

УДК 004.048, 004.272

РЕАЛИЗАЦИЯ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ НА МНОГОЯДЕРНОМ ПРОЦЕССОРЕ SEAFORTH

А. С. Анисимов¹, А. В. Калачев²

¹ФГУП БСКБ «Восток», Барнаул

²Физико-технический факультет АлтГУ, кафедра вычислительной техники и электроники

Получена 4 сентября 2010 г.

Аннотация. Работа посвящена реализации искусственной нейронной сети на матричных многоядерных процессорах. В качестве аппаратной платформы рассмотрен процессор SEAforth40. Приводятся краткие характеристики процессора. Проанализированы варианты построения нейронов и их сетей, применительно к особенностям архитектуры процессора. Подробно рассмотрен нейрон с пороговой функцией активации (персептрон), способ его представления и реализации, варианты расположения и взаимодействия нескольких персептронов. Представлена методика проектирования и работы искусственной нейронной сети. В качестве примера показана возможность реализации нейронной сети Хопфилда.

Ключевые слова. Искусственные нейронные сети, многоядерные процессоры, персептрон, диаграмма потока данных, модульность.

Введение

Существует большой класс задач, обладающих неполной или, наоборот, избыточной информацией. Такая ситуация возникает при распознавании объектов, человеческой речи, эффективного сжатия данных, медицинский и биометрический анализ образов, системы обработки теплового изображения, интеллектуальное управление, экономическое прогнозирование, прогнозирование потребления электричества, интеллектуальные антенны и т. п. Одним из способов решения таких задач являются искусственные нейронные

сети, которые зарекомендовали свою эффективность. Ряд задач требует своего решения в режиме реального времени, в условиях ограниченных ресурсов источников питания или размеров, поэтому нейровычислительные системы интересны с точки зрения архитектурно-схемотехнических решений, иными словами аппаратной реализации.

Элементной базой нейровычислительных систем являются заказные кристаллы (ASIC), встраиваемые микроконтроллеры (mC), процессоры общего назначения (GPP), программируемая логика (FPGA — ПЛИС), транспьютеры, цифровые сигнальные процессоры (DSP) и нейрочипы [1]. Использование любых из них позволяет сегодня реализовать нейровычислители, функционирующие в реальном масштабе времени, однако наибольшее использование при реализации нейровычислителей нашли ПЛИС, DSP и, конечно, нейрочипы.

Основной элементной базой перспективных нейровычислителей являются нейрочипы. Большинство из них на сегодня ориентированы на закрытое использование (для конкретных специализированных управляющих систем).

По способу представления информации нейрочипы можно разделить на цифровые, аналоговые и гибридные. По типу реализации нейроалгоритмов: нейрочипы с полностью аппаратной реализацией и с программно-аппаратной (когда нейроалгоритмы хранятся в ПЗУ).

По характеру реализации нелинейных преобразований: на нейрочипы с жёсткой структурой нейронов (аппаратно реализованные) и нейрочипы с настраиваемой структурой нейронов (перепрограммируемые).

По возможностям построения нейросетей: нейрочипы с жёсткой и переменной нейросетевой структурой [2].

Сводные характеристик наиболее распространенных нейрочипов можно посмотреть в обзоре [3].

Поскольку искусственная нейронная сеть представляет по своей сути - набор простых вычислительных узлов, связанных между собой, очевидный

путь её аппаратной реализации - множество АЛУ (возможно специализированных), имеющих между собой соответствующие связи. Естественные ограничения при этом - сложность изготовления кристалла (вычислительные устройства, связи), возрастающее энергопотребление, синхронизация, масштабирование, размещение и разрядность коэффициентов. Отчасти эти проблемы решаются в большинстве нейрочипов.

Реализация нейронной сети осуществляется в два этапа:

- 1) Выбор типа (архитектуры) нейронной сети.
- 2) Подбор весов (обучение) нейронной сети.

На первом этапе следует выбрать следующее:

- а) какие нейроны мы хотим использовать (число входов, передаточные функции);
- б) каким образом следует соединить их между собой;
- в) что взять в качестве входов и выходов нейронной сети.

Способ построения архитектуры напрямую связан с задачей, которую будет решать искусственная нейронная сеть [4-5].

На втором этапе следует "обучить" выбранную нейронную сеть, то есть подобрать такие значения ее весов, чтобы она работала нужным образом.

В данной статье рассмотрена возможность реализации нейрона и нейронной сети на многоядерном процессоре SEAForth40.

Процессор SEAForth40 (S40C18)

Процессор SEAForth-S40C18 является одним из процессоров семейства масштабируемых матричных процессоров фирмы IntellaSys[6]. Это массив из 40-х вычислительных узлов (ядер), каждый из которых имеет свое хранилище программ и данных (рис.1).

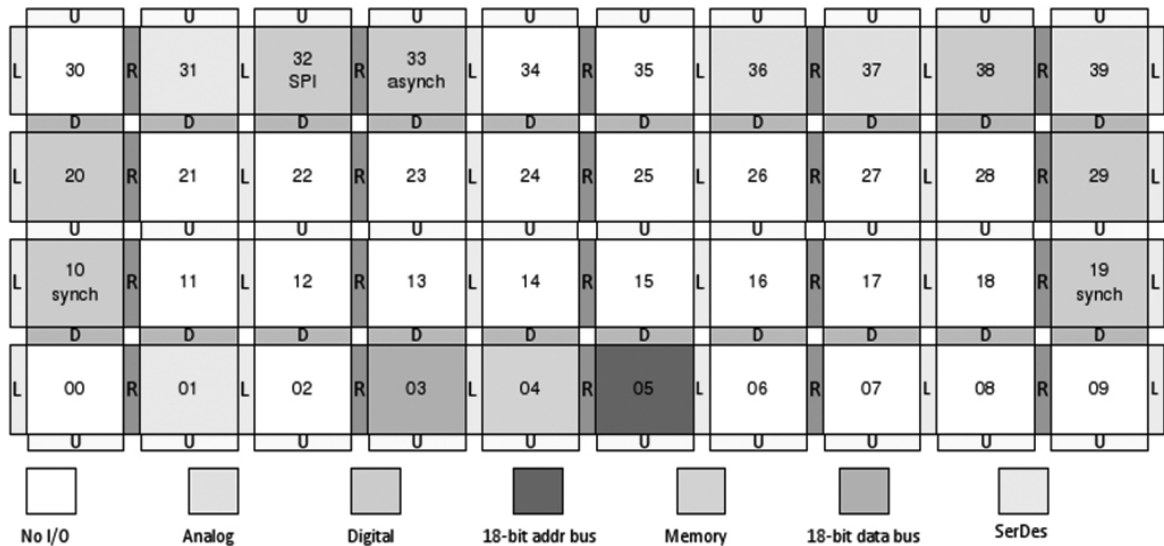


Рис. 1 Устройство процессора SeaForth-S40C18.

Решетка ядер при помощи внешних выводов может быть замкнута в «цилиндр» или «тор».

Каждый узел тактируется собственным тактовым генератором, что позволяет проектировать на основе SEAForth полностью асинхронные системы обработки данных.

Межпроцессорные коммуникации происходят автоматически. Обмен может осуществляться между соседними ядрами посредством общего порта. Ядро, ожидающее данные от соседа, автоматически переходит в спящее состояние, рассеивая при этом менее одного микроватта. И, напротив, узел, посылающий данные, соседу, который не готов к их приему, переходит в спящее состояние до тех пор, пока сосед не примет (не прочитает) данные. Внешние сигналы на линиях ввода-вывода также могут пробудить спящий узел. Отсюда получается низкое потребление электроэнергии процессора в целом. В каждый момент времени часть процессорных узлов, как правило, находятся в спящем состоянии. В типичных приложениях – соотношение активных/спящих ядер примерно 22/40. В дополнении к этому, разработчиками предложено оригинальное внутреннее представление данных в виде логических уровней, также снижающее энергопотребление. В результате, при тактовой частоте 1ГГц активное ядро потребляет порядка 9 мВт мощности.

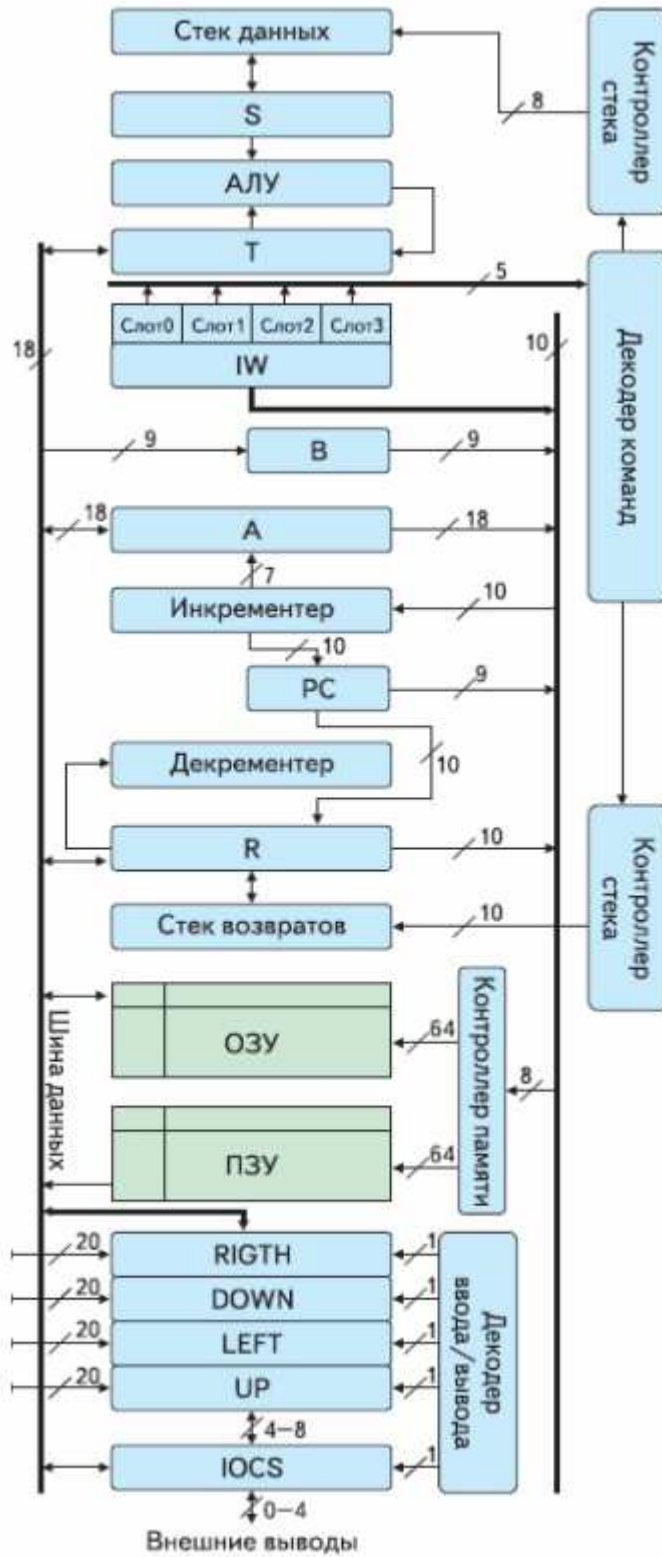


Рис. 2 Структура процессорного ядра.

На рисунке 2 изображено внутреннее устройство отдельного узла.

Каждое ядро процессора является одинаковым, за исключением ряда

периферийных ядер, имеющих дополнительные устройства ввода-вывода, и представляет собой 18-ти разрядную форт-машину с сокращенным набором команд. Ядро имеет два стека (стек данных и стек возвратов) глубиной 10 слов, ОЗУ и ПЗУ объемом 64 слова.

Систему команд ядра - VentureForth - содержит всего 32 команды. Естественно, для описания любой из этих инструкций требуется только 5 бит, что позволяет в одно машинное слово (18 разрядов) «упаковать» до четырех команд. Время выполнения команд от 1 до 4 нс, включая обмен между ядрами.

Как видно, ресурсы отдельного ядра достаточно сильно ограничены, положение несколько исправляет Фон-Неймановская архитектура ядер, позволяя во время работы менять программный код, расположенный в ОЗУ, при этом программный код может распределяться между ядрами и может быть получен с внешних устройств [7].

Возможны три варианта реализации нейронной сети на ядре процессора – это когда ядро реализует один нейрон, слой нейронов, синоптическую связь. Во всех вариантах необходимо учитывать, каким образом организовано поступление данных (входного вектора) и выдачи результатов во внешний мир либо на следующий слой сети.

Синтез нейрона

Процесс вычисления значения нейрона представляет собой движение потока данных и их преобразование. Сначала данные поступают на вход нейрона, следом происходит умножение исходных данных на веса связей, полученные значения преобразуются в одно числовое значение посредством суммирования, далее происходит обработка передаточной функцией нейрона и затем данные поступают на выход. Обобщенная блок-схема вычислений может быть представлена следующим образом (рис. 3)

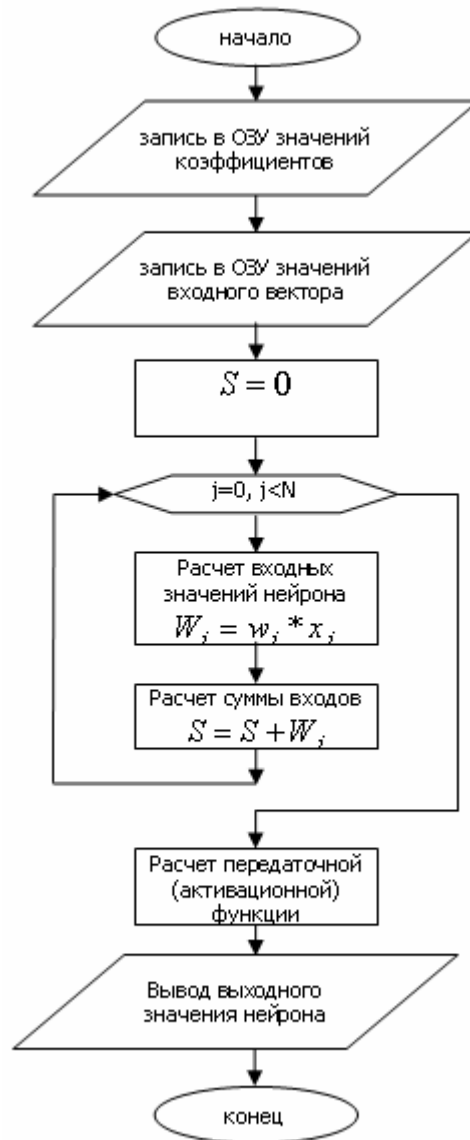


Рис. 3 Обобщенная блок-схема вычисления выходного значения нейрона.

Так как процесс вычисления представляет собой движение и преобразование потока данных, то несложно представить функциональное разбиение программы, реализующей нейрон.

В основе вычисления лежат три функции (слова - в терминологии языка Форт) – это функция подготовки данных к обработке функцией активации *main_loop*, слово, вычисляющее активационную функцию *pered_func* и слово, отвечающее за дальнейшее местоположение результата *rez_output*, то есть, например, передачу результата соседнему ядру. В свою очередь слово *main_loop* разбивается еще на несколько составляющих слов – слово, которое

принимает и подготавливает данные к умножению *take*, выбирает вес и соответствующее входное значение вектора, слово *mul*, производит умножение входного вектора на вес, слово *add* накапливающее сумму. Можно выделить еще одно слово *init*, которая будет подготавливать необходимые регистры и стек к дальнейшей работе.

Рассмотрим различные способы представления нейрона на процессоре данной архитектуры.

Для упрощения считаем, имеется ранее обученная нейронная сеть с известными параметрами. Веса связей известны, находятся во внешней памяти и в начальный момент времени загружаются в ядра вместе с исполнимым кодом. При этом веса могут храниться в оперативной памяти или на стеках.

Вариант первый. Каждый узел представляет собой полноценный нейрон с i входами. При размещении коэффициентов (весов) на стеке возможно реализовать 7-8 связей (коэффициенты 16-18ти разрядные). Количество связей при размещении весов в оперативной памяти будет зависеть от сложности алгоритмов вычисления нейрона (в частности, активационной функции) и, так же от разрядности весов. В общем случае можно рассчитывать в среднем на 10-20 связей.

Работа с входным вектором может быть организована двумя путями: весь вектор записывается в ОЗУ ядра или поступает последовательно через один из портов.

Весь процессор загружен выполнением расчетов, и максимум может представлять сеть из 40х нейронов. Подобная реализация позволяет реализовать многослойную сеть, то есть в заданные промежутки времени процессор представляет собой определенный слой нейронной сети. Для многослойной сети пути передачи данных - уникальны, слова, их реализующие, соответствующим образом модифицируются.

Вариант второй. Рассмотрим случай, когда ядро вычисляет слой ИНС. Этот вариант хорошо подходит, в тех случаях, когда реализуемая сеть достаточно большая в смысле количества слоев и количества нейронов в слое.

В данном случае межслойные связи могут быть уникальны, как и движение потока данных. При этом свободные ядра вместе с ядрами, которые уже отработали слой, могут конфигурироваться для решения иных задач. Все это целиком ложится на плечи разработчика. Недостатком данной схемы является большие временные задержки при вычислении выхода слоя и затраты памяти на хранение коэффициентов и входного вектора.

Третий вариант представляет собой схему, когда нейрон вычисляется на большинстве ядер процессора, остальные ядра могут представлять собой конвейер, по которому движется поток данных: весовые коэффициенты, входные и выходные вектора. В данном случае часть ядер отвечает за синоптические связи (в зависимости от числа синапсов нейрона таких связей может быть до 15), есть ядра (ядро) которые суммируют и накапливают результат, и есть ядра (ядро), которые вычисляют передаточную функцию нейрона. К преимуществам данной схемы стоит отнести постоянную загруженность процессорных ядер, что очевидно обеспечивает разумное использование вычислительных ресурсов.

Время вычисления 4-х входного персептрона, по схеме, когда каждое ядро представляет собой уникальный нейрон, составило около 800 нс, что является хорошим показателем, в сравнение с большинством нейрочипов.

Синтез нейронной сети

Как ранее уже упоминалось, с точки зрения эффективного использования вычислительных ресурсов желательно, чтобы все ядра процессора участвовали в вычисление искусственной нейронной сети.

На рисунке 4 представлен процесс вычисления ИНС на архитектуре процессора SEAforth S40C18.

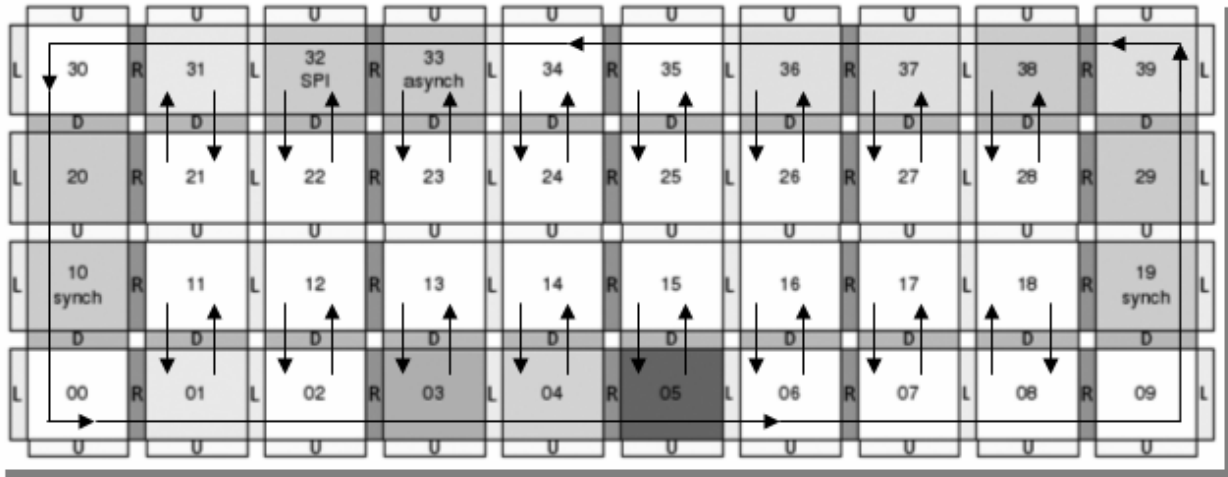


Рис. 4 Процесс вычисления ИНС на процессоре.

Ядра с №11-18 и №21-28 представляют собой нейроны каждого слоя ИНС. Остальные ядра участвуют в передачи выходов от предыдущего слоя следующему, представляя собой «конвейер», стрелками показан процесс движения выходного потока предыдущего слоя всем нейронам следующего. Преимуществом данного метода является возможность реализации программистом любой архитектуры ИНС.

Общий алгоритм вычисления ИНС на процессоре SEAForth S40C18:

- 1) входной вектор \vec{x} размерностью n – поступает в ядро №10 через последовательный синхронный порт, затем ядру №00;
- 2) побитно вектор \vec{x} с ядра №00 поступает на «конвейер», по которому согласно архитектуре реализуемой сети данные поступают на нейроны i -ого слоя сети (ядра №11-18, №21-28);
- 3) после того как выходной вектор \vec{y} получен, он поступает на «конвейер», по которому потом формируется в ядре №00;
- 4) согласно архитектуре сети через интерфейс SPI возможна загрузка в ядра как кода программы (например, другая передаточная функция, иная структура движения потока данных и т. д.) так и необходимых весовых коэффициентов синапсов следующего слоя сети;
- 5) затем вектор, полученный на предыдущей итерации, с ядра №00

поступает на «конвейер» и далее на нейроны следующего слоя;

- б) после того как выходной вектор сети получен, он поступает к ядру №10 и далее во внешний мир через последовательный синхронный порт.

На базе описанного выше алгоритма реализации ИНС на процессоре разработан автомат, представляющий собой совокупность всех ядер и связей между ними. Каждое ядро представляет собой циклическую последовательность выполняемых им действий в соответствии с назначением ядра. На рисунке 5 изображен автомат, состояния которого описаны ниже.

Состояния автомата:

- 0 – принятие \vec{x} \ передача \vec{y} через последовательный синхронный порт;
- 1 – передача вектора \vec{x} ядру 00;
- 2 – принятие вектора \vec{y} , полученного в результате работы сети;
- 3 – принятие вектора \vec{x} от ядра 10;
- 4 – передача i -ой компоненты входного вектора x_i ;
- 5 – принимает j -ую компоненту выходного вектора y_i ;
- 6 – компаратор, определение конца вычисления;
- 7 - принятие i -ой компоненты входного вектора x_i ;
- 8 – выбор из памяти весового коэффициента синапса w_i ;
- a – вычисление умножения с накоплением;
- b – вычисление передаточной функции и вывод результата y_i .

Таким образом, изменяя код того или иного состояния автомата, программист получает возможность реализовывать самые разнообразные связи между слоями искусственной нейронной сети.

Автомат обладает достаточной степенью универсальности, что позволяет реализовывать на процессоре сети разнообразных архитектур, внося соответствующие изменения в код программы ядер.

В структуре автомата реализации ИНС выделяется 4 типа ядер, которые

отвечают за определенные действия при вычислении ИНС: связь с внешними устройствам (ядро №10), анализ конца вычисления результата (№00), транспортировка данных, в том числе и нейронам сети (№01-09, 19, 29, 31-39), вычисления значений самих нейронов (№11-18, 21-28).

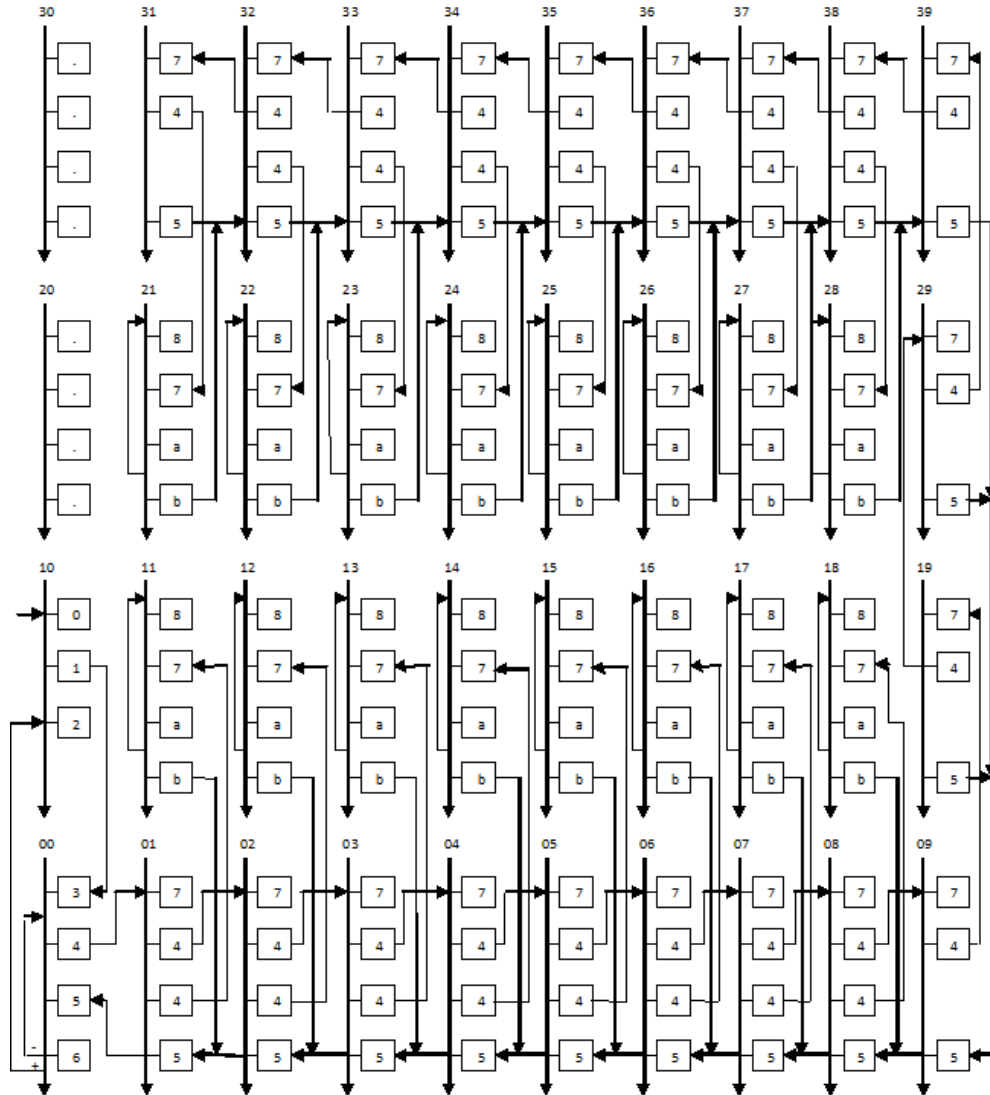


Рис. 5 Процесс вычисления ИНС на процессоре.

В качестве примера, показывающего возможности предложенной методики, реализуем на процессоре сеть Хопфилда [2, 8], состоящую из 16 нейронов, каждый из которых имеет 16 восьмиразрядных синапсов ($n=16$). В качестве активационной функции выступает пороговая. Структурная схема сети приведена на рисунке 6.

Ниже представлен алгоритм вычисления сети Хопфилда:

- 1) входной вектор \vec{x} поступает по синхронному последовательному

- порту в ядро 10, затем целиком передается ядру 00;
- 2) побитно против часовой стрелки вектор поступает 01-му ядру, затем второму и т.д;
 - 3) первый бит с ядра №02 поступает сначала к ядру №03, затем к ядру №12 (второй нейрон сети Хопфилда), одновременно умножаясь на вес соответствующего синапса и складывается с текущей суммой. И так далее для всех нейронов, в соответствии со структурой сети Хофилда;
 - 4) когда результата получен, ядра, отвечающие за нейроны, выдают его на «конвейер» и затем по часовой стрелке результат собирается в ядре №00 (в нем они располагаются в одном регистре);
 - 5) далее происходит сравнение принятого 16-битного выхода с предыдущим, находящемся в памяти. Если результат совпал, значит выход получен, передаем его 10-му ядру, иначе побитно слово снова поступает на «конвейер» против часовой стрелки, на вход нейронной сети. И так далее, пока результат не совпадет с предыдущим.

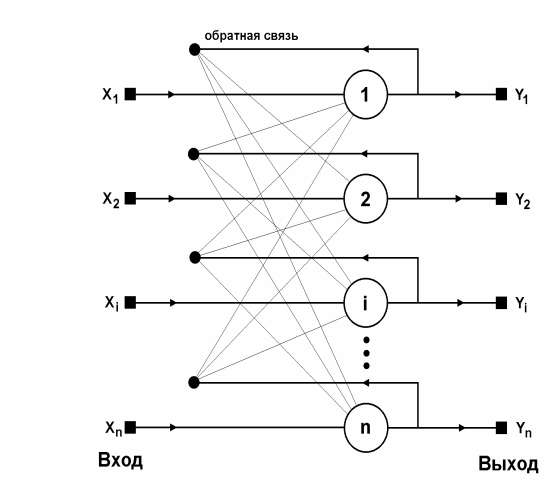


Рис. 6 Структурная схема сети Хопфилда (n=16).

Данный алгоритм был закодирован на языке VentureForth и успешно выполнен в симуляторе процессора. По результатам видно, что в каждый момент времени при вычислении сети большая часть ядер занята. При этом

легко проследить, что во время вычисления значений слоя сети, правила передачи данных на вход следующего слоя могут легко измениться, путем загрузки кода из внешней памяти.

В заключение, хотелось бы отметить преимущества аппаратной реализации нейронной сети на процессоре SEAforth-40. Это, прежде всего, скорость вычисления, что объясняется наличием достаточного числа вычислительных узлов (ядер) работающих параллельно. Надежность, так как полученная схема обладает меньшей отказоустойчивостью в сравнении с вычислительными комплексами. Специальные эксплуатационные режимы работы процессора, которые позволяют снизить энергопотребление системы и, следовательно, сделать ее более мобильной. Безопасность – злоумышленнику будет сложнее выявить принцип и логику работы нейронной сети при таком типе реализации.

Литература

1. Техническая поддержка, [neuroproject.ru](http://www.neuroproject.ru) [Электронный ресурс]: Учебник – Нейронные сети.: 2008 – Режим доступа к ресурсу.: <http://www.neuroproject.ru/neuro.php>.
2. Осовский С. Нейронные сети для обработки информации / Пер. с польского И.Д. Рудинского. – М: Финансы и статистика, 2002. – 344 с.: ил.
3. К.И. Билибин, А.И. Власов, Л.В. Журавлева и др. Под общ. Ред. В.А. Шахнова. Конструкторско-технологическое проектирование электронной аппаратуры: Учебник для вузов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. – 528 с.: ил. – (Сер. Информатика в техническом университете.)
4. Техническая поддержка, md-it.ru [Электронный ресурс]: Аппаратная реализация нейронных сетей.: 2008 – Режим доступа к ресурсу.: <http://md-it.ru/articles/html/article58.html>.
5. Корнеев В. В., Киселев А. В. Современные микропроцессоры. – 3-е изд., перераб. И доп. – СПб.: БХВ-Петербург, 2003. – 448 с.: ил.

6. IntellaSys, inc [Электронный ресурс]: Знакомство с масштабируемым процессором SeaForth-S40C18. / Официальный сайт IntellaSys.: 2008 – Режим доступа к ресурсу.: <http://www.IntellaSys.net/>.

7. Компоненты и технологии: Ежемес. журн.. Процессоры семейства SEAForth. – М., №4 2009. – 43 с.

8. Круглов В. В., Дли М. И., Голунов Р. Ю. Нечеткая логика и искусственные нейронные сети. - 3-е издание.- М.: Питер, 2002 – 224 с.