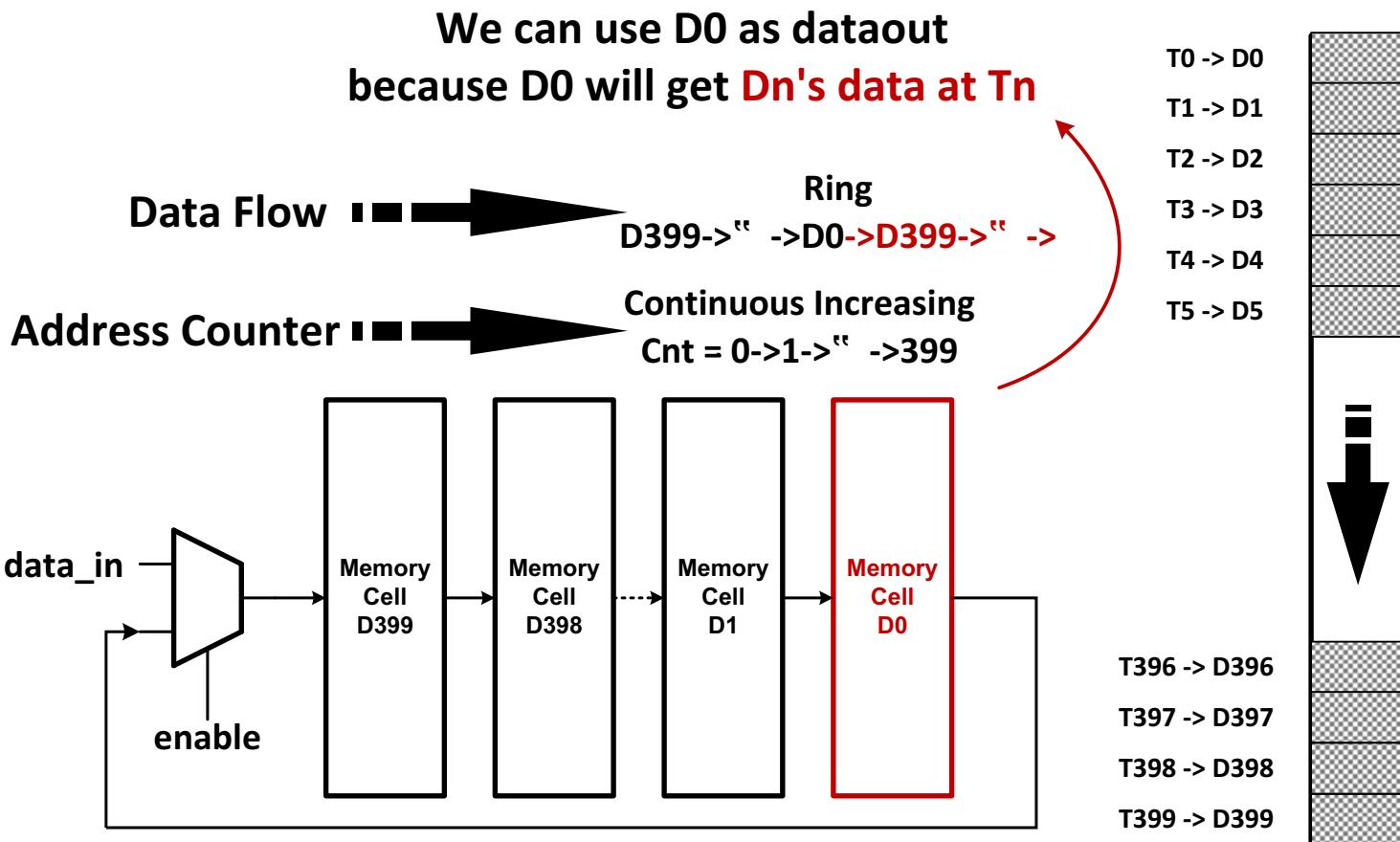
Ring-Flip-Flops

Data Flow → One by One
D399->D398->“ ->**D0**

Address Counter → Continuous Increasing
Cnt = 0->1->“ ->399

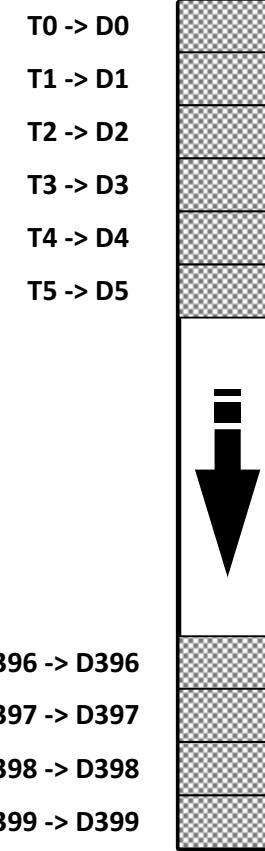
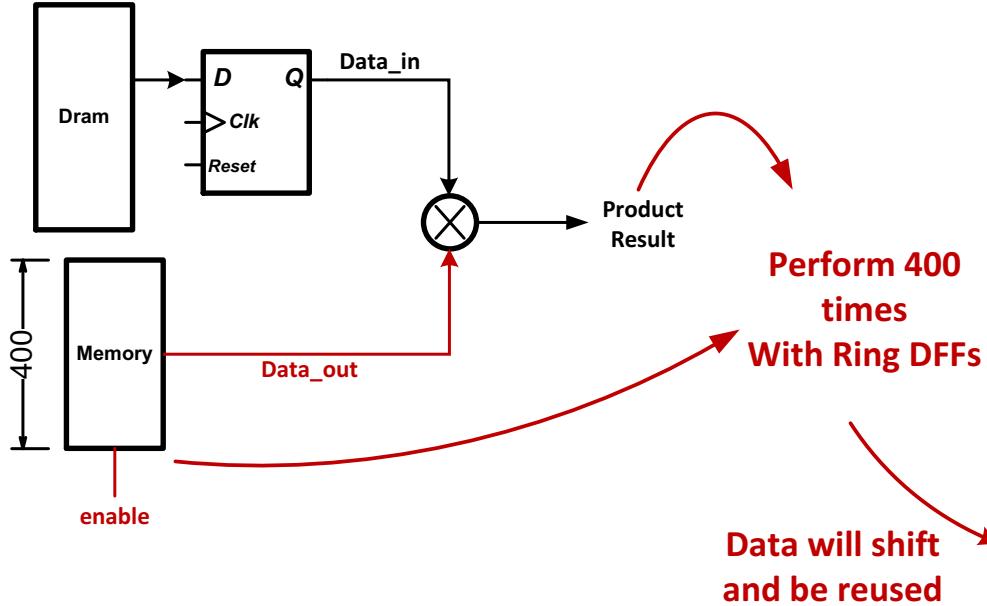


Ring-Flip-Flops



Ring-Flip-Flops

area decreased by $7w \text{ um}^2$

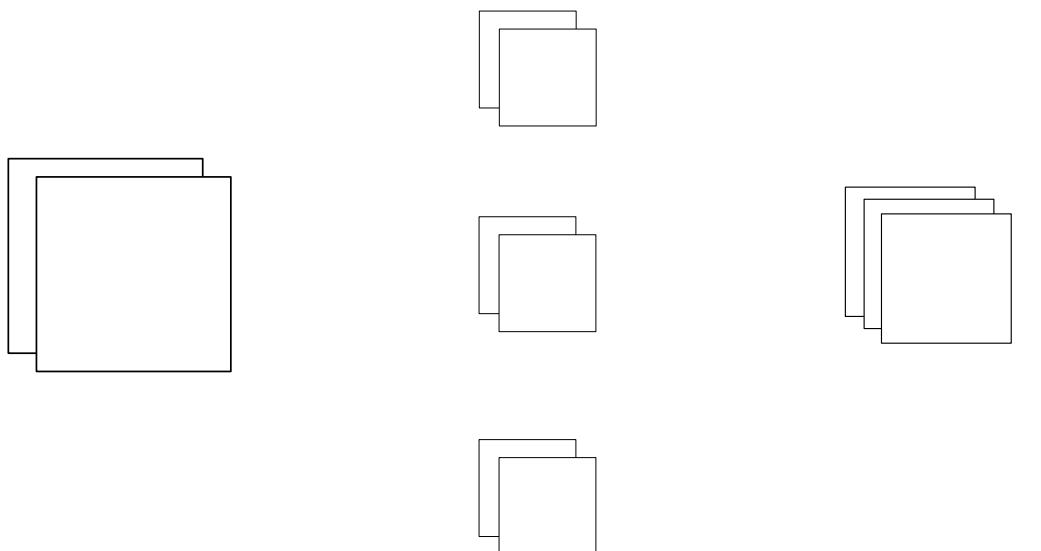


P&R Result

- ❖ Try 1
 - ❖ core utilization: 0.7
 - ❖ left, bottom, right and top io2core: 30
 - ❖ take more than 15 hours
- ❖ Try 2
 - ❖ set number of horizontal power rail to 0
 - ❖ generate many power rails
- ❖ Try n
 - ❖ core utilization: 0.5
 - ❖ left, bottom, right and top io2core: 40
 - ❖ succeed

P&R Result

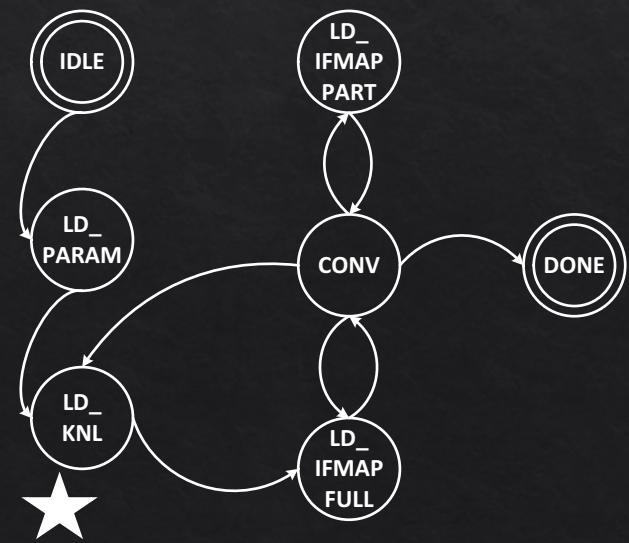
Workflow



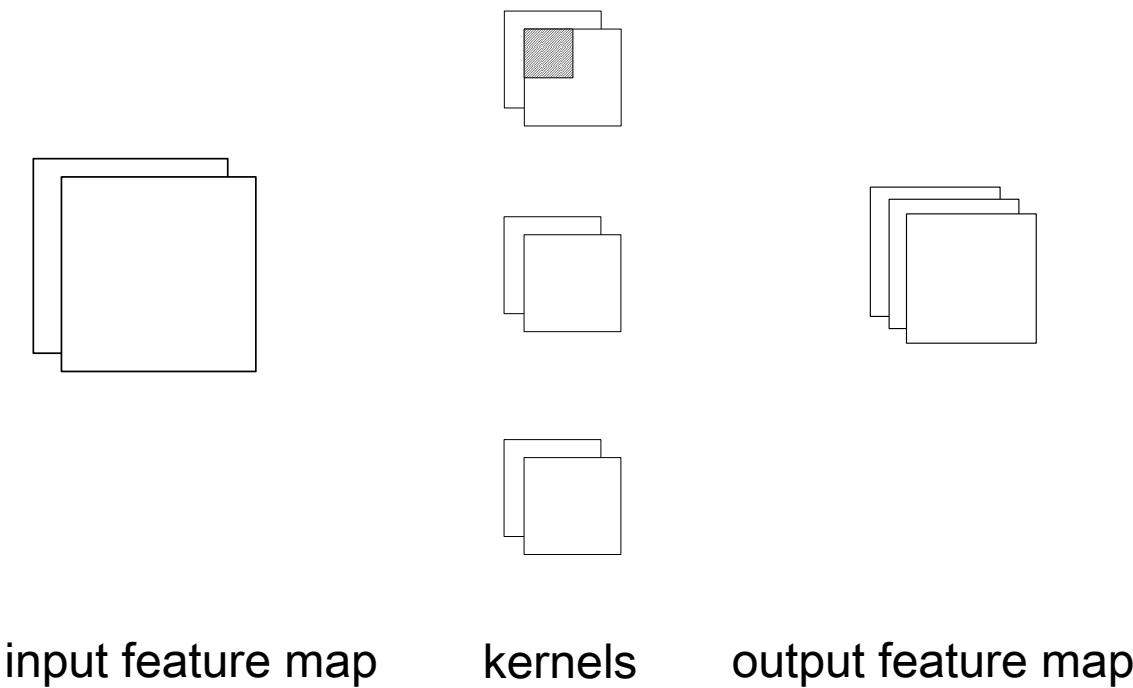
input feature map

kernels

output feature map

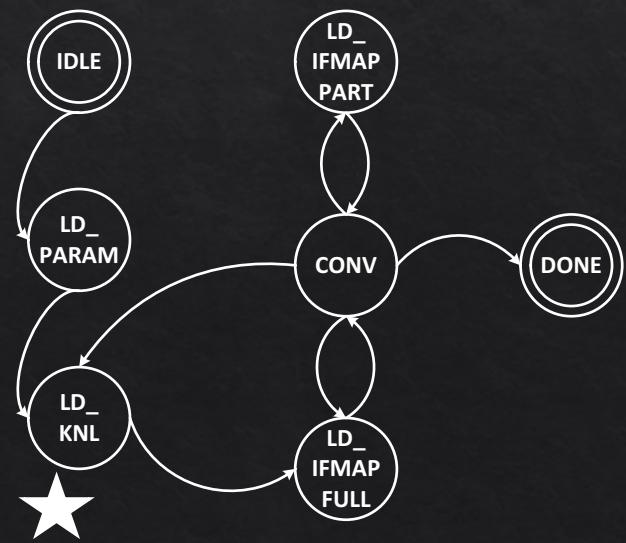


Workflow

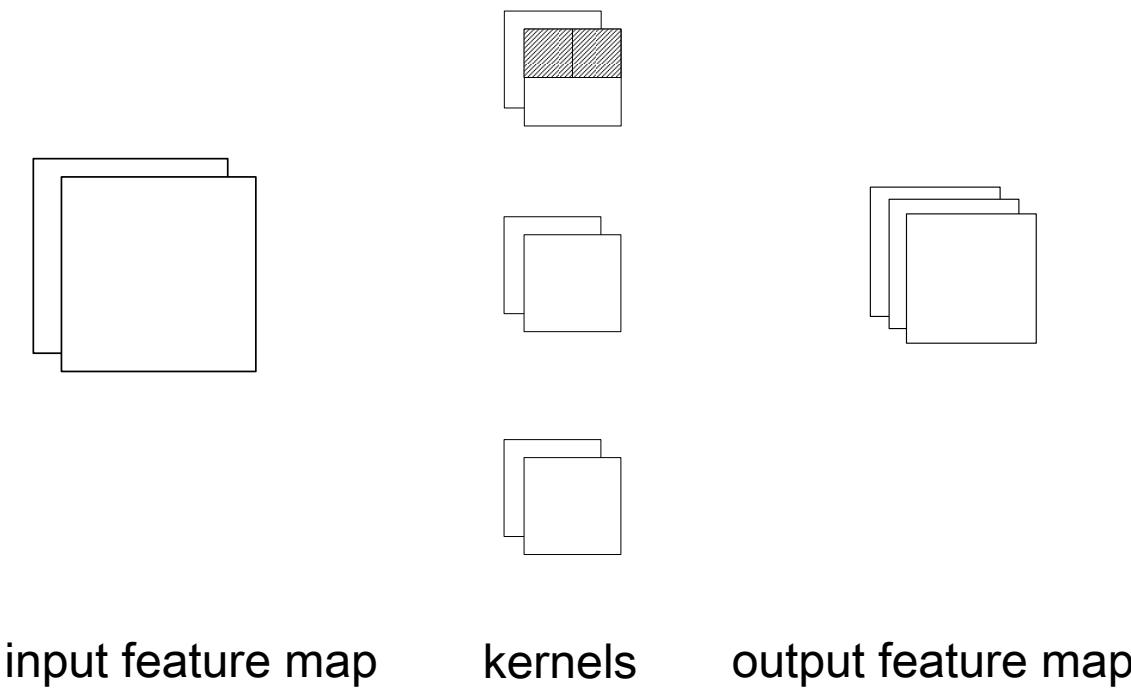


counters

knl_wts	0
knl_id	0
knl_chnl	0

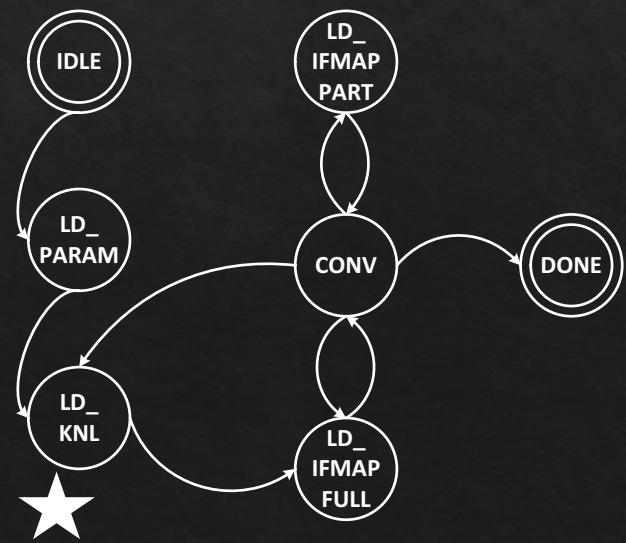


Workflow

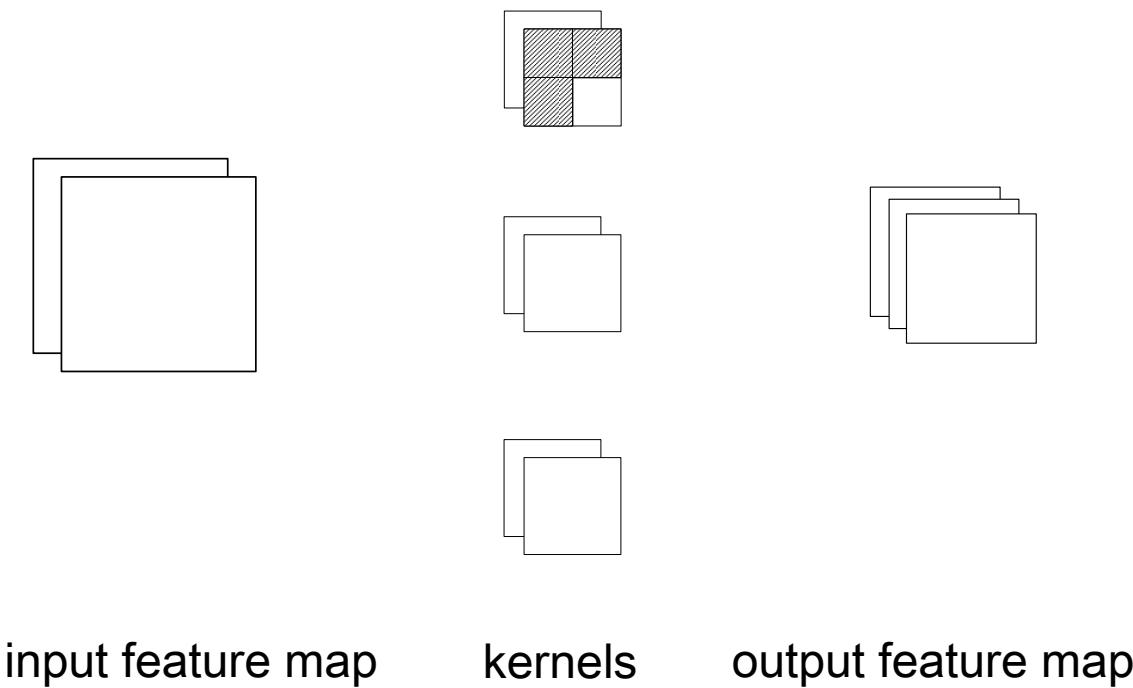


counters

knl_wts	1
knl_id	0
knl_chnl	0



Workflow



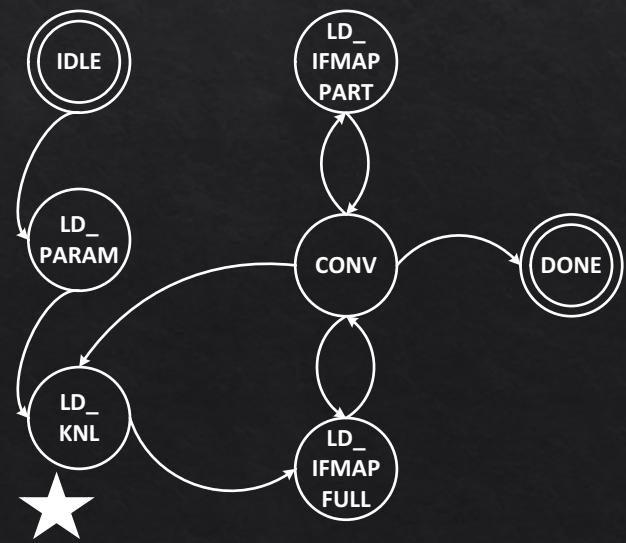
input feature map

kernels

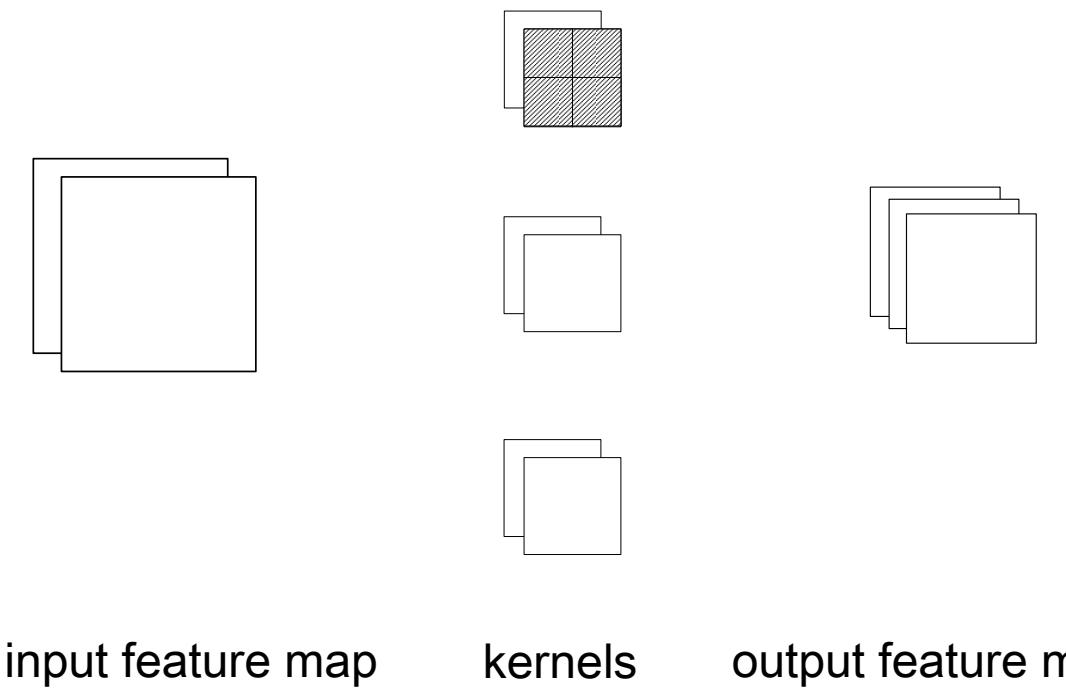
output feature map

counters

knl_wts	2
knl_id	0
knl_chnl	0

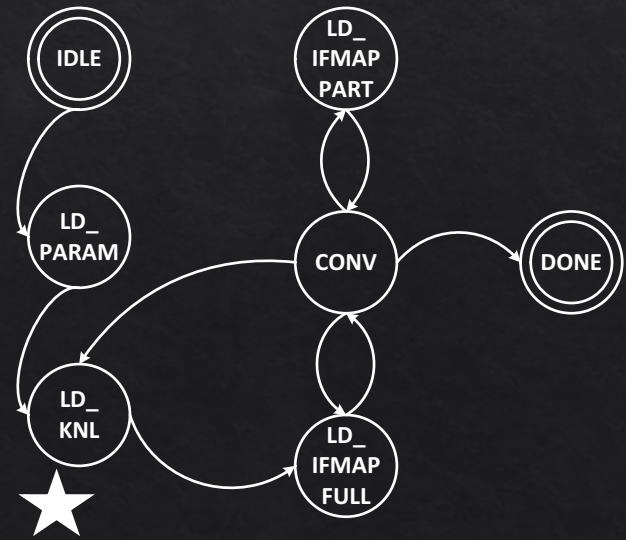


Workflow

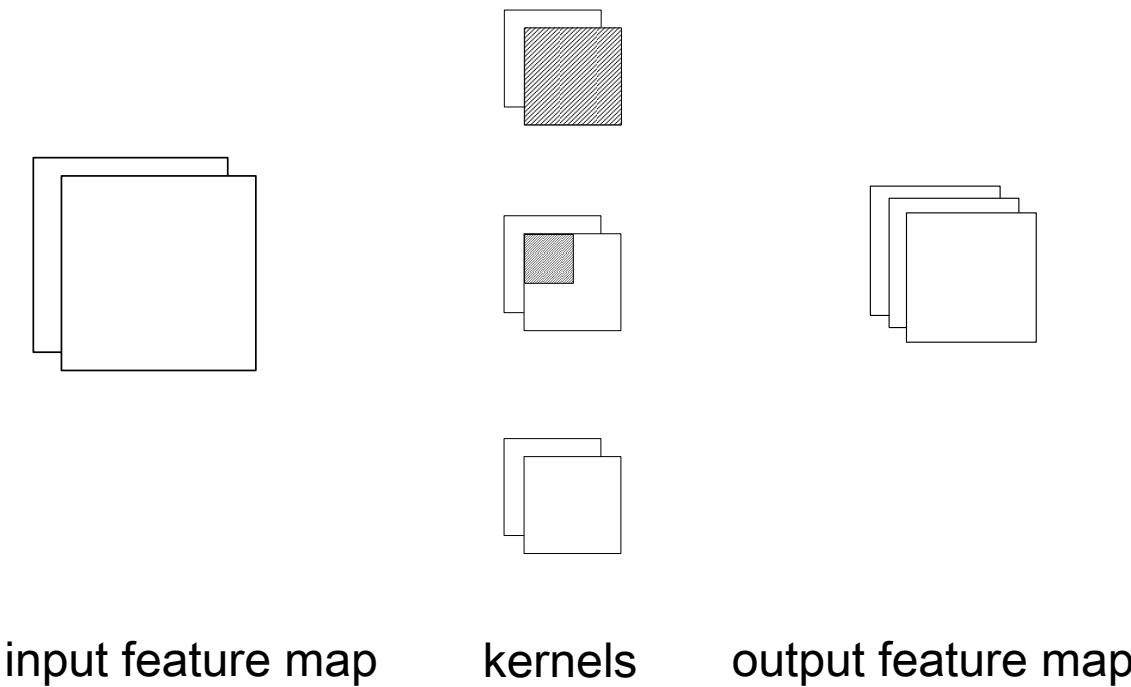


counters

knl_wts	3
knl_id	0
knl_chnl	0



Workflow



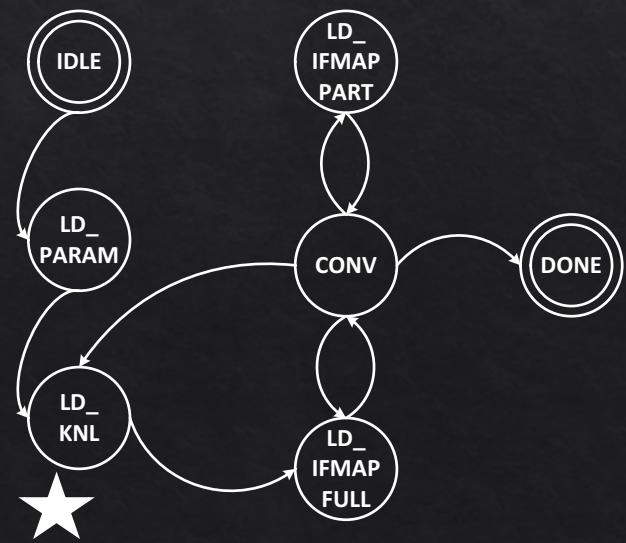
input feature map

kernels

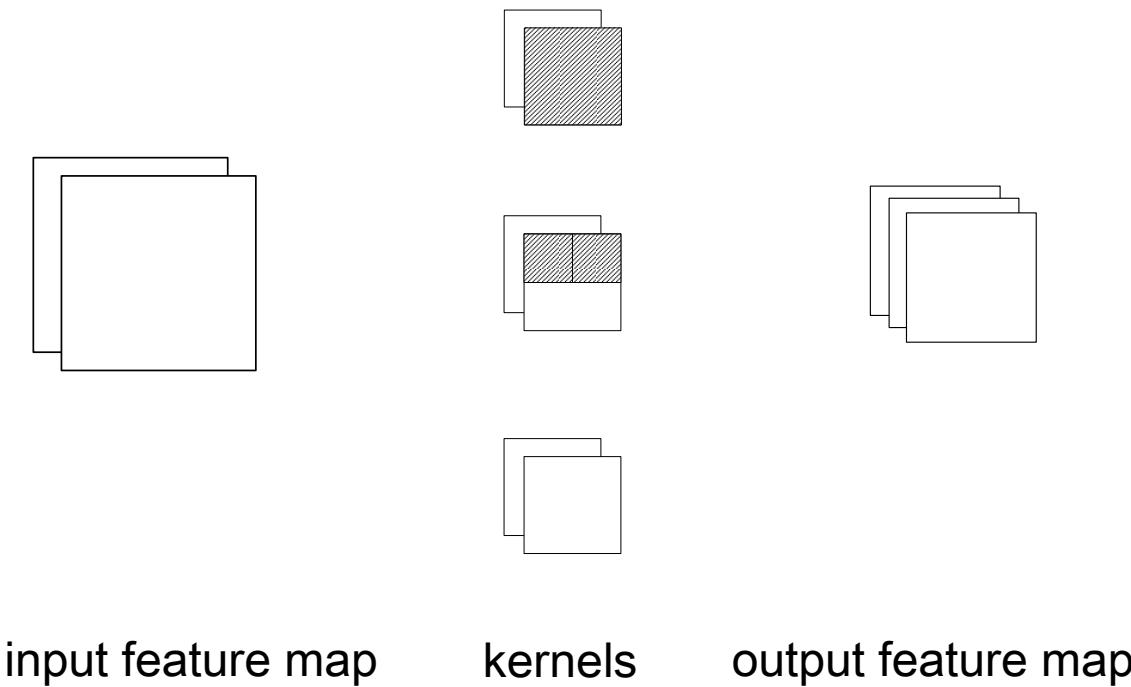
output feature map

counters

knl_wts	0
knl_id	1
knl_chnl	0

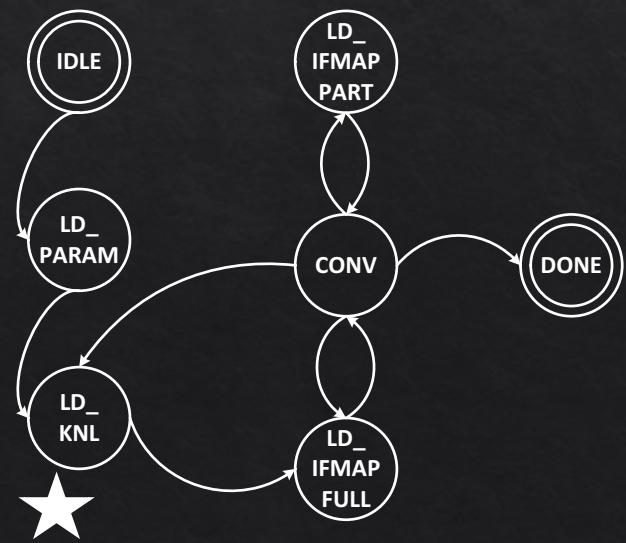


Workflow

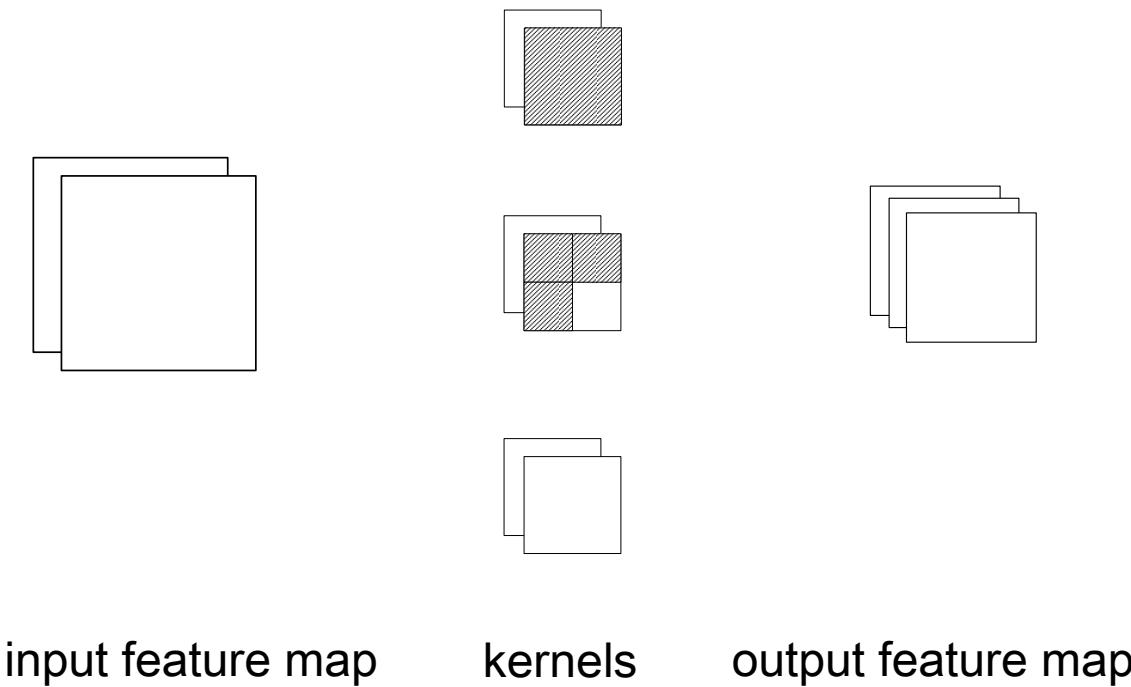


counters

knl_wts	1
knl_id	1
knl_chnl	0

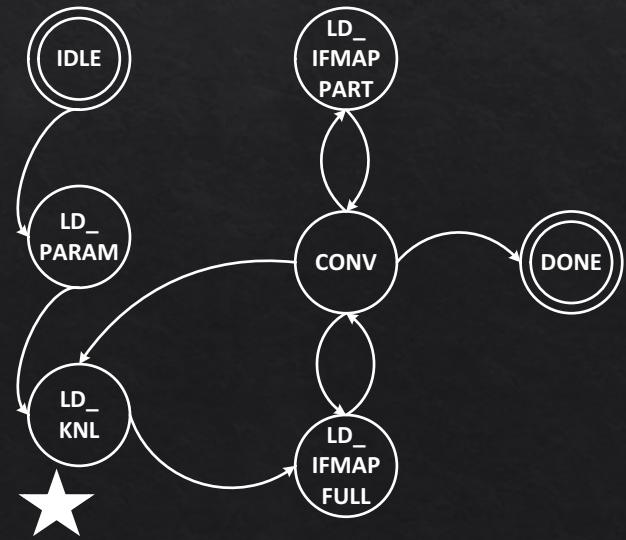


Workflow

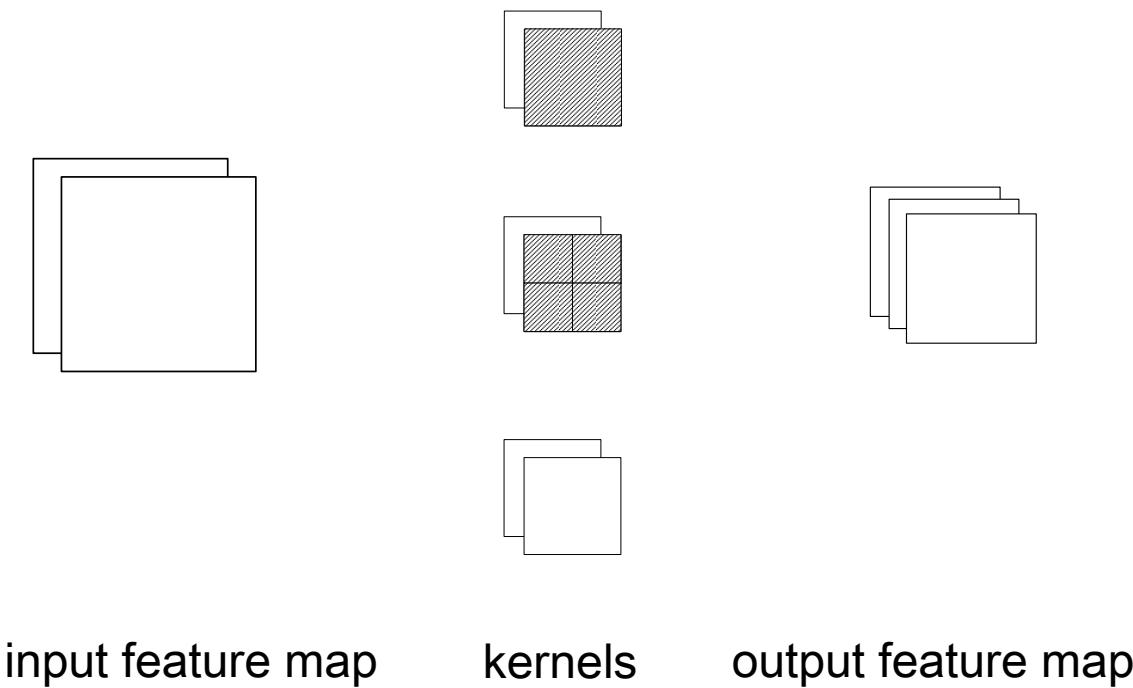


counters

knl_wts	2
knl_id	1
knl_chnl	0

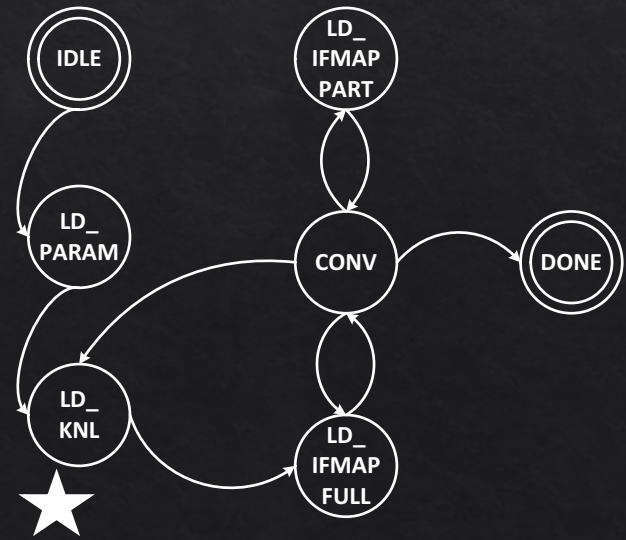


Workflow

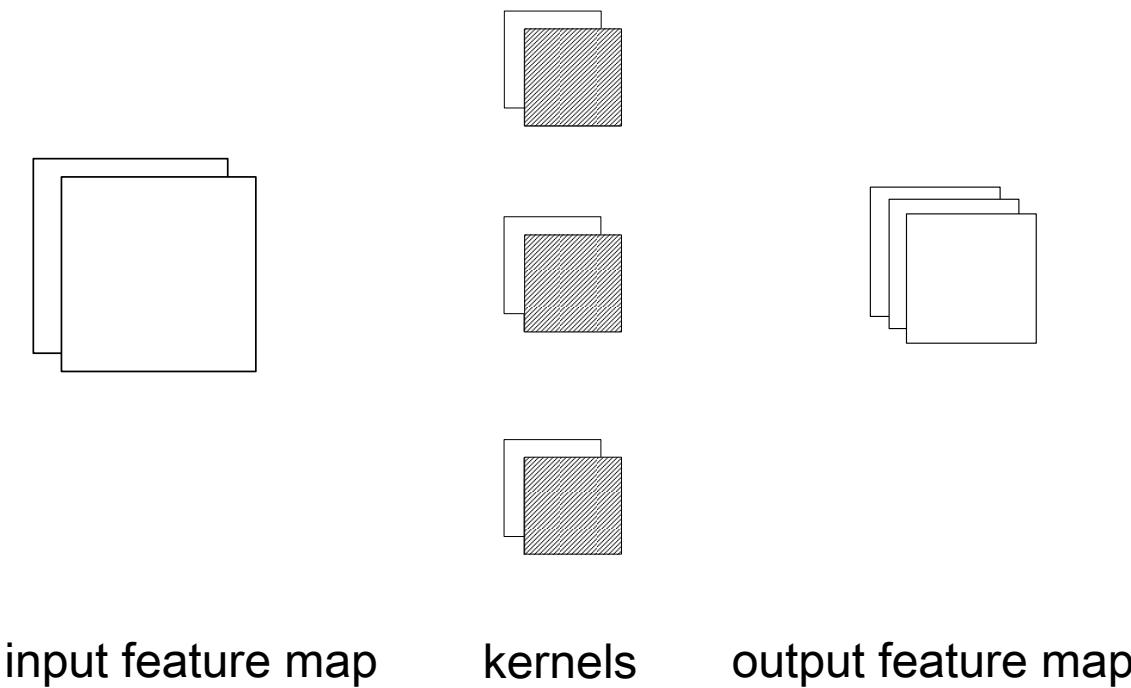


counters

knl_wts	3
knl_id	1
knl_chnl	0

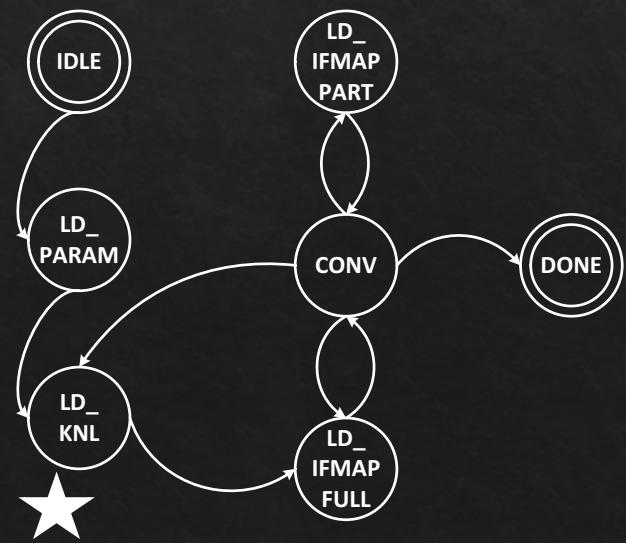


Workflow

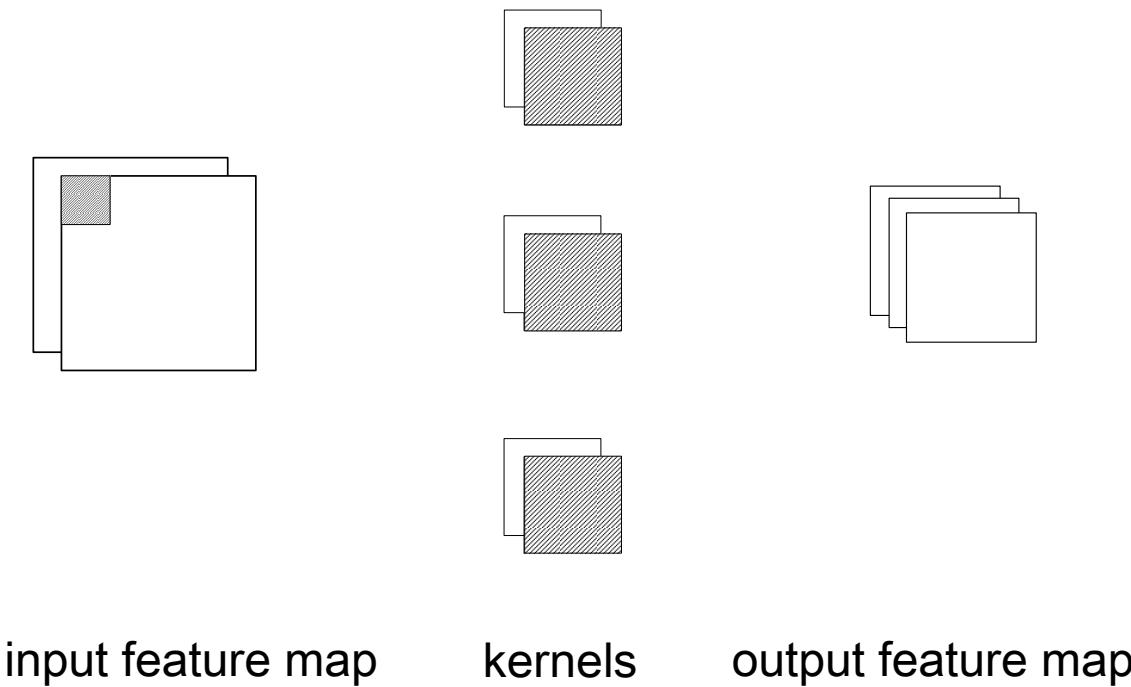


counters

knl_wts	3
knl_id	2
knl_chnl	0

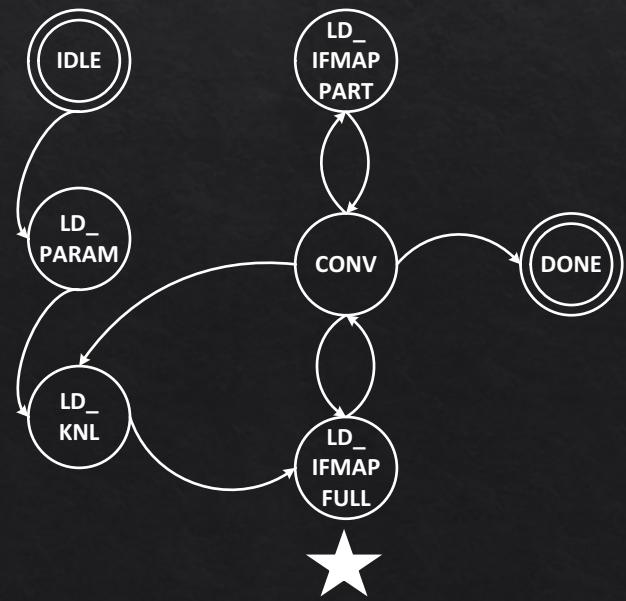


Workflow

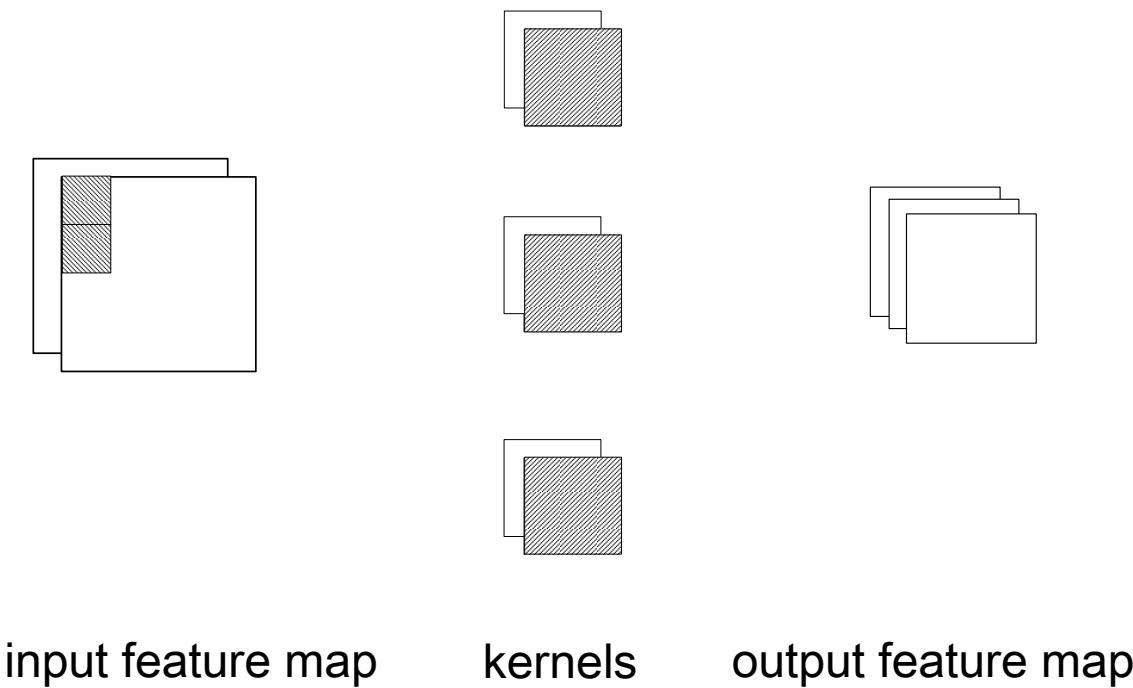


counters

base_x	0
base_y	0
delta_x	0
delta_y	0
ifmap_chnl	0

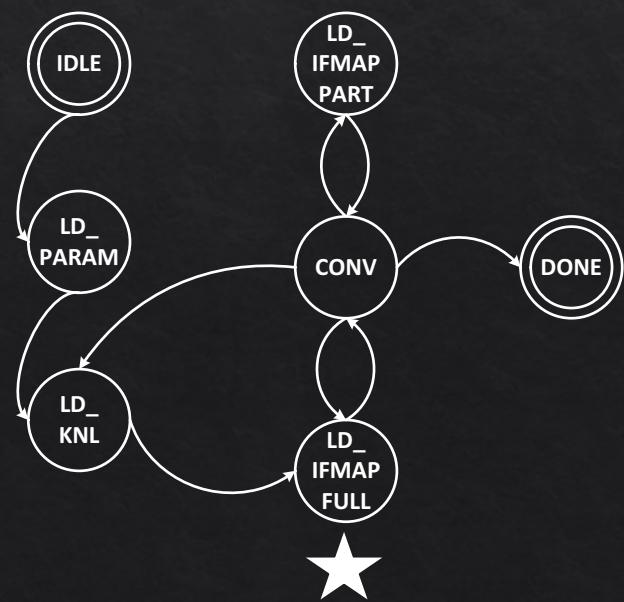


Workflow

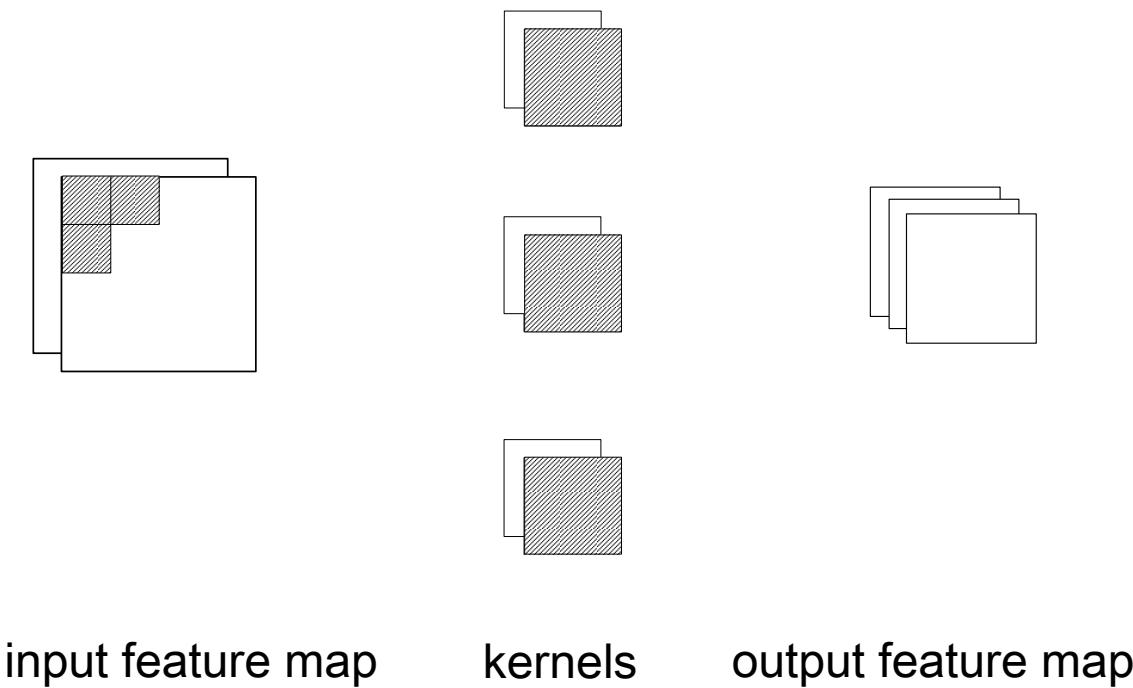


counters

base_x	0
base_y	0
delta_x	0
delta_y	1
ifmap_chnl	0



Workflow



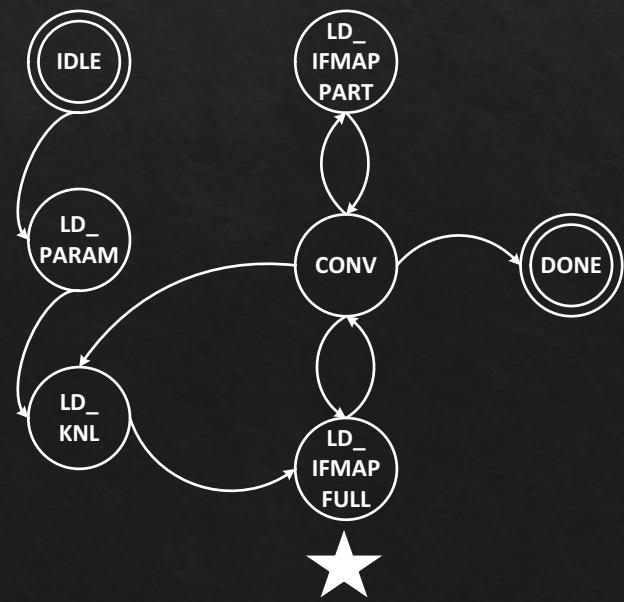
input feature map

kernels

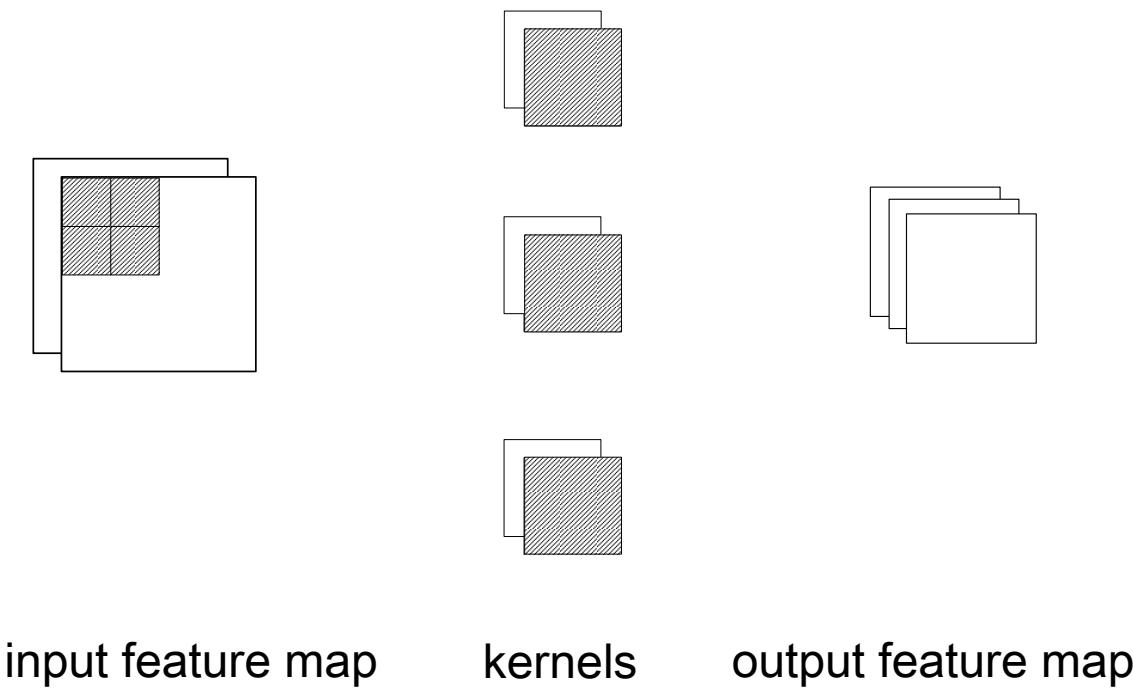
output feature map

counters

base_x	0
base_y	0
delta_x	1
delta_y	0
ifmap_chnl	0

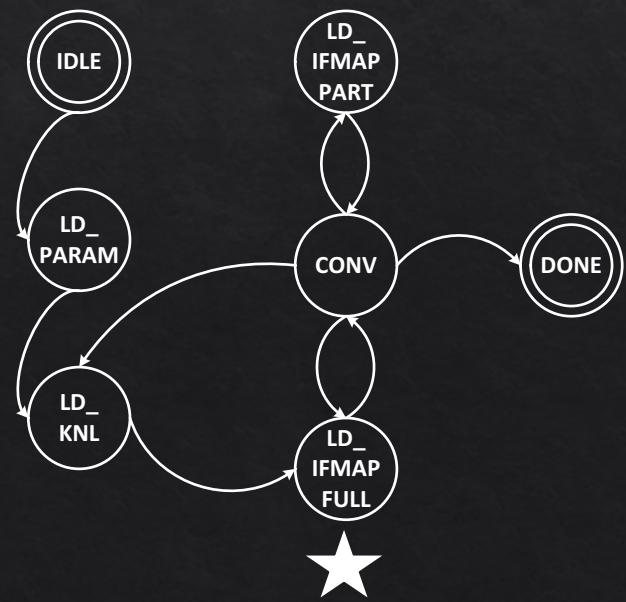


Workflow

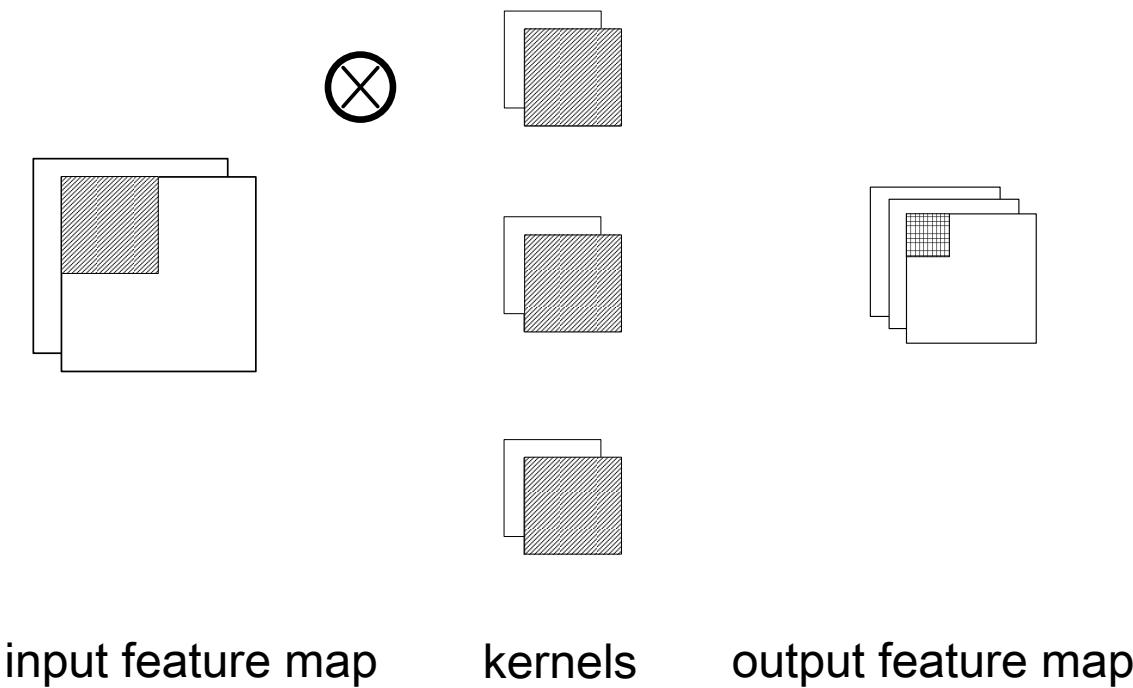


counters

base_x	0
base_y	0
delta_x	1
delta_y	1
ifmap_chnl	0

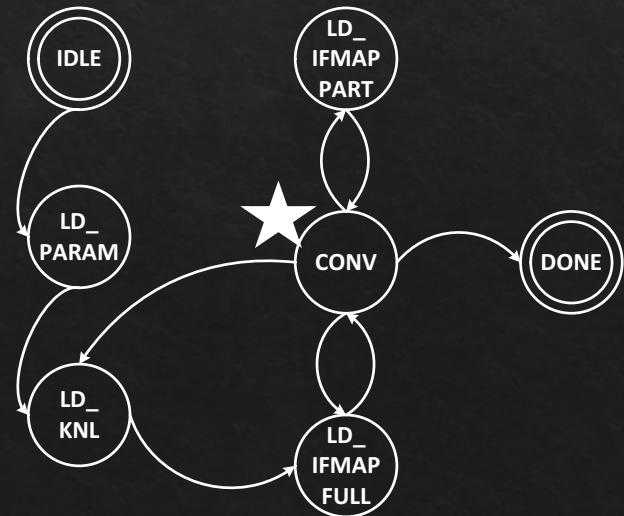


Workflow

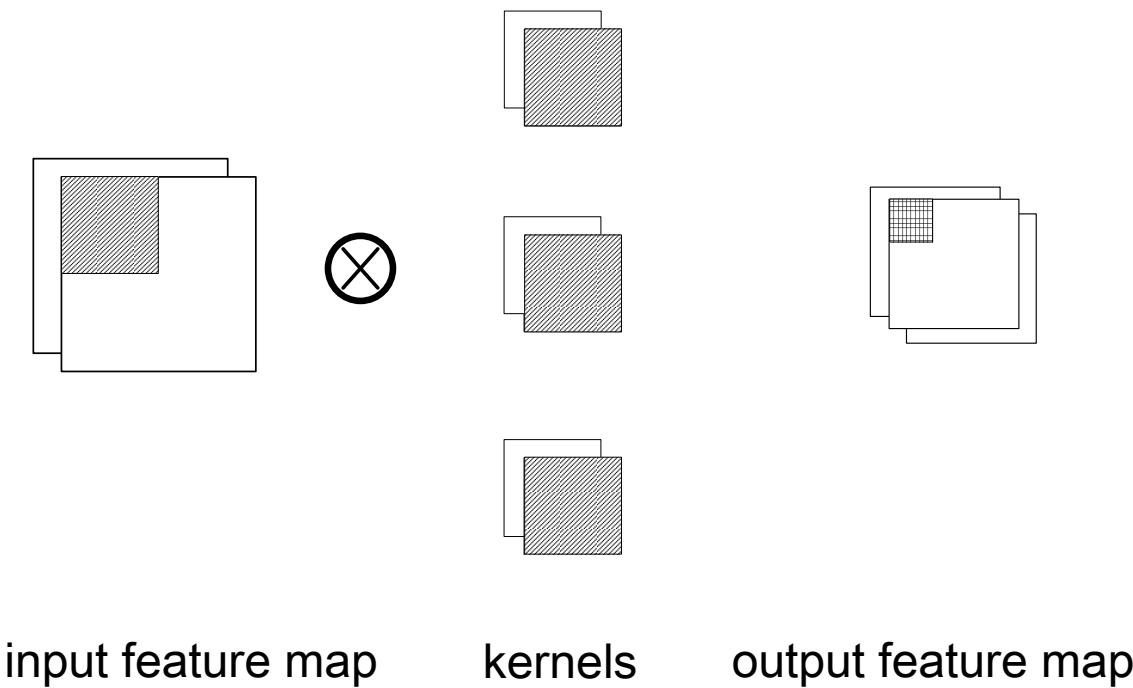


counters

base_x	0
base_y	0
ofmap_chnl	0

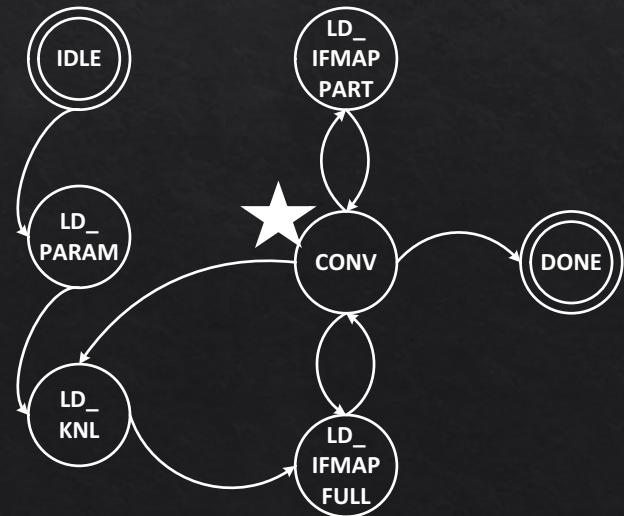


Workflow

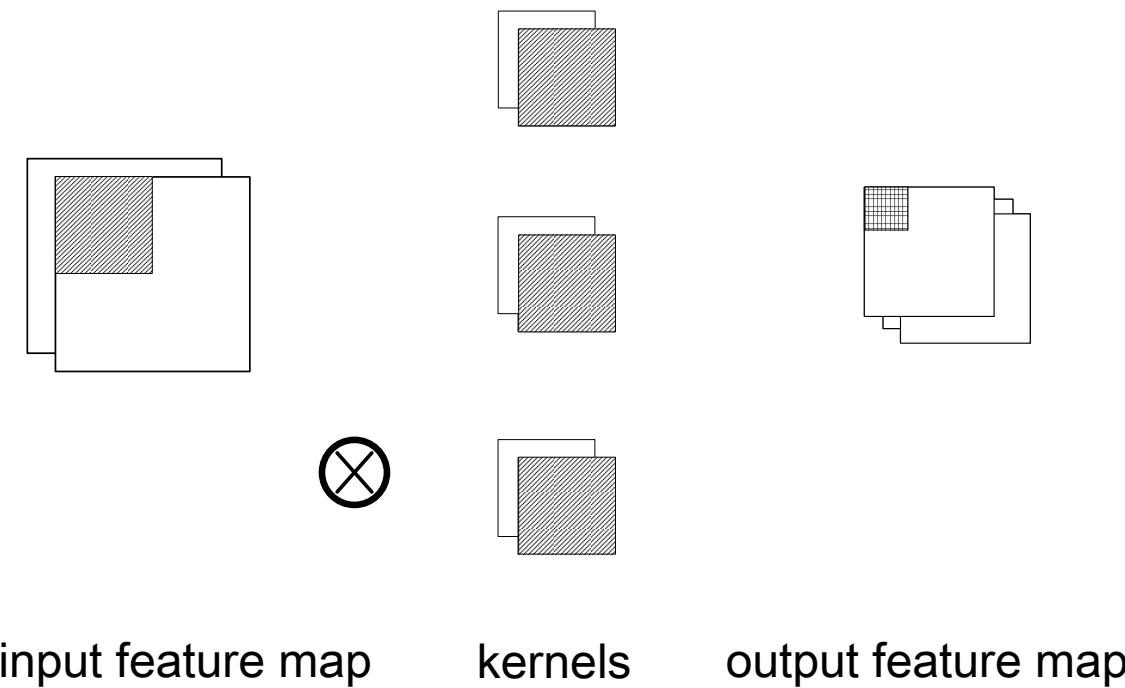


counters

base_x	0
base_y	0
ofmap_chnl	1

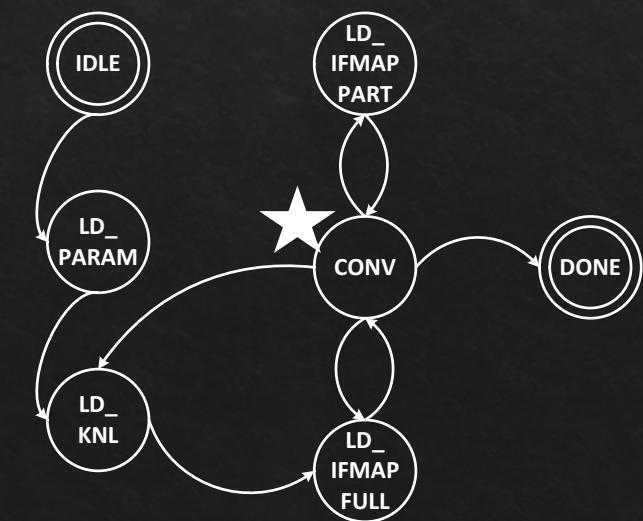


Workflow

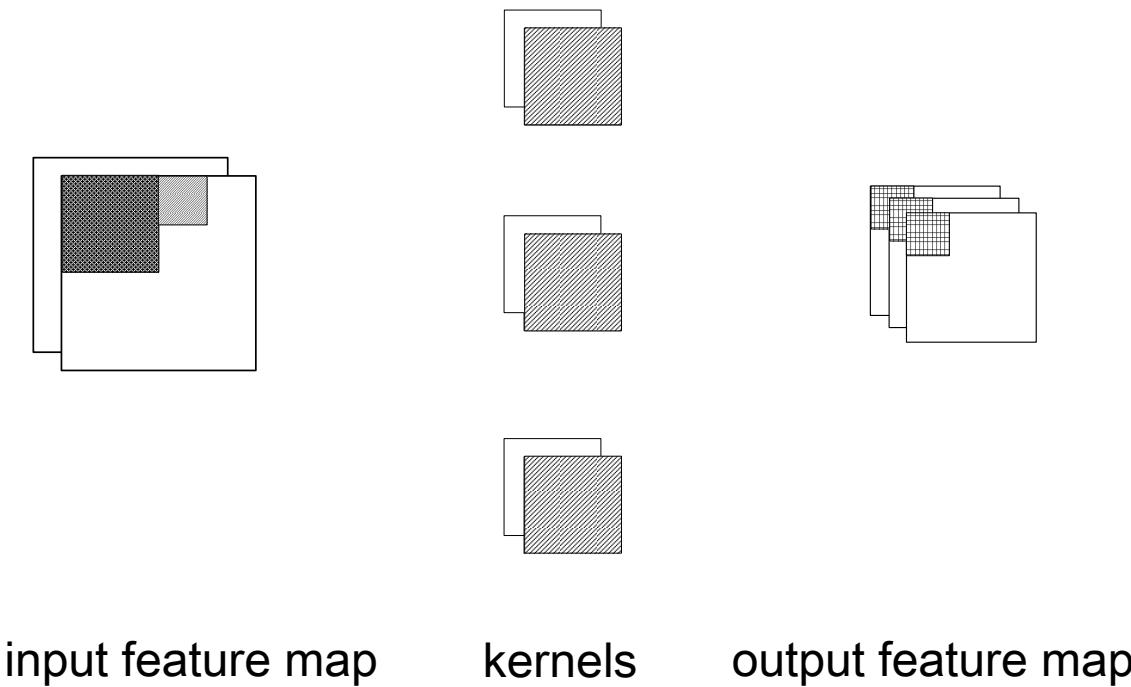


counters

base_x	0
base_y	0
ofmap_chnl	2

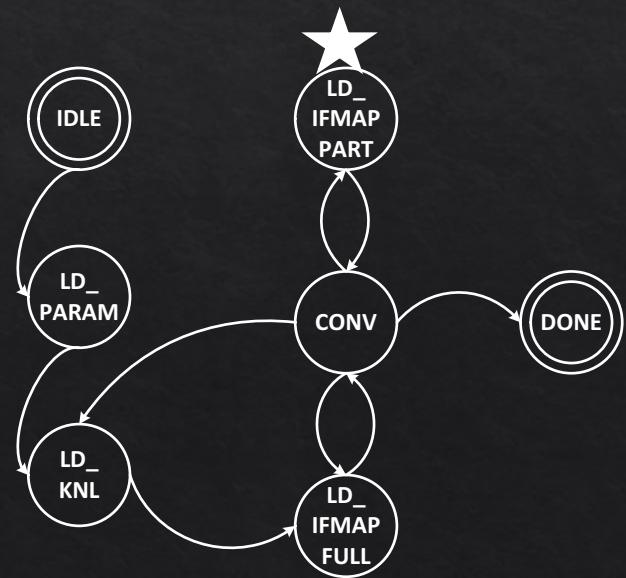


Workflow

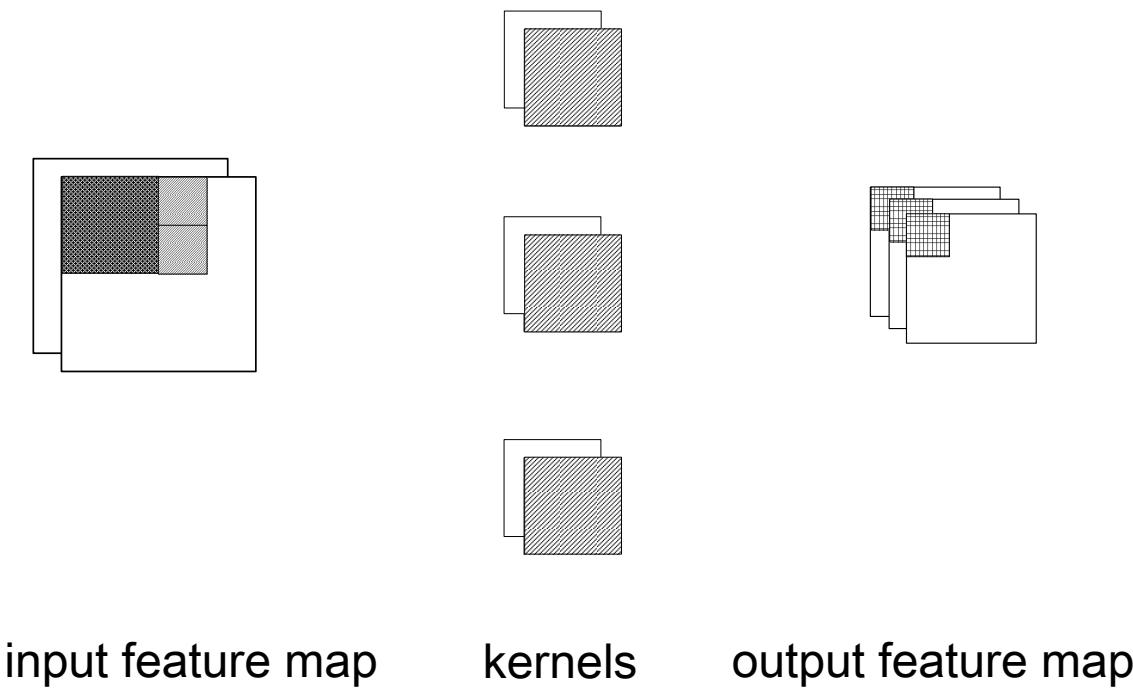


counters

base_x	1
base_y	0
delta_x	0
delta_y	0
ifmap_chnl	0

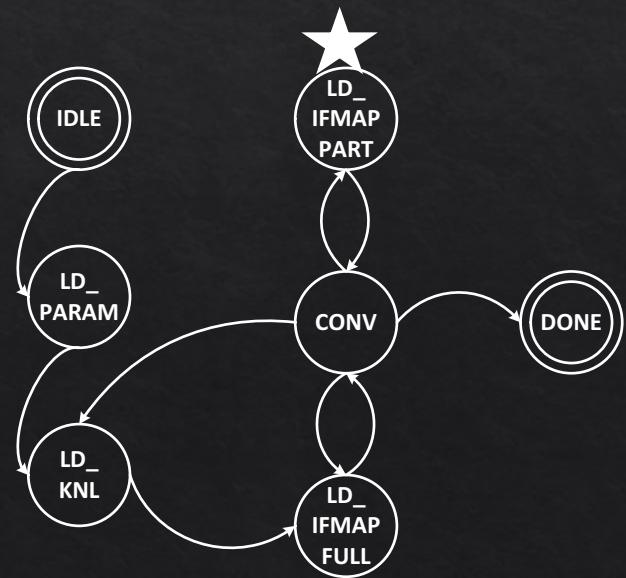


Workflow

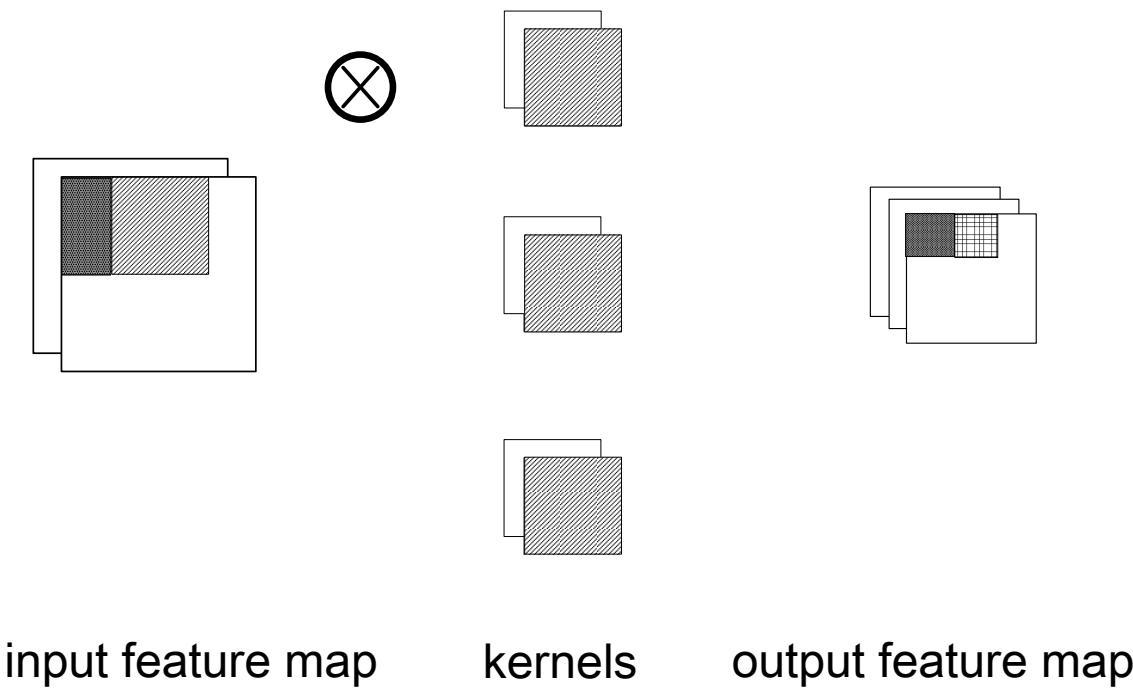


counters

base_x	1
base_y	0
delta_x	0
delta_y	1
ifmap_chnl	0

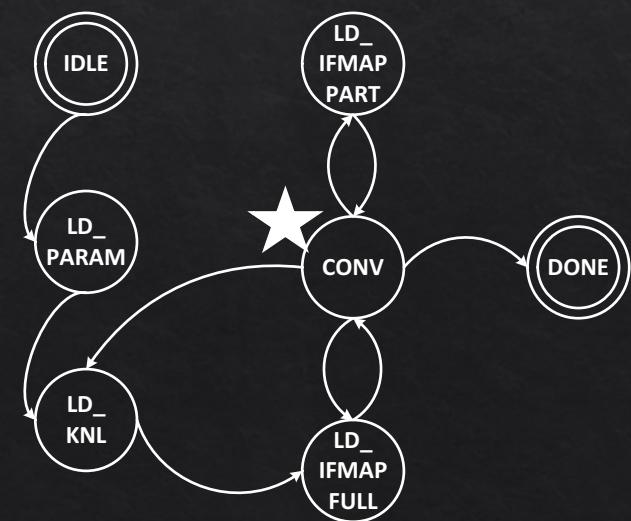


Workflow

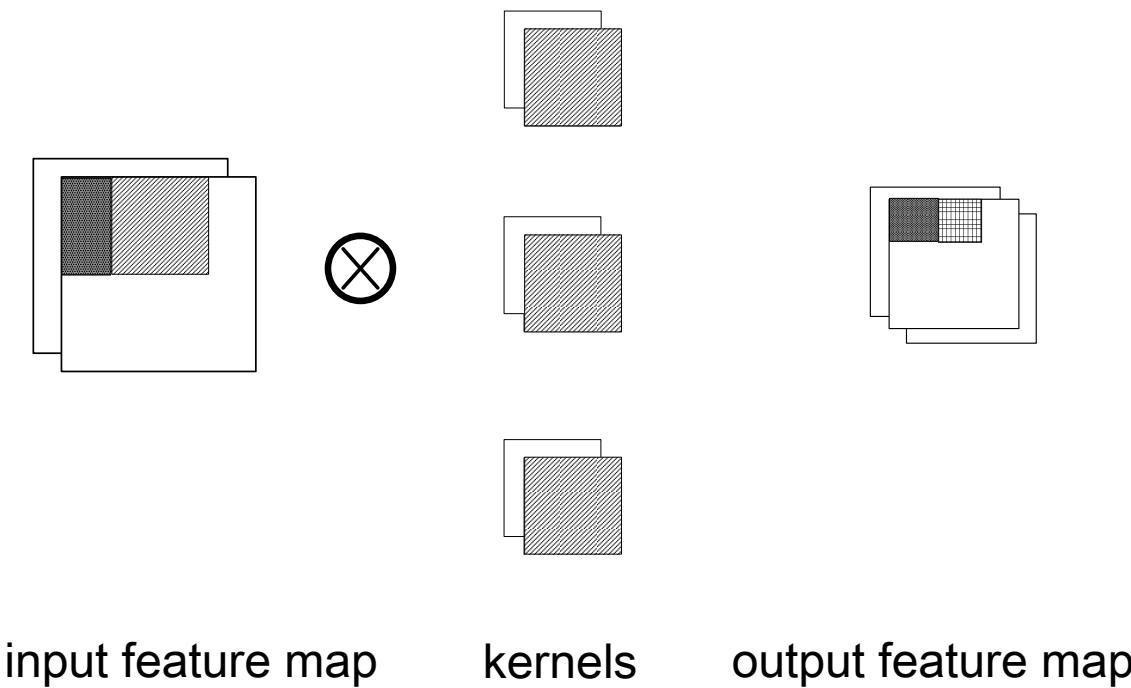


counters

base_x	1
base_y	0
ofmap_chnl	0

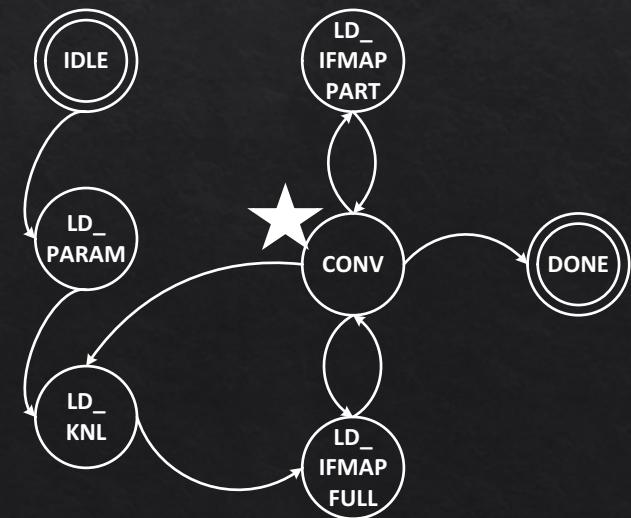


Workflow

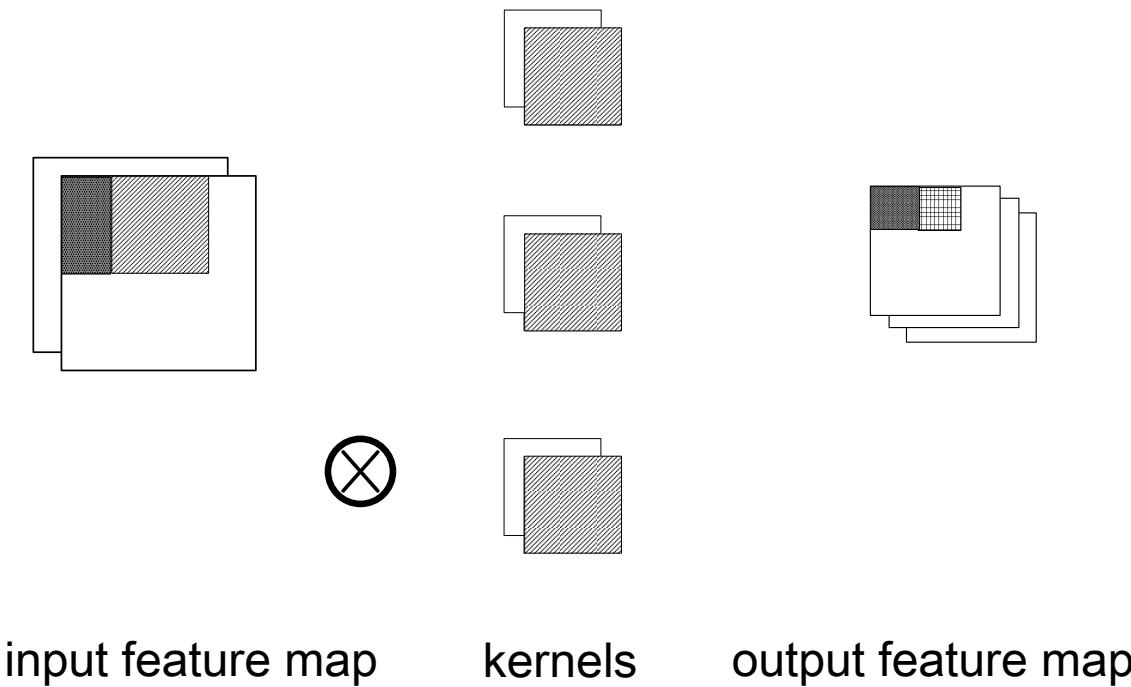


counters

base_x	1
base_y	0
ofmap_chnl	0

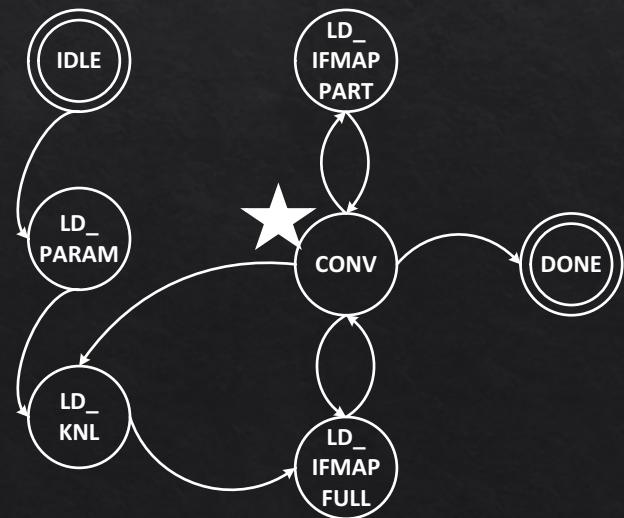


Workflow

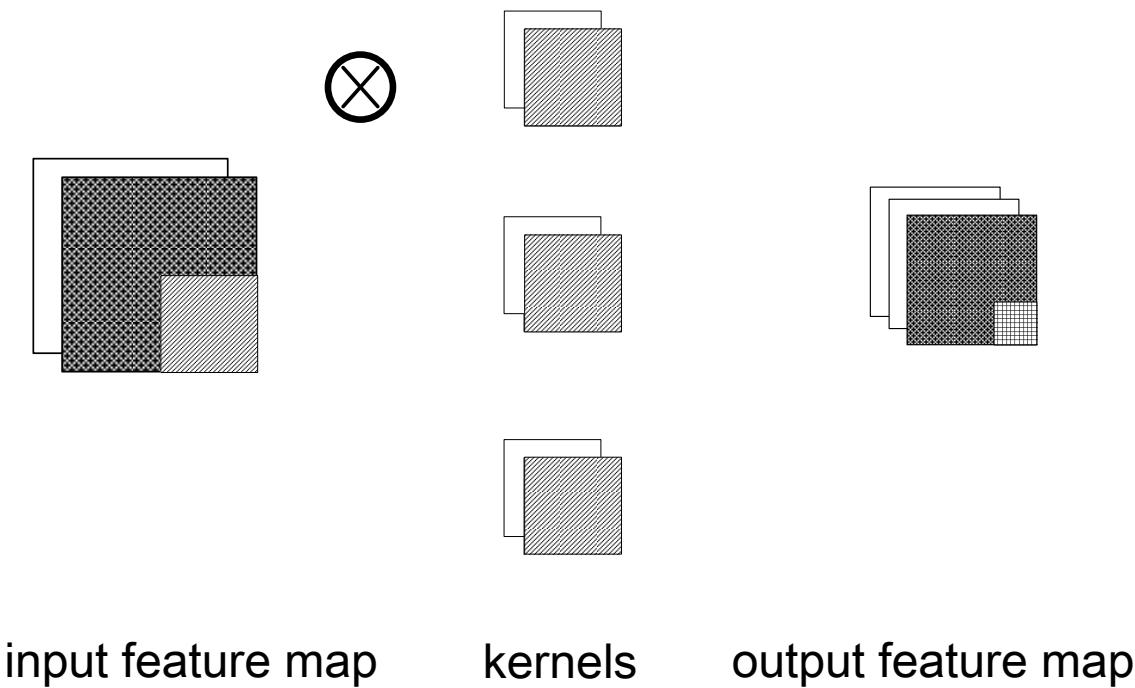


counters

base_x	1
base_y	0
ofmap_chnl	0

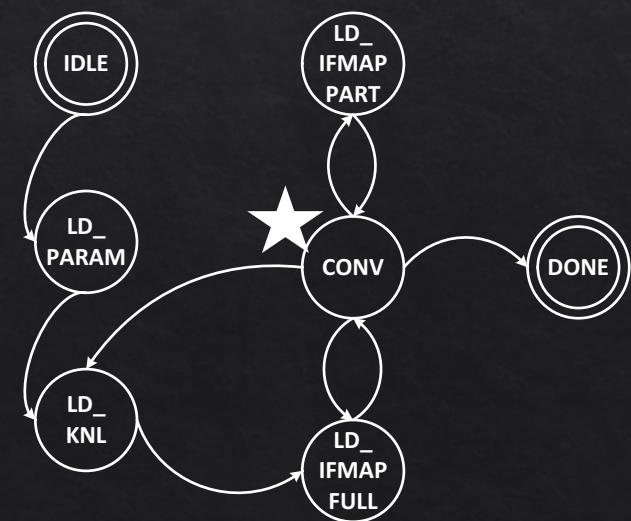


Workflow

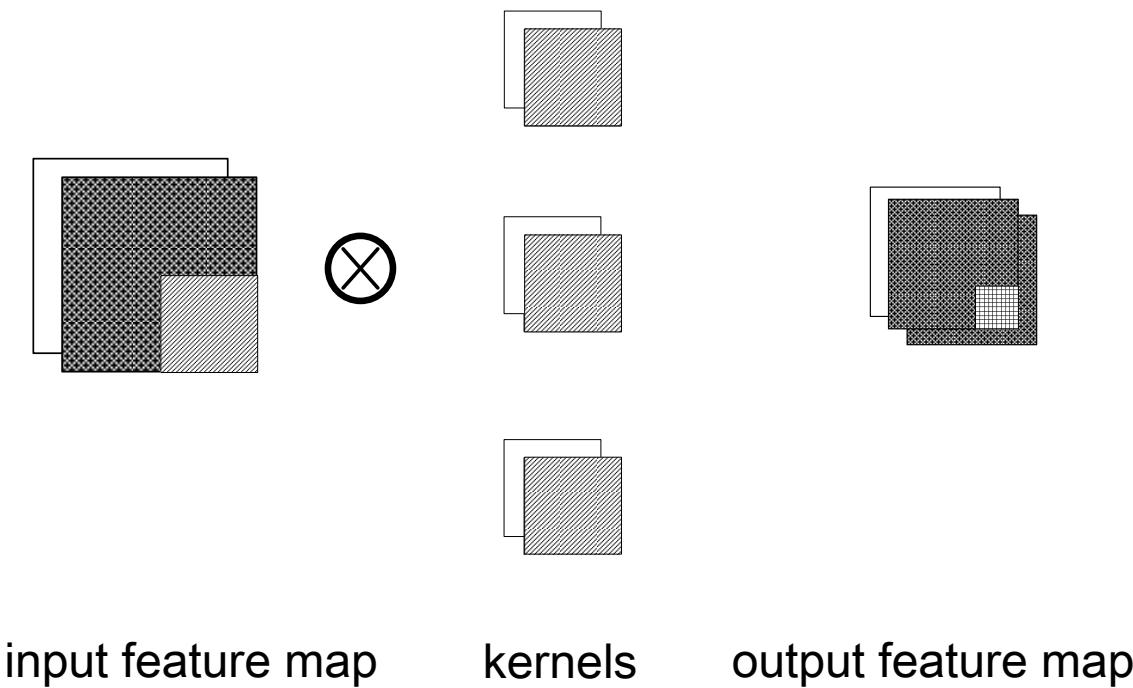


counters

base_x	2
base_y	2
ofmap_chnl	0

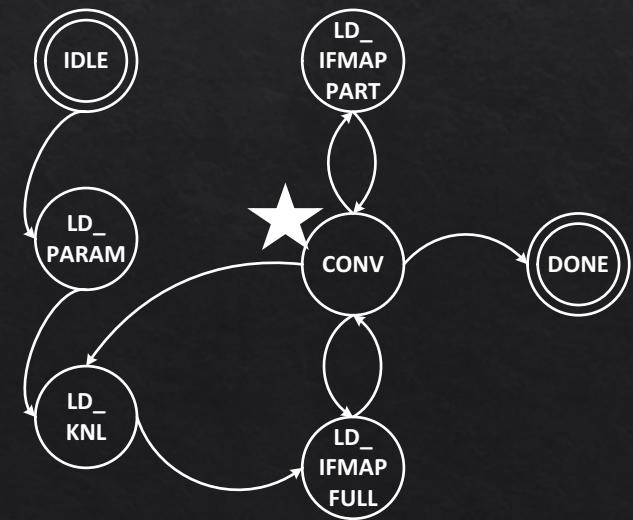


Workflow

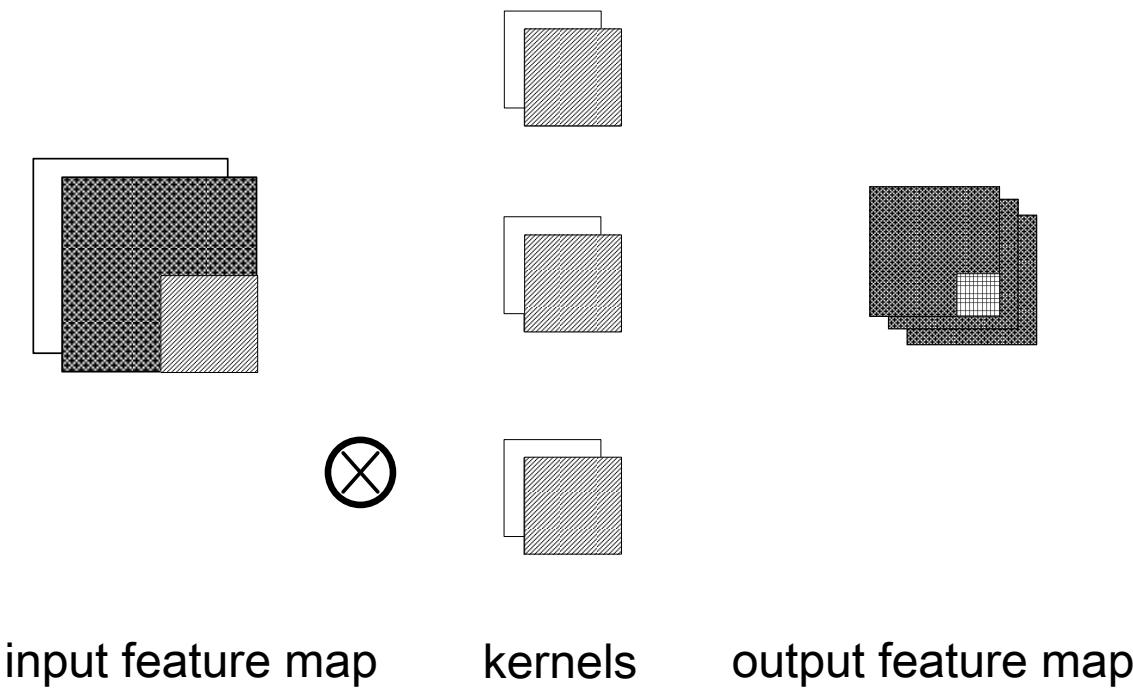


counters

base_x	2
base_y	2
ofmap_chnl	1

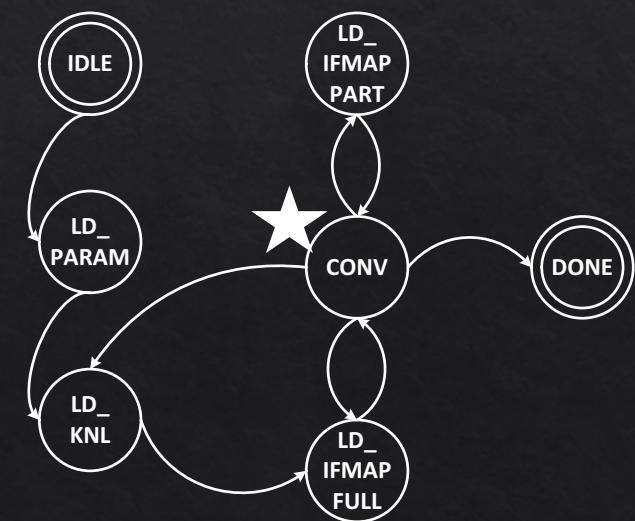


Workflow

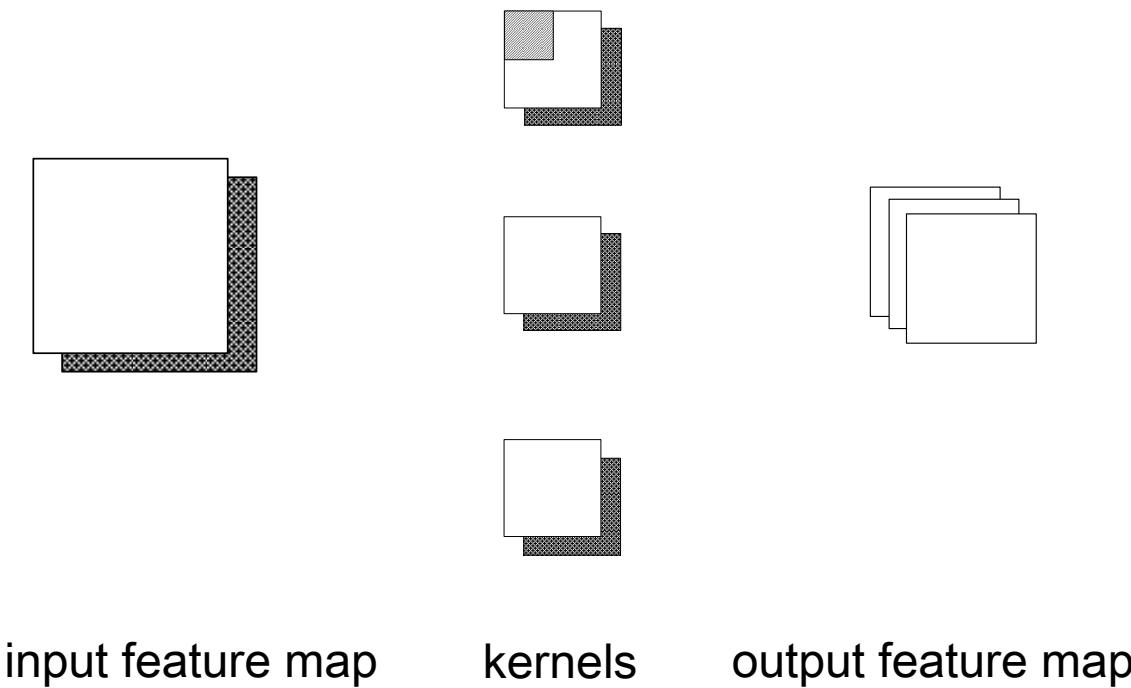


counters

base_x	2
base_y	2
ofmap_chnl	2



Workflow



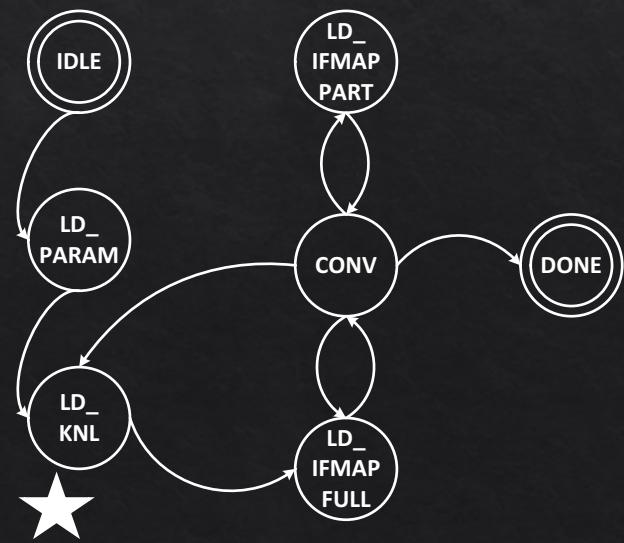
input feature map

kernels

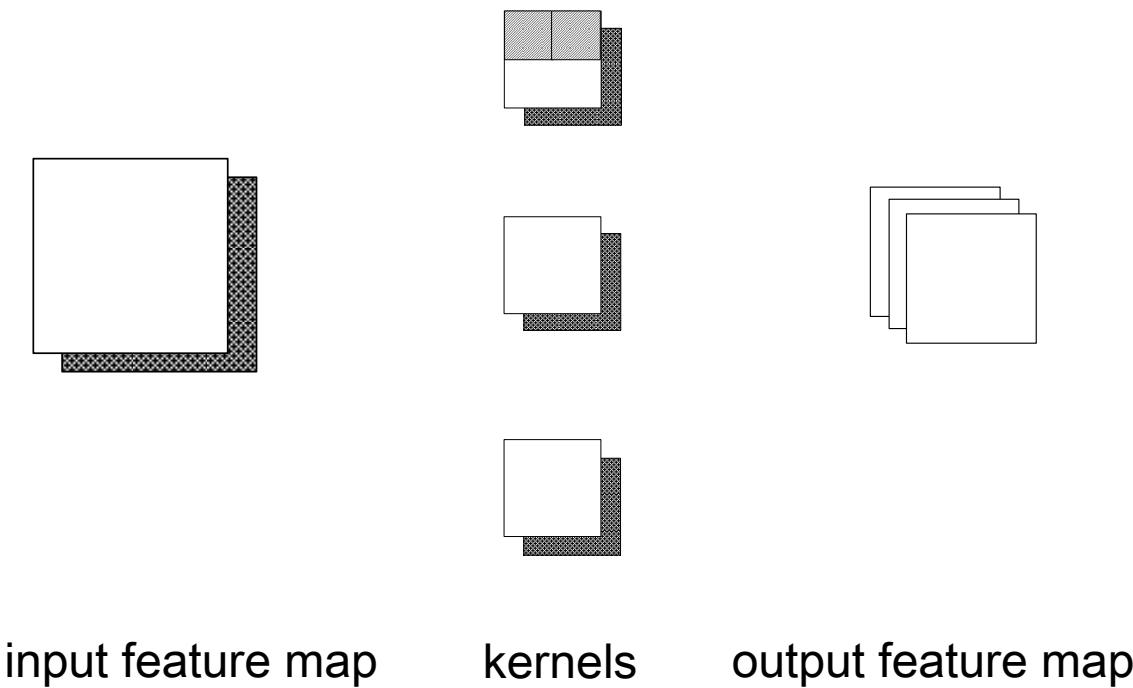
output feature map

counters

knl_wts	0
knl_id	0
ifmap_chnl	1



Workflow



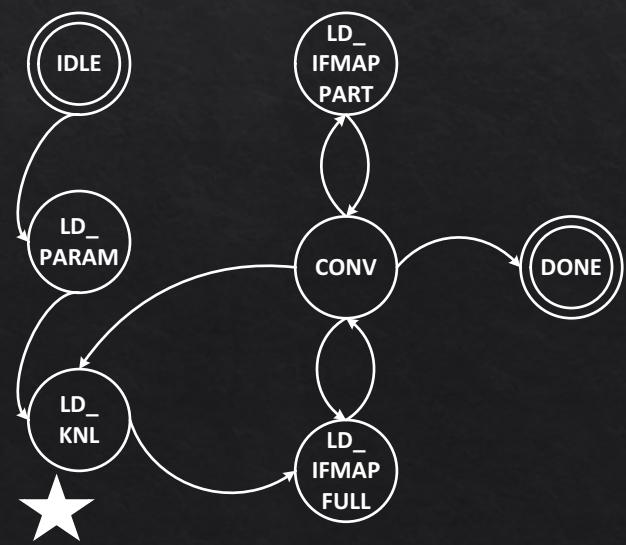
input feature map

kernels

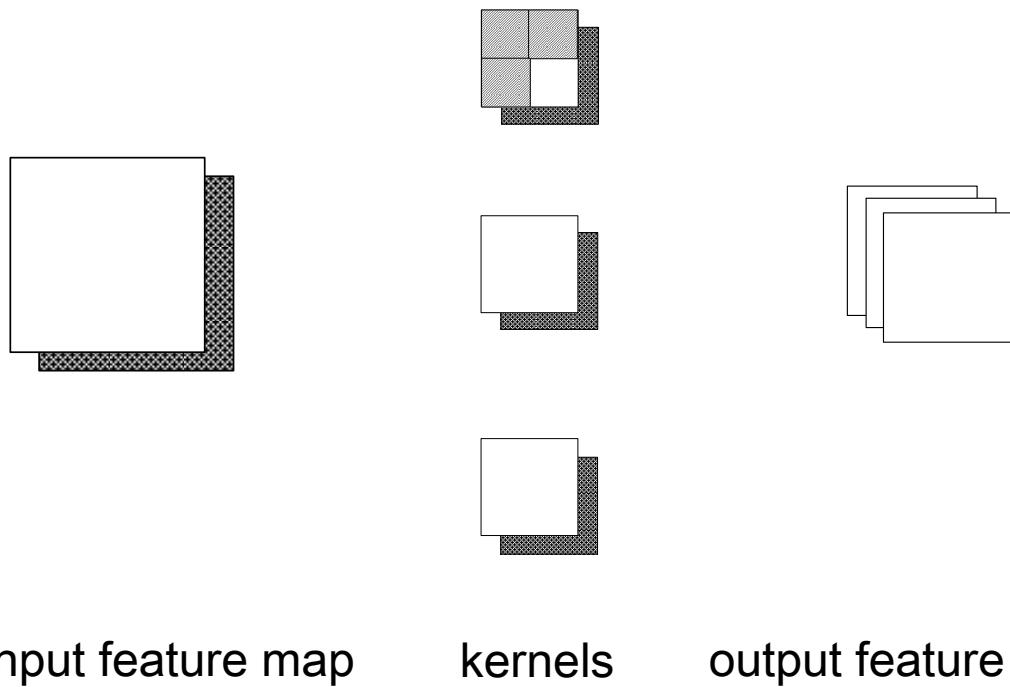
output feature map

counters

knl_wts	1
knl_id	0
ifmap_chnl	1



Workflow



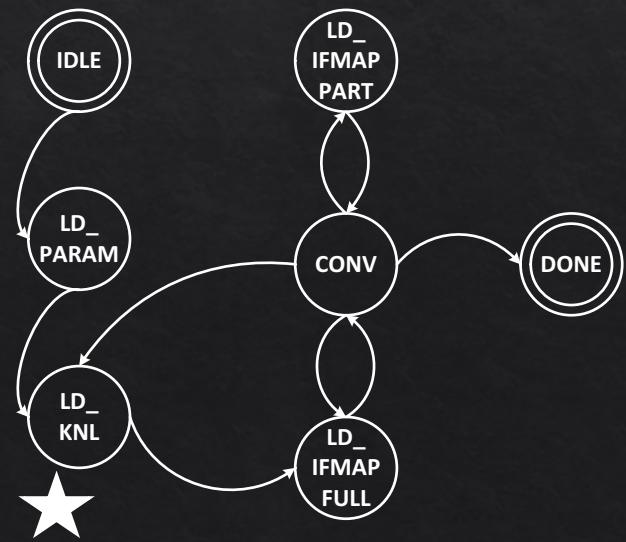
input feature map

kernels

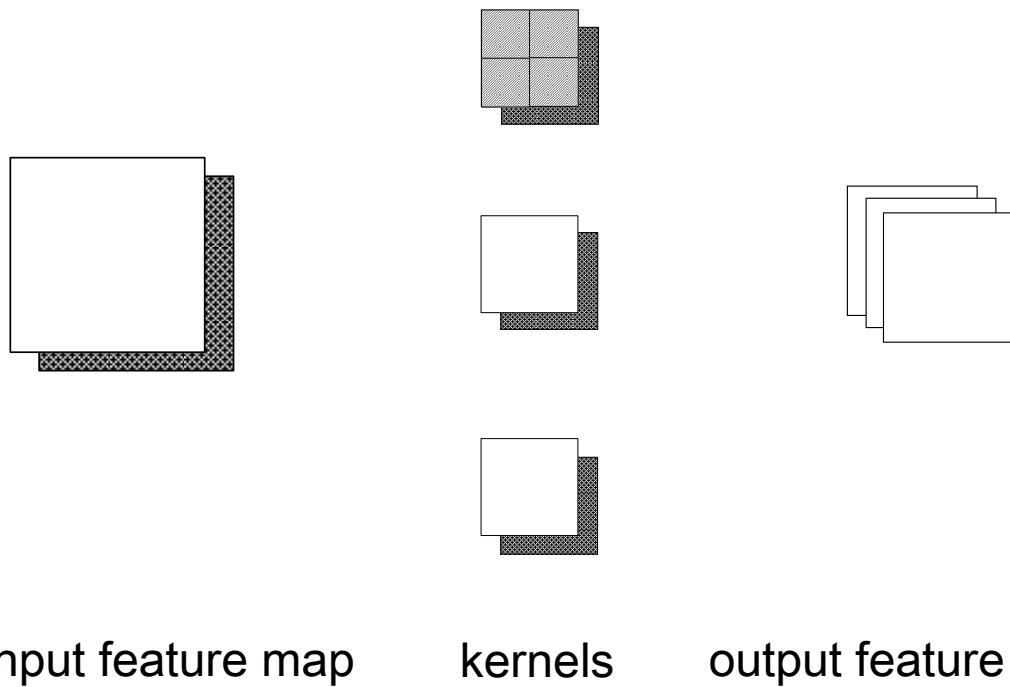
output feature map

counters

knl_wts	2
knl_id	0
ifmap_chnl	1



Workflow



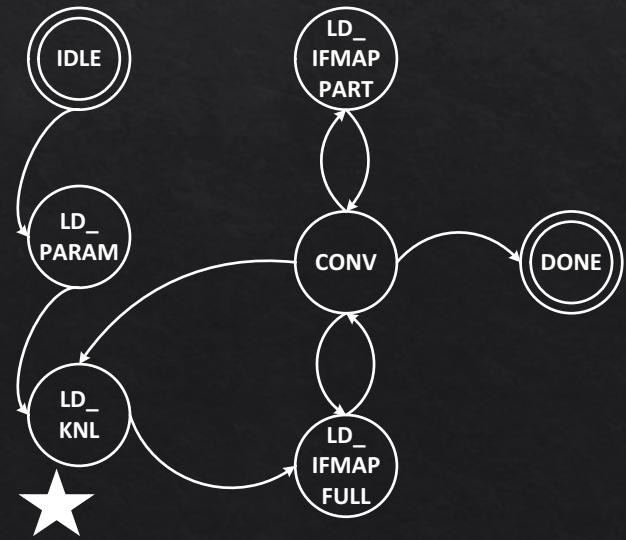
input feature map

kernels

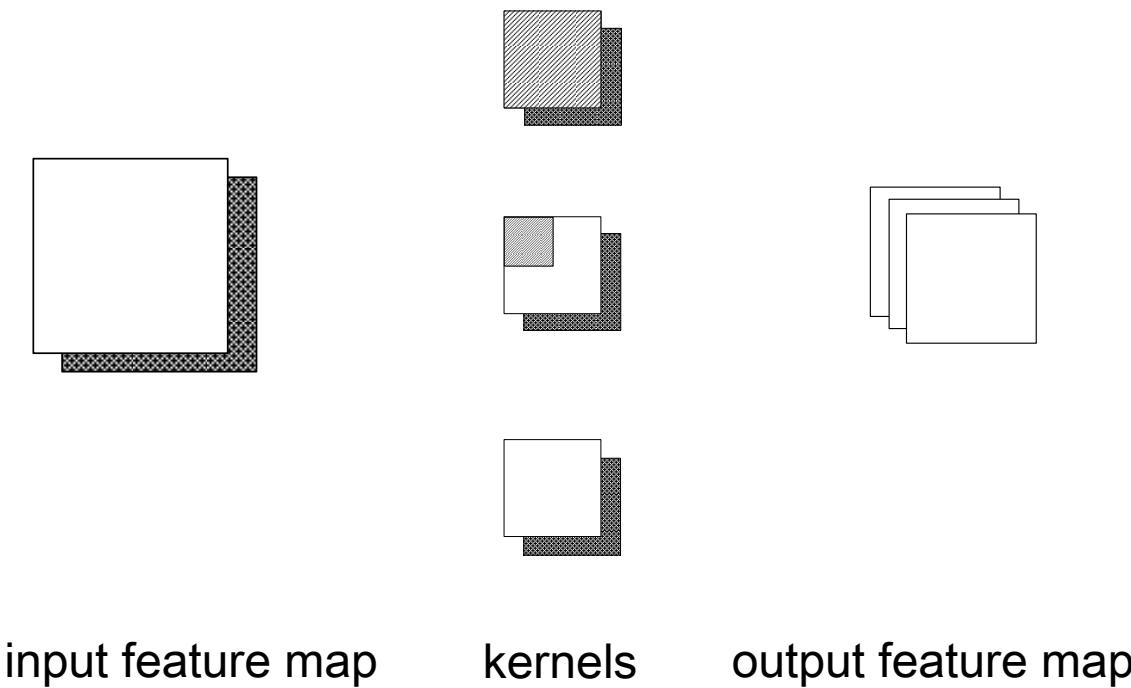
output feature map

counters

knl_wts	3
knl_id	0
ifmap_chnl	1



Workflow



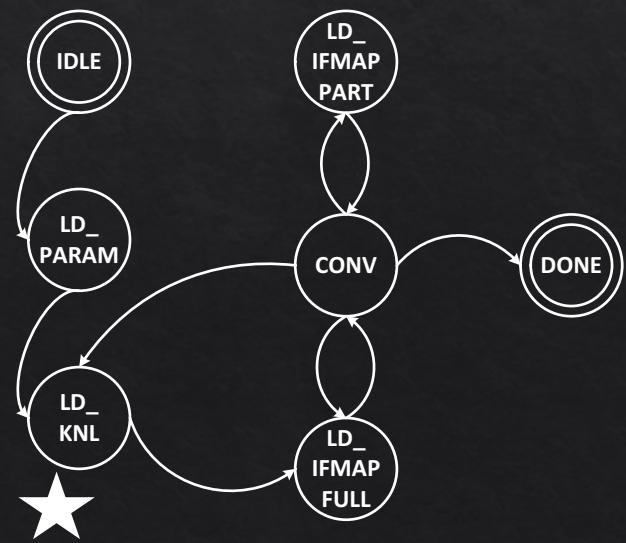
input feature map

kernels

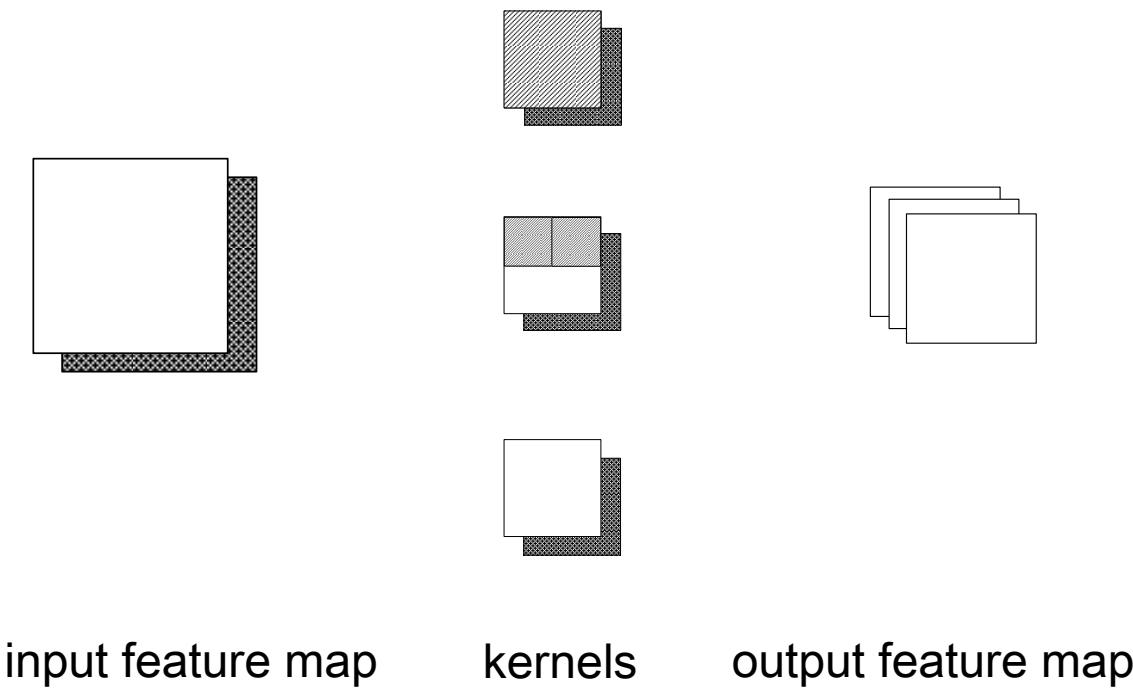
output feature map

counters

knl_wts	0
knl_id	1
ifmap_chnl	1

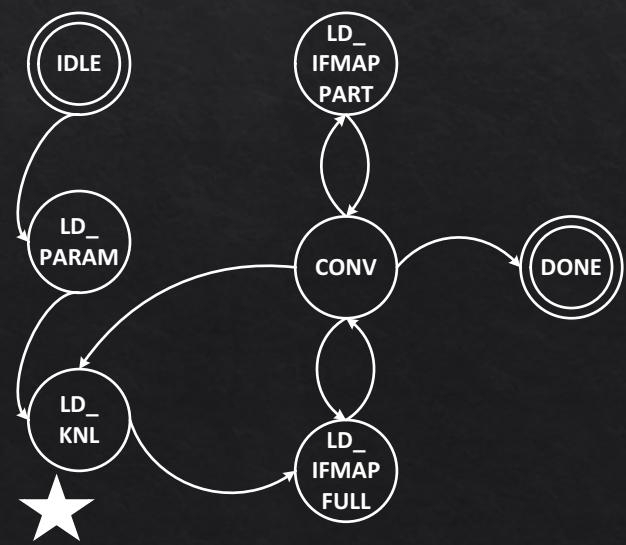


Workflow

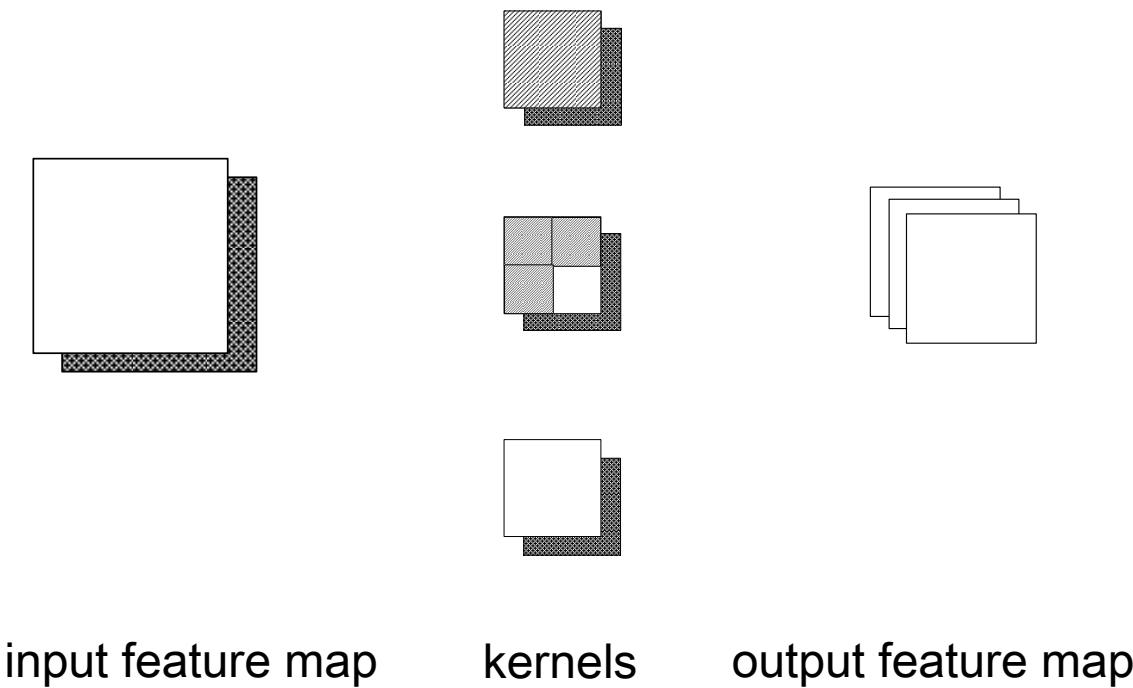


counters

knl_wts	1
knl_id	1
ifmap_chnl	1

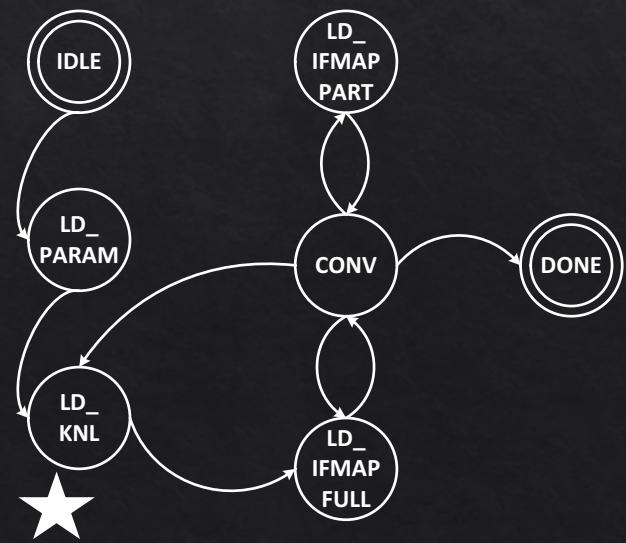


Workflow

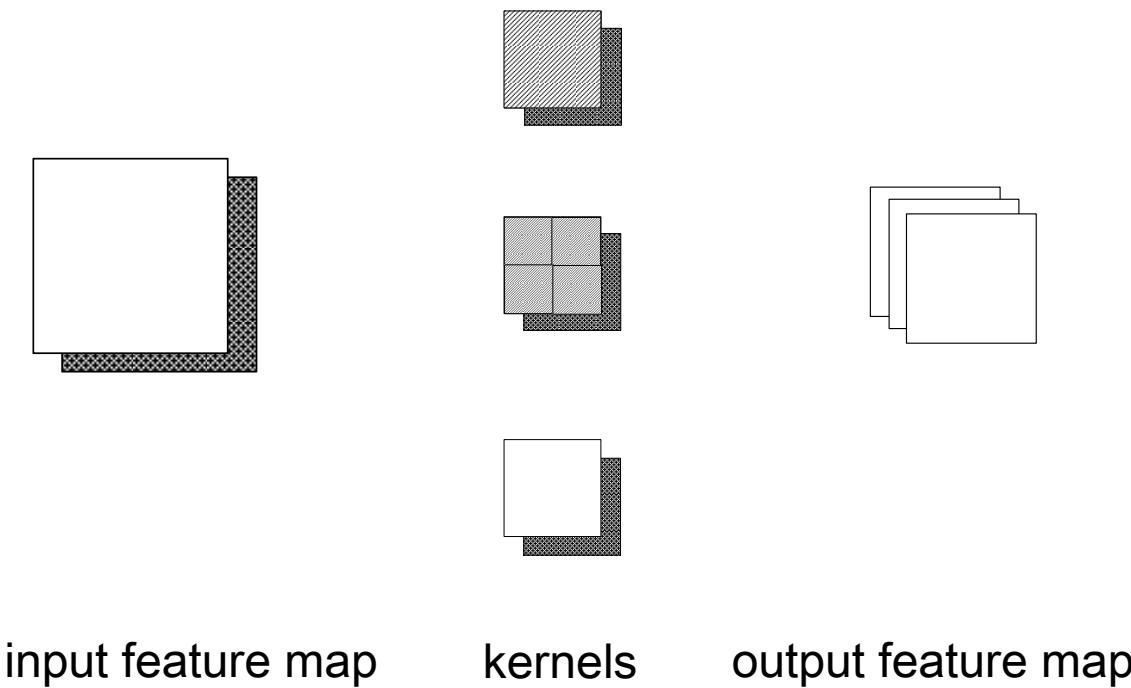


counters

knl_wts	2
knl_id	1
ifmap_chnl	1



Workflow



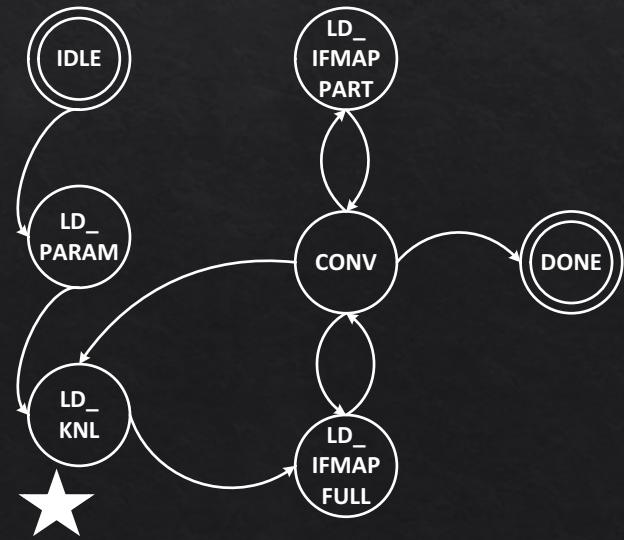
input feature map

kernels

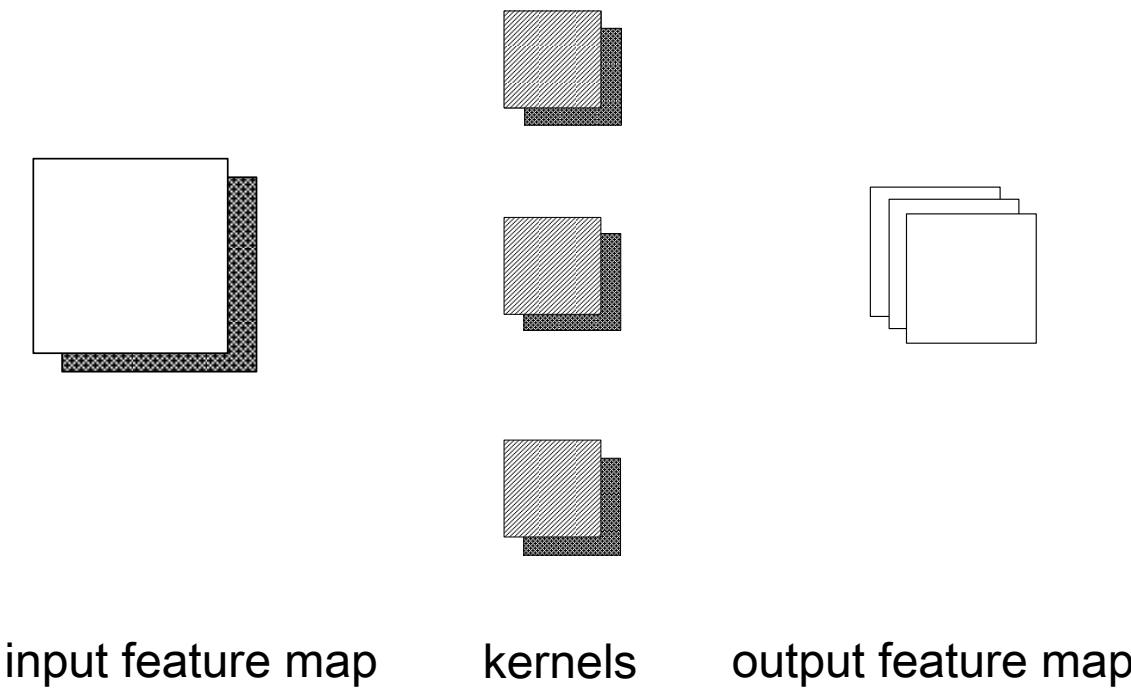
output feature map

counters

knl_wts	3
knl_id	1
ifmap_chnl	1

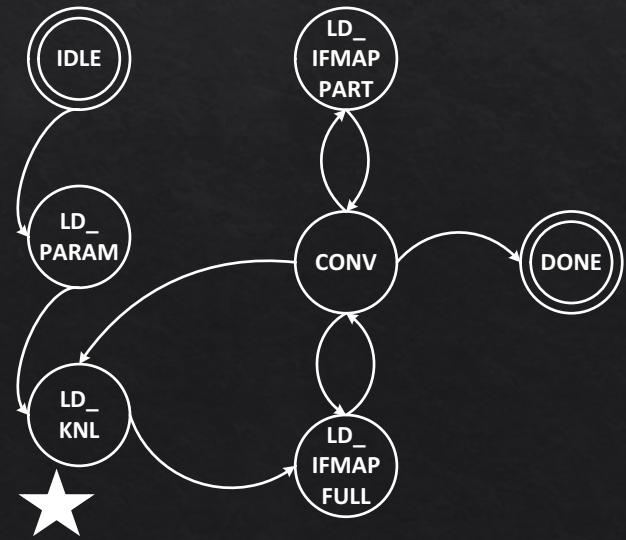


Workflow

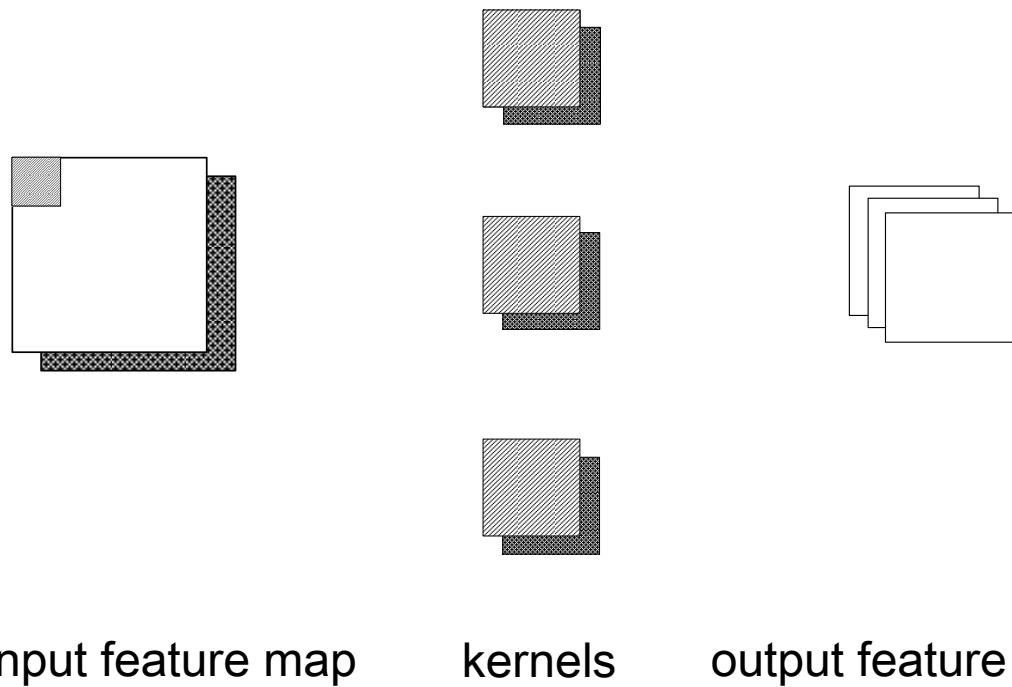


counters

knl_wts	3
knl_id	2
ifmap_chnl	1



Workflow



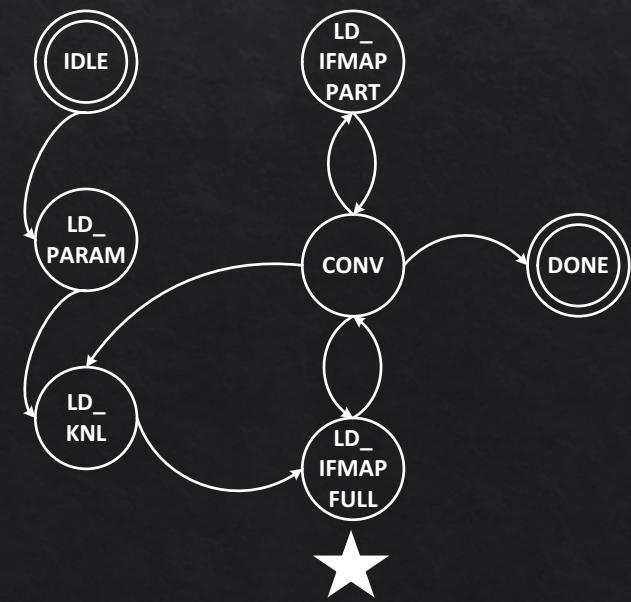
input feature map

kernels

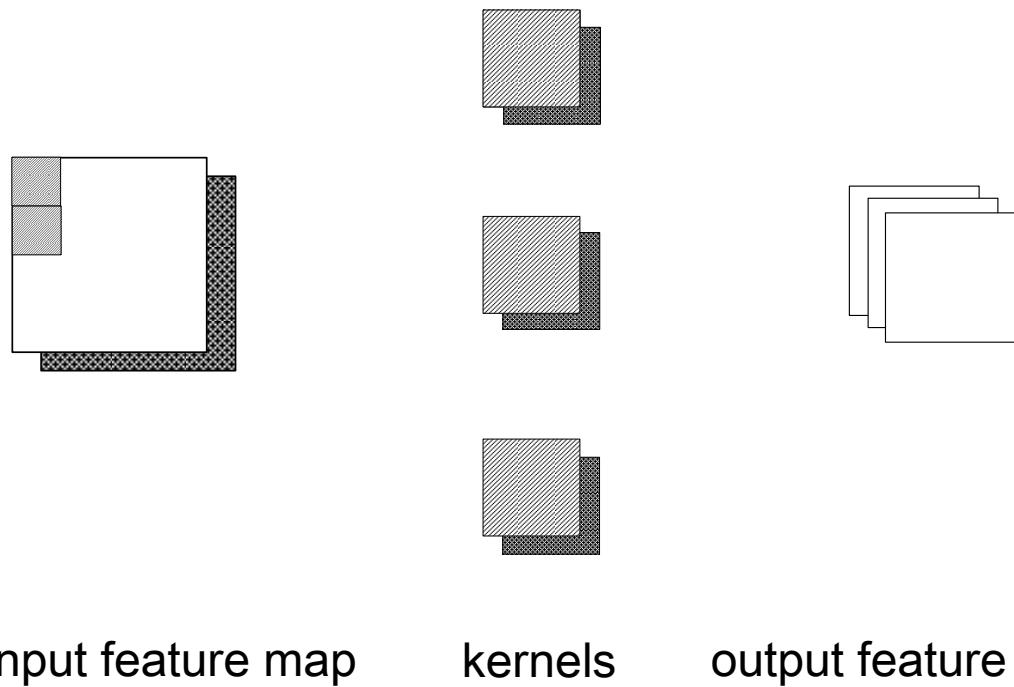
output feature map

counters

base_x	0
base_y	0
delta_x	0
delta_y	0
ifmap_chnl	1



Workflow



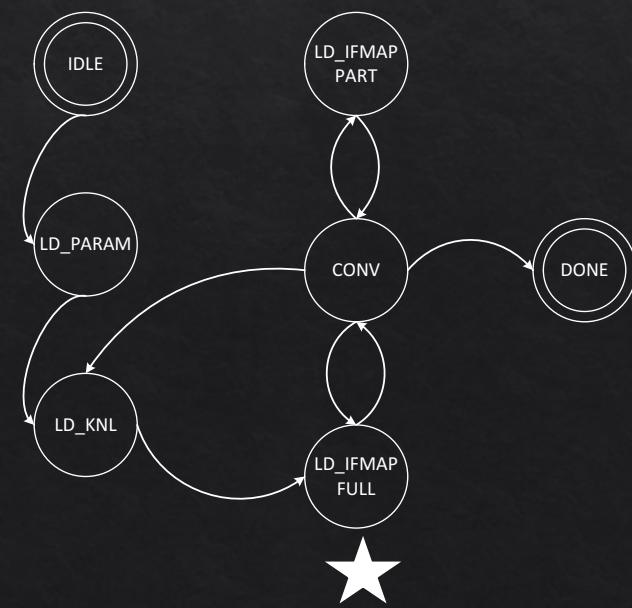
input feature map

kernels

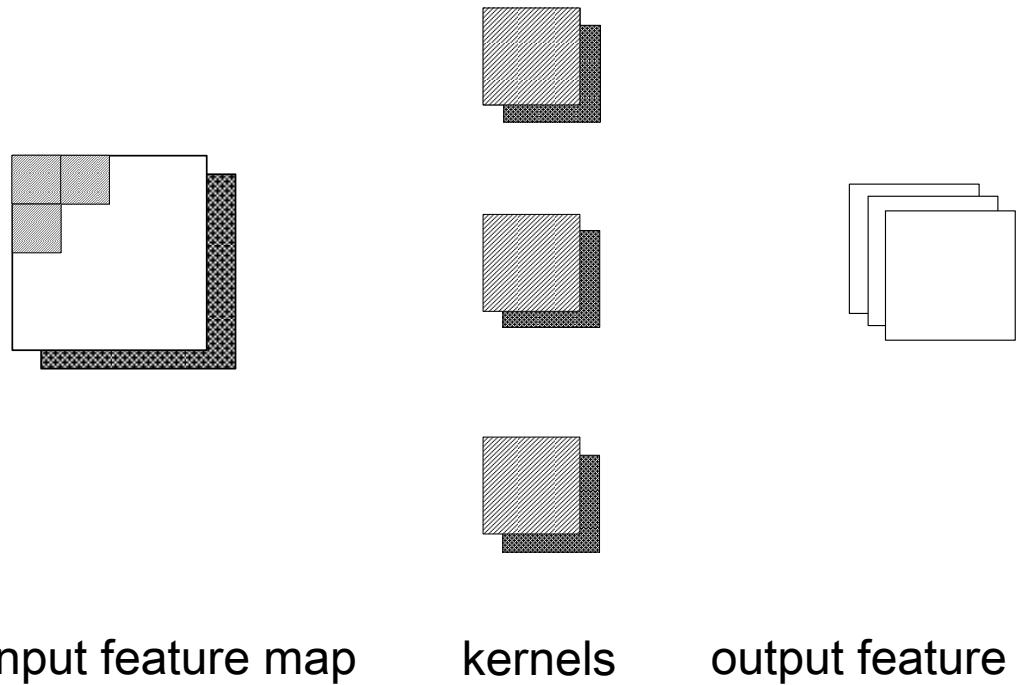
output feature map

counters

base_x	0
base_y	0
delta_x	0
delta_y	1
ifmap_chnl	1



Workflow



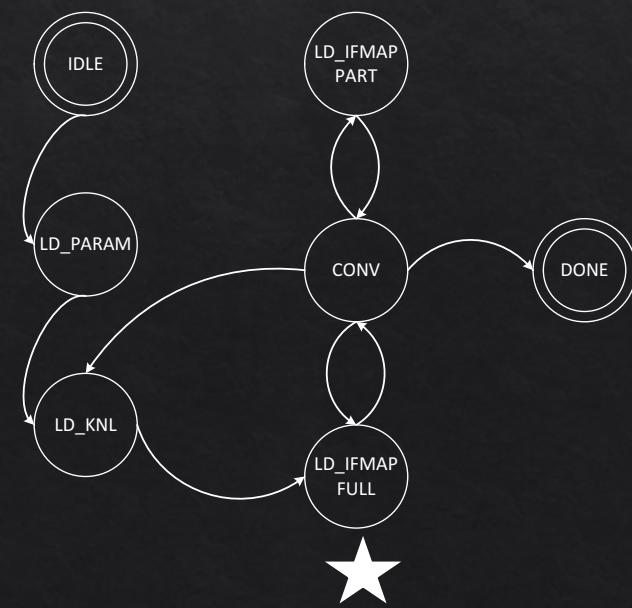
input feature map

kernels

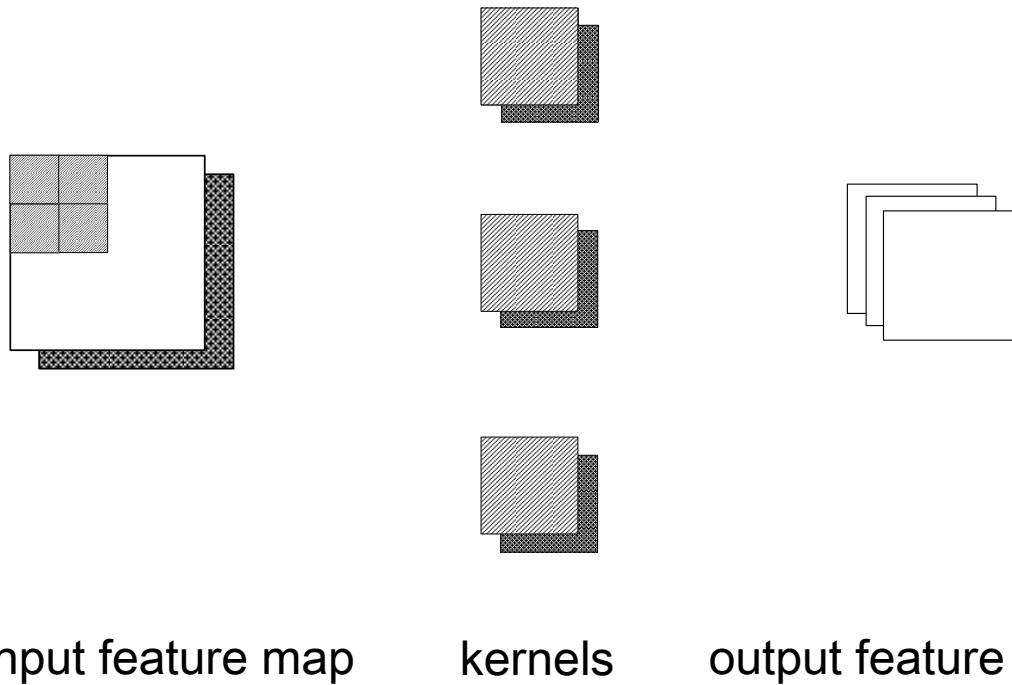
output feature map

counters

base_x	0
base_y	0
delta_x	1
delta_y	0
ifmap_chnl	1



Workflow



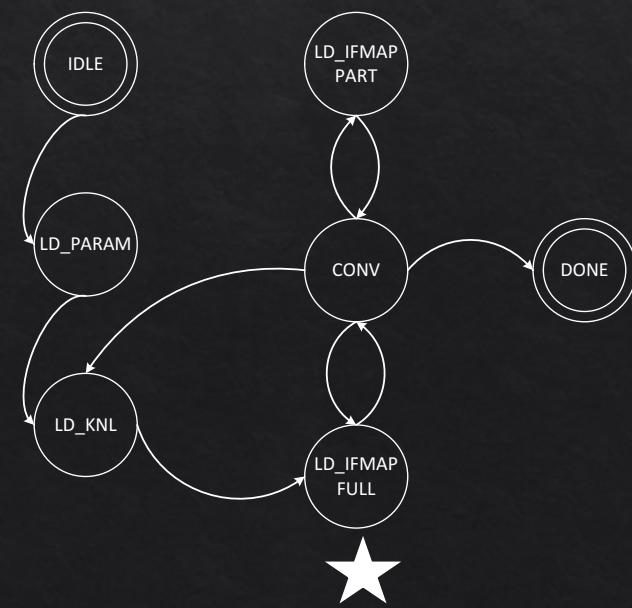
input feature map

kernels

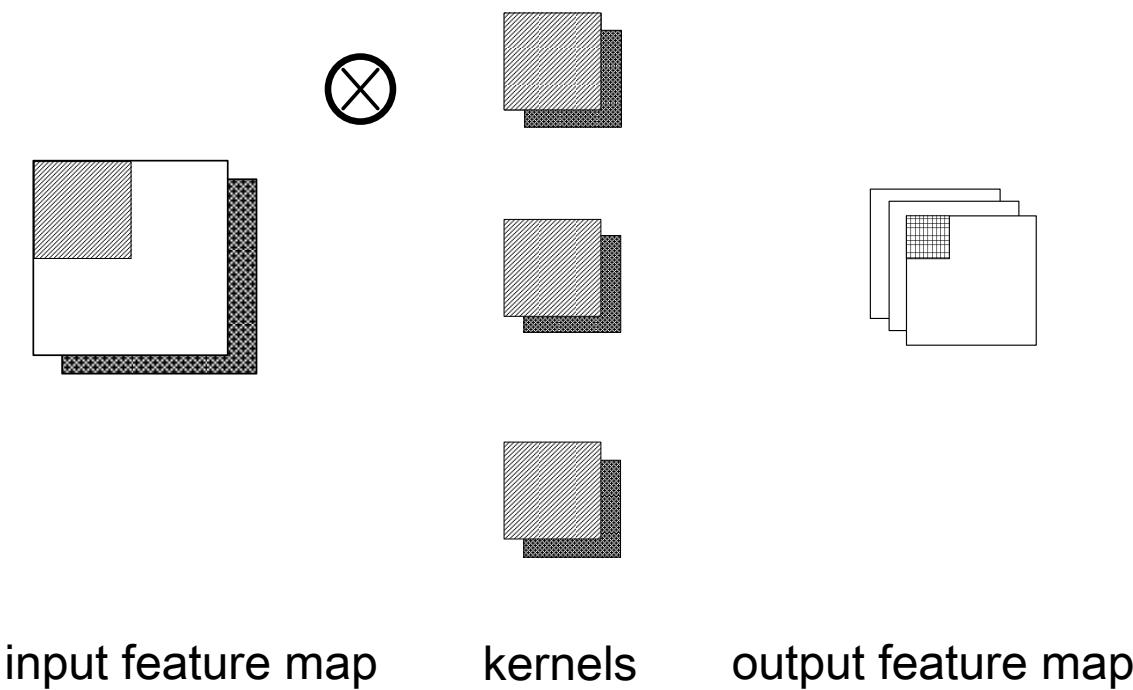
output feature map

counters

base_x	0
base_y	0
delta_x	1
delta_y	1
ifmap_chnl	1

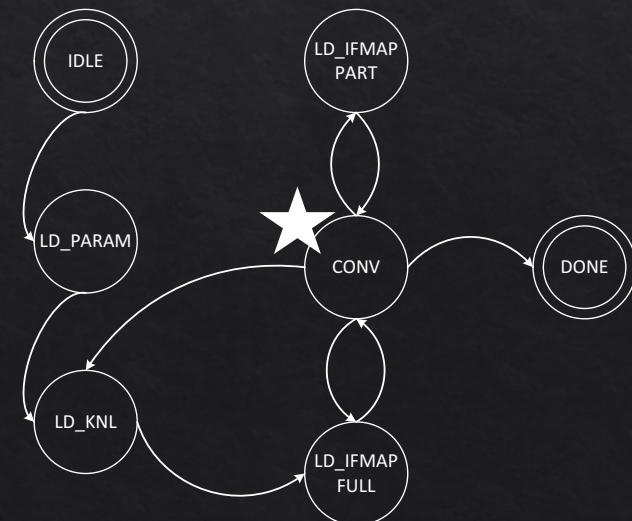


Workflow

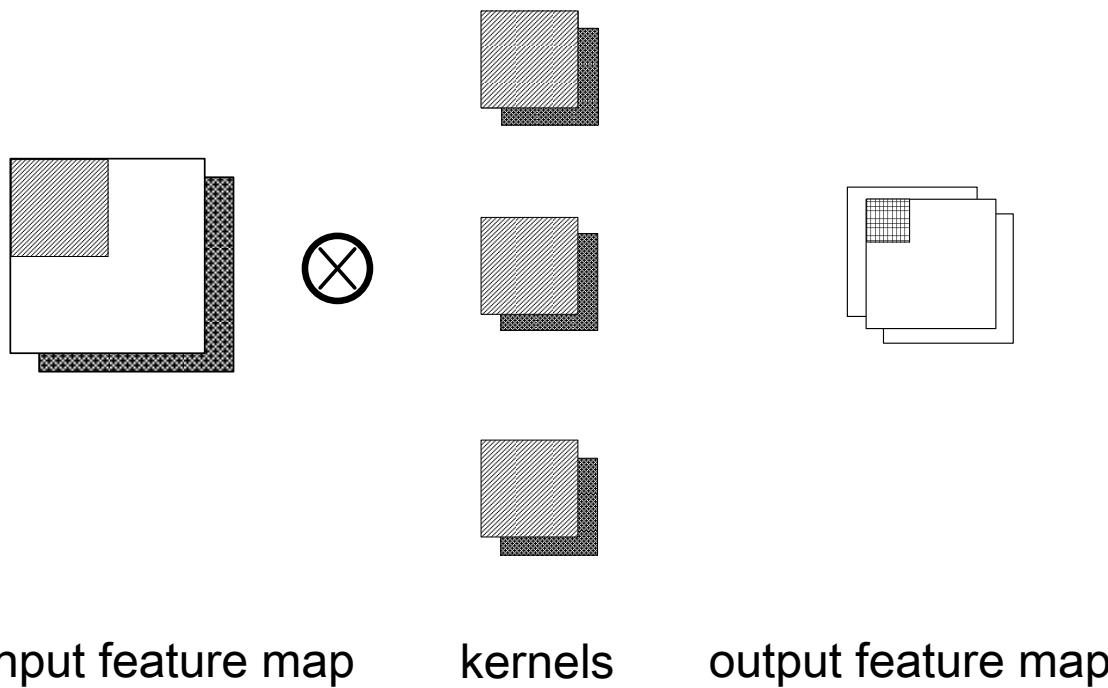


counters

base_x	0
base_y	0
ofmap_chnl	0

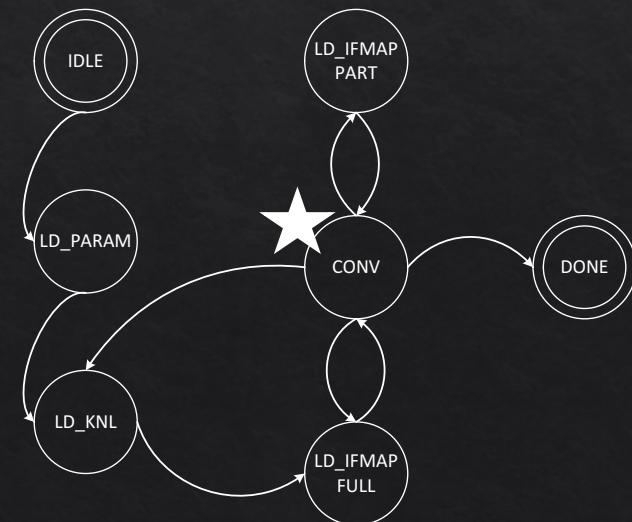


Workflow

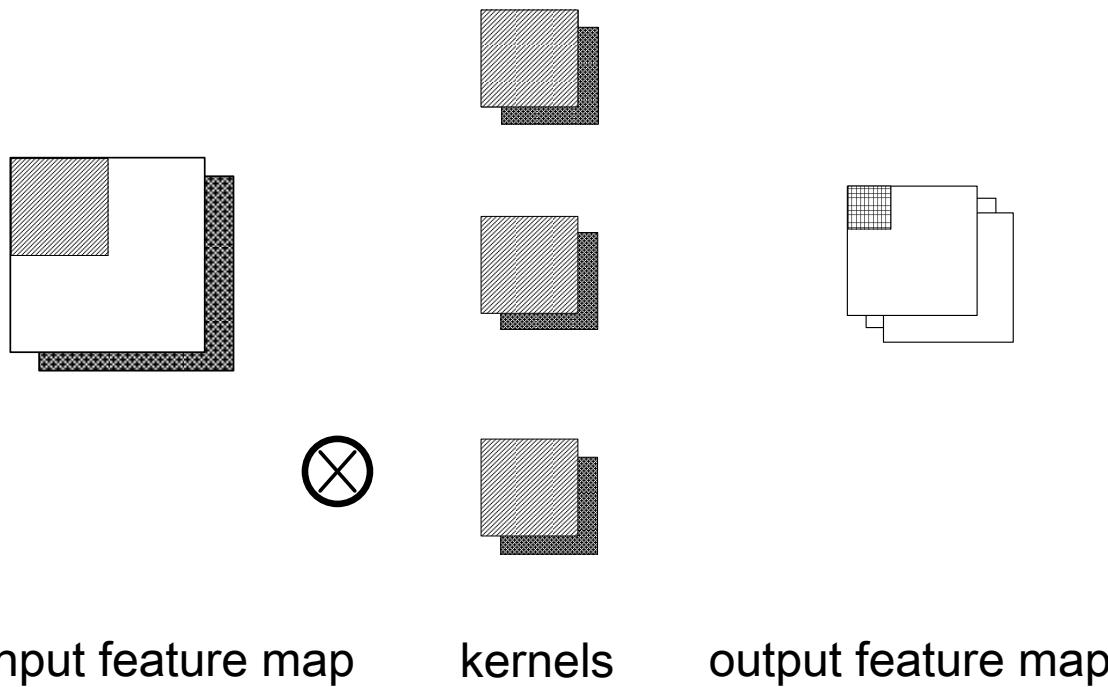


counters

base_x	0
base_y	0
ofmap_chnl	1

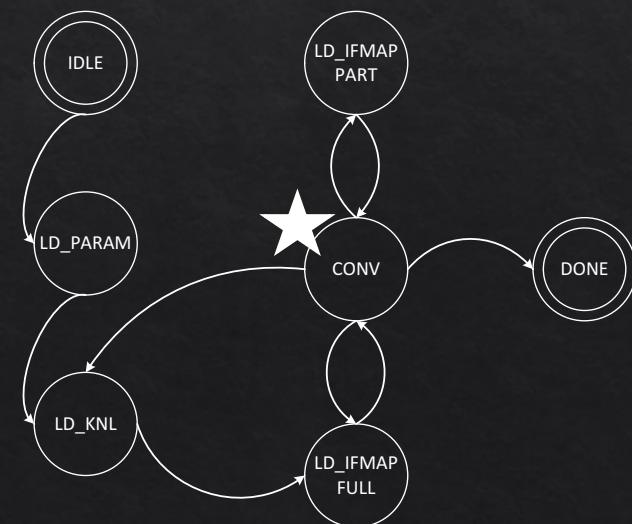


Workflow

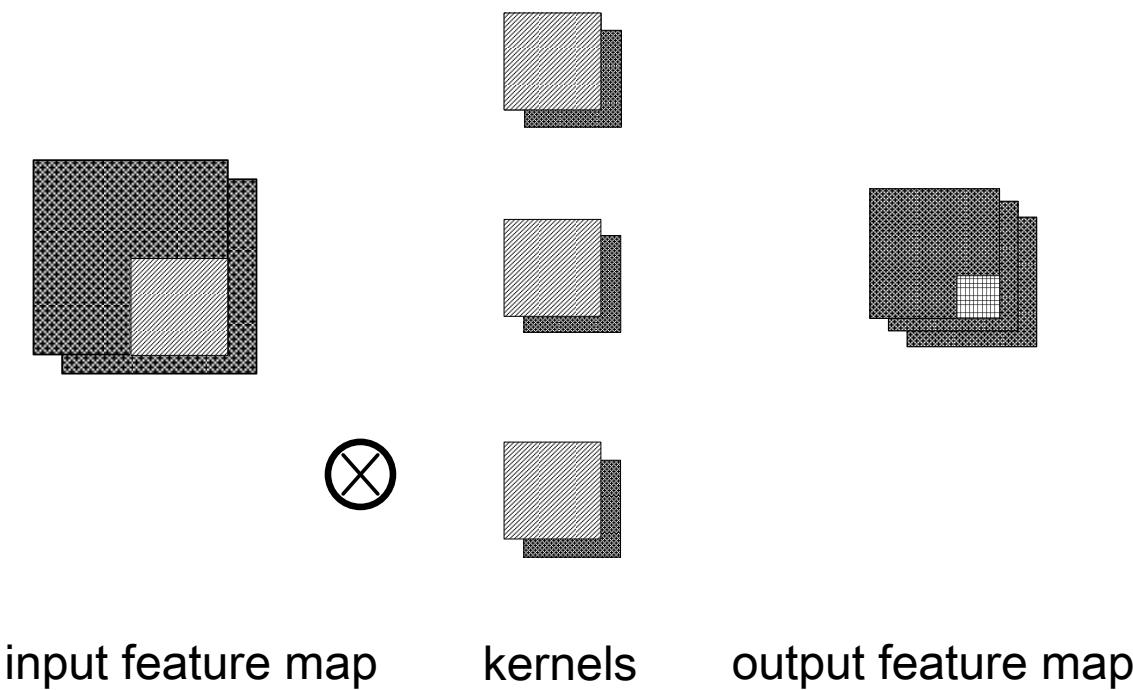


counters

base_x	0
base_y	0
ofmap_chnl	2



Workflow



counters	
base_x	2
base_y	2
ofmap_chnl	2

