FujiNet Apple II Demonstration and Development

Jeff Piepmeier (Apple II lead)
Petar Puskarich (demo-meister)
Thomas Cherryhomes (ring-leader)

FujiNet brought me to the Apple II

My first computer experience: TI Silent 700 with acoustic coupler at age 5

My first computers: Atari 400, 800XL, then Amiga 500

My first retro computers: Atari 130XE, C64, Apple II+

Apple II FujiNet Contributors & Friends

Joseph "Mozzwald" Honold

Thomas Cherryhomes

Robert Justice

Katherine Stark

Petar Puskarich

Andy Diller

Ron Klein

J. Craft

FUJI■■NET

FujiNet is …

- Multi-device emulator & wifi adapter for your 8-bit
- Atari to start; Adam followed; now C64 and Apple II in works
- What does this mean?
  - Load disk images over the cloud
  - Create network games
  - Print to PDF with custom vintage printer fonts
  - Use CP/M via UART

First Cloud Boot Reactions

From: ʞɐᴉuzoʍ ǝʌǝʇs
Subject: First Successful boot of #Apple2 #FujiNet over the cloud! - YouTube
Date: January 6, 2022 at 10:35:21 AM CST

F**King crazy.

https://www.youtube.com/watch?v=L43SsS60Yik

ʞɐᴉuzoʍ ǝʌǝʇs ☐

It doesn't work that way
-- Ariana Gillis, John and the Monster

If you see somebody's cold, Give 'em a coat
— No-no Boy, Boat People

PGP (GPG) fingerprint 5796 CED6 9AC6 7EC1 B42B 56CB 7933 8524 A90B B342

6

FujiNet Hardware and Software Framework

- Open source hardware and software
- ESP32 is the heart of FN (no CPLD or FPGA - so far!)
- Each vintage target has own I/O circuitry and connectors
  - Atari: SIO, open collector serial
  - Apple: IDC20, TTL compatible I/O
- Common software
  - PlatformIO in VSCode
  - Reuse common elements, e.g.,
    - Filesystems
    - Web interface
    - Network protocol adaptors
    - Printer emulators
  - Custom bus and device classes

FN Talks SmartPort (Protocol Converter) Interface & Bus

Generic SmartPort calls

Generic SmartPort calls are explained in detail in the following sections.

Status

The Status call returns status information about a particular device or about the SmartPort itself. Only Status calls that return general information are listed here. Device-specific Status calls can also be implemented by a device for diagnostic or other information. Device-specific calls must be implemented with a status code of $04 or greater. On return from a Status call, the X and Y registers contain a count of the number of bytes transferred to the host. X contains the low byte of the count, and Y contains the high byte of the count.

	Standard call	Extended call
CMDNUM	$00	$40
CMDLIST	Parameter count	Parameter count
	Unit number	Unit number
	Status list pointer (low byte)	Status list pointer (low byte, low word)
	Status list pointer (high byte)	Status list pointer (high byte, low word)
	Status code	Status list pointer (low byte, high word)
		Status list pointer (high byte, high word)
		Status code

Generic SmartPort calls

Init, Status, Ctrl, Write(Block), Read(Block)

WRITE: 4-us edges (or not) for 1 & 0

READ: 1-us pulses (or not) in 4-us windows

Learning the SmartPort Machine Language Interface

```
300:JSR C50D
303:00 is for a status command
304:13 low byte for location of parameter
list
305:03 hi byte location of parameter list
306:STX 310
    STY 311
    STA 312
    RTS
310: low byte number of bytes returned
311: high byte number of bytes returned
312: error code returned
313: 3 (parameter count) parameter list
314: 5 - FujiNet unit number
315: 18
316: 3
317: CMD - status code command to FujiNet
318: begin of return list
```

Learning the SmartPort Machine Language Interface

```
300:JSR C50D
303:00 is for a status command
304:13 low byte for location of parameter
list
305:03 hi byte location of parameter list
306:STX 310
     STY 311
     STA 312
     RTS
310: low byte number of bytes returned
311: high byte number of bytes returned
312: error code returned
313: 3 (parameter count) parameter list
314: 5 - FujiNet unit number
315: 18
316: 3
317: CMD - status code command to FujiNet
318: begin of return list
```

```
BSAVE FNSTAT.BIN,A768,L32
```

Learning the SmartPort Machine Language Interface

```
300:JSR C50D
303:00 is for a status command
304:13 low byte for location of parameter
list
305:03 hi byte location of parameter list
306:STX 310
    STY 311
    STA 312
    RTS
310: low byte number of bytes returned
311: high byte number of bytes returned
312: error code returned
313: 3 (parameter count) parameter list
314: 5 - FujiNet unit number
315: 18
316: 3
317: CMD - status code command to FujiNet
318: begin of return list
```

```
BSAVE FNSTAT.BIN,A768,L32

From BASIC,
POKE 791,status code
CALL 768
ERRCODE = PEEK(786)
List of response starts at 792
List is PEEK(784)+PEEK(785)*256 long

Copy the list into a string and print
the string
```

CONFIG

Mount file hosts
(SD and TNFS)

Select and mount
disk images
(currently PO/HDV
supported)

```
------------------------------ HOST LIST
1 SD
2 APPLE-APPS.IRATA.ONLINE
3 RPI400.KDOMAIN.ORG
4 FUJINET.DILLER.ORG
5 EMPTY
6 EMPTY
7 EMPTY
8 EMPTY


------------------------------ DRIVE SLOTS
1 OFF
2 OFF
3 OFF
4 OFF




1-8:SLOT  EDIT  RETURN:SELECT FILES
CONFIG  TAB:DRIVE SLOTS  ESC:BOOT
```

FUJI●●NET

What's so special about CONFIG?

- Common core *cc65* code for all FujiNets
- Exploits the SmartPort interface specification
- Uses over 30 custom `status` and/or `control` commands
- For example,

```
#define FUJICMD_GET_SSID          0xFE
#define FUJICMD_SCAN_NETWORKS     0xFD
#define FUJICMD_GET_SCAN_RESULT   0xFC
#define FUJICMD_SET_SSID          0xFB
#define FUJICMD_GET_WIFISTATUS    0xFA
#define FUJICMD_MOUNT_HOST        0xF9
#define FUJICMD_MOUNT_IMAGE       0xF8
#define FUJICMD_OPEN_DIRECTORY    0xF7
#define FUJICMD_READ_DIR_ENTRY    0xF6
#define FUJICMD_CLOSE_DIRECTORY   0xF5
```

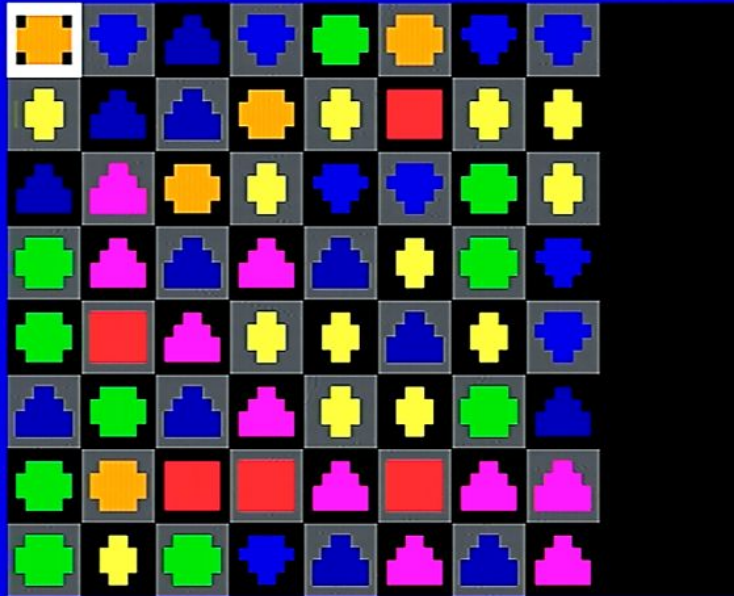*CONFIG & web interface remove need for a dedicated LCD/TFT/OLED display*
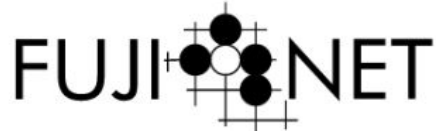
So what can you do?

- Play JOUST of course!

- Save and load binary and BASIC programs
- Can also boot Appleworks, create, save and load documents.

A2 Desktop

R.Klein

FUJI●NET

NET●WORK

FujiNet version	0.5.116ceebb 2022-06-25 17:00:46
Host name	atari-apple-rev0 [Save]
IP address	10.0.0.240 / 255.255.255.0
Default gateway	10.0.0.1
DNS server	75.75.75.75
WiFi SSID	_irisindigo_
WiFi BSSID	AC:DB:48:96:69:8F
WiFi MAC address	8C:4B:14:15:8E:F0
WiFi details	chan=1, chan2=20-none, rssi=-37, auth=WIFI_AUTH_WPA2_PSK, paircipher=WIFI_CIPHER_TYPE_CCMP, groupcipher=WIFI_CIPHER_TYPE_CCMP, ant=0 11b=y, 11g=y, 11n=y, lowr=n, wps=n, (ccode=0x555320, firstchan=1, numchan=11, maxpwr=30, policy=auto)

HARD●WARE

Detected Hardware Version	1.6.1 and up
SPIFFS size	1,920,401
SPIFFS used	848,631
SD size	63,847,890,944
SD used	2,981,888
Uptime	9 minutes, 54 seconds
Current time	Thu Jul 14, 13:12:24 2022 -0400
Free heap	4,055,507
SOC SDK	4.3.1
CPU revision	3
SIO voltage	0V
HSIO Index:Baud	FN_SIO_HSINDEX:

HOSTS●LIST

Host 1	fujinet.diller.org ::
Host 2	zeus.kdomain.org ::
Host 3	FUJINET.ONLINE ::
Host 4	(Empty) ::
Host 5	(Empty) ::
Host 6	(Empty) ::
Host 7	(Empty) ::
Host 8	(Empty) ::

MOUNT●LIST

Drive Slot 1 (D◆:) [EJECT]	FUJINET.ONLINE :: /ISS.po (R)
Drive Slot 2 (D◆:) [EJECT]	:: (Empty)
Drive Slot 3 (D◆:) [EJECT]	:: (Empty)
Drive Slot 4 (D◆:) [EJECT]	:: (Empty)
Drive Slot 5 (D◆:) [EJECT]	:: (Empty)
Drive Slot 6 (D◆:) [EJECT]	:: (Empty)
Drive Slot 7 (D◆:) [EJECT]	:: (Empty)
Drive Slot 8 (D◆:) [EJECT]	:: (Empty)

GS/OS

FUJI●●NET

- Used FN web interface to mount Arkanoid
- Booted GS/OS using MDT
- FN recognized as disk device by GS/OS
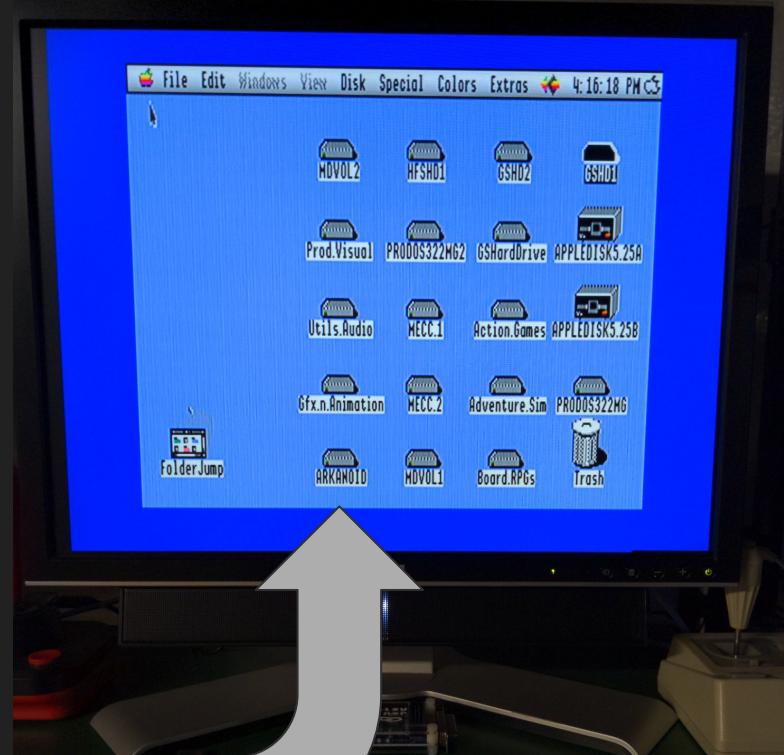
R.Klein

What? An Apple *///*

Rob J.'s "hacked ProDOS1.0 disk image and modified loader.system."

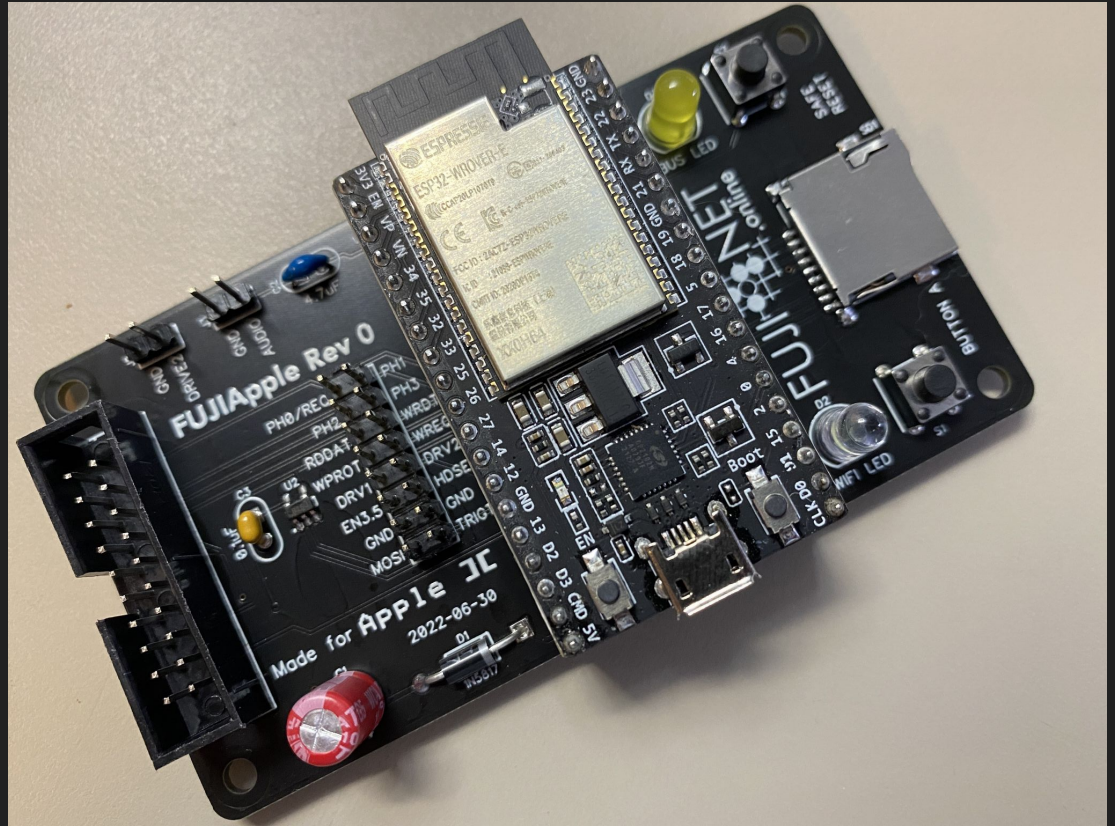"using a Liron card in the Apple /// to connect it"

FUJI NET

ISS Tracker

- Mount with CONFIG
- Boots off FN
- Uses SP calls to the Network device
- Leverages FN-based JSON parser

FUJI NET

Rev 0 dev board

- ESP32 WROVER
- Buttons & LEDs
- microSD Card
- IDC20 connector
- Test points
- DAC out (SAM?)
- Diode-OR'd 5V

Please join us!

- HELP US FINISH THE FUJINET BASIC.SYSTEM HANDLER
- Writing little programs to test things
- Apple /// SOS driver
- Apple][Desktop support
- A2OSX support
- Apple Pascal support
- SmartPort Bus decoder for PulseView