

# Спецификация Binary Digital Small Machine V2

## 1. Виртуальная машина

### 1.1. Общие сведения

Binary Digital Small Machine V2 (далее BDSM) – стековая виртуальная машина;

### 1.2. Память

Команды и данные в BDSM хранятся отдельно. Память команд представляет собой массив, содержащий до 65536 команд, общим размером до 65536 байт, индексация элементов начинается с 0. Память данных – стек, растущий вверх, размером до 65536 байт.

### 1.3. Процессор

Процессор BDSM имеет 2 регистра: 16-разрядный ip, 16-разрядный sp. Регистр ip содержит индекс текущей команды в массиве команд; регистр sp содержит указатель на вершину стека.

### 1.4. Использование

Для выполнения программы нужно запустить исполняемый файл виртуальной машины с параметрами командной строки:

bdsvm -p (--plain) <имя файла> – для выполнения из исходного кода

bdsvm -b (--byte) <имя файла> – для выполнения из байт-кода

## 2. Язык программирования

### 2.1. Общие сведения

BDSM имеет ассемблероподобный язык программирования, компилируемый в собственный байт-код. Программа для BDSM представляет собой текст, в котором каждой непустой строке, не являющейся комментарием, соответствует команда. Комментарий – строка, начинающаяся с любого символа кроме буквы, цифры или знаков '.' и '-'.

Команда := Инструкция [Модификатор\_типа]

Инструкция := Имя инструкции | Константа

Модификатор\_типа := 'c' | 'b' | 'w' | 'd' | 'f'

Константа := 32-разрядное целое число | число с плавающей точкой

Остаток строки игнорируется.

Значение по умолчанию для модификатора типа – ‘b’.

Типы:

‘c’ – символ, 8-бит, используется для ввода/вывода текста

‘b’, ‘w’, ‘d’ – байт, слово, двойное слово; 8-, 16-, 32-разрядные числа соответственно

‘f’ – 32-разрядное число с плавающей точкой

## 2.2.Перечень инструкций

### 2.2.1. Add

Снимает два числа заданного типа, складывает, результат помещает на стек.

### 2.2.2. Byte

Снимает число заданного типа со стека и преобразовывает к байту.

### 2.2.3. Div

Снимает два числа заданного типа, делит, результат помещает на стек.

### 2.2.4. Drop

Снимает число заданного типа со стека.

### 2.2.5. Dup

Удваивает вершину стека.

### 2.2.6. Dword

Снимает число заданного типа со стека и преобразовывает к двойному слову.

### 2.2.7. Float

Снимает число заданного типа со стека и преобразовывает к числу с плавающей точкой.

### 2.2.8. Goto

Снимает слово и число заданного типа со стека. Если число отлично от нуля, передает управление на строку с номером <слово>.

### 2.2.9. Mul

Снимает два числа заданного типа, перемножает, результат помещает на стек.

### 2.2.10. Nop

Пустая инструкция.

### 2.2.11. Read

Считывает число заданного типа с клавиатуры.

### 2.2.12. Rot

Снимает слово, снимает <слово>-1 чисел указанного типа, снимает 1 число указанного типа, помещает в стек сначала <слово>-1 чисел, а затем 1 число.

### 2.2.13. Sub

Снимает два числа заданного типа, вычитает, результат помещает на стек.

### 2.2.14. Word

Снимает число заданного типа со стека и преобразовывает к слову.

### 2.2.15. Write

Выводит число (или символ).

## 2.3. Байт-код

Первые два байта байт-кода – беззнаковое 16-битное целое, указывающее количество команд в коде. За ними идут непосредственно команды переменной длины. Первый байт – байт инструкции, формируется следующим образом:

$\text{Instruction\_byte} := \text{instruction\_code} \mid \text{magic} \mid \text{type\_code},$

где '|' – побитовое «или»

instruction\_code – из таблицы:

Instruction_code	Название команды
0x00	Rot
0x10	Dup
0x20	Drop
0x40	Read
0x50	Write
0x60	Byte
0x70	Word
0x80	Dword
0x90	Float
0xA0	Add
0xB0	Sub
0xC0	Mul
0xD0	Div
0xE0	Goto
0xF0	Nop

magic – 0x08

type\_code – 0x01, 0x02, 0x03, 0x04, 0x05 для 'c', 'b', 'w', 'd', 'f' соответственно.