

**Scott G. Henion**  
**Independent Contractor / Consultant**

<u>KEY SKILLS</u>	<u>EXPERIENCE</u>
Consulting	15 years
Hardware Design	25 years
Software Design	25 years
Embedded	25 Years
Linux	14 Years
Networking	20 years
Linux Router	12 years
Wireless	1 year
C Language	20 years
Visual C++/MFC	15 years

**EDUCATION**

Florida Institute of Technology (now Florida Tech.)  
BS, Electrical Engineering  
BS, Computer Engineering  
Graduated with honors in both degrees.  
Member IEEE and the IEEE Consultants Database.

**Embedded targets:**

Intel: 80186, Zilog Z80, Motorola/Freescale:68705, 68HC11, 680x0, DSP56xxx, Coldfire (Netburner), Microchip PIC16xx and PIC32, Arm (Atmel), Rabbit Semiconductor (Digi) R2000/R3000.

**Development tools:**

Dynamic C, Nuhau, Archimedes, GCC, MS C 6.x, Visual C++/MFC, Sourcesafe, Softtools ANSI C, SDCC, MPLAB (Microchip), Link and Locate (Paradigm), Motorola CLAS DSP development tools.

**Work Experience:**

**Alpha Embedded** – Programmer 7/11-9-11

Contact work to develop a flight simulator application. Written in VisualC++/MFC and ANSI C. Displayed control panel images from a simulated helicopter. Then allowed toggling switches and breakers with visual feedback. Also simulated aircraft position, displayed in various coordinate systems. A user console allowed viewing status, simulating errors and editing display. Also brought up additional simulation programs that it communicated via named pipes and shared memory.

**Scott Henion Designs** – Stone Mountain, Georgia. 8/96 to present, Independent Consultant/Proprietor 1996 to present.

As an independent consultant, concentrated on embedded systems hardware and software design. Of the numerous, concurrent projects, the significant projects include:

**Stevens Systems SLS2200 language lab.**

Completely redesigned a old lab console that used a large panel with switches. New design is microprocessor-controlled and uses a PC for display and user input. Embedded controller handles routing audio for up to 64 students and 12 audio sources. PC applications written in VisualC++/MFC display a simulated console and switches. Lots of new features included saving configurations, toolbars and exporting class lists to Excel.

**Remote Firmware update, Rabbit (Z-Word) Systems:**

Network and serial firmware update on Rabbit-based systems. Library licensed to hundreds of developers, probably 50k or more units in the field. Used a UDP/IP stack written from scratch for small size. Written in ANSI C and assembly. Code adapts to most of Rabbit modules and boards with one executable. Entire application and network stack fit in a 17k executable. An “Hello World” app using standard tools is 30k.

**ANSI C Compiler Support:**

Helped development of Softtools ANSI C for Z-World boards. Ported TCP/IP stack and WiFi stack to ANSI from a non-ANSI C compiler. Have been contracted to provide technical support for the compiler and numerous libraries (LCD Display, network, WiFi, SPI, I2C, graphics).

**CoExec: Cooperative task switcher**

Used on many of my software projects where uCOS was too big and slow or not worth the cost. Library used by many others to allow cooperative tasking for easy programming on systems that did not include a RTOS. Has proven effective even in hard, real-time applications by using interrupts for real-time data and tasks to handle less time-critical functions. Has been available free to others for several years.

**Project: Testing Application:**

Developed a board with a small processor and a DSP (FreeScale 56K) processor for recording audio for language proficiency tests to an embedded IDE drive. Records and plays 40 channels in real time to disk. DSP handles all audio data (320,000 interrupts/sec) real time. General-purpose CPU handles network and file system and runs a custom real-time exec (CoExec). The file system was custom-written to allow fast access and low overhead.

Developed a PC application written in Visual C++ that reads the tests from the boards and displays a test matrix of tasks and grade levels. An examiner can listen to individual students, assign grades and export results or an Excel spreadsheet. Tests can also be exported to MP3 files to be sent to state boards.

**Project: Landslide measurement system:**

Developed firmware for a battery-powered “Smart Rock” for USGS. This measured acceleration through an A/D converter and recorded results on a Secure Digital Card in real-time. Used a small, low-power processor module. Sample rate was 1000 to 8000 samples/sec. Code written in ANSI C using CoExec task switching. Similar project to measure gasses in lava tubes.

**Project: Transcription Interface, Dictation System**

Developed a PC-based sound and telephone card that includes a local CPU and DSP for remote access to a proprietary voice recorder. Design included overall architecture, initial hardware specification and all embedded code for both the CPU and DSP processors.

Embedded real-time software development was done using C and assembler code (80186) and extensive DSP assembler code (Motorola 56002.) This software included card start up, audio playback and recording, messaging and extensive diagnostics. The DSP code included DTMF generation and detection, audio compression (ADPCM, CVSD, PCM, DVI) speed control, modem encoding and decoding, synchronous serial communications, and numerous digital filters. Both the main processor and DSP ran custom real-time OS's written by me. Real-time handled in interrupts.

To support the PC sound card, I developed several Visual C++/MFC programs including console, dialog and frame window applications. These programs included a file I/O driver, a setup program and several diagnostic and debugging tools.

**Project: Auto Monitor, Language Lab**

Developed an embedded smart controller for a language lab system. The language lab system was hard-wired for a specific set of features and did not save any user settings when powered down. The controller monitored the keys and LED's of the system (56 switches), and provided additional functionality. Functions included, saving the current settings, automatic stepping through a programmed set of switches, and a test mode that operated each switch and LED in a specific order for automated system testing. Design included complete hardware and software and also included a built-in programmer for the EPROM CPU (Motorola 68HC711).

**Project: Conferencing Board, Language Lab**

A conferencing system for the above language lab. This board provided eight audio conference groups for any of the 48 audio channels. This system used a single chip CPU (PIC16C71) with

custom hardware, and software. Software was written in assembly language. Provided PCB design and layout along with all documentation for manufacturing.

**Project: Event Monitor, Drive-through system:**

Designed a small board that read various relay inputs and reported the information to a PC via a serial port. This board used a small PIC processor and some custom firmware to emulate a serial port. Board received loop-detect inputs from a drive-through sensors and reported events to a PC. The PC was used to manage user performance (drive-through timing.) Hardware, software and prototype built and tested in 3 days.

A second-generation board was developed that supported a TCP/IP connection and outputted real-time data to large LED displays. This feeds a Windows service written in Visual C++ that records timings in a database and provides logs and real-time statistics in XML.

**Previous Employment**

**Lanier WorldWide**, Tucker, Georgia.  
6/88 through 9/96, PRINCIPAL ENGINEER

Developed two generations of centralized dictation systems. These systems are multi-processor, voice processing systems using custom-designed hardware within a PC-based system. Work included hardware, software and system-level design. Was involved with the entire design process from requirements through design, implementation, manufacturing, documentation and support.

Hardware design included microprocessor, logic, DSP and analog design. Developed several DSP systems for audio and telephone processing. These systems involved multiple DSP processors controlled by a general-purpose CPU. Also designed some of the system peripherals including power supplies, disk controllers, telephone access and input devices.

Software design included real-time software for embedded CPU's and DSP processors. Software was written in C and assembly languages. Advanced the design of an in-house real-time embedded operating system by adding new features and extensive diagnostic capabilities including an on-line debugger. Developed DSP algorithms for voice compression, DTMF generation and detection, communications, modem and echo canceling. DSP coding included FIR and IIR filters, adaptive filters and software emulation of hardware.

Developed design tools and systems for CAD-based hardware design. This included network design, implementation and support. Wrote many utilities for system integration including file translators, print utilities, and design verification tools.

Patents:       Awarded patent #5,519,808 (one of three named inventors.)  
                  "Transcription Interface For a Word Processing Station."

Also named on patent for redundant power-supply design.

**Hetra Computers and Communications**, Sebastian, Florida 6/84 through 6/88, ASSOCIATE ENGINEER

Designed the hardware and software for a 68000 CPU emulator. This design included complete hardware emulation along with firmware for target debugging, download and disassembly.

Designed the CPU (68000,) ECC memory, font storage, backplane and graphics subsystems for a 24ppm laser printer controller. Development software included extensive debugging, diagnostic and manufacturing test programs (68000 assembly.)

Developed several Z80-based communications products. These boards were use for high-speed communication using HDLC or SDLC protocols. Developed BIOS drivers for serial I/O and SCSI (SASI) devices. Wrote board and system-level diagnostics for manufacturing and field use.

Designed a 68000-based communications board for the PC architecture. This board included high-speed serial communications controllers along with an Ethernet controller. The board provided support for high-speed serial interfaces including SDLC, HDLC, Multileaver, and X-25

### **TECHNICAL SUMMARY**

*Development Languages:* Assembly, C, Visual C++/MFC, Pascal, Fortran

*Embedded Operating systems:* Proprietary, UCOS, MicroC, OS/2

*Hardware tools:* PALASM, CUPL, ABEL, Xylinix, PCAD, SPICE, emulators, logic analyzers

*Software tools:* IARC, GNU, Microsoft, Archemedes, 2500AD, Nohau, MicroTek

Visual C++, SourceSafe, PVCS, Borland Version Tracker

Professional Subscription to MSDN.

*Platforms:* Windows 95/98, Windows NT 4.0/2000, Linux, Novell, OS/2, CP/M

*Communications:* Serial RS-232, 422 423, Synchronous (i.e. HDLC, SDLC, TDM, SPI, I<sup>2</sup>C), ARIN429, 1883, named pipes, sockets, anonymous pipes, TCP/IP, UDP.

*Embedded targets:*

Motorola 680x0, 5600x, 68HC11, 6805, Coldfire (Netburner)

AMD and Intel 80x86, 8080, 8085, 80C31

Zilog Z80, Rabbit 2k/3k

Microchip PIC 16Cxx, PIC32

PC PCI and ISA bus, MVIP, VME bus, SPI, I<sup>2</sup>C

Contact Information:

Phone: 678 367-3157

EEmail: shenion@shdesigns.org

Address: Stone Mountain, GA 30088

Relocate: No

Looking for contract work in the Atlanta area.