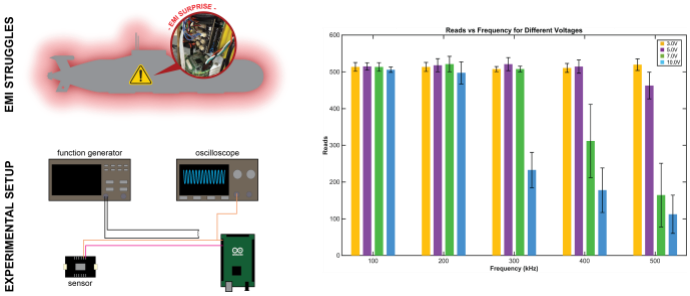
CHARACTERIZING I2C COMMUNICATION DISRUPTION WITH VARYING VOLTAGE AND FREQUENCY

Seth Robles
Massachusetts Institute of Technology
Cambridge, MA, USA

VISUAL ABSTRACT



ABSTRACT

Modern field robotics requires dense packing of electronics, making digital communication lines increasingly vulnerable to electromagnetic interference (EMI) from other electronic systems. This project investigated how EMI disrupts I2C communication between a microcontroller and a sensor. The microcontroller and sensor were placed 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, with communication performance assessed based on the microcontroller's received messages. 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. These results show that compact robotic systems using I2C communication should be intentional about separating essential communication lines from power or other lines with high voltages and switching speeds.

Keywords: Electromagnetic interference (EMI), Signal interference, I2C Communications, electrical noise, wire crosstalk

INTRODUCTION

Electromagnetic interference (EMI) refers to unwanted disturbances imparted by one electronic system on another, and it is becoming more relevant as systems like drones and UAVs push electronics into tighter and tighter spaces [1], [2]. These systems face more EMI challenges than normal consumer products since their tightly integrated electronics leave little room for shielding or filtering solutions [3]. Although extensive research has focused on EMI suppression techniques [4], [5], fewer studies have examined how interference arises under controlled conditions with isolated coupling mechanisms. Those that do have focused on communication protocols that are not I2C [6] or studied short bursts of interference [7]. Instead, this project examines the impact of an extended interference signal on I2C communication across a range of disturbance frequencies and voltage amplitudes.

A function generator produced sinusoidal signals of various frequencies and voltages, driving a wire placed next to the I2C data line to induce controlled interference. Communication integrity was evaluated using waveform observations, FFT measurements, and the microcontroller read dropouts, which together were used to assess the severity and the modes of interference. Although EMI can often be mitigated through shielding or twisted-pair wiring, these approaches are not always feasible in compact robotic platforms such as small AUVs (e.g., MIT's Spurdog class vehicles), where weight, connector constraints, and tight internal routing leave sensor buses physically exposed to nearby power electronics. As a result, understanding how unshielded I2C lines behave under sustained interference is directly relevant to real field-robotics environments.

BACKGROUNDO ON EMI AND I2C

EMI occurs when electromagnetic fields, which surround all circuits and electronic devices, couple with nearby systems, leading to signal distortion or reduced performance. This phenomenon can be explained with classical electromagnetism, particularly Faraday's law of induction. states that a changing magnetic flux will induce

an electromotive force (emf) in a conductor. This is the mechanism responsible for our understanding that any time-varying signal (such as an alternating current or rapid switching power device) can generate electrical noise in nearby circuits, which can result in dropped communications for electronics.

For I2C communication specifically, a predominant model for estimating the voltage induced by crosstalk is the Lumped RC Capacitive Crosstalk model, seen in equation 1. It can be simplified by assuming that the magnitude of $j\omega C_{12}R$ is much less than 1, resulting in equation 2. This simplified model assumes the cable behaves as a "short" transmission line, which generally requires the physical wire length to be less than about one-tenth of the wavelength of the highest significant frequency component [9]. At 100 kHz, the corresponding wavelength is roughly 3000 m, so the one-meter wire used in this experiment is well within the short-line regime.

$$V_{crosstalk} = \frac{j\omega C_{12}R}{1 + j\omega C_{12}R} V_1 \quad (1)$$

$$|V_{crosstalk}| = 2\pi f R C_{12} |V_1| \quad (2)$$

In equations (1) and (2), $V_{crosstalk}$ is the induced noise voltage on the victim line, V_1 is the aggressor-line voltage, C_{12} is the mutual capacitance between the two conductors, R is the victim-line termination resistance, ω is the interference's angular frequency, and $j = \sqrt{-1}$. Using this simplified version of the Lumped RC Capacitive Crosstalk model, the model predicts that the change in voltage will be proportional to the frequency.

BASICS OF I2C PROTOCOL

I2C is a two-wire serial protocol commonly used for communication between microcontrollers and peripheral sensors. It uses an open-drain architecture on both the serial data (SDA) and serial clock (SCL) lines, meaning devices can only pull the lines low, while pull-up resistors restore them to a logic-high level. Each transaction begins with a START condition, followed by an address byte, data bytes, and acknowledgment (ACK/NACK) bits, and concludes with a STOP condition. Because I2C relies on clean voltage transitions and samples data when SCL is high, its timing requirements make it sensitive to voltage disturbances and line distortion.

In this study, a packet refers to a single I2C transaction beginning with a START condition and ending with the corresponding STOP condition, including all address, data, and acknowledgement bits. A frame refers to the oscilloscope acquisition window, which may contain multiple packets depending on update rate and sampling duration.

I2C's sensitivity to voltage disturbances is especially relevant under electromagnetic interference. Small, induced voltages—such as those predicted by the capacitive crosstalk model—can slow rising edges, shift logic thresholds, or introduce unintended transitions on SDA or SCL [8]. These distortions commonly manifest as missed ACKs, corrupted data bytes, or bus stalls in which a device holds a line low. As a result, I2C provides a clear and easily observable platform for studying how EMI affects digital communication reliability.

EXPERIMENTAL DESIGN

The experimental setup (Figure 1) consisted of a Rigol DG4062 function generator, an Arduino Uno, an Adafruit BNO085 IMU, a Rigol DHO804 oscilloscope, and a connected computer. A BNC cable from the function generator was adapted into two one-meter banana cables that served as the signal and return lines, which were tied together at their far ends with a 100Ω resistor. The function generator operated in 50 Ω output mode with a floating ground.

The SDA and SCL lines between the Arduino and IMU were one-meter alligator cables connected to jumper cables for easy attachment to the Arduino and breadboard. To maximize coupling, the SDA line was placed in direct contact with the function generator's signal conductor and taped in place to ensure consistent spacing. The SCL and ground lines were routed farther away to minimize unintentional interference. The I2C bus ran at a 5-V logic level, with the BNO085 breakout board performing the necessary I2C voltage level shifting.

The oscilloscope probed the SDA and SCL lines, with all channels set to 10x attenuation and ground referenced to the Arduino circuit ground. A USB connection powered the Arduino, which then used its onboard 5V and GND pins to power the IMU through a breadboard.

The testing environment was a typical, ambient laboratory setting with potential background EMI from wall and ceiling lights. No other electronics were powered on near the testing area, and benchmark tests confirmed normal communication behavior.

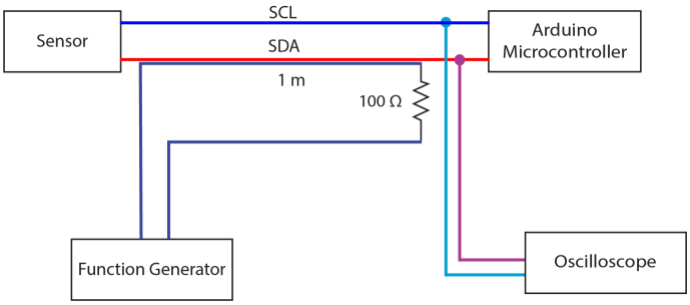


Figure 1: Wiring schematic of the project experimental setup. A function generator outputs a sinusoidal signal through a wire connected by a 100 Ω resistor. The signal line is routed adjacent to the sensor's data line to induce electromagnetic coupling. The microcontroller and sensor communicated over the SDA line close to the wire and an SCL far from it. The Arduino communicates to and is powered by the computer via USB.

WAVEFORM OBSERVATION SETUP

Waveform-level tests used oscilloscope settings with 20.00 ms/div time base, 31.25 MSa/s sampling rate, and 2 V/div vertical scale, recording 10 million points per acquisition. A sinusoidal interference signal was generated by the function generator to induce coupling on the SDA line. This signal was allowed to run for at least 5 seconds to ensure steady-state conditions before the Arduino was manually reset to reinitialize I2C communication.

These captures allowed direct visualization of how interference affected the electrical behavior of the bus. Data was exported via USB for post-processing in MATLAB, with each capture limited to approximately 0.16 seconds (containing up to 8 packets per acquisition) due to the oscilloscope's storage constraints. Using MATLAB, START and STOP bits were found in the waveform and observed for characterization of issues.

READS SETUP

Extended trials were performed using an Arduino program that collected 10 seconds of BNO085 sensor data at its standard update rate of 50 Hz (for a total of 500 reads), while monitoring for bus stalls. If communication paused for more than 400 ms, the Arduino was programmed to recover by re-initializing the I2C bus, which would take approximately another 100 ms. This was selected as an estimate of how a robotic system might handle a temporary loss of sensor data.

Ten independent runs were recorded per unique configuration of voltage and frequency. The number of successful I2C reads in each run was recorded as a metric of communication reliability, representing the sensor data that a host system would likely obtain under similar electromagnetic conditions.

4. RESULTS AND DISCUSSION

Waveform-level measurements were inspected for qualitative comparison of samples with low vs high noise injection. The first observable quality is high-frequency oscillations superimposed on the SDA line, which were absent during baseline communications. Additionally, in several cases, the SDA and SCL lines were momentarily shifted out of phase (as seen in Figure 2), producing conditions where the oscilloscope's I2C decoder failed to identify a valid start or stop bit. These qualitative features suggest that induced noise was interfering with clock-to-data timing margins, which are essential for reliable communication.

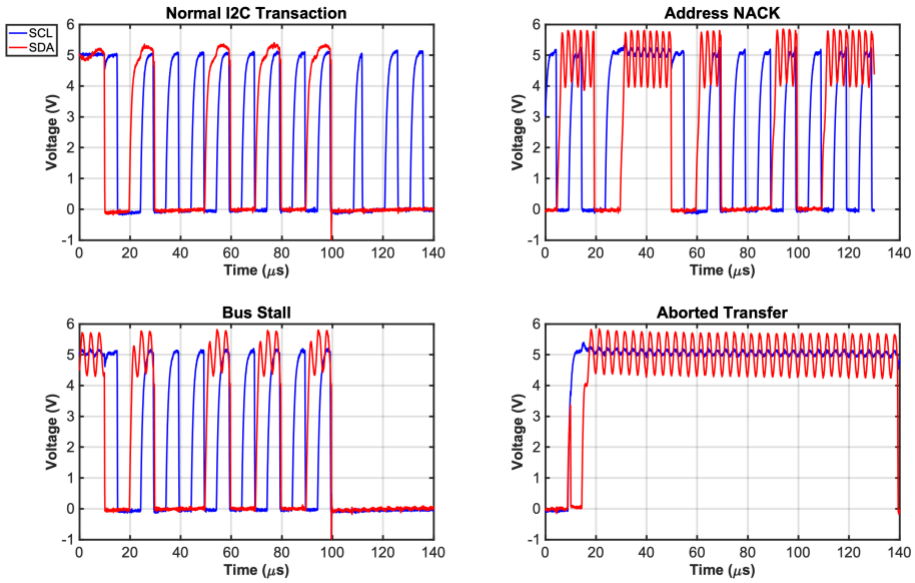


Figure 2: Representative SDA and SCL waveforms demonstrating the major I2C behaviors observed during EMI testing. The top-left plot shows a normal transaction. The top-right plot shows an address NACK event, where interference corrupts the address byte and the sensor fails to acknowledge. The bottom-left plot shows a bus stall, marked by the SDA line being held low, preventing further communication. The bottom-right plot shows an aborted transfer with noise causing communication to terminate prematurely

FREQUENCY DOMAIN ANALYSIS

Fourier transforms of each waveform confirmed the expected trend: higher function-generator amplitudes produced larger peaks at the injected frequencies. Table 1 summarizes the extracted FFT peak amplitudes for 3, 5, 7, and 10 Vpp disturbances.

Table 1: Measured FFT Peak Amplitudes for Induced Noise at Discrete Frequency and Voltage Levels

	100 kHz	200 kHz	300 kHz	400 kHz	500 kHz
3 Vpp	0.11017	X*	0.22589	0.21709	0.18938
5 Vpp	0.16985	0.21224	0.32964	0.39577	0.32964
7 Vpp	0.25715	0.31906	0.50919	X*	0.45226
10 Vpp	0.35134	0.43858	X*	0.59759	0.62226

*Values marked with an X and an asterisk were determined to have errors in post-processing, resulting in their FFT amplitudes being nulled.

These results show an approximately linear relationship between disturbance amplitude and induced voltage at the aggressor frequency – a result consistent with the simplified Lumped RC Crosstalk Model (Equation 2). The relationship with frequency is not linear across the range tested, but in general, higher frequencies and higher voltages tended to produce larger induced voltages. The non-linear relationship with frequency could reflect inductive or resistive effects in the wiring, however further study would likely be required to understand this mechanism more fully.

PACKET-LEVEL COMMUNICATION RELIABILITY

As shown in Figure 3 and 4 below, increasing voltage and frequency generally reduce the number of successful reads. Across 10 trials per setting, communication success remained high at low interference levels (for example, 3 Vpp at 100–200 kHz) where nearly all 50 Hz sensor reads were received. In contrast, at 10 Vpp and 400–500 kHz, many runs showed extended dropouts. In these cases, the Arduino frequently failed to receive ACK bits and triggered an automatic bus reset.

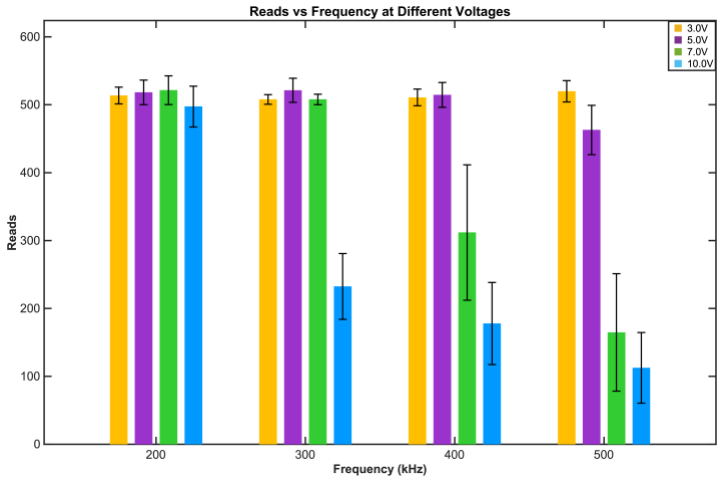


Figure 3: The bar chart shows an inverse relationship between microcontroller reads and increasing aggressor frequencies for aggressor voltages of 7 and 10 Vpp, especially in the 300 to 500 kHz range. It also shows no major relationship between the number of microcontroller reads and increasing aggressor frequencies for aggressor voltages of 3 and 5 Vpp. The error bars show a 95% confidence interval for the expected true mean of microcontroller reads for a 10 second time frame.

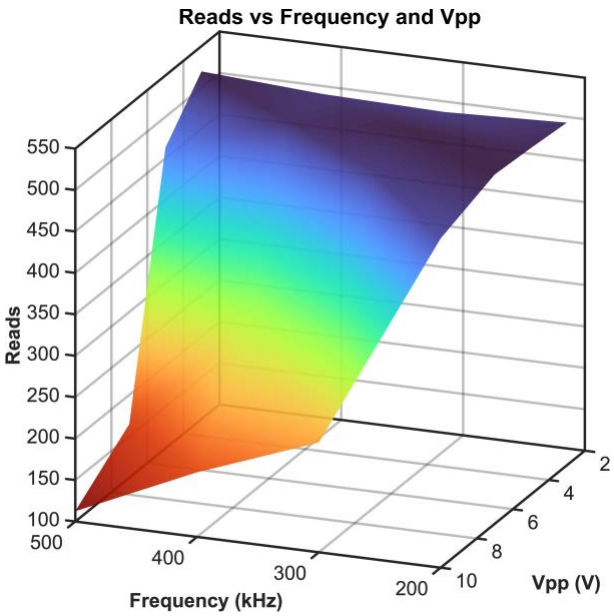


Figure 4: The surface plot maps out the relationship between microcontroller reads, frequency, and Vpp. The data demonstrates that the number of reads decreases significantly as both the frequency approaches 500 kHz and the voltage approaches 10 Vpp. The sharpest decline in reads is observed with both frequency and voltage at maximum values, while the highest number of reads occur in low-frequency and low-voltage regions.

This behavior is consistent with timing-margin shrinkage in the I2C protocol. Because I2C uses an open-drain architecture with pull-up resistors, the rising edges of SDA and SCL are particularly sensitive to capacitive coupling. Induced voltage from the aggressor line can intermittently accelerate or delay these edges, effectively violating setup and hold timing and causing bits to be misinterpreted.

A second, more nonlinear effect appears at high interference levels: once communication fails severely enough to trigger a bus reset, the 400 ms recovery time dominates the packet-count metric. In other words, beyond a certain noise level, it is not just that packets are occasionally corrupted—the bus can effectively go "down" for a noticeable fraction of the run.

LIMITATIONS AND FUTURE WORK

The precision of these results is limited by the experiment's sensitivity to wire geometry, as small changes in spacing between the SDA line and the aggressor alter the mutual capacitance C_{12} and the induced voltage. Additionally, only ten runs per setting were collected, limiting detection of infrequent failures. The parameter space was also coarse, so the findings should primarily be interpreted as qualitative trends or rough benchmarks as opposed to precise operating limits.

Future work could potentially use finer resolution in frequency and voltage, more repetitions, and controlled variations in wire length and routing to quantify how layout affects susceptibility in real robotic systems. A more complete analytical model coupling the Lumped RC Crosstalk framework with I2C timing constraints and switching signals similar to those in robot power systems could better reflect the interference environments seen in compact AUV and robotic platforms.

CONCLUSIONS

This project examined how sustained electromagnetic interference affects I2C

communication by varying disturbance frequency and amplitude. Across all trials, induced noise on the SDA line increased with the aggressor's voltage amplitude, and to a lesser extent, with frequency. As interference strengthened, packet-level reliability fell sharply, with low-noise conditions producing all expected reads, while high-noise conditions led to address NACKs, corrupted transactions, and bus stalls. These failure modes are consistent with expected I2C behavior under voltage disturbances and with the predictions of the Lumped RC Crosstalk Model.

These results have direct relevance to compact robotic and AUV systems, where I2C-based IMUs and sensors may be routed near high-current, high-frequency conductors that drive thrusters or DC-DC converters. The experiments demonstrate that sustained EMI at levels comparable to those produced by these subsystems can reduce effective sensor data rates by a factor of five or more and trigger full communication dropouts. This implies that cable routing, separation, and shielding of aggressor lines may be as critical to mission reliability as sensor and controller hardware.

While a denser set of test points would allow for more precise modeling, the experimental findings already show the practical impact of EMI on the widely used I2C protocol. This highlights that in tightly packed robotic platforms, electrical layout is not merely an implementation detail, but an essential factor in maintaining robust sensing during operation.

ACKNOWLEDGMENTS

The author gratefully acknowledges essential experimental advice from Steve Banzaert, Dr. Harrison Chin and Raymond Turrisi when designing the experiment and searching through the literature on EMI and I2C bus protocol. Additionally, thank you Dr. Barbara Hughey, Dr. Franz Hover, Dr. Matthew Halm, and Claire Berman for continued help for communication advice in all aspects of this project.

REFERENCES

[1] Frost & Sullivan, 2024, *Growth Opportunities in Unmanned Underwater Vehicles, 2023–2032*, Frost & Sullivan Market Research Report. (Accessed Nov. 7, 2025).

[2] Frost & Sullivan, 2024, *US Military Unmanned Aerial Vehicles: Growth Opportunities*, Frost & Sullivan Market Research Report. (Accessed Nov. 7, 2025).

[3] Zhang, Z., Zhou, Y., Zhang, Y., and Qian, B., 2024, "Strong Electromagnetic Interference and Protection in UAVs," *Electronics*, **13**(2), p. 393. https://doi.org/10.3390/electronics13020393.

[4] Lu, Z., Chen, Y., and Xue, Y., 2024, "Overview of Key Techniques for In Situ Tests of Electromagnetic Radiation Emission Characteristics," *Sensors*, **24**(23), p. 7515. https://doi.org/10.3390/s24237515.

[5] Zhang, Z., Hu, Y., Chen, X., Jewell, G. W., and Li, H., 2021, "A Review on Conductive Common-Mode EMI Suppression Methods in Inverter-Fed Motor Drives," *IEEE Access*, **9**, pp. 18345–18360. https://doi.org/10.1109/ACCESS.2021.3054514.

[6] Kim, S.-G., Lee, E., Hong, I.-P., and Yook, J.-G., 2022, "Review of Intentional Electromagnetic Interference on UAV Sensor Modules and Experimental Study," *Sensors*, **22**(6), p. 2384. https://doi.org/10.3390/s22062384.

[7] Jang, J., Cho, M., Kim, J., Kim, D., and Kim, Y., 2023, "Paralyzing Drones via EMI Signal Injection on Sensory Communication Channels," *2023 Network and Distributed System Security Symposium (NDSS)*. https://doi.org/10.14722/ndss.2023.24616.

[8] Tang, K. T., and Friedman, E. G., 1999, "Peak Crosstalk Noise Estimation in CMOS VLSI Circuits," *Proceedings of the 6th IEEE International Conference on Electronics, Circuits and Systems (ICECS '99)*, pp. 1539–1542. https://doi.org/10.1109/ICECS.1999.814464.

[9] Paul, C. R., 2007, *Analysis of Multiconductor Transmission Lines*, 2nd ed., Wiley-Interscience, Hoboken, NJ, pp. 41–42. https://doi.org/10.1002/0470120794.