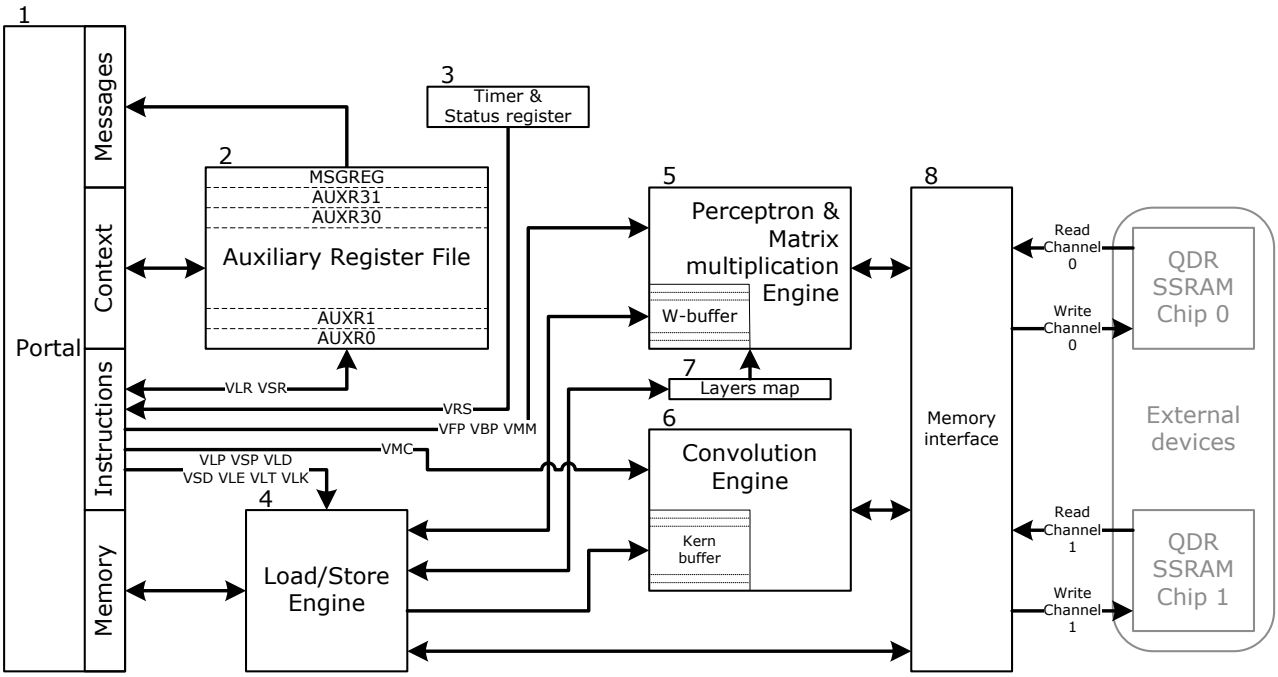
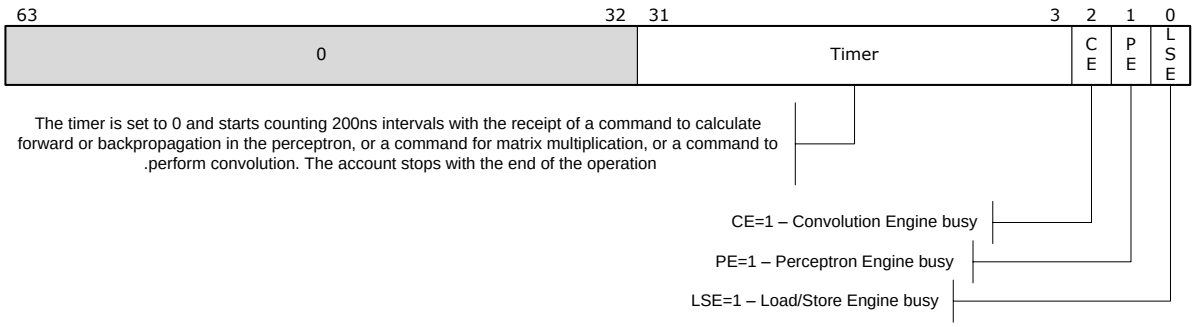
The architecture of the neuroblock.

Table of contents

Neuroblock architecture.



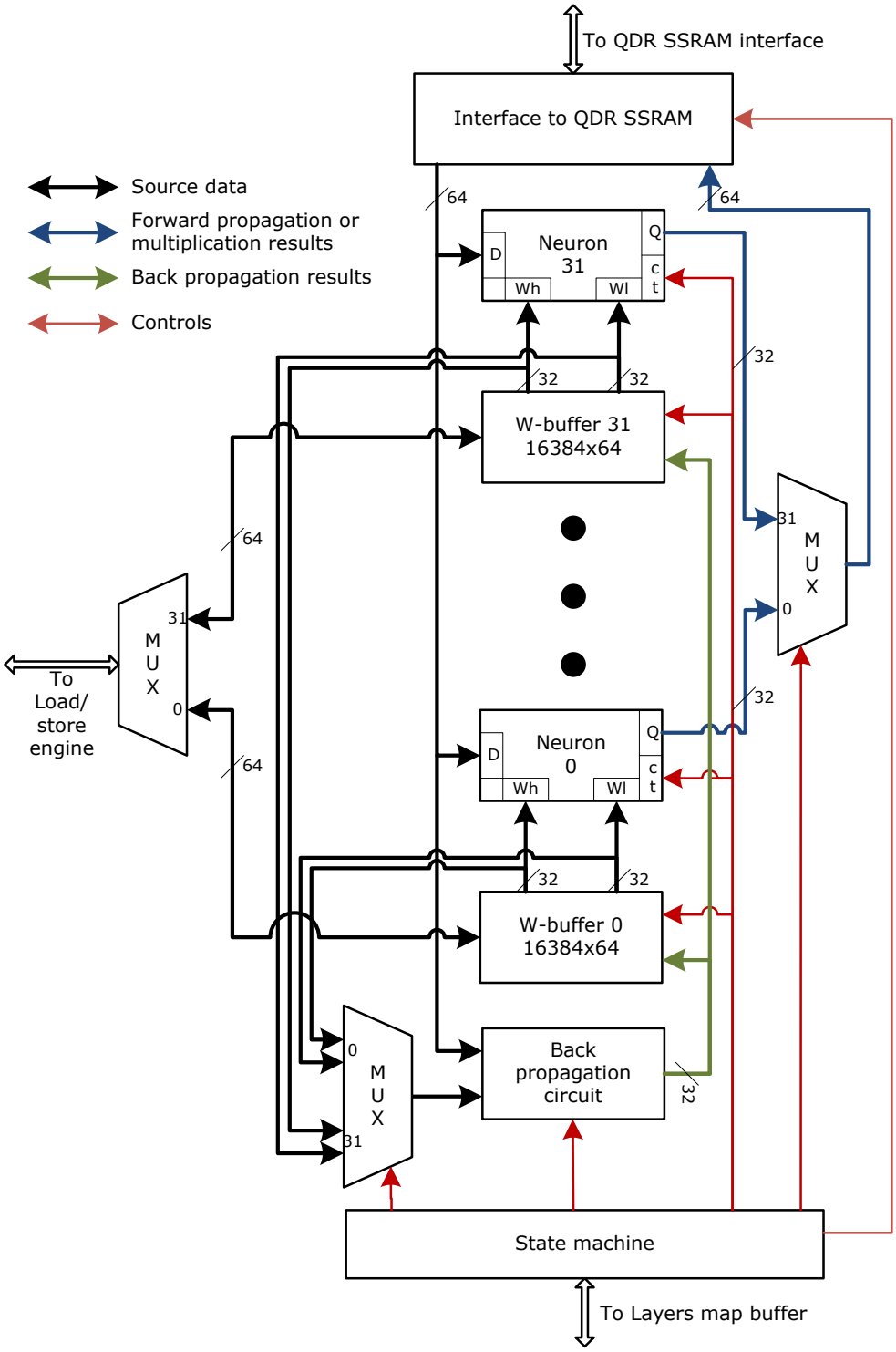
1. Interface to the portal of the base core of the X32Carrier processor. Four channels are used:
    - message channel. It can be used to send messages to the application process about the completion of perceptron forward or backpropagation, matrix multiplication, or the completion of matrix convolution.
    - Context upload/download channel.
    - instruction channel.
    - Access channel to the main memory subsystem.
2. Register file containing 32 64-bit registers. These registers are not used inside the neuroblock. They can be used by application software as additional quick access registers to store any data to free up general purpose registers. The exception is the AUXR16 register. This register has a copy of MSGREG and contains the message index and the message parameter.
3. Timer and status register. The register is used to count the time of execution of forward and back propagation operations in the perceptron, as well as the time of execution of convolution and matrix multiplication operations. The register contains the flags of the readiness of the machines of the neuroblock to perform new operations. The register format is shown in the figure below:



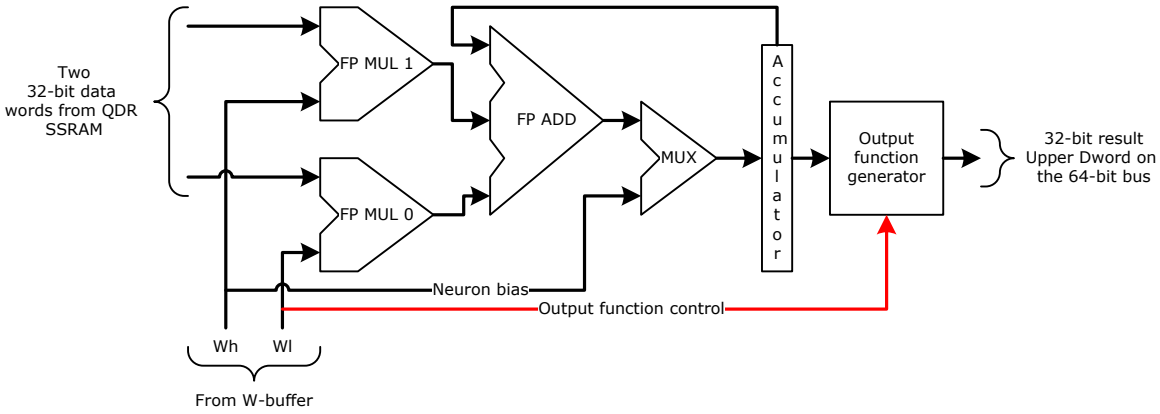
4. Load/Store Engine. The engine performs loading/unloading of the perceptron control object, loading/unloading data from the QDR SSRAM buffers, loading errors into the perceptron, loading transposed matrices before matrix multiplication, and loading kernel for matrix convolution. This engine can transfer data between the main memory and QDR SSRAM buffers in parallel with the operation of the perceptron and convolution engines. Can load the kernel if the convolution engine does not perform operations, while perceptron engine can work, can load or unload the perceptron if the perceptron engine does not perform operations, but convolution may be in progress.
5. Perceptron & Matrix multiplication Engine. The operations of perceptron forward propagation and matrix multiplication are quite similar in their algorithms and therefore are performed using the same engine. When calculating the perceptron, the data of the neuron layers are in the external SSRAM, and the weights and control of the neurons are placed in the W-buffer inside the FPGA. In multiplication operations, one matrix is located in the external SSRAM, and the second in transposed form is written to the W-buffer. Perceptron Engine also performs back propagation.
6. Convolution Engine. It includes a buffer for storing the convolution kernel. The kernel size can be 3,5,7,9,11,13,15.
7. Layers Map. This is a separate buffer into which the perceptron layer configuration is loaded. It contains 32-bit numbers indicating the number of neurons in each layer of the perceptron. The first number indicates the number of inputs. The list of layers ends with a zero after the last layer length specifier. For example, the sequence of numbers 784, 1024, 10, 0 indicates that the perceptron has 784 inputs, 1024 neurons of the first layer, 10 neurons of the second layer, and it is also the last one.
8. Memory Interface. The block multiplexes transaction requests from three sources and distributes them over four channels of two interfaces to QDR SSRAM memory.

Perceptron & Multiplication engine.

Below is a detailed diagram of the Perceptron & Multiplication Engine.



The basis of the engine is a set of 32 neurons that perform multiplication with accumulation in a pipelined mode. Below is the structure of a neuron.

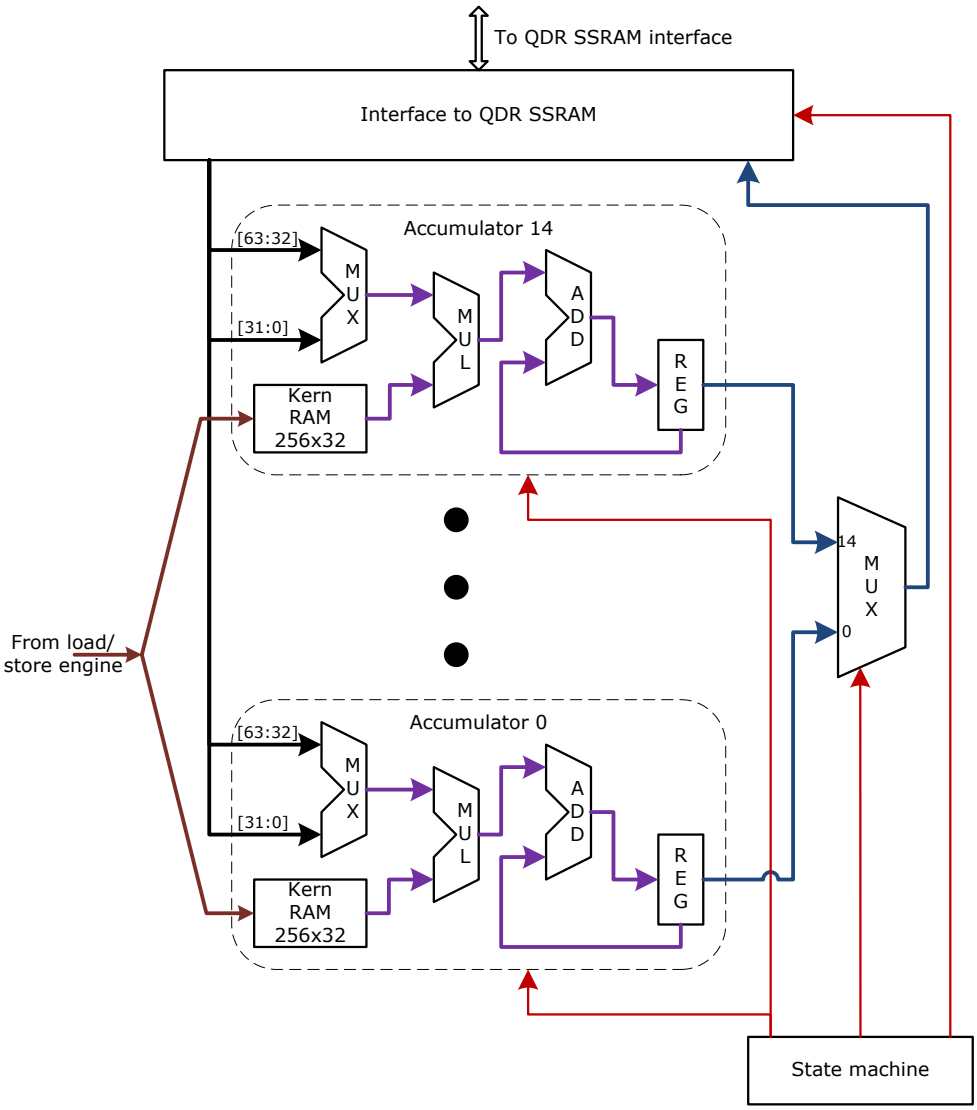


At each clock cycle, each neuron performs 2 multiplication operations and 2 addition operations to accumulate the sum of the products of the input data by their respective weight coefficients. Upon completion of the accumulation, the neuron generates the result in accordance with the algorithm specified in the control word. The result of the operation is written to SSRAM. The recording of 32 results is carried out in parallel with the process of reading new data and forming a new value in the accumulator register. Thus, the peak performance of an assembly of 32 neurons is 128 floating point operations per 1 clock cycle.

The back propagation circuit is a simple accumulator multiplier since all back propagation calculations contain only addition and multiplication operations. All calculations during backpropagation are performed sequentially for each neuron, which causes low performance on backpropagation.

Convolution engine.

The Convolution engine contains 15 identical accumulator multipliers and 15 kernel buffers.



Data is written by the load/store engine simultaneously to all 15 buffers, which generates identical kernel images for all 15 accumulator multipliers. This allows you to calculate in parallel the values of several elements of the resulting matrix. For example, if the kernel size is five, then the machine, extracting one value from memory, calculates immediately for five convolutions at the same time.

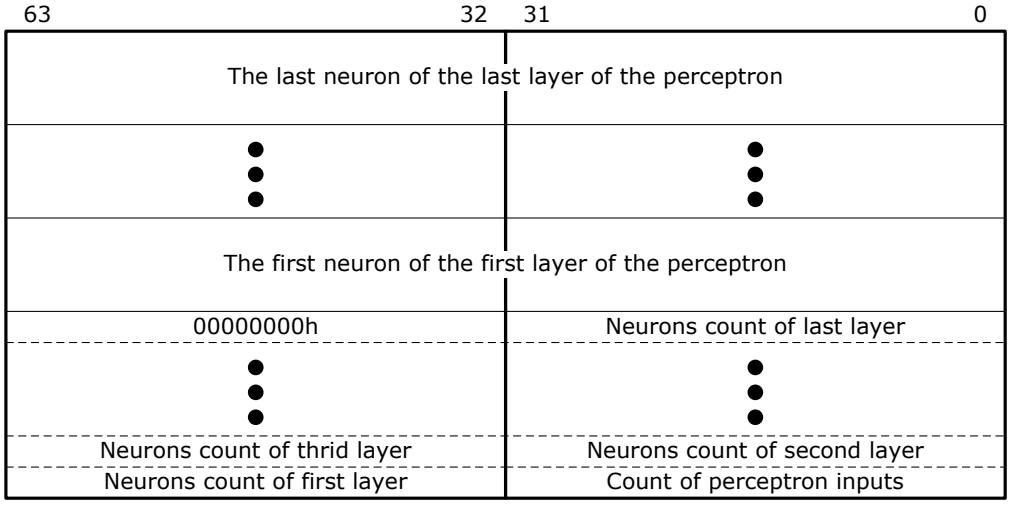
Reading the source matrix data from the QDR SSRAM buffer and writing the resulting matrix data are performed in parallel. To implement this, different buffers are used for read and write operations. If the source matrix is in buffer 0, then the result matrix will be placed in buffer 1, and the offset of the resulting matrix in buffer 1 will be the same as that of the source matrix in buffer 0.

Application-specific instructions of the neuroblock.

VLP. Load perceptron.

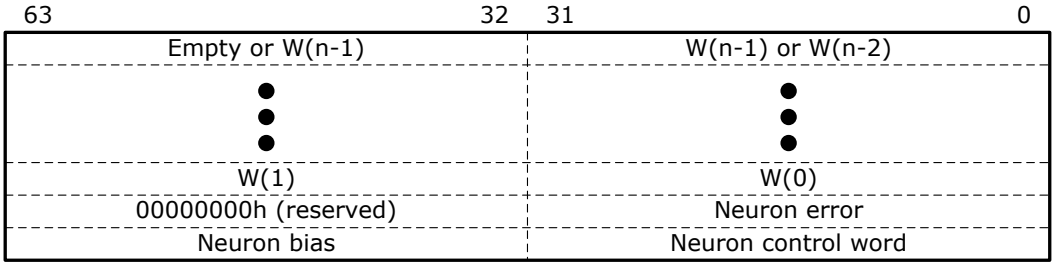
This instruction loads the perceptron parameters into the neuroblock. The parameters include a layer map describing the number of layers and the number of

neurons in each layer. After the layer map, neuron data follows - control and coefficients. The format of the perceptron parameter block is shown in the figure below.

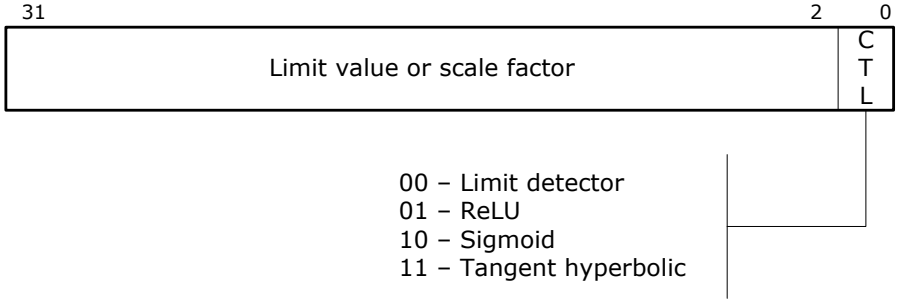


If the neuron counter of the last layer is in bits [63:32], then the perceptron map block and the data of the first neuron are separated by a 64-bit null word. If the last counter is in bits [31:0], then the data of the first neuron is located as shown in the figure above.

Description of the neuron in the block of perceptron parameters.



The first 32-bit word is the neuron control word. It defines 2 parameters:



1. CTL – Neuron output function.
2. A threshold value that is only used if the threshold detector neuron output function is used. If the result in the accumulator of the neuron is less than or equal to the threshold, then the output of the neuron is 0.0, if more, then 1.0. In ReLU detector mode, the value in bits [31:2] is used as a multiplier for the result if the result in the accumulator is negative.

Neuron bias is the bias added to the accumulator when the sum of the weighted input values is accumulated.

Neuron error - used when calculating backpropagation. The neuroblock itself fills in these fields for the neurons of the inner layers in the process of calculating errors for the inner layers. Only the output layer of neurons receives error values from the outside.

Executed by:

Load/store engine

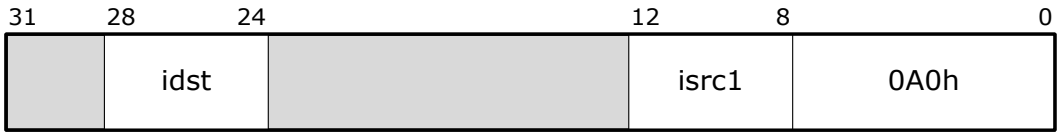
Mnemonics:

VLP idst,isrc1

Example:

vlp r2,r21

Format:



Instruction parameters:

Isrc1 - indicates the register containing the object selector in which the perceptron control block is located from the zero offset.

Idst - is a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VSP. Store perceptron.

The instruction unloads the block of perceptron parameters into an object located in RAM.

Executed by:

Load/store engine

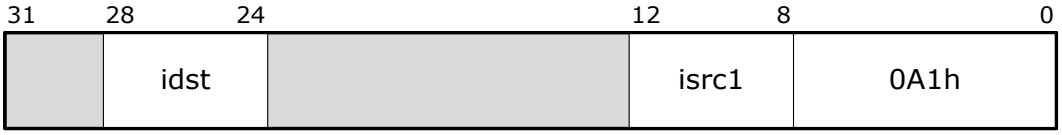
Mnemonics:

VSP idst,isrc1

Example:

vsp r2,r21

Format:



Instruction parameters:

Isrc1 – the register in which the object selector is located. A block of perceptron parameters will be placed in this object starting from the 0th offset.

Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VLD. Load data to the data buffer.

The instruction allows loading a data block into one of two QDR SSRAM buffers. Used to load matrices before matrix multiplication, load input to perceptron.

Executed by:

Load/store engine

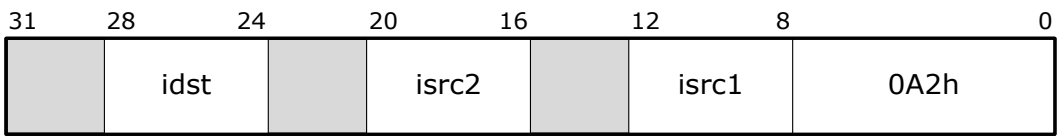
Mnemonics:

VLD idst,isrc2,isrc1

Example:

vld r13,r12,r1

Format:



Instruction parameters:

Isrc1 – the register contains in bits [31:0] the selector of the object from which data will be retrieved, and in bits [63:32] contains a counter of 64-bit words to be loaded into the QDR SSRAM buffer.

Isrc2 – load address of the data block in the SSRAM buffer. Bit 31 determines which of the two buffers the data will be loaded into.

Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VSD. Store data from data buffer.

The instruction transfers a block of data from the QDR SSRAM buffer to an object located in main memory.

Executed by:

Load/store engine

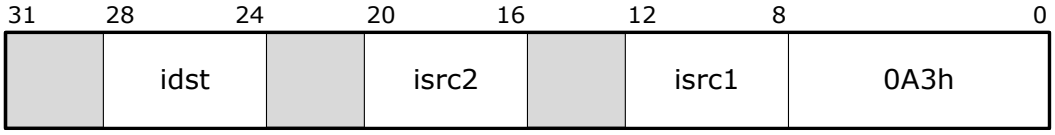
Mnemonics:

VSD idst,isrc1,isrc2

Example:

vsd r13,r2,r3

Format:



Instruction parameters:

Isrc1 – the lower 32 bits of the register contain the object selector, into which data from the buffer is placed from offset zero. The upper 32 bits are used to indicate the number of 64-bit words to be transferred.

Isrc2 – address of the data block in the QDR SSRAM buffer, bit 31 is used to select the buffer.

Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VLE. Load perceptron output error.

The instruction loads the error block for the output layer of neurons before running the backpropagation calculation. The perceptron output errors are calculated

externally and loaded by the VLR instruction, the errors of the intermediate layers and the first layer are calculated during backpropagation.

Executed by:

Load/store engine

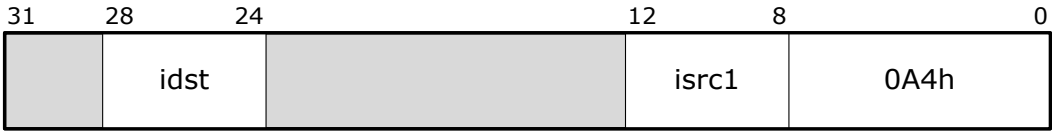
Mnemonics:

VLE idst,isrc1

Example:

vle r4,r2

Format:



Instruction parameters:

Isrc1 – object selector in which the error block is located starting from the zero offset.

Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VLT. Load transposed matrix.

The instruction is used to load one of the matrices into the neuroblock buffer before multiplying the matrices. The transposed matrix is loaded into the FPGA's internal buffers, as are the perceptron coefficients. This solution allows parallel calculation of several (up to 32) cells of the resulting matrix by multiplying the 1st row of the direct matrix by several columns of the transposed one.

Executed by:

Load/store engine

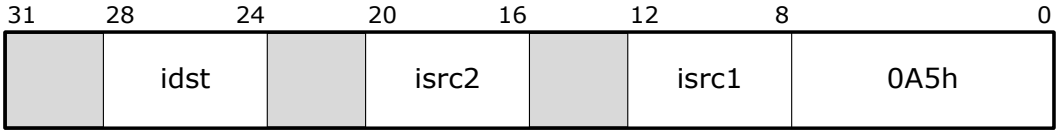
Mnemonics:

VLT idst,isrc2,isrc1

Example:

vlt r5,r6,r1

Format:



Instruction parameters:

Isrc1 – object selector in which the matrix to be loaded into the buffer is located starting from zero offset.

Isrc2 – bits [14:0] contain the number of matrix columns, bits [29:16] the number of rows.

Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VLK. Load kernel.

With this instruction, you can load the kernel matrix before the matrix convolution operation.

Executed by:

Load/store engine

Mnemonics:

VLK idst,isrc1

Example:

vlk r9,r11

Format:

Instruction parameters:

Isrc1 – bits [31:0] contain the object selector in which the kernel data is located starting from the zero offset. Bits [23:16] contain a 64-bit word count of the length of the kernel data. This counter is calculated using the formula:

$N = \frac{1+S^2}{2}$, где: S – kernel size (3x3, 5x5, 7x7 …. 15x15)

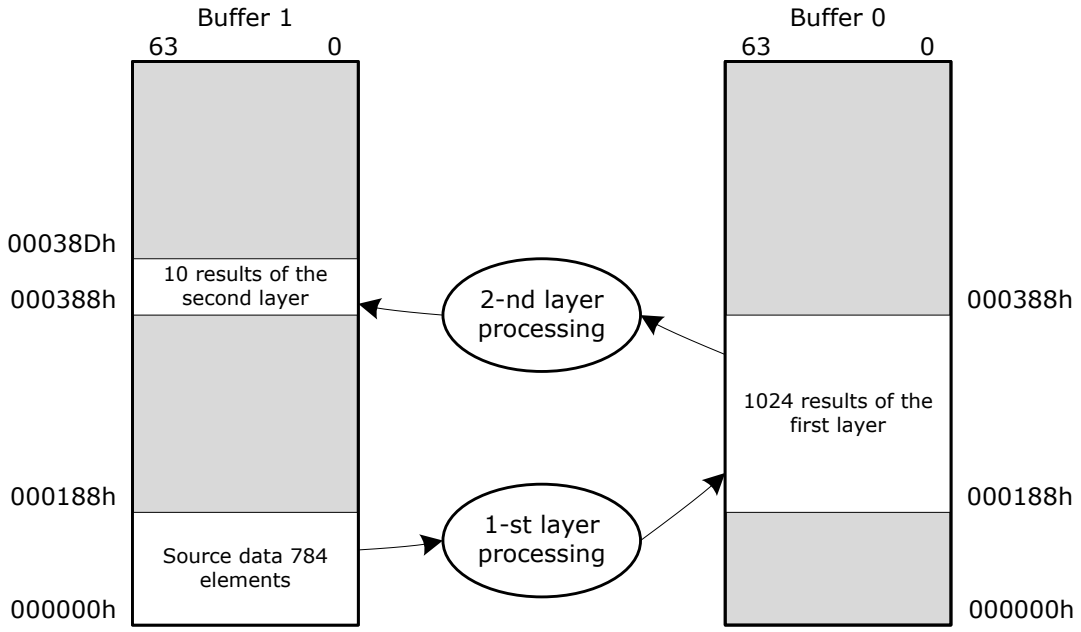
Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VFP. Forward propagation.

The instruction performs a perceptron forward propagation calculation. Before submitting it, you must at least load the perceptron map using the VLP instruction. The data is loaded into one of the buffers by the VLD instruction.

The input data of the perceptron can be located in any of the SSRAM blocks, both in the 0th and in the 1st. The input data must always be located in the selected buffer from address zero.

In the process of calculating forward propagation, the neuroblock alternately changes the result receiver buffer from layer to layer, and then, when calculating the next layer, the data source buffer. Below is an example of data layout for a perceptron of 2 layers of neurons.



The initial data is located in buffer 1. When processing the first layer of neurons, the result is placed in buffer 0 from an offset equal to the length of the initial data.

Executed by:

Perceptron & matrix multiplication engine

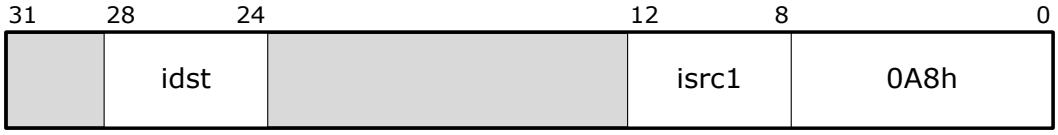
Mnemonics:

VFP idst,isrc1

Example:

vfp r14,r12

Format:



Instruction parameters:

Isrc1 – bit 31 indicates the index of the buffer where the original data is located. Data is always located from address zero.

Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VBP. Backward propagation.

The instruction starts the perceptron backpropagation process. Backpropagation is performed in 2 stages: calculation of errors from the output of the perceptron to its input, and the second stage is the recalculation of weight coefficients starting from the neurons of the first layer and ending with the neurons of the last one.

Executed by:

Perceptron & matrix multiplication engine

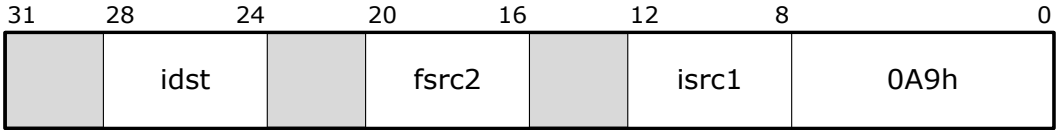
Mnemonics:

VBP idst,fsrc2,isrc1

Example:

vbp r9,rfs10,r12

Format:



Instruction parameters:

Isrc1 – bit 31 indicates the number of the buffer in which the original data is located.

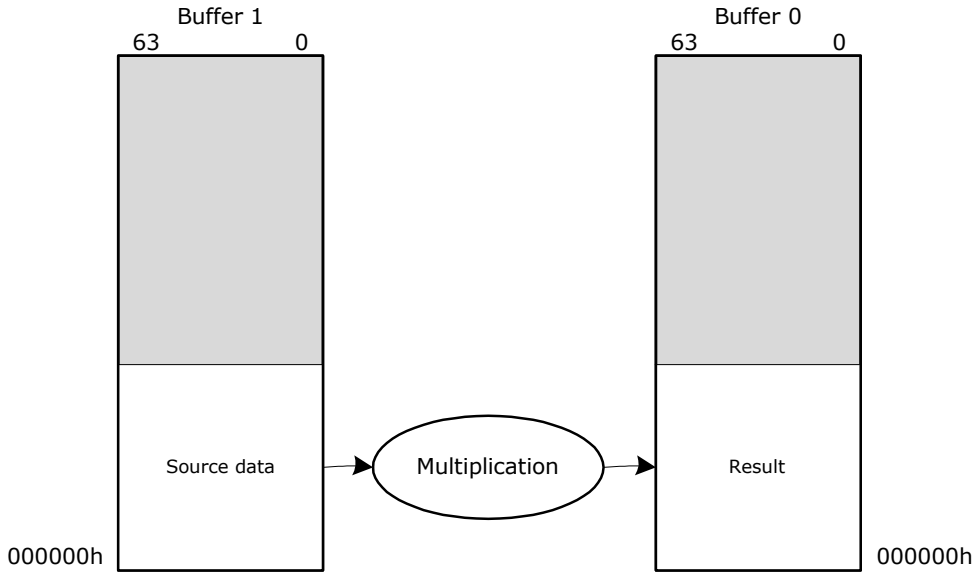
Fsrc2 – the register must contain a single-precision floating-point number that specifies the perceptron's learning rate factor.

Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VMM. Matrix multiplication.

Matrix multiplication. One matrix resides in one of the external QDR SSRAM buffers. The second matrix /in transposed form/ is located in the buffer coefficients of

the perceptron. The result of the operation is placed in the adjacent buffer of the external QDR SSRAM, as shown in the figure below. The initial and resulting matrices are always located in the buffers from zero offset.



Executed by:

Perceptron & matrix multiplication engine

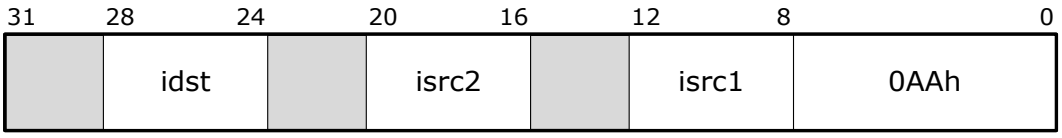
Mnemonics:

VMM	idst,isrc2,isrc1

Example:

vmm	r5,r6,r4

Format:



Instruction parameters:

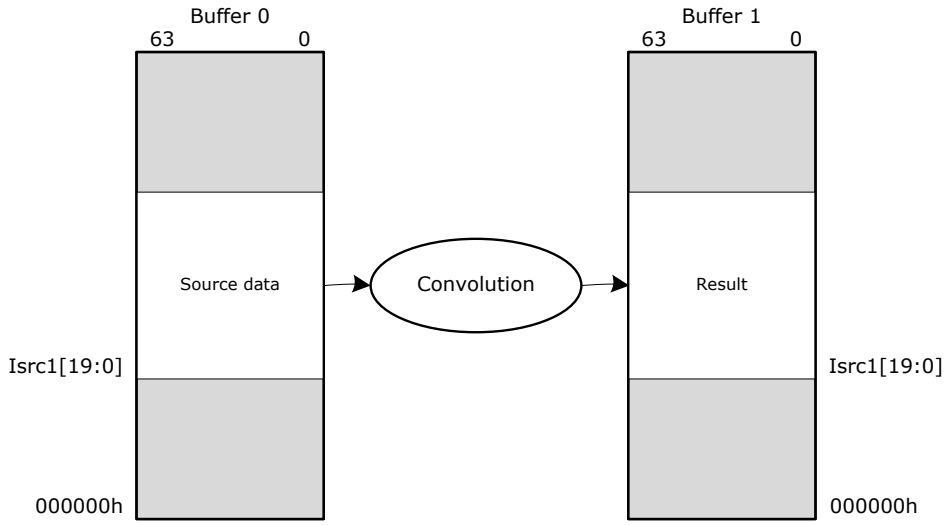
Isrc1 – bit 31 contains the index of the buffer where the original matrix is located. Bits [19:0] contain the number of columns of the matrix located in the external buffer QDR SSRAM. Bits [51:32] contain the number of rows of the same matrix.

Isrc2 – bits [19:0] contain the number of columns of the matrix that was loaded in the transposed form into the coefficient buffer.

Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VMC. Matrix convolution.

Convolution of matrices. The original matrix is in one of the external QDR SSRAM buffers. The kernel matrix must first be loaded into the internal kernel buffer using the VLK instruction. The source matrix can be located in the source buffer at any address. The resulting matrix will be placed in the opposite buffer at the same address.



Executed by:

Convolution engine

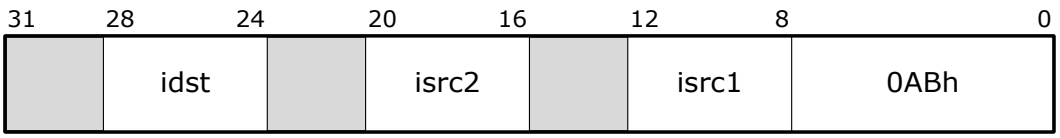
Mnemonics:

VMC idst,isrc2,isrc1

Example:

vmc r5,r6,r4

Format:



Instruction parameters:

Isrc1 – location address of the original matrix in one of the QDR SSRAM buffers. Bits [34:32] contain the kernel size code. 0 - 3x3, 1 - 5x5, 2 - 7x7, 3 - 9x9, 4 - 11x11, 5 - 13x13, 6 - 15x15.

Isrc2 – bits [15:0] – the number of columns of the matrix, bits [31:16] – the number of rows of the original matrix.

Idst – a register where the contents of the neuroblock status register are placed. The ZF flag in the AFR register corresponding to idst is set to 1 if the instruction is accepted for processing. If the neuroblock performs any operation, then ZF=0.

VLR. Load register.

The instruction loads a 64-bit value from a general purpose register into the neuroblock's register file.

Executed by:

Neuroblock logic

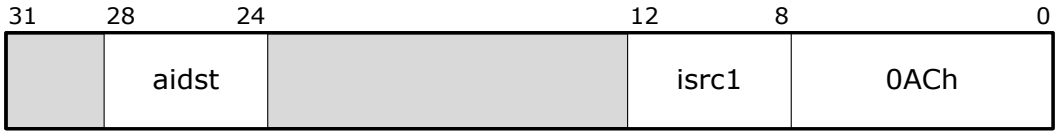
Mnemonics:

VLR aidst,isrc1

Example:

vlr auxr29,r13

Format:



Instruction parameters:

Isrc1 – a general-purpose register whose contents are written to the neuroblock register.

Aidst – neuroblock register.

VSR. Store register.

The instruction transfers a 64-bit word from the selected additional neuroblock register to the general purpose register.

Executed by:

Neuroblock logic

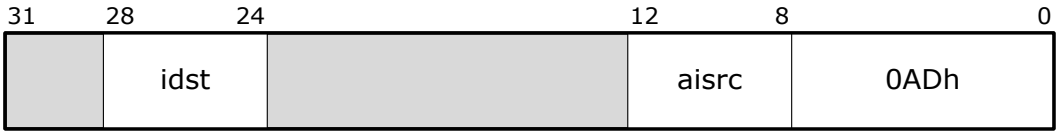
Mnemonics:

VSR idst,aisrc

Example:

vsr rq30,auxr30

Format:



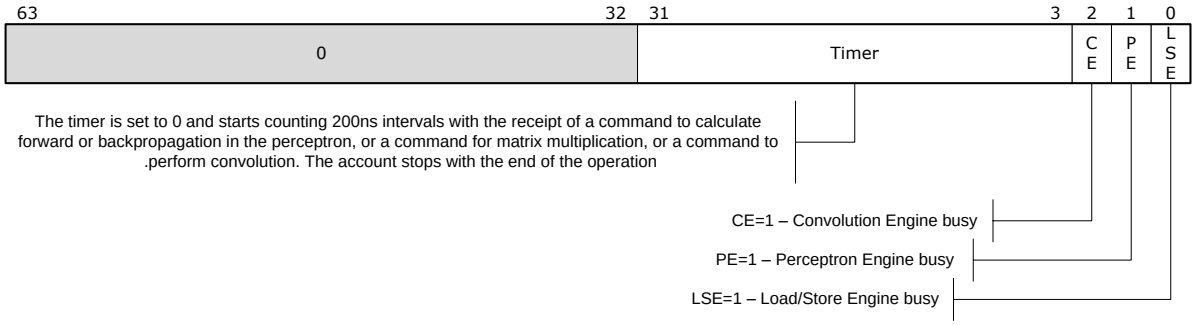
Instruction parameters:

Aisrc – neuroblock register from which a 64-bit value is written to a general-purpose register.

Idst – a general purpose register that receives data from the neuroblock register.

VRS. Read status.

Reading the status of a neuroblock. The status word is returned to the general purpose register:



Executed by:

Neuroblock logic

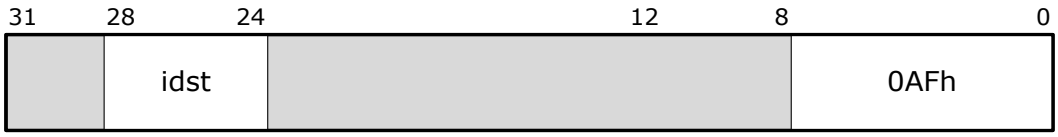
Mnemonics:

VRS idst

Example:

vrs r9

Format:



Instruction parameters:

Idst – a register where the contents of the neuroblock status register are placed.