

## Система команд Snow — BasePC Emulator

Команды эмулятора можно разделить на следующие категории:

1. Работа с памятью
2. Загрузка\сохранение дампов памяти
3. Работа с регистрами
4. Работа с устройствами В\В
5. Выполнение
6. Вспомогательные утилиты
7. Управление эмулятором

### Работа с памятью

**get mem <addr>**

Выводит на экран значение ячейки оперативной памяти с адресом addr

**get mem <startAddr> <endAddr>**

Выводит на экран значения ячеек оперативной памяти с адресами от startAddr по endAddr включительно

**get micro <addr>**

Выводит на экран значение ячейки памяти микрокоманд с адресом addr

**get micro <startAddr> <endAddr>**

Выводит на экран значения ячеек памяти микрокоманд с адресами от startAddr по endAddr включительно

**set mem <addr> <value>**

Меняет значение ячейки оперативной памяти с адресом addr на value

**set micro <addr> <value>**

Меняет значение ячейки памяти микрокоманд с адресом addr на value

**clear mem**

Меняет значения всех ячеек оперативной памяти на 0

**clear micro**

Меняет значения всех ячеек памяти микрокоманд на 0

**clear mem <startAddr> <endAddr>**

Меняет значения ячеек оперативной памяти с адресами от startAddr по endAddr включительно на 0

**clear micro <startAddr> <endAddr>**

Меняет значения ячеек памяти микрокоманд с адресами от startAddr по endAddr включительно на 0

### Загрузка\сохранение дампов памяти

**load <fileName>**

**load mem <fileName>**

загружает в оперативную память дампа из файла fileName

**load micro <fileName>**

загружает в память микрокоманд дампа из файла fileName

**save <fileName>**

**save mem <fileName>**

сохраняет в файл fileName содержимое всех ячеек оперативной памяти

**save micro <fileName>**

сохраняет в файл fileName содержимое всех ячеек памяти микрокоманд

**save <startAddr> <endAddr> <fileName>**

**save mem <startAddr> <endAddr> <fileName>**

сохраняет в файл fileName содержимое ячеек оперативной памяти с адресами со startAddr по endAddr (включительно)

**save micro <startAddr> <endAddr> <fileName>**

сохраняет в файл fileName содержимое ячеек памяти микрокоманд с адресами со startAddr по endAddr (включительно)

### Работа с регистрами

**get reg <shortName>**

выводит на экран значение регистра с коротким именем shortName

**get regs**

выводит на экран значения всех доступных регистров

**set reg <shortName> <value>**

Меняет значение регистра с коротким именем short на value

### Работа с устройствами В\В

**get io**

выводит значения регистров данных устройств ввода\вывода

**get io <deviceID>**

выводит значение регистра данных устройства ввода\вывода с индексом deviceID

**set io <deviceID> <value>**

устанавливает значение регистра данных и флаг готовности устройства ввода\вывода с индексом deviceID

**set ioready <deviceID>**

устанавливает флаг готовности устройства ввода\вывода с индексом deviceID

**unset ioready <deviceID>**

сбрасывает флаг готовности устройства ввода\вывода с индексом deviceID

### Выполнение

**step micro**

выполняет одну микрокоманду

**s**

**step**

**step command**

выполняет один цикл обработки команд

**s <count>**

**step <count>**

**step command <count>**

выполняет count циклов обработки команд

**r**

**run**

эквивалентно step command [runlimit], где [runlimit] — значение параметра runlimit (см set runlimit/get runlimit)

### Вспомогательные утилиты

**desc micro <cmd>**

выводит mnemonic описание микрокоманды с кодом cmd

**desc micro <startAddr> <endAddr>**

выводит mnemonic описания микрокоманд, находящихся в памяти микрокоманд

## Управление эмулятором

### **get runlimit**

выводит значение параметра runlimit

### **set runlimit <value>**

меняет значение параметра runlimit на value

### **exit**

выход из программы

## **Примеры использования эмулятора**

### **Program0.xml — арифметические операции, циклы**

Snow - BasePC Emulator (v0.6.0). Copyright (C) 2008 cleancode.ru  
This program comes with ABSOLUTELY NO WARRANTY. For details see  
LICENSE.txt

Loading microprogram.xml..[done]

>load program0.xml

program0.xml successfully loaded

>get mem 0x10 0x15

Memory:

----- -----
010   F200
----- -----
011   4005
----- -----
012   0007
----- -----
013   C011
----- -----
014   3006
----- -----
015   F000
----- -----

>get regs

Registers

----- -----
acc   0000
----- -----
addr   0000
----- -----
br    0000
----- -----
c     0000
----- -----
cbr    0000
----- -----
cmd    0000

cc	0000
data	0000
kbd	0000
mcmd	0000
mcmdcounter	0001
state	0002

>set reg cc 0x10

Registers:

cc	0010
----	------

>get mem 0x5 0x7

Memory:

005	0078
006	0000
007	FFCE

>run

Completed: 152 of 1000 steps

End of program reached

>get mem 0x5 0x7

Memory:

005	0078
006	1770
007	0000

>get regs

Registers

acc	1770
addr	0015

----- -----
br   0000
----- -----
c   0000
----- -----
cbr   0000
----- -----
cmd   F000
----- -----
cc   0016
----- -----
data   F000
----- -----
kbd   0000
----- -----
mcmd   0000
----- -----
mcmdcounter   0001
----- -----
state   0000
----- -----

>exit  
Good bye!

### **Program2.xml — асинхронный ввод\вывод**

Snow - BasePC Emulator (v0.6.0). Copyright (C) 2008 cleancode.ru  
This program comes with ABSOLUTELY NO WARRANTY. For details see LI

Loading microprogram.xml...[done]

>load program2.xml  
program2.xml successfully loaded

>get mem 0x1A 0x2D

Memory:

----- -----
01F   0000
----- -----
020   E102
----- -----
021   C020
----- -----
022   E202
----- -----
023   E002
----- -----
024   F600
----- -----
025   0005
----- -----
026   C024

	----		-----	
	027		E102	
	----		-----	
	028		C027	
	----		-----	
	029		E202	
	----		-----	
	02A		E002	
	----		-----	
	02B		3006	
	----		-----	
	02C		F000	
	----		-----	
	02D		0000	
	----		-----	

>set reg cc 0x20

Registers:

	----		-----	
	cc		0020	
	----		-----	

>get regs

Registers

	-----		-----	
	acc		0000	
	-----		-----	
	addr		0000	
	-----		-----	
	br		0000	
	-----		-----	
	c		0000	
	-----		-----	
	cbr		0000	
	-----		-----	
	cmd		0000	
	-----		-----	
	cc		0020	
	-----		-----	
	data		0000	
	-----		-----	
	kbd		0000	
	-----		-----	
	mcmd		0000	
	-----		-----	
	mcmdcounter		0001	
	-----		-----	
	state		0002	
	-----		-----	

```
>get io
IO Devices
|---|-----|
| 1 | False 0000 |
|---|-----|
| 2 | False 0000 |
|---|-----|
| 3 | False 0000 |
|---|-----|
```

```
>run
Completed: 1000 of 1000 steps
```

```
>get reg acc
Registers:
|-----|-----|
| acc | 0000 |
|-----|-----|
```

```
>set io 2 0xe2
IO Device #2
|---|-----|
| 2 | True 00E2 |
|---|-----|
```

```
>get io
IO Devices
|---|-----|
| 1 | False 0000 |
|---|-----|
| 2 | True 00E2 |
|---|-----|
| 3 | False 0000 |
|---|-----|
```

```
>run
Completed: 1000 of 1000 steps
```

```
>get io
IO Devices
|---|-----|
| 1 | False 0000 |
|---|-----|
| 2 | False 00E2 |
|---|-----|
| 3 | False 0000 |
|---|-----|
```

```
>get reg acc
```

Registers:

	-----		-----	
	acc		E200	
	-----		-----	

>get regs

Registers

	-----		-----	
	acc		E200	
	-----		-----	
	addr		0028	
	-----		-----	
	br		00A4	
	-----		-----	
	c		0000	
	-----		-----	
	cbr		0000	
	-----		-----	
	cmd		C027	
	-----		-----	
	cc		0027	
	-----		-----	
	data		C027	
	-----		-----	
	kbd		0000	
	-----		-----	
	mcmd		0000	
	-----		-----	
	mcmdcounter		0001	
	-----		-----	
	state		0024	
	-----		-----	

>set io 2 0xe4

IO Device #2

	---		-----	
	2		True 00E4	
	---		-----	

>run

Completed: 5 of 1000 steps

End of program reached

>get io

IO Devices

	---		-----	
	1		False 0000	
	---		-----	
	2		False 00E4	
	---		-----	



3	False 0000

>get reg acc

Registers:

acc	E2E4

>