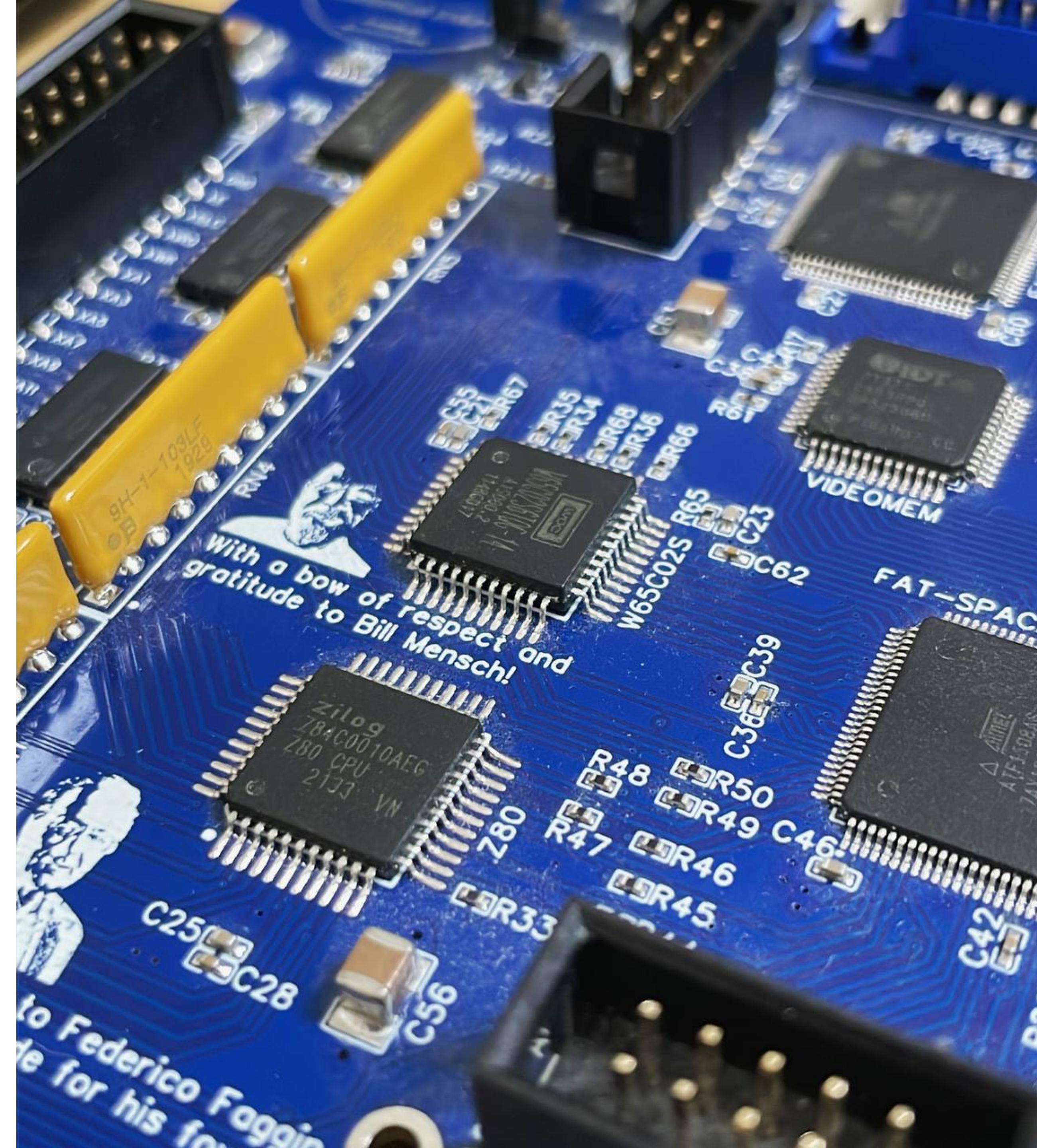


Cerberus W65C02S Basic

Quick start manual



About Cerberus W65C02S Basic

Cerberus W65C02S Basic(next just Basic or Cerberus Basic) is extended port of Microsoft Basic for 6502 CPUs made by Aleksandr Sharikhin.

It isn't based on Commodore Basic or Applesoft Basic, but have the same roots in it's core - so you'll find a lot things similar to many other implementations for 6502 CPU.

Cerberus Basic supports many features of Cerberus 2100(and 2080), have its own screen editor and can be used for education and even for development end user application.

This manual covers differences between any other basic dialect based on MS Basic and Cerberus 2100 W65C02S Basic.

Starting Cerberus Basic

Simplest way to start with Cerberus W65C02S Basic is just type “**basic6502**” on title screen. It will switch active CPU to 6502, loads “basic65.bin” binary and execute it.

Or you can do everything manually:

- Switch CPU to 6502 mode with “**6502**” command
 - Load Basic with “**load basic65.bin**”(or another file name if you renamed it) command
 - And execute it with “**run**” command



Screen Editor

Basic comes with simple but rich enough screen editor. Every command(or line) can be long up to screen lines.

Editor allows you execute immediate commands like **PRINT "Hello, world!"** or **VER**.

Or you can enter program for interpreter(if line starts with line number).

Unlike many old versions of Microsoft Basic, Cerberus Basic have case insensitive token parser - so you can type **Print** or **PRINT** or **PrInT** and you won't get parse errors.

If you made mistake - just on screen to line with it and retype character to right and press **Enter** key.

If mistake was made in long line(two screen lines long) - set cursor to first screen line of code line using right arrow scroll to place where fix is required, apply it and scroll right up to next line. When cursor will be on end of long line press **Enter**.

ESC key is used as **BREAK** key(you can abort program execution with it, stop scrolling etc).

The screenshot shows three windows of the Cerberus BASIC screen editor:

- Top Left Window:** Displays the system information and a successful execution of the command **PRINT "Hello, world!"**. The output shows "Hello, world!" printed on the screen.
- Top Right Window:** Shows the command **PRINT "Hello, world!"** being listed, resulting in the output "Hello, world!".
- Bottom Window:** Demonstrates a long line of code with a mistake ("Misteke") and its correction ("Mistake"). The command **list** is used to show the code, which includes:

```
READY
10 PRINT "Right line of code"
20 PRINT "Misteke was Made in long line
of code"
READY
```

The mistake "Misteke" is highlighted in red, and the corrected version "Mistake" is shown in green.

File Operations

For covering operations with files Cerberus Basic contains next commands:

- **FILES** - displays directory on screen. Scroll request can be aborted with **ESC** key.
- **LOAD “FILENAME”** - loads program from SD card to computer’s RAM
- **SAVE “FILENAME”** - saves program from computer’s RAM to SD card
- **KILL “FILENAME”** - deletes file from SD card
- **BLOAD “FILENAME”, <START ADDRESS>** - loads binary file directly from SD card to memory
- **BSAVE “FILENAME”, <START ADDRESS>, <SIZE>** - saves image of memory area to SD card

The image consists of six screenshots of the Cerberus Basic terminal window, each displaying a different command or its results. The terminal has a black background with white text and a small icon in the top-left corner.

- Screenshot 1:** Shows the output of the **FILES** command, listing various files with their addresses. It includes "ANIMAL.DAT", "DEFENCE.PRG", "FONTED.PRG", "MAZE.PRG", "ANIMAL.BBC", "BASIC65.BIN", "BAS11280.BIN", "CELLS802.BIN", "CELLS800.BIN", "CERBTICOM.IMG", "CHARDEFS.BIN", "F-INDEX.BBC", "F-RAND1.BBC", "F-RAND2.BBC", "F-RSER1.BBC", "F-RSER2.BBC", and "F-RSTD.BBC". Addresses range from 0500 to 0480.
- Screenshot 2:** Shows the output of the **LOAD FILE** command, indicating "FILE LOADED".
- Screenshot 3:** Shows the output of the **SAVE FILE** command, indicating "FILE SAVED".
- Screenshot 4:** Shows the output of the **KILL FILE** command, indicating "FILE KILLED".
- Screenshot 5:** Shows the output of the **BLOAD** command, listing three lines of code: **10 BLOAD "c1.scr",61440 : GOSUB 1000**, **20 BLOAD "c2.scr",61440 : GOSUB 1000**, and **30 BLOAD "c3.scr",61440 : GOSUB 1000**.
- Screenshot 6:** Shows the output of the **BSAVE** command, listing **bsave "test.fnt", 61440, 2048**.

System oriented extensions

There some system oriented extensions for Cerberus computers.

You can execute routine in machine codes using **SYS** command



Example of machine code application that starts from Cerberus Basic can be found in github repository: <https://github.com/nahirash/cerberus-w65c02s-basic/tree/main/prg-asm-template>

Version of your build of Basic can be found with **VER** command(including commit in repository).

You can reset your Cerberus computer using **RESET** command(your computer will be rebooted to BIOS).

```
FURTHI 12.1X
::GRAPH01.BBC" .REM 3E37
::GRAPH02.BBC" .REM 0D48
::GRAPH03.BBC" .REM 0EDA
::GRAPH04.BBC" .REM 12D9
::GRAPH05.BBC" .REM 1DE2
::GRAPH06.BBC" .REM 2111
::GRAPH07.BBC" .REM 28DC
::GRAUB" .REM 41A1
::LIFE.BBC" .REM 33D5
::LIFE.2.BBC" .REM 076D
::RESTIT.4TH" .REM 2BD8
::SERPENT.4TH" .REM 226B
::SOKOZ80.BIN" .REM 4F89
::SOUND.BBC" .REM 613F
::SPACER.4TH" .REM 4D92
::UDG.BBC" .REM 0B80
"TEST.FNT" .REM 0880

READY
ver
KITTY Kernel by Nahirash
Basic built from commit: 2628df9
Build date: 2024-03-24
READY
```

Console extensions

There some extensions that allows you use your Cerberus' console more comfortable.

You always can clean screen with **CLS** command.

If you'd like set position of your cursor you can use **LOCATE** command.

Current position of cursor can read with **POS(N)**, when **N** is equals zero X coordinate will be read else Y.

If you'll need read single key press you can use **KEY\$(N)** function, when N is equals zero it won't block execution else it will wait for key press. Letters will be returned always uppercased.

You can also produce some sound effects using **SOUND <length>, <freq>** command.

If it's required to slowdown execution of program - you can call **PAUSE N** command, where **N** - interrupts(frames) count to wait for.

```
READY
list
5 CLS
10 LOCATE 4,4 : PRINT "Here"
20 LOCATE 10,10 : PRINT "And here"
30 LOCATE 1,1 : PRINT "Bang!" "here"
READY
run
```

```
Bang!
READY
Here
And here
```

```
READY
list
5 CLS
10 LOCATE 4,4 : PRINT "Here"
20 LOCATE 10,10 : PRINT "And here"
30 LOCATE 1,1 : PRINT "Bang!" "here"
35 LOCATE 15,15
40 x=POS(0) : y = POS(1)
50 PRINT STR$(x);STR$(y)
READY
```

```
Bang!
Here
And here
15 15
READY
```

```
READY
10 a$=key$(1)
20 Print "Was pressed: ";a$
30 k$=key$(0)
40 if k$="" then goto 30
READY
```

```
READY
10 a$=key$(1)
20 Print "Was pressed: ";a$
30 k$=key$(0)
40 if k$="" then goto 30
run
Was pressed: X
```

```
READY
list
10 FOR i=1000 TO 50 STEP -40
20 SOUND 1, i
30 NEXT i
READY
run
READY
```

Low Resolution Graphics

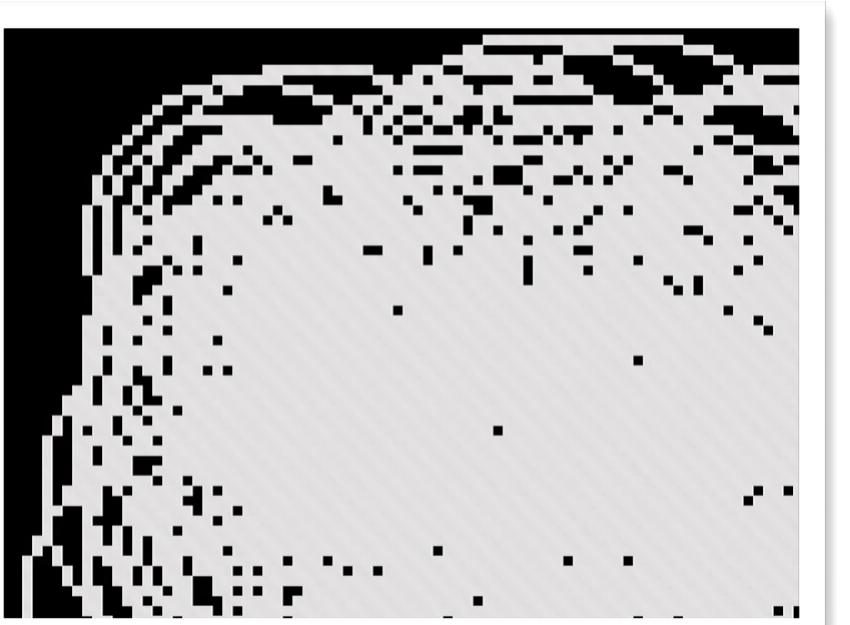
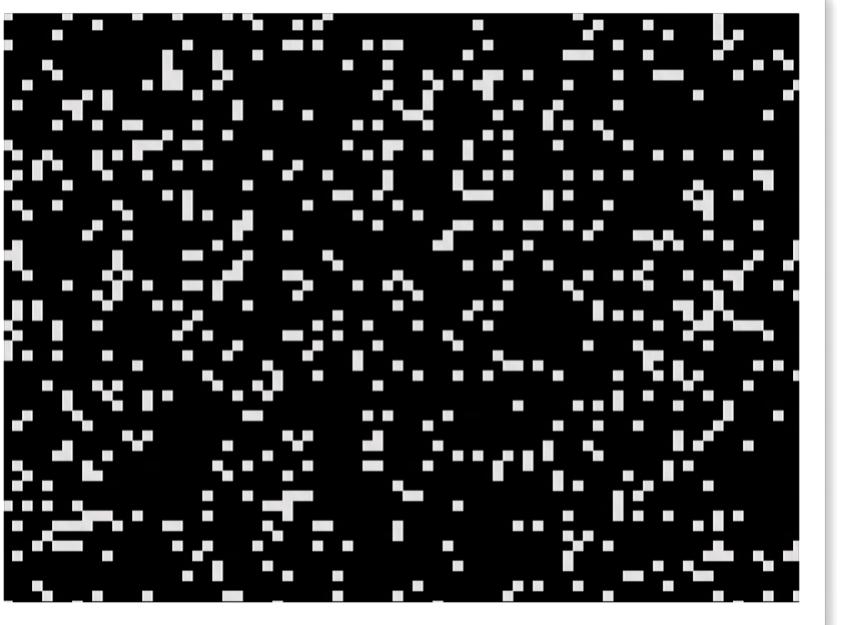
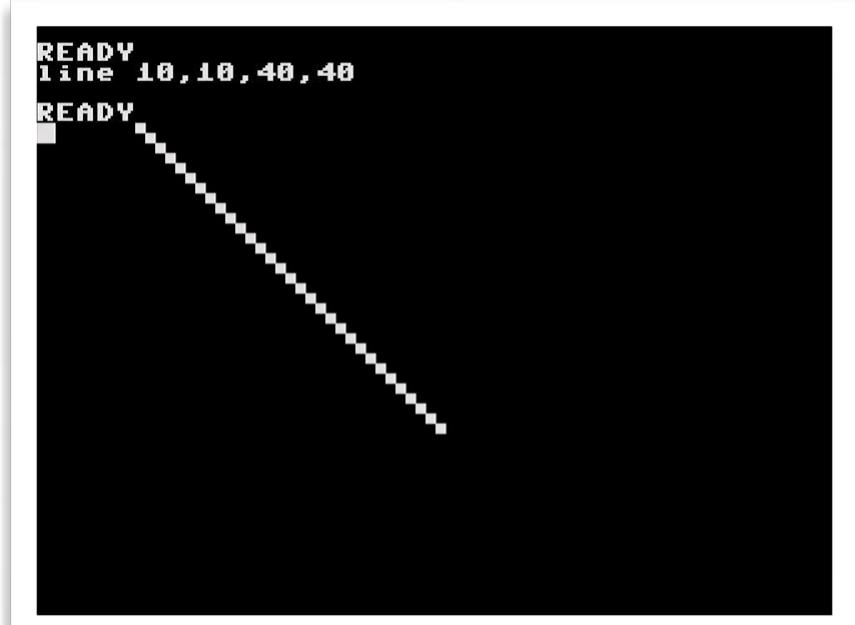
Cerberus 2100(and 2800) have text screen but Cerberus Basic includes support for basic graphic primitives that drawn over text screen as Low Resolution Graphics.

LRG commands are including:

- **PSET X,Y** - sets point on screen
- **CIRCLE X, Y, R** - draws circle with center in X, Y and radius R
- **LINE X1, Y1, X2, Y2** - draws line that connects two points

```
READY
list
10 CLS
20 PSET RND(1)*80, RND(1)*60
30 GOTO 20
READY
```

```
READY
list
10 CLS
20 CIRCLE 20+RND(1)*60, 20+RND(1)*40, RND(1)*20
30 GOTO 20
READY
```



Tile graphics

Cerberus's video circuit is text oriented but important feature of it that you can redefine characters and use them as tile graphics.

You can define up to 47 custom tiles from Cerberus Basic and draw it where you wish.

For defining your own tile you can use **TILEDEF N, B0,B1,B2,B3,B4,B5,B6,B7** command, it accepts tile number and 8 bytes that defines bitmap of tile.

For drawing tile on screen you can use **TILE X,Y,N** command that draws tile with number **N** on position **X, Y**. Zero tile is always empty and cannot be redefined.

As real world example, you can check “**DEFENCE.PRG**” program - it’s simple target-bullet game.

Cerberus 2100 supports color output and **first 24 tiles are colored** in next order **1 2 3 4 5 6 7**.

Rest of tiles are white.

```
READY
list
1 TILEDEF 1, 80, 118, 117, 119, 119, 255
2 102 102
3 TILEDDEF 2,0,48,152,206,255,6,12,0
4 TILEDEF 3,0,32,32,32,32,32,32,86,0
5 CLS : sc=0 : hs = 0
6 REM Let's introduce MAIN game vars
7 px=20 : ex=0:ey=RND(1)*23
8 bx=0 : by=0 : REM bullet coords
9 REM here starts game loop
10 TILE px,29,1 : REM Our truck
11 TILE ex,ey,2 : REM Enemy Plane
12 IF by>0 THEN TILE bx,by,3 : REM bullet
tile will be drawn only if by>0
13 PAUSE 2 : a$=KEY$(0)
14 IF a$=CHR$(8) OR a$=CHR$(21) THEN GOSUB 100
15 IF a$="" THEN GOSUB 200
16 TILE bx,by,0 : REM hide bullet
17 TILE ex,ey,0 : REM hide plane
18 ex=ex+.?
19 IF ex>39 THEN ex=0:ey=ey+1
20 LOCATE 5,0 : PRINT "Score ";STR$(sc);
" Hi-Score: ";STR$(hs)
21 IF by>0 THEN GOSUB 300
22 BREAK
READY
```

```
READY
list
10 FOR i=0 TO 39
20 TILE i,11,i
30 NEXT i
40 LOCATE 8,12
50 PRINT "Color tiles"
READY
run
Color tiles
READY
```



Using Cerberus Basic for running programs written in assembly

Cerberus Basic can be used as simple operating system for running compiled 6502 software.

Template of assembly application can be found here: <https://github.com/nihirash/cerberus-w65c02s-basic/tree/main/prg-asm-template>

It opens possibility for creating first class experience of early 80s computers.

Machine code routines can be executed with **SYS** procedure.

Useful links

- Main repository for Cerberus Basic(including useful tools for Basic and Cerberus 2100): <https://github.com/nihirash/cerberus-w65c02s-basic>
- Main Cerberus 2100 repository(contains “Hardware Manual” and actual SD card content): <https://github.com/TheByteAttic/CERBERUS2100>
- BBC Basic for Z80 part of Cerberus: <https://github.com/breakintoprogram/cerberus>