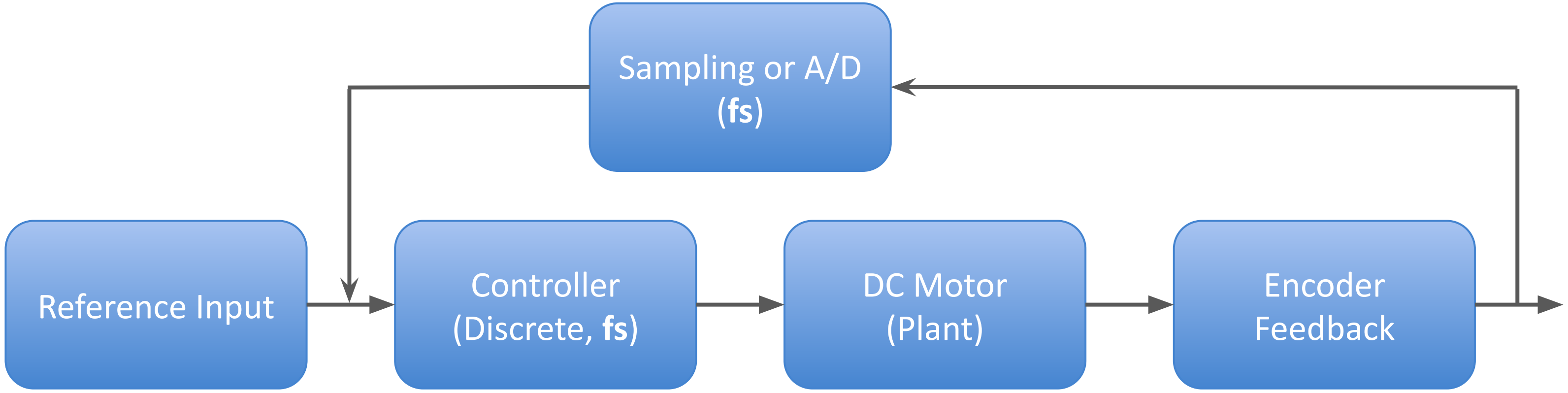
How Sampling Frequency Shapes DC Motor Control Performance

Evaluating RMS Tracking Error and Transient Response in a Closed-Loop DC Motor Speed Control System

UC SANTA BARBARA
College of Engineering

How does the controller sampling frequency affect the RMS tracking error and transient response in a closed-loop DC motor speed control system?



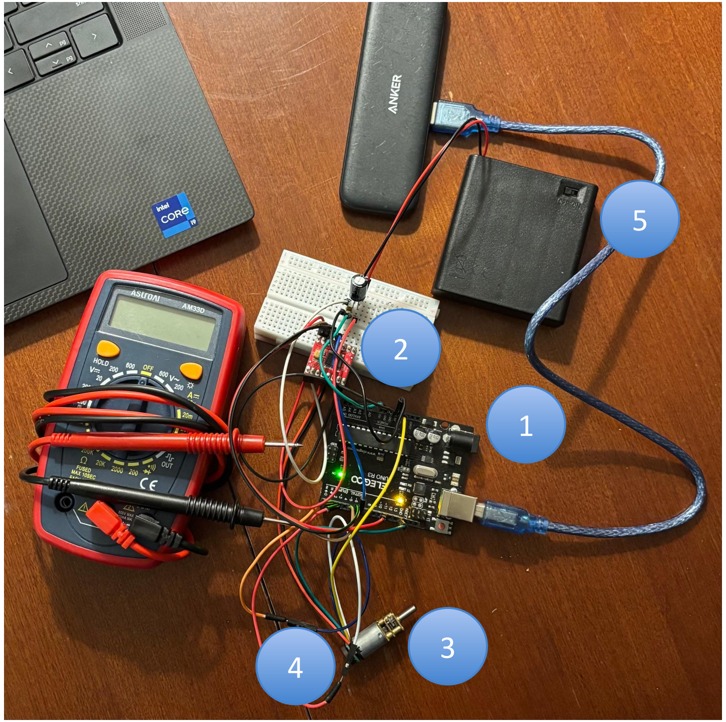
References

- K. J. Åström & R. M. Murray, *Feedback Systems*
- K. Ogata, *Discrete-Time Control Systems*
- R. J. M. Oosterbosch, *Effect of Sampling Frequency on Digital Controllers*

What Phenomenon Is Measured

- The effect of controller sampling frequency on closed-loop motor speed regulation
- How the update rate of a discrete-time PI controller influences:
  - Transient response (overshoot, settling time)
  - Tracking accuracy measured via RMS velocity error
- Motor response to a step change in reference speed (0 $\longrightarrow$ 4 rad/s)

Hardware Setup

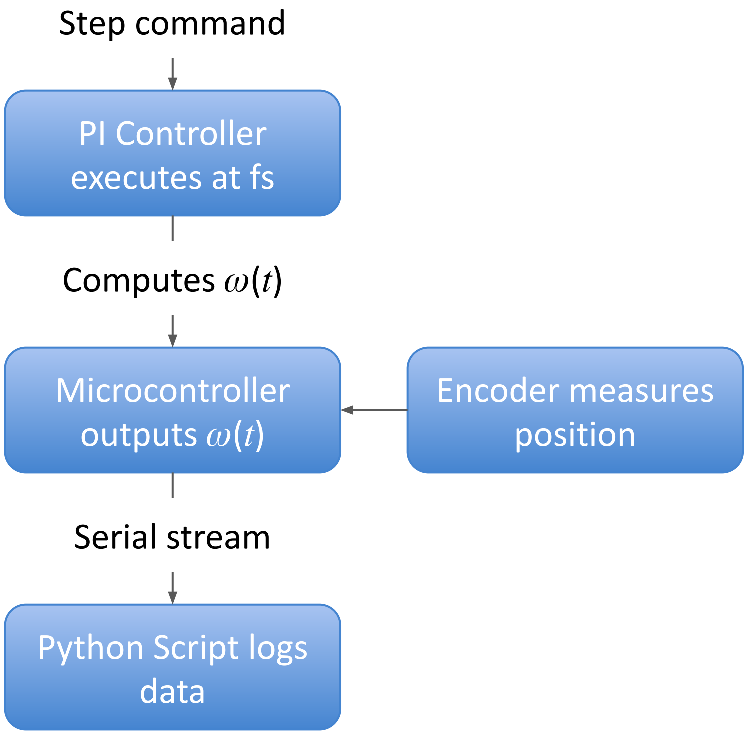


1. **Microcontroller**: runs discrete-time PI at selected sampling frequencies
2. **Motor Driver (H-bridge)**: supplies current and enables bidirectional control
3. **DC Motor**: plant whose velocity response is measured
4. **Quadrature Encoder**: provides high-resolution speed feedback
5. **External Power Supply**: isolates motor power from logic electronics

UC SANTA BARBARA
College of Engineering

How Measurements Are Conducted

- Encoder data is converted to motor velocity $\omega(t)$ on the microcontroller
- PI controller runs at sampling periods Ts = 10, 15, 20, 50, or 100 Hz
- System is excited with a step from 0 $\longrightarrow$ 4 rad/s
- Measured velocity is logged at 200 Hz
- Each sampling frequency is tested separately; CSV data computes RMS error and transient metrics



Data Analysis

Tracking Error

$$e(t) = \omega_{\mathrm{ref}}(t) - \omega(t)$$

RMS Tracking Error

$$\mathrm{RMS} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} e^2(t_i)}$$

- Transient metrics extracted using step response analysis:
  - Percent overshoot
  - Settling time (±10%)
- Results averaged over 3 trials per sampling frequency

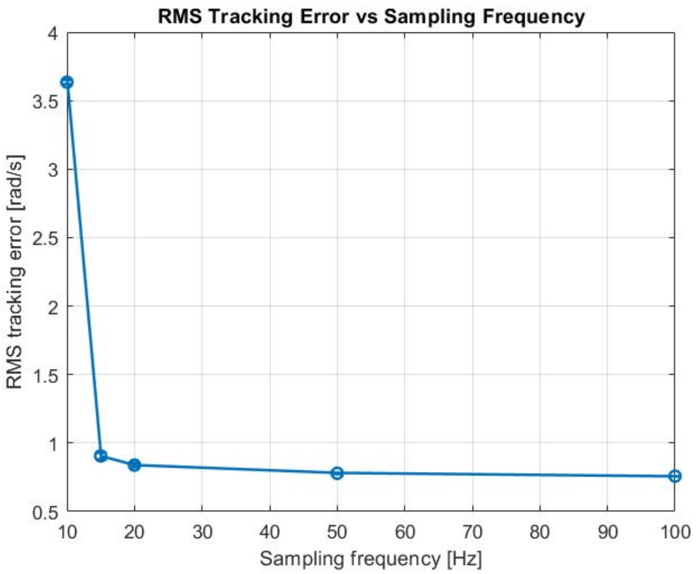
Measurement Uncertainty

- Encoder resolution limits angular velocity precision
- Sampling jitter changes the effective time interval Δt
- Serial timestamp alignment introduces small timing offsets
- Controller quantization limits the command resolution, which becomes more noticeable at lower sampling frequencies

$$u_{\mathrm{RMS}} = \sqrt{\sum \left(\frac{\partial \mathrm{RMS}}{\partial x_i} u_{x_i} \right)^2}$$

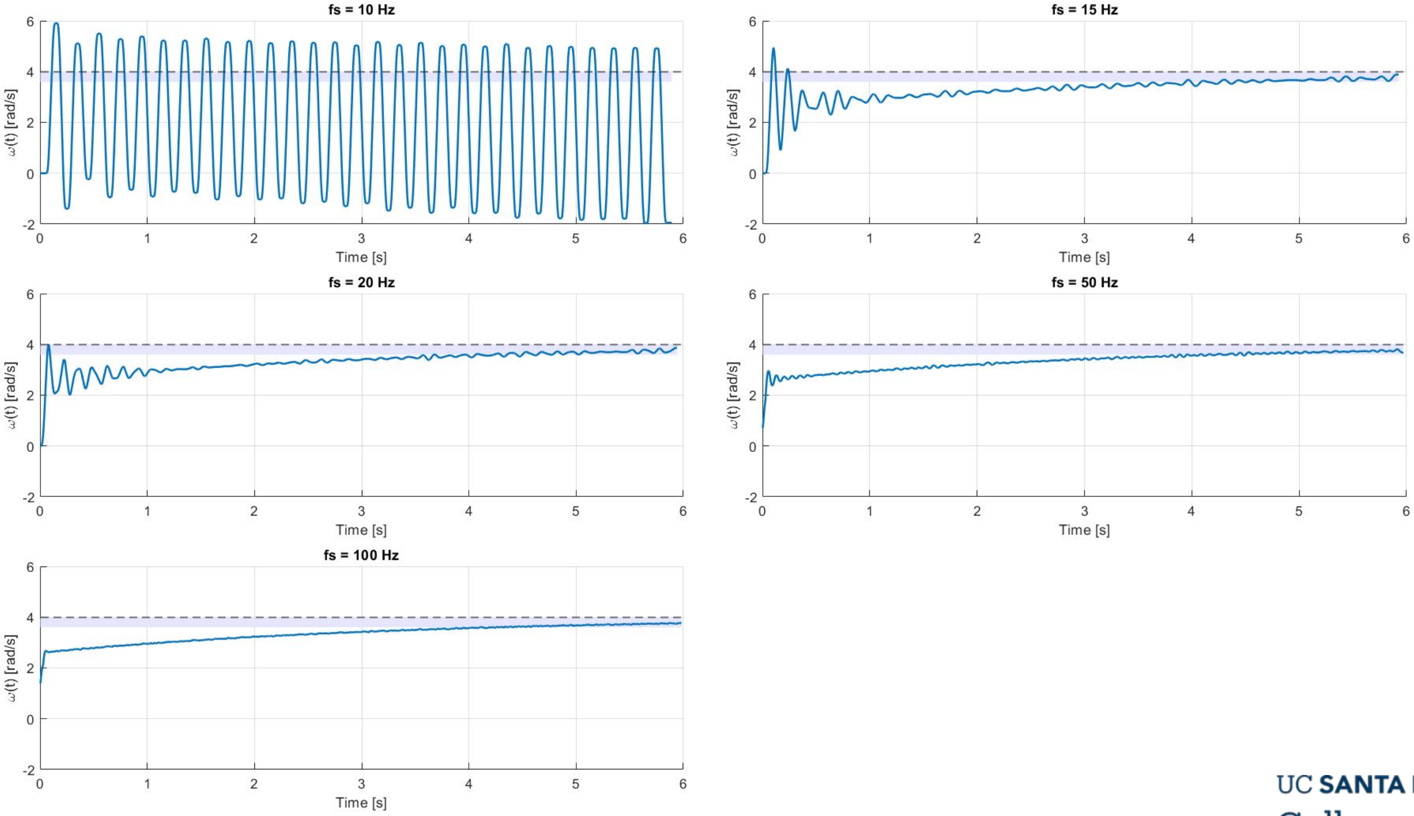
$$u_{\mathrm{RMS}} = \mathrm{std}(\mathrm{RMS}_{\mathrm{error}})$$

Results Overview



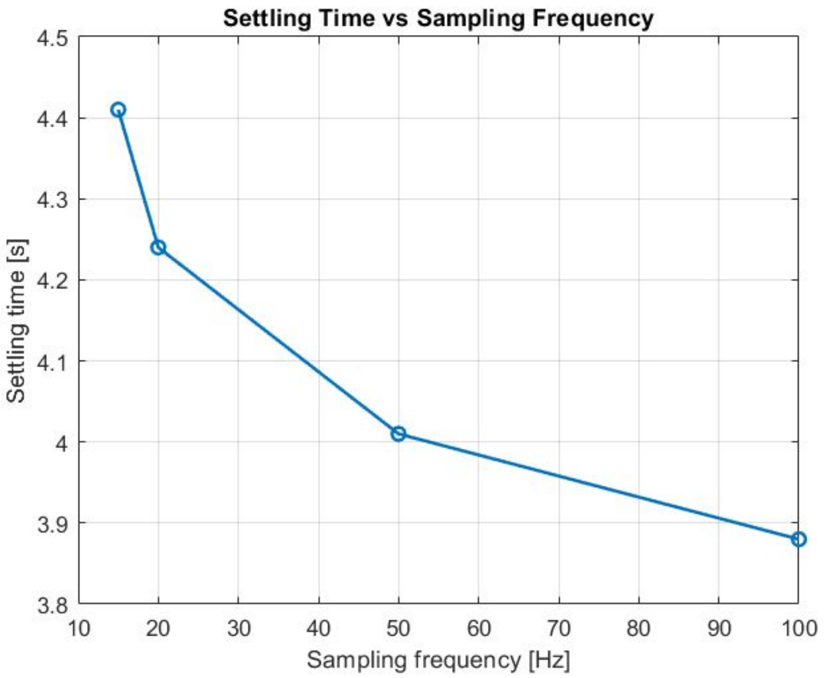
- RMS tracking error decreases as sampling frequency increases
- Lower frequencies (10–20 Hz) exhibit visible oscillations
- High frequency sampling (≥50 Hz) yields faster convergence and minimal overshoot
- Performance plateaus beyond 50 Hz → diminishing returns

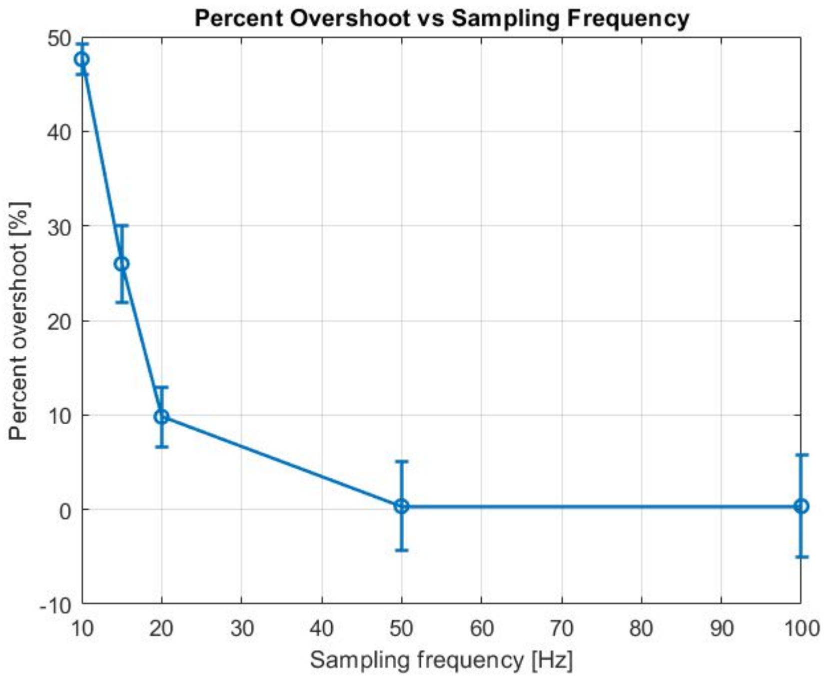
Higher Sampling Frequencies Improve Performance



UC SANTA BARBARA
College of Engineering

Transient Response Metrics





What The Results Mean

- Higher sampling frequencies allow the controller to react faster, reducing lag
- At low sampling frequencies, discretization delays dominate → sluggish tracking
- Diminishing performance improvements beyond ~50 Hz → controller bandwidth limit

Increasing the sampling frequency improves closed-loop velocity tracking, but only up to a practical limit (~50 Hz). Thus, sampling frequency should be selected based on controller bandwidth, not arbitrarily increased.

Role of Uncertainty

- Error bars represent trial-to-trial variability in RMS and percent overshoot
- Low variance at ≥ 50 Hz ⟶ results are robust
- Higher variance at 10–20 Hz ⟶ controller is less stable ⟶ metrics more sensitive to disturbances
- Uncertainty does not change the conclusion; trend remains clear

Sources of Error

- Serial logging delay + OS jitter
- Encoder quantization and missed pulses at high speeds
- Motor friction heating over long trials
- Noise in supply voltage affecting H-bridge output

Improvements & Future Work

- Test beyond 100 Hz to determine true controller bandwidth limit
- Add current sensing to analyze torque response
- Use real-time logging on microcontroller to remove serial overhead
- Tune PID rather than PI for better low-frequency performance

Optimizing sampling frequency is a practical and impactful lever for improving motor control performance.