Is it possible that microcode emerged in the time tunnel?

Product Overview

In an era where digital communication protocols evolve rapidly, the question arises: is microcode programming, reminiscent of the old PDP-11 days, still relevant in 2026? The answer is intriguingly both yes and no. While the legendary engineers Tony and Doug may not be around to witness it (as humorously hinted by the "Microcode_Time_Tunnel.png"), microcode itself quietly persists, powering modern embedded solutions in surprising ways.

One such example is the TrigoPi strip sensors product line STS-XX, which combines a 1-wire front end with I2C sensors including temperature (STS-TE), humidity (STS-HU), and airflow velocity (STS-AF). Bridging the gap between the fundamentally different 1-wire and I2C protocols is a sophisticated microcode-driven device: the Analog Devices DS28E18. This device acts as a protocol translator, leveraging microcode to manage complex communication sequences efficiently and cost-effectively.

Figure 1: A lighthearted take on the absence of original microcode pioneers (Microcode_Time_Tunnel.png).

Technical Highlights

The DS28E18 is a remarkable example of microcode's continued relevance. It employs a 512-byte sequencer that can be programmed to perform I2C transactions such as Start, Stop, and Re-Start, as well as SPI and GPIO operations. This sequencer is a highly efficient state machine, capable of executing a rich command set including reading and writing data to specific addresses, adding programmable delays, and running code from memory addresses. Although debugging capabilities are limited, the device's design prioritizes low cost and industrial robustness.

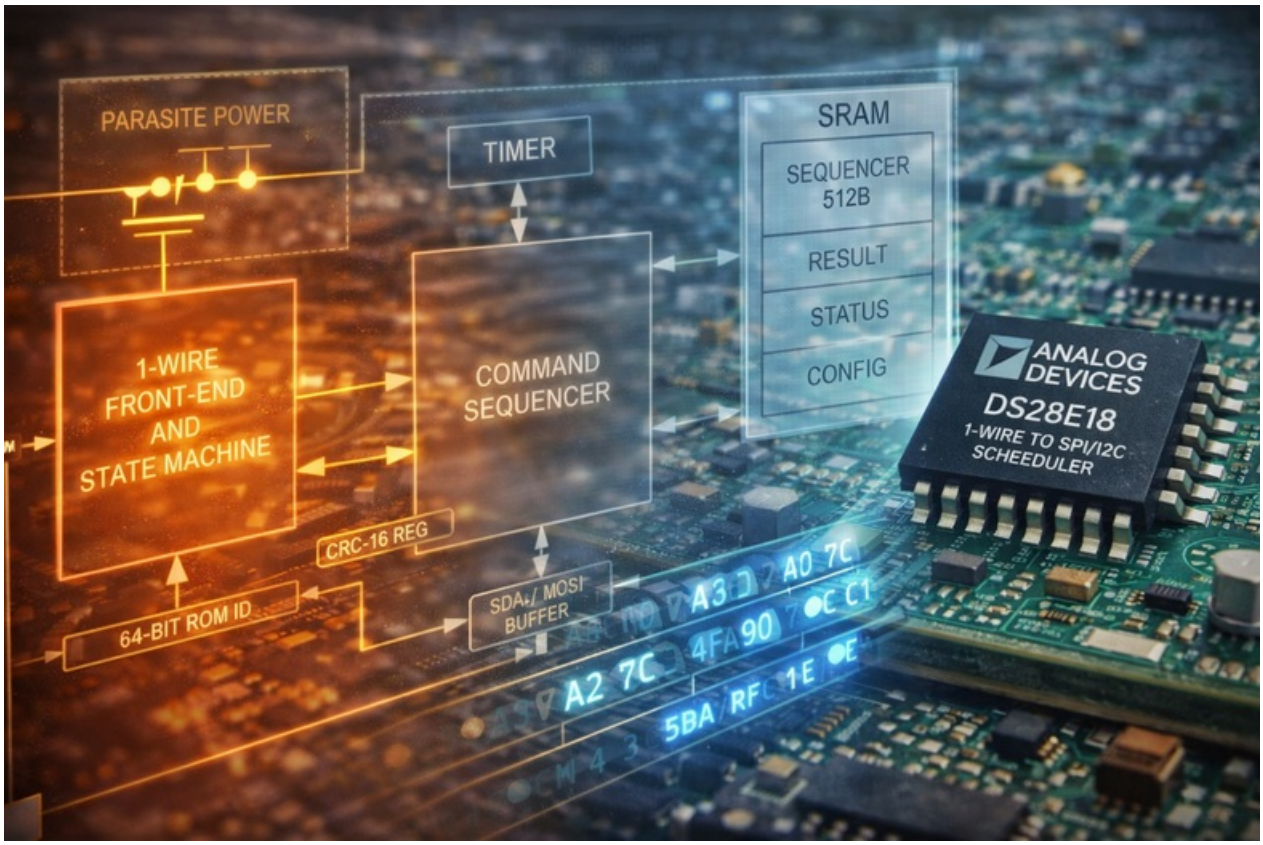

Figure 2: The DS28E18 sequencer architecture illustrating microcode-based control of communication protocols (DS28E18_Sequencer.png).

Key technical features include:

- Fully standard 1-wire front end supporting standard 11 kHz and overdrive 90 kHz speeds.
- Back-end support for I2C communication at 100 kHz, 400 kHz, and 1 MHz.
- Back-end SPI support up to 2.3 MHz and two GPIOs for flexible interfacing.
- Error protection through CRC-16 (polynomial X16 + X15 + X2 + 1) for all transactions, ensuring data integrity.
- Operation across a wide industrial temperature range from -40°C to +85°C.

Command Sets Overview

1-WIRE ROM Function commands

- READ ROM (0x33)
- MATCH ROM (0x55)
- SEARCH ROM (0xF0)
- SKIP ROM (0xCC)
- RESUME (0xA5)
- OVERDRIVE SKIP (0x3C)
- OVERDRIVE MATCH (0x69)

Device Function commands

- WRITE SEQUENCE (0x11)
- READ SEQUENCE (0x22)
- RUN SEQUENCE (0x33)
- WRITE CONFIGURATION (0x55)
- READ CONFIGURATION (0x6A)
- WRITE GPIO CONFIGURATION (0x83)
- READ GPIO CONFIGURATION (0x7C)
- DEVICE STATUS (0x7A)

Sequencer commands

I2C COMMANDS
- START (0x02)
- STOP (0x03)
- WRITE DATA (0xE3)
- READ DATA (0x04)
- READ DATA W/NACK END (0xD3)

SPI COMMANDS
- SPI WRITE/READ BYTE (0xC0)
- SPI WRITE/READ BIT (0xB0)
- SS HIGH (0x81)
- SS LOW (0x80)

UTILITY COMMANDS
- DELAY (0xDD)
- SENS_VDD ON (0xCC)
- SENS_VDD OFF (0xBB)
- GPIO_BUF WRITE (0xD1)
- GPIO_BUF READ (0xD2)
- GPIO_CNTL WRITE (0xE2)
- GPIO_CNTL READ (0x2E)

Figure 3: The rich command set of the DS28E18 sequencer enabling flexible protocol transactions (DS28E18_Commands_set.png).

Integration & Applications

The TrigoPi STS-TE temperature sensor showcases practical integration of the DS28E18 with a Texas Instruments TP117 temperature sensor. The DS28E18 acts as a protocol bridge, handling 1-wire communication on the front end and I2C interactions on the back end with the TP117. This allows for configuration register writes and temperature reads through the microcode sequencer's commands, effectively enabling seamless sensor management despite the differing protocols.

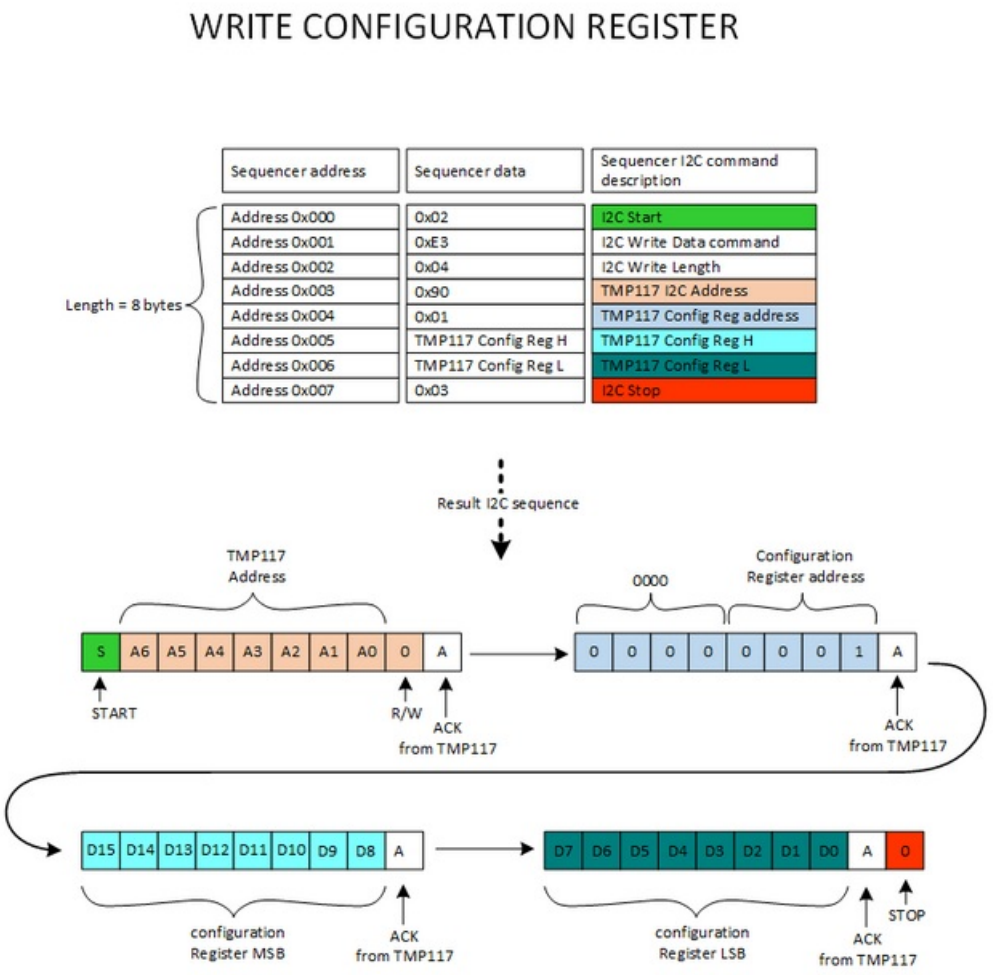

Figure 4: Writing configuration registers to the TP117 temperature sensor using DS28E18 microcode sequencing (DS28E18_TP117_Write_Configuration.png).

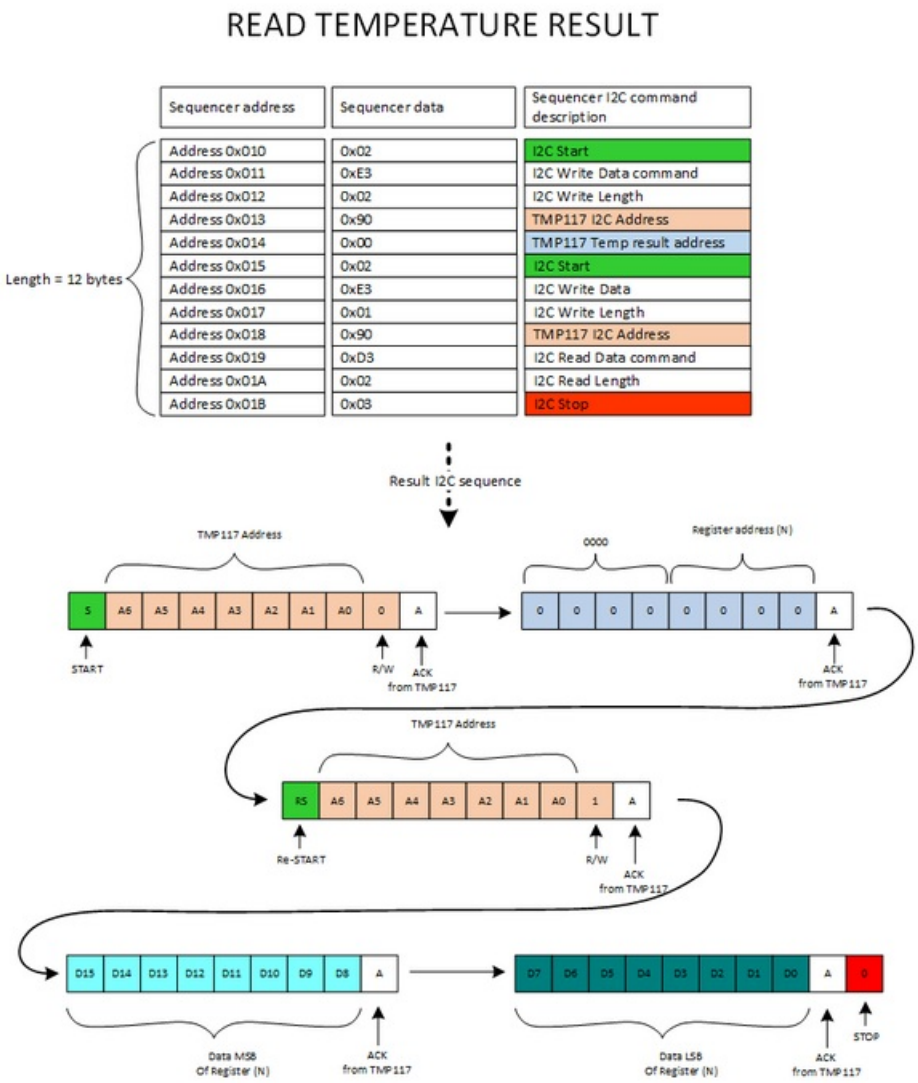

Figure 5: Reading temperature data from the TP117 sensor through DS28E18 microcode sequences (DS28E18_TP117_Read_Temperature.png).

This integration approach highlights how microcode can be leveraged to bridge legacy and modern communication protocols smoothly. The DS28E18's programmable sequencer enables custom transaction sequences tailored to the sensor's requirements, providing a flexible and robust solution for sensor interfacing challenges.

Variants & Support

The TrigoPi STS-XX sensor family extends beyond temperature sensing to include humidity and airflow velocity sensors, all benefiting from the same microcode-driven 1-wire to I2C/SPI translation architecture. This modular approach simplifies system design by standardizing the front-end interface while supporting diverse back-end sensors.

Engineers integrating these sensors can rely on comprehensive documentation and application notes available on the TrigoPi website, which provide detailed diagrams, command sequences, and configuration examples. This support accelerates development cycles and ensures reliable deployment in industrial environments.

Implementation Tips

- Ensure proper timing parameters are configured in the microcode sequencer to match the speed requirements of both 1-wire and I2C/SPI protocols.
- Utilize the CRC-16 error checking features to maintain data integrity, especially in electrically noisy industrial settings.
- Plan for adequate temperature range testing to confirm sensor and DS28E18 operation across expected environmental conditions.
- Leverage the programmable delays and command chaining in the sequencer to optimize communication

sequences and reduce bus contention.

Safety & Reliability Notes

The DS28E18's error detection via CRC-16 and robust industrial temperature rating contribute significantly to system reliability. However, designers should consider proper ESD protection and power sequencing to prevent damage during sensor installation or replacement. Additionally, thorough validation of microcode sequences is recommended to avoid communication deadlocks or unintended bus states.

TrigoPi’s expertise in integrating these components ensures that the sensors not only meet performance expectations but also maintain long-term reliability in demanding industrial applications.

Conclusion

While microcode programming might seem like a relic from the PDP-11 era, its practical utility in devices like the Analog Devices DS28E18 confirms its ongoing relevance. By enabling seamless protocol translation and sophisticated transaction control, microcode continues to play a vital role in modern sensor integration solutions such as the TrigoPi STS-XX product line.

If you are exploring sensor solutions that require flexible protocol bridging with proven reliability, consider the TrigoPi STS-XX family. We refined this approach at TrigoPi in real projects—reach out if you’d like to see how it applies to your case or to receive datasheets and evaluation kits.