Instituto Tecnológico y de Estudios Superiores de Occidente

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018, publicado en el Diario Oficial de la Federación del 29 de noviembre de 1976.

Departamento de Electrónica, Sistemas e Informática

Especialidad en Sistemas Embebidos

Design and Implementation of a Satellite Communication System using Iridium Network for Environmental Data Transmission

TRABAJO RECEPCIONAL que para obtener el **GRADO** de
ESPECIALISTA EN SISTEMAS EMBEBIDOS

Presenta: **GUSTAVO ALEJANDRO SÁNCHEZ ARVIZU**
MIGUEL ÁNGEL MANZANO HERNÁNDEZ
HÉCTOR MIGUEL HERNÁNDEZ VARGAS
LUIS ALONSO JIMÉNEZ ZEPEDA
SERGIO RAMÍREZ PRECIADO
ADRIANA RAMOS GARCÍA

Tutor **LUIS RIZO DOMÍNGUEZ**

Tlaquepaque, Jalisco. agosto de 2025

Design and Implementation of a Satellite Communication System using the Iridium Network for Environmental Data Transmission

Héctor Miguel Hernández Vargas
Departamento de Electrónica, Sistemas e Informática
Instituto Tecnológico y de Estudios Superiores de Occidente
Tlaquepaque, México
hhernandez.vargas@iteso.mx

Luis Alonso Jiménez Zepeda
Departamento de Electrónica, Sistemas e Informática
Instituto Tecnológico y de Estudios Superiores de Occidente
Tlaquepaque, México
ljimenez.zepeda@iteso.mx

Sergio Ramírez Preciado
Departamento de Electrónica, Sistemas e Informática
Instituto Tecnológico y de Estudios Superiores de Occidente
Tlaquepaque, México
sergio.ramirezp@iteso.mx

Adriana Ramos García
Departamento de Electrónica, Sistemas e Informática
Instituto Tecnológico y de Estudios Superiores de Occidente
Tlaquepaque, México
adriana.ramosg@iteso.mx

Miguel Ángel Manzano Hernández
Departamento de Electrónica, Sistemas e Informática
Instituto Tecnológico y de Estudios Superiores de Occidente
Tlaquepaque, México
angel.manzano@iteso.mx

Gustavo Alejandro Sánchez Arvizu
Departamento de Electrónica, Sistemas e Informática
Instituto Tecnológico y de Estudios Superiores de Occidente
Tlaquepaque, México
ie716000@iteso.mx

Luis Rizo Domínguez
Departamento de Electrónica, Sistemas e Informática
Instituto Tecnológico y de Estudios Superiores de Occidente
Tlaquepaque, México
lrizo@iteso.mx

This work presents the design of an environmental data transmission system tailored for the seventh iteration of EMIDSS suborbital mission. In addition, an artificial intelligence algorithm was implemented to enhance temperature data accuracy. The system architecture incorporates the Iridium 9603N modem in a modular board interfaced through an emulated I²C driver protocol with the NXP S32K144 microcontroller. The EMIDSS system demonstrated robust and reliable data transmission with sustained accuracy under low-temperature conditions. This work demonstrates the advantages of code reuse, the robustness of NXP's automotive platforms, and the use of AI in the temperature sensor.

Keywords—*Satellite transmission, embedded system, Iridium network, S32K144W microcontroller, polynomial regressions*

I. INTRODUCTION

Nanosatellites are efficient and affordable devices widely used to collect data such as temperature, humidity and pressure from the Earth's stratosphere. One of the primary difficulties encountered by nanosatellites is effectively transmitting the data they collect. Embedded systems are designed to process and transmit data more efficiently while operating with limited power and space constraints. Furthermore, the integration of embedded systems in nanosatellites facilitates real-time transmission and enhances overall system performance.

A challenge equally critical as data transmission is precise temperature behavior modeling under extreme weather. Polynomial regression has shown strong performance in capturing nonlinear thermal variations with low computational cost, outperforming more complex algorithms such as neural networks and support vector machines [1]. This makes it a reliable and efficient option for correcting sensor deviations and enhancing measurement accuracy in harsh environments.

The Experimental Module for the Iterative Design for Satellite Subsystems (EMIDSS) is a research project developed by the *Instituto Tecnológico y de Estudios Superiores de Occidente* (ITESO), in collaboration with the *Universidad Nacional Autónoma de México* (UNAM) and the *Instituto Politécnico Nacional* (IPN), with support from the National Aeronautics and Space Administration (NASA). EMIDSS aims to collect data from the stratosphere during suborbital missions. It uses an embedded system connected to a high-altitude balloon to measure temperature, pressure, and humidity. EMIDSS has undergone six prior iterations (EMIDSS 1-6). In EMIDSS-5, an embedded system architecture was designed following AUTOSAR standards using onboard sensors and memory to collect data powered by batteries [2]. However, data storage occurred locally and became accessible only after the balloon returned to Earth. The data required physical recovery before it could be read and processed.

In the EMIDSS-7 mission, the EMIDSS-5 platform served as the foundation for collecting and transmitting stratospheric data via an integrated satellite communication capability. To enable this functionality, the Iridium 9603N satellite modem was interfaced with the S32K144W microcontroller through the implementation of a FlexIO I²C driver. Furthermore, an algorithm was developed to improve the robustness and reliability of temperature data acquisition.

This paper is organized as follows. Section II presents system design and architecture. Section III describes hardware development. Section IV details the configuration of the FlexIO I²C driver and the procedure for enabling satellite transmission. Section V explains the polynomial algorithm structure. In section VI presents experimental results and section VII the conclusion and future work.

II. DESIGN AND ARCHITECTURE

The EMIDSS-7 system design followed a modular approach to ensure scalability and ease of integration. The system architecture is divided into key functional blocks as shown in "Fig. 1"

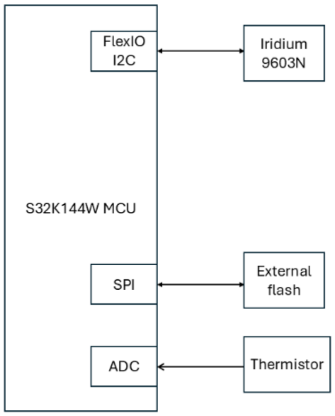

Fig 1. EMIDSS-7 System Architecture Design.

A flash memory module is connected through the SPI interface. This memory is used for local storage of environmental data, ensuring backup in case of transmission failure. The use of SPI allows high-speed transfers with minimal resource usage from the MCU.

Satellite communication is handled by the Iridium 9603N modem. A dedicated communication channel for the modem was implemented using the FlexIO peripheral to emulate the I²C protocol in the microcontroller S32K144W. This architecture separates memory access and communication, reducing the risk of bus noise and allowing concurrent data logging and transmission. The modular design also supports future expansion, such as the integration of new sensors or additional peripherals without requiring architectural redesign.

Temperature measurement is complemented by integrating a thermistor routed directly to the ADC peripheral of the microcontroller.

III. HARDWARE DEVELOPMENT

A. Hardware Implementation for Data Transmission

The hardware design of EMIDSS-7 builds upon PCB developed in the EMIDSS-5 iteration, preserving the base layout while introducing key upgrades to enable real-time satellite communication and mechanical integration. A six-pin connector and four mounting points were added to interface the Iridium 9603N modem and its satellite antenna with the system.

To establish communication with the NXP S32K144W microcontroller, dedicated traces were routed to FlexIO-compatible pins PTA0 and PTA1 (pins 50 and 49), enabling the emulation of the I²C protocol required by the Iridium 9603N modem. This communication link allows the modem to transmit environmental data via the Iridium satellite network, which was selected due to its global availability and proven reliability in academic research missions.

B. Hardware Implementation for Thermistor

A critical challenge during space environment missions is obtaining reliable data. Among the parameters exhibiting significant variation during ascent is temperature, which can drop from room temperature (27°C to 31°C) to approximately –50°C [3]. Therefore, a robust sensing solution for data processing is essential during suborbital missions.

To tackle this, the sensor from the previous mission (MS860702BA01-50) was supplemented with a thermistor (B57421V2223H062), which provided an extended temperature range (–50°C to 125°C). The thermistor was implemented in a voltage divider configuration along with a 22 kΩ resistor, connected to pin 31 of the microcontroller (ADC0_SE7 channel), featuring a 12-bit resolution and powered by a 3.3 V reference input, as illustrated in "Fig. 2".

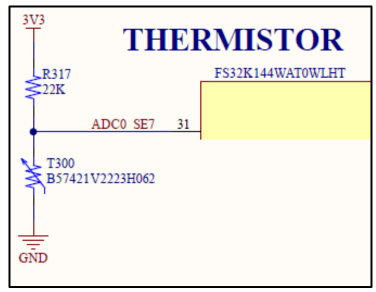

Fig. 2. Schematic illustrating the thermistor integration.

The PCB was designed using Altium Designer. "Fig. 3" shows the final board layout and "Fig. 4" presents the assembled hardware used in EMIDSS-7.

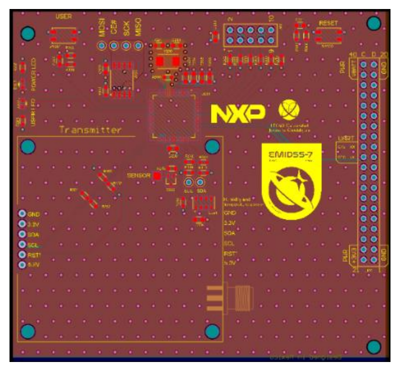

Fig. 3. 2D Model of the EMIDSS-7 PCB layout. The design was adapted from the EMIDSS-5 board, with an added connector and trace routing for the Iridium 9603N module.

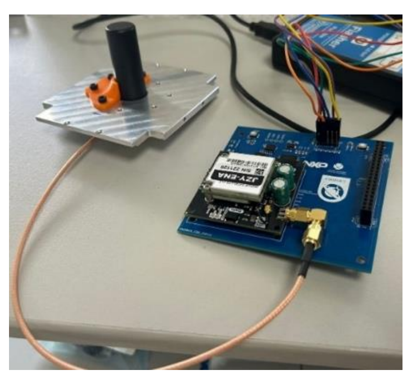

Fig.4 Assembled EMIDSS-7 hardware. The modem, antenna connector, and mounting points are visible, confirming structural and electrical integration.

IV. SATELLITE COMMUNICATION

A. FlexIO_I²C Driver Configuration

The FlexIO module is a highly configurable peripheral developed by NXP and integrated into several of its microcontroller families. Its purpose is to support a wide range of functionalities, including the emulation of communication protocols such as UART, I²C, I²S, SPI, among others. This module provides a flexible interface for extending microcontroller functionality when a project requires more instances of a specific peripheral. Among its key features are the use of 32-bit shift registers operating in transmit, receive and data match modes, as well as double buffering to enable continuous data transfers. It also supports register concatenation to handle larger data blocks, automatic generation of start and stop bits, and operation via interrupts, DMA or polling. Additionally, it provides programmable baud rates independent of the bus clock and includes highly flexible 16-bit timers that support a wide range of internal and external conditions for triggering, resetting, enabling and disabling operations [4].

In EMIDSS-7, the FlexIO peripheral was configured to emulate the I²C communication protocol to transmit and receive data from the Iridium 9603N hardware module with the microcontroller acting as the master. To achieve this, two shifter buffers and two timers were implemented. The first shifter was used to transmit data to the module, while the second was configured to receive data. The first timer was responsible for generating the SCL signal, and the second timer was used to control the operation of the shifter buffers. "Fig. 5" shows the FlexIO module microarchitecture for I²C emulation.

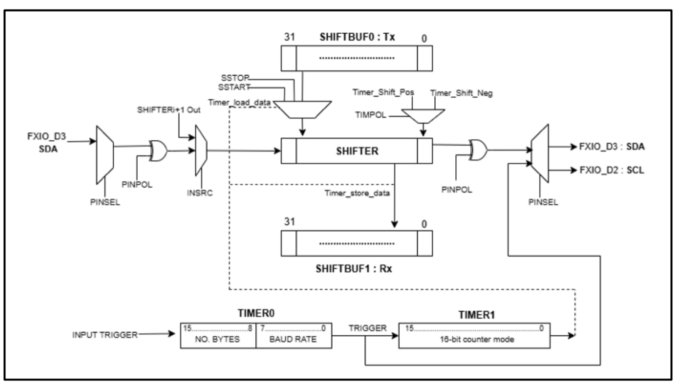

Fig. 5. I²C emulation with the FlexIO module.

B. Iridium9603 Driver Implementation

The Iridium 9603N module utilizes the Iridium Short Burst Data (SBD) service to transmit short messages with low latency via the Iridium satellite constellation. This capability enables global communication regardless of the device's geographic location. Once an SBD message is transmitted, it is relayed by the satellite network to a designated email address, facilitating reliable remote data collection [5].

To establish communication with the module, a dedicated driver was developed leveraging the FlexIO peripheral configured as an I²C master. This interface is used to send AT commands that control and configure the Iridium 9603N module, thereby enabling data transmission over the satellite network. "Fig. 6" illustrates the module's initialization sequence.

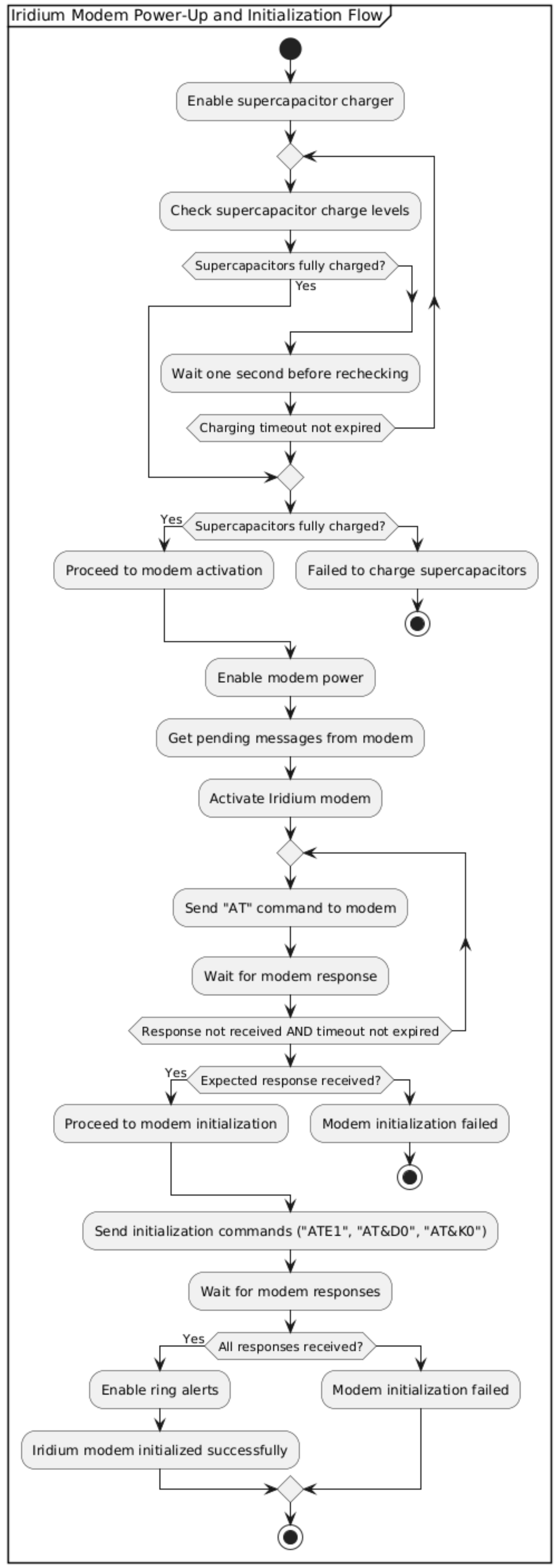

Fig. 6 Modem initialization sequence.

Once successfully initialized, the module is ready to receive AT commands for configuration and control. These commands allow management of core functionalities including power state, message transmission, modem behavior, and signal quality monitoring. Table I provides a summary of the AT commands employed in this implementation, detailing their purposes and expected responses.

Table I. Summary of AT Commands.

AT Command	**Purpose**	**Expected response**
AT\r	Verifies modem connectivity and ensures it is responsive.	OK
ATE1\r	Enables command echo for diagnostic visibility.	OK
AT&D0\r	Disables DTR (Data Terminal Ready) reset behavior.	OK
AT&K0\r	Disables hardware flow control.	OK
AT+SBDMTA=1\r	Enables RING alert notifications for incoming messages.	OK
AT+SBDIX\r	Initiates a Short Burst Data (SBD) session for sending and receiving.	+SBDIX: <MO> (STATUS)
AT+CSQ\r	Requests signal quality level from the modem.	+CSQ: <0-5>
AT+SBDWT\r	Prepares the modem to receive a text message for transmission.	READY
AT+SBDD0\r	Clears the mobile-originated (MO) message buffer.	OK

For transmitting mobile-originated (MO) messages, a dedicated function that receives a pointer to an ASCII-formatted data buffer was implemented. "Fig. 7" shows the send logic for this function, which encapsulates the required AT command sequence and manages the interaction with the modem via the FlexIO-based I^2C driver. The initialization and transmission sequence were based on SparkFun IridiumSBD I2C Arduino library [6].

C. Data Transmission Application

The data transmission application was implemented over a task executed every minute where sensor data is collected for transmission. A 10-minute interval wait is required and only successfully acquired data is considered for delivery. When the conditions for transmission are met, the system turns on the satellite modem and prepares the most recent environmental data for delivery. Prior to initiating communication, the signal quality is evaluated to ensure a reliable link with the satellite network. Transmission is attempted only if the signal strength exceeds 4 points.

Once a valid signal is detected, the modem's internal message buffer is cleared to prevent residual data from affecting the new transmission. The environmental data, formatted as an ASCII message, is then sent over the Iridium network. After the operation concludes, the message buffer is cleared again to maintain data integrity and the modem is turned off to conserve energy. This process ensures that data is transmitted efficiently and reliably under favorable signal conditions.

Each message follows the structure shown below:

TXX.XX HXX.XX PXXX.XX

For example: T27.18 H48.28 P844.51

Where T is temperature in °C, H is relative humidity in %, and P is atmospheric pressure in hPa.

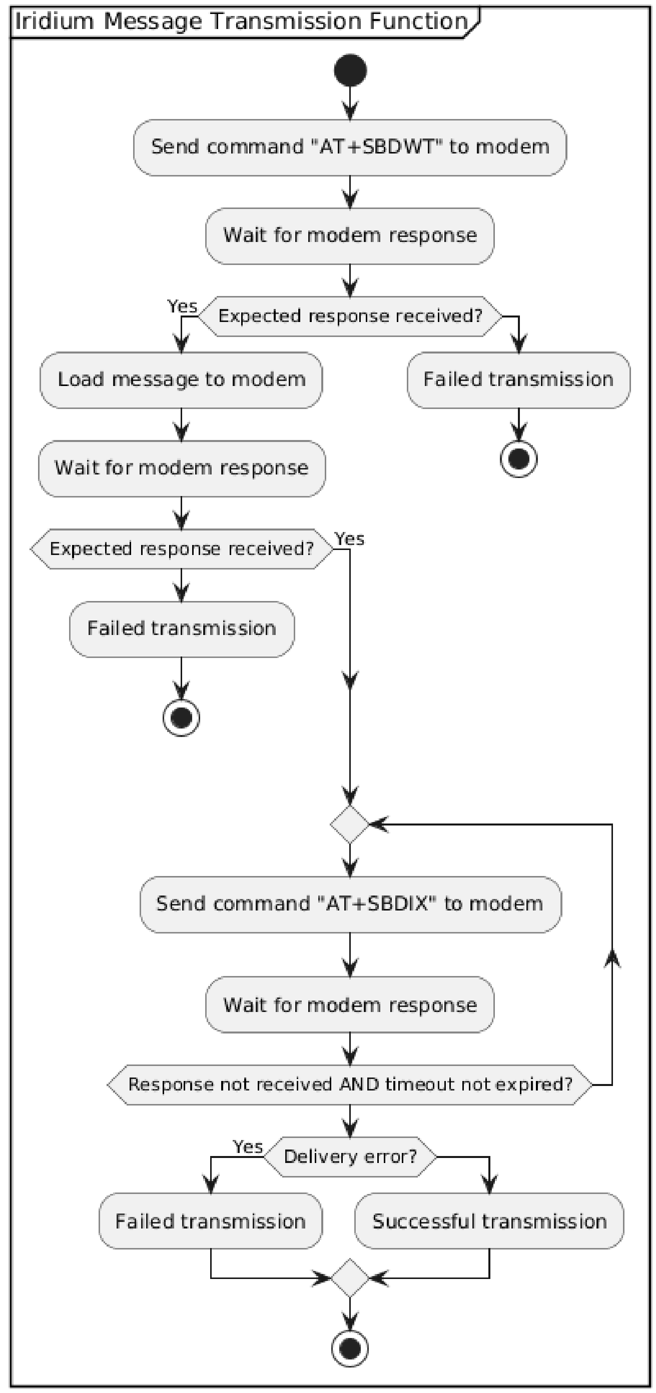

Fig. 7 Message transmission function.

V. Polynomial Regression Solution

To determine the temperature based on the resistance variation of the thermistor, the manufacturer provides tabulated values of temperature and the ratio R_T/R_{25} shown in Table II, where R_T represents the thermistor's resistance at current temperature and R_{25} corresponds to its resistance at 25 °C (22 kΩ).

Table II. Ratio of Rt/R25 vs temperature.

Temperature in °C	Rt/R25	Temperature in °C	Rt/R25
-55	96.158	50	0.35981
-50	66.892	55	0.29819
-45	47.127	60	0.24837
-40	33.606	65	0.20787
-35	24.243	70	0.17479
-30	17.681	75	0.14763
-25	13.032	80	0.12523
-20	9.702	85	0.10667
-15	7.2923	90	0.091227
-10	5.5314	95	0.078319
-5	4.2325	100	0.067488
0	3.2657	105	0.058363
5	2.54	110	0.050647
10	1.9907	115	0.044098
15	1.5716	120	0.03852
20	1.2494	125	0.033752
25	1	130	0.029663
30	0.80552	135	0.026146
35	0.65288	140	0.023111
40	0.53229	145	0.020484
45	0.43645	150	0.018203

To estimate the temperature, the measured thermistor ratio is compared with the tabulated values. However as observed in Table II, the thermistor exhibits a highly nonlinear behavior particularly at lower temperatures where resistance changes are more pronounced. This characteristic is illustrated in "Fig. 8", which shows the thermal response curve of the NTC thermistor.

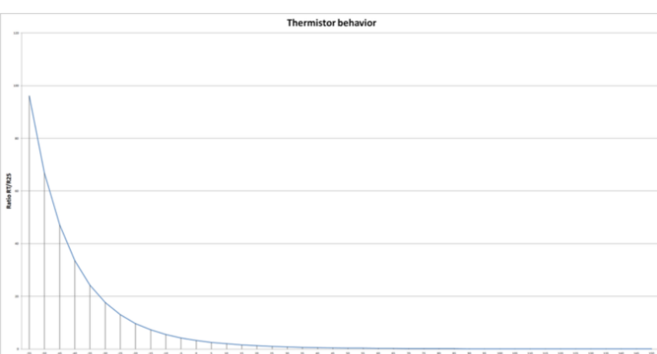

Fig. 8. Graph of the characteristic thermal behavior of the NTC thermistor.

Initially, a polynomial regression method [7] was applied to approximate the thermistor's nonlinear response. The approach involved fitting a polynomial curve of degree n over the tabulated data to model the relationship between the thermistor ratio and temperature. Using least squares fitting, the coefficients $a_0, a_1, \dots, a_n$ were computed to form the polynomial:

$$P(x) = a_0 + a_1 x + a_2\ x^2 + \cdots + a_n x^n \tag{1}$$

This fitting was performed over a temperature range of –55 °C to 150 °C to cover all possible operating conditions during ascent. The calculated coefficients were integrated into the EMIDSS-7 firmware to compute real-time temperature values from ADC readings:

$$T_{corrected} = \sum_{i=0}^{n} a_1 \cdot temp^i = a_0 + a_1 \cdot temp + \\ + a_2 \cdot temp^2 + \cdots + a_n \cdot temp^n \tag{2}$$

Although this method theoretically enabled continuous temperature estimation a significant limitation was exhibited. The polynomial curve introduced large oscillations in specific segments of the temperature range which caused temperature predictions to deviate sharply from the actual sensor behavior and compromised accuracy [8]. As shown in "Fig. 9" the polynomial regression failed to accurately represent the thermistor's behavior across the full temperature range.

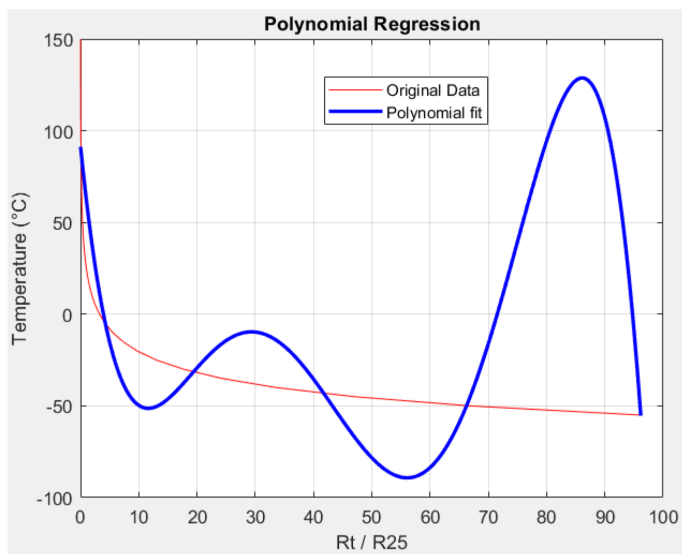

Fig. 9. Polynomial regression results of the thermistor.

To address these issues, a second-degree rational function was adopted [9]. This alternative provided a more stable and accurate fit to the thermistor data, significantly reducing approximation errors:

$$T_{(ratio)} = \frac{coef_1 \cdot ratio^2 + coef_2 \cdot ratio + coef_3}{ratio^2 + coef_4 \cdot ratio + coef_5} \tag{3}$$

The rational function avoided the oscillatory behavior observed with polynomial fitting and ensured consistent performance across the entire operational range. This improvement was critical for achieving precise and reliable temperature measurements in the EMIDSS-7 system.

VI. Experimental Results

Laboratory and field tests were conducted to validate the satellite communication and data processing performance of the EMIDSS-7 system.

Communication with the Iridium modem was confirmed to be successful, as measurements obtained with a logic analyzer demonstrated correct module responses. "Fig. 10" illustrates the response to an AT command during testing.

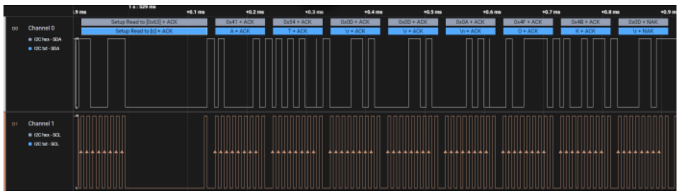

Fig. 10. Logic analyzer measurements showing modem response to an AT command.

Satellite transmission was tested in an open field with an unobstructed view of the sky to ensure good signal quality and establish a reliable connection to the Iridium satellite network. During these tests, the system successfully achieved a stable connection with the satellite network and environmental sensor data was transmitted without errors. The data packets, formatted in ASCII as described previously, were successfully relayed through the Iridium constellation and received via email at the designated endpoint. This confirmed the end-to-end functionality of the satellite communication link from data acquisition to remote delivery.

The email received included information about the satellite's location, such as latitude, longitude, and transmit time, along with the environmental sensor data. "Fig. 11" presents the dataset acquired during the transmission tests.

IMEI: 301434061309000
MOMSN: 41
Transmit Time: 2025-07-03T17:20:22Z UTC
Iridium Latitude: 20.5940
Iridium Longitude: -104.3156
Iridium CEP: 97.0
Iridium Session Status: 2
Data:
5432372e3639204834362e333120503834342e3834
T27.69 H46.31 P844.84

Fig. 11 Email data from satellite trajectory.

To evaluate thermal resilience, the system was exposed to low temperatures using a temperature chamber. The modem maintained stable uplink communication during these tests, confirming its operational reliability under temperature variations comparable to those observed under typical flight conditions.

Regarding the thermistor temperature values, Equation (4) exhibited minimal error when fitting the expected mathematical model, making it well-suited for the robustness and processing efficiency required by the embedded system. This behavior is illustrated in "Fig. 12".

$$T = \frac{-53.88 \cdot ratio^2 + 175.05 \cdot ratio + 60.14}{ratio^2 + 5.43 \cdot ratio + 0.34} \quad (4)$$

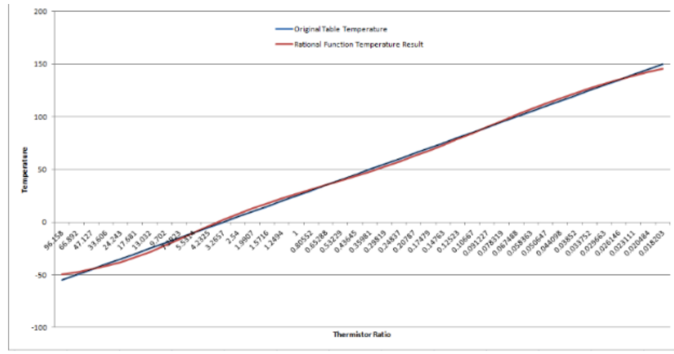

Fig. 12. Linearization results of the temperature behavior with respect to the thermistor ratio.

Finally, the assembled board was integrated into the payload container and connected to the power supply system in preparation for launch. "Fig. 13" displays the final structure of the EMIDSS-7 system

Fig. 13. Fully integrated EMIDSS-7 system.

VII. Conclusion and Future Work

The seventh version of EMIDSS integrates satellite message transmission, enhancing the system's capability for reliable remote data collection. This advancement supports more precise temperature measurements, particularly under low-temperature conditions.

For future work, the embedded system is expected to support the reception of satellite messages to allow remote control capabilities. In addition, integrating a real-time operating system (RTOS) has been proposed to improve the performance of the satellite communication subsystem, optimize data querying, and support additional functionalities.

VIII. REFERENCES

[1] S. Ishaque, N. Ullah, and M.-H. Kim, "A comparative analysis of machine learning techniques for predicting the performance of microchannel gas coolers in CO_2 automotive air-conditioning systems," *Energies*, vol. 17, no. 20, p. 5086, Oct. 2024, doi: 10.3390/en17205086.

[2] J. C. Aristizábal Aristizábal, M. Mancilla Torres, M.A. Medrano Acosta, N. Guadalupe Ortiz, C. A. Zepeda López, L. Rizo Domínguez, "Design of an automotive embedded system for stratospheric environments", de DESI -*Trabajos de fin de especialidad en Sistemas Embebidos*, Inst. Tecnol. Estud. Super. Occidente, Tlaquepaque, 2024.

[3] A. J. Fernández Soler, "Ascent phase thermal study of stratospheric balloon-borne payloads," Ph.D. dissertation, E.T.S. de Ingeniería Aeronáutica y del Espacio, U. Politécnica de Madrid, Apr. 2024.

[4] NXP Semiconductors, "S32K1xx Reference Manual", Rev. 14, Chapter 54: Flexible I/O (FlexIO), Sep. 2021.

[5] Ground Control, "Global Satellite Communication for Qwiic Projects from SparkFun," *Ground Control Blog*, Jul. 6, 2023. [Online]. Available: https://www.groundcontrol.com/blog/global-satellite-communication-for-qwiic-projects-from-sparkfun/

[6] SparkFun Electronics, *SparkFun IridiumSBD I2C Arduino Library Documentation*, GitHub, [Online]. Available: https://github.com/sparkfun/SparkFun_IridiumSBD_I2C_Arduino_Library/tree/main/documentation

[7] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[8] T. J. Rivlin, *An Introduction to the Approximation of Functions*, Dover Publications, 1981.

[9] L. L. Schumaker, *Spline Functions: Basic Theory*, Cambridge University Press, 2007.