

First steps to make a small variation Gigatron rom with your small games/demos:

1. Download and unzip Gigatron-rom: <https://github.com/kervinck/gigatron-rom>

2. Edit Makefile and copy an existing ROM chapter into a new one for you.
Example ROMv5a to ROMv5b (Change your apps and path to them!):

```
# my own version of v5a with more games/demos
ROMv5b.rom: Core/* Apps/*/* Makefile interface.json
python3 Core/ROMv5b.asm.py\
    packedPictures=Apps/Pictures/packedPictures.rgb\
    Snake=Apps/Snake/Snake_v3.gcl\
    SYS_Racer_v1.py\
    zippedRacerHorizon=Apps/Racer/Horizon-256x16.rgb\
    Racer=Apps/Racer/Racer_v2.gcl\
    Mandelbrot=Apps/Mandelbrot/Mandelbrot_v1.gcl\
    SYS_Loader_v4.py\
    Pictures=Apps/Pictures/Pictures_v3.gcl\
    Loader=Apps/Loader/Loader_v4.gcl\
    Credits=Apps/Credits/Credits_v3.gcl\
    Tetronis=Apps/Tetronis/Tetronis_v2.gt1\
    Bricks=Apps/Bricks/Bricks_v2.gt1\
    Puzzles=Apps/v5x/puzzles.gt1\
    Parsec=Apps/v5x/parsec.gt1\
    Arcibo=Apps/v5x/arcibo.gt1\
    Mosaic=Apps/Mosaic6502/Mosaic6502.gt1\
    Egg=Apps/Horizon/Horizon_at67_v1.gt1\
    Boot=Apps/CardBoot/CardBoot_v1.gcl\
    Main=Apps/MainMenu/MainMenu_v5b.gcl\
    Reset=Core/Reset_v5.gcl
```

```
burnv5b: ROMv5b.rom
    minipro -p 'AT27C1024 @DIP40' -w "$<" -y -s
```

3. Copy Core/ROMv5a.asm.px to ROMv5b.asm.py
(No changes required in copied file, but new rom file name seems to depend on this)

4. Copy Apps/MainMenu_v5.gcl to Apps/MainMenu_v5b.gcl

5. Edit Apps/MainMenu_v5b.gcl to reflect your new apps:

First place:

```
42 {Gray} Color= PrintArrow!
```

```
[def
    ##\Snake
    ##\Racer
```

```

##\Mandelbrot
##\Pictures
##\Credits
##\Loader

##\Tetronis
##\Bricks
##\Puzzles
##\Parsec
##\Arecibo
##\Mosaic
]

```

Second place:

] Newline=

```

[def
  #9 `Snake #9 `` ` #9 `Tetronis    #10
  #9 `Racer #9 `` ` #9 `Bricks      #10
  #9 `Mandelbrot #9 `Puzzles        #10
  #9 `Pictures `` ` #9 `Parsec      #10
  #9 `Credits #9 ` #9 `Arecibo      #10
  #9 `Loader #9 `` ` #9 `Mosaic     #10 #0
] MainMenu=

```

Third place:

Comment out special handling for MSBASIC/Apple1:

```

{ Load and start the selected program }
\SYS_Exec_88 _sysFn=                {ROM loader}
Program _sysArgs0=                  {ROM image}
\userCode _vLR=                      {Start of execution address}
{ MenuItem 10- [if<>0 1-] }         {XXX MSBASIC and Apple1 need
workaround}
[if=0
  \channelMask_v4, $f8&              {Clear channelMask before
loading}
  \channelMask_v4.]
88!!                                {Load from ROM and execute}

```

6. Run make ROMv5b.ROM

```

Create file ROMv5b.lst
Create file ROMv5b.rom
ROM bytes 131072 words 65536
Words used 49920 unused 15616
Assembly OK

```