Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и

радиоэлектроники»

Филиал

«Минский радиотехнический колледж»

# ОТЧЁТ

по лабораторной работе №11

по учебной дисциплине «Арифметико-логические основы вычислительной

техники»

«Разработка и отладка программ с использованием команд пересылки и

передачи управления»

Вариант 2

Выполнил: учащийся гр. 1К9391

Смаргун Е.А.

Проверил: Н.Е. Прибыльская

Минск 2023

# Лабораторная работа № 11

**Тема работы:** «Разработка и отладка программ с использованием команд

пересылки и передачи управления»

# Цель работы

Формирование практических навыков разработки и отладки программ с использованием команд пересылки и передачи управления.

# Задание

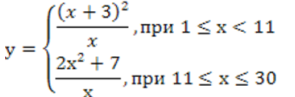
Разработать программу вычисления и вывода значения функции с использованием команд пересылки и передачи управления.

# Оснащение работы

Техническое задание, ПК, программная модель эмулятора учебной микроЭВМ.

# Порядок выполнения работы

Вариант 2



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Адрес | | Команда | | | | Примечание | |
| Мнемокод | | Код | |  | |
| 1 | | 2 | | 3 | | 4 | |
| 000 | | IN | | 010000 | | Ввод х | |
| 001 | | WR 50 | | 220050 | | Запись Х в 50 ячейку | |
| 002 | | SUB #31 | | 241031 | | Вычитание 31 | |
| 003 | | JNS 22 | | 140022 | | Переход на 22 строку, если положительно | |
| 004 | | RD 50 | | 210050 | | Чтение 50 ячейки | |
| 005 | | SUB #1 | | 241001 | | Вычитание 1 | |
| 006 | | JS 22 | | 130022 | | Переход на 22 строку, если отрицательно | |
| 007 | | RD 50 | | 210050 | | Чтение 50 ячейки | |
| 008 | | SUB #11 | | 241011 | | Вычитание 11 | |
| 09 | | JS 16 | | 130016 | | Переход на 16 строку, если отрицательно | |
| 010 | | RD 50 | | 210050 | | Чтение 50 ячейки | |
| 011 | | MUL 50 | | 250050 | | Умножение 50 ячейки на Х | |
| 012 | | MUL #2 | | 251002 | | Умножение ячейки на 2 | |
| 013 | | ADD #7 | | 231007 | | Прибавление к ячейке 7 | |
| 014 | | DIV 50 | | 260050 | | Деление 50 ячейки на Х | |
| 015 | | JMP 21 | | 100021 | | Безусловный переход на 21 строку | |
| 016 | | RD 50 | | 210050 | | Чтение 50 ячейки | |
| 017 | | ADD #3 | | 231003 | | Добавление 3 к ячейке | |
| 018 | | WR 51 | | 220051 | | Запись в 51 ячейку | |
| 019 | | MUL 51 | | 250051 | | Умножение на 51 ячейку | |
| 020 | | DIV 50 | | 260051 | | Деление ячейки на Х | |
| 021 | | OUT | | 020000 | | Вывод | |
| 022 | | HLT | | 090000 | | Стоп | |

|  |  |  |
| --- | --- | --- |
| PC | Acc | M(50) |
| 1 | 2 | 3 |
| 000 | 000012 |  |
| 001 |  | 000012 |
| 002 | 100019 |  |
| 003 |  |  |
| 004 | 000012 |  |
| 005 | 000011 |  |
| 006 |  |  |
| 007 | 000012 |  |
| 008 | 000001 |  |
| 009 |  |  |
| 010 | 000012 |  |
| 011 | 000144 |  |
| 012 | 000288 |  |
| 013 | 000295 |  |
| 014 | 000024 |  |
| 015 |  |  |
| 016 |  |  |
| 017 |  |  |
| 018 |  |  |
| 019 |  |  |
| 020 |  |  |
| 021 |  |  |
| 022 |  |  |

Таблица 2 – Содержание регистров

**Вывод:** сформировали практические навыки разработки и отладки программ с использованием команд пересылки и передачи управления.

# Ответы на контрольные вопросы

1. **Какие команды относятся к классу арифметических команд?**

*Арифметические команды* рассматривают коды операндов как числовые двоичные или двоично-десятичные коды. Эти команды могут быть разделены на пять основных групп:

* команды операций с *фиксированной запятой* (сложение, вычитание, умножение, деление);
* команды операций с *плавающей запятой* (сложение, вычитание, умножение, деление);
* команды очистки;
* команды инкремента и декремента;
* команда сравнения.

1. **Как используются команды условного, безусловного переходов?**

Безусловный переход — операнд указывает адрес перехода. Существует два способа указания этого адреса, соответственно различают прямой и косвенный переходы.

Условный переход — сначала вычисляется значение условного выражения, а затем, в зависимости от его результата, выполняются те или иные действия.

1. **Как работает механизм непосредственной, регистровой адресации?**

Непосредственная адресация — в команде содержится не адрес операнда, а непосредственно сам операнд. При непосредственной адресации не требуется обращения к памяти для выборки операнда и ячейки памяти для его хранения. Это способствует уменьшению времени выполнения программы и занимаемого ею объёма памяти. Непосредственная адресация удобна для хранения различного рода констант.

Регистровая адресация является частным случаем укороченной. Применяется, когда промежуточные результаты хранятся в одном из рабочих регистров центрального процессора. Поскольку регистров значительно меньше чем ячеек памяти, то небольшого адресного поля может хватить для адресации.