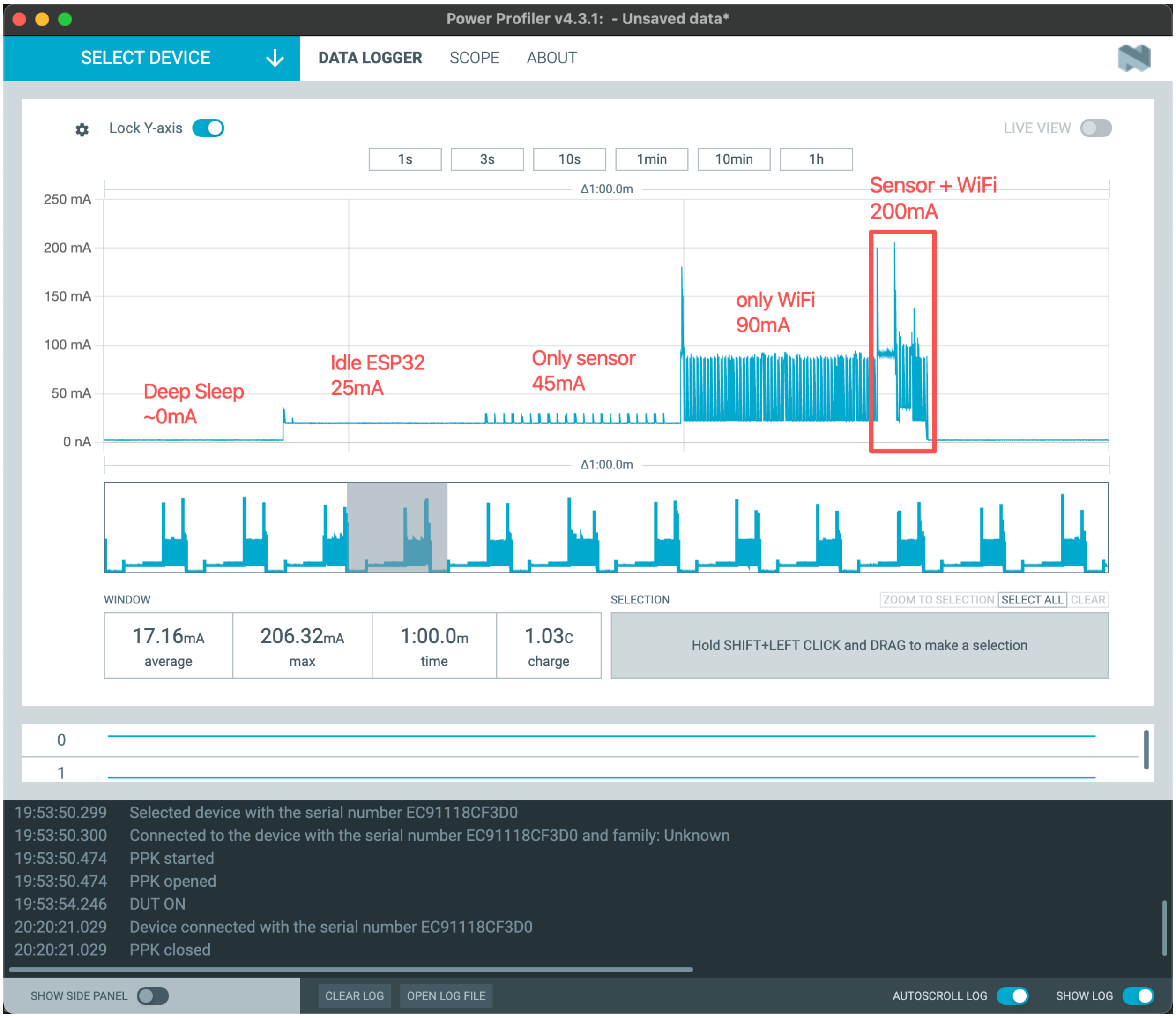
Lab 5 - Power Management Lab

Zihan Yang

1. Screenshot of your firebase RTDB receiving the ultrasonic sensor readouts.

2. Annotated screenshot of the plot on your power profiler app to show the power consumption in 5 different stages of ESP32S3's usage.

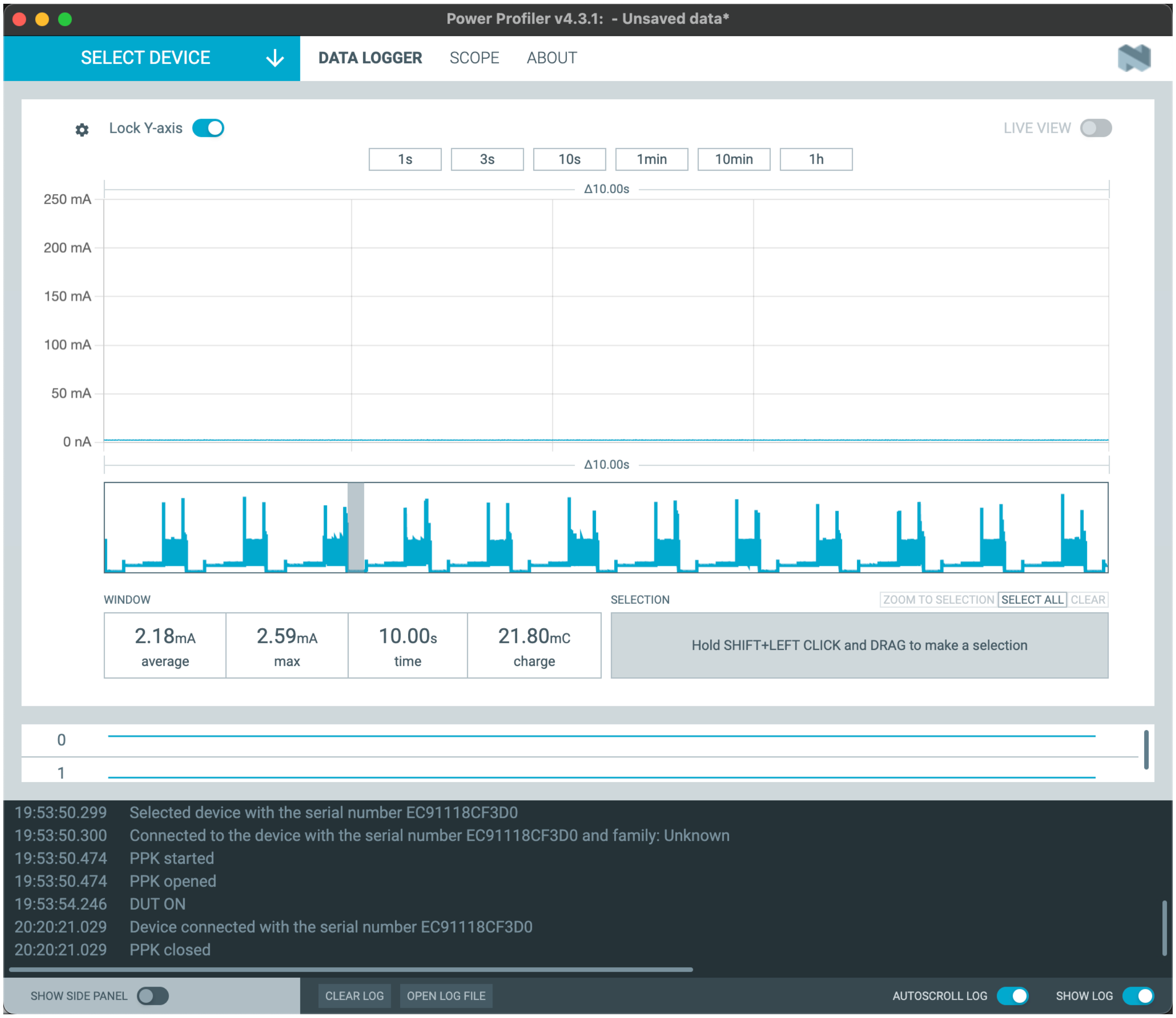


3. Calculations of each stage's power consumption and estimated battery-lasting time, respectively.

Stage 0: Deep Sleep mode

Average Current: 2.18mA

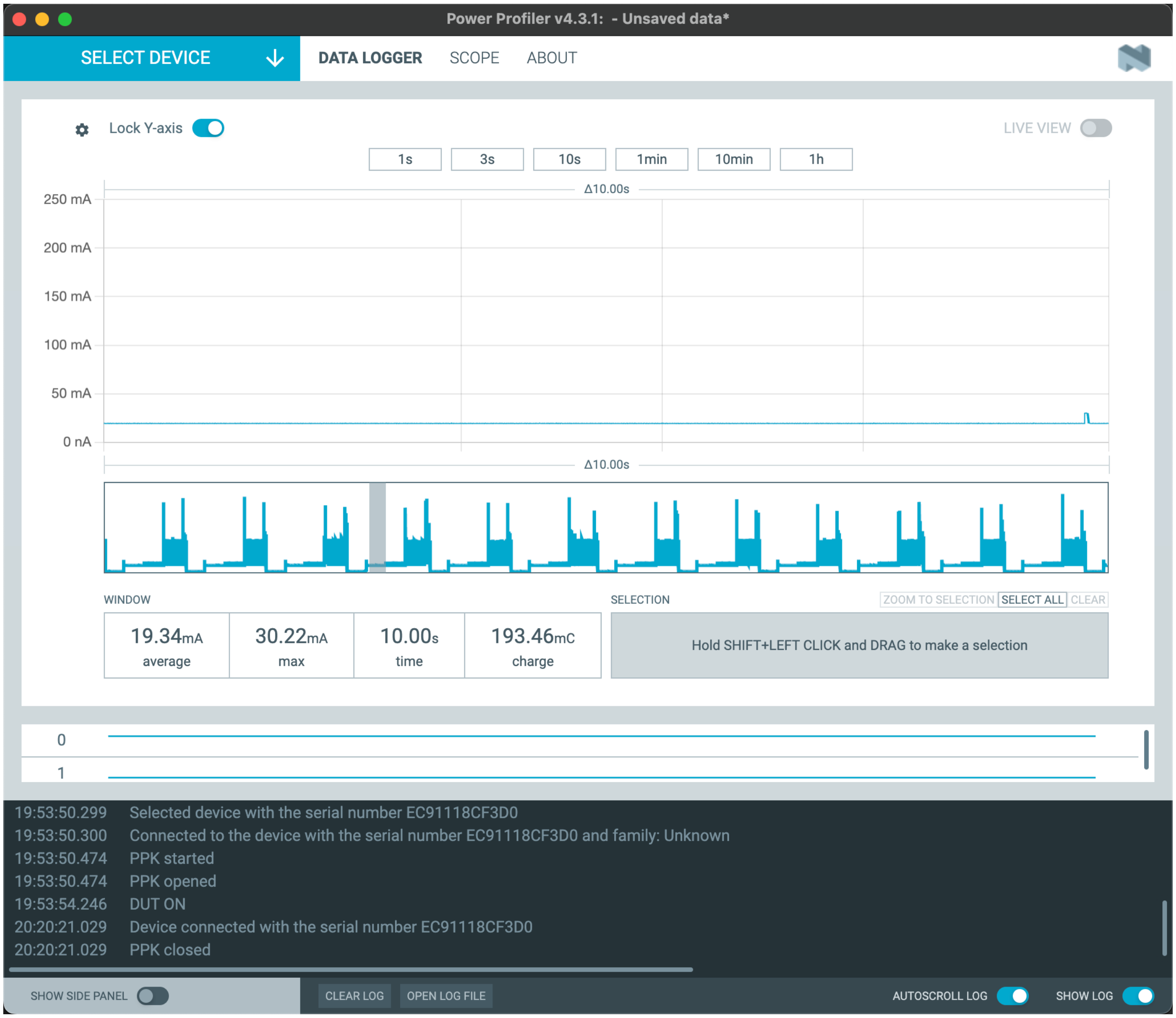
Battery Life: 229.36 hours (9.56 days)



Stage 1: Idle ESP32

Average Current: 19.34mA

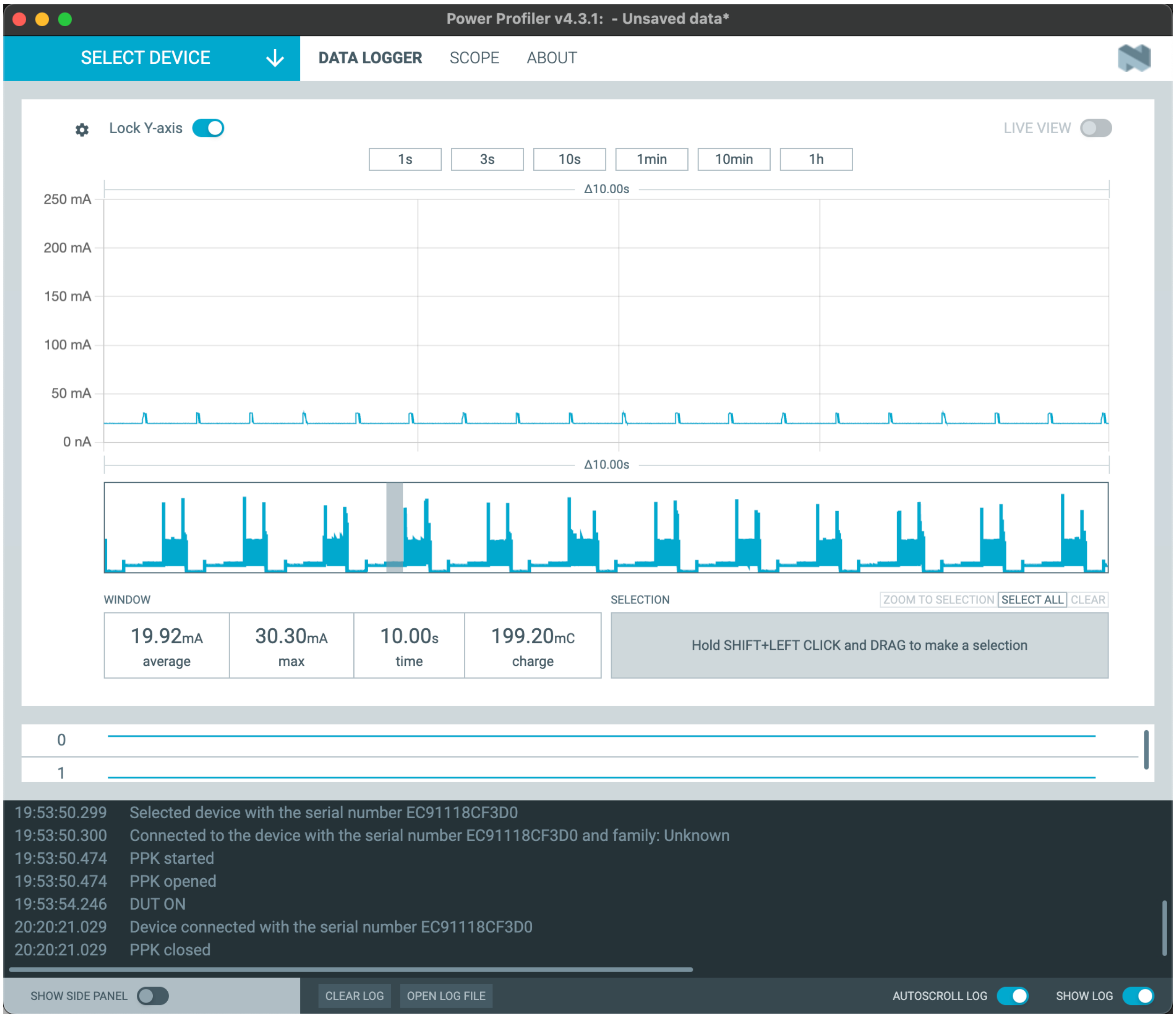
Battery Life: 25.85 hours (1.08 days)



Stage 2: Only ultrasonic sensor working

Average Current: 19.92mA

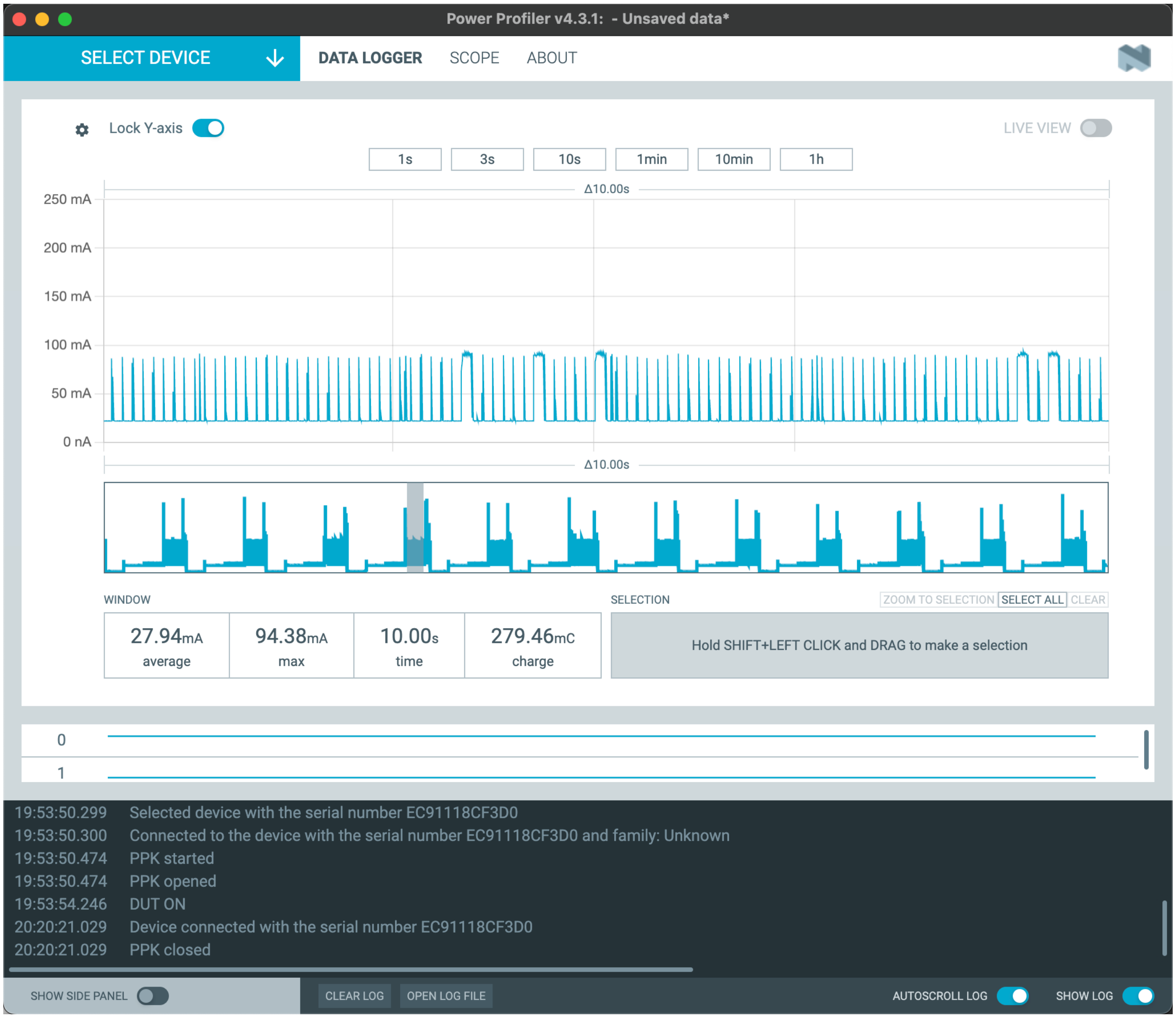
Battery Life: 25.10 hours (1.05 days)



Stage 3: WiFi + no Ultrasonic

Average Current: 27.94mA

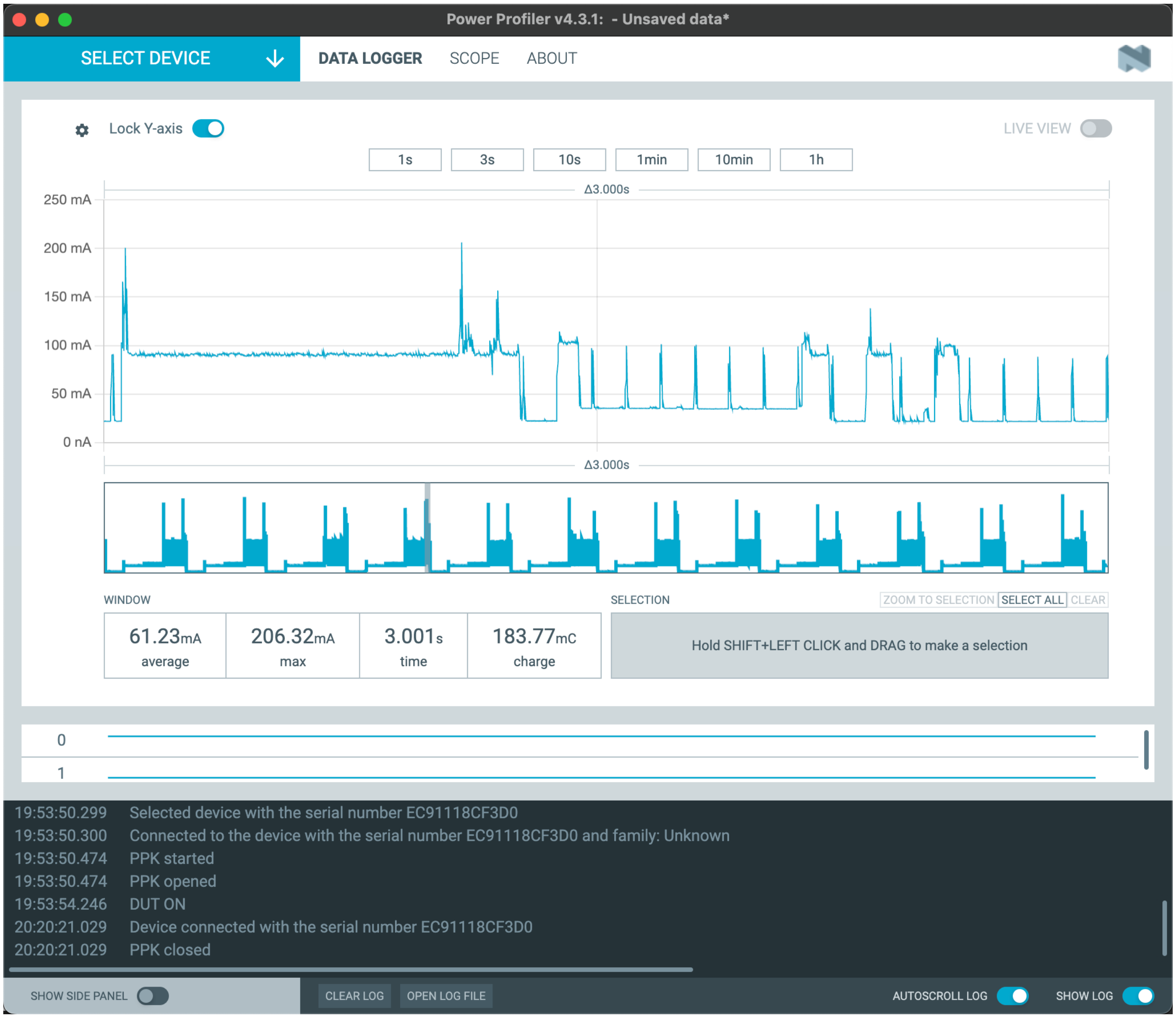
Battery Life: 17.89 hours (0.75 days)



Stage 4: Ultrasonic + Wifi + Sending data to Firebase

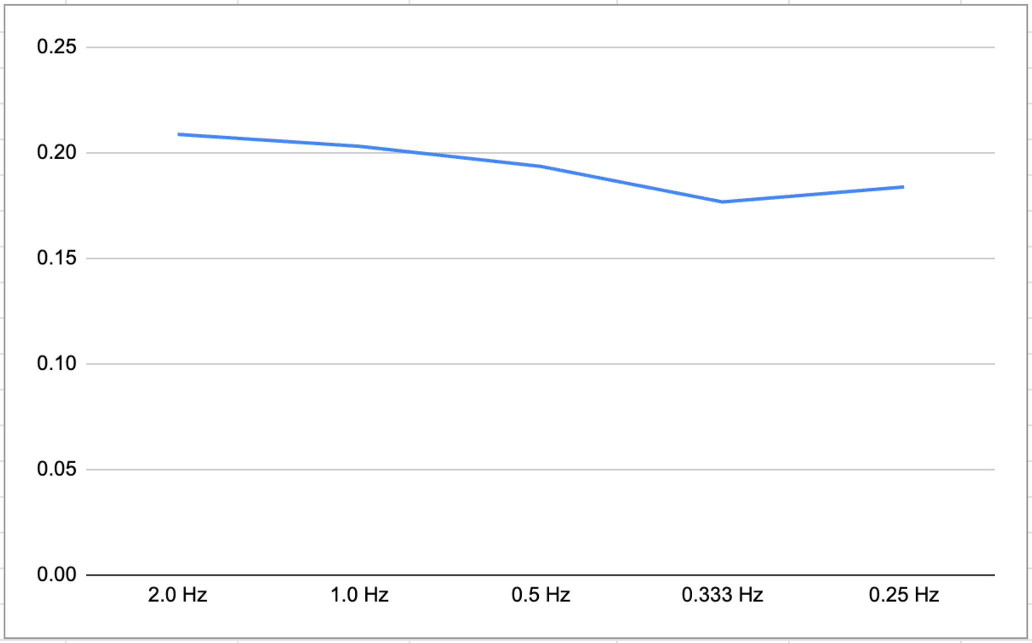
Average Current: 61.23mA

Battery Life: 8.17 hours (0.34 days)



4. Demonstrate the power consumption (W) during the data transmitting stage with different data transmitting frequencies. Plot the figure demonstrating the correlation between data transmitting frequency (Hz) and power (W).

A	B	C	D	E	F	G
	Frequency	Interval	Avg Current (mA)	Voltage (V)	Power (mW)	Power (W)
1	2.0 Hz	500 ms	63.28	3.3	208.824	0.208824
2	1.0 Hz	1000ms	61.57	3.3	203.181	0.203181
3	0.5 Hz	2000ms	58.68	3.3	193.644	0.193644
4	0.333 Hz	3000ms	53.58	3.3	176.814	0.176814
5	0.25 Hz	4000ms	55.73	3.3	183.909	0.183909



X-axis: Frequency

Y-axis: Power (W)

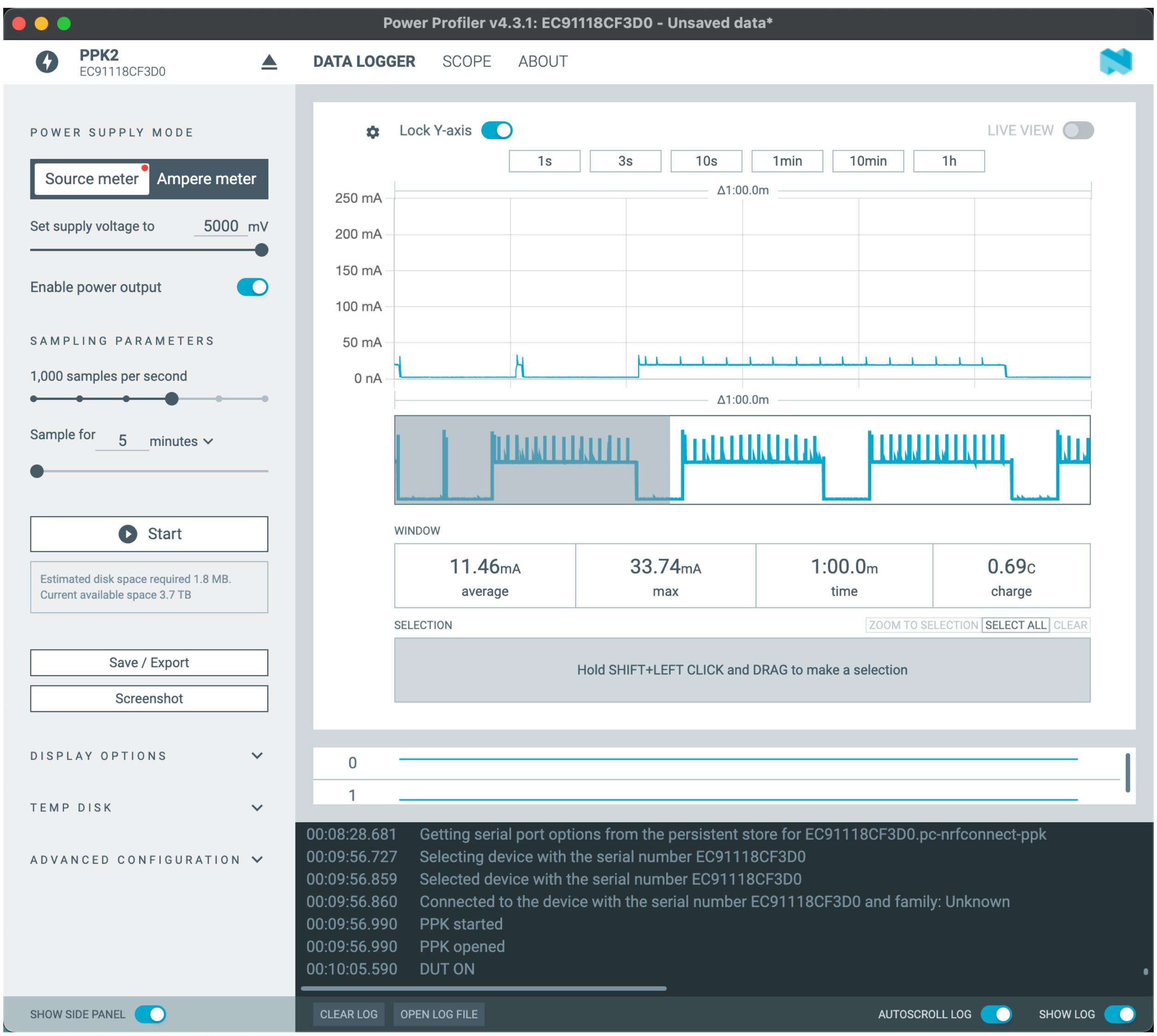
5. Description of your own power-saving strategies/policies.

The device operates primarily in deep sleep mode, with each sleep cycle lasting 10 seconds. After each sleep period, the device wakes up for a brief half-second interval to perform a quick check of the ultrasonic sensor. During this quick check, if no significant motion is detected, meaning the distance measurement has changed by less than 10 centimeters, the device immediately returns to deep sleep mode to conserve power.

However, if motion is detected during the quick check, the device enters an active monitoring state where it continuously monitors the sensor every 2 seconds for a duration of 30 seconds. This extended monitoring period allows the device to confirm that the detected motion is ture and not a false trigger from sensor noise or temporary disturbances.

Only after motion has been confirmed through this 30-second monitoring period does the device proceed to connect to WiFi and upload the event data to Firebase. This upload process takes approximately 3 seconds to complete. Once the data transmission is successful, the device immediately disconnects from WiFi and returns to deep sleep mode to maximize battery life.

6. Annotated screenshots of the power consumption plot of your power-saving strategy's use case. The screenshot should at least include the deep-sleep stage and the working stage of your device.



7. Estimate your strategy's electrical consumption (mAh) in 24 hours under your simulated scenario. Prove that your strategy can make your device work for at least 24 hours with a 500mAh battery.

- Total electrical consumption = 11.46 mA × 24 h = 275.04 mAh < 500mAh

8. The link to your edited code with your power-saving strategy on GitHub. You can either create a new repo or push the updates to your existing repo for all labs (recommended).