

G. A. M.

Graphics Access Manager for the QL computers

1. Revision history

1987 version 2.3 by Thomas Much

2014 release 3.1 by Tomas Kral

Sources of initial version 2.3 were kindly provided by the author from the pages <http://www.snailshell.de/QL/>.

The code was updated to compile with TURBO, serial mouse support along with some extra functionality were added.

2. Foreword

The GAM is a simple graphical front end for the QL computers, it builds upon QDOS system's core functionality while providing easy to use pointer driven environment. It is intended for all unexpanded genuine pre-SMS QLs, it has been tested on JS ROM with 128K of RAM. It is not perfect and is not meant a replacement to QRAM QPAC series. It takes a different approach of very much simplified pointer and window management, reducing code complexity and fitting in the standard QL memory.

It is all programmed in standard Super BASIC with no specific extension.

3. Installation

The installation ZIP package contains following components:

GAM_EXE	- core executable
MOUSE_BIN	- serial mouse driver SER1
DRAM_BIN	- dynamic RAM disk
FTYPE_BIN	- file type utilities
BOOT	- boot file
GAM_RUN	- loader

Default installation media is MDV1_, you are expected to unzip and install these files onto preformatted microdrive and reboot from there. MDV2_ is data drive for your own use, GAM does not require MDV2_ to run. GAM offers dynamic ramdisk functionality to the user, it also makes use of ramdisk for temporary files.

Most unzip programs will not run on standard unexpanded QLs due to lack of memory. If you transfer unzipped files to the QL, you will loose the executable header of GAM_EXE,

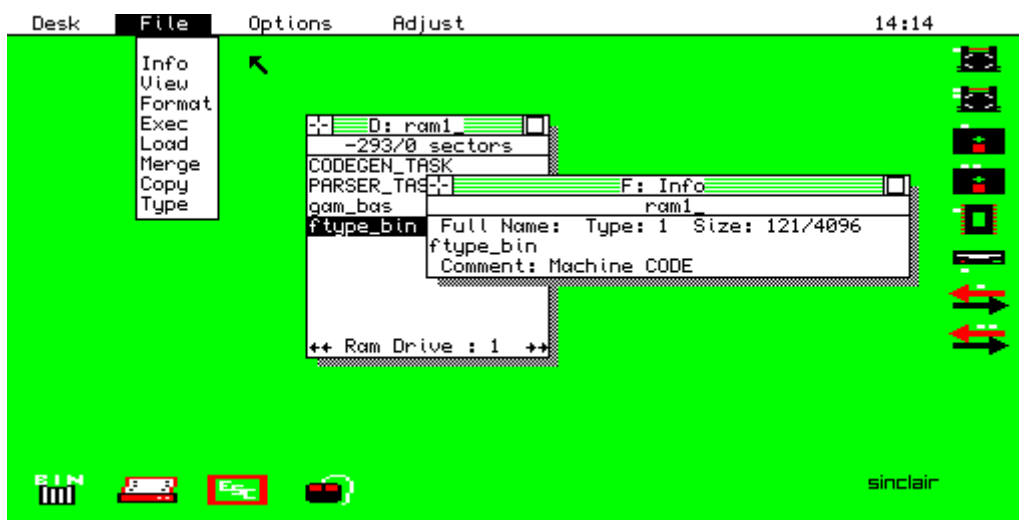
that can easily reconstruct as it needs some 12288 bytes of stack dataspace, you can type:

```
base=RESPR(47530) [enter]
LBYTES mdv1_GAM_EXE, base [enter]
DELETE mdv1_GAM_EXE [enter]
SEXEC mdv1_GAM_EXE, base, 12288 [enter]
```

Other files of the installation package are not executables, so simple file transfer should work for you.

4. First steps

After you boot to GAM from MDV1_ the little green desktop should appear similar to the screen capture below.



You can move pointer by mouse also by arrow keys. If you QUIT or exit from GAM by error, you can come back by typing on the command line

```
RUN [enter]
```

or

```
LRUN GAM_RUN [enter]
```

5. Standard ports

SER1 – reserved for serial mouse

SER2 – reserved for printer

both ports can also be used for file transfer

6. Windows & mouse pointers

GAM uses four pointer types, their meaning changes in the context with windows and menus you can point and click on, and lets you perform different user actions



– general pointer expects desktop and menu actions



– window pointer expects: close, move and bring-to-top actions




– file pointer selection action



– window command execution pointer

Pointer moves with mouse also by keyboard arrow keys, left mouse button will perform user action, similarly as SPACE bar on the keyboard.

Holding CTRL-Z together will produce HARDCOPY screen capture on the printer attached to SER2.

bring to TOP action |
CLOSE action -  - MOVE action

7. File types and dataspace

GAM lets you set file type and dataspace. This is one of its added extra functionality.

Some common QL file types are:

- 0 - a data file or SuperBASIC program
- 1 - an executable file
- 2 - SROFF relocatable object file
- 4 - font file in 'The Painter'
- 5 - pattern file in The Painter
- 6 - 4 colour mode compressed picture in The Painter
- 11 - 8 colour mode compressed picture in The Painter

8. Calculator

GAM offers a simple stack calculator with reverse Polish notation, up to 10 operands in a single expression.

As an example, consider following formula,

$$-5 * (12 + 8)$$

in reverse notation this enters as

0 5- 12 8+*

Supported operators and functions are:

pi 3.14... constant

sin, cos, tan, log, ln, e

+ - * /

\ reverse value 1 / x

square root

! factorial

| integer division

% integer modulus

^ x power y