



Commodore 64  
Programming  
Languages:  
**PIROMAL**  
ShadowM/ECCC 2014

## speaker bio

- long-time collector of Commodore 64 compilers/interpreters
- a more-or-less neglected area for Commodore 64 enthusiasts
- hope to do a series of talks on some of the more interesting programming languages available

## common features

does the language support...

- multiple drives?
- devices other than 8 and 9?
- writing large, modular programs?
- assembly language routines?
- standalone programs?

and, very important...

- does it have a good editor?

## the PROMAL saga

- originally became familiar with it from Q-Link downloads, searched for years to find more information
- in late 2013, Jeff (Vintage Volts) sent me a complete set
- shortly afterward, an even more complete set appeared on eBay:  
all disks and hundreds of pages of documentation plus brochures
- all that's missing is the source...

## PROMAL's disk speeder

- "DYNODISK", enabled by default
- not compatible with more than one drive, 1571, JiffyDOS, printers, cartridges...
- also applies to standalone runtime
- I strongly recommend that you permanently disable it:

```
LOAD PROMAL, 8
```

```
POKE 3554, 128
```

```
SAVE "C0:PROMAL", 8
```

## two-drive support

- PROMAL issues "IO" commands to both 8 and 9 on startup, so if there's no disk in 9, this will bang the drive head
- if you have a drive on device 9, make sure there's a disk in it...
- or fix by passing 0 to SETNAM:

```
LOAD PROMAL, 8
```

```
POKE 2596, 0
```

```
SAVE "00:PROMAL", 8
```



# the PROMAL environment

- has its own shell with command recall, batch scripting
- standalone runtime available
- source compiles to p-code
- overlays, modules, chaining
- adjustable memory map
- removable floating-point support
- good assembly support (JSR keyword)
- formatted listings with addresses
- memory map and hex dump commands

## the PROMAL loader

- programs relocatable, up to six can be in memory at once
- UNLOAD command to remove individual programs or all programs
- assembly code can be turned into a relocatable module with RELOCATE
- a module referenced from the main program must be loaded before it (compile with OWN, use stub loader)



# PROMAL editor, compiler

- everything in ASCII (not PETSCII)
- full-screen editor with  
4-way scrolling, indent/outdent,  
cut/paste, search/replace
- full-featured runtime API
- extensive libraries including  
graphics and relative file support
- conditional compilation
- XREF utility (appends to listing)

# PROMAL language features

- mandatory indentation (like Python)
- pointer arithmetic (like C)
- data types: BYTE, INT, WORD, REAL
- any non-zero value considered true
- procedures and functions with arguments and local variables
- labels can be up to 31 characters
- variables declared; not initialized
- ESCAPE/REFUGE
- recursion supported

# disk handling

- assumes two drives (0: and 1:), but that's device no, not unit no.
- "channel" in docs means logical file number, not secondary address
- command channels kept open on logical file numbers 15 and 14, file APIs use use 2 and above, so use 8 and above when doing file handling in machine language

## summary

- visit my site ([www.lyonlabs.org](http://www.lyonlabs.org)) to get disk images and documentation:

`/commodore/onrequest/PROMAL/index.html`

- there's also a cheat sheet there with commonly used commands &c.

- I can demo PROMAL and other programming languages at my table

# DEMO

(slides and sample code  
will be on my web site)