

6. 정규 속도 단계별 상승 실험 보고서

1. 실험 개요

- 실험 목적:** 표준 보드레이트(1200 ~ 115200 bps)를 단계별로 상승시키며 **소프트웨어 비트뱅(Bit-bang)** 방식의 송신 안정성을 검증하고, 속도 변화에 따른 **비트 폭(Bit Width)**의 물리적 변화와 통신 한계점을 확인한다.
- 연결 구성:** Arduino Uno D8 (Software TX) ↔ D0 (Hardware RX) [Loopback]
- 측정 방법:** 각 속도별로 5개의 테스트 문자(a , A , 0 , 9 , !)를 3회 반복 송신(총 15회)하여 수신 성공률을 측정.

2. 실험 결과 데이터 (Summary)

Baud Rate (bps)	Bit Width (이론값, μs)	Total Tests	Success	Errors	No Data	Success Rate (%)	비고
1200	833.33	15	15	0	0	100.00%	매우 안정적
2400	416.67	15	15	0	0	100.00%	매우 안정적
4800	208.33	15	15	0	0	100.00%	매우 안정적
9600	104.17	15	15	0	0	100.00%	안정적
19200	52.08	15	15	0	0	100.00%	안정적 (실용 한계)
38400	26.04	15	6	9	0	40.00%	불안정 (데이터 깨짐)
57600	17.36	15	0	15	0	0.00%	수신 불가
115200	8.68	15	0	15	0	0.00%	수신 불가

참고: 19200 bps까지는 데이터 무결성이 보장되나, 38400 bps부터 급격한 성능 저하가 발생 함.

3. 결과 분석

A. 저속 구간(1200 ~ 19200 bps)의 안정성

측정된 데이터에 따르면 1200 bps부터 19200 bps까지의 모든 테스트에서 **성공률 100.00%**를 기록하였다.

- **비트 폭의 여유:** 19200 bps 기준 비트 폭은 약 **52.08 μs**이다.
- **소프트웨어 처리 능력:** 아두이노(ATmega328P, 16MHz)의 GPIO 제어 및 루프 오버헤드(Overhead)는 수 μs 수준이다. 50 μs 이상의 비트 폭 내에서는 이러한 오버헤드가 통신 타이밍 오차에 큰 영향을 주지 않아 안정적인 통신이 가능하다.

B. 고속 구간(38400 bps 이상)의 한계 및 실패 원인

38400 bps부터 성공률이 40%로 급락하고, 57600 bps 이상에서는 통신이 불가능하였다.

- **타이밍 마진 부족:** 38400 bps의 비트 폭은 **26.04 μs**이다. 소프트웨어 비트뱅 방식에서 `delayMicroseconds()` 함수 호출과 포트 조작 명령어 실행 시간(Instruction Cycle)이 전체 비트 폭에서 차지하는 비중이 커지면서 타이밍 오차(Jitter)가 누적된다.
- **샘플링 불일치:** 수신 측(Hardware UART)은 정확한 타이밍에 샘플링을 시도하지만, 송신 측(Software)의 미세한 지연으로 인해 Start Bit와 Data Bit의 경계가 모호해져 프레임 에러(Frame Error)가 발생한 것으로 분석된다.

C. 전송 속도와 비트 폭의 상관관계

$$T_{bit} = \frac{1}{\text{Baud Rate}}$$

보드레이트가 증가함에 따라 비트 폭은 반비례하여 감소하며, 비트 폭이 줄어들수록 소프트웨어 방식의 타이밍 제어 난이도가 기하급수적으로 상승한다.

4. 결론

본 실험을 통해 **소프트웨어 비트뱅 방식의 유효 대역폭은 최대 19200 bps**임이 확인되었다.

- **권장 속도:** 안정적인 데이터 전송을 위해서는 **9600 bps 이하** 사용을 권장한다.
- **한계 속도:** 환경에 따라 **19200 bps**까지 사용 가능하나, 그 이상의 속도(38400 bps~)가 필요한 경우 반드시 **하드웨어 UART**를 사용해야 한다.