

KANSASFEST 2021

ADVENTURES IN VIRTUALITY

=====
FRIDAY JULY 23
=====

All times: CDT, GMT-5

0945-1000

Welcome to Virtual KFest

Introduction of committee members; tips and tricks for having two fantastic days.

1000-1030

Nox Archaist: A look back and the road ahead

Mark Lemmert & Chris Torrence

Mark Lemmert and Chris Torrence talk about some of the challenges and excitement during the development of Nox Archaist, and share their thoughts on future projects from 6502 Workshop.

1030-1045

Remote Key Control of Your Hardware Apple IIe and IIGS

Jay Craft

Jay Craft will describe the hardware and software required for remote keyboard control of your Apple //GS or //e from a modern-day computer. He will demonstrate use cases in testing, development, controlling multiple Apple // systems simultaneously, and sharing the hardware Apple //'s keyboard with someone in a remote location.

1045-1115

A.P.P.L.E.'s Blast From the Past

Brian Wiser & Bill Martens

Join Brian Wiser and Bill Martens for highlights of new A.P.P.L.E. software and books that we've produced this year and are introducing at KansasFest, along with some special announcements and surprises! As the oldest user group from 1978, we've created a variety of Web sites (like MECC, Beagle Bros, Applied Engineering, Apple Archives), over 40 previous books, and dozens of programs.

1115-1130

Coffee Break

1130-1200

Double Hi-Res like you've never seen it before

Kris Kennaway

Kris will describe][-Pix, a Double Hi-Res image converter for modern PCs, designed to produce colour-optimized images that take full advantage of the peculiarities of DHGR. Double Hi-Res graphics are difficult to program. Like all Apple II graphics

modes, they also suffer from colour interference effects that can be difficult to understand. In order to make sense of this, we'll do a deep dive into DHGR colour, coming up with some simple rules for understanding the colour interactions, and answering the question: How many colours does Double Hi-Res support, anyway? With this deeper understanding, we can turn the complexities of the colour model to our advantage, and use them to produce higher quality images than had previously been possible.

1200-1230

The first IIGS game: The Creation of Tass Times in Tonetown

Rebecca Heineman, aka Burger Becky

How did a game for the Apple][get ported to a brand new platform, using a new CPU and new graphics and sound, and get done in 4 weeks? I'll tell you. You may go mad.

1230-1245

Maximizing Apple Color Bit by Bit

Lucia Grossberger Morales

In 1979, Lucia Grossberger Morales was devastated when she became allergic to her artistic medium of oil paints. When she saw the Apple II computer, she bought it to create art. LGM understood the basics of computers and was confident she could make art, though she had no idea what Apple II art would look like. LGM's inspiration for creating art on the Apple II Hi-res screen was needlepoint, not paint. When coauthoring the Designer's Toolkit, published by Apple Computer in 1982, she included a color/pattern creation program where she carefully picked out different patterns on a 4x4 grid. (Examples of her work are available at luciagrossbergermorales.com.)

1245-1300

Porting a Vector Arcade Game to the Apple][

Steven Hugg

A mysterious email from a 1970s game manufacturer kicks off the port of a vector arcade game to the Apple][. We'll take the 6502 code from the arcade game and integrate it with a software-defined DVG (Digital Vector Generator) that renders the game faithfully.

1300-1345

Lunch Break

1345–1415

A2osX and the Languages of Development

Patrick Kloepfer

Patrick will introduce and demonstrate the multiple new languages available in the A2osX 0.94 release. The languages will include the Bash-like interactive and scripted shell (bin/sh), an assembler (bin/asm) fully compatible with the S-C Macro Assembler, a Basic-like scripting language based on ACOS for developing a full fledged BBS in A2osX, a Forth implementation based on Forth-79 with extensions for rich graphics, and lastly a C shell (bin/csh) language with access to all of A2osX's APIs and libraries for building complex applications and utilities. This presentation will cover each of these languages, its primary use cases and running examples for each.

1415–1445

Sinotype 3: Chinese word processing on the Apple II

Bruce Rosenblum & Tom Mullaney

This session will describe the development of Sinotype 3, a Chinese word processor for the Apple II that was developed in 1981. It will be co-presented by Bruce Rosenblum, the developer of Sinotype 3, and Tom Mullaney, associate professor of Chinese History at Stanford University and author of the forthcoming book, *The Chinese Computer*.

1445–1515

Calibrating a Disk II the HARD way!

Petar Puskarich

After many years, I have finally gained all the tools necessary to follow the Level III guides to properly align and restore a Disk II floppy drive. Includes the oscilloscope, Special Apple Alignment floppies, and my attempt to recreate them, the special aluminum floppy tracking tool, as well as getting a Disk II to actually align after many years. Will cover a few common faults and chip issues during the process. Will discuss the various Disk II controller pcb versions.

1515–1530

Coffee Break

1530–1600

Tiny Programs for the Apple II

Vince Weaver

From the Demoscene to the Apple II Basic Twitter bot, there has been a resurgence in writing tiny code for the Apple II. Vince will discuss techniques used to write graphical demos ranging in size from 280 bytes down to 8 bytes, including some tricks used to get 142-byte machine language programs to fit in a plain-text tweet.

1600–1615

The Apple][as a Home-brew Computer Development Platform

Scott Berk

I built a computer from scratch out of transistors and taught my Apple][+ to talk to it. The "Spikeputer" is a 16-bit RISC Microprocessor made of discrete electronic components and LEDs in order to visualize computation. I built an SPI peripheral card for the Apple and an SPI client board for the computer so the Apple can be used to access memory bi-directionally and serve as an I/O device for the computer. The session will describe the Spikeputer architecture and build plan, then discuss the hardware and software components for communication with the Apple][+.

1615–1645

CyanIIIDE, a web-based Apple II dev environment

April and Melody Ayres-Griffiths

April and Melody return to demonstrate CyanIIIDE, a web-based Apple II development environment, allowing for the straightforward creation of Apple II assembly-language software in the browser. CyanIIIDE has full reverse-debugging capabilities and the ability to freeze and restore debugging sessions, as well as the ability to save and load projects, and build directly into a running emulator session or to a bootable disk image. Planned features include cloud-based collaboration tools, library sharing, and the ability to create 'packaged' executables to distribute for Windows, Mac, Linux or on the web. If time allows, they will also provide updates on their microM8 Apple II emulator, turtleSpaces (an expanded 3D Apple Logo II environment) and other new projects.

1645–1715

DMAC The DMA Memory and System Controller

Michael Morrison

Michael will present the DMAC: a hobbyist custom card for the Enhanced Apple IIe that provides hardware-assisted Direct Memory Access to accelerate graphics, text, memory management and control audio. The DMAC augments the Apple IIe by providing programmable elements similar to the hardware in an Amiga™ such as a 3-source, 256 ROP Blitter; Primitive rendering including lines, rectangles and triangles; a "Copper" that is able to control soft switches per-scanline; and an Audio engine for driving the Apple's 1-bit audio using pulse-width modulation. All text and graphics modes will be supported, including Double Hires. The presentation will consist of an overview of the device, programming information, demos, and future directions.

1715–1720

Day 1 Wrap Up

1900–????

Join us on Discord for games, chat, & video.

=====

SATURDAY JULY 24

=====

0945–1015

Apple][Forever Awards

Presentation of awards to individuals for their special contributions to the Apple II community.

1015–1030

Funny mode on the Apple ///

Paul Hagstrom

The Apple /// has an Apple II emulation mode, often mocked for its limitations. It is, however, possible to operate the Apple /// in a kind of hybrid mode ("funny mode"), but it requires some trickery to jam a II-shaped-peg into a ///-shaped hole, and the fit is still only approximate. This presentation will provide a medium-speed jog through George Oetzel's trilogy of articles in *Softalk* (1983), to demo and explain the challenges and limitations posed by the differences in hardware architecture—in the process, providing some appreciation of why the Apple II emulation mode is limited in the ways that it is. And then a couple of neat things you can do with the hybrid mode, with 2 MHz processing, fancier colors, but in the comfort of Applesoft.

1030–1045

Building out the ultimate IIe Card for your Macintosh

Ron McAdams

So you've got a IIe Card for your Macintosh, but no idea how to get started using it to live out your wildest dreams of running classic Apple II software. This session will cover modern update solutions for the IIe Card. What works, what doesn't, how to make use of one of Apple's best tools ever made for educators!

1045–1115

Setting up a ProDos boot partition on the Macintosh Hard Drive with your Apple IIe Card

Jay Graham

I will show how to setup the Macintosh Hard Drive in an LC to boot to ProDos using the Apple IIe emulator card.

1115–1130

Coffee Break

1130–1145

Memories of an Apple II hacker

Jean-Philippe Cunnet

The vibrant European 1980s Apple hacker scene: memories from a major hacker, Apple's Godfather, author of the "Deplombage mode d'emploi" series of hacking lessons and more. Here is my interview during the Apple Festival:

<https://youtu.be/ajYzZg9ztPU>

and more info:

<http://boutillon.free.fr/Underground/People/Godfather/Godfather.html>

1145–1215

Entropy on the Apple II

Kevin Greene

Is your Apple II random enough? Kevin Greene will explore a way of generating entropy using hardware you probably already have.

1215–1230

Ample MAME Emulation on the Mac OS

Kelvin Sherlock

MAME is one of the best Apple II emulators available. But nobody ever accused it of being easy to use. Tame your MAME with Ample, a MacOS front-end for MAME. <https://github.com/ksherlock/ample>

1230–1300

Honeylisp

Jeremy Penner

What could programming the Apple II be like if we fully harnessed the ridiculous power of modern computers? Jeremy Penner will present Honeylisp, an ongoing experiment in building an interactive development environment using Fennel, a young lisp-inspired language. Honeylisp includes an assembler, graphics editors, and deep emulator integration which allows for live updating of running code. It has so far been used to create the puzzle game Neu] [ower, and 8-Bitsy, a tool for creating simple tile-based exploration games.

1300–1345

Lunch Break

1345–1415

Black IIc: Modding a Classic

Javier Rivera

The Apple IIc is a classic and iconic computer, but I will show there's always ways to upgrade or make it better: presenting the black Apple IIc, V2.0. This computer has been enhanced beyond color, which I intend to showcase.

1415–1445

Reverse Engineering Jef Raskin

Eric Rangell

Jef Raskin did a lot more than design the Macintosh. As publications manager for Apple, he taught a lot of us how to use and program our Apple II computers, and how they worked under the hood. He had very specific visions about human computer interaction, which led to the development of the Swift

card and the Canon Cat. In this session, Eric Rangell will explore Jef's interests and present new software that lets any Apple //e or //c run the Swyft system (designed by Information Appliance) either on real hardware or in emulation.

1445–1515

Inside BuGS Graphics Code

Jeremy Rand

Jeremy Rand will walk through some of the key aspects of the graphics code in the BuGS arcade game released this year. This game runs at 60 frames per second on a standard Apple IIGS. The session will cover some of the standard approaches for high speed graphics on the GS and some unique things about BuGS in particular.

1515–1530

Coffee Break

1530–1545

How I accidentally archived hundreds of Apple II source code floppies

Kay Savetz

This year, Kay Savetz's hunt for a copy of a single computer magazine article led to them being handed a massive stack of Apple II source code disks — all in the Forth language. How did that happen? What the heck is Forth anyway? Did Kay ever find that article?

1545–1600

KBOOHK's DIY SmartPort Rom

Petar Puskarich

Discussing the details of the setup and usage of the DIY form of the KBOOHK Smartport ROM that lives on the Grappler Plus card. Quick demo of the setup and usage with a FloppyEMU and the special caveats that must be considered when using this system. Will show on a II+ and IIe if time allows.

1600–1630

Understanding Blockchain with the Apple II

Charles Mangin

Charles Mangin is back with his cryptocurrency shenanigans. Last year, he demonstrated solving Bitcoin hashes and "mining" coins (very slowly) with an Apple II. This time around, he will use the Apple to demonstrate the concept underlying cryptocurrency and NFTs: the blockchain.

1630–1700

Video playback from a CFFA3000

Kris Kennaway

At KFest 2019 Kris demonstrated][-Vision, a (Double) Hi-Res video and audio player for the Apple II using the Uthernet II ethernet adapter to stream from a modern PC. In this talk he'll describe how he extended this to support local playback using a CFFA3000 compact flash adapter card. He'll discuss how he reverse-engineered the CFFA3000 firmware and dis-

covered some key details that effectively allow executing an unlimited amount of 6502 code from mass storage, at about 66% of native speed.][-Vision was then extended to compile modern video files into 6502 assembly programs, which cause the Apple II to render a double hi-res video animation together with audio playback, when executed using this technique.

1700–1715

An Apple II Emulator Inside the Unreal Engine

Josh Weier

Josh Weier has written an Apple II emulator that runs entirely inside the Unreal Engine. This required not only recreating the internal solid-state components of the computer, but also replicating the external physical components like the lights and drive motors. He'll be spending some time showing what this all looks like, how it came about, and how he achieved it. There will also be a little discussion of the larger game project this fits into, called "zero_page".

1700–1730

KFest Closing - See you next year in Kansas City!



Check out our new store at kansasfest.org/shop
Your purchases help support KansasFest!