

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

ЛАБОРАТОРНАЯ РАБОТА №1

по дисциплине
“Системы на кристалле”

Вариант №3

Студент:

Чернова Анна Ивановна

Миху Вадим Дмитриевич

Группа Р34301

Преподаватель:

Леонид

г. Санкт-Петербург

2024

Цель работы:

Получить базовые навыки создания микропроцессорной СнК на базе сложно функциональных блоков (IP-ядер).

Вариант 3:

Реализовать КИХ фильтр цифрового сигнала. Использовать КИХ фильтр для вычисления скользящего среднего. Для решения данной прикладной задачи коэффициенты фильтра рассчитываются по формуле.

Выполнение:

Листинг разработанной программы с комментариями:

```
#include <stdio.h>

#include "platform.h"

#include "xil_printf.h"

#include "xil_io.h"

#include "microblaze_sleep.h"


int read_int() {
    int num = 0;
    char ch = getchar();
    while (ch >= '0' && ch <= '9') {
        num = num * 10 + (ch - '0');
        putc(ch, stdout);
        ch = getchar();
    }
    print("\n");
    return num;
}


void put_int(int num) {
    char buffer[10];
    int i = 0;
```

```

while (num > 0) {
    buffer[i++] = (num % 10) + '0';
    num /= 10;
}
while (i--) {
    putc(buffer[i], stdout);
}
print("\n");
}

int main()
{
    init_platform();
    print("Enter p (sliding window): ");
    int p = read_int();
    print("Enter n (number count): ");
    int n = read_int();

    int buffer[100];
    int mid_buffer[100];

    Xil_Out16(0x40000000, 0xAAAA);
    print("Write numbers:\n");
    // Read numbers and calculate sliding window
    for(int i = 0; i < n; i++) {
        buffer[i] = read_int();
        if (i > p - 2) {
            int sum = 0;
            for (int j = 0; j < p; j++) {
                sum = sum + buffer[i - j];
            }
            int med = sum / p;

```

```

        put_int(med);

        mid_buffer[i - p + 1] = med;

    }

}

//Output numbers

for (int k = 0; k < n - p + 1; k++) {

    Xil_Out16(0x40000000, mid_buffer[k]);

    MB_Sleep(1000);

}

Xil_Out16(0x40000000, 0xAAAA);


cleanup_platform();

return 0;

}

```

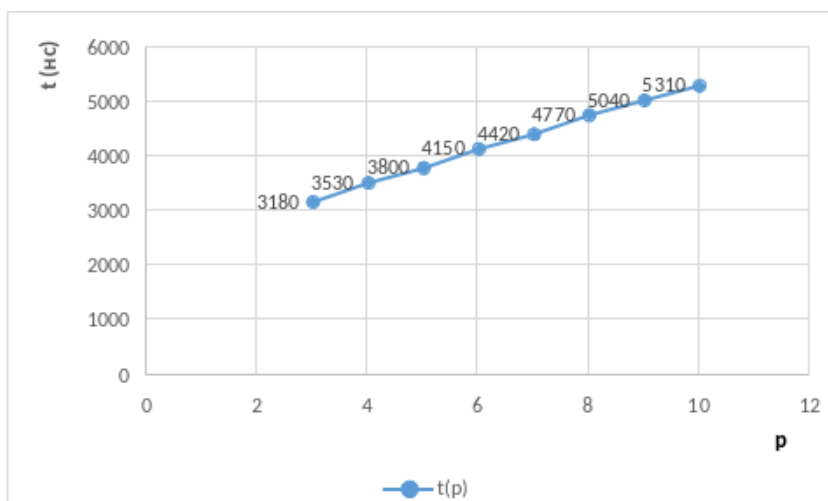
Анализ характеристик:

Используемые ресурсы на ПЛИС

Name	Slice LUTs (63400)	Slice Registers (126800)	F7 Muxes (31700)	Slice (15850)	LUT as Logic (63400)	LUT as Memory (19000)	Block RAM Tile (135)	DSPs (240)	Bonded IPADs (2)	BUFIO (24)	PLLE2_ADV (6)
design_1_wrapper	1157	950	108	425	1041	116	950	16	20	2	1
design_1_i (design_1)	1157	950	108	425	1041	116	16	0	0	0	0

Зависимость размерности

Зависимость линейная от количества элементов аппроксимации (скользящего окна), так как расчет среднего арифметического зависит только от количества элементов



Вывод:

Мы научились создавать базовые программы, используя айпи ядро микроблейз, синтезировать его на плис.