

[1]

Disrupting I2C Communications

Seth Robles

2.671 Measurement and Instrumentation



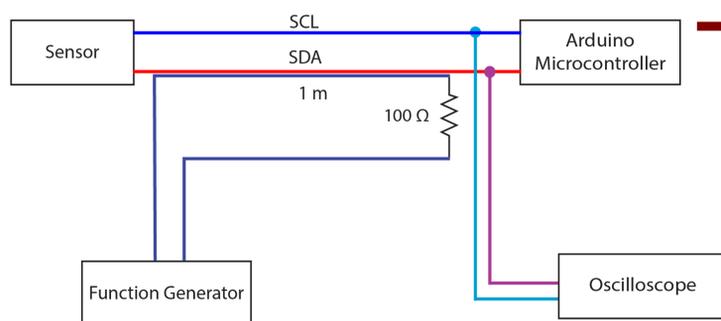
Abstract

Modern robotics oftentimes requires packing of electronics, making digital communication lines increasingly vulnerable to electromagnetic interference (EMI) from nearby components. This project investigated how EMI disrupts I2C communication between a microcontroller and a sensor. This microcontroller and sensor were laid near a cable loop driven by a function generator, which resulted in **coupled sinusoidal noise** of varying frequencies and amplitudes onto the I2C data line. The communication rate fell from a **nominal 50 Hz baseline** in standard conditions to a minimally observed 10 Hz at high disruption voltage and frequency. There is a significant drop in performance, with higher voltages having a drop in communication integrity.

Basics of Serial Communication and I2C

I2C is communication bus protocol used frequently in robotics between microcontrollers and peripherals. It uses a shared (SCL) and data line (SDA) with open-drain drivers and pull-up resistors, so devices pull the lines low and the resistors restore them high. Communication is organized into messages by START and STOP conditions, followed by 8-bit data bytes and an ACK or NACK bit from the receiver.

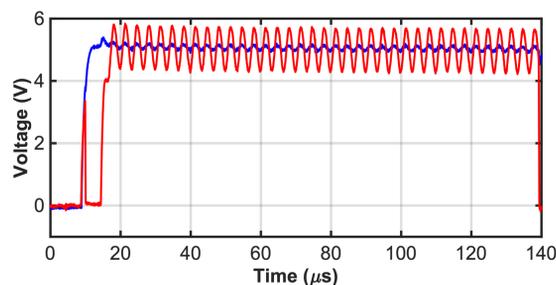
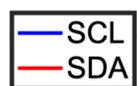
Experimental Method



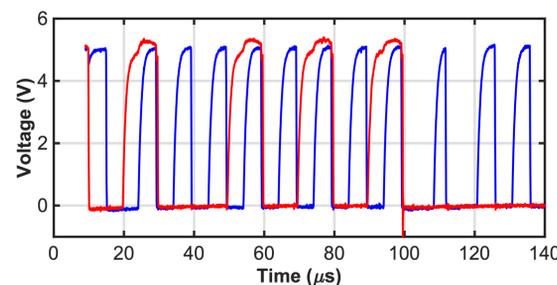
Data collection relied on Serial communication from the Arduino to a computer for 10 second samples

An oscilloscope probed the SDA and SCL lines to observe waveform abnormalities and failures

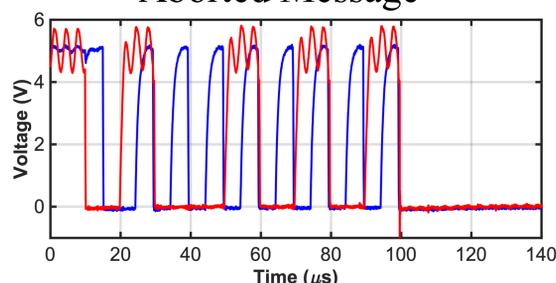
I2C Observed Waveform Behaviors



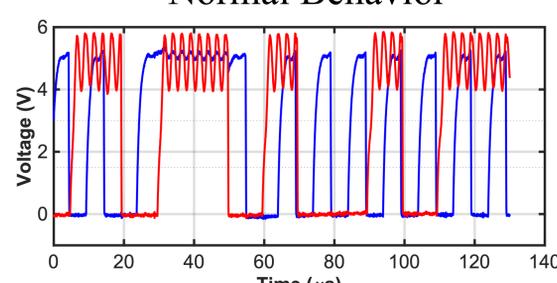
Aborted Message



Normal Behavior



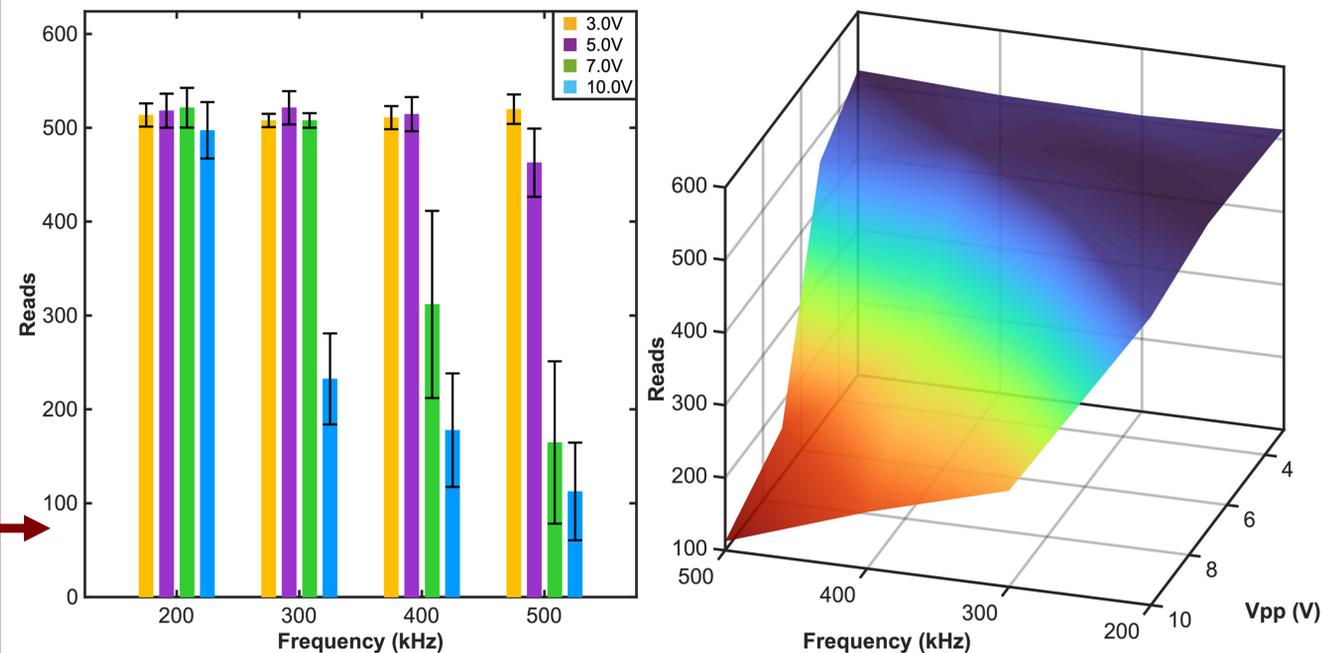
Pull Low Failure



Address Failure

Results

Arduino Reads within 10-second Time Frame



Conclusions

- From the waveform data, **3 modes of I2C failure** were observed, which corresponded with literature understanding of I2C communication failures due to EMI coupling
- At higher voltages and frequencies, **communication reaches a catastrophic shutdown**, with no back-and-forth communication between the IMU and microcontroller and requires constant rebooting
- At low voltages (3 and 5 Vpp), there was no significant change in communication integrity over a 10 second time frame
- Further work may involve a more rigorous characterization of failure modes, especially relating to the restart issues observed at catastrophic levels of interference

Acknowledgements

Thank you to Dr. Hughey, Dr. Hover, Dr. Halm, Dr. Chin, and Mr. Banzaert for guidance and feedback essential to the project

References

[1] Image: "Microcontroller Icon", Flaticon (licensed under Free License with Attribution). Available at https://www.flaticon.com/free-icon/microcontroller_2752877.