УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Программная инженерия

Дисциплина «Организация вычислительных систем»

Лабораторная работа № 3

Выполнил

Зиновичев Е. С.

Группа

P4119

Преподаватель

Быковский С. В.

г. Санкт-Петербург

2023 г.

Задачи

1. Разработать архитектуру вычислителя, предназначенного для моделирования работы нейронной сети, полученной по итогам выполнения лабораторной работы №1. В качестве основы можно использовать структуру вычислителя, представленную на рисунке 1. Рассмотреть варианты распараллеливания и конвейеризации вычислений.
2. Разработать микроархитектуру функциональных узлов нейросетевого процессора.
3. Разработать формат описания сети и алгоритм конфигурирования/программирования процессора

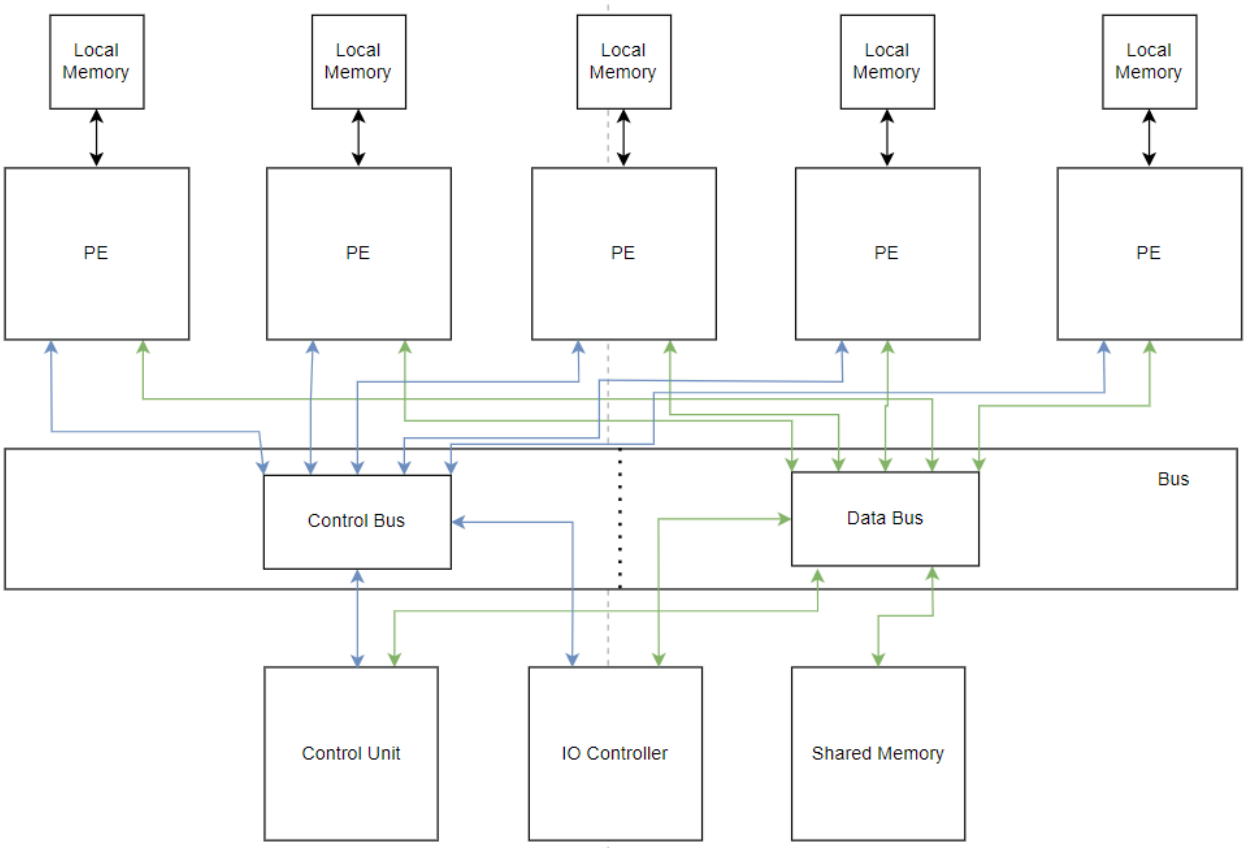
Структура выбранной нейронной сети для тестирования

По итогам лабораторной работы № 1 была выбрана следующая структура нейронной сети:

- Количество входов: 49
- Количество выходов: 3
- Количество слоев: 2
- Количество нейронов в скрытом слое: 5

Однако представленная архитектура поддерживает и другие структур нейронных сетей. Для тестирования будут использоваться все рассмотренные структуры нейронных сетей в лабораторной работе № 1 для оценки качества распределения ресурсов на нейросетевом процессоре.

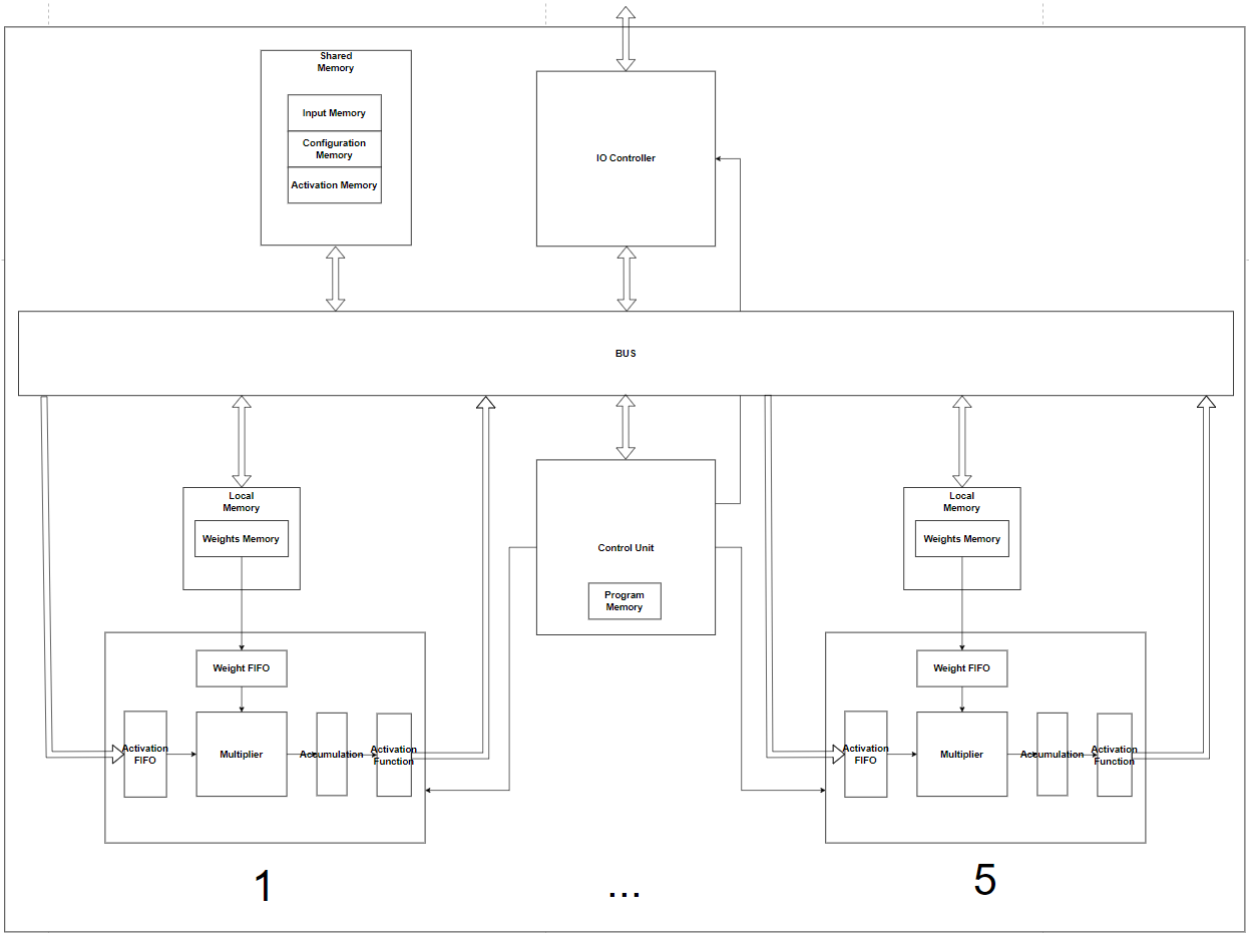
Структура нейросетевого процессора



Неройсетевой процессор будет содержать:

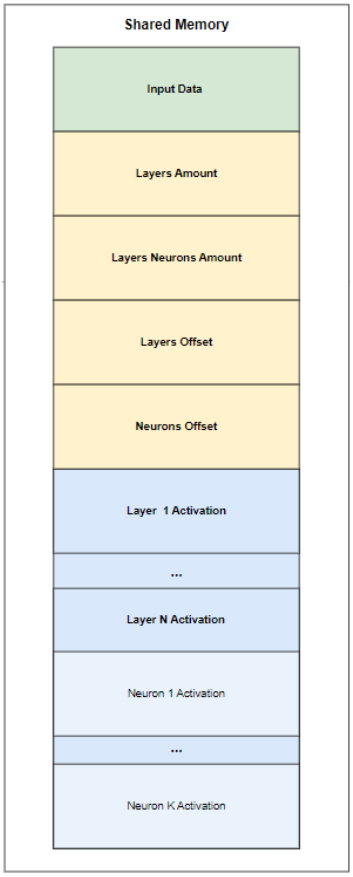
- 5 вычислительных ядер. Было принято такое решение, опираясь на выбранную структуру нейронной сети из первой лабораторной работы, такая организация подойдет и для других вариантов структур
- Устройство для управления вычислительными блоками, взаимодействия с устройствами ввода-вывода, чтения/записи конфигурационных данных и данных для вычислений
- Общую память для хранения данных для проведения вычислений
- Шину данных для общения функциональных узлов

Архитектура нейросетевого процессора



На блок-схеме представлена архитектура нейросетевого процессора, где показаны соединения шины данных, а также управляющих сигналов от устройства управления. В данном процессоре расчет выхода нейрона будет занимать один вычислительный блок.

Организация общей памяти



Память разделена на следующие сегменты:

1. Сегмент входных данных, который содержит входные данные для расчета нейронной сети.
2. Сегмент, в котором хранится количество слоев
3. Сегмент, в котором хранится количество нейронов на каждом слое
4. Сегмент, в котором хранится количество вычисленных слоев
5. Сегмент, в котором хранится количество вычисленных нейронов на слое
6. Сегмент, в котором хранятся активации нейронов

Память подключается к общей шине, каждое ядро может имеет чтения/записи памяти.

Первые 3 сегмента не изменяются в процессе вычисления выхода сети, коэффициенты активаций нейронов и веса изменяются вычисляются и подстраиваются.

Организация локальной памяти

Локальная память необходима для хранения весов для вычисления выхода сети. Использование локальной памяти позволяет снизить нагрузку на общую шину данных и увеличить быстродействие за счет близкого расположения к вычислительному ядру.

Линии локальной памяти:

	clk_i
	addr_bi
	data_bi
local mem	data_o
	wr_i
	rd_i
	addr_i

Устройство управления

Устройство управления ответственно за следующее:

1. Загрузка конфигурационных данных, активаций и весов сети в общую и локальную память процессора через контроллер ввода-вывода.
2. Вывод результата работы нейросети через контроллер ввода-вывода.
3. Распределение нагрузки на вычислительные блоки процессора и передачи конфигурационных данных в данные блоки для загрузки коэффициентов активаций и весов.
4. Опрашивание вычислительных блоков на их занятость.
5. Подсчет номера вычисляемого слоя и нейрона.

Устройство управления отдельно связано сигнальными линиями с другими функциональными узлами, по которым передаются управляющие сигналы.

Также в нем будет содержаться память для прошивки, которая реализует логику управления.

Линии устройства управления:

clk_i
ioc_wr_o
ioc_rd_o
ioc_busy_i
first_cpu_busy_i
second_cpu_busy_i
third_cpu_busy_i
fourth_cpu_busy_i
fifth_cpu_busy_i
data_bi
addr_bo
rd_bo
wr_bo
first_cpu_start_o
second_cpu_start_o
third_cpu_start_o
fourth_cpu_start_o
fifth_cpu_start_o
a_size_first_cpu_o

	a_size_second_cpu_o
	a_size_third_cpu_o
	a_size_fourth_cpu_o
control unit	a_size_fifth_cpu_o
	curr_neuron_first_cpu_o
	curr_neuron_second_cpu_o
	curr_neuron_third_cpu_o
	curr_neuron_fourth_cpu_o
	curr_neuron_fifth_cpu_o
	w_offset_first_cpu_o
	w_offset_second_cpu_o
	w_offset_third_cpu_o
	w_offset_fourth_cpu_o
	w_offset_fifth_cpu_o
	a_offset_first_cpu_o
	a_offset_second_cpu_o
	a_offset_third_cpu_o
	a_offset_fourth_cpu_o
	a_offset_fifth_cpu_o
	n_offset_first_cpu_o
	n_offset_second_cpu_o
	n_offset_third_cpu_o
	n_offset_fourth_cpu_o
	n_offset_fifth_cpu_o

Контроллер ввода-вывода

Контроллер ввода-вывода по запросу от управляющего устройства осуществляет чтение данных из файла, вывод результата и загрузку данных в общую и локальную память по шине данных.

Линии контроллера ввода-вывода:

	clk_i
	addr_bo
	data_bi
	data_bo
io controller	cu_wr_i
	cu_rd_i
	wr_bo
	rd_bo
	busy_o

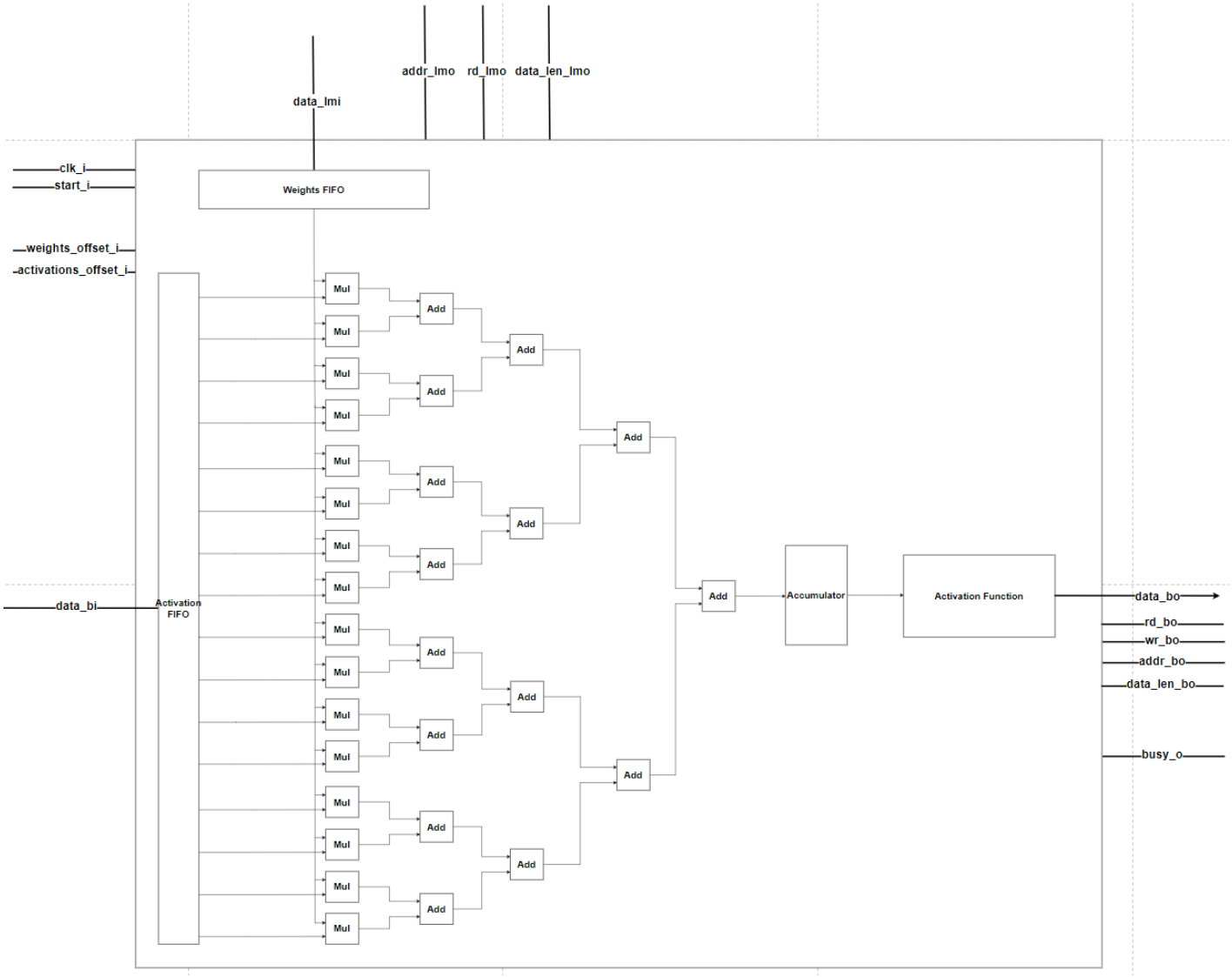
Вычислительный блок

Вычислительный блок осуществляет подсчет выхода одного нейрона. Для этого он загружает коэффициенты активаций и соответствующие веса. Они перемножаются между собой и аккумулируются, после этого к ним применяется функция активации. На выходе

результат будет записан в общую память в те секции, которые содержат старые коэффициенты активаций текущего слоя.

Ядро содержит следующие блоки:

1. Очереди активаций и весов необходимые для вычисления текущего нейрона.
2. 16 умножителей и 8 сумматоров
3. Блок вычисления функции активации



На рисунке представлена микроархитектура вычислительного ядра. На них изображены сигналы для общения по общей шине, управляющие и конфигурационные сигналы от устройства управления и линии для связи с локальной памятью.

Линии вычислительного блока:

	clk_i
	addr_bo
	data_bi
	start_i
	a_size_i
	curr_neuron_i
	w_offset_i
	a_offset_i
cpu	n_offset_i
	data_lmi
	busy_bi
	d_ready_i
	data_bo
	addr_lmo
	rd_lmo
	wr_o
	rd_o
	busy_o

Общая шина

Через шину осуществляется общения всех функциональных узлов процессора. Сигналы представляют собой одноразрядное значение, данные кодируются 32-разрядными значениями, что позволяет использовать типы int и float. Она содержит следующие линии:

clk_i
first_cpu_addr_i
second_cpu_addr_i
third_cpu_addr_i
fourth_cpu_addr_i
fifth_cpu_addr_i
ioc_addr_i
first_cpu_data_i
second_cpu_data_i
third_cpu_data_i
fourth_cpu_data_i
fifth_cpu_data_i
ioc_data_i
cu_addr_i
first_cpu_wr_i
second_cpu_wr_i
third_cpu_wr_i
fourth_cpu_wr_i
fifth_cpu_wr_i
ioc_wr_i
first_cpu_rd_i

	second_cpu_rd_i
bus	third_cpu_rd_i
	fourth_cpu_rd_i
	fifth_cpu_rd_i
	ioc_rd_i
	cu_rd_i
	cu_wr_i
	first_cpu_data_o
	second_cpu_data_o
	third_cpu_data_o
	fourth_cpu_data_o
	fifth_cpu_data_o
	first_lmem_addr_o
	first_lmem_data_o
	second_lmem_addr_o
	second_lmem_data_o
	third_lmem_addr_o
	third_lmem_data_o
	fourth_lmem_addr_o
	fourth_lmem_data_o
	fifth_lmem_addr_o
	fifth_lmem_data_o
	ioc_data_o
	cu_data_o
	busy_o
	first_cpu_busy_o
	second_cpu_busy_o
	third_cpu_busy_o
	fourth_cpu_busy_o
	fifth_cpu_busy_o
	first_cpu_d_ready_o
	second_cpu_d_ready_o
	third_cpu_d_ready_o
	fourth_cpu_d_ready_o
	fifth_cpu_d_ready_o
	first_lmem_wr_o
	first_lmem_rd_o
	second_lmem_wr_o
	second_lmem_rd_o
	third_lmem_wr_o
	third_lmem_rd_o
	fourth_lmem_wr_o
	fourth_lmem_rd_o
	fifth_lmem_wr_o
	fifth_lmem_rd_o

Алгоритм расчета выхода сети на ресурсах процессора

1. Модуль управления подает сигнал контроллеру ввода-вывода для загрузки конфигурационной информации и коэффициентов активаций в общую память.
2. Модуль управления подает сигнал контроллеру ввода-вывода для загрузки весов сетей в локальную память.
3. Модуль управления проверяет наличие свободных ядер, если таковые есть, то он подает сигнал начала работы на данные ядра, передает данные свободному ядру для загрузки коэффициентов активаций из общей памяти и весов из локальной.
4. Свободное ядро выполняет загрузку коэффициентов активаций и весов в буферы, а также проводит расчет выхода нейрона.
5. Далее устройство управления ожидает свободного ядра, если все выходы нейронов текущего слоя посчитаны, то выполняется переход на пункт 6.
6. Иначе модуль управления подает сигнал начала работы свободного ядра, передает необходимые данные для загрузки весов, необходимых для расчета выхода следующего нейрона, в вычислительный блок из локальной памяти, далее выполняется пункт 4.
7. Если посчитанный слой последний, то устройство управления подает сигнал на контроллер ввода-вывода для осуществления вывода результата в консоль, иначе алгоритм повторяется с пункта 3.

Оценка времени вычисления выхода сети на ресурсах процессора и загрузки вычислительных ядер

Для оценки данных характеристик была использована следующая структура нейронной сети:

- Количество входов: 49
- Количество выходов: 3
- Количество слоев: 2
- Количество нейронов в скрытом слое: 5

С учетом того, что процессор будет работать на частоте 100 МГц, мы получаем следующие характеристики:

Время вычисления выхода сети: *220 нс*

Среднее время загрузки процессорного ядра: *23 нс*

Вывод

В данной лабораторной работе была разработана архитектура нейросетевого процессора, которая может поддерживать различные структуры нейронных сетей для вычисления. Наличие 5 вычислительных ядер обеспечивает параллелизм вычислений, однако узким местом в данной системе является общая шина данных, через которую общаются функциональные узлы, так как возникает необходимость ожидания готовности шины данных к работе для получения данных из памяти. После этапа проектирования

удалось создать модель данного нейросетевого процессора с использованием библиотеки SystemC.