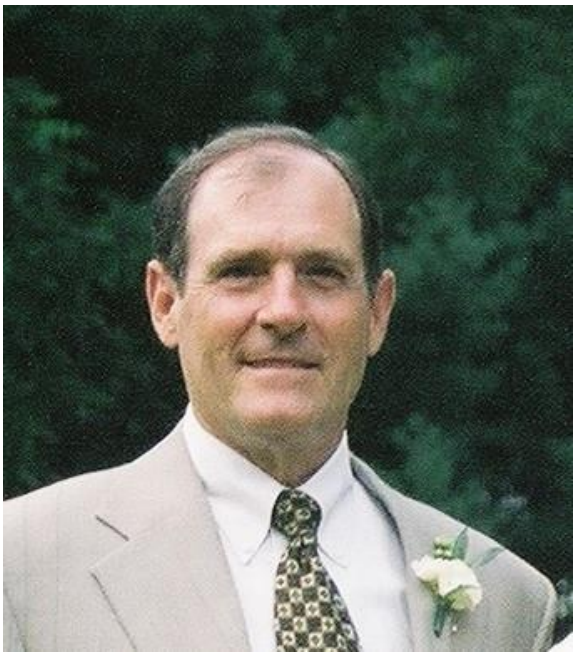


Francis Nelson Henderson

858-668-6943 (Cell with text)

6045 E Ridgeline Dr, #A201
South Ogden, UT 84405
francisnh12@gmail.com

US Citizen with Active Security Clearance: Top Secret.

Education: BYU Provo, Utah, B.A. Physics.
Minor – Mathematics, Business Administration.
Electrical Engineering (15 Units each)
Post Graduate work in Computer Science (UCSB)

Hobby: Medical Research on World Community Grid
Private Pilot's License

July 11, 2022 – Present, BAE Systems, Hill AFB, UT

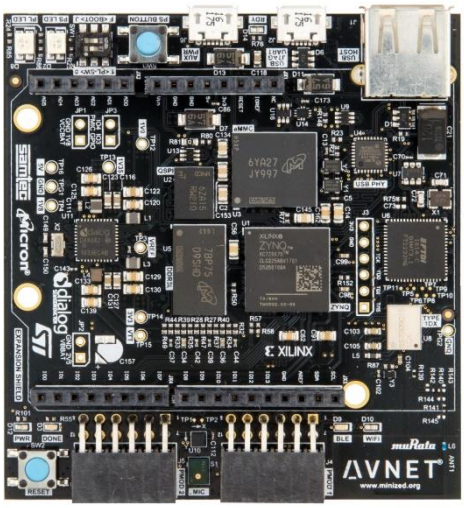
Ground Based Nuclear Deterrent: Sentinel ICBM flight simulation. Sensitivity Studies, Simulation Description for Users, BAE advisor to Government about Northrup Grumman Software Progress.

June 14, 2021 – April 1, 2022, Spectra Systems, Providence, RI

I learned Spectra Systems Software to help them recover their Intellectual Property that was lost after two key SW engineers resigned in succession. My work was on two Intel i7 OEM cards each doing high speed Bank Note counterfeit detection. Optical data is DMA'ed into Intel i7 CPU memory across the PCIe bus. Tasks and interrupts at the FPGA PCIe interface are handled by the Real-Time OS, Ontime RTOS 32, while the development environment is Microsoft Visual Studio.

I wrote "how to" documentation and wrote embedded diagnostic instrumentation in "C": 1.) A state machine to capture and display out of order forbidden events that are a problem. 2.) I instrumented 11 critical tasks using a high-resolution timer showing that Core utilization or CPU loading are not problems. 3.) I added instrumentation to inject one unchanging image file, thus allowing isolation and correction to unexpected calculation differences. Real-time instrumentation overhead is minimized by recording data into RAM for display after test runs.

Feb, 2020 – Mar 2021, Self-training on Xilinx Vivado & Vitis Tools during Covid-19 lock down.



During the Covid-19 lockdown, I self-trained on the Avnet Zynq MiniZed development board. I implemented a 1 milli-sec TCC0 interrupt driving bare-metal software written in “C” and executing on the ARM Cortex A9 CPU. I also brought up FreeRTOS running the “Test Port” Application seen here.

Sept 31, 2019 – Jan 17, 2020 Software Engineer, Northrop Grumman, Melbourne FL

First, I researched and recommended documentation tools for Northrop Grumman (NG) to use that are NOT Microsoft Office tools. Second, I studied how to configure secure communications on the Navy's E2D aircraft. Third, I worked on an assignment showing how a dozen test discrepancies were actually outside the DOORS requirements, hence did not need to be fixed. Strictly speaking NG had met the requirements which they also wrote, ignoring that requirements can themselves have defects. NG adheres strictly to the letter of the law. I would like to have made the software repairs myself except the schedule could not allow. My software development skills were not utilized and likely would not be. Hence, I resigned. Then the coronavirus hit.

May 21, 2019 – Aug 2, 2019 Software Engineer, Lockheed Martin, Palmdale CA

As a “contractor” doing Real-Time VxWorks C/C++ at Skunk Works helping a project get back on schedule. Designed the algorithm and code for locking the high resolution local CPU Time Base Register readings to the high accuracy GPS 1pps strobe. Defined Lockheed-Martin (LM) and Board Support Package vendor software work requirements for “scrubbing” single bit errors out of DRAM. On release I was told LM wanted the work I was doing be performed inhouse.

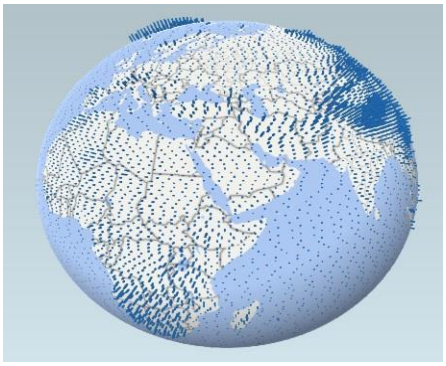
Aug 29 – Dec 21, 2018 Software Engineer, Lockheed Martin, Littleton CO

As a “contractor” I began Xilinx FPGA VHDL training for Hardware-In-The-Loop Satellite simulation. I debugged and fixed simulator Printed Circuit Boards (PCB). I added a Python script improving test bed verification. OS environments are Microsoft for Xilinx ISE tool and Linux for Python and GitLab. LM approached me about employment on “contract-to-hire,” however ISYS Technologies refused to release me saying it’s not their business model.

Mar 26 – June 6, 2018 Software Engineer, Lockheed Martin, Colorado Springs

As a “contractor” I prepared for my assignment to develop embedded software running on the Greenhills/Integrity RTOS. Target is a Military RISC CPU. The project moved to Utah while I elected not to relocate.

Dec 2016 to Sept 2017 Software Engineer GPS Source, Pueblo CO



I wrote a GPS serial message simulator (a tester) running on a PC driving serial messages into the GPS Source product as if located at positions all over the world. The tester validated content of the serial message reply from the embedded ARM processor's coordinate translations against the expected results.

The project successfully concluded, and I received written praise from the Director of Program Management saying, "… it's a real quality product you have delivered. Really stellar job. Thank you!"

Second, I wrote C code and built the executables in order to validate calculations of the ARM CPU and the IAR compiler (floating point math libraries) by translating 42,233 datum points from Geodetic (Lat, Lon, Alt) to GeoCentric (X, Y, Z) in three (3) completely different configurations, i.e., 3 different CPU's (Intel, AMD, ARM), 3 different compilers (Microsoft, GNU, IAR), 3 different OS's (Windows, Linux, RTOS) . The Geodetic (Lat, Lon, Alt) to GeoCentric (X, Y, Z) translations produced identical results in the three configurations, thereby validating the ARM CPU and IAR compiler combination.

Third, the project provided a point and click GUI using Excel, VBA, and a custom DLL written in C in order to drive the unit under test. The DLL allowed reuse of proven mathematical conversions the company owns. C also provided essential compiler features missing from VBA like "Union" and the "#Pragma Pack" directive that is missing from VBA. Structure packing is necessary at the serial port in order to remove extra padding added by compilers for memory alignment. It worked!

Mar 2010 to Nov 2014 Sr System Engineer Scientific Research Corp North Charleston SC

Contributor to the analysis of Software Defined Radio (SDR) waveform software submitted to the government for Information Assurance inspection according to NSA rules.

Contributed to the specification for Porting "Have Quick II" Software Defined Radio onto the SPAWAR Atlantic JTRS compliant hardware platform.

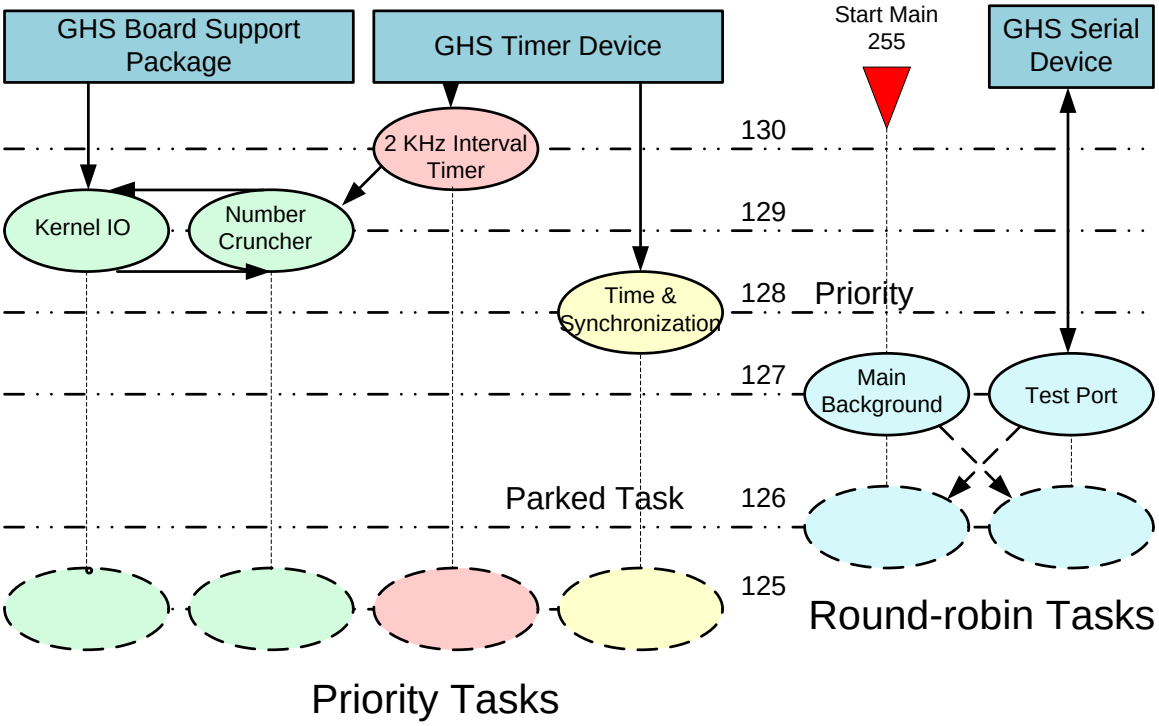
Modified the GreenHills Kernel of the IBM PPC440GP and Freescale MPC8548 microprocessors to receive the GPS time mark interrupt, and to receive the GPS serial message at the on-chip UART peripheral into the SDR timing service. Used Greenhill's integrity "178B" RTOS and development tools.

I took independent System-on-Chip training on the new Xilinx Zynq containing an ARM dual Core CPU and FPGA.

I am an active independent contributor to open source autopilot software development on an ARM CPU interfacing to 3-axis rate gyros and accelerometers and magnetometers, GPS receiver, airspeed and barometric sensor inputs. I am a Licensed Private pilot, a Radio Control airplane flyer, a "Pixhawk" autopilot user. I am a student of the open source autopilot software design. I successfully created the build environment and compiled the "Plane" code.

2008 – 2010 Principle Engineer EMS Technologies Inc. Norcross, GA

Designed, coded, and unit tested the software tasking structure and “test port” interface running on the Freescale MPC8560 embedded micro-processor. Ported the antenna pointing application code from the Linux PC environment to the Greenhill’s Integrity RTOS on embedded microprocessor. Championed FPGA/Microprocessor System on a Chip architecture.



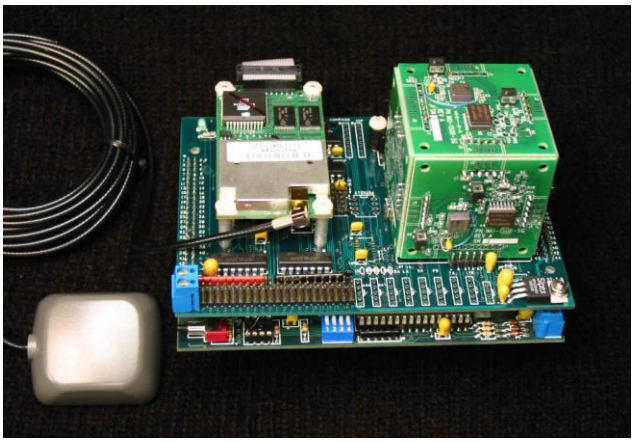
Wrote the theory of operation describing the aircraft antenna pointing “slave navigator.” Initiated the search for lower cost IMU’s and investigated dual antenna GPS receivers for initial heading reference. Contributed to and wrote the FPGA functional description at the PCI bus to micro-processor interface.

Chaired the weekly software technical coordination meeting between EMS and project partner L-3 Communications.

Selected to attend the Boeing AEHF satellite network design review, and to receive two weeks training.

2002 – 2007 Founder Inertial Solutions Poway, CA

I did system design, hardware and software development of an Attitude & Heading Reference System (AHRS). I did Analog and Digital Hardware Design, Printed Circuit Board (PCB) layout, PCB revisions, PCB procurement, PCB assembly seen below. I learned Earth models, navigation methods, Earth coordinate systems and coordinate transformation methods.



Designed and developed interface firmware to a 6 degree of freedom Inertial Measurement Unit (IMU), Air Data sensors for airspeed, altitude, and temperature. Software interfacing of GPS receivers to MC68332 processors. Pulse Width Modulation servo outputs controlled by adjustable bump-less set-points and IMU feedback.

Developed a UAV Flight Control Center containing a graphical cockpit instrument display, a virtual reality real-time display of the UAV, and a bi-directional telemetry channel from the above on-board processor to ground control.

Hardware design of MC68332 and HD647180 processor cards. Hardware design of Instrumentation Amplifier interfaces and 12 bit A-D conversion. Design of IO drivers for numerous peripherals including the Time Processing Unit (TPU) of the MC68332 micro-processor. Precise micro-processor synchronization to the GPS time mark. Work is seen here.

2004 – 2006 Northrop Grumman Radio Systems, Carmel Mountain, San Diego, CA

Developed the Antenna Pointing Software for the Inter-Flight Data Link (IFDL) waveform on the new F-35 Joint Strike Fighter aircraft. IFDL is a software radio waveform loaded into Northrop Grumman's JTRS compliant software infrastructure. Software development used Greenhill's Integrity C++ compiler and Secure Real-time Operating System.

Technical point of contact between Northrop Grumman and ViaSat for implementation of the ViaSat Satellite Communications software waveform into the Northrop Grumman Software Defined Radio infrastructure.

1998 – 2002 Senior Staff Engineer NSI Communications, San Diego

Took over and successfully rescued final implementation of the new MPC860 embedded microprocessor. Ported the TDMA modem code from the MC68020 processor to the MPC860 pSOS Real-time environment.

System design and analysis, software architect, and modem software development and test of a satellite network protocol for Frame Relay Bandwidth on Demand. The protocol managed the allocation of network capacity divided across time and frequency. The protocol incorporated an algorithm to perform load balancing. It considered two levels of priority, guarantees, and fair sharing. The protocol synchronized burst allocation, hitless burst reallocation, and de-allocation.

Provided network sizing and burst overhead analysis for sales. Helped define customer requirements and specify software modifications. Developed training materials and provided training to NSI's software engineering staff in Canada.

1994 - 1998 Manager, Real-time Software, ComStream, San Diego, CA

Manager, Real Time Software to 15 programmers. Expanded the network capacity from 120 TDMA terminals to 1000 terminals.

Responsibilities included contributions to Product Line Management, Project Engineering, and Sales. Preparation and presentation to prospective customers. For example, presentations to Dow Jones & Co in the US, and to the Ministry of Communications in China. Traveled to China twice, provided technical support to the Ministry of Petroleum customer in China.

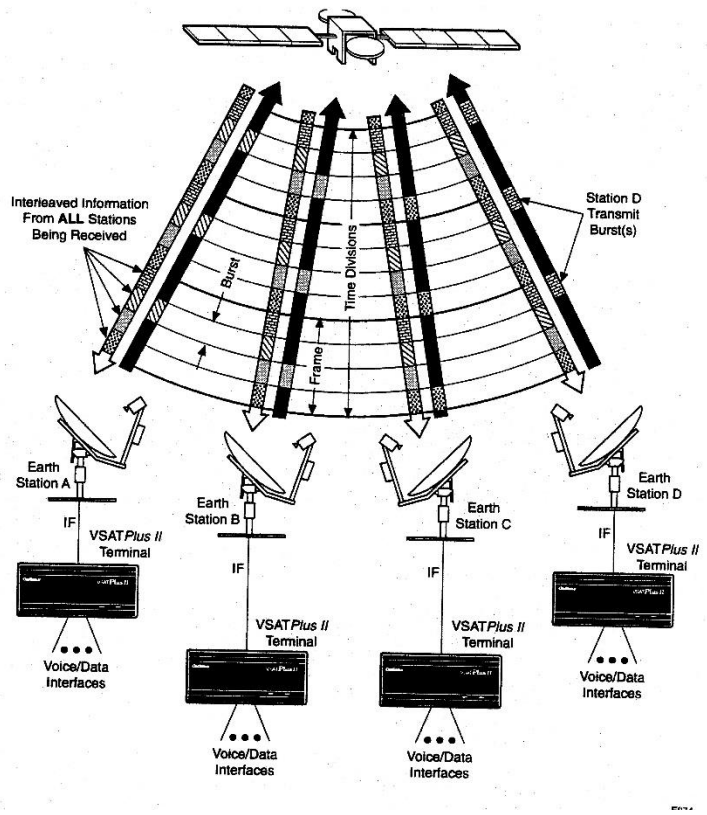
Performed system analysis and redesign as required for acquisition and synchronization to inclined orbit (degraded orbit) geo-synchronous satellites. Performed system analysis for satellite acquisition and synchronization on-board ship.

Took course work on Object Oriented Design and C++. Used the Microsoft Visual C++ environment. Also, used the Visual Basic environment for GUI development.

1993 – 1994 Sr. Software Engineer, ARK Communications, Santa Maria, CA

I specified and procured DSP firmware for voice compression from 64 kbps down to 4.8 kbps while preserving speech quality.

1984 – 1993 Sr. Software Engineer, SPAR Aerospace, Santa Maria, CA



I succeeded in unifying the modem and the interface card firmware structure, tools, and language, thus reducing development cost and schedules. I designed unified circuit routing and interoperation between differing telephony line protocols (Subscriber, FXO, E-1, T-1).

I did the system design, performance analysis, development, and test of the network protocol that provided guaranteed delivery of new network maps, new software code distributions, and new FPGA code distributions. Network map changes were synchronized to execute hitlessly, network wide.

I did system design of satellite transponder hopping controls and path verification. Served as development engineer and project engineer. Performed acceptance testing and delivery to Dow Jones & Co. Received letter of commendation.

I served as manager for six software developers for a period after SPAR's purchase of the COMTEL TDMA.

1979 – 1984 Director of TDMA Systems, COMTEL Santa Maria, CA

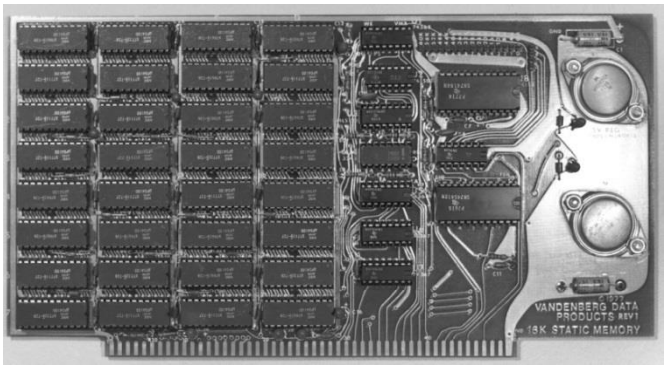
I am a founding member and "key employee" of the original COMTEL that created the first and second generation TDMA satellite communications terminal. I contributed architecturally to the first generation product and I developed all the satellite modem firmware. As Director of TDMA Systems I launched development of the second generation 60 MBit satellite terminal. I supervised 13 hardware and software engineers during this period.

Developed Burst Modem Acquisition and Synchronization to geo-synchronous satellites. Developed software controlled TDMA framing and burst transmission to the satellite. Designed the underlying modem software architecture. Wrote and tested modem code tested in the lab and on the satellite.

I designed and developed the distributed network protocol for voice channel Demand Assignment at the satellite side, and telephony signaling at the terrestrial trunk side. The design automatically accommodated the entry or exit of any node or trunk equipment in the network. Wrote code and tested T-1, E-1, “two wire” subscriber and “four wire” trunk signaling. Made voice signaling inter-operate network wide. Performed unit test. Wrote “design validation and test” procedures.

I did course preparation and customer training. Organized and conducted factory acceptance testing.

1976 – 1979 Founder and President, Vandenberg Data Products, Santa Maria, CA



This small company grossed $350,000 selling add-on memories in the early days of the personal computer revolution for Intel 8080 and Zilog Z80 CPU’s. I did the engineering design, documentation, advertising, technical support, procurement, financing, and I organized sales, manufacturing, test, customer service, and quality assurance.

1977 – 1978 Digital Design Engineer, General Electric Company, Lompoc, CA

Designed and developed a data acquisition system to support engineering test of a nuclear power plant. Developed a test fixture for diagnosing faults in newly assembled printed circuit cards.

1968 – 1977 Electronics Engineer, Air Force Western Test Range

Member of the Range Safety Division Testing Minuteman Ballistic Missile Nuclear deterrence.

Air Force counterpart as Operations Manager of an IBM 360/65 computer, a major computing resource at Vandenberg Air Force Base.

Managed the procurement of a computer (SDS) for the gathering and processing weather balloon data.

Project Director for the first use of micro-computers at the Range, the Datapoint 2200 preceding, but using the Intel 8080 machine language instruction set.