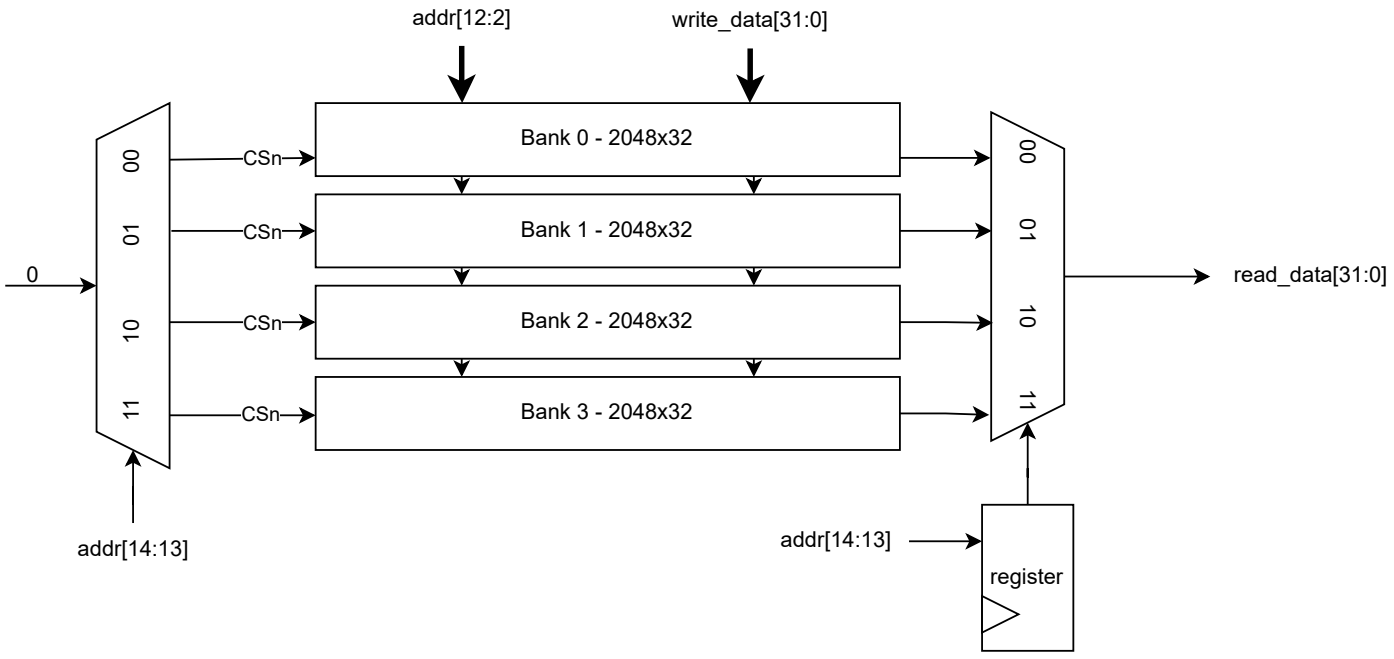
RAM IP Wrapper Architecture

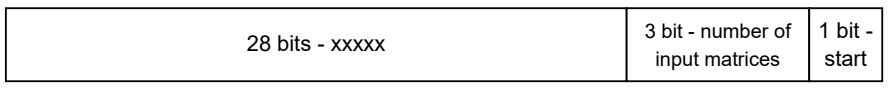
2 bits - bank select		11 bits - address space of each bank (2048 rows)		2 bits - byte select	
14	13	12	2	1	0

write enable is used together with byte enable to allow byte selection of a row during the write cycle



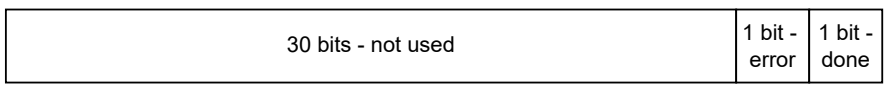
MMU and CONV Register Map

MMU Control register - ctrl_reg - 1x32 bits



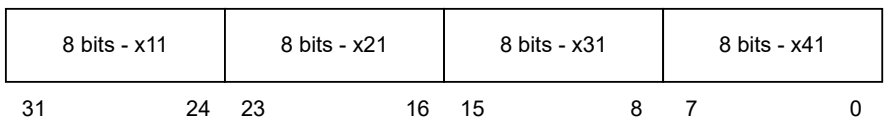
x1A10_3000

MMU Status register - status_reg - 1x32 bits



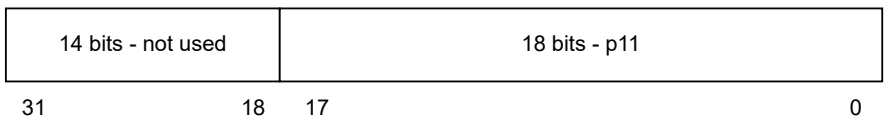
x1A10_3004

MMU Input data registers - indata_reg - 64x32 bits



x1A10_3100
x1A10_31FF

MMU Output data registers - outdata_reg - 160x32 bits

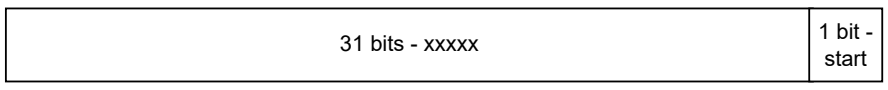


x1A10_3200
x1A10_3480

Software operation:

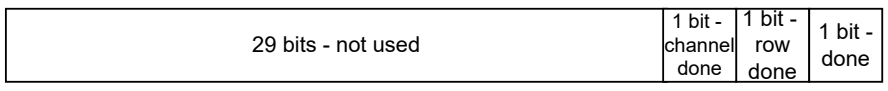
- set the input data registers by sending input matrix elements. Each element is assumed to be 8 bits, unsigned. 4 elements are stored per register. The sequence of the elements should be column-wise - x11, x21, x31... At least 8 registers should be loaded - corresponding to 32 elements, one 4x8 matrix size
- set the number of input matrices in ctrl_reg[3:1]. The max number of matrices that can be loaded and calculated at one time is 7.
- set the start flag in ctrl_reg[0] to initiate MMU
- wait until done bit in status_reg[0] or error bit in status_reg[1] are set

Conv Control register - ctrl_reg - 1x32 bits



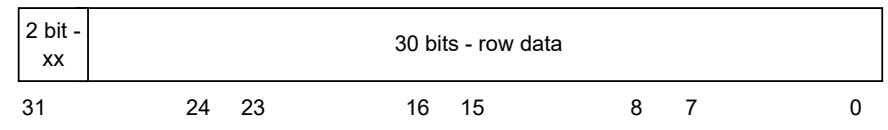
x1A10_3500

Conv Status register - status_reg - 1x32 bits



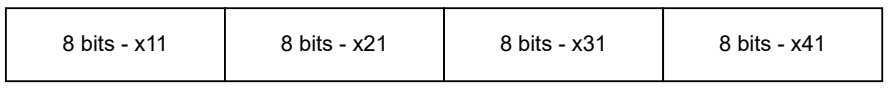
x1A10_3504

Conv Input data register - indata_reg - 1x32 bits



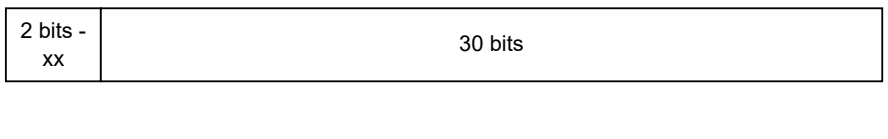
x1A10_3508

Conv Output data registers - outdata_reg - 196x32 bits



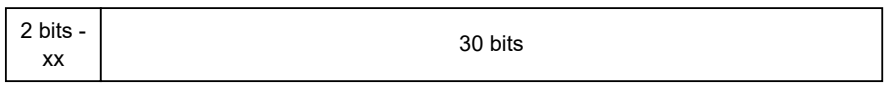
x1A10_350C
x1A10_3818

Conv Filter data registers 1 - filt_reg1 - 1x32 bits



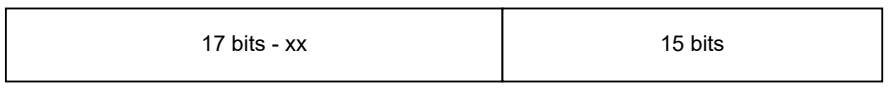
x1A10_3900

Conv Filter data registers 2 - filt_reg2 - 1x32 bits



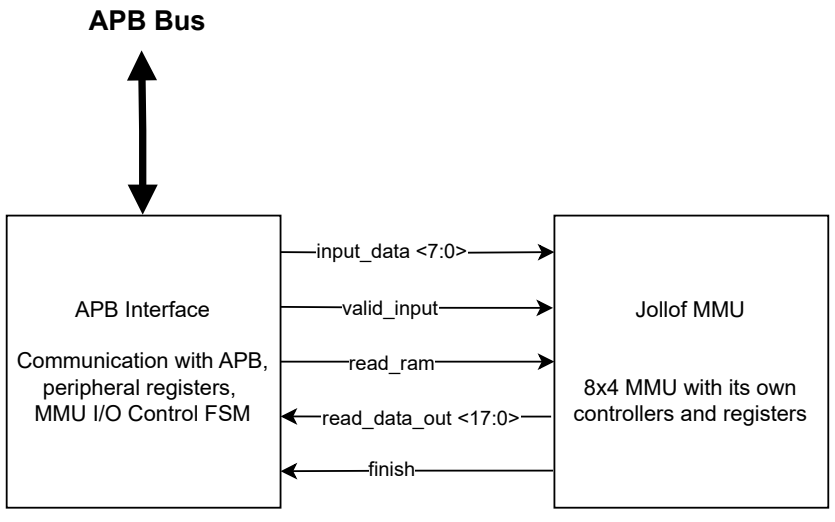
x1A10_3904

Conv Filter data registers 3 - filt_reg3 - 1x32 bits

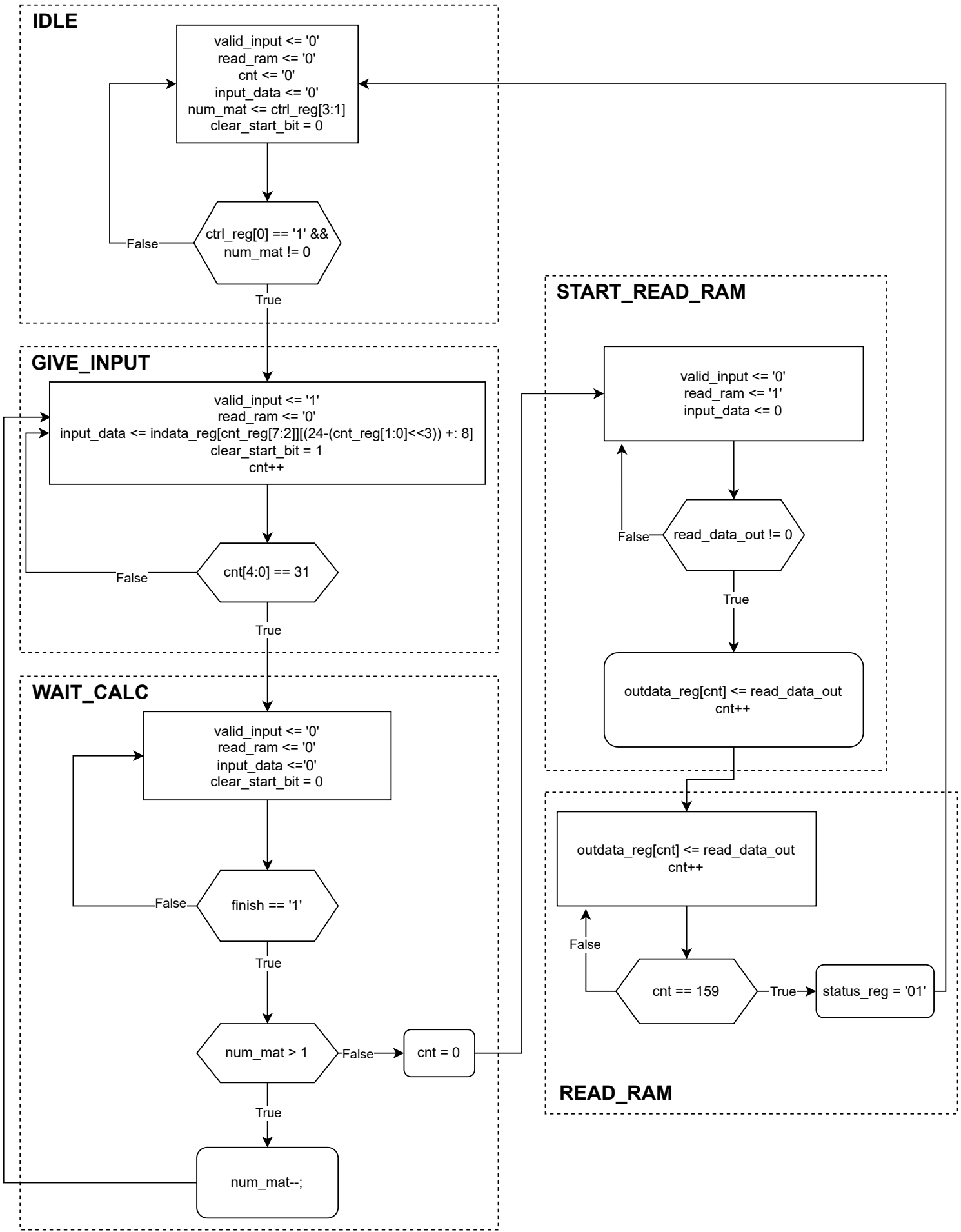


x1A10_3908

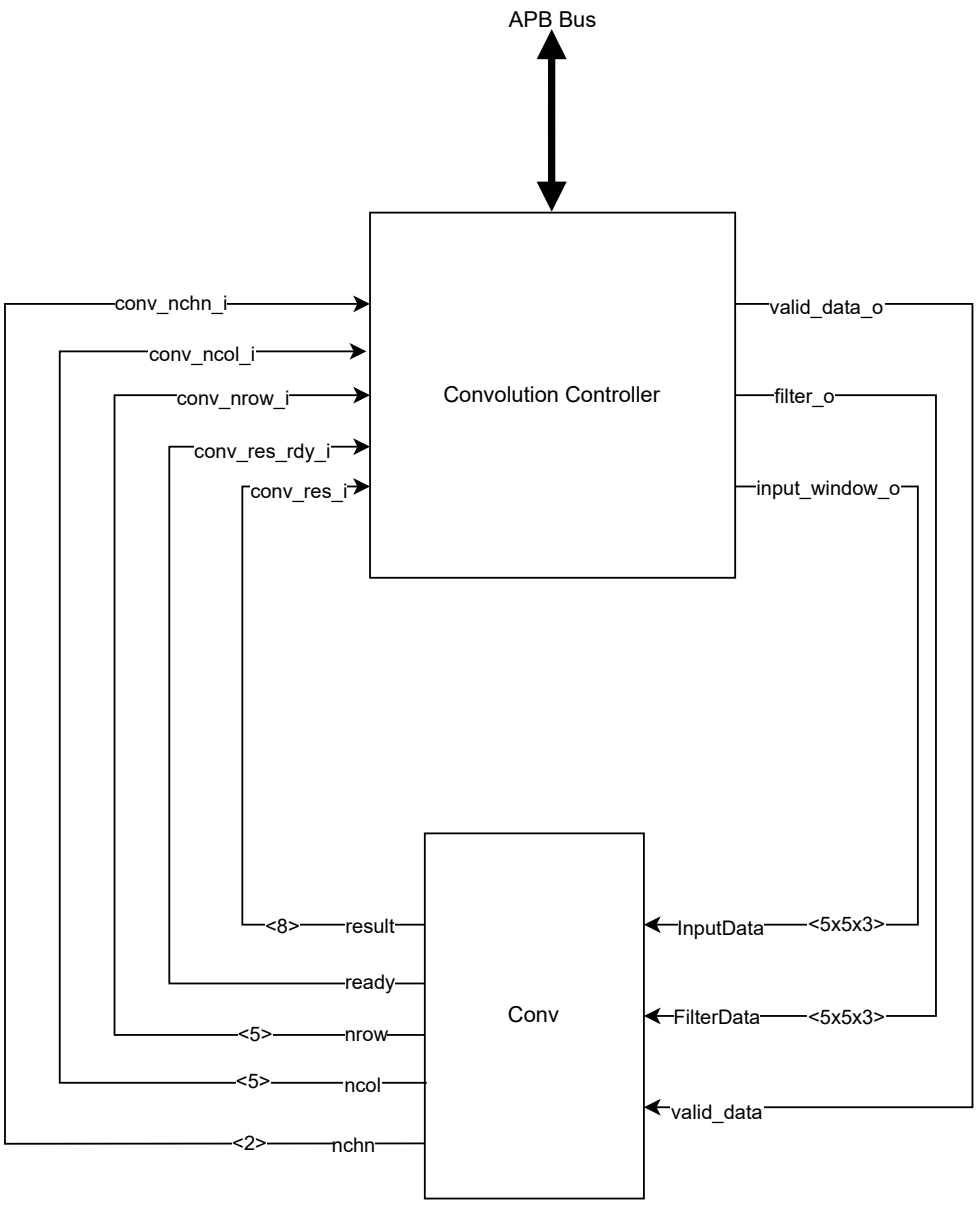
APB MMU Wrapper Architecture



APB MMU Wrapper ASMD



APB CONV Architecture



Convolver sub-module ASMD

Convolution Controller sub-module ASMD