**Lab 3 – Battery Characterization**

**Objective**

The first goal of this experiment was to write a MATLAB script to read the instantaneous voltage and current from a power source and record the results every second. The second goal of the experiment was to implement a fan motor that would begin spinning once the wheel motor (prepared in the previous experiment) hit 50% of its maximum RPM and stops again when the wheel falls back below 50%. The final goal of this experiment is to generate, using MATLAB, several graphs, including a graph of Battery Voltage vs. Time, Battery Current vs. Time, Battery Instantaneous Power vs. Time, and Battery Cumulative Energy Delivered vs. Time.

**Equipment List**

|  |
| --- |
| * HP Oscilloscope |
| * HP Digital Multimeter |
| * Fan with fan blades |
| * Arduino Mega 2560 |
| * DC Motor with wheel |
| * H-bridge |
| * (4) Banana jack-to-alligator clip cables |
| * (2) 9V batteries |
| * Optical counter |

**Procedure**

First, the H-bridge is set up and tested using a 9v battery as a power source (instead of the 12v power supply). The H-bridge is then further configured so that the IN1 pin acts as a PWM source. Leads are then connected to the battery from the multimeter, as well as the oscilloscope in order to measure current and voltage respectively. A MATLAB script is then written that will take the data every second until the voltage in the battery drops to 4.8v. The time taken, voltage and current are all recorded and used to produce the graphs.

Beyond measuring the battery as a power source, a fan motor needs to be implemented into the overall design. Code is added to the main Arduino that says if the RPM read in from the wheel motor, reaches 50% of a maximum threshold (set arbitrarily), a digital pin should be written high to send a signal to start the fan motor. In order to minimize the noise coming from the power supply in the circuit a separate bread board should be used to help isolate the fan circuit, capacitors should also be added across the terminals of the fan to filter out low and high frequency noise. These steps help mitigate noise and allow for better readings of the RPM from the optical sensor.

**Deliverables**

1. MATLAB and Arduino code
2. A fan motor that starts when the wheel spins at 50% of a pre-determined max RPM and stops when the wheel is operating lower than 50% of that max RPM
3. Plot of battery voltage vs. Time (demonstrating the battery being discharged)
4. Plot of battery current vs. Time (demonstrating the battery being discharged)
5. Plot of battery instantaneous power vs. Time (demonstrating the battery being discharged)
6. Plot of battery cumulative energy delivered vs. Time (demonstrating the battery being discharged)

**Results**

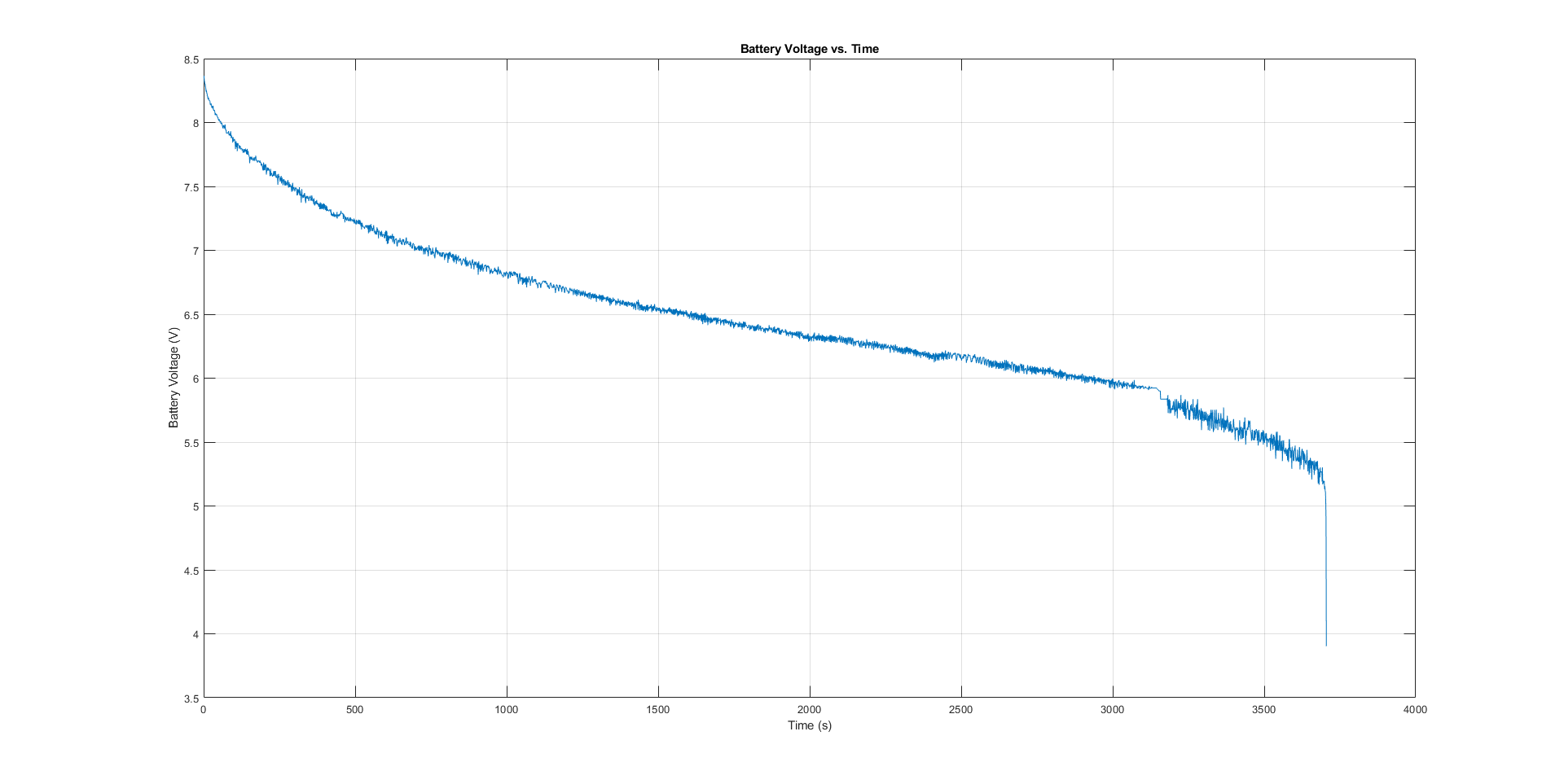


Figure 1: Voltage vs. Time Graph

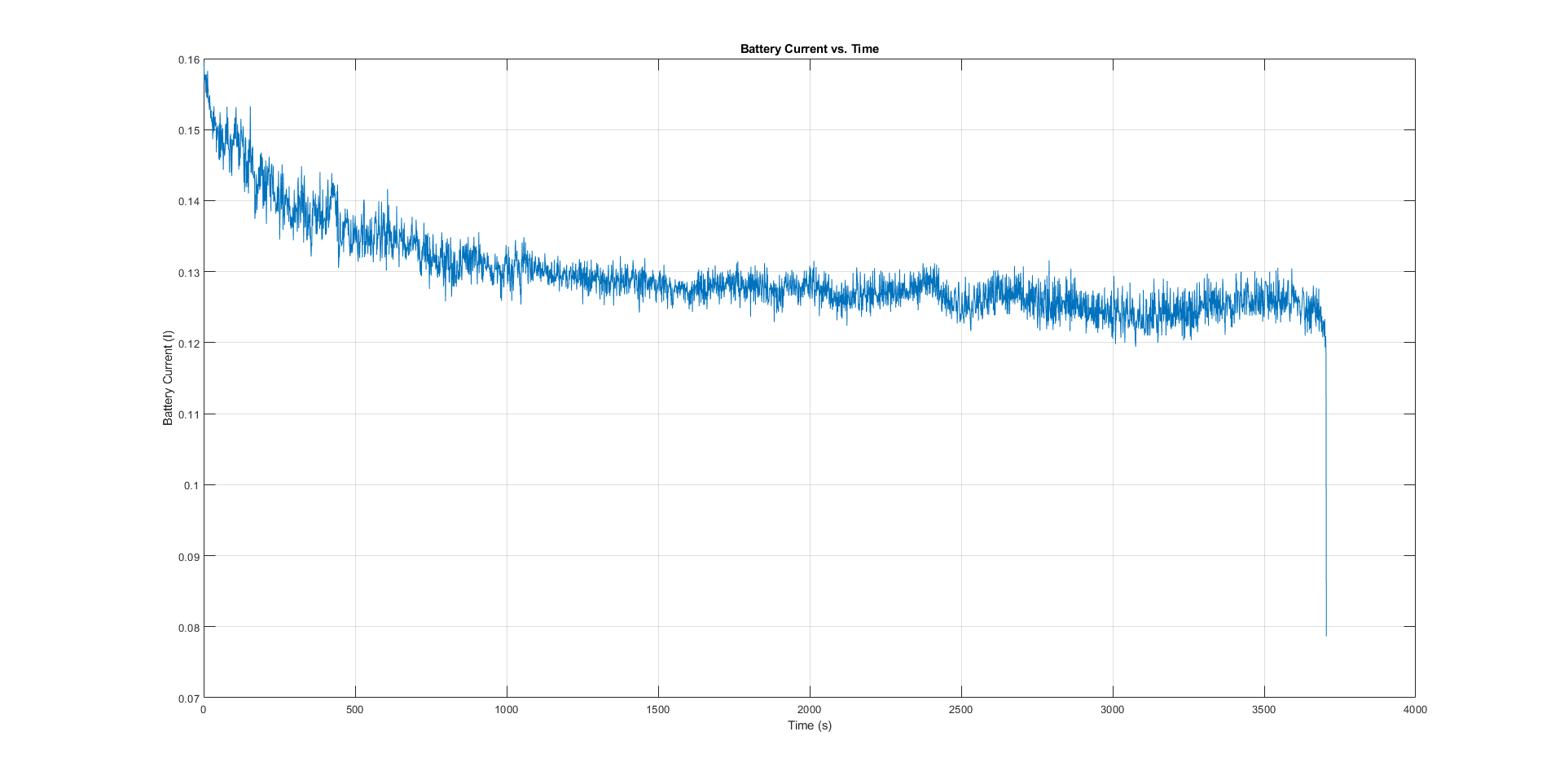


Figure 2: Current vs. Time Graph

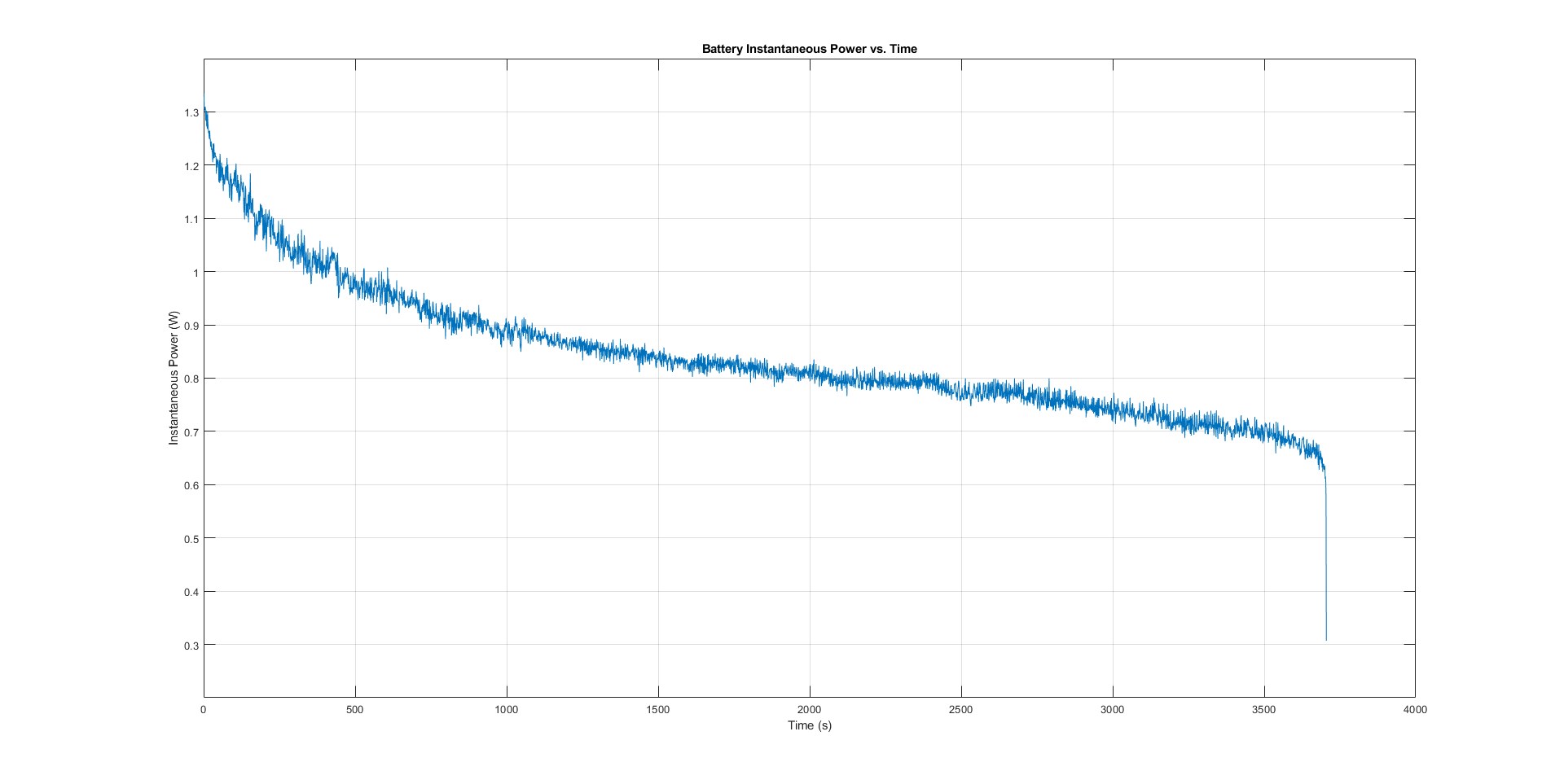


Figure 3: Instantaneous Power vs. Time Graph

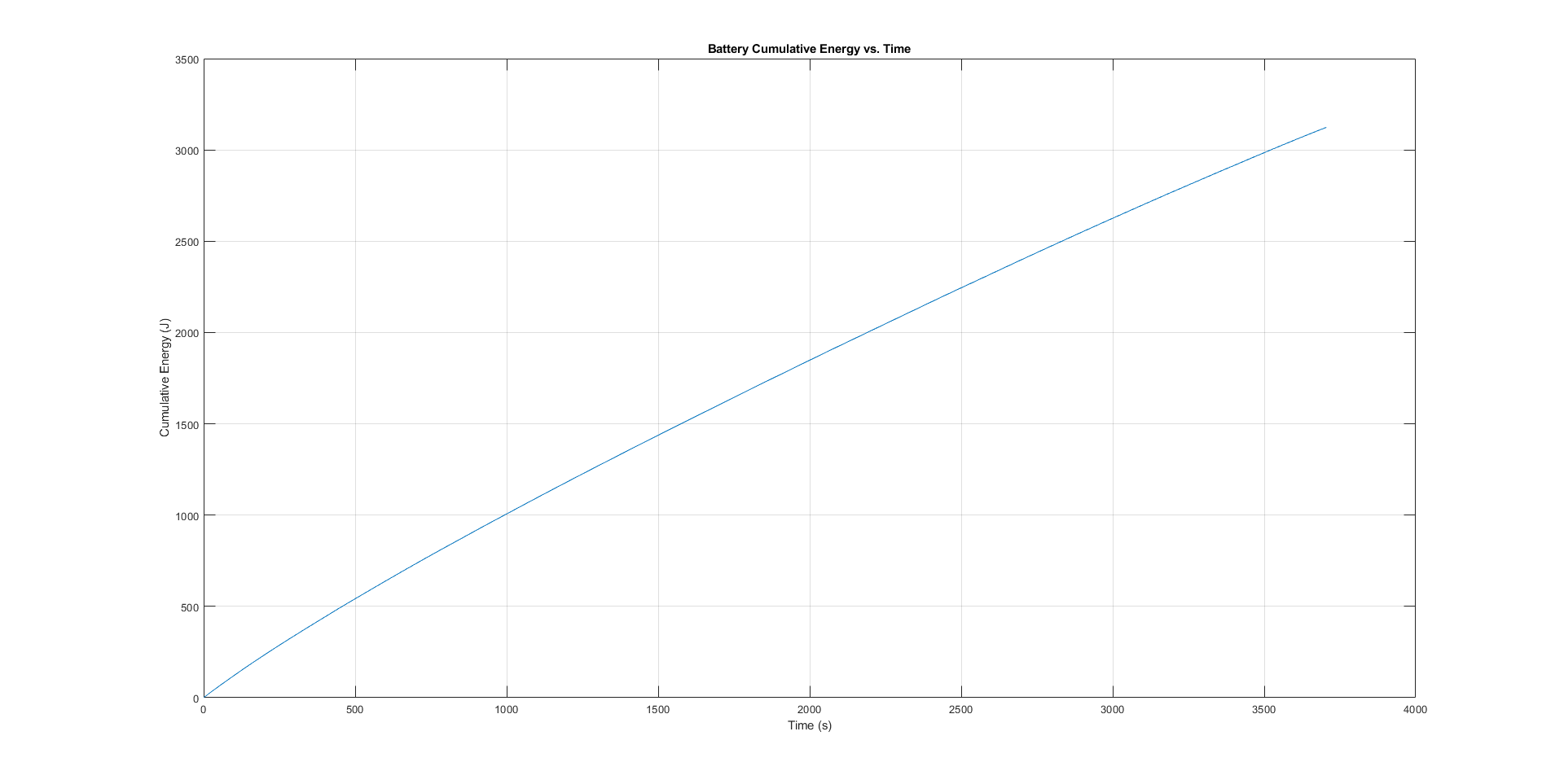


Figure 4: Cumulative Energy vs. Time Graph