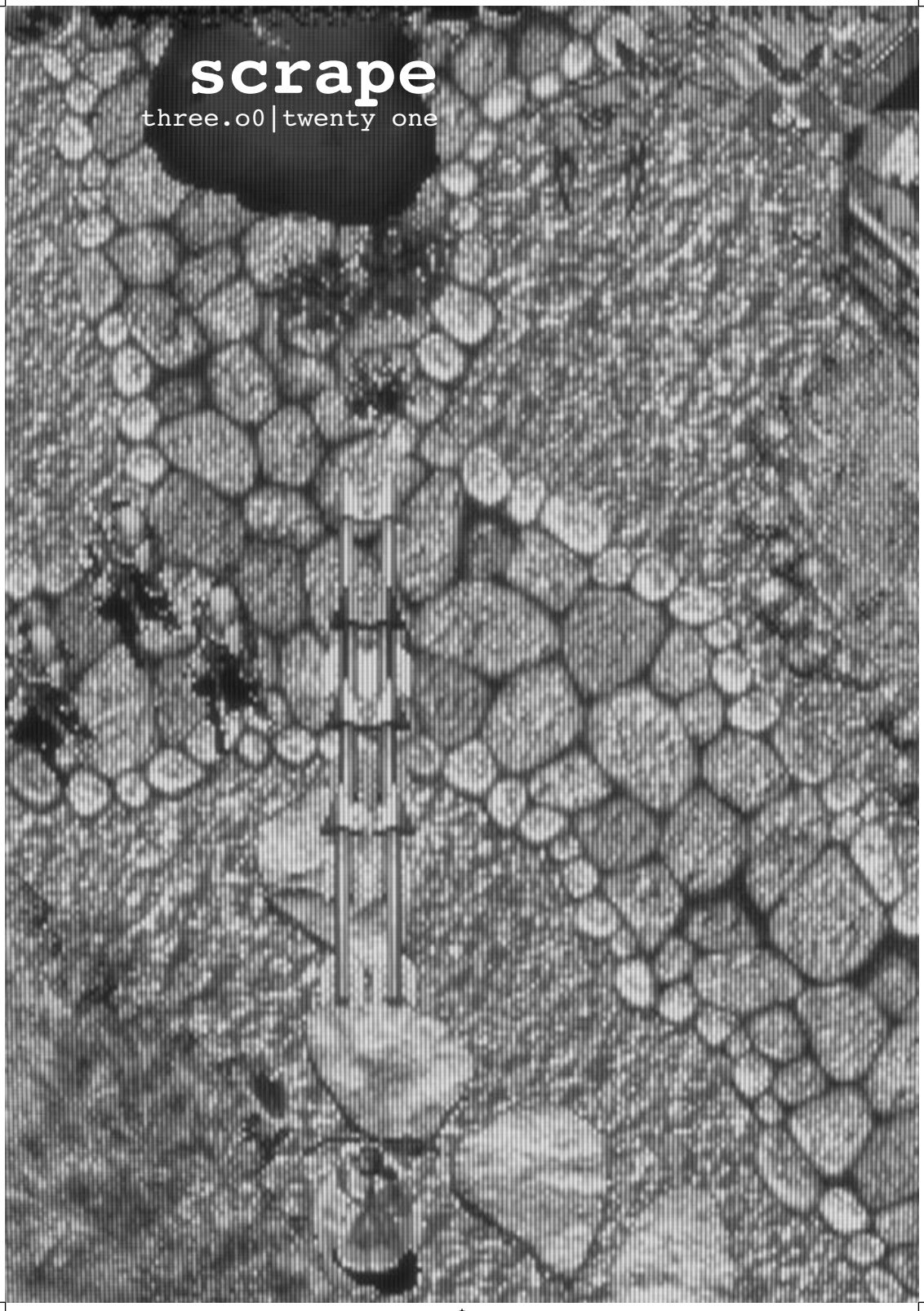

scrape
three.o0|twenty one

1/edit/three.o0/mame_all_month

this is going to be a mame issue, i've been pissing about with it alot this month just trying different things playing with old systems and the like. I hadn't actually played a single arcade title on it this month until i was doing the cover/bgfx article for this issue weird eh.

mame is one of those projects i've been following for a good 20y+ i'd expect, my son is almost 17 now and i was into mame long before he was even a thought. that is astounding to think about. i should've been more involved with it by now.

so yes as i started to try and put this issue together there really wasn't anything else of note i'd done this month.

i had some plans around mame for march. ethernet, serial stuff, wanted to get something running on an apple1 since i'd probably never see one, I wanted to get an x86 nextstep install up, os/2, windows 95, windows 2000, etc just a bunch of random work with the ct486/at486 driver to see how far it'd go. I didn't get to it all.

all of these articles are taken from my experience(s) so if i'm wrong or something makes you sad. let me know. the writing feels off this month, but mame is an odd subject to try and write about, the project means alot to me.

ironically as i looked around, alot of the stuff i tried was already done years ago by the mess guys but that wasn't really the point.

i just wanted to try it

- ed

this issue

want to write something?

contact us with your idea.
: tech, gaming, retro stuff
: anything else (discuss)
: we are after cover art.
: submissions welcomed

@@scrapezine_at_mameau.com@@

cover: guwange (mame)
author: cave
date: 1999

2/re/19??/mame/ten_pins_one_apple

welcome to the alley, we are going bowling on an apple one.

this is a fairly decent game for the time. written in basic, you will first need to load up the basic program into the machine memory then load the program into the correct memory location for basic to run it.

after you have loaded the game successfully you will be presented with the welcome screen.
did you bring friends? sure, it supports multiple players.

```
>RUN
                BOWL
CREATIVE COMPUTING MORRISTION, NJ


WELCOME TO THE ALLEY
BRING YOUR FRIENDS
OK LET'S FIRST GET ACQUAINTED

THE INSTRUCITONS (Y/N)?Y

THE GAME OF BOWLING TAKES MIND AND SKILL
. DURING THE GAME
THE COMPUTER WILL KEEP SCORE. YOU MAY CO
MPETE WITH
OTHER PLAYERS UP TO FOUR. YOU WILL BE PL
AYING TEN FRAMES
ON THE PIN DIAGRAM 'O' MEANS THE PIN IS
DOWN...'+' MEANS THE
PIN IS STANDING. AFTER THE GAME THE COMP
UTER WILL SHOW YOUR
SCORES.
FIRST OF ALL...HOW MANY ARE PLAYING?@
```

but if not go for a roll yourself.

the game provides a decent set of instructions which you can review by answering the appropriate question when prompted. it will let you know that 0 means the pin is down and + means the pin is standing.

standard bowling rules apply. there is nothing overly complicated or different here, type 'roll' to roll your ball and the screen will print out nice ascii graphics showing the state of the pins. this is quite a nice addition as it would have been easier to just show off your score instead.

```
FIRST OF ALL...HOW MANY ARE PLAYING?1

VERY GOOD...
TYPE ROLL TO GET THE BALL GOING.
?ROLL
PLAYER: 1FRAME: 1 BALL: 1

 0 0 0 0
  0 0 +
    + 0
      0

TYPE ROLL TO GET THE BALL GOING.
?@
```

an interesting side effect of failing to pickup the spare is the game will print 'error!!' in place of 'missed' whereas it will happily report 'spare' and 'strike' as applicable so i have to wonder if that was an oversight or the game logic is wrong.

i suppose this is something we could look into for an article at some point and perhaps fix the code, that might be a cool thing to do.

you will continue to type roll until your 10 frames are complete at which time you will be presented with another nice ascii scoreboard showing the results of the match.

```
FRAMES
  1  2  3  4  5  6  7  8  9  10
-------------------------------
  8  6  9  7  8  7  7  7  7  7
  1  3  1  3  1  3  2  1  2  3

DO YOU WANT ANOTHER GAME?@
```

all in all this is a fun game for the time and hardware while the machine is more challenging that you might be used to in regards to getting software running it is worth some time

- sairuk

3/poc/~/os/install

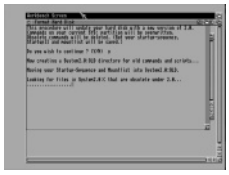
i've played around with a number of os installations in mame on the ct486 driver, some successful some not but it was all interesting so lets take a look at what where we got to.

there is a forum post starting back in the mess days on bannister.org where others have already achieved alot with the driver but this was my run at it.

starting out with dos 6.22 was relatively uneventful after i managed to work my way though chdman and generating flopping images. grabbing the standard 6.22 disks from winworld was enough, attach the chd as hard1 and boot the machine remembering to configure you chs for the hdd geom in the bios. attach the dos setup floppy disk1 to a: drive with the inbuilt ui filemager and run through the standard setup.

os/3 warp 3 was next and again starting with a clean chd, the red disc cd (blue didn't work) step through the installer with basic vga support, this completed with ui and sound all up and running. i haven't done much more since the installation though and need to go back to it.

somewhere in this mix i jumped across to install workbench into the a600 with a chd attached. an odd choice given i never owned one. but thats what it is all about,



its not just the games. there was a guide on bannister by the duke (iirc) and i just followed along to have a go at it. it worked as documented in the post and i was up and running shortly after.

then i moved onto nextstep x86, this was no dice and not for lack of trying, it just couldn't mount the iso. I am not sure why but after trying everything i thought of, i put this one away.

and the last of the os installs i got to this month was windows 95C, yes thats the OEM release if you didn't recognise it. pretty cool, the only real issue is it was too dumb to partition and format the hdd and the installer wouldn't even run until i'd done so in dos.

still pretty cool no? i hope this gives you some idea of the capabilties of mame these days. we may look at serial and network device support in another article on day.

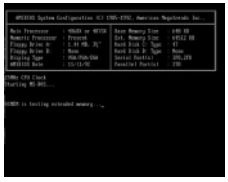
- sairuk

the mame ct486 driver is relatively well established, at least enough to do stupid stuff like install dos and remember how much time you used to burn on memory management and getting simple stuff like software onto the machine.

one thing you might not remember offhand is that mame used to ship a dos version.

by now you can probably guess where this is going. stupid stuff for stupidities sake.

while playing around with dos in the mame ct486 driver it seems fairly reasonable to try and run some software. it also seemed rather silly not to run dos mame on dos running in mame on an emulated 486.



with dos 6.22 loaded onto a chd and booting successfully i grabbed mame0.1 pulled across a couple of romsets and renamed as appropriate for the build.

from there we needed to get these files onto the machine. the easiest approach was to build a floppy image (1.44m not 2.88m) and drop the files across.

the path of least resistence to make sure the dos version would run was to use dosbox on my main machine to test for dependancies and unsuprisingly it did. we just needed cwsdpmi before the executable would run.

having obtained cwsdpmi to do the whole real mode/protected mode dos thing we could launch mame 0.1 in dosbox, time to move into mame.

load the disk through the file manager and run mame from a: which didn't work. ok interesting, so we'll just grab another version, in this run i grabbed 0.7b for a reason that now escapes me but going the process actually ran pacman succesfully.

one thing i did run into was running mame of the floppy drive corrupted the file system on the floppy on the ct486 machine when the sb16 driver were loaded (at486 was fine) so copy it to the hdd to run it instead.

thats nuts, mame can run itself and i suppose eventually mame will be able to run a mame that can run a mame.

again as with all things, this was already done years ago in mess but was still a case of why not.

- sairuk

::History

one of my favourite projects over the last 10t years has been mame, the multiple arcade emulator. there is a variety of ways this could be covered, but even if i dedicated an entire issue to MAME (hey i did -ed) i couldn't barely cover of what mame encompasses.

a little history as it seems appropriate. nicola salmoria started the ball rolling by emulating pac man, this effort was expanded into the multipac emulator as more variants were added. multipac grew into what became mame 0.1 released in early '97.

mame continues to grow, in its infancy and through its teens it was developed away from the public eye. this lack of transperancy may have (in some part) contributed to its percieved borg like nature.

interrestinly public opinion has shifted over the years we've seen mame equated to the borg by the few; ever assimulating other projects into its core unfair as it may be.

i'd guess that the architecture of mame has be refactored over the years such that many systems can be built relatively easily and with that onboarding developers from other projects may well have been easier.

mame established its core value early on, focused on architecture and documentation of at risk systems. the ability to play games is a by-product of the verification process.

::Derivatives

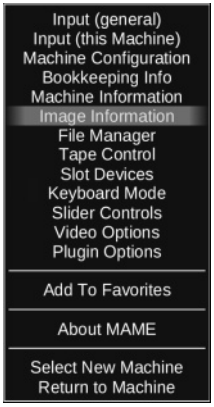
with mame focused solely on the 'a' for arcade and the project being source available (not opensource), the opportunity arose for the emulation of additional non-arcade systems. this lead to a number of derivative builds and sister projects that were published outside of the core development team covering computers, consoles, calculators, fruits, pinball, homebrew, hacks.

eventually in a major shift, these derivative works would be merged into the core project. perhaps recognition that all the projects would benefit from a unified code base and the developers across these projects could contribute to

more systems within the same tree.

::Direction
the release co-ordinator roles in the mame team shifted over the years and with it the direction of the project appeared to shift, even with the dynamic nature of the project the focus on accuracy and documentation has remained.

::Availablity
the mame code is now available on github, no longer locked behind a private svn (or whatever). with the core of the project gpl licensed but be aware the licensing varies per source file. we no longer have to wait for a release, in the past releases used to be sporatic but now the project has moved to the a fixed release cycle. i think this speaks more to the maturing of the project than is generally mentioned.



each release still comes backed with a whatsnew.txt including all the changes and mametesters bugs fixed.

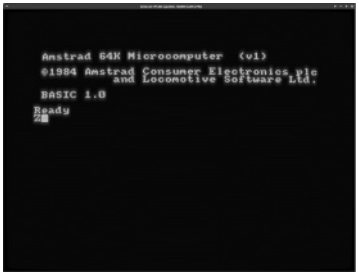
back in the day you'd start with whatsnew when a release was published and then maybe dig through the source archive to look at changes. with the code available on github you can track the commits and keep up to date between releases or pull down a commit and try it out at any point in the development cycle or easily see the code diff per commit if you pull down the source tree.

::Documentation
you can read through the source code for mame and learn about arcades, computers, consoles, fruits and almost everything in between you can read the history of the driver in the commit or the comments in the source files about the discoveries over time and yes ultimately you can boot a system and use it.

::Entry
i found mame because someone mentioned a "T&C Surf designs" in a chat once, which I'd never heard of and that lead me into emulation, unfortunately alot of the public focus for the project is the 'free games' aspect and this was how it presented in the chat. for me tho it was never the free games aspect, i'd always thought of it as 'these games' instead. where emulation opened up a world of games and systems i would have never known to experience.

::Education
this is key, mame can be used to provide experiences that are becoming rarer with each passing year.

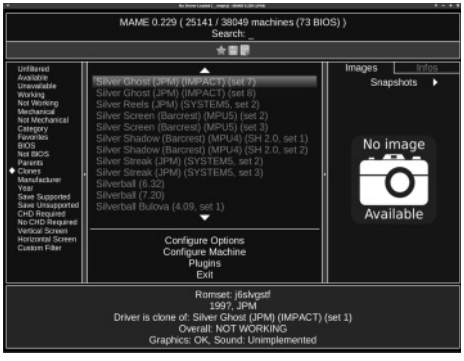
i will probably never see an



apple1 in the wild and if i ever did it'll be behind some rope in a museum somewhere. mame allows me to boot the apple1 driver and have a go at using one, or even develop on it. yes 'its not real hardware' but in leui of real hardware should we never experience it? mame provides an amazing learning opportunity, its not the only project but its arguable the most accessible

::Troubleshooting

another core use for mame i have read about is using it to assist in troubleshooting real hardware issues. people have reported dumping problem roms and loading them into MAME to see if the problem is present in the driver, or validate the dump against the known good file hashes cataloged by the project. aiding in the preservation efforts at a hardware level.



::Public_Opinion

most peoples introduction to mame is a mame cab with 1000s of titles (for 'free') on it at a mates place, search any social platform and you'll confirm the same.

this is starting to change for the better as social platforms. they give rise to the cultural voice of preservation over the industry diatribe of piracy and those would be mamecab heads.

even if unintentional, the like's of haze's live streams on these platforms are at least providing the opportunity for others to have an alternate introduction to the project.

i doubt i've communicated this as well as i'd have wanted too but go experiment and learn systems in mame. you may never get to experience the real hardware in your lifetime.

- sairk

making mame visually fit the era isn't such a struggle anymore; the before and after shots throughout this article should demonstrate how advantageous and gorgeous the crt 'emulation' is as it is packaged with mame. if not, try it for yourself.

we'll look at the bgfx chains in this article specifically the crt-geom-delux chain. which i tried out for the first time last night and it was a revelation.

i don't normally go in for this stuff, even after having pretty much the only article around for a long time on the glsl shader setup for linux (on mameau), in reality i barely used it in general use.

it was more a curiosity.

the bgfx chain however. the change between the default and crt-geom-deluxe chains is dramatic across multiple drivers.

so what does it take to get it up and running? two changes to the mame.ini for the basic implementation. it is a lot simpler these days, we are

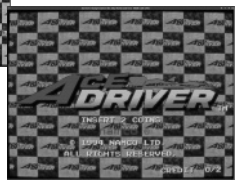
```
Edit the mame.ini
 vi ~/.mame/mame.ini
 make the following changes

Set the video mode
 video bgfx

Select your bgfx chain
 bgfx_screen_chains <chain name>
```

currently using the mame 0.229 release which comes packed with ~12 chains (best i can tell) and these are constantly evolving.

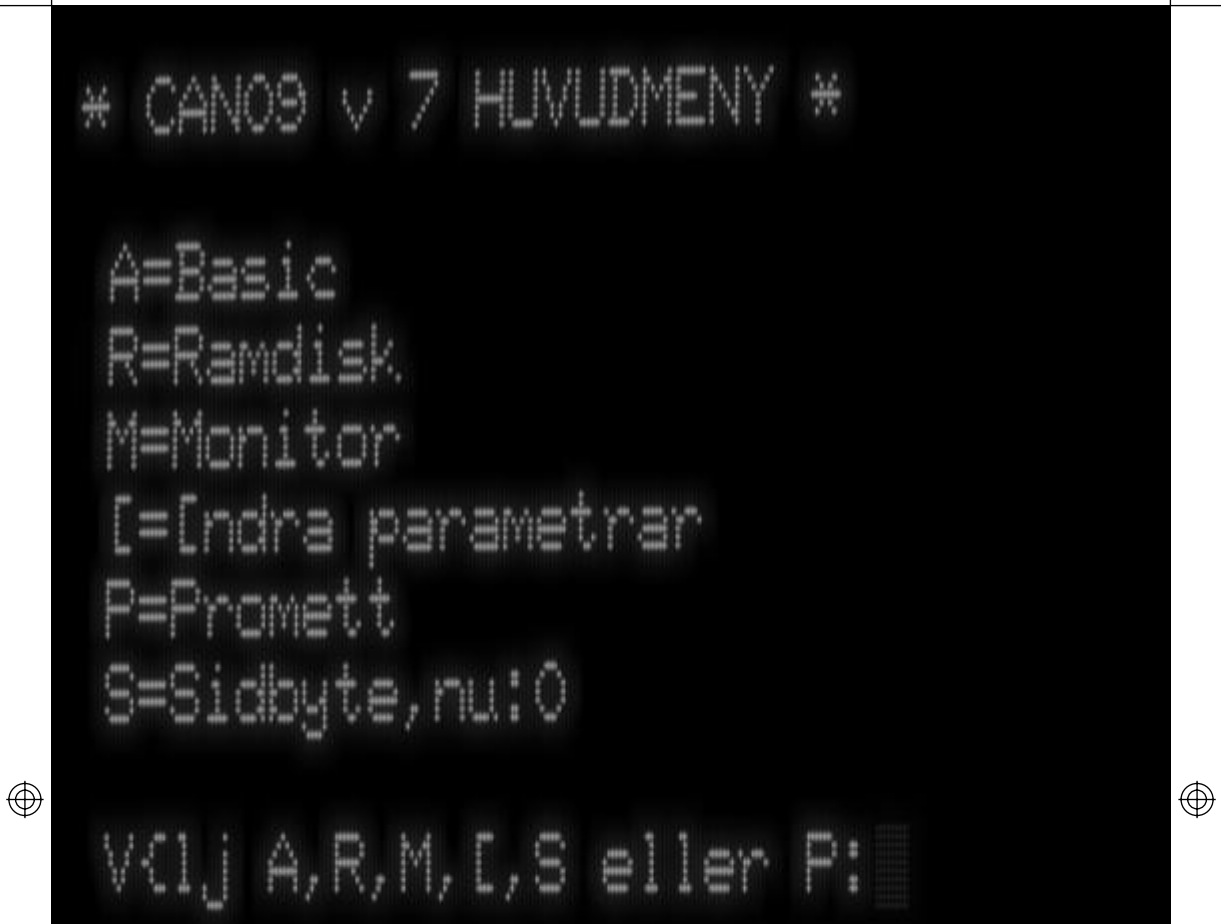
the chains themselves are a json file that seem to document a number of layers applied globally, perhaps i'll get the time to dig into this more in a follow up article one day.



whatever your feelings on scanlines or crt emulation in general i think you would be hard pressed to find an argument against this implementation in leui of access to a real crt.

as with all things i would suggest starting with the official doco over anything else, not yet a crt replacement but its edgig ever closer considering how varied as crts really were, one chain will probably never cover everthing but its exciting to see the work.

- sairuk



About

Members

ed - sairuk (@sairukau)

Greets (no order)

slateman
blahjedi
mamedev
haze
stiletto
spacey
duke
arbee
sh0ck
mucci
newuzer

//scrape.mameau.com

Creds

mamedev
- https://www.mamedev.org/

mamehaze youtube channel id
- UCcd55nv9pTcFNlXOIax7fiQ

bannister forums
- https://forums.bannister.org/

mametesters
- https://mametesters.org/

messing around in mame yt list id
- PLcXJO3ljjoqQrNQSid5wp6P4MPtJmfoC-