1. Project file structure description

project.tal is all the code of this project. It is a rewrite of the Fortran source code of the funkttal compiler to uxntal.

The .tal files in the unit\_test folder are the code for individual subroutines or important modules in the compiler source code.

The .rom files in the .rom\_files folder are the compiled Uxn binaries. When a .tal file is processed by Uxn's assembler, it is converted into a .rom file. This .rom file contains binary code that can be directly executed by the Uxn virtual machine.

Uxn (https://wiki.xiiivv.com/site/uxn.html) is an 8-bit virtual machine for resource constrained systems. It is programmed using a stack-based assembly language called Uxntal (https://wiki.xiiivv.com/site/uxntal.html).

1. Installation Tutorial

uxn virtual machine download website: https://sr.ht/~rabbits/uxn/, there are detailed installation tutorials on the website.

https://github.com/hundredrabbits/awesome-uxn#applications is also a relatively complete website for downloading uxn virtual machines.

**Linux/OS X**

To build the Uxn emulator, you must install SDL2 for your distro. If you are using a package manager:

sudo pacman -Sy sdl2 # Arch

sudo apt install libsdl2-dev # Ubuntu

sudo xbps-install SDL2-devel # Void Linux

brew install sdl2 # OS X

Build the assembler and emulator by running the build.sh script. The assembler(uxnasm) and emulator(uxnemu) are created in the ./bin folder.

./build.sh

--debug # Add debug flags to compiler

--format # Format source code

--install # Copy to ~/bin

If you wish to build the emulator without graphics mode:

cc src/devices/datetime.c src/devices/system.c src/devices/console.c src/devices/file.c src/uxn.c -DNDEBUG -Os -g0 -s src/uxncli.c -o bin/uxncli

**Windows**

Uxn can be built on Windows with MSYS2. Install by downloading from their website or with Chocolatey with choco install msys2. In the MSYS shell, type:

pacman -S git mingw-w64-x86\_64-gcc mingw64/mingw-w64-x86\_64-SDL2

export PATH="${PATH}:/mingw64/bin"

git clone https://git.sr.ht/~rabbits/uxn

cd uxn

./build.sh

If you'd like to work with the Console device in uxnemu.exe, run ./build.sh --console instead: this will bring up an extra window for console I/O unless you run uxnemu.exe in Command Prompt or PowerShell.

#Getting Started

#Emulator

To launch a .rom in the emulator, point the emulator to the target rom file:

bin/uxnemu bin/piano.rom

You can also use the emulator without graphics by using uxncli. You can find additional roms here, you can find prebuilt rom files here.

#Assembler

The following command will create an Uxn-compatible rom from an uxntal file. Point the assembler to a .tal file, followed by and the rom name:

bin/uxnasm projects/examples/demos/life.tal bin/life.rom

My personal method of compiling and running uxntal:

After installing msys2 on Windows, open the terminal, enter the file location where the uxntal program is located, enter the following command to complete the compilation and run it at the same time:

./uxnasm project.tal project.rom && ./uxnemu project.rom