

Fred Padilla

PRESIDENT, C-HORSE SOFTWARE, INC.

chorsesoftware97@gmail.com
321.508.7841

Developing software and hardware solutions since the dawn of desktop publishing.

This portfolio is broken up into two parts. Software and Hardware. In both sections, the projects are listed in reverse chronological order, dating back to the late 80's.

- Desktop publishing RIP
- Pre-press proofing RIP
- Direct to garment RIP
- Direct to film RIP (recent)
- Printer management utilities
- Web based copy protection (validation)
- Industry specific electronics (controller boards, etc.)
- FPGA development in Verilog
- Embedded software (firmware)
- Motion control electronics.
- Head control electronics including custom waveform support.

I served as Vice President of Research and Development for Birmy Graphics, which later became iProof Systems, Inc. The automatic white layer creation algorithm I developed, which is used in all the RIP products, is one of the best in the industry. In 1997 I founded C-Horse Software, Inc. A lot of companies and in the proofing, pre-press and DTG industry are familiar with my work. Following is a partial list of projects I've been involved in in reverse chronological order. Some cannot be listed due to NDA's currently in place.



In addition, C-Horse has a private forum where each OEM has their own area which no one else can access. This is used for release notes, bug reports and distribution of software builds. It acts as a repository that both the OEM and C-Horse can access. This saves both me and the OEM from having to look through emails looking for a particular message.



Fred Padilla

PRESIDENT, C-HORSE SOFTWARE, INC.
chorsesoftware97@gmail.com
321.508.7841

C-HORSE
SOFTWARE, INC.
www.c-horsesoftware.com

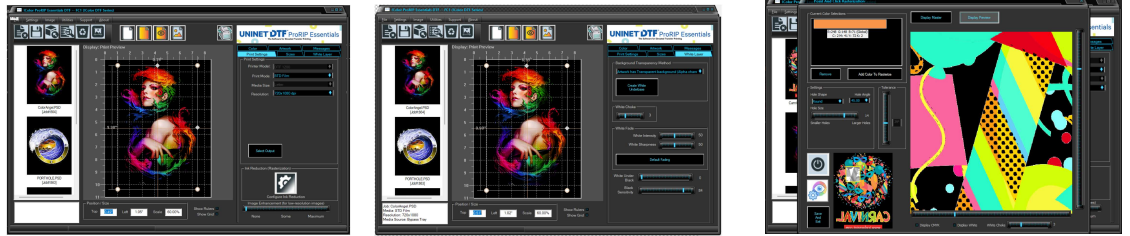
Software

This section outlines most of the RIP versions I've developed as well as other utilities.

iColorProRIP Essentials DTF

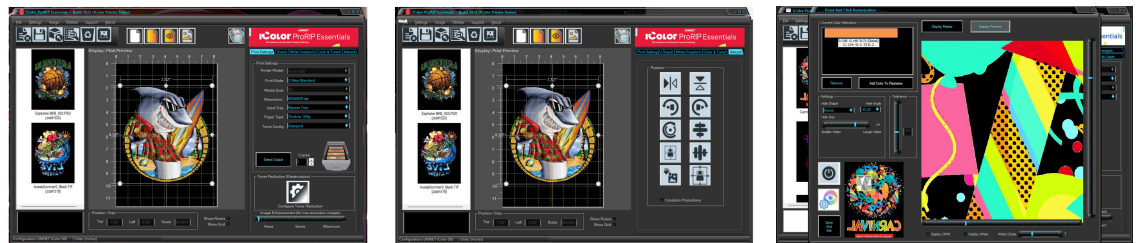
In Beta testing as of this writing (4/5/26)

This version offers all the functionality and ease of use as the toner version (see below). This product is based on the C-Breeze RIP code (see below) and uses the web-based validation system I developed for UNINET.



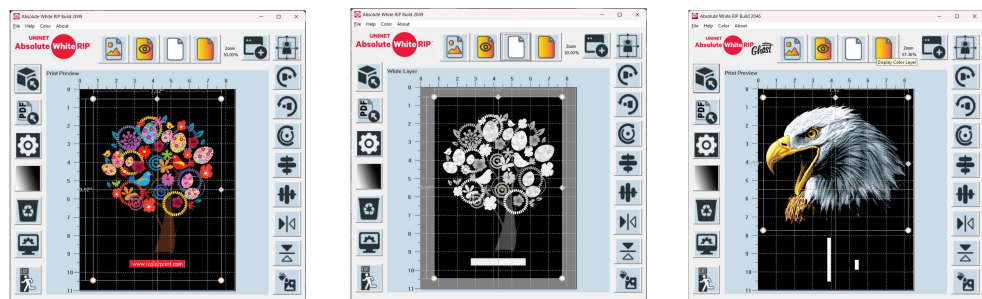
iColorProRIP Essentials

This software has been shipping for a couple of years as of this writing (4/5/26). This version was written for the UNINET white toner printers. Like all the RIPs I developed for UNINET, includes the "rasterization" feature, which essentially reduces the pigment by using halftone screening. The user can select the spot shape, frequency, angle and can be targeted to certain colors or areas. This software is based on the C-Breeze RIP



Absolute White RIP

This software creates a 2-page PDF from an artwork file. One page contains the CMYK layer, the other contains the white layer. There is also a private labeled "Ghost" version for the German distributor. Although there are many similarities, this product has its own core and is not based on the C-Breeze RIP.



Software (cont.)

C-Breeze RIP Series

This is the RIP that all the OEM versions are based on, including the iColorProRIP versions. Below are links to the user's guide and the quick-start page, which contains links to videos. Although the user's guide and videos refer to the SCF2000 version, all the versions have the same or very similar features.

User's Guide: https://www.c-breezerip.com/cb_f2000.htm

Quick-Start: <https://www.c-breezerip.com/quickstartf2000.htm>

Printing Dark Shirts: <https://www.c-breezerip.com/quickstartf2000.htm>

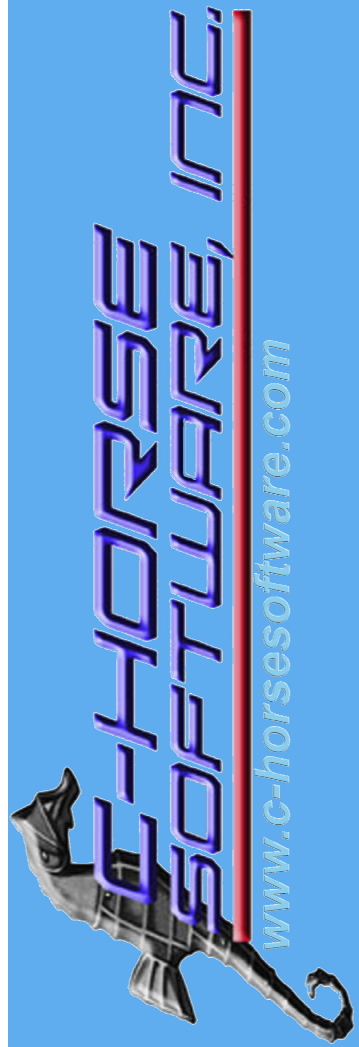
Printing Light Shirts: https://www.youtube.com/watch?v=_a3pf-MU56g

Step And Repeat: <https://www.youtube.com/watch?v=4faE6BHhd0U>



- 1 Distance of edge to top of platen
- 2 Distance of edge to left of platen
- 3 Size ratio of original file
- 4 New job
- 5 Save job
- 6 Import graphic
- 7 Zoom view.
- 8 User settings management

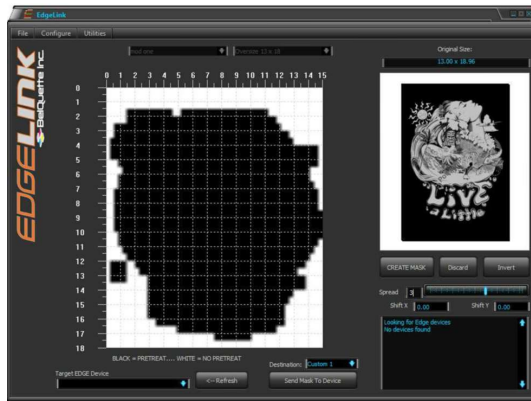
- 9 Delete (artwork from platen or job from printed jobs queue)
- 10 View Select. Selects between the layer views.
- 11 Printed jobs. Double-click to bring back into platen and edit size, position, print settings, white ink, etc.
- 12 Select white-generation method.
- 13 When clicked, generates the white layer



Software (cont.)

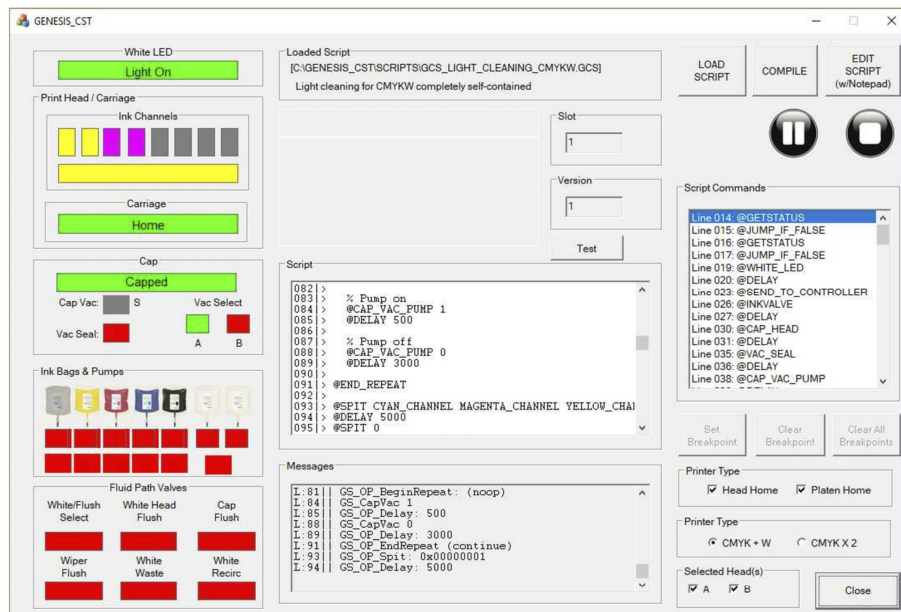
EdgeLink

This is the control software for the Belquette EDGE targeted pre-treat machine. This software creates the mask, and communicate with the device. I also wrote the EDGE's firmware (see hardware section for more). The software would also interact with the RIP. The control/GUI app that runs on the Edge was written in C++ for Windows CE



Genesis CST

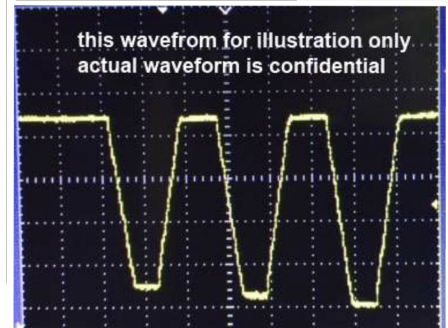
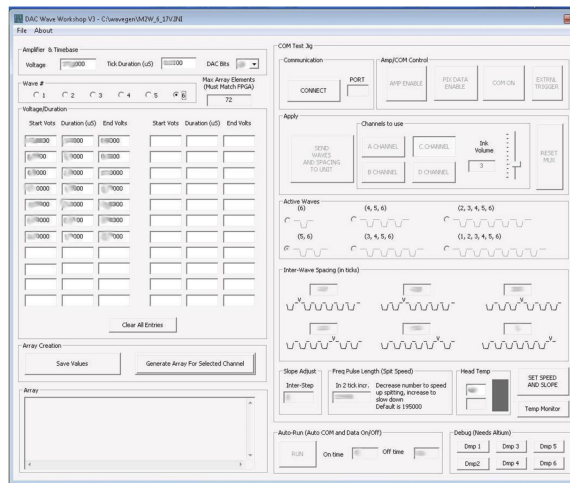
This software was used to program cleaning routines to be used on the Genesis. I developed a proprietary language that the users could use to write scripts using the software. The software would also execute the scripts and show the user what the printer would be doing, using icons and indicators. The user could then tweak the script until they were satisfied. The script would then be uploaded to the Genesis. It would store the scripts and they would be used when cleaning.



Software (cont.)

DAC Wave Workshop

This utility is used to create the COM waveforms for driving heads. It is proprietary and can only be used on C-Horse developed boards. It allows for creation of any shape waveform by allowing the user to specify, with great accuracy, every spline and space. The waveform is downloaded to the Genesis electronics (no longer sold, The Saturn board set (see hardware section) or the Jupiter board (see hardware section)



PowerRIP DTG versions

This was an iProof product. It sold very well in the DTG market. iProof had several firms selling private-labeled versions of this RIP. It provided both in-rip job creation as well as acting as a print serve

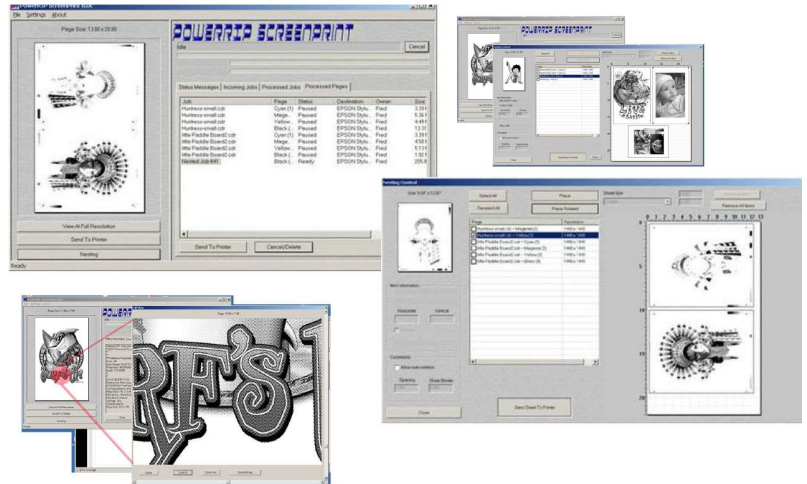
- SWF East (now Coldesi)
- SWF West
- Belquette
- Sawgrass
- Springsun
- Lawson
- Melco



Software (cont)

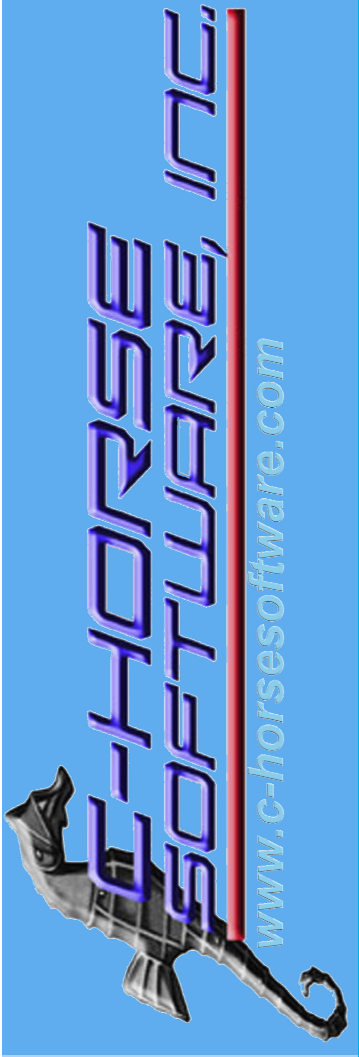
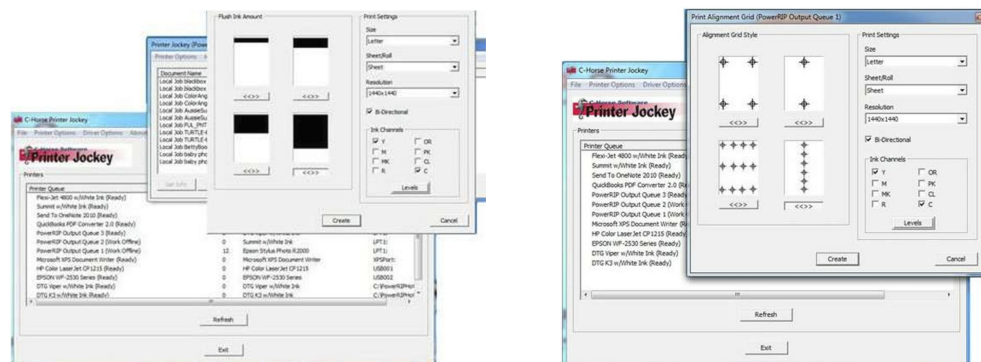
PowerRIP ScreenPrint

This version is a screen printing version which was designed to print on film for burning screen. We also made a direct-to-plate version of this software for Mitsubishi. It featured advanced options like step & repeat and n-up printing.



Printer Jockey Printer Utility

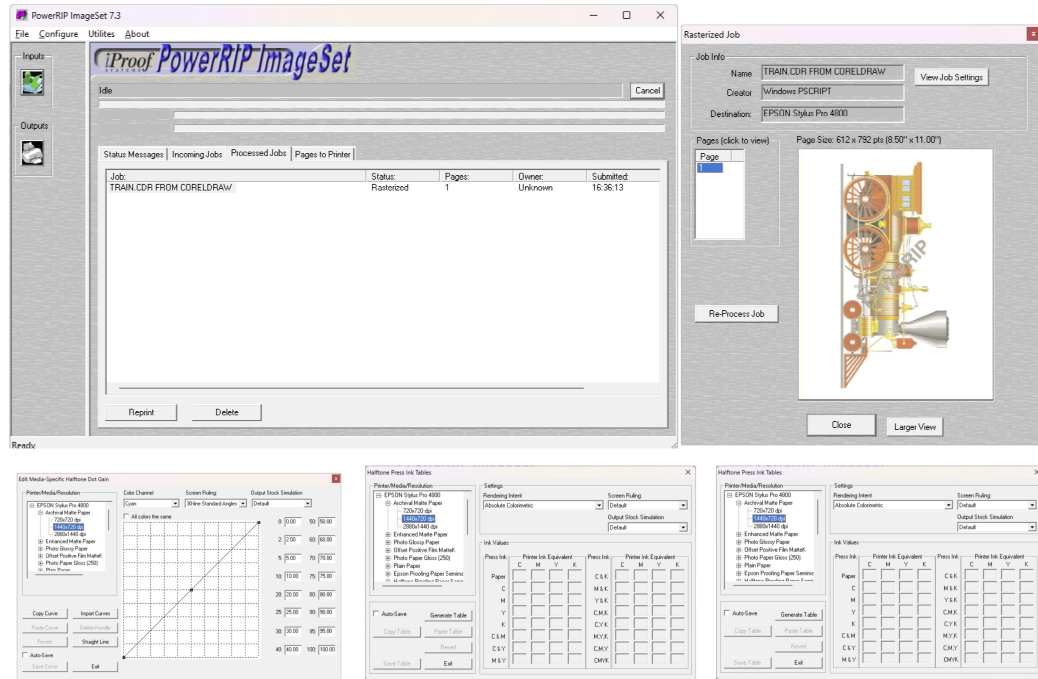
I wrote this utility mostly to help DTG users flush a single channel when clogged. This saved user from wasting ink from channels that were operating properly. It supported all the DTG printers at the time. I sold quite a few of them before the supported printers were discontinued. In addition to the ink flush, it can also be handy for manipulating printers/drivers/ports, doing production runs and timed cleanings. I now offer this utility for free.



Software (cont)

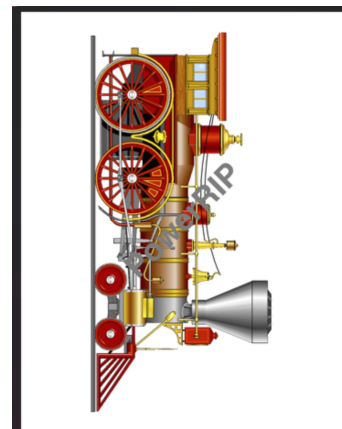
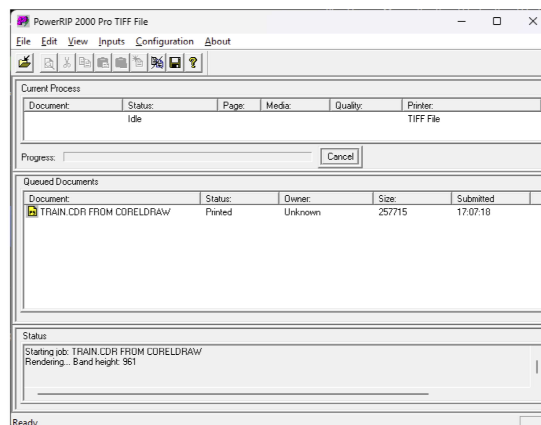
PowerRIP ImageSet and OEM versions

This version was designed for proofing. It contained advanced color features, halftone overrides, density curves, ink tables for accurate press proofing and many other features. It was private labeled for several companies. This was a server product. Users printed to a published name.



PowerRIP 2000

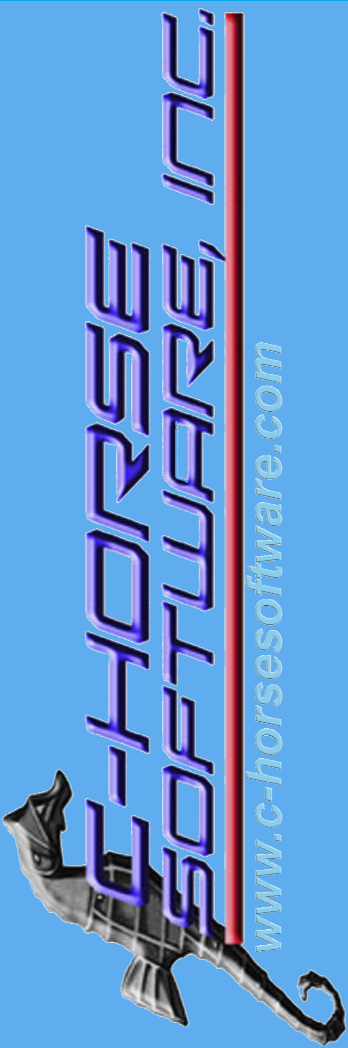
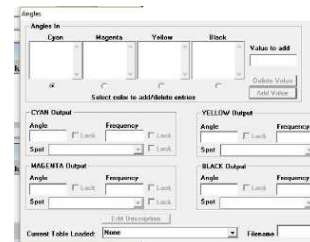
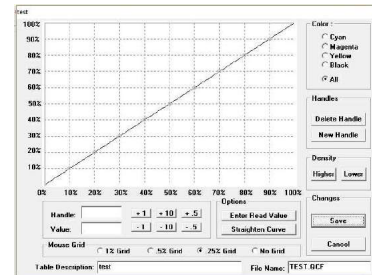
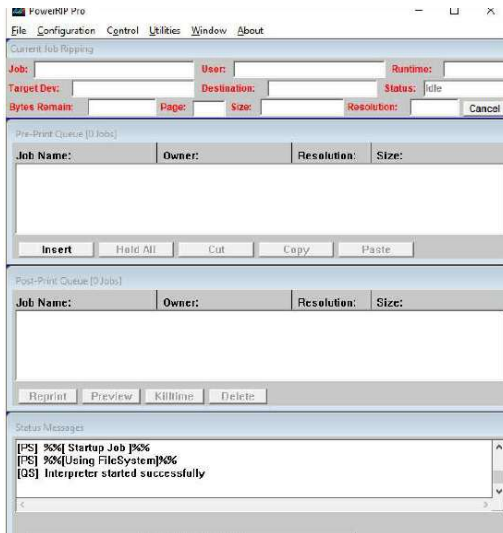
An earlier version of the proofing version. This version, although not as advanced as ImageSet, did well in the proofing market. It could also create PDFs and TIFFs. It was also a server type RIP where users would print to a published name.



Software (cont)

PowerRIP Pro Laser Imagesetter RIP

This version was developed to drive laser Imagesetters such as the ECRM Pelbox, Lino 300 and 500, Varityper, Ulte and Ulte-based devices. It was based on the Adobe CPSI software RIP. I was the first Adobe OEM to multi-thread the CPSI libraries, even though Adobe claimed that CPSI did not support multi-threading. I employed worker threads which called into the libraries to solve the problem. My version ran smoother on Windows NT than any other OEM's CPSI based solution



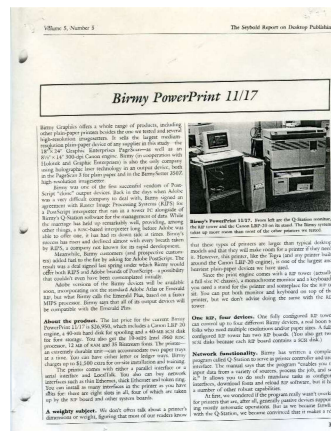
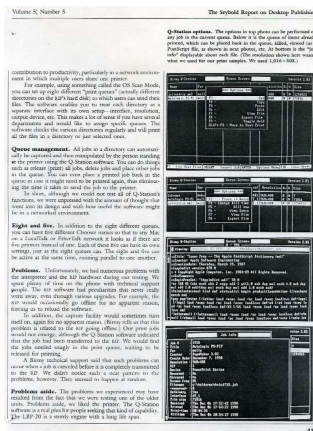
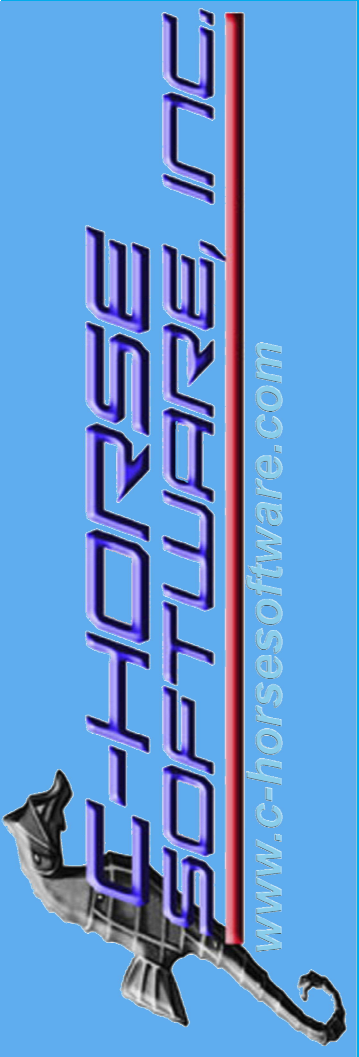
Software (cont)

Queue Station

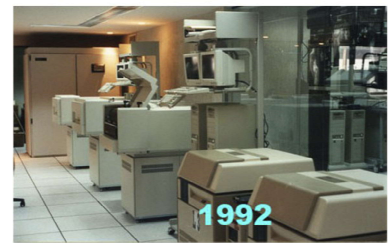
This was Birmy's first foray into the RIP market, back in the 80's. The initial version used a coprocessor board for ripping as the Intel 80286 was a 16-bit processor and not fast enough to rip PostScript. When the 80386 was released, we moved the RIP code to the PC and ran it in extended mode using the Pharlap DOS extender. The Q-Station code was merged with the RIP code. However, the Q-Station code had to remain in 16 bit mode as it used several drivers, such as the EtherTalk boards, etc. while the RIP ran in 32-bit mode. The software utilized the system ticks to switch back and forth in real time between 32 bit and 16 bit modes to RIP and perform queueing, networking in a manner that appeared multi-threaded (DOS never supported multithreading).

I also developed a product named "RIP NET" which supported an unlimited number of RIPs (Q-Station) on a network, hooked up to an unlimited number of imagesetters. This was designed for newspapers. The users would print to any of the RIPs, it would in turn hand it off to an idle RIP, then send it to the first available idle imagesetter. We installed several of these. The Reforma newspaper in Mexico City, was the first newspaper in the world that was totally digital from day 1. Reforma used RIP NET.

Below are pages showing a writeup in the Seybold Report trade magazine (a well respected publication) which covered all the latest digital technology for the Desktop Publishing and Pre-Press markets.



RIPNet® advanced RIP network for newspapers



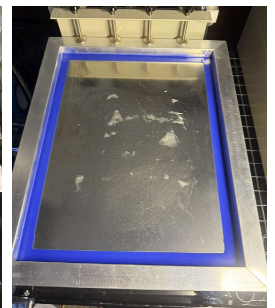
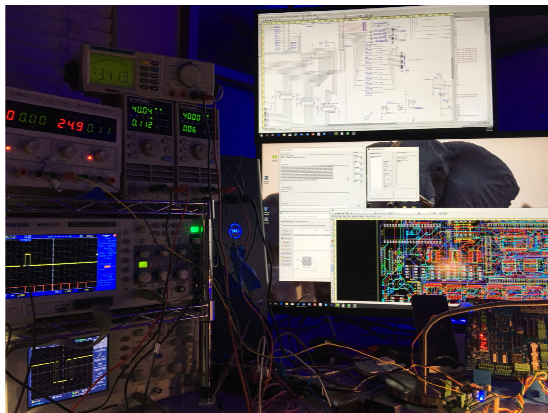
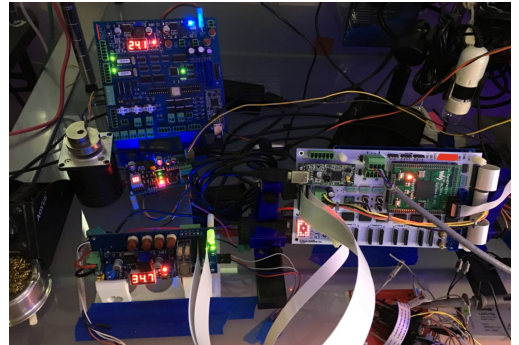
Hardware

This section outlines some of the hardware products I've developed. Some projects cannot be listed due to strict NDA's

Development Tools and Resources

A partial list of the test equipment, etc. Able to create prototype boards of any size, including multi-layer boards with surface mount devices and devices on both top and bottom. Along with a full complement of every-day tools, I also have in my labs:

- Soldering equipment, from coarse to super-fine tips
- De-Soldering equipment
- Surface mount ovens
- High speed oscilloscopes
- Logic analyzers.
- Rework station (hot air, etc.)
- High resolution 3D printers
- 20" x 40" CNC machine. Plastic and soft metals
- High speed Windows workstations
- Circuit board design software
- C/C++ Development Tools.
- FPGA Development tools and programmers.
- Embedded C compilers for Microchip PIC microcontrollers.
- Development tools for ARM based processors.

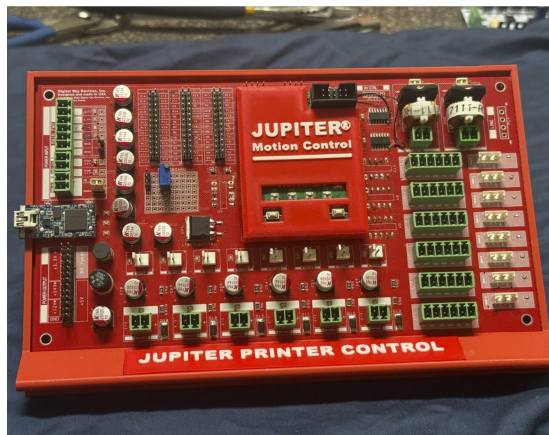
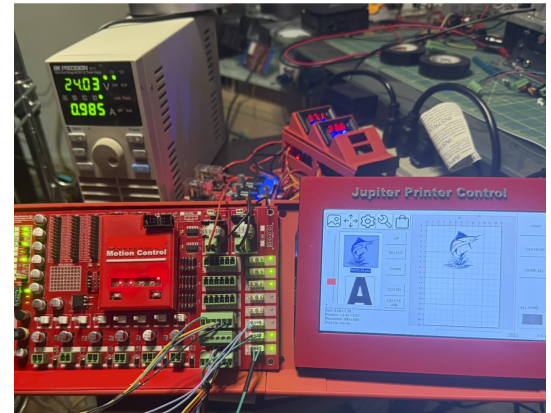
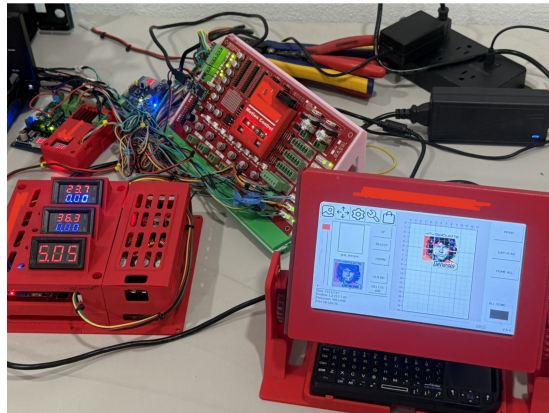


Hardware (cont)

Jupiter Printer Control Board Set

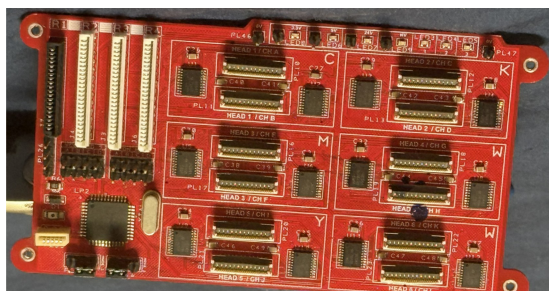
This board set is the latest design and it is currently in development. I broke away from the "all in one" design of the Saturn II and made this board set modular. Separate boards for motion control, head control, power, etc. As with the Saturn, this board set is mostly mean to be used as a starting point for a custom boards set. OEMs usually have different requirements as to sensors, I/O ports, number of motion axis, etc.

As with previous C-Horse printer control solutions, it employs a mini-PC running Windows Embedded and a touch screen to act as the printer's controller. The head control and motion control boards are driven with high speed USB.



The motion control board has the following I/O:

- 1 x Mini USB (commands in)
- 2 x bi-directional peristaltic pump drivers
- 6 x servo/stepper axis
- 8 x sensor ports (5V signal)
- 8 x 5V DC peripheral outputs
- 8 x 24V DC peripheral outputs
- 4 x 40 pin sockets with spare I/O
- 1 breadboard area



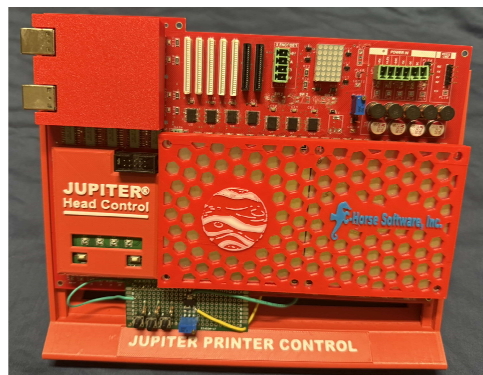
The Reah board is the head breakout board. It connects to the head driver board with ribbon cables, the heads then connect to it with individual ribbon cables. This board rides along in the carriage.

Hardware (cont)

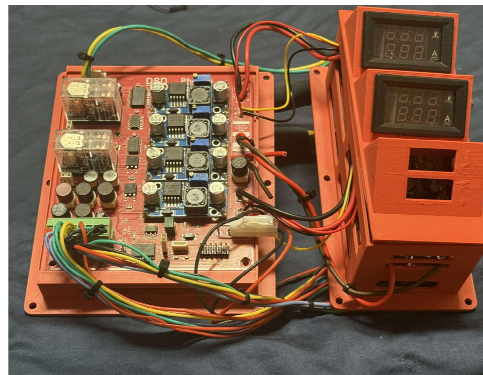
Jupiter Printer Control Board Set (cont.)

This board set is the latest design and it is currently in development. I broke away from the "all in one" design of the Saturn II and made this board set modular. Separate boards for motion control, head control, power, etc. As with the Saturn, this board set is mostly mean to be used as a starting point for a custom boards set. OEMs usually have different requirements as to sensors, I/O ports, number of motion axis, etc.

As with previous C-Horse printer control solutions, it employs a mini-PC running Windows Embedded and a touch screen to act as the printer's controller. The head control and motion control boards are driven with high speed USB.



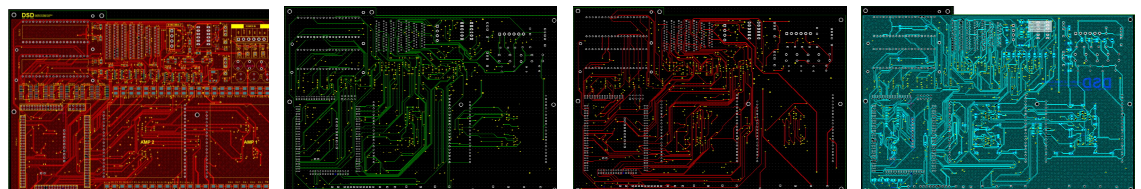
The head control board supplies the analog COM wave (head waveform) as well as the digital signals to the heads via the Reah break out board (previous page) this version supports up to 8 Ricoh GH2220 or compatible heads.



The Phobos power supply system consists of an off-the-shelf step-up transformer which converts +24V DC to +37V DC. The Phobos main board supplies:

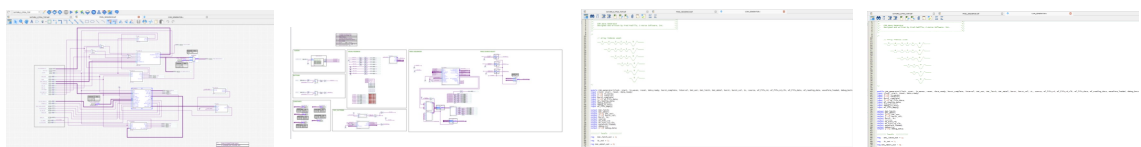
- +5V DC Constant
- +3.3V DC Constant
- +24V DC Constant
- +24V DC Switched
- +37V DC Switched
- -5V DC Switched

The head control board uses a signal port to cause the Phobos to turn on the switched volt-ages used for printing.



The majority of C-Horse circuit boards are 4-layers with a mixture of through-hole and surface mount devices. Above are the 4 layers that make up the head control board.

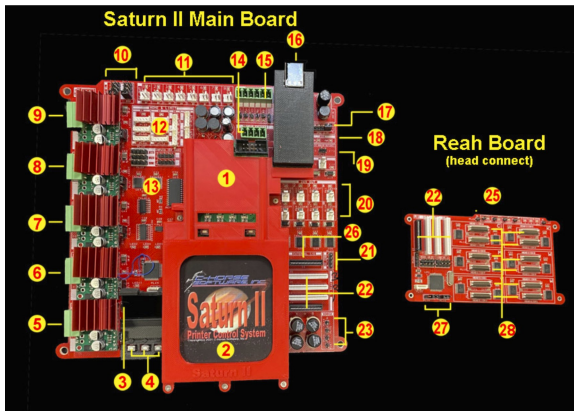
FPGA / Verilog



Hardware (cont)

Saturn II Printer Control Board

Unlike the Jupiter board set, the Saturn II board combines both the motion control, sensors, I/O and head control on the same board. It also uses the Phobos power supply system and the Reah head break-out board. Although no OEM took the board "as is", it was the starting point for several projects. I've discontinued the Saturn II as the motion control was a bit limited due to the number of logic elements in the FPGA module. It supported 5 axis but they were controlled by an external processor instead of the FPGA.



1. FPGA module. Powerful but expensive. Off-the-shelf to keep manufacturing costs down.
2. Amplifier. This device supplies the high-voltage pulses necessary to drive the print heads. As with the FPGA, it is an off-the-shelf module to keep manufacturing cost down.
3. Tiny OLED monitor which can be used for status messages or debugging. If you don't need it, you don't have to include it in your version.
4. 3 buttons which can be configured for any function (debugging, special features, special modes, etc.)

5. Stepper driver for X axis. Configurable step resolution and current.
6. Stepper driver for Y axis. Configurable step resolution and current.
7. Stepper driver for Z axis. Configurable step resolution and current.
8. Stepper driver for Cap axis. Configurable step resolution and current.
9. Stepper driver for Aux axis. Configurable step resolution and current.
10. Configuration for Aux stepper driver (it can follow another stepper driver but in reverse, for example)
11. 5V output signals for peripherals.
12. Home and limit sensors for X, Y, Z and Cap axis.
13. Eight 5V inputs for sensors, etc.
14. Connector for X linear encoder.
15. Power input. The voltages are supplied by the Phobos IV board (not shown)
16. High speed USB for pixel data and low-speed USB for commands.
17. 4 extra IO ports which can be programmed for anything.
18. USB status LED indicators.
19. +24V supply for 24V outputs. Right below is the 5V analog output (0 - 5V) which can be used to control things like the intensity of UV LEDs.
20. Eight 24V switchable outputs.
21. High-voltage request bus. Used to request the Phobos board that the amp voltages turn on.
22. FFC sockets for ribbon cables which connect the main board to the Reah head-breakout board. It carries the digital and analog signals to the heads.
23. Test points for the COM waveforms.
25. Reah head-breakout board. Connects to the heads. By having different Reah board versions, the Saturn II can be used (with different firmware, of course) to drive different types of head. The Reah has a built-in processor which can be used to drive servos, valves, lights, etc. which might be on the carriage.
26. Connector for external I/O board (not shown)
27. Head mode jumpers. The GH2220 can be driven in different modes. These jumpers allow selecting which.
28. Head connectors. 12 channels maximum for this version (Ricoh GH2220).

Hardware (cont)

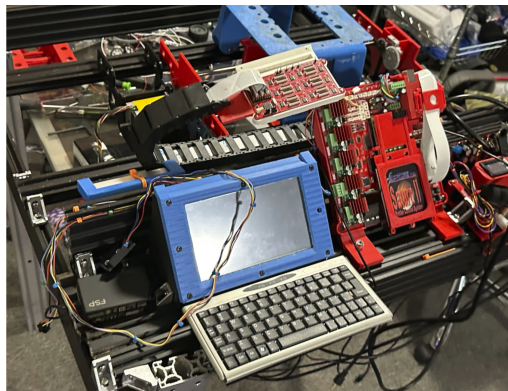
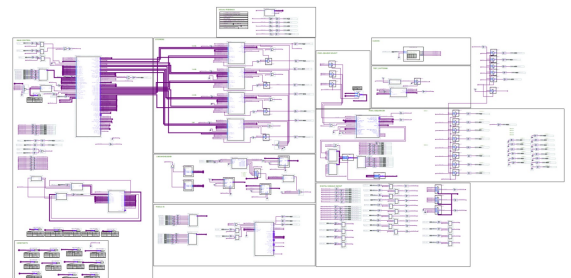
Saturn II Printer Control Board (cont.)

I've discontinued the Saturn II as the motion control was a bit limited due to the number of logic elements in the FPGA module. It supported 5 axis but they were controlled by an external processor instead of the FPGA.

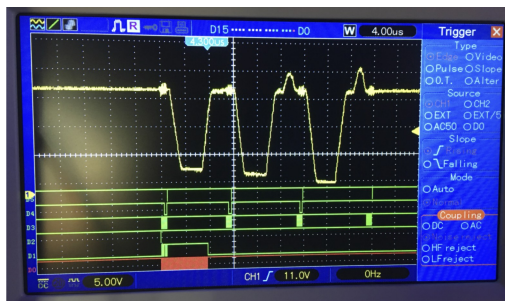


The Saturn II contains a small OLED monitor for debugging, etc. It is there mostly for development purposes but it can also be used in an end-product for various tasks.

FPGA Project, top sheet:



As I mentioned earlier, the Saturn was not adopted "as is" by OEMs, Modified versions of it were, however. Also, C-Horse built several experimental printers to demo the board. Pictured here is the early stages of an experimental bottle printer which would spin the cylindrical media underneath a stationary head. The carriage was articulated so it could "wrap around" the media.



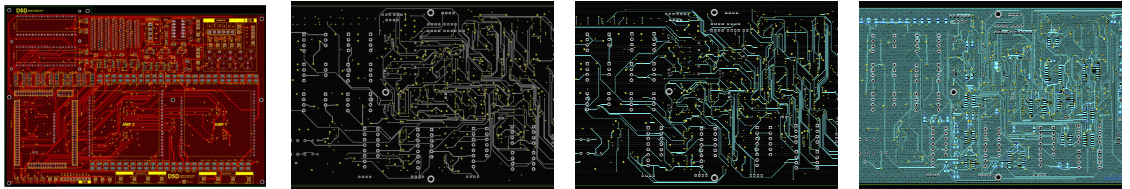
Here we can see a waveform I experimented with. As that is a mixed signal oscilloscope, you can see the digital signal lines below and the analog COM wave above. Ultimately, a modified version of that wave was adopted.

Hardware (cont)

Polyprint Pandora Motion Ctrl. And Peripheral I/O Board

I was contracted to design and build a peripheral I/O board for a new Polyprint printer. After the I/O board we were going to design and build a motion control and head control set. Unfortunately, COVID 19 hit and that project (along with others) was scrapped. The product never shipped.

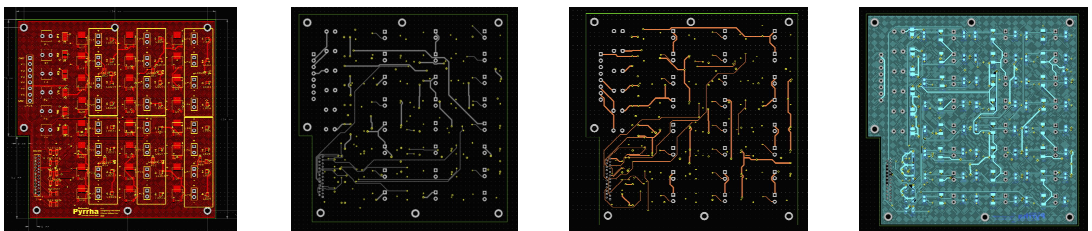
Main Board



The main motion control board (Pandora) supports:

- 10 x Stepper / Servo signals
- 12 x 24V peripheral outputs
- 16 x Sensor input ports. 5V or 24V
- 1 x USB in command channel.

Output Extender



The Pandora board set also included an optional output extender which added an additional 6 24V outputs

Hardware (cont)

UV Printer for European Company (name withheld)

I was contracted to create the following for a new UV printer:

- All electronics
- Ink Delivery
- Firmware
- FPGA
- RIP

Basically everything except the chassis and motors. The printer is based on the Ricoh GH2220 head. The board set consisted of The head control board, the motion control board, the amplifier board (to amplify the waveform) and the power board. Like my other designs, this printer was controlled by a mini-PC running Embedded Windows. This system also used the Phobos power supply.

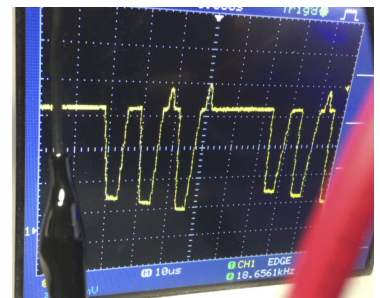
You can see a video of an early prototype printing here:
<https://www.youtube.com/watch?v=vHz7jhJ2xSA>



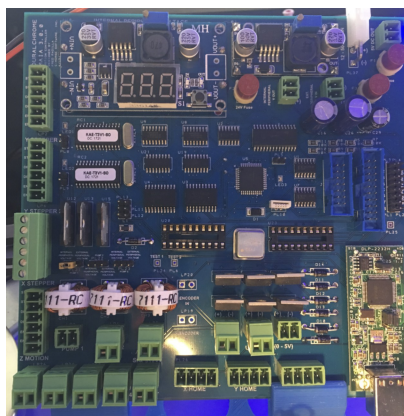
In this implementation, the ink reservoirs travelled with the carriage. This eliminated the need for ink lines.



The complete unit. The maximum platen size is bigger than depicted.



This waveform gave me the best quality for this particular UV ink.

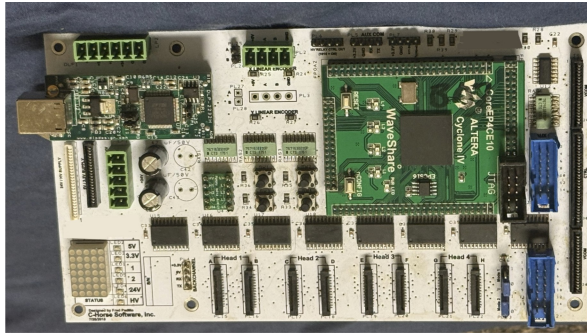


The motion control board. It has the following on board:

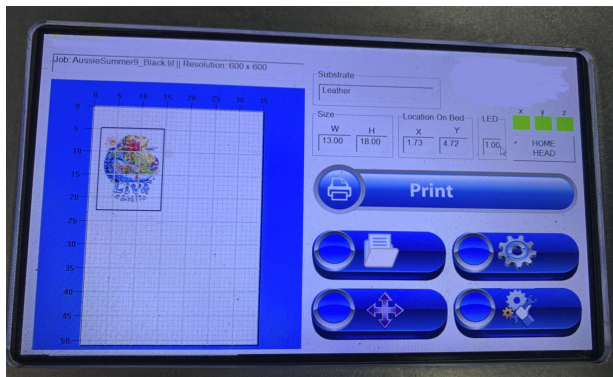
- USB (control)
- X Axis signals & Home sensor
- Y Axis signals & Home sensor
- Z Axis signals & Home sensor
- 3 x bidirectional peristaltic pump drivers
- 4 x 24V general purpose outputs
- 2 x UV LED drivers
- 2 x auxiliary connectors for daughter-

Hardware (cont)

UV Printer (cont.)



The head driver board. This board generated all the analog (waveform and digital head signals). This version supports 4 GH2220 heads. It uses the Arche amplifier (not shown) to amplify the waveform.



The GUI for the printer's controller (mini PC running Windows Embedded) the software communicates with the motion control board, the head control board and the network.



Test printer being put through its paces at a local ink manufacturer. There is still some banding in this version as one of the gears was not quite right but my electronics and software performed flawlessly.

Hardware (cont)

Hermes / Nova stepper controller

I developed a user programmable stepper controller which would give the OEM complete control over the:

- Steps per revolution
- Running Current
- Idle Current
- Active/Inactive signal

In addition, the Hermes/Nova controller allows the user to bench-test the unit without a microprocessor installed. When initiated in bench mode (internal mode), the user can spin a connected stepper in both directions, thus verifying the board and the stepper work properly before installing them. The Nova/Hermes configuration app (see below) is used to set the above mentioned parameters.

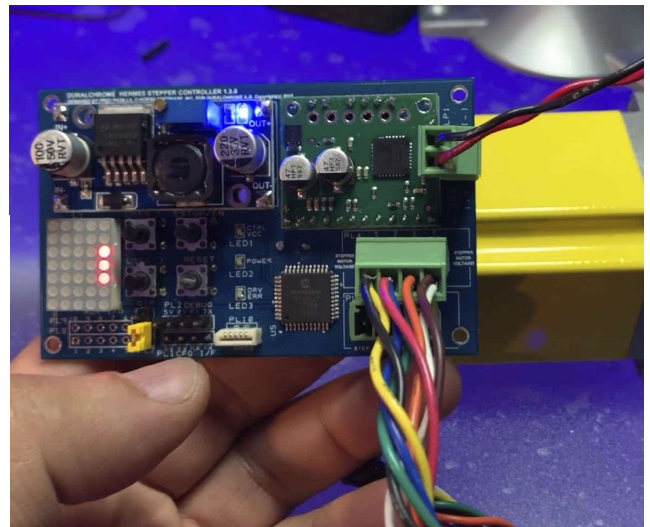
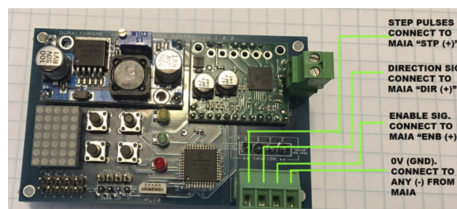
You can see a video of the Hermes in bench mode here:

https://youtu.be/_EJN4YmHnXg

The only difference between the Hermes and Nova is that the Hermes has an on-board +5V DC converter while the Nova accepts +5V from an external source.

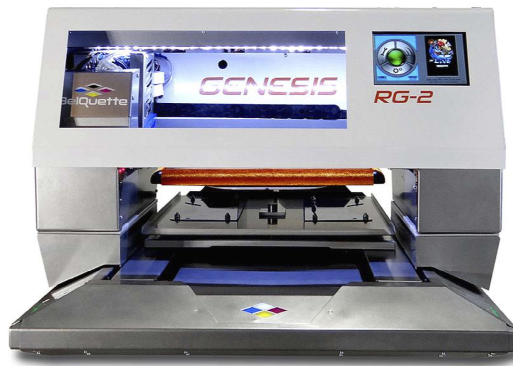


The Hermes/Nova configuration application. Allows the user to set the running current, hold current, steps per resolution and what happens when you disable the controller (disable completely or go in to hold mode).



Hardware (cont)

GENESIS Direct To Garment Printer



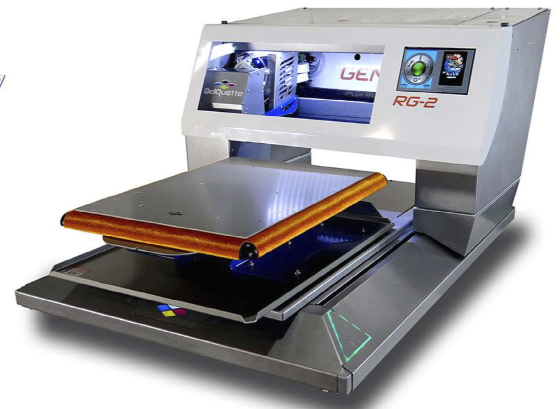
Several years in the making, This printer contained many innovations and industry firsts. It was based on two Ricoh Gen5 heads. One 4-channel, 4 colors for CMYK and one 4 channel, 2 color for white. The printer fit through a normal door even though the print area was 18" x 20". The it was approximately twice as fast as the F2000/F2100

with equal quality. It was easy to network and had very intuitive features. It also came with an IoT function. The printer would communicate with a main server to report print jobs, ink usage, time, errors, etc. The user would then be able to log on to the server and, using the printer's serial number, retrieve the information to help in job management, billing, maintenance, etc. I developed all the motion control, FPGA, Firmware, Software and RIP.

You can see a video of the Genesis printing here:

<https://youtu.be/nH0VcYNkry8>

<https://youtube.com/shorts/mQs2oxXx1-M?>



Like the Jupiter and Saturn solutions, the heart of the job processing, networking and user interface was a mini PC running Windows Embedded. The esthetically pleasing GUI along with the stylish case, made the Genesis not only one of the fastest, feature rich and innovative DTG printers, It was also a beautiful machine.

C-HORSE SOFTWARE, INC.
www.c-horsesoftware.com

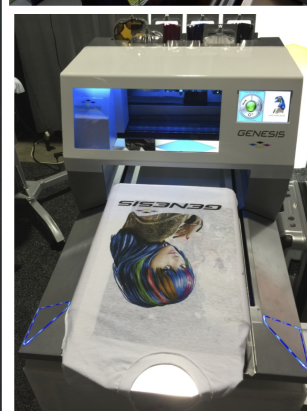
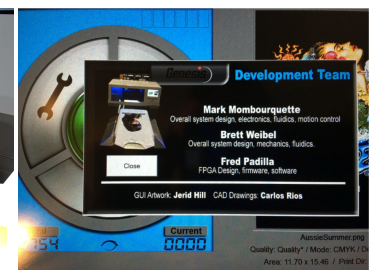
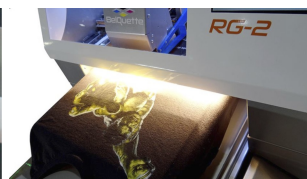
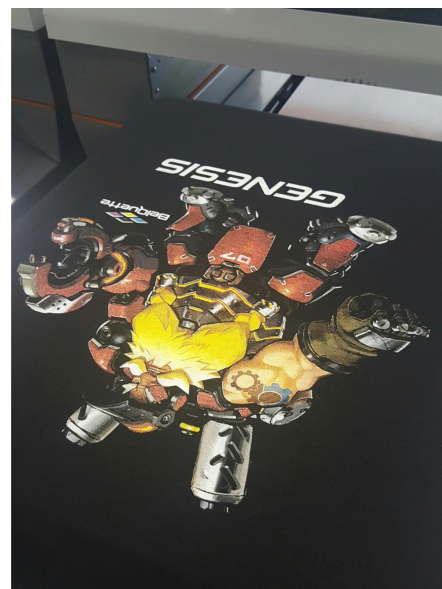
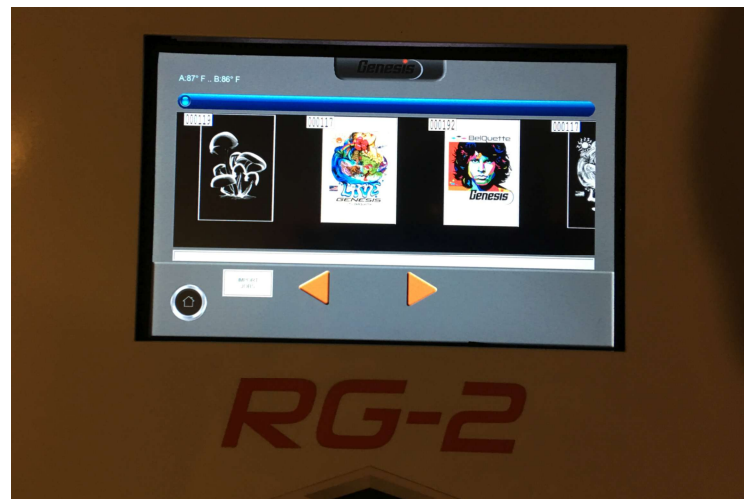
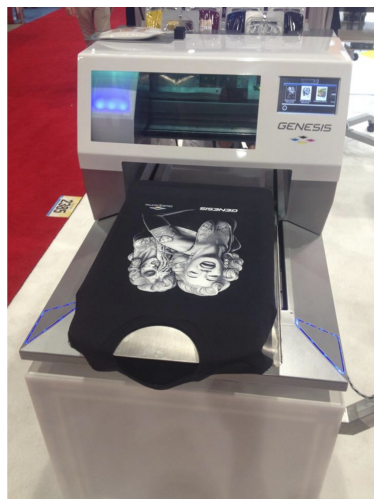


Hardware (cont)

GENESIS Direct To Garment Printer (cont.)

C-HORSE SOFTWARE, INC.

www.c-horsesoftware.com



Trade show exhibit, "Genesis farm" at Cafe Press, dark shirt print at trade show, The "Jobs" screen on the printer's GUI. A CMYKW print, a CMYK print. A super-saturating 6-wave waveform I developed for the white ink channels. It had incredible density.

Hardware (cont)

Jet Genie Head Cleaner



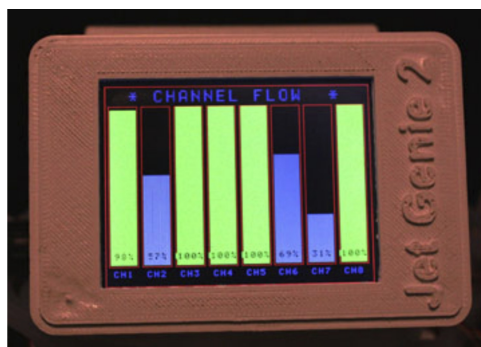
There were no viable automatic head cleaners on the market back then so I developed a cleaner that would clean an Epson DX5 head. I discovered that pushing cleaning fluid through the ink nipple and out the nozzles can actually push particles deeper into the head which will get stuck on the inside of the nozzle plate. A better solution, in my opinion, is to immerse the nozzle plate in cleaning solution and pull from the nipple so that it sucks the cleaning fluid up through the nozzle plate, dislodging particles which will then exit the head via

the much wider ink feed nipple.

The head is mounted on a "raft" which floats in a bath of cleaning solution at a height where only the nozzle plate is submerged. It then applies suction to a channel or multiple channels in a preset pattern. There is also a filter and pump which keeps the cleaning fluid bath particle free.

The unit has a color touch screen and can offer audible signals. The majority of the unit was 3D printed.

You can see a video of Jet Genie here:
<https://youtu.be/ZqkMn6sDYB4>



The unit uses pressure sensors to measure cleaner flow through the nozzles. It uses that data to concentrate on the most clogged nozzles first.

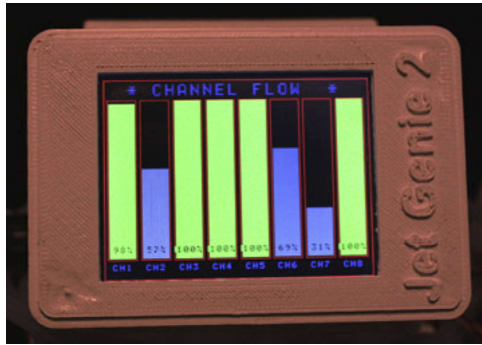


Using a liquid-proof speaker, the unit emits a high-frequency tone directly up the nozzles. The high amplitude and high frequency help break up clogs. I don't use ultrasound as that can delaminate the heads.



Hardware (cont)

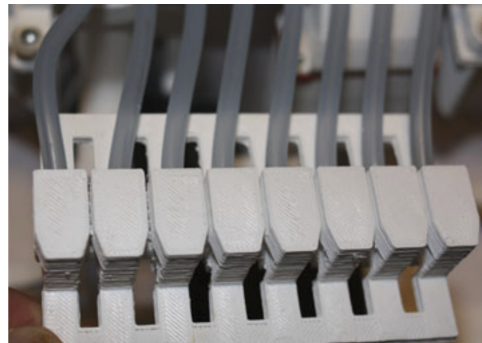
Jet Genie Head Cleaner (cont.)



The unit uses pressure sensors to measure cleaner flow through the nozzles. It uses that data to concentrate on the most clogged nozzles first.



The user can check the nozzles to determine the state of the head, then select which nozzles to clean.



The manifold connector attaches to the head, one line per ink nipple. This allows for treating all 8 channels independently.

Hardware (cont)

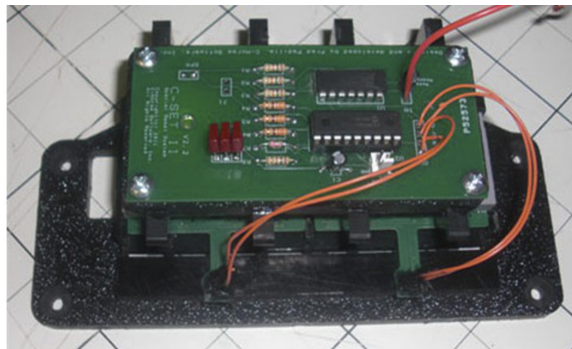
C-Set Ink Chip Resetter (cont.)

A few years back, several companies created DTG printers from desktop Epson models 1800 and 1900. They employed aftermarket ink delivery systems which included "resettable chips". This enabled the user to reset the chips back to full when the printer considered the cartridge to be empty. This meant that the user had to reset the chips every 3 shirts or so. This could take a bit of time as the printer did a cleaning every time the chips were reset.

I was working on an "auto-resetter" board that would do that automatically. When the printer thought it was out of ink, the board would short the switch leads together using an IC and a simple PIC processor.

I found (quite by accident) that because of the IC's I selected, while simulating the reset, the ground (0V) was not true 0 but hovering just above. Apparently this confused the Epson printers somehow and saw the cartridge as full, but SKIPPED the cleaning cycle. This meant that the printer would not stop every 3 shirts for a few minutes. This translated to much higher production per printer.

Soon I was selling as many as I could make to the US and Europe DTG printers manufacturers that used the R1800 and R1900. I sold many hundreds of units.



Hardware (cont)

JRC (Job Release Controller)



My first hardware product developed for the DTG industry.

There was a DTG printer being sold by SWF East (now Coldesi) based on the Epson 4800 named the Viper. I added support that printer to our DTF RIP Pro software. The printer had a conveyor belt which moved the platen in and out of the printer. The way the printer worked, however, you had to make sure that the printer was in the home position before you send it data or it would malfunction (sometimes it would launch the platen onto the floor). This made production runs impossible on our RIP because the jobs would be queued up and read and the printer would take them only when ready. This did not work on this printer. It would try to start printing the next job before it was home and ready.

I developed the JRC which is a very simple device. It consisted of a USB-to-RS232 converter. A small circuit board with a microprocessor (PIC18) and a large button. When the button was pressed, the JRC would send a command via RS232 (serial port) to the RIP, telling it to release the next job. This allowed the user to stay by the printer during a production run and just press the button once per print cycle after the printer was home, loaded with a shirt and ready. I sold hundreds of these as they were shipped with every printer.



Fred Padilla

PRESIDENT, C-HORSE SOFTWARE, INC.

chorsesoftware97@gmail.com

321.508.7841