

Using CM1K with your Microcontroller TECHNICAL BRIEF

This Technical Brief describes how easy it is to teach the neurons and use them to classify patterns from a developer's standpoint. For more details the CogniMem Reference Guide will give you information about the knowledge building concept and methodology, the usage of multiple neural networks, and more.

15 Registers and 4 Simple Operations

Usage of the CM1K comes down to 4 simple operations:

- Learn a vector
- Recognize a vector
- Save the knowledge built by the neurons
- Load a knowledge into the neurons

These operations are executed through Read and Write commands to the 15 neuron registers of the CM1K chip:

Register	Description	Addr	
COMP	Component	0x01	RW
LCOMP	Last Component	0x02	RW
DIST	Distance	0x03	R
CAT	Category	0x04	RW
AIF	Active Influence Field	0x05	R
NCR	Neuron Context Register	0x00	RW
NID	Neuron Identifier	0x06	R
NSR	Network Status Register	0x0D	RW
GCR	Global Context Register	0x00	RW
MINIF	Minimum Influence Field	0x06	RW
MAXIF	Maximum Influence Field	0x07	RW
FORGET	Clear the neurons	0x0F	W
NCOUNT	Committed neurons	0x0F	R
RESETCHAIN	Reset the neuron chain	0x0C	W

In the following examples, we will refer to vectors with a length L where L can range from 1 to 256. The L components of a vector are represented by vector[i] with i ranging from 0 to L-1.

Learning a Vector

Learning consists of broadcasting a vector to the neurons and assigning it a category. Optionally you can read the number of committed neurons to observe if a new neuron was committed.

```
For (i = 0; i < L-2, i++) Write CM_COMP, Vector (i);
Write CM_LCOMP, Vector (L-1)
Write CM_CAT, Category-to-learn
Read CM_NCOUNT, Ncount
```

Ncount will not be incremented if an existing committed neuron already recognizes Vector as belonging to Category-to-learn. However, the content of the committed neurons may have changed - in particular their influence fields.

Recognizing a Vector

Recognition consists of broadcasting a vector to the neurons and reading the distance and category of the "firing" neurons until the distance is equal to xFFFF. A firing neuron is a neuron recognizing the vector.

```
For (i = 0; i < L-2, i++) Write CM_COMP, Vector (i);
Write CM_LCOMP, Vector (L-1)
Do {
    Read CM_DIST, Output-distance
    Read CM_CAT, Output-category
```

```
} Loop Until (Output-distance == 0xFFFF)
```

Save the Knowledge Built by the Neurons

Reading the knowledge (to save it to file for example) consists of switching the network to Save-and-Restore mode, reading the memory and registers of all the committed neurons, and switching the network back to Learn-and-Recognize.

```
Write CM_NSR, 16
Write CM_RESETHAIN, 0
Do {
    For (i = 0; i < L, i++) Write CM_COMP, Vector(i);
    Write CM_NCR, Neuron_Context
    Write CM_AIF, Neuron_InfluenceField
    Write CM_CAT, Neuron_Category
} Loop Until (Neuron_Category == 0)
Write CM_NSR, 0
```

Load a Knowledge into the Neurons

Loading a pre-defined knowledge to the neurons consists of switching the network to Save-and-Restore mode, writing the memory and registers of the neurons, and switching the network back to Learn-and-Recognize.

```
Write CM_FORGET, 0
Write CM_NSR, 16
Write CM_RESETHAIN, 0
For (j=0; j < Number-of-Vectors; j++)
{
    For (i = 0; i < L, i++) Write CM_COMP, Vector(i);
    Write CM_NCR, Neuron_Context
    Write CM_AIF, Neuron_InfluenceField
    Write CM_CAT, Neuron_Category
}
Write CM_NSR, 0
```

Reading the Content of a Neuron

As another example, reading the content of a specific neuron with index k in the chain consists of switching the network to Save-and-Restore mode, pointing to the neuron k, reading its memory and registers, and switching the network back to Learn-and-Recognize.

```
Write CM_NSR, 16
Write CM_RESETHAIN, 0
For (i = 0; i < k-1, i++) Read CM_CAT, Dummy
For (i = 0; i < L, i++) Write CM_COMP, Vector(i);
```

```
Write CM_NCR, Neuron_Context
Write CM_AIF, Neuron_InfluenceField
Write CM_CAT, Neuron_Category
Write CM_NSR, 0
```

Timing Information

Reading and writing the neuron registers takes one system clock cycle with the exception of Write LCOMP (3cc), Read CAT (3 or 19 cc), Read DIST (18cc). The following table summarizes the execution time of the 5 basic commands:

Operation	Clock cycles
Broadcast a vector of length L	L+3
Learn the last vector	18
Recognize the last vector	37 (per firing neuron)
Read K th neuron	3+(K-1)+(L+3)
Save N neurons	3+(L+3)*N
Restore N neurons	3+(L+3)*N