System Requirements Specification

for

Ingenion Telemetry Web Server

Version 1.1.1 approved

Prepared by Ryan Flinchum
Morgan Smith
Thomas Swenson
Ricky Nelson
Jack Capuano
Conrad Prisby
Hamilton Henneberg

2/5/2024

Table of Contents

Revision History

Revision Author	Date	Reason For Changes	Version
Hamilton Henneberg	9-22-23	Creation & Initialization of Document	1.0
Jack Capuano	9-29-23	Initial Draft of Functional & Non Functional Requirements	1.1
Ricky Nelson & Ryan Flinchum	9-29-23	Finalization of draft	1.2
Jack Capuano	10-31-23	V1 Feedback Implementation	2.0
Ryan Flinchum	10-31-23	• Removed extraneous requirements • Adjusted formatting to feedback • Adjusted existing requirements to implement feedback suggestions	2.1
Jack Capuano	12-1-23	Final Revision	3.0
Ryan Flinchum	2-5-24	Updated necessary information	1.1.1

1. Introduction

1.1 Purpose & Scope

The Ingenion Telemetry Web Server is purpose-built to serve as an integral tool in the testing and validation of satellite hardware, providing a real-time telemetry data interface that is both robust and user-friendly. Its scope is defined by the imperative to offer engineers a system that not only reliably processes telemetry data but also does so in a manner that facilitates ease of access and interaction. The server is intricately designed to augment the capabilities of existing Total Verification Systems (TVS), used extensively by space agencies, with an enhanced range of features that cater to modern satellite testing requirements.

This includes the strategic integration of a Xilinx MicroBlaze soft-core CPU within an Artix 7 FPGA, alongside the deployment of FreeRTOS—an open-source real-time operating system. These integrations ensure that the Ingenion Telemetry Web Server not only hosts an interactive web server but also establishes TCP over Ethernet connections for seamless data management. While the server is adept at handling complex telemetry data, it is crafted with a focus on simplicity and user-centric design, ensuring that even non-specialist users can operate it with minimal training.

It is important to note that while the Ingenion Telemetry Web Server is comprehensive in its functionalities, it is not intended to serve as a development platform for telemetry algorithms. Instead, it is a facilitator, enabling the display and management of telemetry data derived from satellite testing operations. The development of a graphical user interface (GUI) is not within the scope of this project, as the system is designed to complement existing interfaces. The onus of ensuring compatibility of telemetry algorithms with this web server rests with the users, aligning with the project's goal of offering a streamlined and efficient telemetry data management system. The project is focused on individual telemetry streams, providing a specialized solution that does not extend to broader data collection methods or the complexities of managing drone swarms or other such entities.

1.2 Document Conventions

i. The software requirements document (SRS) adheres to industry-standard conventions for clarity and consistency. No specific fonts or highlighting conventions have been employed. We chose not to assign priorities to requirements as requirements must be completed for us to achieve our end goal for this project. The SRS aims to provide a clear and unambiguous representation of the software requirements without relying on formatting nuances for conveying significance.

1.3 Intended Audience and Reading Suggestions

i. **Developers:** Start with the "Product Scope" and "Overall Description" sections for a high-level understanding. Dive into "Design and Implementation Constraints" for technical considerations. Explore "External Interface Requirements" for integration details.
ii. **Scrum Master:** Begin with the "Introduction" and "Product Scope" for project context. Focus on "Assumptions and Dependencies" for potential risks. Review "Other Nonfunctional Requirements" and "Business Rules" for project constraints and standards.
iii. **External Users:** Begin with the "Introduction" and "Product Scope" to grasp the project's purpose. Explore "User Interfaces" for interactions with the system.

1.4 Product Scope

i.Our objective for this phase of the Ingenion project is to focus on the software development for enhancing the Total Verification System (TVS), a critical testing tool used by NASA's Goddard Space Flight Center for simulating, testing, and verifying satellite hardware components. With the FPGA hardware already developed, our attention shifts to refining and expanding the system's software capabilities.This project will consist of the integration of FreeRTOS, an open-source real-time operating system, on the Xilinx MicroBlaze soft-core CPU implemented on the Artix 7 FPGA. In addition to the FreeRTOS implementation, we will develop an interactive web server hosted on the FPGA. The web server will facilitate connections to external computers using TCP over Ethernet, enabling the system to read and display telemetry data from peripherals including GPIO, SPI, UART, and I2C.

1.5 References

i. Xilinx Wiki. (n.d.). ARTY FreeRTOS Web Server. Retrieved from https://xilinx-wiki.atlassian.net/wiki/spaces/A/pages/18841844/ARTY+FreeRTOS+Web+Server

ii. Digilent. (n.d.). Nexys 4 DDR - Getting Started with Microblaze Servers. Retrieved from https://digilent.com/reference/learn/programmable-logic/tutorials/nexys-4-ddr-getting-started-with-microblaze-servers/start

2. Overall Description

2.1 Product Perspective

i. The Ingenion Telemetry Web Server project serves as an augmentation to the existing Total Verification System (TVS). It enhances the TVS by integrating a Xilinx MicroBlaze soft-core CPU on the Artix 7 FPGA and implementing FreeRTOS as an open-source real-time operating system. These enhancements enable the system to host an interactive web server, establish TCP over Ethernet connections with external computers, and manage telemetry data from external,

user-friendly connections. This integration will provide a vital bridge between existing satellite hardware components and the ability to monitor and interact with them remotely.

2.2 Product Functions

i. The system will host an interactive web server, establish TCP over Ethernet connections, and manage telemetry data. The primary functions of the Ingenion Telemetry Web Server project are as follows:
   - Hosting an interactive web server to enable remote monitoring and control of satellite hardware components.
   - Establishing TCP over Ethernet connections with external computers to facilitate data exchange and remote management.
   - Managing telemetry data from external sources, making it accessible through user-friendly interfaces.

2.3 User Classes and Characteristics

i. Potential users of the system may include:
   - Developers involved in system design and implementation.
   - Engineers responsible for hardware and software integration.
   - Project managers overseeing the project's progress.
   - External users who access the web server for telemetry data and control.

2.4 Operating Environment

i.The operating environment for the Ingenion Telemetry Web Server project includes:
   - Hardware components, such as the Artix 7 FPGA and Digilent Nexys A7 development board.
   - Software components, including FreeRTOS as the real-time operating system.
   - Network infrastructure to support TCP over Ethernet connections.
   - Integration with the existing Total Verification System (TVS) at NASA's Goddard Space Flight Center.

2.5 Design and Implementation Constraints

i. Constraints for the Ingenion Telemetry Web Server project include:

- Resource Allocation: The project must operate within predefined resource constraints, including FPGA resources, memory, and processing power. Trade-offs have been considered to optimize resource utilization without compromising functionality.
- Compatibility: Compatibility with existing systems, including the TVS and external computer systems, imposes constraints on the design to ensure seamless integration.
- Time and Labor: The project operates within specific time, budget, and labor constraints, in order to complete this project within the given time period (1 semester), and labor (7 team members).

2.6 User Documentation

i. Helpful links/ User Manuals
https://www.xilinx.com/video/fpga/microblaze-webserver-demo.html
https://www.youtube.com/watch?v=Pg7BHJoUPDg

2.7 Assumptions and Dependencies

i. Assumptions and dependencies for the Ingenion Telemetry Web Server project include:
- Interface Agreements: If interface agreements are difficult to establish, contingency plans will involve identifying alternative communication protocols or temporary solutions.
- Architectural Stability: In the event that architectural stability issues arise with the chosen FPGA or CPU platform, the project team will explore alternative hardware options or adapt the design to mitigate instability.
- Project Scale: If, during the course of the project, it is determined that the scope is either too large or too small to align with project objectives, contingency plans will be activated to scale the project accordingly. This may involve expanding or reducing features, adjusting resource allocation, or revisiting project goals to ensure alignment with project size and complexity

3. External Interface Requirements

3.1 User Interfaces

REQ-3.1.1: The system shall be connected to users via an interactive web server that will have display capabilities of telemetry from peripherals.

REQ-3.1.2: The system shall be able to receive commands to the CPU over ethernet, sent by the user.

3.2 Hardware Interfaces

REQ-3.2.1: The system shall have ethernet capabilities in order to connect with the user's computer through an ethernet interface at 10/100 Mbps

REQ-3.2.2: The software product shall interface with the DDR3 memory in the FPGA

REQ-3.2.3: The system interface with the DDR3 memory shall support read and write operations for data storage

REQ-3.2.4: The software product shall interface with the GPIO soft cores within the FPGA,

REQ-3.2.5: The system interface with the GPIO cores shall enable configuration and monitoring of general-purpose I/O signals

3.3 Software Interfaces

REQ-3.3.1: The system must run FreeRTOS on the MicroBlaze CPU, with performance metrics meeting customer defined benchmarks.

REQ-3.3.2: The system must utilize FreeRTOS for task scheduling, with quantifiable performance thresholds.

REQ-3.3.3: The system shall have a web server software component

REQ-3.3.4: The system's web server software component shall be compatible with FreeRTOS

REQ-3.3.5: The web server software must interface with FreeRTOS to retrieve telemetry data, demonstrating measurable response times.

REQ-3.3.6: The system must handle TCP/IP communication over Ethernet, with specific performance criteria.

REQ-3.3.7: The system must process telemetry data from the AXI bus, adhering to defined speed/accuracy standards.

REQ-3.3.7: The system must process telemetry data from peripherals, adhering to defined speed/accuracy standards

REQ-3.3.8: The system must display telemetry data in real-time, with a defined maximum latency.

3.4 Communications Interfaces

REQ-3.4.1: The system must use HTTP for web interactions, meeting predefined protocol standards.

REQ-3.4.2: The system must implement WebSockets for real-time communication, with specified performance benchmarks.

REQ-3.4.3: The system's Ethernet communication shall comply with IEEE 802.3 standards.

REQ-3.4.4: The system must accept user commands via HTTP POST requests, with specific response time criteria.

REQ-3.4.5: The system must log communication events, errors, and warnings

REQ-3.4.6: The system's logging subsystem must meet the customer's formats, retention and security policies

4. System Features

4.1 System Feature Example

i. Communication Link Establishment

4.1.i.1 Description and Priority: This feature involves establishing and maintaining a reliable communication link over Ethernet using the TCP/IP protocol suite.

4.1.i.2 Stimulus/Response Sequences:

- *Stimulus:* User initiates telemetry data capture or configures system settings via the web interface. *Response:* The system establishes and maintains a communication link for real-time telemetry data display.

4.1.i.3 Functional Requirements:

REQ-4.1.1: Establish and maintain an Ethernet communication link using TCP/IP.

REQ-4.1.2: Ensure seamless interaction with external computers via the established link.

REQ-4.1.3: Implement robust error handling in the communication link setup.

ii. User Web Interface

4.1.ii.1 Description and Priority: This feature focuses on providing a user web interface, allowing users to initiate telemetry data capture, request specific data, and configure system settings.

4.1.ii.2 Stimulus/Response Sequences:

- *Stimulus:* User interacts with the web interface to initiate telemetry data capture, request data, or configure settings. *Response:* The web interface provides options for these actions.

4.1.ii.3 Functional Requirements:

REQ-4.1.4: Allow users to initiate telemetry data capture through the web interface.

REQ-4.1.5: Enable users to request specific data via the web interface.

REQ-4.1.6: Provide configurable system settings through the web interface.

iii. Telemetry Data Parsing and Display

4.1.iii.1 Description and Priority: This feature involves parsing of telemetry data upon request and presenting it in a structured format for clarity.

4.1.iii.2 Stimulus/Response Sequences:

- *Stimulus:* System receives a request for telemetry data. *Response:* Telemetry data is parsed and displayed in a structured format on the web interface.

4.1.iii.3 Functional Requirements:

REQ-4.1.7: Support the parsing of telemetry data in a predefined format.

REQ-4.1.8: Ensure telemetry data is displayed in a clear, structured format

5. Other Nonfunctional Requirements

5.1 Performance Requirements

REQ-5.1.1: Conduct functionality verification of the Xilinx MicroBlaze CPU with defined benchmarks for reliability and performance.

REQ-5.1.2: The system shall ensure real-time data processing by FreeRTOS, with specific latency and throughput metrics.

REQ-5.1.4: The system shall host an interactive web server for equipment monitoring, with defined uptime and response time standards.

REQ-5.1.5: The system's web server shall allow for control of test equipment, meeting the same uptime and response time standards.

REQ-5.1.6: The system shall acquire telemetry data from the AXI bus with a maximum allowable delay and error rate.

REQ-5.1.7: The system shall support command transmission to the CPU, adhering to a specified protocol with measurable transmission times.

5.2 Safety Requirements

REQ-5.2.1: The system shall continuously monitor for hardware malfunctions

REQ-5.2.2: The system's malfunction monitoring shall alert the user of identified issues

REQ-5.2.3: The system shall implement self-diagnostic routines for fault detection, specifying the frequency and scope of these diagnostics.

REQ-5.2.4: The system shall export detected abnormalities to the user, specifying reporting formats and timelines.

REQ-5.2.5: The system shall monitor for other abnormal behavior, specifying the log format, retention period, and access protocols.

5.3 Security Requirements

REQ-5.3.1: The system shall incorporate measures for data integrity during telemetry data acquisition, specifying allowable error rates.

REQ-5.3.2: The system shall maintain data integrity during telemetry transmission, with defined security protocols and encryption standards.

REQ-5.3.3: The system shall implement redundant storage to prevent data loss, specifying redundancy levels and recovery processes.

REQ-5.3.4: The system shall establish redundant backup systems, detailing backup frequency, storage locations, and restoration procedures.

5.4 Software Quality Attributes

REQ-5.4.1: The system shall achieve a defined MTBF that aligns with customer reliability expectations.

REQ-5.4.2: The system shall maintain a low MTTR with effective error logging, specifying log details and error resolution protocols.

REQ-5.4.3: The system shall recover gracefully from unexpected failures or errors, detailing recovery processes and maximum allowable downtime.

REQ-5.4.4: The system shall ensure optimal performance for processing telemetry data, defining performance benchmarks.

5.5 Business Rules

REQ-5.5.1: The system and all subsystems shall adhere to industry safety standards, specifying the relevant standards and compliance verification processes.

REQ-5.5.2: The system shall ensure compatibility with existing systems, particularly with TVS, detailing integration and testing processes.

6. Other Requirements

REQ-5.6.1: The system shall adhere to licensing terms allowing modification or distribution, detailing the specific terms.

REQ-5.6.2: The system shall come with comprehensive documentation on intellectual property rights, licensing, and usage permissions.

Appendix A: Glossary

Ingenion Telemetry Web Server	The software component responsible for facilitating communication and interaction between the Digilent Nexys A7 development board, Xilinx Artix 7 FPGA, MicroBlaze soft-core CPU running FreeRTOS, and external client devices via Ethernet. It is a critical part of the Total Verification System (TVS) project
Total Verification System (TVS) Project	A project aimed at enhancing satellite hardware testing by integrating advanced features and improving telemetry data management
Digilent Nexys A7	A development board used in the project for interfacing with satellite hardware components. Xilinx Artix 7 FPGA: A Field-Programmable Gate Array (FPGA) from Xilinx used to host the MicroBlaze soft-core CPU and execute the software
MicroBlaze	A soft-core microprocessor designed by Xilinx for implementation on FPGAs. FreeRTOS: An open-source real-time operating system used for running tasks on the MicroBlaze CPU
Ethernet	A network communication protocol used for connecting the Nexys A7 board to external client devices

TCP/IP	Transmission Control Protocol/Internet Protocol, the suite of communication protocols used for transmitting data over networks, including the internet
User Interface (UI)	The graphical interface that allows end-users to interact with and control the telemetry system. Telemetry Data: Data collected from satellite hardware components, used for monitoring and control
External Computer	A device connected to the telemetry system via Ethernet for data exchange and management. Real-Time Web Server: A web server that provides immediate and interactive access to telemetry data and system controls
Communication Protocols	Standards like Ethernet, TCP/IP, UART, I2C that are used for reliable data exchange within the system
MicroBlaze	The soft-core CPU integrated into the Artix 7 FPGA for processing task
HTTP Protocol	Hypertext Transfer Protocol, used for communication between web browsers and the web server.
Websockets	A communication protocol that enables real-time, bidirectional communication between the web server and connected clients
IEEE 802.3	A standard for Ethernet communication
HTTP POST Requests	A method for sending data from a user to a web server

System Features	Key functionalities or capabilities of the telemetry web server, such as communication link establishment, user web interface, and telemetry data parsing and display

Appendix B: Analysis Models

Figure 1

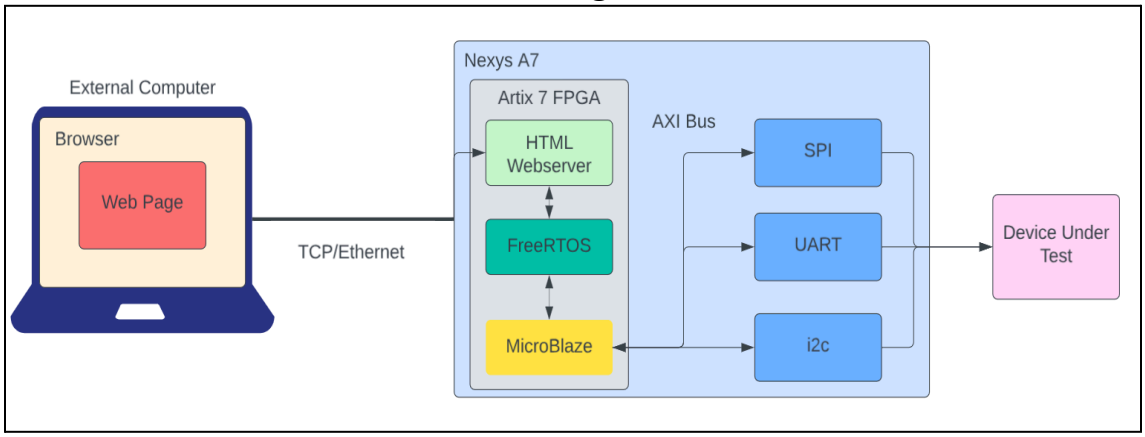

Figure 1 shows a graphical representation of a general component diagram for the system

Appendix C: To Be Determined List