Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное

образовательное учреждение высшего образования

Санкт-Петербургский университет аэрокосмического приборостроения

КАФЕДРА № 2

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доцент |  |  |  | А.А. Попов |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБАРАТОРНОЙ РАБОТЕ |
| ПРИНЦИПЫ ОРГАНИЗАЦИИ ПАРАЛЛЕЛЬНОГО ВЫПОЛНЕНИЯ КОМАНД |
| по дисциплине: АРХИТЕКТУРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4134к |  |  |  | Столяров Н.С. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт – Петербург, 2023

ЦЕЛЬ РАБОТЫ: освоение принципов построения приложений на языке ассемблера для системы Texas Instruments, ознакомление с командами и правилами построения программ в соответствии с особенностями организации циклов.

## Вариант 19

Разработать программу подсчета суммы элементов массива у которых нечетные индексы. 

## Листинг программы

; Разработать программу подсчета суммы элементов массива у которых нечетные

; индексы.

**.global** \_c\_int00

;///////////////////////////////////////

**.data** ;секция данных

**array:** .ubyte 1, 2, 3, 4, 5, 1

**sum:** .ubyte 0

size **.set** 6

;///////////////////////////////////////

**\_c\_int00:**

**.text** ;секция кода

;Инициализация:

MVKL .S1 array, A3 ;загружаем адрес массива1 в A3

MVKH .S1 array, A3

MVKL .S2 array, B3 ;загружаем адрес массива1 в A3

MVKH .S2 array, B3

MVK .S1 size, A2

|| MVK .S2 -1, B2

MVK .S1 sum, A8

LDB .D1 \*A8, A8

;; NOP 4

**LOOP:**

**SUB** .S1 A2, 1, A2

|| ADD .S2 B2, 1, B2

;; проверка на чётность индекса

SHR .S1 A2, 1, A1

|| SHR .S2 B2, 1, B1

**MPY** .M1 A1, 2, A1

|| MPY .M2 B1, 2, B1

**NOP** 2

**SUB** .L1 A2, A1, A1

|| SUB .L2 B2, B1, B1

LDB .D1 \*A3[A2], A4

|| LDB .D2 \*B3[B2], B4

**NOP** 4

[!A1] B .S1 SKIPONE

**NOP** 5

**ADD** .L1 A8, A4, A8

**SKIPONE:**

[!B1] B .S2 SKIPTWO

**NOP** 5

**ADD** .L1 A8, B4, A8

**SKIPTWO:**

;; проверка на середину массива

**SUB** .L1 A2, B2, A0

[!A0] B .S1 EXIT

**NOP** 5

|| ADD .L1 B2, 1, A0

**SUB** .L1 A2, A0, A0

[!A0] B .S1 EXIT

**NOP** 5

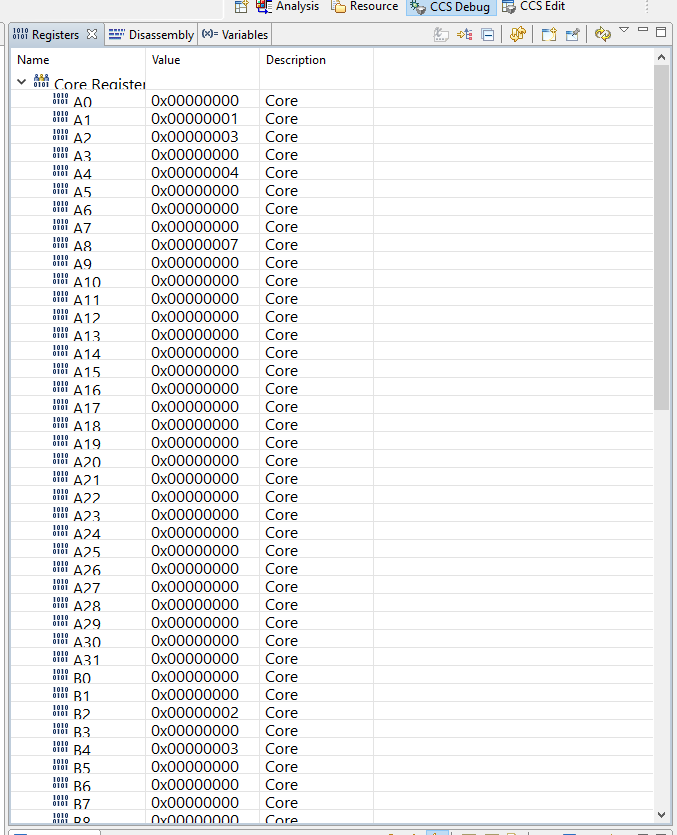
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

**B** .S1 LOOP ;переход если A2 <> 0

**NOP** 5

**EXIT:**

## Результат работы

****

## **Вывод**

Я освоил принципы программирования микроконтроллера и ознакомился с принципами построения программы на assembler. Программа выполняется за 136 тактов, что почти в три раза быстрее предыдущей реализации без параллельного метода