УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Основы профессиональной деятельности»

**Отчет**

По лабораторной работе №7

Вариант 321

Студент

*Митрофанов Е. Ю.*

*P3114*

Преподаватель

*Николаев В. В.*

Санкт-Петербург, 2020 г.

Текст задания:

Синтезировать цикл исполнения для выданных преподавателем команд. Разработать тестовые программы, которые проверяют каждую из синтезированных команд. Загрузить в микропрограммную память БЭВМ циклы исполнения синтезированных команд, загрузить в основную память БЭВМ тестовые программы. Проверить и отладить разработанные тестовые программы и микропрограммы.

* ANDSP - Логическое И двух верхних чисел на вершине стека, результат поместить на стек, установить признаки N/Z
* Код операции - 0F01
* Тестовая программа должна начинаться с адреса 040416

Исходный код синтезируемой команды

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес МП | Микрокоманда | Описание | Комментарий |
| BB | 81F0014002 | if CR(8) = 1 then GOTO @ E0 | Исправленная ячейка интерпретатора |
| F0 | 0080009008 | SP -> AR | Первое значение стека |
| F1 | 0100000000 | MEM(AR) -> DR | в DR |
| F2 | 0020009001 | DR -> BR | Первое значение в BR |
| F3 | 0080009408 | SP + 1 -> AR | Второе значение стека |
| F4 | 0100000000 | MEM(AR) -> DR | в DR |
| F5 | 0001809821 | BR & DR -> DR; N, Z | Лог. И, знаки N, Z |
| F6 | 0088009208 | ~0 + SP -> SP, AR | Результат выражения |
| F7 | 0200000000 | DR -> MEM(AR) | в стек |
| F8 | 80C4101040 | GOTO INT @ C4 | Переход к циклу прер. |

Код тестовой программы на языке ассемблер

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | ORG | 0x3F8 |
| ARG1: | WORD | 0x0000 |
| ARG2: | WORD | 0xAAAA |
| ARG3: | WORD | 0x5555 |
| ARG4: | WORD | 0x7FFF |
| ARG5: | WORD | 0x8765 |
| CHECK1: | WORD | 0xFFFF |
| CHECK2: | WORD | 0xFFFF |
| CHECK3: | WORD | 0xFFFF |
| FINAL: | WORD | 0xFFFF |
| RES1: | WORD | 0x0000 |
| RES2: | WORD | 0x0000 |
| RES3: | WORD | 0x0765 |
| START: | CLA |  |
|  | CALL | TEST1 |
|  | HLT |  |
|  | CALL | TEST2 |
|  | HLT |  |
|  | CALL | TEST3 |
|  | HLT |  |
|  | LD | #0x1 |
|  | CLC |  |
|  | AND | CHECK1 |
|  | AND | CHECK2 |
|  | AND | CHECK3 |
|  | ST FINAL |  |
| STOP: | HLT |  |
|  |  |  |
| TEST1: | LD | ARG1 |
|  | PUSH |  |
|  | PUSH |  |
|  | WORD | 0x0F01 |
|  | HLT |  |
|  | POP |  |
|  | CMP | RES1 |
|  | BEQ | DONE1 |
|  | ST | RES1 |
|  | POP |  |
|  | POP |  |
|  | CLA |  |
|  | ST | CHECK1 |
|  | RET |  |
| DONE1: | ST | RES1 |
|  | POP |  |
|  | POP |  |
|  | LD | #0x1 |
|  | ST | CHECK1 |
|  | CLA |  |
|  | RET |  |

|  |  |  |
| --- | --- | --- |
| TEST2: | LD | ARG2 |
|  | PUSH |  |
|  | LD | ARG3 |
|  | PUSH |  |
|  | WORD | 0x0F01 |
|  | HLT |  |
|  | POP |  |
|  | CMP | RES2 |
|  | BEQ | DONE2 |
|  | ST | RES2 |
|  | POP |  |
|  | POP |  |
|  | CLA |  |
|  | ST | CHECK2 |
|  | RET |  |
| DONE2: | ST | RES2 |
|  | POP |  |
|  | POP |  |
|  | LD | #0x1 |
|  | ST | CHECK2 |
|  | CLA |  |
|  | RET |  |
| TEST3: | LD | ARG4 |
|  | PUSH |  |
|  | LD | ARG5 |
|  | PUSH |  |
|  | WORD | 0x0F01 |
|  | HLT |  |
|  | POP |  |
|  | CMP | RES3 |
|  | BEQ | DONE3 |
|  | ST | RES3 |
|  | POP |  |
|  | POP |  |
|  | CLA |  |
|  | ST | CHECK3 |
|  | RET |  |
| DONE3: | ST | RES3 |
|  | POP |  |
|  | POP |  |
|  | LD | #0x1 |
|  | ST | CHECK3 |
|  | CLA |  |
|  | RET |  |

Таблица трассировки микропрограммы

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| МР до выборки МК | Содержимое памяти и регистров процессора после выборки микрокоманды | | | | | | | | | |
| **MR** | **IP** | **AR** | **AR** | **DR** | **SP** | **BR** | **AC** | **NZVC** | **СчМК** |
| F0 | 0080009008 | 411 | 0F01 | 7F4 | 0F01 | 7F4 | 410 | 5555 | 0001 | F1 |
| F1 | 0100000000 | 411 | 0F01 | 7F4 | 5555 | 7F4 | 410 | 5555 | 0001 | F2 |
| F2 | 0020009001 | 411 | 0F01 | 7F4 | 5555 | 7F4 | 5555 | 5555 | 0001 | F3 |
| F3 | 0080009408 | 411 | 0F01 | 7F5 | 5555 | 7F4 | 5555 | 5555 | 0001 | F4 |
| F4 | 0100000000 | 411 | 0F01 | 7F5 | AAAA | 7F4 | 5555 | 5555 | 0001 | F5 |
| F5 | 0001809821 | 411 | 0F01 | 7F5 | 0000 | 7F4 | 5555 | 5555 | 0101 | F6 |
| F6 | 0088009208 | 411 | 0F01 | 7F3 | 0000 | 7F3 | 5555 | 5555 | 0101 | F7 |
| F7 | 0200000000 | 411 | 0F01 | 7F3 | 0000 | 7F3 | 5555 | 5555 | 0101 | F8 |
| F8 | 80C4101040 | 411 | 0F01 | 7F3 | 0000 | 7F3 | 5555 | 5555 | 0101 | C4 |

Методика проверки

1. Загрузить тестовую программу в память базовой ЭВМ.
2. Запустить основную программу с адреса 0x404 в режиме работа.
3. Дождаться останова.
4. Проверить значение ячейки памяти FINAL с номером 400, если значение 0x0001 – все тесты выполнены успешно.

Комментарии к методике

* Для проверки используется три пары значений: 0000 & 0000, AAAA & 5555, 7FFF & 8765
* Данные значения показывают правильную работу программы с отрицательными, нулевыми и положительными числами.
* В ходе проверки флаги N и Z меняются с 0 на 1 и с 1 на 0 в двух разных случаях, что говорит о правильном выставлении флагов.
* Результат каждого теста записывается в соответствующую ячейку RES, значение 0x0001 означает успешное выполнение. Любое другое – результат выполнения синтезированной команды
* При успешном выполнении всех тестов значение FINAL станет 0x0001, иначе 0x0000

Сопоставление полученного и теоретического результата

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ячейка с результатом** | | **Первое число** | **Второе число** | **Теоретический результат** | **Полученный результат** |
| RES1 | 0x401 | 0000 | 0000 | 0000 (N=0, Z=1) | 0000 (N=0, Z=1) |
| RES2 | 0x402 | AAAA (N=1, Z=0) | 5555 | 0000 (N=0, Z=1) | 0000 (N=0, Z=1) |
| RES3 | 0x403 | 7FFF (N=0, Z=0) | 8765 | 0765 (N=0, Z=0) | 0765 (N=0, Z=0) |

Вывод

* В ходе выполнения лабораторной работы я изучил алгоритм синтеза собственной команды БЭВМ с помощью микропрограмм и методику проверки сделанной программы