A novel sensor based portable energy optimization method for battery powered electro-mechanical devices

Sanaya Telang

Word Count: 2272

Abstract:

Galileo clock mechanisms remain relevant due to their aesthetic appeal and ability to inspire interest in STEM. However, an issue with the classic design is that the escapement is powered by the slow descent of a weight, which must be frequently reset as it reaches the end of its range of motion. This design leads to a loss of accuracy and requires continuous user intervention. Alternatively, electro-mechanical Galileo-based designs consume extensive power due to continuous motor operation.

I propose a novel energy-efficient clock escapement design that uses optimization techniques via software control and hardware assisted power gating. I created a prototype using a commercial off-the-shelf microcontroller based embedded system, 3D printed motor housing, and infrared sensors to disable the battery when the pendulum is not in contact with the gear system, accepting modest instantaneous accuracy tradeoffs in exchange for energy savings.

Experimental measurements demonstrate that, despite voltage not dropping to zero, no current is drawn during inactive intervals at an average run frequency of 400 ms. Stratified sampling confirms this behavior across multiple oscillation cycles. Thus, the system achieved 44.2% energy savings which scales linearly over time and enables sustained reductions in power consumption.

The compact and portable design is broadly useful to other mechanically driven systems with cycle-dependent power requirements. Potential applications include electric lawn mowers and air compressors, where this optimization approach could increase lawn coverage from 4000 sq ft to 6000 sq ft and extend continuous compressor runtime from 30 minutes to approximately 42 minutes.

Key Words:
Energy, Optimization, Escapement, Sensor, Electro-mechanical

Introduction:

Mechanical clock escapements have existed since the 13th century C.E. when Henry de Vick discovered how an oscillating mechanism could efficiently track time [1]. Over the centuries, a wide variety of escapement designs have been invented and refined. In general, mechanical clock escapements are a way of telling time using a mechanism that oscillates at equal time intervals [2]. However, the issue with many of these clocks is that they are weight driven and therefore have a finite run time.

In this research, I designed an electromechanical Galileo clock escapement that operates accurately and reliably while improving energy efficiency. Traditional electromechanical Galileo escapement consumes continuous power. However, for most of the pendulum's oscillation cycle, the motor is not required to be active as the pendulum is not making contact with the gear train system. In this study, I opportunistically turn off the battery, thereby reducing power consumption.

Power has two components, dynamic power and leakage power, and is a function of current (I) and voltage (V), as shown in Equation 1.

$$P = V * I \quad \textbf{(1)}$$

Voltage is defined by Ohm's Law as the product of resistance (R) and current drawn (I), as shown in Equation 2.

$$V = I * R \tag{2}$$

Substituting Equation 2 into Equation 1 shows that power is proportional to the square of current drawn as shown in Equation 3.

$$P = I^2 * \mathrm{R} \tag{3}$$

In this design, leakage power is held constant while dynamic power is reduced [3]. Power savings are achieved by turning the motor off during portions of the pendulum cycle, thereby dropping the current to zero [4]. Reducing power consumptions directly conserves energy since energy is the integral of power, as shown in Equation 4.

$$E = \int P\, dt \tag{4}$$

In any battery powered device, energy is directly related to the number of hours of battery life (HOBL) and therefore affects user experience. The proposed energy optimization method extends battery life by allowing the system to remain in operation for longer periods without drawing power. This approach also reduces charging frequency, minimizing user interruptions and improving both operational efficiency and user experience.

Methods

The investigation consisted of two key steps: (1) the design and fabrication of the clock mechanism and (2) experimental testing the clock mechanism for accuracy, efficiency, and reliability.

I began by creating a basic prototype of the Galileo clock escapement and analyzing the design's limitations, particularly how the pendulum's period could be leveraged for energy savings [3]. During initial testing, I observed variability in the pendulum's period resulting from the difference in the inertia during contact of pendulum to the gear as well as the coefficient of restitution [5]. To account for this variability, I chose the 'average time period' as the primary metric for analysis.

Next, I measured the mass of the weight and radius of the spool to calculate the amount of torque required by a motor. The product of these values yields the required torque, which was used to properly size a motor and gear train to power the clock mechanism. Because the motor and gear train needed to be housed within the clock, I designed a custom housing whose dimensions were derived from the existing clock structure. This housing was designed in SolidWorks and fabricated using a 3D printer. The motor housing design is shown in Figure 1.

As shown in Figure 2, the pendulum only interacts with the clock mechanism during the portion of its swing when in contact with the gear train. Supplying torque when the pendulum is not interacting with the mechanism results in unnecessary energy loss. To mitigate this inefficiency, I installed an infrared (IR) sensing system on the clock. Arduino code was programmed such that when the sensor detects the pendulum, the motor stays off. In contrast, when the pendulum is not detected and is therefore interacting with the gear train, the motor turns on. This logic is implemented in lines 66-70 of the code shown in Figure 4. This approach allows the system to supply the pendulum with its optimal energy to keep the clock running with a minimal amount of electricity wasted. All electrical components are controlled using a custom-programmed Arduino Uno equipped with a motor shield.

Three performance attributes of the clock evaluated in this investigation. The first was accuracy, defined as the consistency of the pendulum's period over time. To evaluate the energy efficiency of the clock, I measured the electrical energy consumed by the clock with and without the sensing system over 30 cycles. Initial attempts to measure current with a multimeter were unsuccessful due to insufficient sampling resolution, which motivated the transition to more precise and high frequency data sampling. Consequently, an INA219 sensor was integrated into the circuit, as shown in figure 5, to measure current, voltage, and power. This integration required precise re-soldering, presenting a significant technical challenge. A supplemental video demonstrating the circuit setup is provided.

The Arduino code was modified such that the INA219 would attempt to get readings as fast as possible, roughly every 5 milliseconds as seen in line 13 of the arduino code in Figure 3. In practice, the INA219 took readings every 7.5 milliseconds on average, yielding between 40-60 samples per cycle. This sampling rate was sufficient to support descriptive statistical analysis. By comparing the measured energy consumption with and without the sensing system, I calculated expected improvement in energy efficiency.

Lastly, the reliability of the system was evaluated qualitatively by comparing its operational duration to that of a traditional weight-driven Galileo escapement. Because the proposed system is electrically powered and energy optimized, it is capable of operating for significantly longer periods without interruption, improving long-term reliability.

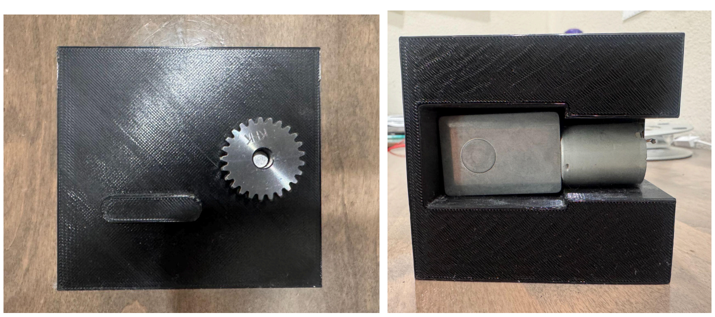

Figure 1. Motor housing design with motor and gear

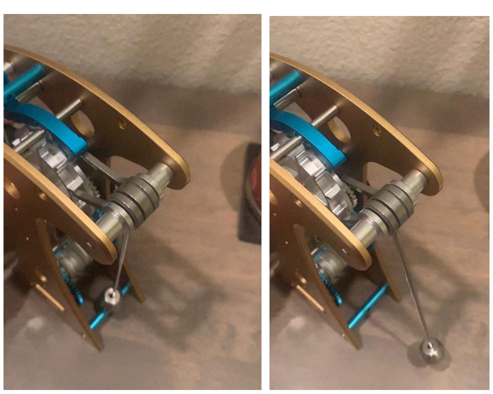

Figure 2. Positioning of the pendulum when touching the gear system vs. when not in contact with the gear train

```
#include <Wire.h>
#include <Adafruit_INA219.h>

Adafruit_INA219 ina219;

// -------------------- CONSTANTS --------------------
const int PIN_TO_SENSOR = 2;
const int MOTOR_DIR_PIN = 12;
const int MOTOR_BRAKE_PIN = 9;
const int MOTOR_PWM_PIN = 3;
const int ANALOG_PIN = A3;

const uint32_t SAMPLE_PERIOD_US = 5000; // 0.005 s = 200 Hz

// -------------------- TIMING --------------------
uint32_t nextSampleUs;

// -------------------- SETUP --------------------
void setup() {
  pinMode(PIN_TO_SENSOR, INPUT);
  pinMode(MOTOR_DIR_PIN, OUTPUT);
  pinMode(MOTOR_BRAKE_PIN, OUTPUT);
  pinMode(MOTOR_PWM_PIN, OUTPUT);

  Serial.begin(500000);
  Wire.begin();

  if (!ina219.begin()) {
    Serial.println("ERROR: INA219 not found");
    while (1);
  }

  // Optional: better resolution if you are below ~400 mA
  // ina219.setCalibration_16V_400mA();

  // CSV header
  Serial.println("t_s,sensor_digital,analog_val,bus_V,shunt_mV,current_mA,power_mW");

  nextSampleUs = micros();
}

```

Figure 3. Arduino Code 1 (programs the INA219 and outputs of voltage, current, power, and sensor reading)

```
// -------------------- LOOP --------------------
void loop() {
  uint32_t nowUs = micros();

  // precise 100 Hz scheduler
  if ((int32_t)(nowUs - nextSampleUs) >= 0) {
    nextSampleUs += SAMPLE_PERIOD_US;

    // ---------- TIMESTAMP ----------
    float time_s = nowUs / 1e6;  // seconds since boot

    // ---------- SENSOR READS ----------
    int sensorState = digitalRead(PIN_TO_SENSOR);
    int analogVal = analogRead(ANALOG_PIN);

    float busV     = ina219.getBusVoltage_V();
    float shuntmV  = ina219.getShuntVoltage_mV();
    float currentmA= ina219.getCurrent_mA();
    float power_mW = ina219.getPower_mW();

    // ---------- MOTOR LOGIC ----------
    digitalWrite(MOTOR_DIR_PIN, HIGH);
    digitalWrite(MOTOR_BRAKE_PIN, LOW);

    if (sensorState == HIGH) {
      analogWrite(MOTOR_PWM_PIN, 0);
    } else {
      analogWrite(MOTOR_PWM_PIN, 255);
    }

    // ---------- CSV OUTPUT ----------
    Serial.print(time_s, 4); Serial.print(",");
    Serial.print(sensorState); Serial.print(",");
    Serial.print(analogVal); Serial.print(",");
    Serial.print(busV, 6); Serial.print(",");
    Serial.print(shuntmV, 6); Serial.print(",");
    Serial.print(currentmA, 6); Serial.print(",");
    Serial.println(power_mW, 6);
  }
}
```

Figure 4. Arduino Code 2 (programs the motor such that it will turn off when the sensor detects the pendulum)

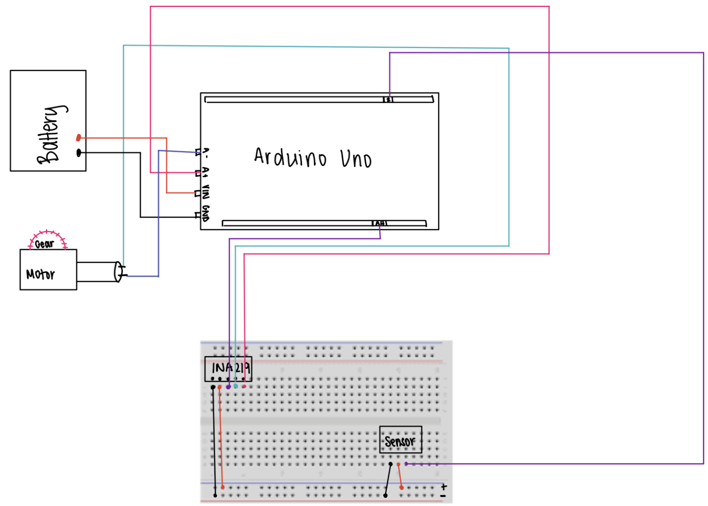

Figure 5. Labeled circuit diagram

Results

The experimental results demonstrate that selectively actuating the power supply using the sensor system significantly reduces energy consumption. A similar technique was used in a different study optimizing algorithms to reduce power consumption in VLSI circuits; my project uses a near identical algorithm of turning off voltage supply when it is not needed [6]. The tradeoff for this energy savings was a modest increase in the variance of the oscillation period [6].

Across all sensor-controlled operations, the battery supply was disabled for approximately 42.2% of each oscillation period. The rise and fall times of the measured current were observed to be effectively zero within the resolution of the INA219 measurement capabilities, indicating that the switching behavior of the system was nearly instantaneous. This suggests that the observed power reduction was due to a discrete on/off actuation of the motor.

As seen in Figures 6, 7, and 8, a modest increase in the variance of the oscillation period was observed in the sensor-controlled tests compared to continuous motor operation. The increased variability from 0.02 sec for continuous operation to 0.06 sec for the sensor-controlled operation is likely due to contact dynamics between the pendulum and the gear system in the mechanical design [7]. When power is applied only during specific phases of the oscillation, small differences in the timing of the engagement produced measurable fluctuations in the cycle period. Importantly, this increased variance was evenly spread across the experiment with no trend of the oscillations becoming shorter or longer over time. This indicates that the mean oscillation period remained stable over the duration of operation.

Notably, the mean cycle period increased by 40 ms during the sensor-controlled operation. For an ideal small-angle pendulum, the period is independent of amplitude and only depends on the mass of the bob and pendulum length. However, this system is not ideal for several reasons including the mechanical coupling of the motor and gears and the intermittent forcing of the pendulum. By turning off the motor for portions of the cycles, the pendulum spent more time near its amplitude, effectively increasing the period. The period increase of 14.3% (28 ms continuous operation vs. 32 ms sensor-controlled operation) indicates that the system is still receiving sufficient energy to sustain oscillations, with the sensor-controlled actuation still preserving the functional motion. So any energy savings are achieved without ultimately destabilizing the system. This is important for many real-world applications since many use cases do not require precise timing as long as the system is stable.

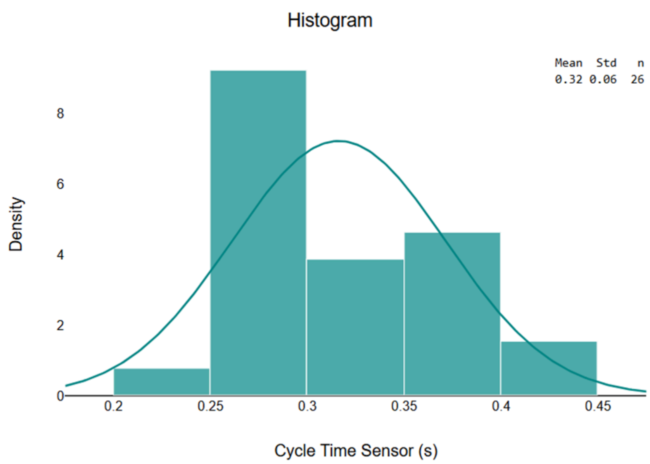

Figure 6. Measurements of cycle times with power optimization system

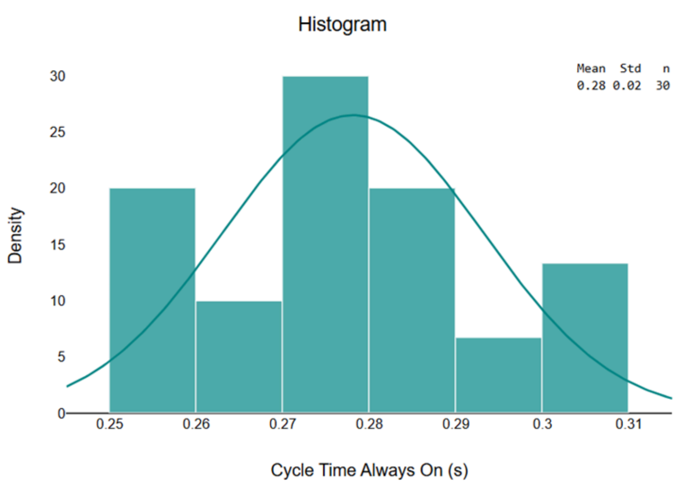

Figure 7. Measurements of cycle times without power optimization system

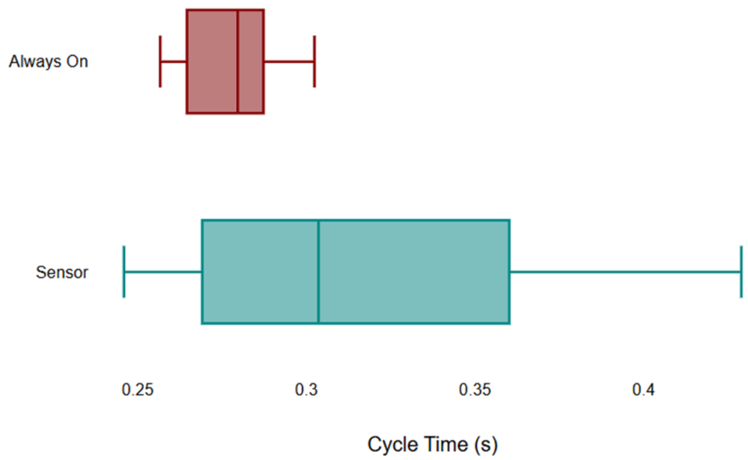

Figure 8. Comparison of cycle time distribution with power optimization vs. without

Figures 9 and 10 show representative time-series measurements across the motor from the INA219 circuit of voltage, current, and power during sensor-enabled trials. When the sensor detected the pendulum, the current measurement dropped to 0 mA, confirming the complete stop of electrical current flow to the motor. The measured power closely followed the current, consistent with the relationship described in Equation 1. When current dropped to zero, the power also dropped to zero. Small transient spikes observed during switching are measurement noise and rapid switching dynamics as the motor has to overcome the system's inertia and begin rotation [5].

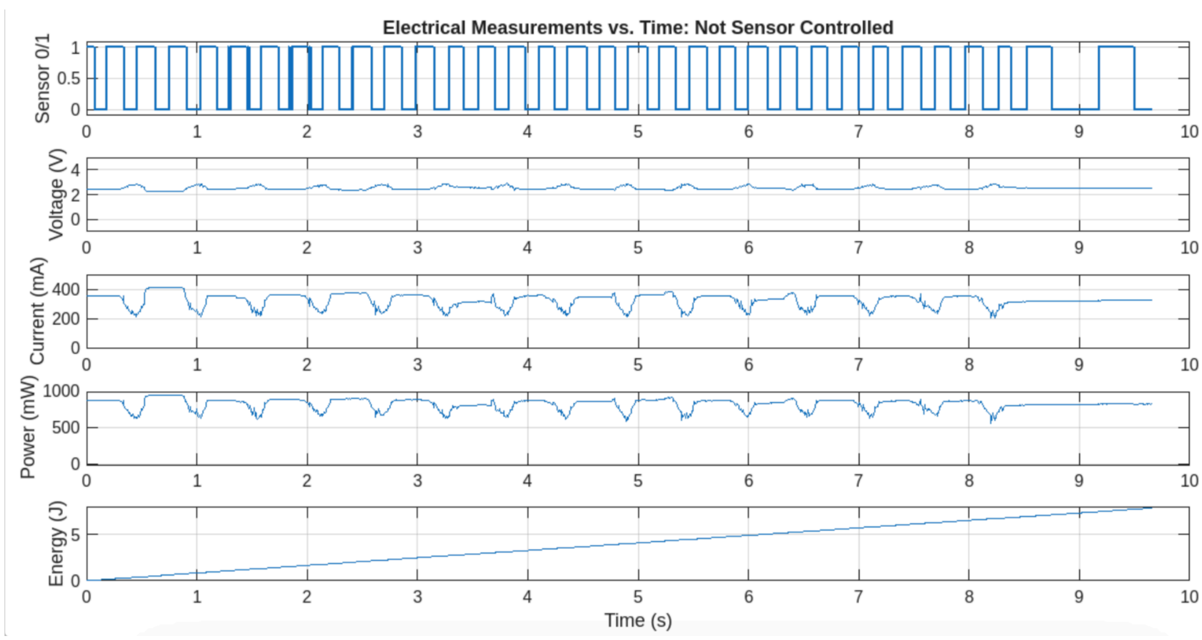

Figure 9. Graphs of electrical measurements taken by INA219 circuit without power optimization method

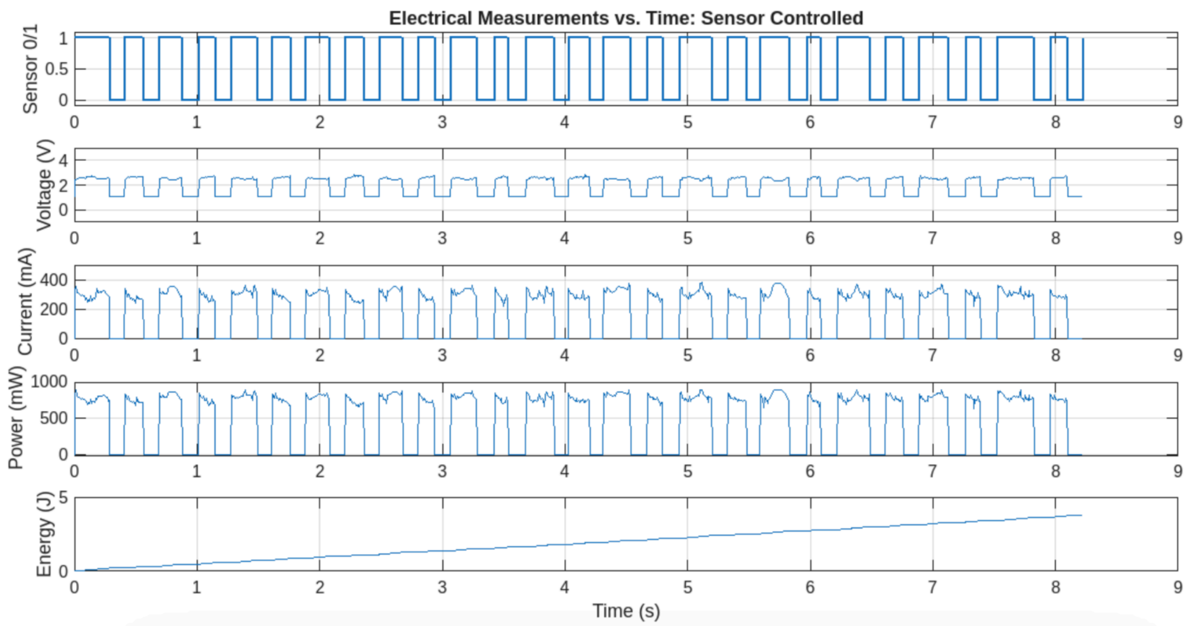

Figure 10. Graphs of electrical measurements taken by INA219 circuit with power optimization method

Figure 11 compares the cumulative energy consumption with and without the sensor during an extended interval. The energy difference between the two configurations show a linear trend since the sensor triggers the battery to be disabled for a constant fraction of each oscillation. Average energy savings of 44.2% were achieved over the duration of the experiments.

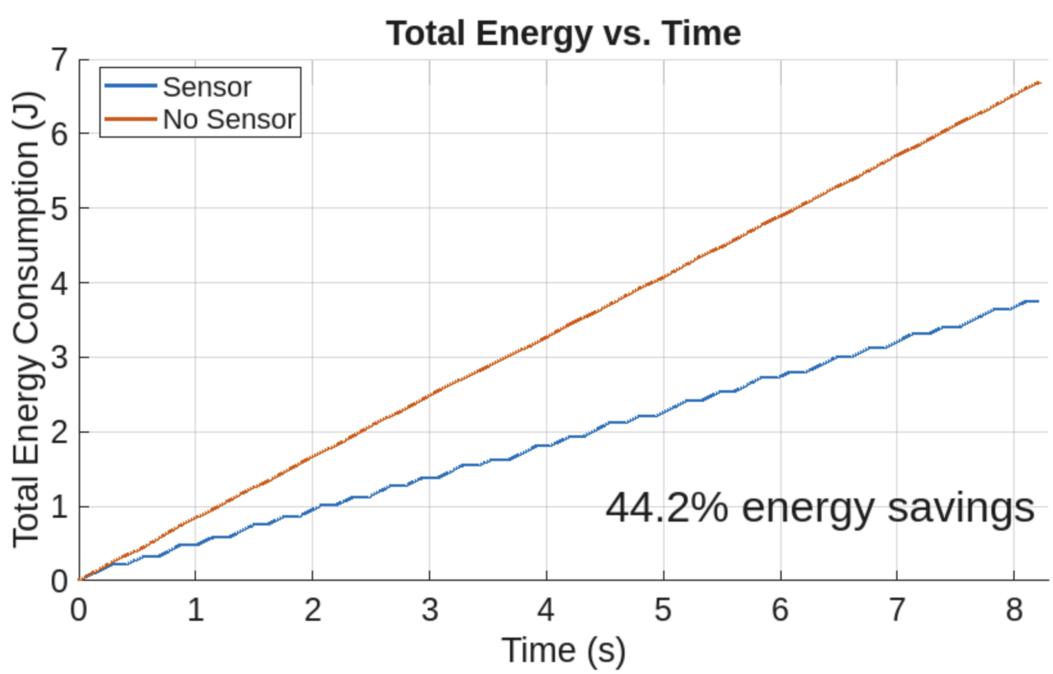

Figure 11. Comparison of total energy consumed during runtime with power optimization

method vs. without

As shown in Equation 5, I used a linear regression model to relate the energy consumed per cycle to the duty cycle of the motor. My results demonstrate that energy per cycle remains statistically constant, and consistent with the motor being inactive for 42.2% of each cycle. Per the residual analysis, remaining variation is minimal and can be attributed to noise rather than system behavior.

$$E_{cycle} = \boldsymbol{\beta}_0 + \boldsymbol{\beta}_1(on - time\ fraction) + \varepsilon \qquad \textbf{(5)}$$

Discussion

Compared to other similar research, such as an Atmos clock, the model presented in this study differs fundamentally in its method of actuation. While an Atmos clock is powered by changes in atmospheric pressure and operates purely mechanically, the proposed design employs an actively controlled electro-mechanical escapement [3]. Additionally, both the POWSER-based system and my approach accept modest reductions in accuracy in exchange for improved energy efficiency [8]. While POWSER reported 34% power savings, my work obtained 44.2% energy savings, demonstrating its effectiveness.

Future work will focus on minimizing design-induced variability by tuning the pendulum length and mass so that the oscillation period remains consistent regardless of the phase at which the pendulum engages the gear train. Further energy optimization will be explored by determining how many cycles the pendulum can run without needing a boost of energy from the motor, allowing for longer periods of time where power supply is not needed and saving significantly more energy. I also plan to utilize a MOSFET component in my circuit to ensure all power comes from the battery, none from the computer.

Beyond the lock mechanism, this power optimization strategy will be simulated and tested in real world applications to compare measured energy savings against theoretical predictions based on Equation 1. I will also measure the increase in uninterrupted runtime. One such application is an electric lawn mower that selectively disables power when sensing a dead patch of grass or areas already cut.

For example, a RYOBI mower rated at 40V and 40A consumes approximately 1600 W during continuous operation. Applying selective actuation can reduce power to 892.8W, extending runtime by a factor of 1.6. Given a maximum advertised runtime of 80 minutes and an average cutting rate of approximately 3000 sq ft per hour, this optimization could increase runtime to roughly 128 minutes and enable coverage of approximately 6400 sq ft.

Another promising application is a reciprocating air compressor. In such systems, the compression stroke demands the most torque but the intake stroke requires little [9]. Motor assistance could therefore be applied selectively during the compression stroke, reducing peak motor sizing, reducing weight, cost, and wear. For instance, applying this sensor based system to a VIAIR brand air compressor with a rating of 24 V and 23 amps current, can save up to 243 W based on calculations from equation 1. Given a recommended runtime for a VIAIR air compressor of 30 minutes, this optimization could extend operation to approximately 42 minutes.

Additional applications include piston-driven pumps powered by DC motors. Near top dead center and bottom dead center, applied torque approaches zero, making energy input at these points mostly wasted [10]. Conversely, energy put into the system mid-stroke will provide maximum torque and is highly efficient [11]. Lastly, this power optimization method could be used as a hybrid assist on single-cylinder internal combustion engines. Because combustion engine torque is inherently non-uniform, the crankshaft may stall near top or bottom dead center. A DC motor could provide brief electric assistance during these phases and remain inactive during the power stroke, improving efficiency without continuous electrical load [12].

Conclusion

This project demonstrates that the proposed sensor-based power optimization method, validated on a Galileo clock escapement, can reduce energy consumption by up to 44.2% compared to continuous motor operation. By selectively disabling power during periods when actuation is unnecessary, the system achieves these energy savings.

This optimization technique is applicable to a wide range of systems, particularly those with intermittent or cyclical power requirements where energy input during idle periods are not needed. In addition to reducing energy consumption, this approach enhances user experience by extending uninterrupted operating time. For example, when applied to an electric lawn mower, this method could increase continuous lawn coverage to 6400 sq ft. Similarly, for an air compressor the sustained runtime increases from 30 minutes to 42 minutes.

Overall, these results highlight the potential of selective power actuation as an effective strategy for improving both energy efficiency and operational performance in both simple and complex electromechanical systems.

References

[1] Harper, Douglas (2001). "Verge". *Online Etymology Dictionary*. Retrieved 2008-06-22.

[2] Lepschy, A.M., et al. “Feedback control in ancient water and mechanical clocks.” *IEEE Transactions on Education*, vol. 35, no. 1, 1992, pp. 3–10, https://doi.org/10.1109/13.123411.

[3] Patel, Shreyas, et al. “Modeling and analysis of an atmospheric driven Atmos clock with mechanical escapement control.” *2013 European Control Conference (ECC)*, July 2013, https://doi.org/10.23919/ecc.2013.6669118.

[4] Wang, Chua-Chin, et al. “A 24.9% power reduction active gate driver with power gating and current modulation for power mosfets.” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 72, no. 11, Nov. 2025, pp. 1640–1644, https://doi.org/10.1109/tcsii.2025.3588105.

[5] A. V. Roup, D. S. Bernstein, S. G. Nersesov, W. M. Haddad and V. Chellaboina, "Limit cycle analysis of the verge and foliot clock escapement using impulsive differential equations and Poincare maps," Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148), Arlington, VA, USA, 2001, pp. 3245-3250 vol.4, doi: 10.1109/ACC.2001.946422. keywords: {Limit-cycles;Clocks;Differential equations;Aerospace engineering;Aerodynamics;Feedback control;Space technology;Teeth;Gears;Angular velocity},

[6] Dwibedi, Rajat Kumar, et al. “Hybrid optimization algorithms for dynamic and static power reduction in low-power vlsi circuits.” *2024 International Conference on Advances in Computing, Communication and Materials (ICACCM)*, 22 Nov. 2024, pp. 1–6, https://doi.org/10.1109/icaccm61117.2024.11059187.

[7] “Describing function analysis using MATLAB and simulink.” *IEEE Control Systems*, vol. 21, no. 4, Aug. 2001, pp. 19–26, https://doi.org/10.1109/37.939940.

[8] Gandhe, Maithili P, et al. “POWSER: A novel user-experience based power management metric.” *2017 Eighth International Green and Sustainable Computing Conference (IGSC)*, Oct. 2017, pp. 1–8, https://doi.org/10.1109/igcc.2017.8323606.

[9] Verma, Nishchal K, et al. “Statistical approach for finding sensitive positions for condition based monitoring of reciprocating air compressors.” *2011 IEEE International Conference on System Engineering and Technology*, June 2011, pp. 10–14, https://doi.org/10.1109/icsengt.2011.5993411.

[10] Roup, A.V., and D.S. Bernstein. “On the dynamics of the escapement mechanism of a mechanical clock.” *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No.99CH36304)*, vol. 3, pp. 2599–2604, https://doi.org/10.1109/cdc.1999.831320.

[11] Guo, Feng, et al. “Study on piston motion control strategy of single piston hydraulic free-piston engine.” *2012 Second International Conference on Intelligent System Design and Engineering Application*, Jan. 2012, pp. 1140–1143, https://doi.org/10.1109/isdea.2012.555.

[12] Wang, Hao, et al. “Application of battery-capacitor hybrid power supply system in improving the motion performance of medium and Small Power Electric Vehicles.” *2022 7th International Conference on Intelligent Computing and Signal Processing (ICSP)*, 15 Apr. 2022, pp. 1079–1082, https://doi.org/10.1109/icsp54964.2022.9778748.

Acknowledgements

The author thanks Noah Graff for guidance on the model design and experimental setup. James Geraci for assistance with circuit assembly and data visualization.The author also appreciates the inspiration given by Professor Michael Littman at Princeton University for this study.