SEGA CHIHIRO
SEGA NAOMI
SEGA TRIFORCE

WiPi Netbooter

Manual

VERSION 6.1

Table of Contents

WiPi Netbooter Overview

This version of the Pi Netbooter code is a scratch rewrite of the original solution written by devtty0 and has been enhanced with a new user interface and richer functionality. It has full support for all netbootable Sega arcade ROMs for the Naomi, Naomi2, Triforce, Chihiro and the Atomiswave conversions made possible by Darksoft. This version also includes the card reader emulator code for games that support it, the original python scripts were written by Winteriscoming on the arcade-projects.com forums and have been adapted for use in a web interface. The entire netbooting suite of scripts including the on screen menu and server mode was written by DragonMinded and integrated into WiPi. OpenJVS is provided as well, and was developed by bobbydilley.

Requirements

- Raspberry Pi v3B, 3B+ or 4B and microSD Card - 32GB Class 10 card recommended
- Naomi, Naomi2, Triforce or Chihiro with a netdimm running firmware 3.03 or greater
- Standard network cable and 5v power source for the Pi – you can make a custom cable to draw power directly from the system (see below)
- Web Browser :)

Optional

- (***Recommended***) Zero security pic chip
- Trendnet TU-S9 or Prolific PL230x based USB-Serial adaptor and custom serial cable for the Card Emulator. *See* **Advanced** for cable pinout.
- FTDI or CP2102x based RS485 to USB adaptor for OpenJVS (see https://github.com/OpenJVS/OpenJVS for more information)
- OpenJVS Pi HAT (see https://github.com/OpenJVS/OpenJVS for more information)
- ACS ACR122U or ACR1281U-C1 NFC Card Reader
- Logitech C270 Webcam for Mario Kart GP and GP2 camera support

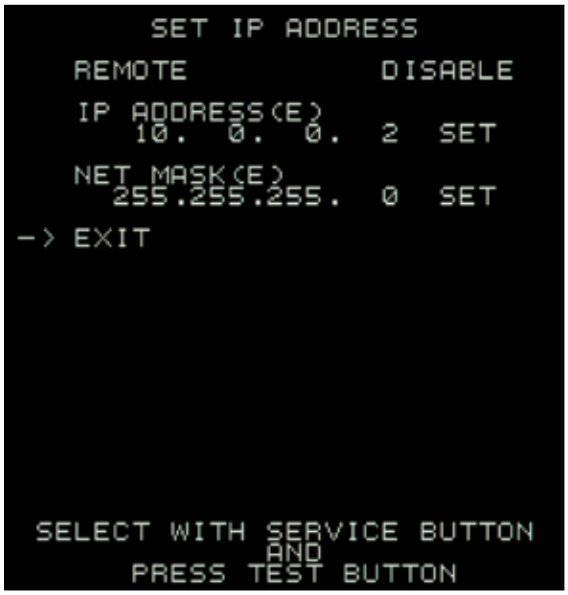
Getting Started

Download all files from the download link and extract the img file using WinRAR or similar.

Write the image to a 32GB microSD card using Win32DiskImager or Etcher and insert into the Pi.

Power up your arcade board, and plug the network cable between the Pi and the netdimm network port. Power on the Pi.

When the Pi first boots up it will start broadcasting a wireless network called *WiPi-Netbooter*. Join either a computer or mobile device (recommended) to the network using password "**segarocks**". Once joined, open a browser and navigate to http://netbooter.local, this will take you to the main game menu.



The first time you boot up you will need to add a netdimm to the Pi. The network interface on the Pi is set to use IP address 10.0.0.1. Enter the service menu on your arcade board and set the netdimm IP address in the NETWORK SETTING menu. Set NETWORK TYPE to ETHER, REMOTE to DISABLE and use the buttons to enter the address, I recommend using IP address 10.0.0.3 and Subnet Mask 255.255.255.0. Next navigate to the Setup Menu on the Pi and **MKGP, MKGP, WMMT and WMMT2 - Triforce J3 or Chihiro CN5**, you can either manually add your netdimm details or use the scanner function to find it. *See* **Network Configuration** for more info.

When you add your first netdimm you can use the network scanner to find it. This searches the local network connection for available netdimm devices.

Launching Games

The main page of the web interface is used for launching games, simply browse through the game list and select a game to send the ROM to your arcade system. If you are using the Advanced Menu mode (default), when you select a game, you will see extended game information and a gameplay video if one is present on the Pi along with a Launch Game link. Once the loading process is complete a success message is shown, after that you can safely browse to other pages or shutdown the Pi.

Note the Pi needs to stay running if you are using the Time Hack mode with no zero key pic chip. *See* **Options Menu** below for more information.

Main Menu

By pressing Menu in the top left of any screen of the WiPi web interface, you'll bring up a blue overlay that'll allow you to configure various aspects of the Pi system.

- **Game List** – Return to the listing of all currently selectable games. *See* **Launching Games** in previous section
- **Favourites** – View a list of games you've favourited
- **Options** – Display and toggle mode settings
- **Setup** – Used for one-time setup functions and additional features
- **Shutdown** – Safely shuts down the Pi

Options Menu

The Options menu displays the current mode settings and provides links to toggle between them.

Simple Menu

There are two menu modes available: Advanced (default; setting set to disabled) and Simple. Advanced mode links to a game information page that shows you extended information about the game, a video preview if available and a link to launch the game and is the default when Simple mode is not enabled. Advanced mode also allows you to set or remove a game as a favourite for use in the Favourites menu. This information can be easily edited by updating a CSV file held on the Pi.

Simple mode (change setting to enabled) allows you to boot the ROM directly from the main game list page.

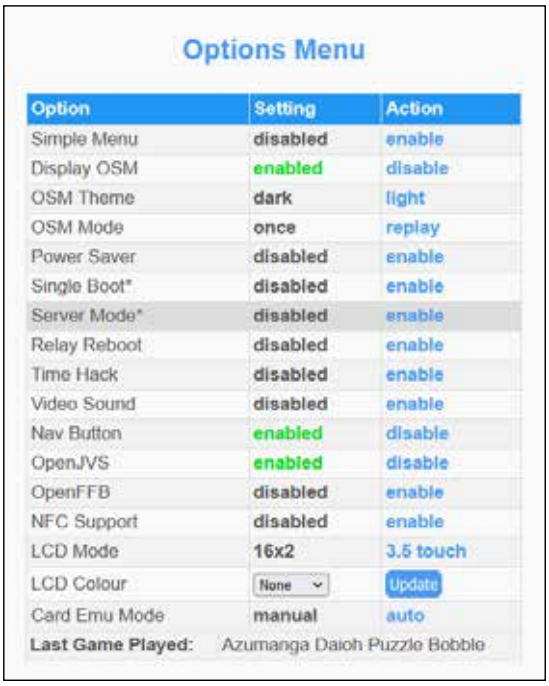
System Filter

This option will auto filter the main game list based on the systems you have configured. Disabled by default.

Display OSM

The OSM or On-Screen Menu is a homebrew Naomi ROM created by DragonMinded and will send a menu selection screen to a Naomi or Naomi 2. You need to have a Naomi netdimm set up for the menu item to appear and it will be pinned to the top of the main game list for convenience.

OSM Theme



Toggle between light (black text on white background) or dark (default; white text on black background) modes.

OSM Mode

When set to once (default), the Pi will send the menu one time. Setting to Replay will send the menu again when the arcade board is restarted (and **Single Boot** is enabled). This is useful if your Pi is powered separately from the cabinet. If you have **OpenJVS** enabled and the OSM has been loaded at least once since Pi power on, pressing the **Exiting a Game** combination will reload OSM.

Note that you may have an issue with OSM freezing if you're using OpenJVS for controls. Discrete JVS IO boards should work without any problem.

Power Saver

There are two power modes to choose from, Always-On and Power Saver. Always-On works as its name suggests, you should use the Shutdown link from the web interface to safely shut down the Pi, this is the default setting.

Generally speaking, you probably won't get any problems from simply powering the Pi off, but there is a chance that the microSD card may become corrupted if you do. Power Saver will start a timer when the Pi is booted and runs for 10 minutes before shutting the Pi down. This leaves enough time to open up the web interface and change any options you need to.

Note that the timer cannot be stopped or started from the web interface, so switching between power modes requires a reboot of the Pi.

Single Boot

The WiPi Netbooter has two different boot modes: Multi (default; setting set to disabled) and

Single (setting set to enabled). Multi-mode requires you to manually launch the game from the web interface every time you want to play and is the default. Enable Single Boot mode to automatically boot the selected default game for each netdimm when the Pi starts up (*See* **MKGP, MKGP, WMMT and WMMT2 - Triforce J3 or Chihiro CN5**). The Pi will wait until it can communicate with each netdimm before sending the game so power up timing is not important.

Note that some games may not allow you to hot boot another game while one is already loaded. This affects the Atomiswave games in particular. These games can take up to 30 seconds to restart the system, so to avoid getting stuck in a loop when booting those games in Single boot mode you need to wait for the restart. If you do get stuck in a loop, disable Single mode, reboot the Pi and select your new game before re-enabling Single mode.

Server Mode

This is another feature written by DragonMinded and works like Single mode, but allows the Pi to remain powered up on the network and acts as a netbooting server. Simply select which default game(s) you would like for each netdimm (*See* **Manage Netdimms**) and Server Mode will always boot those games when the netdimm can be reached on the network.
To change a game simply update the default and Server Mode will launch the new game immediately. The main game list is not available in this mode as booting any other game will result in Server Mode overriding that game with the default. Disabled by default.

Note that Server Mode requires the Memory Check to run, so loading games will be slightly slower. You will notice an asterisk (*) next to Single Boot Mode and Server Mode. When you enable Server Mode, Single Mode is also enabled. This is intended as Server Mode is an extension of Single Boot Mode.

Relay Reboot

Relay Reboot mode is for use with an optional relay connected to the Naomi power or fan speed wire. When a game is launched, and this is enabled it will send out a signal on GPIO pin 40/GPIO26 which triggers the relay to cut power and soft reboot the Naomi. This is for games that will not allow you to hot boot another while it is running. The feature is now largely redundant with the latest version of the Naomi multibios. Disabled by default.

Time Hack

Time Hack mode is used when a null pic chip is not present in the netdimm. When enabled this will send a special packet to the netdimm to reset its security check. This requires the Pi to be left connected to the netdimm and powered on while the game is running. Disabled by default.

Video Sound

Video sound enables or disables sound in the preview videos in Advanced menu. *See* Simple Menu above. Disabled by default.

Nav Button

Nav button enables and disables a “jump to top” floating button on the main game select menu. Enabled by default.

OpenJVS

OpenJVS is a software JVS IO emulator which allows you to connect the Pi to the Naomi using an RS485 to USB connector and play games with virtually any USB game controller. See the OpenJVS github page (*see* Page 1) for full instructions. Bluetooth controller support has been added in version 6 which allows scans, pairing and removal via the web interface. If you were lucky enough to purchase an OpenJVS HAT you can enable support for it by selecting the option in the **OpenJVS Configuration** menu (*See* **Enable HAT Support**). Disabled by default.

Hat Rotary (requires OpenJVS Hat)

If you are using an OpenJVS hat you can enable Hat Rotary to allow for changing the OpenJVS mapping on the fly. OpenJVS will work as normal until you turn the rotary dial which activates the new setting. To override it you can simply netboot another game and it will return to normal operation. Disabled by default.

Hat Serial (Pi4 only, requires OpenJVS Hat)

Enabling Hat Serial will default the card emulators to use the physical serial port on the OpenJVS hat. Pinout for making a cable can be found in the **Advanced** section below. Disabled by default.

OpenFFB

OpenFFB adds support for Aganyte’s Sega FFB (Force Feedback) Controller board, the software has been kindly provided by FredoBedo. Does not work with with **OpenJVS**. Disabled by default.

NFC Support

This allows you to enable NFC card reader support by connecting a supported card reader to the Pi to allow save data from the card emulator to be stored on physical cards. There will be more NFC enabled functionality coming in future releases, so look out for those! You can use NTAG215, NTAG216, Mifare 1k or Mifare 4k cards which are cheap and readily

available. Disabled by default.

LCD Mode

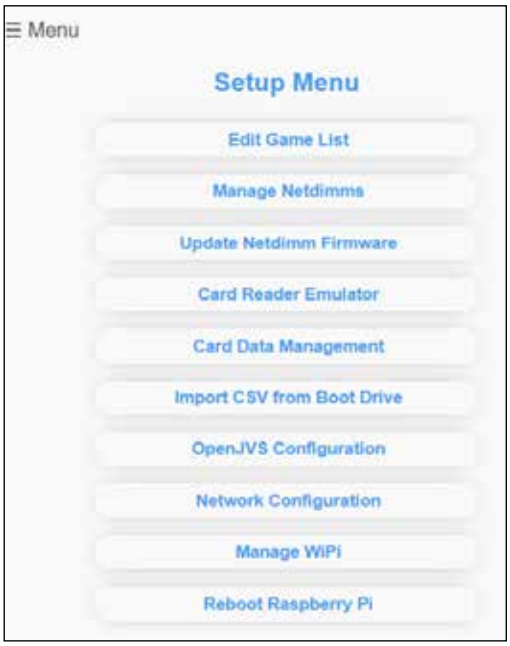
LCD Mode allows you to switch between a 16x2 LCD display or MHS type 3.5-inch touchscreen attached directly to the Pi as an alternative to the web interface via a browser. If no screen is detected the Pi will switch to display only via its webpages. If you have an RGB 16x2 display you can choose the backlight colour. If you have an older non-RGB model, choose None. 16x2 by default.

Card Emu Mode

The card emulator scripts run in either manual or auto modes. *See* **Card Reader Emulator** for more information. Manual by default.

Last Game Played

Shows the last game successfully launched.



Setup Menu

The Setup menu is used for one-time setup functions and additional features.

Edit Game List

The Edit Game List function is used to show and hide games in the main game list. This is useful if you want to load a full set of ROMs onto your SD card but you'd like to hide all vertical, analog and driving games for instance. Use the link in the Enabled column to toggle the setting between Yes and No to Show/Hide the game.

Manage Netdimms

Here you can set up as many netdimms as you need to support. One Pi can be used to netboot multiple systems. You can add and edit them here. The status of the netdimm is shown as "ONLINE" in green if the Pi can reach the netdimm. The system type must be specified as it will be checked when a relevant game is sent. Here you can also choose a default game that will be started automatically on boot up in **Single Boot** and **Server**

Mode, and select a single netdimm for use with **OpenJVS**. To change a netdimm's IP address you'll need to delete and re-add it.

If using a Naomi netdimm you can specify if the cab is horizontal or vertical, this will apply as a filter for the games in the On-Screen Menu. If you have **OpenJVS** enabled you will also see a tick box that is used to tell the Pi which system you are connected to. *Only one can be used for OpenJVS at once.*

Update Netdimm Firmware

Here you can update type 1 netdimms up to version 4.03. It is recommended if upgrading from version 3.x that you update to 4.01 first. An attempt is made to contact the netdimm to determine what version is currently running. This will be displayed on the update page.

Card Reader Emulator

The Card Emulator runs various scripts on the Pi to send and receive data to your Naomi, Chihiro or Triforce to emulate the magnetic card readers used on the original machines. This is useful if you want to get the most out of your games or have simply run out of cards.

To use it you need a serial connection between the Pi and the arcade board. You can either use a USB-Serial convertor or use the OpenJVS Hat serial port. Connection to your arcade system is made via a custom cable. Any Serial-to-USB adaptors that use the Prolific PL230x chipset are compatible. Pinouts for the cables are shown in the **Advanced** section below.

The Card Emulator saves and loads card data via files held locally on the Pi. There are separate files and folders for each of the games as they are not compatible with each other.

The emulator mode is controlled from the **Options Menu** and runs in either manual or auto mode. In manual mode you need to manually launch the emulator by selecting a card to play with or creating a new one. This starts the emulator for that game and loads the card data into memory.

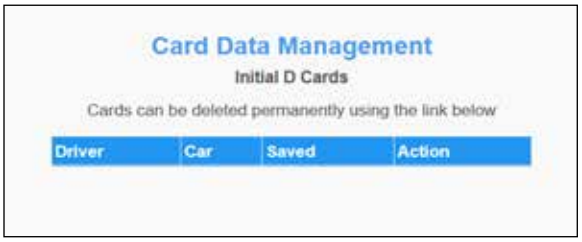
In auto mode the emulator will generate a new blank card to play with and the correct emulator will launch when you load the game, ready to play. If you want to play with a particular card you can select it from the web page when the game is at the Insert Card screen. Auto mode is highly recommended as that is the closest to a real arcade experience.

To launch the emulator either hit the submit button if you are creating a new card or the card name link to launch an existing one. The emulator will fire up in the background and start

communicating over the serial link. If you've already booted the game, reset it via the arcade system's test menu, and be sure to enable the reader in the game test mode! Once you have finished your game and saved progress to your card you can launch another from the menu to continue playing.

The Card Emulator now supports NFC card saves and card printing in the web interface for all emulated games and for all regions. The Pi captures and decodes the print packets that would have been sent to the physical card printer and virtually prints the cards for display in the web interface. All original cards have been scanned and saved in WiPi, and you will be issued a random card at purchase just like the original arcade game.

To use the NFC card feature, enable NFC support in the **Options Menu**, attach a supported NFC reader and use NTAG215, NTAG216, Mifare 1k or Mifare 4k cards to save your data. You can use **Card Data Management** (*See* below) to wipe a card, copy an existing save file or check the contents of a card.



When playing, simply present the card to the reader — the simplest option is to leave it in place on the reader so your data is automatically saved after card ejection. If you have removed the card you will have 10 seconds to replace the card to write save data back. If you are too slow and miss the 10 second window you can manually copy the save data back to the card using the NFC Copy function.

The card reader has beep and light codes:

Card Reading:

- Single short beep with green light – card detected
- Single long beep with orange and green lights – card read successfully
- Double beep with orange and green lights – card read successfully but data not recognised
- Triple beep with orange and green lights – card read error

Card Writing:

- Single short beep with green light - card detected
- Double long beep with red light - card write successful

Card upgrades and renewals are supported via the web interface and the NFC reader.
You will find an orphaned file left behind by the process which can be safely removed. If you have upgraded, this means you can carry on racing with the old card if you wish.

If you are playing Mario Kart or Wangan Midnight games, please note that if you quit a game during a race, you may see incomplete card prints in the interface. This is due to the

printing process on those games and is normal. Card save data is unaffected and print data will be restored after the next full game is completed.

Card Data Management

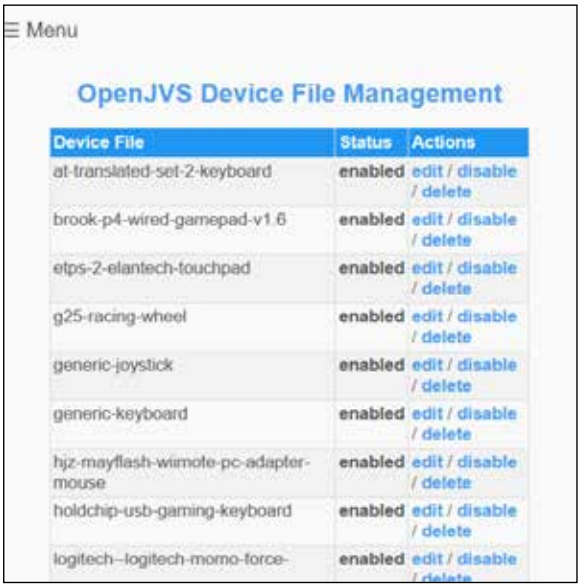
Here you can select one of the supported games and delete (and copy if using NFC) saved card data associated with them.

NFC Game Write

This menu will be present if **NFC Support** has been enabled in the **Options Menu**. From here you can create NFC game cards that, when scanned on a compatible reader attached to the Pi, will netboot that game. You can also write the On-Screen Menu to a card if you would like to use that as a reset mechanism.

Import CSV from boot drive

This is used if you have made any changes to the CSV data on an external computer via the boot drive copy. This will import your changes and overwrite the copy used to populate the main game list. The CSV file holds all the relevant data used by the Pi, including ROM file name, images and videos, description, system type and much more.



OpenJVS Configuration

This is where OpenJVS can be configured for use on your Pi, and assumes you either have a USB RS485 adaptor using the FTDI or CP2102x chipsets or the OpenJVS Hat and it is wired up correctly.

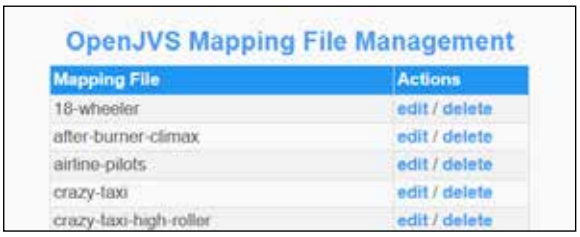
Scan and Configure Devices

This option will look at all of the attached USB hardware and let you know if a mapping file already exists for your controller. Provided you have a config file enabled you are good to go, if not you can use the config tool to create one. This is best done on a PC based web browser.

Manage Device Files

Here you can see all of the installed device files. From this screen you can enable or disable the mappings for a particular controller. You can also use an in-browser editor if you want to changes

change button mappings or analog axis configurations. This is where the physical buttons are mapped to the OpenJVS controller standard layout. Refer to the OpenJVS github for more information or the **Controller Reference** option below.



Manage Mapping Files

Here you can view, edit or delete any of the mapping files. This is where the OpenJVS controller inputs are mapped to in-game inputs.

Update Game Mappings

Here you can change the mapping file to game assignment. This is useful if you have created your own custom mapping file for a game and need to assign it.

Bluetooth Devices

Use this to scan for local Bluetooth devices and pair them with the Pi for use with OpenJVS. *Can't be used with a Pi3 and OpenJVS HAT together.*

OpenJVS Control

From this screen you can start up OpenJVS for testing, or loading a specific game mapping for use with other JVS systems like Lindbergh.

Update OpenJVS

Run a process to update OpenJVS to the latest version. Your config file will be backed up and restored during the update. You'll be prompted to confirm or cancel the operation.

Rotary Mappings

This menu will be present if you have enabled Hat Rotary in the Options menu, here you can assign the OpenJVS mappings to rotary switch positions. When adding a new mapping you must assign the next available number, any unused rotary positions will be set to generic.

Enable HAT Support

Runs a script to enable support for the OpenJVS Pi

Hat device. The Pi will automatically restart after updating. Once run it will not be displayed in the menu again.

Controller Reference

Displays the OpenJVS controller standard layout (Xbox 360/One/Series graphic with a table of mapping keywords) to assist with building device and mapping files.

Network Configuration

This is used to view the existing network configuration for the Wifi and Wired interfaces and allows you to customise your setup. A basic knowledge of networking is required to navigate these settings, if for any reason you lose communication with the Pi, it can be reset by creating a text file called *reset.txt* in the boot drive. You can join the Pi to your home network either using the Wifi or Wired networks, use DHCP or fixed IP addressing, details of the supported configurations are detailed in the pages here.

By default, WiPi Netbooter is set to Hotspot Direct mode. To change this, on the Pi go to the Setup menu and then choose Network Setup.

To use Home Wifi Direct mode so that the Pi connects to your local wifi for management, but is wired directly a single netdimm, click Wifi Setup and enter the SSID and password for your network. Then click *Appy and Reboot*.

To use Hotspot Router mode, connect the Pi and your netdimm by ethernet to your router. You can use a static or DHCP address for buth the Pi and the netdimm, but using static IP addresses is highly recommended. One advantage with this setup is that the Pi can be reached via the home router network as well as the hotspot.

To set up Home Wifi Router mode, connect the Pi to your router by wifi and your netdimm to your router by ethernet. This mode is useful if you don't want to change wifi networks each time you use the netbooter, but the downside is you need to locate your Pi on your home network.

The hostname can also be changed from the *netbooter* (default) to something else. Useful if you have multiple WiPi setups. The default URL will also changes to http://yournewname.local.

Manage WiPi

File Manager

This is a useful file manager utility so you can browse the file system and download files directly from the Pi. You can also launch a terminal console using the Shell button – note the console runs as the default web user so any commands will probably need sudo elevation.

Download Logs

Here you can download log files from WiPi to aid with troubleshooting and diagnostics. First click *Create Zip* and then *Download*. Click *Setup Menu* to go back to the main Setup page.

WiPi Updater

This will display the current version of WiPi Netbooter. Click *Check Updates* to see if there is an update, and if so, you can choose to install it.

Reboot Raspberry Pi

Safely restarts the Pi.

Shutdown

Safely shuts down the Pi.

Exiting a Game

If you're using OpenJVS you can exit a game by pressing and holding Start + any two digital controller buttons for three seconds and then releasing. Exiting the game can take up to 30 seconds, but not every game supports this feature.

Advanced

Connection information for SSH/SCP/SFTP is as follows:

USERNAME: **root**
PASSWORD: **raspberry**

Making a cable to power the Pi from the Naomi is simple. You need a good quality USB cable with either a micro or C-type connector depending on the model of Raspberry Pi and a JST 4-way connector and pins, plus some heat shrink if you like to be tidy! Cut off the type A connector and strip back the shielding to reveal the wires underneath. The 5v and ground are the only 2 you need, and they are normally colour coded red for 5v and black for ground. Trim the green and white data wires off as you don't need them. Crimp JST pins onto the wires and heat shrink the join for a nice finish. On a Naomi you can use the CN12 connector for power, the pinout as you look at the pins is GND-GND-5V-5V so use the outer 2 pins with black to ground and red to 5v.

For those of you who like to code, you can access the source files for the web interface in */sbin/piforce* and */var/www/html*. Feel free to have a poke around, generally if something cannot be done in PHP it's due to permissions, the PHP page calls a python script to execute it on its behalf. OpenJVS files and mappings can be found in */etc/openjvs*. The boot process is as follows:

When the Pi starts up it executes a file called *rc.local* that fires up a python script */sbin/piforce/check.py*. This script checks a few files in the piforce folder to get settings for the power and boot modes. It then sends a netboot command if set to single mode and a shutdown command with a timer if the power mode is set to auto. The CSV file is copied back to the boot partition as part of the shutdown routine.

Most of the web code is PHP so the pages are generated as they are loaded, the benefit is you can make changes on the fly without having to restart the Pi. There is a *sidebarstyles.css* file in */var/www/html/css* that can be modified to change the colours and look and feel of the menus and webpages.

All data for the games is scraped from the *romsinfo.csv* file held in the */boot/config* folder, if you wish to add more columns bear in mind the existing scripts refer to the absolute column reference so you'll need to add any new ones after the existing columns. There is a way to import CSV data as a multidimensional array using PHP but I got lost quite quickly in the coding for that, so my script just reads and writes line by line.

Here are the cable pinouts you need for the Card Emulator to work. I bought a straight through female serial connector cable, cut it in half and crimped a JST NH connector on the end.

F-Zero - Triforce J3

Pin 3 [or 8] (GND)	-	Pin 5 (GND)
Pin 4 (TXD)	-	Pin 2 (RXD)
Pin 5 (RXD)	-	Pin 3 (TXD)
Pin 6 (RTS)	-	Pin 8 (CTS)
Pin 7 (CTS)	-	Pin 7 (RTS)

IDAS, ID2 and ID3 -- Naomi CN8

Pin 1 (RXD)	-	Pin 3 (TXD)
Pin 2 (TXD)	-	Pin 2 (RXD)
Pin 3 (GND)	-	Pin 5 (GND)
Pin 4 (RTS)	-	Pin 8 (CTS)
Pin 5 (CTS)	-	Pin 7 (RTS

MKGP, MKGP, WMMT and WMMT2 -
Triforce J3 or Chihiro CN5

Pin 3 [or 8] (GND) - Pin 5 (GND)
Pin 4 (TXD) - Pin 2 (RXD)
Pin 5 (RXD) - Pin 3 (TXD)

Note: when using the OpenJVS Hat, the RTS and CTS pins must be linked to each other at the arcade board connector with a small jumper wire and TXD/RXD is reversed in the above pinouts.

Version History

6.1-9:

- SRAM save and restore in OSM
- ID card emulator refresh - includes JP support
- Naomi netdimm hori/vert setting for OSM auto filter
- New settings definition files for OSM
- Fix time hack code
- MiFare card support 1k/4k
- Hat Rotary works seamlessly
- Hat Serial support for card emulator
- NFC netbooting
- Main Menu System filters
- File Manager with terminal console

6.1-8:

- Doom Shareware Alpha
- Various OSM Fixes
- Atomiswave in default list

6.1-7:

- WMMT and WMMT2 card printing
- Fix to MKGP workflow

6.1-6:

- Mario Kart GP card printing
- Mario Kart GP camera support
- Improved NFC workflow

6.1-5:

- New light theme for the OSM
- Change via the Options Menu
- New Log Download screen
- Updater moved to Manage WiPi
- Loading bar fixes for non-Chrome
- Timing updates for OSM
- New OSM modes, Once and Replay

6.1 - Feb. 3, 2022

- Proper multi dimm support, all areas now covered, OpenJVS has an extra setting in netdimm management so WiPi knows which system it's attached to
- F-Zero AX card printing and NFC support on Triforce
- DragonMinded's excellent on screen menu for Naomi, this will be in sync with the main game list if you've edited it in the web interface
- Single boot mode now works as originally intended, once booted it will wait until it can reach each configured system then boot the default game once (this can also be the on screen menu)
- New server mode based on DragonMinded's netdimm_ensure script, this works like single boot mode but you can leave the Pi powered up on your network
- New update system, future releases will be uploaded as deb packages and installed via apt so no more giant downloads
- LCD 16x2 colour option added as requested
- No more Triforcetools on the backend, replaced with DragonMinded's netboot suite of tools