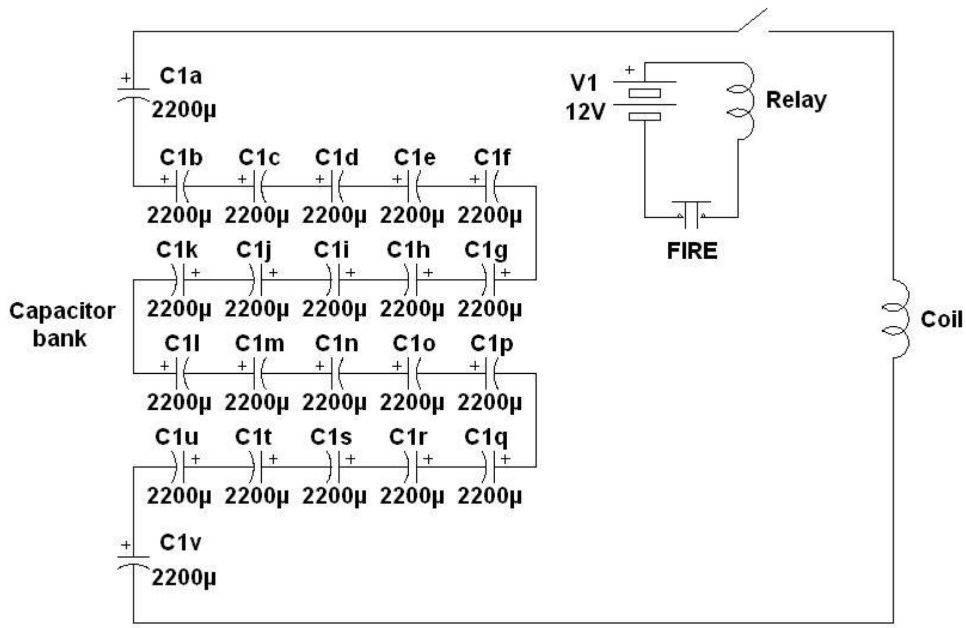
Physics of an axial coil gun

Part II – Circuit analysis

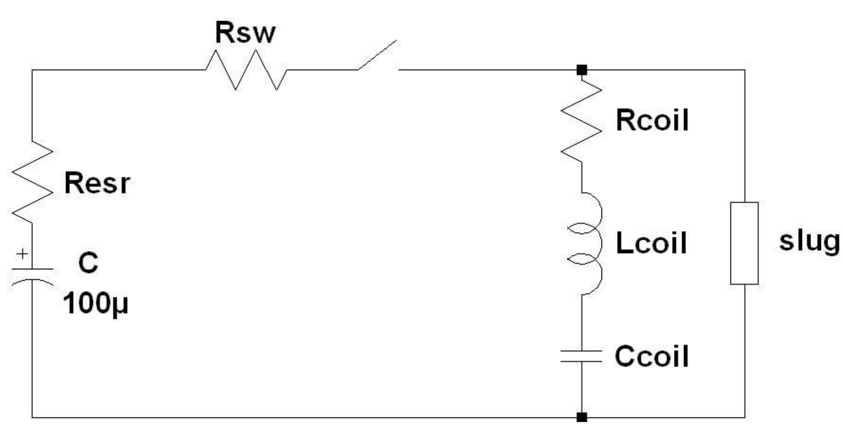
In this part of the paper, we will look at how the slug extracts energy from the magnetic field set up by current flowing through the coil. The following figure shows the layout of the physical components in the circuit.



The coil gun itself is simply the coil shown as an inductor on the right-hand side of the figure. Everything else is the power supply or, more appropriately, the energy stock. The energy stock we will use in this application is described in another paper, titled *A high-voltage Buck-Boost capacitor charger*. The capacitor bank consists of twenty-two 2200µF capacitors wired in series, giving a total capacitance of 100µF. Each capacitor has a voltage rating of 450V, so the capacitor bank should be able to withstand up to 9900V. The charger described in the earlier paper takes five minutes to charge the capacitor bank up to a voltage of 4000V, at which time it holds energy of 800 Joules.

The control relay, too, is included in the charger. When the operator presses the FIRE pushbutton, the relay contacts close and the voltage over the capacitor bank is applied to the external circuit. In our case, the external circuit consists of the coil gun's coil.

From an electrical point-of-view, the situation is a little different. The following schematic diagram shows the circuit we will examine in this part of the paper.



I have placed the slug as a black box in parallel with the coil. It is, after all, the coil which delivers energy to the slug. Whether this is appropriate remains to be seen. If necessary, we will move it into a series position.

The most important characteristic of the capacitor bank, other than its capacitance, is its equivalent series resistance R_{esr}. In the earlier paper, we estimated the R_{esr} of the capacitor bank to be about 1.98Ω. In this paper, we will use a value of 2Ω for the equivalent series resistance of the capacitor bank.

In the earlier paper, we also specified that the FIRE relay would be Digikey's part number 374-1108-ND. This is a relay manufactured by Meder Electronics and used, among other purposes, for RF surgery. Its contacts are rated at 10KV. The datasheet for the component states that its static contact resistance is 150mΩ. In this paper, we will use a value of 0.15Ω for the resistance of the switch (R_{sw}). The datasheet also states that the relay's closing time, including the debounce, is about 3 milliseconds. This could be a significant fraction of the whole duration of a run. In Part I of this paper, we looked at examples where the acceleration of the slug took place over 6 milliseconds. We will have to bear this limitation of the relay in mind. To be precise about it, the voltage which is imposed over the coil will not be a "clean" 4KV at time $t = 0$ when the switch is closed. Instead, it will be erratic, as the relay's contacts open and close for a short interval. There may be sparks across the gap, with accompanying surges of current. (The contacts of the relay used in the earlier paper were also rated up to three Amperes. That is not nearly enough for this application, which will draw tens or perhaps hundreds of Amperes. However, there are lots of other relays. A quick search on the internet showed one which could carry 150 Amperes. It also has an even lower static contact resistance.)

The coil itself also has several characteristics, which I have shown in the schematic diagram as the series combination of a resistance, an inductance and a capacitance. Since we have not yet specified any features for the coil, the best we can do at this point is to make educated guesses. If we are lucky, the guesses will within a factor of ten. But, they will give us a starting point for figuring out what the coil should be like.

So, let us suppose that:

- the coil has an overall length, or height, of about 50 cm, and
- has an inside diameter of 2 cm.

The inside diameter of the coil should be thought of as the diameter of the core on which the wire will be wound. We will want to use very heavy wire. Suppose we use #4 gauge enameled copper wire. Its main characteristics are:

- the wire diameter, excluding the enamel coating, is 5.189 mm,
- its resistance is 0.2485Ω per thousand feet,
- it is rated for 55.7 continuous Amperes and
- permits a maximum current of 83.5 Amperes.

Note that this is heavy wire – we will be able to wind only 4.9 turns per inch along the coil. Let us wind a coil with a round number of 96 turns. The exact length of the coil will be 49.81 cm, which is approximately 50 cm.

The average radius of each turn, taking into account the wire's diameter as well as the core radius, is 1.259 cm. Therefore, the average circumference of each turn ($2\pi r$) is 7.91 cm, or 3.11 inches. 96 such circumferences will use up 298.6 inches, or 24.88 feet, of wire. This is a small fraction of one thousand feet, so the resistance will be a small fraction of 0.2485Ω. In fact, the total resistance of this length of wire is only 0.0062Ω.

We could wind many layers of this wire before the resistance of the wire begins to compare with the 2Ω equivalent series resistance of the capacitor bank. Adding more turns would certainly be useful – we know that the strength of the magnetic field set up increases as the number of turns increases.

However, increasing the number of turns also increases the coil's inductance, which may not be a good thing. For a coil which is long compared with its diameter, and which has an air core, the following formula for the inductance can be used:

$$L_{coil} = \mu_0 \frac{N_{total}^2 A_{coil}}{H_{coil}} \quad (1)$$

where μ_0 is the permeability of free space ($4\pi \times 10^{-7}$ H/m), N_{total} is the total number of turns in the coil, A_{coil} is the area of a cross-section of the coil and H_{coil} is the length of the coil. Using the physical parameters we have already set, the inductance can be computed as follows:

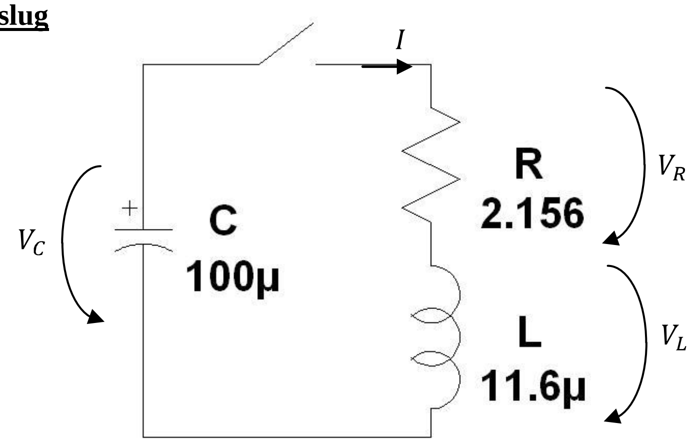
$$\begin{aligned} L_{coil} &= (4\pi \times 10^{-7}) \frac{(96^2) \times (\pi 0.01259^2)}{0.4981} \\ &= 11.6\mu\text{H} \end{aligned}$$

The circular turns of wire which make up the coil are positioned face-to-face. The adjacent turns are not unlike the plates of a capacitor. In this sense, each pair of turns acts like a small capacitor. Indeed, the very wire itself has an innate amount of capacitance. Any piece of the wire is a small volume of copper, and a certain amount of charge needs to flow into the volume before it starts to "express" a voltage. We will not delve into this phenomenon. I expect that the coil and its wire have an aggregate capacitance of a hundred picoFarads, or less. Whatever the exact value of this capacitance, it will be negligible compared with the capacitance of the 100μF capacitor bank.

So, how many layers of turns should the coil have? I do not have a proper answer for that question just yet. What we will do is this. Let us start off by assuming the coil has one layer, with 96 turns, and therefore a resistance of 0.0062Ω and an inductance of 11.6μH. After we see what happens, we may circle back and make a new assumption. As we increase the number of turns from 96 to, say, N, the resistance will increase (approximately) with the ratio $N/96$ and the inductance will increase (approximately) with the ratio $(N/96)^2$.

<u>The response of the circuit, excluding the slug</u>

Since we know nothing about how the slug affects the circuit, let us begin by ignoring it entirely. We will discharge the capacitor bank through the empty coil. Gathering up the various values of the components, the circuit looks as shown to the right.



Component R is the series combination of: (i) the equivalent series resistance of the capacitor bank, (ii) the static resistance of the relay contacts and (iii) the resistance of the wire in the coil. Component L is the inductance of the coil and component C is the capacitance of the capacitor bank. The switch is closed at time $t = 0$. The initial voltage over the capacitor is 4000V.

I have labeled in the figure the four circuit variables of interest. They are:

- V_C is the instantaneous voltage drop over the capacitor,
- V_R is the instantaneous voltage drop over the series resistance in the circuit,
- V_L is the instantaneous voltage drop over the ideal inductance of the coil and
- I is the instantaneous current flowing through the circuit.

All four variables are functions of time t. (There is a convention that capital letters should be used to represent voltages and currents which are constant with respect to time and that small letters should be used to represent instantaneous or time-varying voltages and current. In our case, there are no constant quantities at all, so I have used capital letters throughout. This will avoid confusion between V for voltages and v for the slug's speed.)

To solve for four circuit variables, we need four circuit equations. They are:

<u>The V-I characteristic of the capacitance</u>

$$V_C = V_0 - \frac{1}{C}\int_{\tau=0}^{t} I d\tau \quad (2)$$

I have introduced the symbol V_0 for the initial voltage over the capacitor. The instantaneous voltage over the capacitor is reduced from its initial voltage by the cumulative amount of current which has been extracted from it, divided by the capacitance.

<u>The V-I characteristic of the resistance</u>

The voltage-current characteristic of the resistor is simply Ohm's Law:

$$V_R = IR \quad (3)$$

<u>The V-I characteristic of the inductance</u>

$$V_L = L\frac{dI}{dt} \quad (4)$$

When the current flowing through the inductance increases, a voltage (the so-called "back EMF") is developed over the inductance which is proportional to the rate of change of the current.

<u>The sum of the voltage drops around the circuit</u>

Since this is a series circuit, the sum of the voltage drops around the closed loop must be equal to zero. So, after the switch is closed, we must have:

$$V_C = V_R + V_L \quad (5)$$

The four equations can be combined quite easily, and the result is a second-order differential equation in the single variable I.

$$L\frac{d^2I}{dt^2}+R\frac{dI}{dt}+\frac{I}{C}=0 \quad (6)$$

It is likely that a solution of this differential equation will have the form $\alpha e^{\beta t}$, where α and β are two constants. Since this is a second-order differential equation, there should be two solutions, both of which will have this form. Since this form is assumed to be a solution of Equation (6), it had better satisfy Equation (6). Substituting, and taking the derivatives, gives:

$$\begin{aligned} & L\alpha\beta^2 e^{\beta t}+R\alpha\beta e^{\beta t}+\frac{\alpha e^{\beta t}}{C}=0 \\ \rightarrow \quad & \alpha e^{\beta t}\left(L\beta^2+R\beta+\frac{1}{C}\right)=0 \\ \rightarrow \quad & L\beta^2+R\beta+\frac{1}{C}=0 \qquad (7) \end{aligned}$$

Notice how conveniently the exponential term can be eliminated, leaving behind a quadratic equation in the parameter β, the so-called “characteristic equation”. By the way, we were able to eliminate the exponential term by making the following observation. The exponential term is never exactly equal to zero, so the only way in which the middle expression can always be equal to zero (that is, be equal to zero at any arbitrary time t) is if the coefficient of the exponential term is exactly equal to zero.

Quadratic equations have two roots. (A “root” is a value of β at which the expression computes out to zero.) In this case, the two roots are:

$$\beta=\frac{-R\pm\sqrt{R^2-\frac{4L}{C}}}{2L} \quad (8)$$

We could have one small difficulty. If R, L and C have the right combination of values, the term under the square root will be positive, and there will be two real values for β. But, if the three components have the wrong(?) combination of values, the term under the square root will be negative. If that is the case, then we will have to invoke the imaginary number j, which is the square root of -1, which is to say, $j=\sqrt{-1}$. When β has an imaginary part, the solution will have a form like $e^{j|\beta|t}$. Exponential terms with imaginary exponents are just another way of expressing sinusoidal terms consisting of sine and cosine functions.

Using our component values, $R^2=2.156^2=4.648$ and $4L/C=46.4\mu/100\mu=0.464$. So, the term under the square root is positive, and β has the following two values:

$$\begin{aligned} \beta \quad &= \frac{-2.156\pm\sqrt{4.184}}{23.2\mu} \\ &= -4{,}760 \ \text{ and } \ -181{,}000 \end{aligned}$$

The physical units of β are actually 1/second, so that the product of β and time t is dimensionless. These two exponents correspond to two different solutions of Equation (6). Using a subscript 1 to identify one of the solutions and a subscript 2 to identify the other solution, we can write down the general form of the solution of Equation (6) as:

$$I(t) = \alpha_1 e^{-4760t} + \alpha_2 e^{-181000t} \quad (9)$$

To figure out the two remaining constants α_1 and α_2, we need to apply the initial conditions for the circuit. We know two things: (i) that the initial current is equal to zero and (ii) that the initial voltage over the inductor is equal to V_0. (Note that the third thing we know, that the initial voltage over the capacitor is equal to V_0, has already been incorporated into the first circuit equation, and therefore does not constitute "new" information.)

It is obvious that the current will be zero before the switch is closed. The first initial condition does not mean this. What it means is that the current will also be zero immediately <u>after</u> the switch is closed. Our circuit includes an inductor and the current flowing through an inductor cannot change instantaneously. Since no current flows immediately after the switch is closed, there will be no voltage drop over the resistor, and the inductor must absorb the full voltage drop over the capacitor.

Applying the first initial condition is easy. We simply substitute $t = 0$ into Equation (9), at which time the current must be zero, so that:

$$\begin{aligned} I(0) &= \alpha_1 e^0 + \alpha_2 e^0 \\ &= \alpha_1 + \alpha_2 \\ &= 0 \quad \text{being the 1st I.C.} \\ \rightarrow \quad \alpha_2 &= -\alpha_1 \qquad (10) \end{aligned}$$

Applying the second condition requires that we first get an expression for the voltage over the inductor. We can do this by substituting the expression for the current given by Equation (9) into the circuit equation given in Equation (4). We proceed as follows.

$$\begin{aligned} V_L(t) &= L\frac{dI(t)}{dt} \\ &= L\frac{d}{dt}(\alpha_1 e^{-4760t} + \alpha_2 e^{-181{,}000t}) \\ &= -L(4760\alpha_1 e^{-4760t} + 181{,}000\alpha_2 e^{-181{,}000t}) \quad (11) \end{aligned}$$

Now that we have an expression for $V_L(t)$, we can apply the second initial condition by evaluating it at time $t = 0$. We get:

$$\begin{aligned} V_L(0) &= -L(4760\alpha_1 e^0 + 181{,}000\alpha_2 e^0) \\ &= -L(4760\alpha_1 + 181{,}000\alpha_2) \\ &= -L(4760\alpha_1 - 181{,}000\alpha_1) \quad \text{using the 1st I.C.} \\ &= 176{,}240L\alpha_1 \\ &= V_0 \quad \text{being the 2nd I.C.} \\ \rightarrow \quad \alpha_1 &= \frac{V_0}{176{,}240L} \qquad (12) \end{aligned}$$

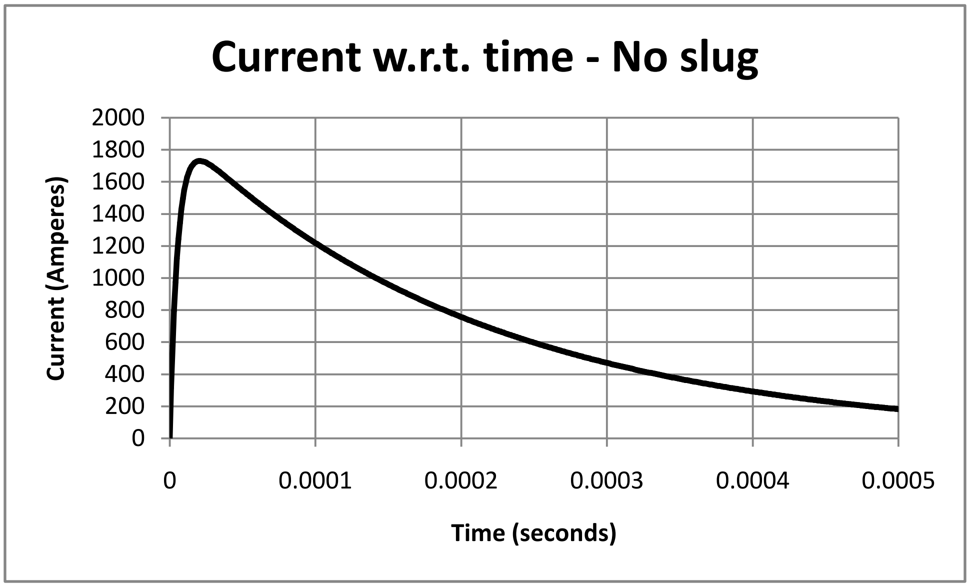
Substituting our physical values, we can compute α_1 as follows.

$$\alpha_1 = \frac{4000}{176{,}240 \times 11.6\mu}$$
$$= 1960\ A$$
$$\text{and} \quad \alpha_2 = -1960\ A \qquad (13)$$

The constants α_1 and α_2 have units of amperes. Now that we have found α_1 and α_2, we can write down the complete expression for the current, as:

$$I(t) = 1960(e^{-4760t} - e^{-181{,}000t})A \qquad (14)$$

The following plot shows the waveform of this current.



This is a relatively fast discharge. The horizontal scale represents a total time of one-half millisecond. The current peaks about 20 microseconds after the start of the discharge. And, it peaks at about 1960 Amperes, which is a huge current. There are several points to note.

1. Do not forget that it will take up to 3 milliseconds for the contacts of the relay to close. As the contacts are closing, the current which is able to force its way through will be less than shown here. Furthermore, current of this magnitude will vaporize the relay contacts as well as the wire itself.

2. The "fast" term in the solution is the exp $(-4760t)$ term. It is the term which makes the algebraically-positive contribution to the current, so it drives the current up to its peak. After the peak, the "slow" term exp $(-181{,}000t)$ takes over and eases the current back down to zero.

3. Increasing the resistance of the coil by adding turns will slow the timing down and, at the same time, reduce the peak current experienced.

4. Adding turns to the coil will also increase its inductance. At some point, the inductance will become large enough to introduce a sinusoidal waviness to the discharge.

Before we start adding turns to the coil, merely for the purpose of "slowing the discharge down", let us do something else.

Modeling the slug as an electrical component

The slug is not unlike an ideal transformer, in that it draws its energy from the magnetic field set up by the coil. In an ideal transformer, the secondary circuit draws its energy from the primary circuit. In an ideal transformer, though, the exchange of energy is indirect. The secondary circuit uses a second coil to draw the energy from the magnetic field set up by the primary coil. In order for a voltage to be developed over the secondary coil, the magnetic field must be varying. If the magnetic field set up by the primary coil does not vary, the secondary coil cannot extract any energy from the magnetic field. For this reason, an ideal transformer will not function unless the current, and magnetic field, vary or are alternating.

There is no such requirement in the case of the slug. It can extract energy directly from the coil's magnetic field whether the field is changing or not. In fact, a coil powered by a dc current might very well be a better choice for a coil gun, if we could get a dc current large enough.

The voltage drop V_L over the coil does depend on the change in the current flowing through it, whether there is any slug present or not. That voltage is given by Equation (4). The instantaneous power consumed by the coil, in the absence of the slug, is the product of this voltage and the instantaneous current, namely:

$$P_L(t) = V_L(t)I = LI\frac{dI}{dt} \quad (15)$$

The energy which is absorbed by the coil is the power multiplied by the time over which it is drawn. If the current started from zero at time $t = 0$, then the energy absorbed by the coil is equal to:

$$\begin{aligned} E_L(t) &= \int_{\tau=0}^{t} \left(LI\frac{dI}{d\tau}\right) d\tau \\ &= \int_{\tau=0}^{t} LI dI \\ &= \tfrac{1}{2}LI^2 \qquad (16) \end{aligned}$$

The energy absorbed by the inductor is stored in the magnetic field which exists in and around the coil.

We can calculate the power consumed by the slug in a similar way. In Part I of this paper, we looked at the kinetic energy of the slug, which at any time is equal to:

$$K = \tfrac{1}{2}mv^2 \quad (17)$$

where m is the mass of the slug and v is its instantaneous speed. The power consumed by the slug P_S is the change in its kinetic energy (dK) which takes place in some small interval of time (dt). That is:

$$P_S = \frac{dK}{dt} = mv\frac{dv}{dt} \quad (18)$$

Now, the derivative of the slug's speed with respect to time is defined as its acceleration. From Newton's Law, $F = ma$, we know that the slug's acceleration is equal to the applied force F divided by the slug's mass. Therefore,

$$P_S = vF \quad (19)$$

This is the instantaneous power consumed by the slug. It must be supplied by the coil. Therefore, in the electrical circuit, the instantaneous power consumed by the coil will be the sum of Equations (15) and (19), thus:

$$\begin{aligned} P_{total} &= P_L + P_S \\ &= LI\frac{dI}{dt} + vF \quad (20) \end{aligned}$$

Anticipating what we will find in Part III of this paper, the force on the slug F is the product of two terms: (i) a spatial distribution which does not depend on time, and (ii) the square of the current flowing through the coil. As we did in Part I of this Paper, we will let $\mathcal{F}(z)$ be the spatial distribution of the force, which we can, say, calculate at a reference current of one Ampere. We can then write the total force acting on the slug as:

$$F(I, z) = I^2\mathcal{F}(z) \quad (21)$$

This nicely separates the dependence of the force on time from the dependence of the force on the slug's displacement (although the slug's location does also depend on time). Substituting this form for the force into Equation (20) gives:

$$\begin{aligned} P_{total} &= LI\frac{dI}{dt} + I^2 v\mathcal{F}(z) \\ &= I\left[L\frac{dI}{dt} + Iv\mathcal{F}(z)\right] \\ &= I[V_L + Iv\mathcal{F}(z)] \qquad (22) \end{aligned}$$

It makes eminent sense to define a "voltage drop" V_S for the slug. Since power is always the product of a current and a voltage, Equation (22) suggests that we define the following voltage drop:

$$V_S = Iv\mathcal{F}(z) \quad (23)$$

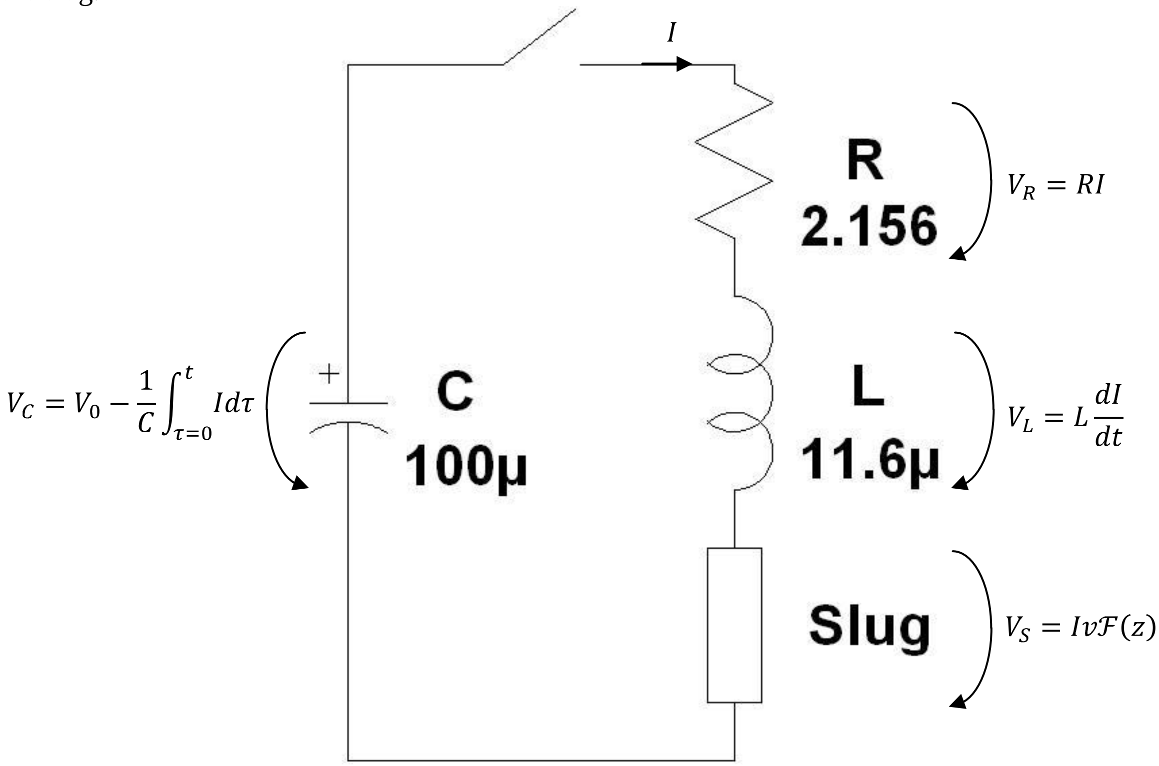
If we do this, then the power drawn by the coil-slug combination is equal to:

$$P_{total} = I[V_L + V_S] \quad (24)$$

This says that a good way to model the slug's electrical impact on the circuit is to place it in series with the coil, so that the same current flows through both. (This is not how we assumed it would be, in the schematic diagram above.) Treated in this way as a series component, the voltage drop of the slug is $Iv\mathcal{F}(z)$. For a normal resistor, the voltage drop is expressed by Ohm's Law as $V_R = IR$. Comparing the two expressions makes it clear that we can think of the slug as a variable resistor, whose resistance at any moment is given by $v\mathcal{F}(z)$, the product of its speed and the spatial component of the force factor.

The revised circuit schematic, taking the slug into account

The following schematic diagram shows how the electrical circuit should be represented to account for the slug.



For ready reference, I have shown the V-I characteristic of each component, including the slug. I have shown the slug as being in series with the ideal inductance of the coil. This is consistent with Equation (24), which has the slug making an additive (that is, series) contribution to the voltage around the loop.

Once again, the sum of the voltage drops around the circuit must add up to zero. Therefore:

$$\begin{aligned} V_C &= V_R + V_L + V_S \\ V_0 - \frac{1}{C}\int_{\tau=0}^{t} I d\tau &= RI + L\frac{dI}{dt} + Iv\mathcal{F}(z) \quad (25) \end{aligned}$$

It is useful to express the current I in Equation (25) in terms of the charge Q_C on the capacitor. This is easily done on the left-hand side – the voltage drop over the capacitor V_C is equal to the stored charge Q_C divided by the capacitance. Generally speaking, current is defined as the rate at which charge passes by a given point in the circuit. In our case, a positive current, as defined in the schematic, corresponds to decreasing charge on the capacitor, so that:

$$I = -\frac{dQ_C}{dt} \quad (26)$$

Substituting this into Equation (25) gives:

$$\frac{Q_C}{C} = -\frac{dQ_C}{dt}[R + v\mathcal{F}(z)] - L\frac{d^2Q_C}{dt^2}$$

$$\rightarrow \quad L\frac{d^2Q_C}{dt^2} = -\frac{dQ_C}{dt}[R + v\mathcal{F}(z)] - \frac{Q_C}{C} \quad (27) \quad Circuit$$

I have called this the Circuit equation because it encompasses all aspects of the electric circuit. However, we are going to have to solve this differential equation jointly with another, which describes the dynamics of the slug. Using Newton's Law:

$$a = \frac{F}{m}$$

$$\rightarrow \quad \frac{dv}{dt} = \frac{I^2\mathcal{F}(z)}{m} \quad (28) \quad Dynamics$$

We will also need the slug's speed v for Equation (27), which we get by integrating Equation (28).

$$v(t) - v(0) = \frac{1}{m}\int_{\tau=0}^{t} I^2\mathcal{F}(z)d\tau \quad (29) \quad Mechanics$$

<u>Numerically integrating the two equations</u>

A joint closed-form solution of Equations (27), (28) and (29) is not possible. However, they are in a form which makes numerical integration quite convenient. We will divide the slug's run into very short increments of time, whose length ΔT we will call a time step. At the beginning of each time step, we will know the following quantities, which will be results from the calculations for the previous time step:

for the slug	z_{start} and v_{start}
for the circuit	Q_{start} and I_{start}

In general, variables have the subscript $start$ for their values at the start of a time step and the subscript end for their values at the end.

We are also going to approximate the spatial distribution of the force $\mathcal{F}(z)$. Since the slug does not move very far during a short time step, the value of $\mathcal{F}(z)$ will be almost constant during the step. Let us assume that we can find some average value $\mathcal{F}_{mid}$ for $\mathcal{F}(z)$ which applies during the whole step. Since the slug's speed will not change by much during the time step, it being so short, it is reasonable to estimate that the slug's position at the end of the time step will be very close to:

$$z_{end}|_{est} = z_{start} + v_{start}\Delta T \quad (30)$$

We can look up in the force field table the values of the spatial component $\mathcal{F}(z)$ at the beginning and the end of the time step. The average value of the spatial component (average over distance, not time) will be:

$$\begin{aligned}\mathcal{F}_{avg} &= \tfrac{1}{2}(\mathcal{F}_{start} + \mathcal{F}_{end}|_{est}) \\ &= \tfrac{1}{2}[\mathcal{F}(z_{start}) + \mathcal{F}(z_{end}|_{est})] \quad (31)\end{aligned}$$

We first apply the Dynamics Equation (28) to find the slug's acceleration at the start of the time step:

$$\left.\frac{dv}{dt}\right|_{start} = \frac{I_{start}^2 \mathcal{F}_{avg}}{m} \quad (28')$$

Then, we apply the Mechanics Equation (29) to find the slug's speed at the end of time step.

$$v_{end} = v_{start} + \Delta T \left.\frac{dv}{dt}\right|_{start} \quad (29')$$

Using the Taylor series expansion of any function, we can write the slug's position at the end of the time step as follows:

$$\begin{aligned} z(\tau) &= z(\tau = 0) + \left.\frac{dz}{d\tau}\right|_{\tau=0} \tau + \tfrac{1}{2} \left.\frac{d^2 z}{d\tau^2}\right|_{\tau=0} \tau^2 + \mathcal{O}(\tau^3) \\ &= z_{start} + \tau v_{start} + \tfrac{1}{2}\tau^2 \left.\frac{dv}{dt}\right|_{start} \\ \rightarrow \quad z_{end} &= z_{start} + \Delta T v_{start} + \tfrac{1}{2}\Delta T^2 \left.\frac{dv}{dt}\right|_{start} \qquad (32) \end{aligned}$$

Then, we apply the Circuit Equation (27) to calculate the second derivative of the charge:

$$\begin{aligned} L\frac{d^2 Q_C}{dt^2} &= -\frac{dQ_C}{dt}[R + v\mathcal{F}(z)] - \frac{Q_C}{C} \\ \rightarrow \quad L\left.\frac{d^2 Q_C}{dt^2}\right|_{start} &= -\left.\frac{dQ_C}{dt}\right|_{start}\left(R + v_{start}\mathcal{F}_{avg}\right) - \frac{Q_{C\,start}}{C} \\ \rightarrow \quad \left.\frac{d^2 Q_C}{dt^2}\right|_{start} &= \frac{1}{L}\left[I_{start}\left(R + v_{start}\mathcal{F}_{avg}\right) - \frac{Q_{C\,start}}{C}\right] \qquad (27') \end{aligned}$$

We can now determine what the current will be at the end of the time step:

$$\begin{aligned} \frac{d}{dt}\left(\frac{dQ_C}{dt}\right) &= \frac{d^2 Q_C}{dt^2} \\ \rightarrow \quad \int_{\tau=0}^{\tau=\tau} \frac{d}{d\tau}\left(\frac{dQ_C}{dt}\right) d\tau &= \int_{\tau=0}^{\tau=\tau} \left(\frac{d^2 Q_C}{dt^2}\right) d\tau \\ \rightarrow \quad \left.\frac{dQ_C}{dt}\right|_{\tau=\tau} - \left.\frac{dQ_C}{dt}\right|_{start} &\cong (\tau - 0)\left.\frac{d^2 Q_C}{dt^2}\right|_{start} \\ \text{so, at } \tau = \Delta T \qquad -I_{end} + I_{start} &\cong \Delta T \left.\frac{d^2 Q_C}{dt^2}\right|_{start} \qquad (33) \end{aligned}$$

Using Taylor's expansion once again, we can write the charge at the end of the time step as follows:

$$\begin{aligned} Q_C(\tau) &= Q_C(\tau = 0) + \left.\frac{dQ_C}{d\tau}\right|_{\tau=0} \tau + \tfrac{1}{2}\left.\frac{d^2 Q_C}{d\tau^2}\right|_{\tau=0} \tau^2 + \mathcal{O}(\tau^3) \\ \rightarrow \quad Q_{C\,end} &\cong Q_{C\,start} - \Delta T I_{start} + \tfrac{1}{2}\Delta T^2 \left.\frac{d^2 Q_C}{dt^2}\right|_{start} \qquad (34) \end{aligned}$$

Having completed the calculations for this time step, we can now move on to the next time step, where the quantities z_{end}, v_{end}, $Q_{C\,end}$ and I_{end} become the corresponding starting quantities for the next time step.

Monitoring the error of the numerical integration

The errors which arise from using the approximations during each time step will accumulate during the course of a run of the slug. Fortunately, there is a delightful way to monitor the cumulative error. Energy must be conserved during the run. We can calculate the total energy in the system at any point in time. The following table sets out the energies of each component at the beginning and end of a time step.

Component	Energy at start	Energy at end	Change in energy
Capacitor	$\frac{1}{2}CV_{C\,start}^2$, or $\frac{1}{2C}Q_{C\,start}^2$	$\frac{1}{2}CV_{C\,end}^2$, or $\frac{1}{2C}Q_{C\,end}^2$	$\frac{1}{2C}\left(Q_{C\,end}^2 - Q_{C\,start}^2\right)$
Resistor	Cumulative heat_{start}	Cumulative heat_{end}	$\int RI^2 dt$
Inductor	$\frac{1}{2}LI_{start}^2$	$\frac{1}{2}LI_{end}^2$	$\frac{1}{2}L\left(I_{end}^2 - I_{start}^2\right)$
Slug	$\frac{1}{2}mv_{start}^2$	$\frac{1}{2}mv_{end}^2$	$\frac{1}{2}m\left(v_{end}^2 - v_{start}^2\right)$

Energy stored in the capacitor

The energy stored in a capacitor is usually written in terms of its voltage, as $\frac{1}{2}CV^2$. Since the charge Q stored in the capacitor is equal to CV, the energy stored can also be written in terms of the charge as $Q^2/2C$. Note that, since the capacitor is the source of the energy in our application, $Q_{C\,end}$ will be less than $Q_{C\,start}$ and the change in the capacitor's energy will be negative. In any event, the capacitor's energy at the end of any particular time step is given by:

$$E_{C\,end} = \frac{Q_{C\,end}^2}{2C} \qquad (35)$$

Energy burned by the resistor

The resistor does not hold its energy *per se*, but we can describe the energy it burns as the "cumulative heat" dissipated up to a certain time during the run. We will, of course, keep a running total of this heat, for which we will use the symbol $E_{R\,start}$ at the beginning of each time step. The resistor burns heat at the instantaneous rate of RI^2, so the heat burned off by the resistor during one time step will be equal to:

$$\begin{aligned}
\Delta E_R &= \int_{\tau=0}^{\tau=\Delta T} RI^2 d\tau \\
&= R\int_{\tau=0}^{\tau=\Delta T} \left(I_{start} - \tau \frac{d^2Q_C}{dt^2}_{start}\right)^2 d\tau \\
&= R\int_{\tau=0}^{\tau=\Delta T} \left(I_{start}^2 - 2\tau I_{start}\frac{d^2Q_C}{dt^2}_{start} + \tau^2 \frac{d^2Q_C}{dt^2}^2_{start}\right) d\tau
\end{aligned}$$

I have kept terms of higher-order powers in time because this calculation is part of a check of the accuracy. While the extra terms may not be significant, keeping them removes one more source of uncertainty. Continuing with the integration gives:

$$\begin{aligned}\Delta E_R &= R\left(\tau I_{start}^2 - \tau^2 I_{start} \frac{d^2Q_C}{dt^2}_{start} + \frac{1}{3}\tau^3 \frac{d^2Q_C}{dt^2}^2_{start}\right)\Bigg|_{\tau=0}^{\tau=\Delta T} \\ &= R\left(I_{start}^2 - \Delta T I_{start} \frac{d^2Q_C}{dt^2}_{start} + \frac{1}{3}\Delta T^2 \frac{d^2Q_C}{dt^2}^2_{start}\right)\Delta T \qquad (36)\end{aligned}$$

Finally, the cumulative heat burned off by the resistor by the end of a time step can be written as:

$$E_{R\ end} = E_{R\ start} + \Delta E_R \qquad (37)$$

Energy stored in the inductor

The energy stored in the inductor at the end of a time step will be equal to:

$$E_{L\ end} = \frac{1}{2} L I_{end}^2 \qquad (38)$$

Energy stored in the slug

The energy stored in the slug – its kinetic energy – at the end of a time step is similar:

$$E_{S\ end} = \frac{1}{2} m v_{end}^2 \qquad (39)$$

Total energy

The total energy of the system, including energy dissipated as heat, should always equal the energy which was stored in the capacitor before the start of a run. At the end of each time step, it should be the case that:

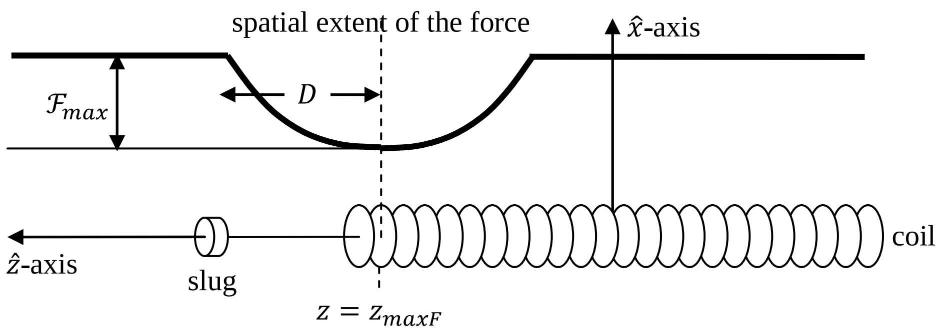
$$E_{R\ end} = E_{R\ end} + E_{L\ end} + E_{S\ end} \qquad (40)$$

Monitoring the sum of the voltage drops

A second useful check during the numerical integration is to ensure that the sum of the voltage drops around the circuit is always zero. This is easily done by evaluating the quantities in the Circuit Equation (27) at the start or end of each time step. If the sum of the voltage drops varies from zero, it is a sign that the time step ΔT is likely too large, so that the approximations used to linearize the speed and model the charge as a quadratic are too imprecise. Decreasing the time step should fix that.

Results of the numerical integration

For the purpose of this example, we will use the same spatial distribution for the force field as we used as an example in Part I of this paper. There, we assumed that the spatial distribution of the force field had a parabolic shape, reaching a maximum value somewhere near the face of the coil and extending a fixed distance on either side of that maximum. The following figure shows the shape of the force field.



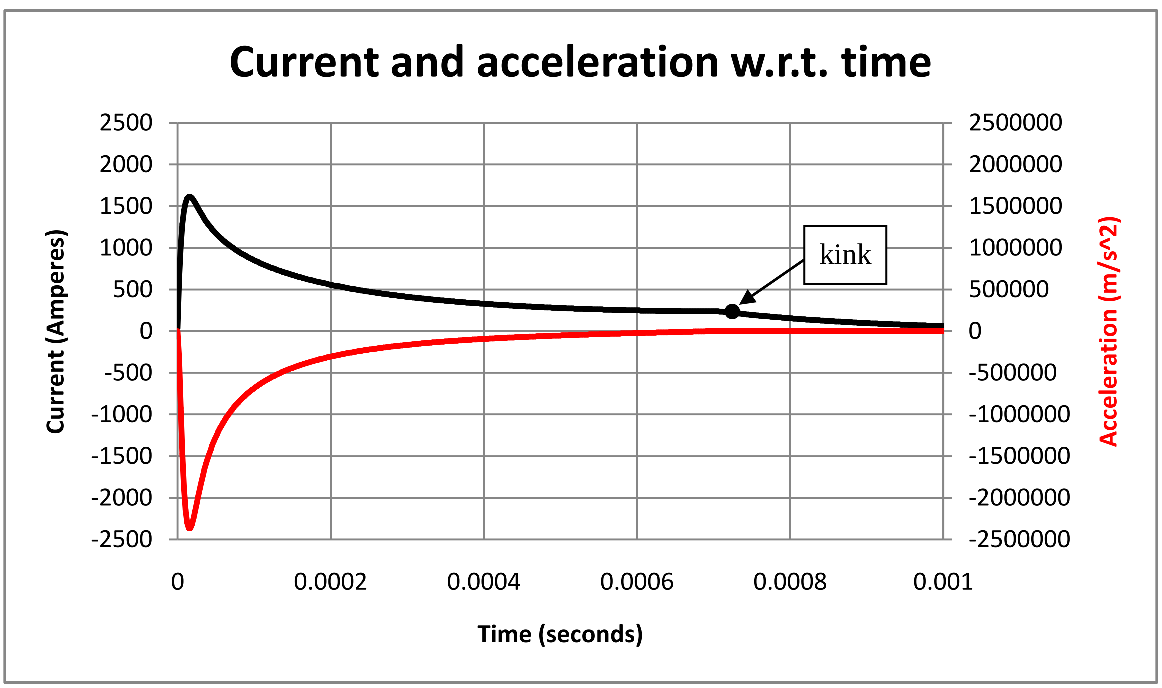
In the earlier example, the spatial distribution was quantified as follows:

$$\mathcal{F}(z) = -|\mathcal{F}_{max}|\left[1 - \left(\frac{z - z_{maxF}}{D}\right)^2\right] \text{ for the range } |z - z_{maxF}| \le D \quad (Part\ I - 10)$$

For the example in Part I, we assumed that the maximum value of the force was 100 Newtons, that the peak of the force occurred at a distance $z_{maxF} = 24$ cm from the center of the coil and that its spatial "extent" was characterized by $D = 10$ cm. Note that the coil we are simulating is 50 cm long, so its face is 25 cm from the center of the coil. The force field was zero outside of the range $|z - z_{maxF}| \le D$ and, to ensure this, the current was "stopped" once the slug had passed through the force field.

We need to change things a little. The total force is the product of $\mathcal{F}(z)$ and the square of the current and we will set $\mathcal{F}_{max}$ taking this into account. The graph below are based on $\mathcal{F}_{max} = -0.01$. I chose this value so that a current of 100 Amperes would give rise to a total force of 100 Newtons at its peak spatial point. If the current is higher, the strength of the force field will everywhere be proportionately greater.

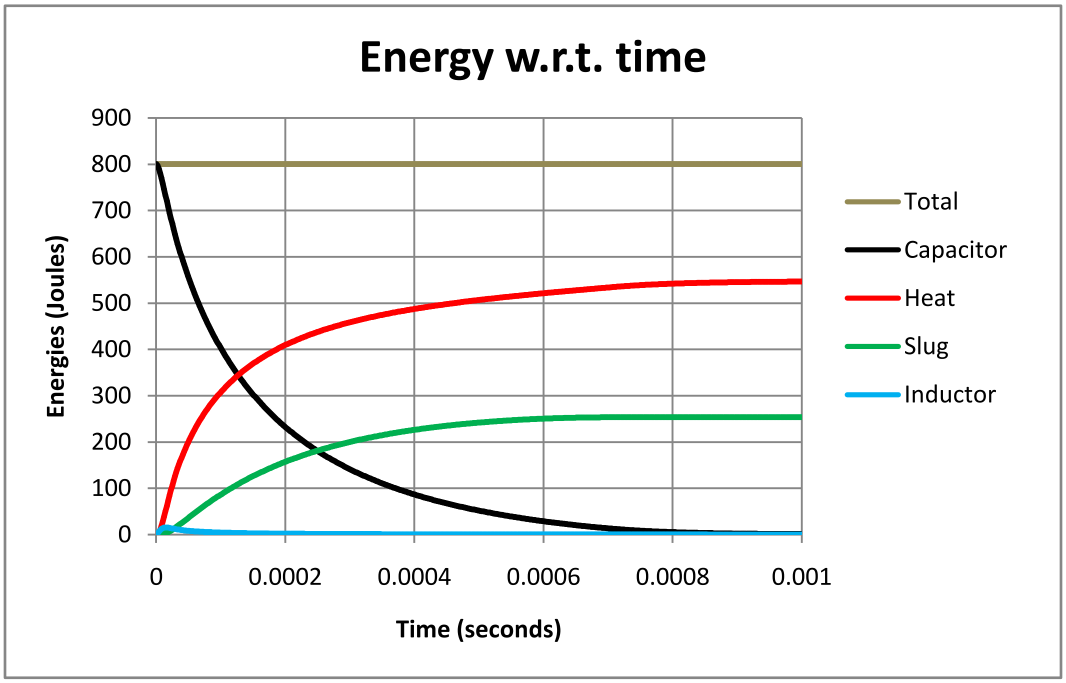
Appendix "A" attached hereto is a listing of a short Visual Basic program (named *Integration3*) which carries out this numerical integration. The following curves show what happens when the slug (with a mass of 10 grams) is placed 27 cm from the origin (being 7 cm inside the left edge of the force field and 3 cm to the left of the optimal displacement) and the capacitor bank is discharged.



This graph shows the current (in black) and the slug's acceleration (in red) over a time period of one microsecond. There is a kink in the current's waveform, at a time of about 720 µs. This is the instant when the slug has passed through the horizontal extent of the force field and the slug, in essence, disappears from the electrical circuit. Thereafter, the circuit behaves as a normal R-L-C circuit.

The values for the acceleration are negative, of course, because the slug is being accelerated towards the right, in the direction of the negative $\hat{z}$-axis.

The following graph shows the system energies over this same period of time.



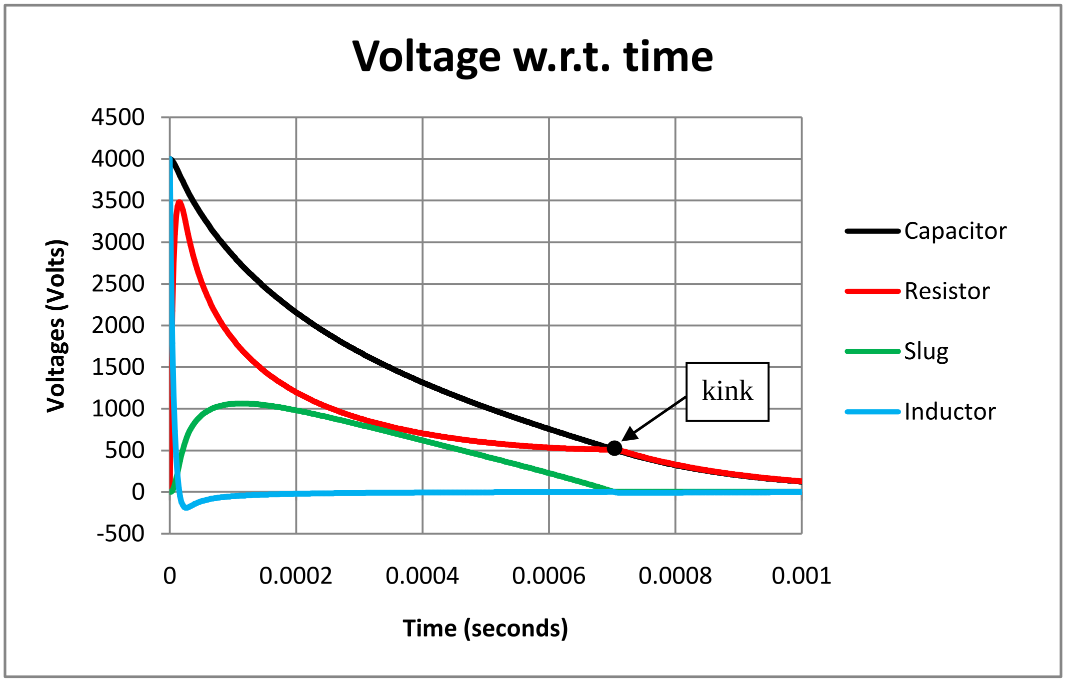
The capacitor began the run holding 800 Joules of energy. The principal user of energy is the resistor. By the end of the run, it had burned off about 540 Joules, or 67.5% of the original energy. The slug managed to receive about 260 Joules, or 32.5% of the original energy. The inductor did not retain any energy in its magnetic field. In fact, the inductor was hardly a player in the activity at all. At its maximum, the energy stored in the magnetic field was about 15 Joules.

Let me just say a word about the accuracy of the numerical integration. The following table sets out the total system energy at the end of the run against the time step used in the numerical integration.

Time step ΔT	Ending energy E_T
50 ns	800.15 Joules
100 ns	800.30 Joules
500 ns	801.52 Joules
1 µs	803.10 Joules
5 µs	819.17 Joules
10 µs	failure

When the time step is 10 µs or greater, the calculations fail. Modeling the charge as a quadratic function of time is not sufficient – the values of some quantities vary too widely from one time step to the next.

The following graph shows the waveforms of the component voltages during the run.



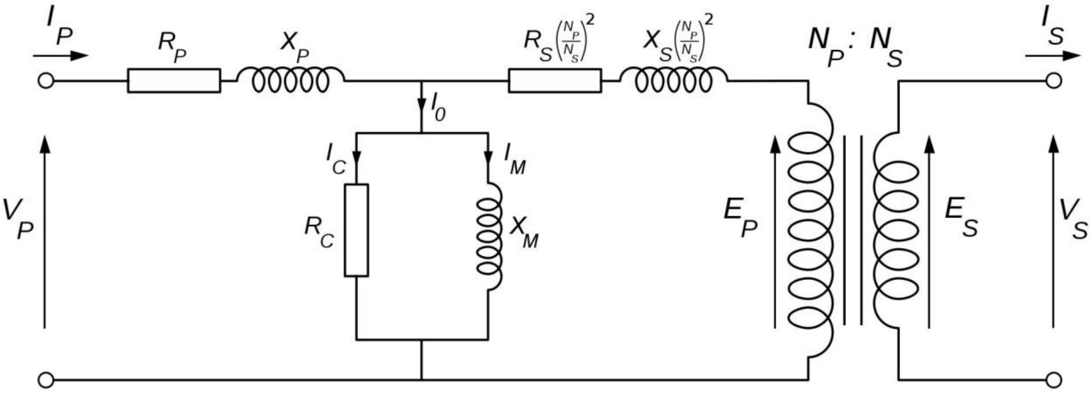
Because the current peaks so quickly after the start of the run, the inductor spends most of the run with a small negative voltage drop. The "voltage drop" over the slug, which is proportional to the product of the slug's speed and the current, with a gradual contribution from the shape of the spatial distribution of the force field, has an interesting waveform.

A physical interpretation of the "voltage drop" over the slug

I have been troubled by the slug's placement in the schematic diagram. It is placed in series with the ideal inductance of the coil, with a voltage-current characteristic $V_S = Iv\mathcal{F}(z)$. This voltage-current characteristic is identical to that of the resistor, $V_R = IR$. It would seem that the coil is a variable resistance, with its resistance given by $v\mathcal{F}(z)$.

This is troubling because it is so unlike the model one uses for a transformer, which transfers energy via the magnetic field, not unlike our coil-slug combination. One models the ideal transformer as the parallel combination of: (i) the primary coil with no load, and (ii) the transformed impedance of the load. Importantly, the voltage over these two components is the same.

On the other hand, another way to think about the slug is as a "leak" of flux from the inductor. The leak of magnetic flux in a real transformer, due to less than perfect coupling between the primary and secondary coils, is a shortcoming of the component. In our application, we want to maximize this so-called "leak". In any event, the slug extracts energy directly from the magnetic field and could be modeled in the same way as flux leakage is modeled in the case of a less-than-ideal transformer. The following schematic (taken from Wikipedia) is what is known as the exact equivalent of an ideal transformer. Although it is called "exact", it still differs from a real transformer by excluding the effects of non-linearity and saturation. In the figure, X_M is the no-load primary coil and R_P, in series with the no-load primary coil, is the Ohmic resistance of the wire in the winding. We already have both of these components in our circuit. In the figure, R_C is the effective resistance of the core material, that is, the losses in the core due to eddy current and such like. Our coil has an air-core so core losses are relatively unimportant.



The interesting component in the figure is X_P. This inductor represents the flux leakage from the primary coil. R_S and X_S are the Ohmic resistance and flux leakage in the secondary coil, transformed by the square of the turns-ratio.

If we model the slug as a flux leakage, instead of as a secondary circuit, then the electrical equivalent of our slug will take the place of X_P in Wikipedia's schematic diagram. The secondary side of the circuit can be entirely ignored. Flux leakage, in the normal sense, absorbs power like an inductor, so the component which models flux leakage is normally shown as an inductor, as above. Our slug, on the other hand, absorbs power like a variable resistor. This is how we modeled it in our schematic diagram above, using the European circuit symbol for a resistor. The important thing is that flux leakage, both in the normal sense and in the case of our slug, is represented as being in series with the no-load coil. The same current flows through both.

In conclusion, the quantity we have referred to as the "voltage drop" over the slug should be thought of as the voltage drop caused by flux leakage. A voltmeter placed over the coil would show a voltage equal to: (i) the voltage drop over the no-load coil, and (ii) the voltage drop due to flux leakage. The waveforms which resulted from the numerical integration show that the voltage drop over the no-load coil is very small compared with the voltage drop due to "flux leakage" equivalent of our slug.

Can we arrange the resistance and inductance to do better?

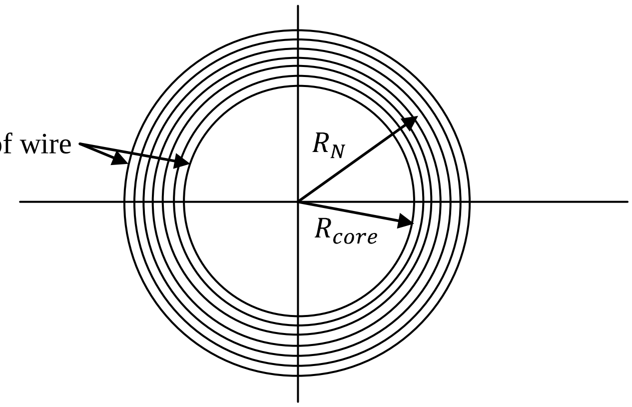
The resistance and inductance used in the preceding example were based on a coil wound with 96 turns of wire. (Note, though, that the spatial distribution of the force was not based on any physical coil.) We have a lot of choice in winding the coil. We can add more turns, either to the length or on the outside, which will increase the coil's resistance and increase its inductance even more. In other words, we have some control over the ratio of the coil's resistance to its inductance.

Let us take a preliminary look at the consequences of adding layers of turns to the coil. For the time being, let us continue to use #4 gauge enameled copper wire, whose characteristics were set out above. For convenience, we will also keep the length of the coil the same – 49.81 cm – so that we can wind 96 turns on each layer. The following figure shows the dimensions we need to consider when adding layers. The cross-section of the coil is shown as are several layers of the winding.

We will use the following symbols:

- $H_{coil} = 49.81$ cm is the length of the coil,
- $D_{wire} = 0.005189$ meters is the outside diameter of a single wire,

- N_{layers} is the number of layers of wire,
- R_{core} = 1 cm is the inside radius of the coil and
- R_N is the radius of each turn in the N^{th} layer.



We will assume that the wire is wound onto the coil in such a way that the length of its centerline is preserved. As it is forced into a circular shape, the inner side of the wire experiences compression to the same extent that the outer side of the wire experiences expansion. We will ignore the effects near the end of each layer, where the wire must extend up to the next layer. We will also ignore the spiral effects of the coil's real shape and consider the coil to be a set of truly circular turns. Then, the radius of the N^{th} layer, measured to the centerline of the wire, can be written as:

$$R_N = R_{core} + \left(N - \tfrac{1}{2}\right) D_{wire} \quad (41)$$

The longitudinal length of wire in the N^{th} layer is equal to

$$96 \times 2\pi \times R_N$$

and the total longitudinal length of wire in the whole coil is equal to

$$\sum_{N=1}^{N_{layers}} 96 \times 2\pi \times \left[R_{core} + \left(N - \tfrac{1}{2}\right) D_{wire}\right]$$

Since the resistance per unit length of the wire is 0.0008153Ω per meter (equal to 0.2485Ω per thousand feet), we can compute the resistance of the coil as:

$$Resistance_{coil} = 0.0008153 \times 96 \times 2\pi \sum_{N=1}^{N_{layers}} \left[R_{core} + \left(N - \tfrac{1}{2}\right) D_{wire}\right] \Omega \quad (42)$$

The inside radius of the first turn is equal to R_{core} and the outside radius of the last turn is equal to $R_{outer} = R_{core} + (N_{layers} D_{wire})$, so we can write the average radius of the coil as:

$$\begin{aligned} R_{avg} &= \tfrac{1}{2}\{R_{core} + [R_{core} + (N_{layers} D_{wire})]\} \\ &= R_{core} + (\tfrac{1}{2} N_{layers} D_{wire}) \quad (43) \end{aligned}$$

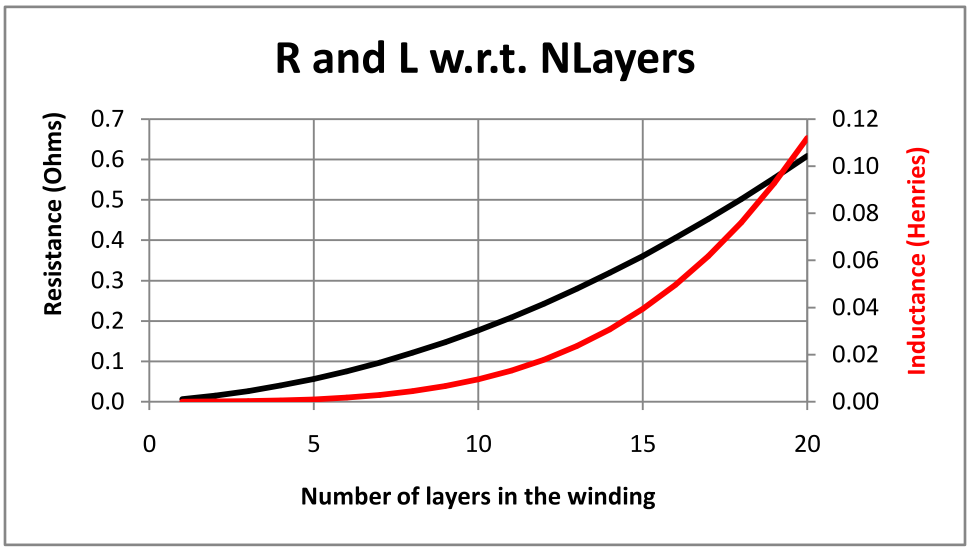
The average area of the cross-section of the coil can then be found as:

$$\begin{aligned} A_{avg} &= \pi R_{avg}^2 \\ &= \pi\left[R_{core} + (\tfrac{1}{2}N_{layers}D_{wire})\right]^2 \qquad (44) \end{aligned}$$

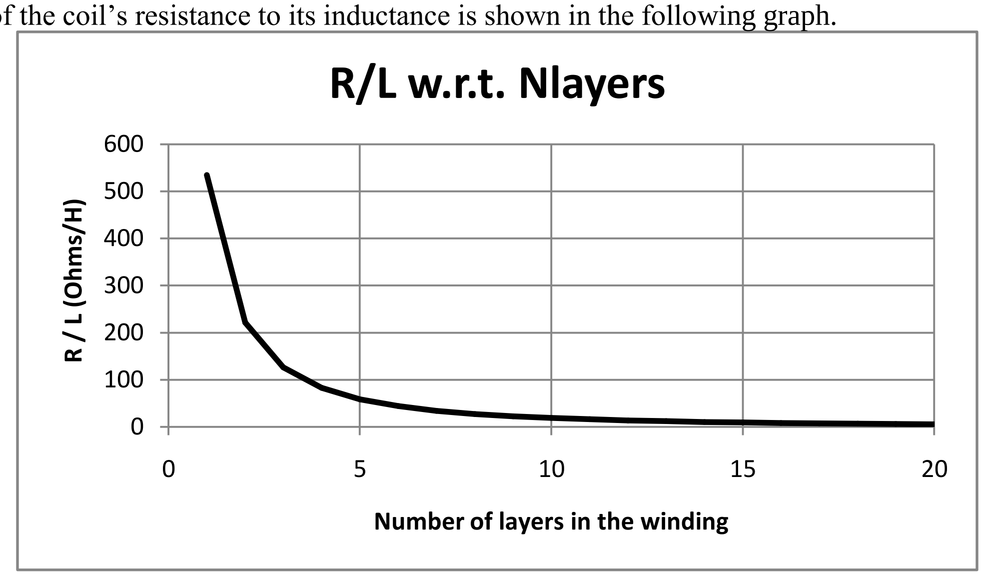
So long as we do not wind so many layers that the coil ceases to be "much longer than its diameter", then we can continue to use Equation (1) to estimate the coil's inductance.

$$\begin{aligned} Inductance_{coil} &= \mu_0 \frac{N_{total}^2 A_{coil}}{H_{coil}} \\ &= (4\pi \times 10^{-7}) \frac{\left(96 \times N_{layers}\right)^2 \pi\left[R_{core} + (\tfrac{1}{2}N_{layers}D_{wire})\right]^2}{H_{coil}} \qquad (45) \end{aligned}$$

The following graph shows the coil's resistance and inductance as we add layers. The resistance is the black curve and the inductance is the red curve.



The ratio of the coil's resistance to its inductance is shown in the following graph.



The R/L ratio changes significantly by the time we add three layers. Suppose we do that. With three layers, the resistance and inductance of the coil are:

$$R_{coil} = 0.026\Omega$$
$$L_{coil} = 208\mu\text{H}$$

The total series resistance in the circuit is then:

$$\begin{aligned} R \quad &= R_{esr} + R_{sw} + R_{coil} \\ &= 2 + 0.15 + 0.026 \\ &= 2.176\Omega \end{aligned}$$

Before running *Integration3* again, note that this combination of resistance and inductance changes the nature of the original circuit, excluding the slug. Before, we had two exponential solutions which corresponded to the two real roots of the characteristic equations. Recall that the two roots of the characteristic equation are given by Equation (8), which I repeat here for convenience.

$$\beta = \frac{-R \pm \sqrt{R^2 - \frac{4L}{C}}}{2L} \quad (8)$$

With the new resistance and inductance, the term under the square root is now negative.

$$\begin{aligned} \beta \quad &= \frac{-2.176 \pm \sqrt{2.176^2 - \frac{4 \times 0.000208}{0.0001}}}{2 \times 0.000208} \\ &= \frac{-2.176 \pm \sqrt{-3.585}}{0.000416} \end{aligned} \quad (46)$$

If we use the symbol j for the imaginary number, then the two values of β are:

$$\beta = -5230 \pm j4550 \quad (47)$$

There are two solutions, of course, each corresponding to one of the values of β. As we did before, the total current is some linear combination of the two solutions. Using two coefficients α_1 and α_2, as we did before, the total current can be written as:

$$I(t) = \alpha_1 e^{-5230t+j4550t} + \alpha_2 e^{-5230t-j4550t} \quad (48)$$

We have the same initial conditions as we did before, too. The first of them is that the current at time $t = 0$ is zero. Substituting $t = 0$ into Equation (48) gives us $\alpha_2 = -\alpha_1$. That enables us to re-write Equation (48) through the following steps:

$$\begin{aligned} I(t) \quad &= \alpha_1 e^{-5230t+j4550t} - \alpha_1 e^{-5230t-j4550t} \\ &= \alpha_1 e^{-5230t}\left(e^{+j4550t} - e^{-j4550t}\right) \\ &= \alpha_1 e^{-5230t}[(\cos 4550t + j\sin 4550t) - (\cos 4550t - j\sin 4550t)] \\ &= 2\alpha_1 e^{-5230t}(j\sin 4550t) \end{aligned} \quad (49)$$

I have used the identity that $e^{j\theta} = \cos\theta + j\sin\theta$, for any argument θ. I have also used the symmetries of the cosine and sine functions, namely, $\cos(-\theta) = \cos\theta$ and $\sin(-\theta) = -\sin\theta$, for any argument θ. We are left with a j, but that does not necessarily mean that the result will be imaginary when we are finished. Let us just keep plugging away.

Our next step will be to apply the second initial condition – that the voltage drop over the capacitor will initially be imposed entirely over the idealized inductor. As before, the voltage over the inductor is given by circuit Equation (4). We must first take the derivative, as follows:

$$\begin{aligned} V_L(t) &= L\frac{dI(t)}{dt} \\ &= L\frac{d}{dt}[2\alpha_1 e^{-5230t}(j\sin 4550t)] \\ &= 2L\alpha_1 j\frac{d}{dt}(e^{-5230t}\sin 4550t) \\ &= 2L\alpha_1 j(-5230e^{-5230t}\sin 4550t + 4550e^{-5230t}\cos 4550t) \\ &= 2L\alpha_1 je^{-5230t}(-5230\sin 4550t + 4550\cos 4550t) \qquad (50) \end{aligned}$$

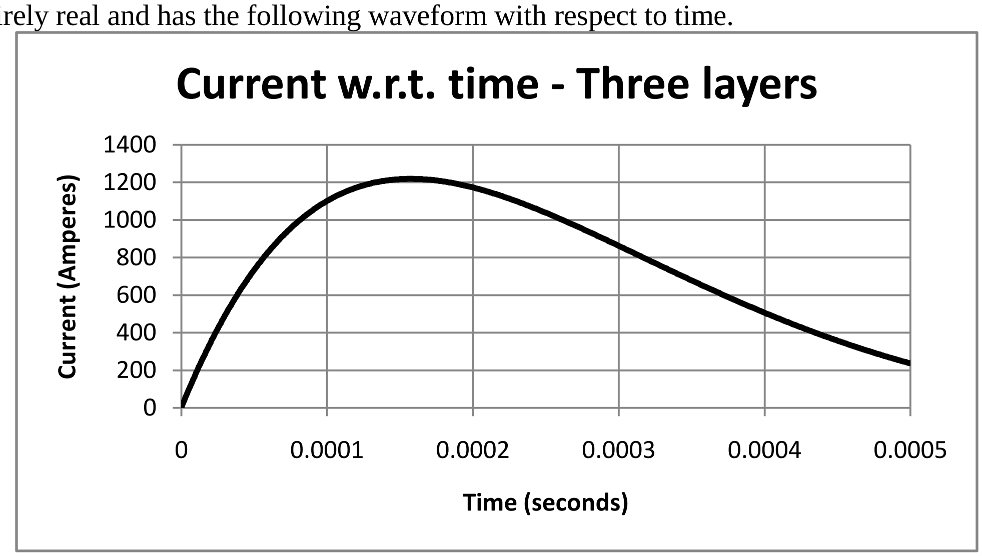
Then, we can apply the actual condition, that $V_L = V_0$ at time $t = 0$. This gives:

$$\begin{aligned} V_0 &= 2L\alpha_1 je^0(-5230\sin 0 + 4550\cos 0) \\ &= 9100L\alpha_1 j \\ \rightarrow \quad \alpha_1 &= \frac{V_0}{9100Lj} \qquad (51) \end{aligned}$$

So, it turns out that α_1 is also an imaginary number, albeit a constant. Substituting this value for α_1 into the solution for $I(t)$ gives:

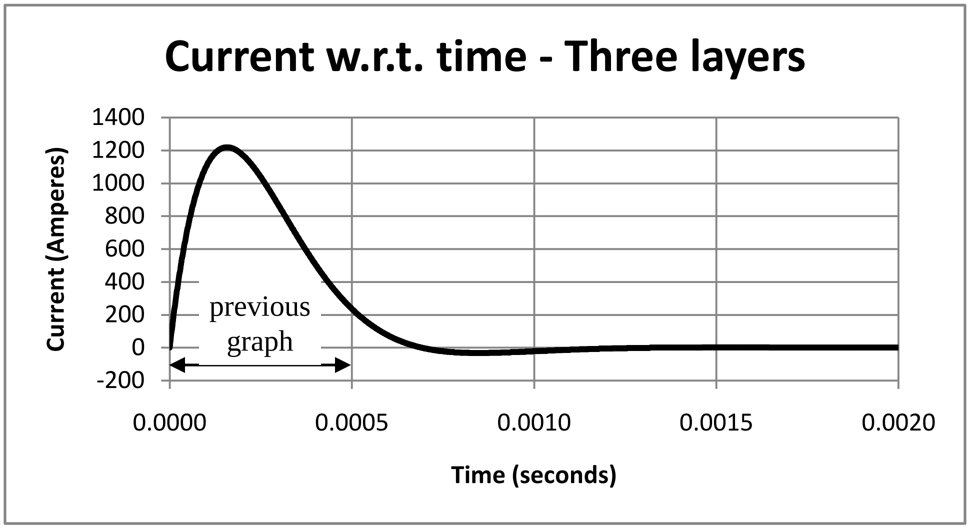
$$\begin{aligned} I(t) &= 2\left(\frac{V_0}{9100Lj}\right)e^{-5230t}(j\sin 4550t) \\ &= \frac{V_0}{4550L}e^{-5230t}\sin 4550t \qquad (52) \end{aligned}$$

This is entirely real and has the following waveform with respect to time.



At first blush, this appears to be a much "better" current than we had before. It rises more slowly, it peaks later and it decays more slowly, than the coil with only a single layer. This will give the slug more time, and more distance, through which to be accelerated.

So, what is the downside? The downside happens a little later in time. The graph above showed the first one-half millisecond. The following graph shows what happens over a longer period of time – two milliseconds.



The circuit oscillates. The energy stored in the inductor becomes more important than it was before. There comes a time during the run when the inductor releases more energy than the capacitor, and the current starts to flow in the negative direction – charging up the capacitor. This may or may not cause problems with the power supply. As we increase the inductance relative to the resistance, the oscillations will become more pronounced.

Fortunately, the reversal of the current is not a problem for the slug. This is important. **<u>The force exerted on the slug will be an attractive force no matter which way the current flows.</u>** Until such time as the slug passes through the center of the coil, oscillations in the current would not be a problem.

<u>The situation with ten layers on the coil</u>

If we wind ten layers of wire on the coil, its resistance and inductance are:

$$R_{coil} = 0.177\Omega$$
$$L_{coil} = 9.44\text{mH}$$

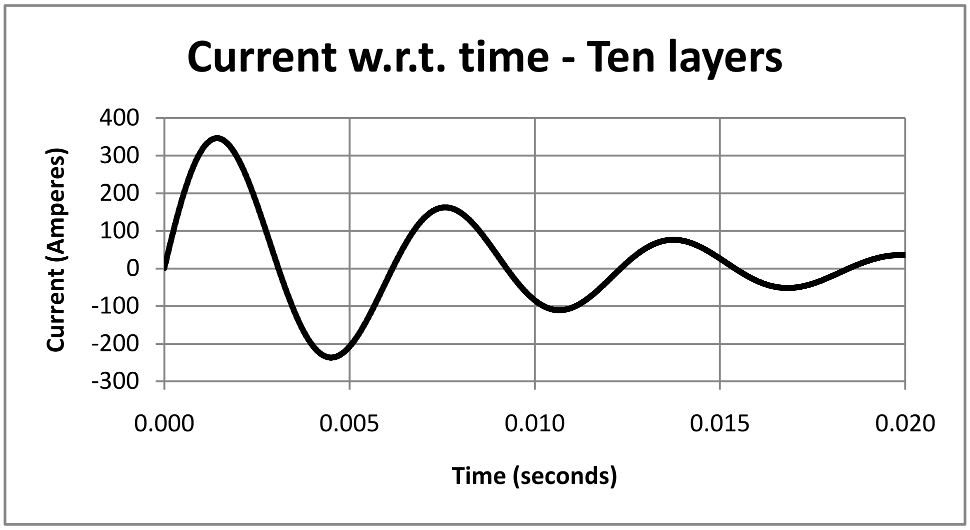
The main milestones in the calculations of the current are then as follows:

$$\beta = -123 \pm j1020$$
$$I(t) = 2\alpha_1 e^{-123t}(j \sin 1020t)$$
$$V_L(t) = 2L\alpha_1 j e^{-123t}(-123 \sin 1020t + 1020 \cos 1020t)$$
$$\alpha_1 = \frac{V_0}{2040Lj}$$

And, the final solution for the current is:

$$I(t) = \frac{V_0}{1020L} e^{-123t} \sin 1020t$$

This current has the following waveform. (Take note, though, that this is the current without any slug.)



Under the influence of this current, the slug would receive its acceleration in a series of pulses. The graph shows six and one-half pulses over the course of 20 milliseconds. On the other hand, the maximum current is now only about 350 Amperes. ("Only" is a relative term – 350 Amperes is still a huge current.)

Multiple layers have an additional benefit, which is extremely important here. Ten layers carrying a given current produce ten times as much force on the slug as does a single layer carrying the same current. With the lower current, the force exerted at any time on the slug will also be lower. But, the slug will be accelerated through a longer distance.

Is there an optimal number of layers?

There is a tradeoff among the many variables we can change. As we increase the number of layers, the maximum current during a run decreases, but the current and corresponding force act for a longer time and, therefore, over a greater distance of the slug's travel. If the slug's acceleration is too low, it does not get as much benefit from the spatial distribution of the force as it could have. If the acceleration is too high, the slug will pass through the spatial extent of the force field before the capacitor has delivered as much energy as it could have.

As we add layers, the current decreases. However – and this is extremely important – the total force acting on the slug is now proportional to the square of the number of layers multiplied by the current. When we wrote Equation (21) for the total force acting on the slug, we assumed that the coil had a single layer. When the coil has N_{layers} in its winding, the total force acting on the slug should be revised to:

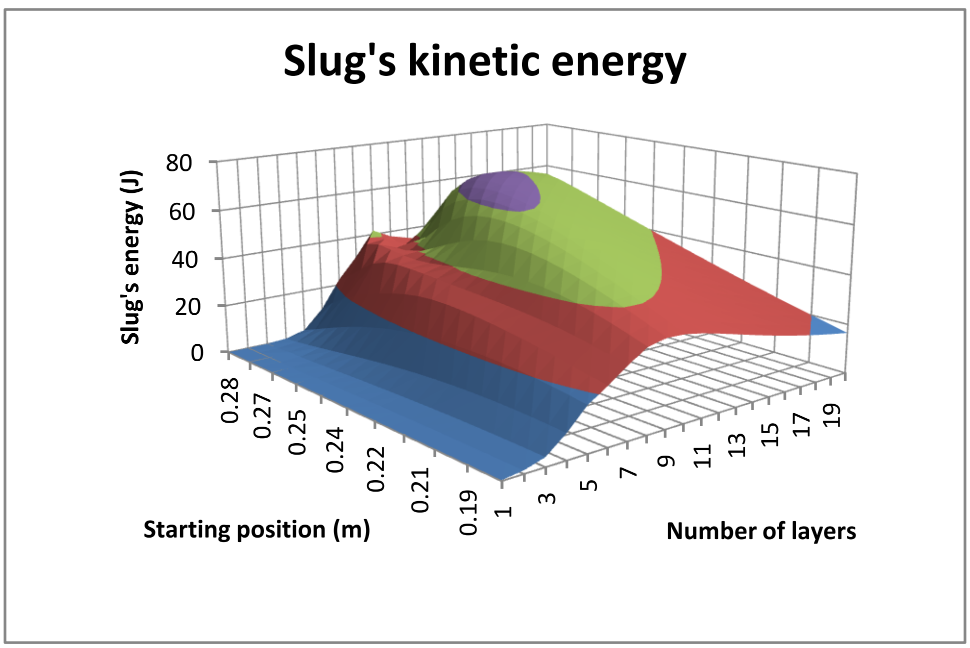
$$F(I, z) = \left(N_{layers} I\right)^2 \mathcal{F}(z) \quad (21_{Nlayers})$$

Adding layers may decrease the current flowing through the circuit, but the total force acting on the slug may actually increase. Let me re-state here the revised form which some of the equations above take when there is more than one layer.

$$F(I,z) = \left(N_{layers} I\right)^2 \mathcal{F}(z) \qquad (21_{Nlayers})$$
$$P_{total} = I\left[V_L + N_{layers}^2 I v \mathcal{F}(z)\right] \qquad (22_{Nlayers})$$
$$V_S = N_{layers}^2 I v \mathcal{F}(z) \qquad (23_{Nlayers})$$
$$\frac{dv}{dt} = \frac{\left(N_{layers} I\right)^2 \mathcal{F}(z)}{m} \qquad (28_{Nlayers}) \quad Dynamics$$
$$v(t) - v(0) = \frac{1}{m}\int_{t=0}^{t} \left(N_{layers} I\right)^2 \mathcal{F}(z) dt \qquad (21_{Nlayers}) \quad Mechanics$$
$$\frac{dv}{dt}_{start} = \frac{\left(N_{layers} I_{start}\right)^2 \mathcal{F}_{avg}}{m} \qquad (28'_{Nlayers})$$
$$\frac{d^2 Q_C}{dt^2}_{start} = \frac{1}{L}\left[I_{start}\left(R + N_{layers}^2 v_{start} \mathcal{F}_{avg}\right) - \frac{Q_{C\,start}}{C}\right] \qquad (27'_{Nlayers}) \quad Circuit$$

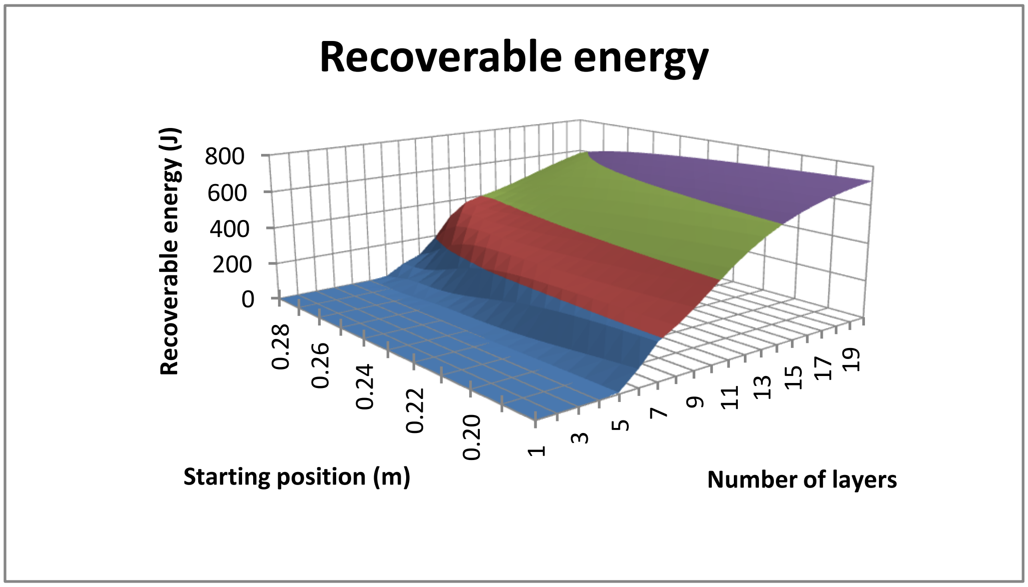
In this section, we will integrate a number of test cases to determine if there is such a thing as an optimal number of layers. In essence, we will repeat the integration in *Integration3* a number of times. We will look at the differences as we vary the number of layers in the coil from one to ten. As we observed in Part I of this paper, the kinetic energy of the slug at the end of a run also depends on its starting position with respect to the force field. Therefore, we will also carry out multiple tests for each number of layers. For each number of layers, we will look at the differences as we vary the starting position of the slug. We will try 21 different starting positions, starting 33 cm from the center of the coil (which is 1 cm to the right of the left-most extent of the force field) and progressing to 23 cm from the center of the coil (which is 1 cm to the right of the force field's peak). Recall that the maximum of the force field's spatial distribution is located 24 cm from the center of the coil. For each pair of number of layers and starting position, we will make a note of the slug's final kinetic energy.

Appendix "B" attached hereto is a listing of *Integration4*, which carries out this procedure. The slug's final kinetic energy is plotted in the following graph.



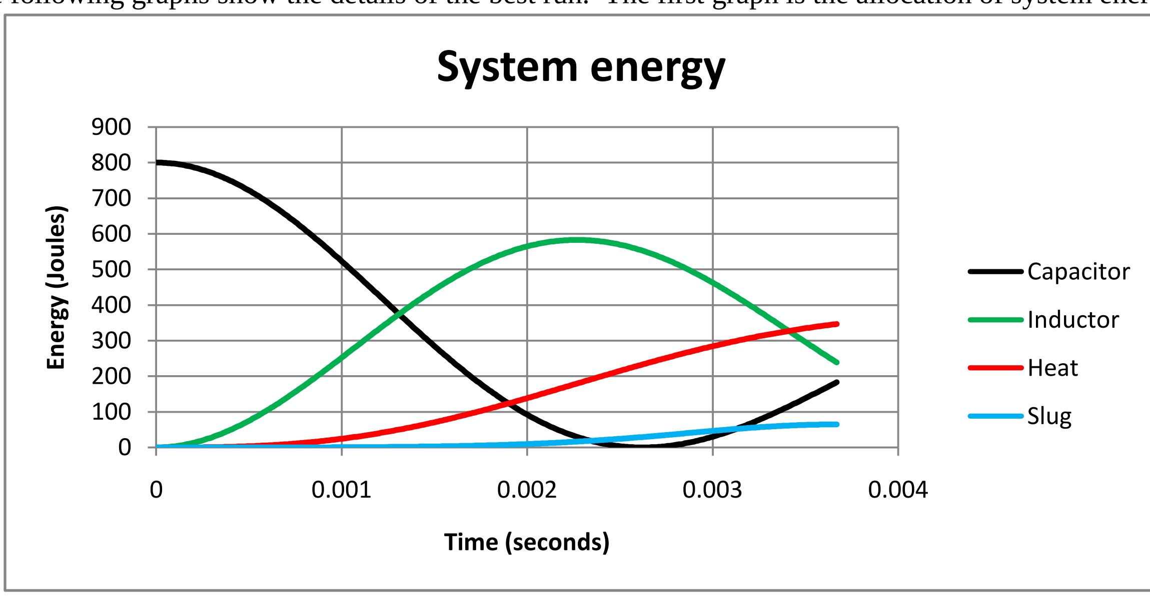
The horizontal axis on the left is the starting displacement of the slug. The horizontal axis on the right is the number of turns wound on the coil. The slug's kinetic energy is a maximum with 13 turns and a starting displacement of 26 cm. At this pair, the slug's kinetic energy (the vertical axis) is about 64.3 Joules.

I observe that one of reasons why the kinetic energy decreases with greater numbers of turns is that the slug passes through the force field before it can absorb all of the energy which is available. One way to assess this matter is shown in the following graph. The following graph shows the amount of energy which is still "recoverable" when the slug reaches the right-hand extent of the force field. This recoverable energy is the sum of the energy in the capacitor and the energy in the inductor at that instant.



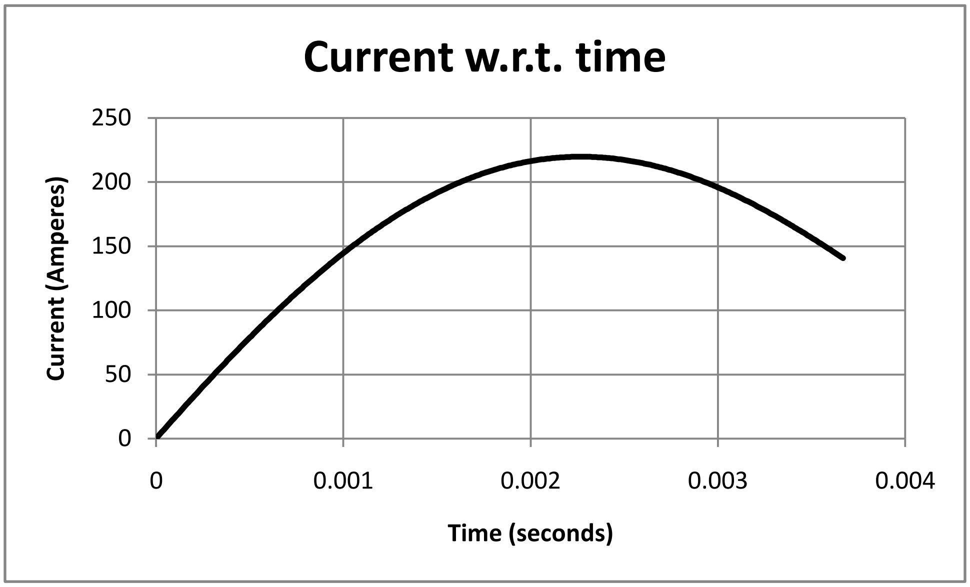
Even when the slug gets the maximum amount of kinetic energy (13 turns and 26 cm), there are still about 400 Joules of recoverable energy remaining. In order for the slug to benefit from this extra energy available, the horizontal extent of the force field would have to be wider, to give the slug more time / distance in which to absorb energy.

The following graphs show the details of the best run. The first graph is the allocation of system energy.



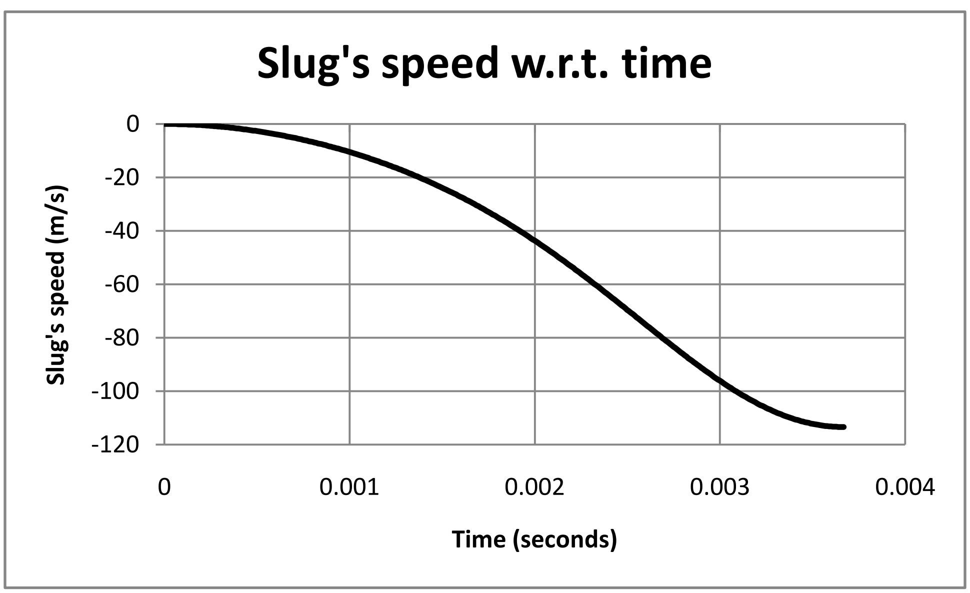
The capacitor is completely discharged about 2.7 ms after the start of the run. But, it then begins to charge up again with energy released by the inductor. The traces end when the slug leaves the force field, at which time the combined energy in the capacitor and inductor is about 400 Joules, or one-half of the initial amount. It can be seen that a significant amount of the slug's final kinetic energy was obtained after the time of the capacitor's discharge. This is, in fact, the explanation behind the dip in the surface in the third graph above. It relates to the "end effect" – how far from the end of the run the balance of energy between the capacitor and inductor changes.

The following graph shows the current during the best run.



The timing of the run is such that the slug experiences only a part of the first sinusoidal cycle. It seems to be the case that the slug's extracting energy from the system has the effect of increasing the period of the oscillation. Given what we concluded above, regarding the oscillation of current with no slug in the coil, we would have expected a half-dozen or so cycles with 13 turns on the coil.

The following graph shows the speed of the slug during the best run.



As I observed, the spatial extent of the force field is not big enough. The slug passes through the force field before it gains as much energy as it could. But, there is little point in increasing the horizontal extent of the force field – distance D – since that was simply an example.

In Part III of this paper, we will look more closely at the real spatial extent of the force field.

Jim Hawley
September 2012

An e-mail setting out errors and omissions would be appreciated.

Appendix "A" – Listing of Visual Basic program *Integration3*

```
Option Strict On
Option Explicit On

' Integration3 - One layer only.

Public Class Form1
    Inherits System.Windows.Forms.Form

    '/////////////////////////////////////////////////////////////////////////////////
    '/////////////////////////////////////////////////////////////////////////////////
    '//
    '//                                Variables
    '//
    '/////////////////////////////////////////////////////////////////////////////////
    '/////////////////////////////////////////////////////////////////////////////////

    ' Timing variables
    Public delT As Double = 0.0000001   ' The time step for the integration is 100 ns
    Public T As Double                  ' The current time in the integration
    Public Zfinished As Double = 0      ' End the integration when the slug gets here

    ' Initial conditions
    Public VC0 As Double = 4000         ' Initial voltage over the capacitor
    Public QC0 As Double = 0.4          ' Initial charge on the capacitor
    Public Z0 As Double = 0.27          ' The slug's starting position
    Public S0 As Double = 0             ' The slug's starting speed

    ' Physical parameters
    Public R As Double = 2.156          ' Resistance = 2.156 Ohms
    Public L As Double = 0.0000116      ' Inductance = 11.6uH
    Public C As Double = 0.0001         ' Capacitance = 100uF
    Public M As Double = 0.01           ' The slug weighs ten grams

    ' Voltages
    Public VCstart As Double            ' Voltage over the capacitor at start of step
    Public VCend As Double              ' Voltage over the capacitor at end of step
    Public VRstart As Double            ' Voltage over the resistor at start of step
    Public VRend As Double              ' Voltage over the resistor at end of step
    Public VLstart As Double            ' Voltage over the inductor at start of step
    Public VLend As Double              ' Voltage over the inductor at end of step
    Public VSstart As Double            ' Voltage over the slug at start of step
    Public VSend As Double              ' Voltage over the slug at end of step
    Public ErrVstart As Double          ' Error in the total starting voltages
    Public ErrVend As Double            ' Error in the total ending voltages

    ' Currents
    Public QCstart As Double            ' Charge on capacitor at start of step
    Public QCend As Double              ' Charge on capacitor at end of step
    Public Istart As Double             ' Current at start of step
    Public Iend As Double               ' Current at end of step
    Public d2Qdt2start As Double        ' d2Q/dt2 at start of step

    ' Energy stocks
    Public ECstart As Double            ' Energy in the capacitor at start of step
    Public ECend As Double              ' Energy in the capacitor at end of step
```

```
Public delEC As Double              ' Change in capacitor's energy during step
Public ERstart As Double            ' Cumulative heat at start of step
Public ERend As Double              ' Cumulative heat at end of step
Public delER As Double              ' Change in heat during step
Public ELstart As Double            ' Energy in the inductor at start of step
Public ELend As Double              ' Energy in the inductor at end of step
Public delEL As Double              ' Change in inductor's energy during step
Public ESstart As Double            ' Energy in the slug at start of step
Public ESend As Double              ' Energy in the slug at end of step
Public delES As Double              ' Change in slug's energy during step
Public ETstart As Double            ' Total system energy at start of step
Public ETend As Double              ' Total system energy at end of step
Public delET As Double              ' Change in total energy during step

' Variables for the slug
Public Zstart As Double             ' The slug's position at start of a step
Public Zend As Double               ' The slug's position at end of a step
Public Sstart As Double             ' The slug's speed at start of a step
Public Send As Double               ' The slug's speed at end of a step
Public Astart As Double             ' The slug's acceleration at start of step

' Variables for the force during a single step
' Variable names F_ refer to the spatial distribution factor only.
Public ZendEst As Double            ' Estimate of the slug's position at the end
Public F_start As Double            ' Force at the slug's starting position
Public F_endEst As Double           ' Force at the slug's estimated ending position
Public F_avg As Double              ' Average force during the step

' Variables for the spatial force field, expressed as points in a table
Public Ftable(1001, 2) As Double    ' (*,1) is the position; (*,2) is the magnitude
'                                   ' Force values are stored negative
Public FnumZ As Int32 = 1001        ' The number of horizontal points in the table

' Variables used as an example to generate a force field
Public Fmax As Double = -0.01       ' The force is 100 Newtons at 100 Amperes
Public ZmaxF As Double = 0.24       ' The force is a maximum 24 cm from the origin
Public D As Double = 0.1            ' The force extends 10 cm from the maximum

' Variables needed to save the results
' (Add COM file "Microsoft Excel 12.0 Object library" to the project's References.)
Public ExcelFileName As String = "C:\Integration3Results.xlsx"
Public objExcel As Microsoft.Office.Interop.Excel.Application
Public objExcelWB As Microsoft.Office.Interop.Excel.Workbook
Public objExcelWS As Microsoft.Office.Interop.Excel.Worksheet
Public RowNum As Int32              ' The current output row in the spreadsheet

' Output control
Public delTdisplay As Double = 0.0000001 ' Display results every 100 ns
Public Tlastdisplay As Double            ' The time of the last display event
Public delTsave As Double = 0.000002     ' Save results every 2 us
Public Tlastsave As Double               ' The time of the last save event

Public Sub New()
    InitializeComponent()
    With Me
        Name = ""
        Text = "Axial Coil Gun - Integration #3"
        FormBorderStyle = Windows.Forms.FormBorderStyle.FixedSingle
```

```
        Size = New Drawing.Size(800, 800)
        CenterToScreen()
        .Visible = True
        Controls.Add(labelDisplay)
        labelDisplay.BringToFront()
        PerformLayout()
        BringToFront()
    End With
    Initialization()
End Sub

Public labelDisplay As New Windows.Forms.Label With _
    {.Size = New Drawing.Size(250, 600), _
     .Location = New Drawing.Point(5, 5), _
     .Text = "", .TextAlign = ContentAlignment.MiddleLeft}

Public Sub Initialization()
    '////////////////////////////////////////////
    ' Build the table for the sample force field.
    Dim Fleft As Double  ' The left-most abscissa at which to calculate the force
    Dim Fright As Double ' The right-most abscissa at which to calculate the force
    Dim FdelZ As Double  ' The spacing of the abscissas in the table
    Dim ZF As Double     ' An abscissa at which to calculate the force
    Dim F As Double      ' The value of the force at an abscissa
    Fleft = ZmaxF + D
    Fright = ZmaxF - D
    FdelZ = (Fleft - Fright) / (FnumZ - 1)
    For I As Int32 = 1 To FnumZ Step 1
        ' Calculate the next abscissa in the table.
        ZF = Fleft - ((I - 1) * FdelZ)
        ' Calculate the magnitude of the force.
        F = Fmax * (1 - (((ZF - ZmaxF) / D) ^ 2))
        ' Save the abscissa and the magnitude in the table.
        Ftable(I, 1) = ZF
        Ftable(I, 2) = F
    Next I
    '/////////////////////
    ' Open the Excel file.
    Try
        objExcel = CType(CreateObject("Excel.Application"),  _
            Microsoft.Office.Interop.Excel.Application)
        objExcel.Visible = False
        objExcelWB = CType(objExcel.Workbooks.Open(ExcelFileName),  _
            Microsoft.Office.Interop.Excel.Workbook)
        objExcelWS = CType(objExcelWB.Sheets("Sheet1"),  _
            Microsoft.Office.Interop.Excel.Worksheet)
    Catch ex As Exception
        Cursor.Current = Cursors.Default
        MsgBox("Could not open the Excel file to save the output.", vbOKOnly)
        Exit Sub
    End Try
    ' Save a description of this configuration in the Excel file.
    With objExcelWS
        .Cells(1, 1) = "Integration #3"
        .Cells(2, 1) = "The force field is a parabola."
        .Cells(4, 1) = "Mass = " & Trim(Str(M)) & " kilograms"
        .Cells(5, 1) = "Fmax = " & Trim(Str(Fmax)) & " Newtons"
        .Cells(6, 1) = "Max at Z = " & Trim(Str(ZmaxF)) & " meters"
```

```
    .Cells(7, 1) = "D = " & Trim(Str(D)) & " meters"
    .Cells(8, 1) = "Num points = " & Trim(Str(FnumZ))
    .Cells(9, 1) = "Capacitance = " & Trim(Str(C)) & " Farads"
    .Cells(10, 1) = "Resistance = " & Trim(Str(R)) & " Ohms"
    .Cells(11, 1) = "Inductance = " & Trim(Str(L)) & " Henries"
    .Cells(12, 1) = "VC0 = " & Trim(Str(VC0)) & " Volts"
    .Cells(13, 1) = "Z0 = " & Trim(Str(Z0)) & " meters"
    .Cells(14, 1) = "Time step = " & Trim(Str(delT)) & " seconds"
End With
' Write the headers for the output.
With objExcelWS
    .Cells(16, 1) = "Ending"
    .Cells(17, 1) = "Time"
    .Cells(18, 1) = "(s)"
    .Cells(15, 2) = "Ending"
    .Cells(16, 2) = "Slug's"
    .Cells(17, 2) = "Position"
    .Cells(18, 2) = "(m)"
    .Cells(15, 3) = "Ending"
    .Cells(16, 3) = "Slug's"
    .Cells(17, 3) = "Speed"
    .Cells(18, 3) = "(m/s)"
    .Cells(15, 4) = "Starting"
    .Cells(16, 4) = "Slug's"
    .Cells(17, 4) = "Acceleration"
    .Cells(18, 4) = "(m/s^2)"
    .Cells(16, 5) = "Ending"
    .Cells(17, 5) = "Charge"
    .Cells(18, 5) = "(C)"
    .Cells(15, 6) = "Ending"
    .Cells(16, 6) = "Total"
    .Cells(17, 6) = "Current"
    .Cells(18, 6) = "(A)"
    .Cells(15, 7) = "Spatial"
    .Cells(16, 7) = "Force"
    .Cells(17, 7) = "Factor"
    .Cells(18, 7) = "(N/A^2)"
    .Cells(15, 8) = "Ending"
    .Cells(16, 8) = "Capacitor's"
    .Cells(17, 8) = "Energy"
    .Cells(18, 8) = "(J)"
    .Cells(15, 9) = "Ending"
    .Cells(16, 9) = "Inductor's"
    .Cells(17, 9) = "Energy"
    .Cells(18, 9) = "(J)"
    .Cells(15, 10) = "Ending"
    .Cells(16, 10) = "Cumulative"
    .Cells(17, 10) = "Heat"
    .Cells(18, 10) = "(J)"
    .Cells(15, 11) = "Ending"
    .Cells(16, 11) = "Slug's"
    .Cells(17, 11) = "Energy"
    .Cells(18, 11) = "(J)"
    .Cells(15, 12) = "Ending"
    .Cells(16, 12) = "Total"
    .Cells(17, 12) = "Energy"
    .Cells(18, 12) = "(J)"
    .Cells(15, 13) = "Starting"
```

```
        .Cells(16, 13) = "Capacitor's"
        .Cells(17, 13) = "Voltage"
        .Cells(18, 13) = "(C)"
        .Cells(15, 14) = "Starting"
        .Cells(16, 14) = "Inductor's"
        .Cells(17, 14) = "Voltage"
        .Cells(18, 14) = "(V)"
        .Cells(15, 15) = "Starting"
        .Cells(16, 15) = "Resistors's"
        .Cells(17, 15) = "Voltage"
        .Cells(18, 15) = "(V)"
        .Cells(15, 16) = "Starting"
        .Cells(16, 16) = "Slug's"
        .Cells(17, 16) = "Voltage"
        .Cells(18, 16) = "(V)"
        .Cells(15, 17) = "Starting"
        .Cells(16, 17) = "Voltage"
        .Cells(17, 17) = "Error"
        .Cells(18, 17) = "(V)"
    End With
    ' Set the starting row number for adding entries to the spreadsheet.
    RowNum = 20
    ' Begin the integration
    StartIntegrating()
End Sub

Public Sub StartIntegrating()
    ' Write the starting data to the file, for time t=0.
    RowNum = RowNum + 1
    With objExcelWS
        .Cells(RowNum, 1) = T           ' Time
        .Cells(RowNum, 2) = Z0          ' Ending slug's position
        .Cells(RowNum, 3) = S0          ' Ending slug's speed
        .Cells(RowNum, 4) = 0           ' Starting slug's acceleration
        .Cells(RowNum, 5) = QC0         ' Ending charge
        .Cells(RowNum, 6) = 0           ' Ending current
        .Cells(RowNum, 7) = ""          ' Spatial force factor
        .Cells(RowNum, 8) = 0.5 * C * VC0 * VC0 ' Ending capacitor's energy
        .Cells(RowNum, 9) = 0           ' Ending inductor's energy
        .Cells(RowNum, 10) = 0          ' Ending cumulative heat
        .Cells(RowNum, 11) = 0D         ' Ending slug's energy
        .Cells(RowNum, 12) = 0.5 * C * VC0 * VC0 ' Ending total energy
        .Cells(RowNum, 13) = VC0        ' Starting capacitor's voltage
        .Cells(RowNum, 14) = VC0        ' Starting inductor's voltage
        .Cells(RowNum, 15) = 0          ' Starting resistor's voltage
        .Cells(RowNum, 16) = 0          ' Starting slug's voltage
        .Cells(RowNum, 17) = 0          ' Starting voltage error
    End With
    ' Set things up for the first iteration.
    T = 0
    Tlastdisplay = -1000 ' Dummy previous time for displaying results
    Tlastsave = -1000    ' Dummy previous time for saving results
    ' For the slug:
    Zstart = Z0
    Sstart = S0
    ' For the energies:
    QCstart = QC0
    ECstart = 0.5 * C * VC0 * VC0
```

```
ERstart = 0
ELstart = 0
ESstart = 0
ETstart = ECstart + ERstart + ELstart + ESstart
' For the current:
Istart = 0
'////////////////////////////////
' Main loop for the integration.
Do
    ' Increment the time to the end of the next time step.
    T = T + delT
    ' Estimate the ending position of the slug - Equation (30).
    ZendEst = Zstart + (Sstart * delT)
    ' Find the spatial force factor at the beginning of the time step.
    F_start = InterpolateForceField(Zstart)
    ' Estimate the spatial force factor at the end of the time step.
    F_endEst = InterpolateForceField(ZendEst)
    ' Calculate the average force factor - Equation (31).
    F_avg = 0.5 * (F_start + F_endEst)
    ' Calculate the slug's acceleration at the start - Equation (28').
    Astart = Istart * Istart * F_avg / M
    ' Calculate the slug's speed at the end - Equation (29').
    Send = Sstart + (Astart * delT)
    ' Calculate the slug's position at the end - Equation (32).
    Zend = Zstart + (Sstart * delT) + (0.5 * Astart * delT * delT)
    ' Calculate the second derivative of the charge - Equation (27').
    d2Qdt2start = Istart * (R + (F_avg * Sstart))
    d2Qdt2start = d2Qdt2start - (QCstart / C)
    d2Qdt2start = d2Qdt2start / L
    ' Calculate the current at the end - Equation (33).
    Iend = Istart - (d2Qdt2start * delT)
    ' Calculate the charge at the end - Equation (34).
    QCend = QCstart - (Istart * delT) + (0.5 * d2Qdt2start * delT * delT)
    '//////////////////////
    '// Energy calculations
    ' Calculate the ending energy in the capacitor - Equation (35).
    ECend = 0.5 * QCend * QCend / C
    ' Calculate the heat dissipated during the time step - Equation (36).
    delER = (Istart * Istart) - (Istart * d2Qdt2start * delT) + _
        (d2Qdt2start * d2Qdt2start * delT * delT / 3)
    delER = R * delER * delT
    ' Calculate the cumulative heat at the end - Equation (37).
    ERend = ERstart + delER
    ' Calculate the ending energy in the inductor - Equation (38).
    ELend = 0.5 * L * Iend * Iend
    ' Calculate the ending energy of the slug - Equation (39).
    ESend = 0.5 * M * Send * Send
    ' Calculate the total ending energy.
    ETend = ECend + ERend + ELend + ESend
    '///////////////////////
    '// Voltage calculations
    ' Calculate the starting and ending voltages on the capacitor.
    VCstart = QCstart / C
    ' Calculate the starting and ending voltages over the resistor.
    VRstart = R * Istart
    ' Calculate the starting and ending voltages over the inductor.
    VLstart = -L * d2Qdt2start
    ' Calculate the starting and ending voltages over the slug.
```

```
VSstart = Istart * Sstart * F_start
' Calculate the starting and ending voltage errors.
ErrVstart = VCstart - (VRstart + VLstart + VSstart)
'////////////////////////////////////////////////////
'// If it is time, then save the data for this step.
If ((T - Tlastsave) >= delTsave) Then
    RowNum = RowNum + 1
    With objExcelWS
        .Cells(RowNum, 1) = T           ' Time
        .Cells(RowNum, 2) = Zend        ' Ending slug's position
        .Cells(RowNum, 3) = Send        ' Ending slug's speed
        .Cells(RowNum, 4) = Astart      ' Starting slug's acceleration
        .Cells(RowNum, 5) = QCend       ' Ending charge
        .Cells(RowNum, 6) = Iend        ' Ending current
        .Cells(RowNum, 7) = F_avg       ' Spatial force factor
        .Cells(RowNum, 8) = ECend       ' Ending capacitor's energy
        .Cells(RowNum, 9) = ELend       ' Ending inductor's energy
        .Cells(RowNum, 10) = ERend      ' Ending cumulative heat
        .Cells(RowNum, 11) = ESend      ' Ending slug's energy
        .Cells(RowNum, 12) = ETend      ' Ending total energy
        .Cells(RowNum, 13) = VCstart    ' Starting capacitor's voltage
        .Cells(RowNum, 14) = VLstart    ' Starting inductor's voltage
        .Cells(RowNum, 15) = VRstart    ' Starting resistor's voltage
        .Cells(RowNum, 16) = VSstart    ' Starting slug's voltage
        .Cells(RowNum, 17) = ErrVstart  ' Starting voltage error
    End With
    Tlastsave = T
End If
'//////////////////////////////////////////////////////
'// Check if we should stop integrating.  If so, then
'// save the data for the last step and then exit.
If (Zend <= Zfinished) Then
    RowNum = RowNum + 1
    With objExcelWS
        .Cells(RowNum, 1) = T           ' Time
        .Cells(RowNum, 2) = Zend        ' Ending slug's position
        .Cells(RowNum, 3) = Send        ' Ending slug's speed
        .Cells(RowNum, 4) = Astart      ' Starting slug's acceleration
        .Cells(RowNum, 5) = QCend       ' Ending charge
        .Cells(RowNum, 6) = Iend        ' Ending current
        .Cells(RowNum, 7) = F_avg       ' Spatial force factor
        .Cells(RowNum, 8) = ECend       ' Ending capacitor's energy
        .Cells(RowNum, 9) = ELend       ' Ending inductor's energy
        .Cells(RowNum, 10) = ERend      ' Ending cumulative heat
        .Cells(RowNum, 11) = ESend      ' Ending slug's energy
        .Cells(RowNum, 12) = ETend      ' Ending total energy
        .Cells(RowNum, 13) = VCstart    ' Starting capacitor's voltage
        .Cells(RowNum, 14) = VLstart    ' Starting inductor's voltage
        .Cells(RowNum, 15) = VRstart    ' Starting resistor's voltage
        .Cells(RowNum, 16) = VSstart    ' Starting slug's voltage
        .Cells(RowNum, 17) = ErrVstart  ' Starting voltage error
    End With
    Exit Do
End If
'////////////////////////////////////////////////////////////
'// If it is time, then display interim results to the user.
If ((T - Tlastdisplay) >= delTdisplay) Then
    labelDisplay.Text = _
```

```
                "Time = " & Str(T) & vbCrLf & _
                "ZendEst = " & Str(ZendEst) & vbCrLf & _
                "F_avg = " & Str(F_avg) & vbCrLf & _
                "Astart = " & Str(Astart) & vbCrLf & _
                "Send = " & Str(Send) & vbCrLf & _
                "Zend = " & Str(Zend) & vbCrLf & _
                "d2Qdt2start = " & Str(d2Qdt2start) & vbCrLf & _
                "Istart = " & Str(Istart) & vbCrLf & _
                "QCstart = " & Str(QCstart) & vbCrLf & _
                "Iend = " & Str(Iend) & vbCrLf & vbCrLf & _
                "ECend = " & Str(ECend) & vbCrLf & _
                "ERend = " & Str(ERend) & vbCrLf & _
                "ELend = " & Str(ELend) & vbCrLf & _
                "ESend = " & Str(ESend) & vbCrLf & _
                "ETend = " & Str(ETend) & vbCrLf & vbCrLf & _
                "VCstart = " & Str(VCstart) & vbCrLf & _
                "VRstart = " & Str(VRstart) & vbCrLf & _
                "VLstart = " & Str(VLstart) & vbCrLf & _
                "VSstart = " & Str(VSstart) & vbCrLf & _
                "ErrVstart = " & Str(ErrVstart)
            labelDisplay.Refresh()
            Tlastdisplay = T
        End If
        '//////////////////////////////////
        '// Update for the next iteration.
        Zstart = Zend
        Sstart = Send
        Istart = Iend
        QCstart = QCend
        ECstart = ECend
        ELstart = ELend
        ERstart = ERend
        ESstart = ESend
        ETstart = ETend
    Loop
    ' Save and close the output file.
    objExcelWB.Save()
    objExcelWB.Close()
    objExcel.Quit()
    ' Alert the user to completion.
    MsgBox("All finished.")
End Sub

'////////////////////////////////////////////////////////////////////////////////
'////////////////////////////////////////////////////////////////////////////////
'//
'//                      Interpolation in the force field table
'//
'////////////////////////////////////////////////////////////////////////////////
'////////////////////////////////////////////////////////////////////////////////

Public Function InterpolateForceField(ByVal Z As Double) As Double
    ' This function returns the interpolated magnitude of the force field at
    ' for the abscissa passed as the argument.  The returned value will be
    ' negative unless the given abscissa lies outside the force field, in
    ' which case the function returns zero.
    Dim IndexToLeft As Int32
    Dim Slope As Double
```

```
        ' Check if the slug is positioned outside the force field, to the left.
        If (Z > Ftable(1, 1)) Then
            InterpolateForceField = 0
            Exit Function
        End If
        ' Check if the slug is positioned outside the force field, to the right.
        If (Z < Ftable(FnumZ, 1)) Then
            InterpolateForceField = 0
            Exit Function
        End If
        ' Find the index of the abscissa in the force field table which lies just
        ' to the left of the given abscissa.  Note that this routine does not
        ' assume that the points in the table are equally-spaced.  It assumes only
        ' that they are in order of decreasing values, that is, each point is
        ' closer to the origin than the preceding point.
        For I As Int32 = 1 To (FnumZ - 1) Step 1
            If ((Z <= Ftable(I, 1)) And (Z > Ftable(I + 1, 1))) Then
                IndexToLeft = I
                Exit For
            End If
        Next I
        ' Let us check for the trivial case, where we are exactly at a data point.
        If (Z = Ftable(IndexToLeft, 1)) Then
            ' By chance, we are exactly at a data point.
            InterpolateForceField = Ftable(IndexToLeft, 2)
            Exit Function
        End If
        ' We deal here with the general case, where we are between two data points.
        Try
            Slope = (Ftable(IndexToLeft + 1, 2) - Ftable(IndexToLeft, 2)) / _
                (Ftable(IndexToLeft + 1, 1) - Ftable(IndexToLeft, 1))
        Catch ex As Exception
            MsgBox("Error in the force field table.  Two abscissas are the same.")
            InterpolateForceField = 0
            Exit Function
        End Try
        InterpolateForceField = Ftable(IndexToLeft, 2) + _
            (Slope * (Z - Ftable(IndexToLeft, 1)))
    End Function

End Class
```

Appendix "B" – Listing of Visual Basic program *Integation4*

```
Option Strict On
Option Explicit On

' Integration4 - Multiple layers.

Public Class Form1
    Inherits System.Windows.Forms.Form

    '/////////////////////////////////////////////////////////////////////////////////
    '/////////////////////////////////////////////////////////////////////////////////
    '//
    '//                                Variables
    '//
    '/////////////////////////////////////////////////////////////////////////////////
    '/////////////////////////////////////////////////////////////////////////////////

    ' Timing variables
    Public delT As Double = 0.0000001   ' The time step for the integration is 100 ns
    Public T As Double                  ' The current time in the integration
    Public Zfinished As Double = 0.14   ' End the integration when the slug gets here

    ' Independent variables
    Public Nlayers As Int32             ' Number of layers of wire in the winding
    Public Z0 As Double                 ' Starting position of the slug

    ' Initial conditions
    Public VC0 As Double = 4000         ' Initial voltage over the capacitor
    Public QC0 As Double = 0.4          ' Initial charge on the capacitor
    Public S0 As Double = 0             ' The slug's starting speed

    ' Physical parameters of a coil
    Public Rcore As Double = 0.01       ' The core is 2 cm in diameter
    Public Hcoil As Double = 0.4981     ' The coil is 49.81 cm long
    Public Dwire As Double = 0.005189   ' The wire has a diameter of 5.189 mm
    Public NTurns As Int32 = 96         ' Turns in each layer
    Public LenWire As Double            ' The length of wire used in the winding
    Public RPerM As Double = 0.0008153  ' Resistance of the wire in Ohms per meter
    Public RADavg As Double             ' Average radius of a turn in the winding
    Public AREAavg As Double            ' Average cross-sectional area of the winding
    Public Rcoil As Double              ' Resistance of the coil, in Ohms
    Public Lcoil As Double              ' Inductance of the coil, in Henries

    ' Characteristics of other components
    Public Resr As Double = 2           ' Equivalent series resiatcne of the capacitor
    Public Rsw As Double = 0.15         ' Static resistance of the switch
    Public C As Double = 0.0001         ' Capacitance = 100uF
    Public R As Double                  ' Total series resistance in the circuit
    Public L As Double                  ' Total series inductance in the circuit
    Public M As Double = 0.01           ' The slug weighs ten grams

    ' Voltages
    Public VCstart As Double            ' Voltage over the capacitor at start of step
    Public VCend As Double              ' Voltage over the capacitor at end of step
    Public VRstart As Double            ' Voltage over the resistor at start of step
    Public VRend As Double              ' Voltage over the resistor at end of step
```

```
Public VLstart As Double            ' Voltage over the inductor at start of step
Public VLend As Double              ' Voltage over the inductor at end of step
Public VSstart As Double            ' Voltage over the slug at start of step
Public VSend As Double              ' Voltage over the slug at end of step
Public ErrVstart As Double          ' Error in the total starting voltages
Public ErrVend As Double            ' Error in the total ending voltages

' Currents
Public QCstart As Double            ' Charge on capacitor at start of step
Public QCend As Double              ' Charge on capacitor at end of step
Public Istart As Double             ' Current at start of step
Public Iend As Double               ' Current at end of step
Public d2Qdt2start As Double        ' d2Q/dt2 at start of step

' Energy stocks
Public ECstart As Double            ' Energy in the capacitor at start of step
Public ECend As Double              ' Energy in the capacitor at end of step
Public delEC As Double              ' Change in capacitor's energy during step
Public ERstart As Double            ' Cumulative heat at start of step
Public ERend As Double              ' Cumulative heat at end of step
Public delER As Double              ' Change in heat during step
Public ELstart As Double            ' Energy in the inductor at start of step
Public ELend As Double              ' Energy in the inductor at end of step
Public delEL As Double              ' Change in inductor's energy during step
Public ESstart As Double            ' Energy in the slug at start of step
Public ESend As Double              ' Energy in the slug at end of step
Public delES As Double              ' Change in slug's energy during step
Public ETstart As Double            ' Total system energy at start of step
Public ETend As Double              ' Total system energy at end of step
Public delET As Double              ' Change in total energy during step

' Variables for the slug
Public Zstart As Double             ' The slug's position at start of a step
Public Zend As Double               ' The slug's position at end of a step
Public Sstart As Double             ' The slug's speed at start of a step
Public Send As Double               ' The slug's speed at end of a step
Public Astart As Double             ' The slug's acceleration at start of step

' Variables for the force during a single step
' Variable names F_ refer to the spatial distribution factor only
Public ZendEst As Double            ' Estimate of the slug's position at the end
Public F_start As Double            ' Force at the slug's starting position
Public F_endEst As Double           ' Force at the slug's estimated ending position
Public F_avg As Double              ' Average force during the step

' Variables for the spatial force field, expressed as points in a table
Public Ftable(1001, 2) As Double    ' (*,1) is the position; (*,2) is the magnitude
'                                   ' Force values are stored negative
Public FnumZ As Int32 = 1001        ' The number of horizontal points in the table

' Variables used as an example to generate a force field
Public Fmax As Double = -0.01       ' The force is 100 Newtons at 100 Amperes
Public ZmaxF As Double = 0.24       ' The force is a maximum 24 cm from the origin
Public D As Double = 0.1            ' The force extends 10 cm from the maximum

' Variables needed to save the results
' (Add COM file "Microsoft Excel 12.0 Object library" to the project's References.)
Public ExcelFileName As String = "C:\Integration4Results.xlsx"
```

```
Public objExcel As Microsoft.Office.Interop.Excel.Application
Public objExcelWB As Microsoft.Office.Interop.Excel.Workbook
Public objExcelWS As Microsoft.Office.Interop.Excel.Worksheet
Public RowNum As Int32              ' The current output row in the spreadsheet

' Output control
Public delTdisplay As Double = 0.00001   ' Display results every 10 us
Public Tlastdisplay As Double            ' The time of the last display event
Public delTsave As Double = 0.000002     ' Save results every 2 us
Public Tlastsave As Double               ' The time of the last save event

Public Sub New()
    InitializeComponent()
    With Me
        Name = ""
        Text = "Axial Coil Gun - Integration #4"
        FormBorderStyle = Windows.Forms.FormBorderStyle.FixedSingle
        Size = New Drawing.Size(800, 800)
        CenterToScreen()
        .Visible = True
        Controls.Add(labelDisplay)
        labelDisplay.BringToFront()
        PerformLayout()
        BringToFront()
    End With
    Initialization()
End Sub

Public labelDisplay As New Windows.Forms.Label With _
    {.Size = New Drawing.Size(250, 600), _
     .Location = New Drawing.Point(5, 5), _
     .Text = "", .TextAlign = ContentAlignment.MiddleLeft}

Public Sub Initialization()
    '//////////////////////////////////////////
    ' Build the table for the sample force field.
    Dim Fleft As Double  ' The left-most abscissa at which to calculate the force
    Dim Fright As Double ' The right-most abscissa at which to calculate the force
    Dim FdelZ As Double  ' The spacing of the abscissas in the table
    Dim ZF As Double     ' An abscissa at which to calculate the force
    Dim F As Double      ' The value of the force at an abscissa
    Fleft = ZmaxF + D
    Fright = ZmaxF - D
    FdelZ = (Fleft - Fright) / (FnumZ - 1)
    For I As Int32 = 1 To FnumZ Step 1
        ' Calculate the next abscissa in the table.
        ZF = Fleft - ((I - 1) * FdelZ)
        ' Calculate the magnitude of the force.
        F = Fmax * (1 - (((ZF - ZmaxF) / D) ^ 2))
        ' Save the abscissa and the magnitude in the table.
        Ftable(I, 1) = ZF
        Ftable(I, 2) = F
    Next I
    '///////////////////
    ' Open the Excel file.
    Try
        objExcel = CType(CreateObject("Excel.Application"),  _
            Microsoft.Office.Interop.Excel.Application)
```

```
        objExcel.Visible = False
        objExcelWB = CType(objExcel.Workbooks.Open(ExcelFileName),  _
            Microsoft.Office.Interop.Excel.Workbook)
        objExcelWS = CType(objExcelWB.Sheets("Sheet1"),  _
            Microsoft.Office.Interop.Excel.Worksheet)
    Catch ex As Exception
        Cursor.Current = Cursors.Default
        MsgBox("Could not open the Excel file to save the output.", vbOKOnly)
        Exit Sub
    End Try
    ' Save a description of this configuration in the Excel file.
    With objExcelWS
        .Cells(1, 1) = "Integration #4"
        .Cells(2, 1) = "The force field is a parabola."
        .Cells(4, 1) = "Mass = " & Trim(Str(M)) & " kilograms"
        .Cells(5, 1) = "Fmax = " & Trim(Str(Fmax)) & " Newtons"
        .Cells(6, 1) = "Max at Z = " & Trim(Str(ZmaxF)) & " meters"
        .Cells(7, 1) = "D = " & Trim(Str(D)) & " meters"
        .Cells(8, 1) = "Num points = " & Trim(Str(FnumZ))
        .Cells(9, 1) = "Rcore =" & Trim(Str(Rcore)) & " meters"
        .Cells(10, 1) = "Hcoil = " & Trim(Str(Hcoil)) & " meters"
        .Cells(11, 1) = "Dwire = " & Trim(Str(Dwire)) & " meters"
        .Cells(12, 1) = "RPerM = " & Trim(Str(RPerM)) & " Ohms/m"
        .Cells(13, 1) = "Resr = " & Trim(Str(Resr)) & " Ohms"
        .Cells(14, 1) = "Rsw = " & Trim(Str(Rsw)) & " Ohms"
        .Cells(15, 1) = "C = " & Trim(Str(C)) & " Farads"
        .Cells(16, 1) = "VC0 = " & Trim(Str(VC0)) & " Volts"
        .Cells(17, 1) = "QC0 = " & Trim(Str(QC0)) & " Coulombs"
        .Cells(18, 1) = "Time step = " & Trim(Str(delT)) & " seconds"
    End With
    ' Write the headers for the output.
    With objExcelWS
        .Cells(20, 2) = "Nlayers ="
        .Cells(21, 2) = "R ="
        .Cells(22, 2) = "L ="
        For I As Int32 = 1 To 10 Step 1
            .Cells(20, 2 + I) = Trim(Str(I))
            BuildTheCoil(I)
            .Cells(21, 2 + I) = FormatNumber(Rcoil, 4)
            .Cells(22, 2 + I) = FormatNumber(Lcoil, 7)
        Next I
        ' Header for table with slug's final energy.
        .Cells(24, 1) = "Z0 ="
        ' Header for table with final recoverable energy.
        .Cells(24 + 25, 1) = "Z0 ="
    End With
    ' Set the starting row number for adding entries to the spreadsheet.
    RowNum = 25
    ' Begin the integration
    StartIntegrating()
End Sub

Public Sub StartIntegrating()
    '//////////////////////////////////////////////////////////
    '// Main loop to step through the slug's starting position.
    '// First iteration has Z0 = 0.335 meters, which is 0.5 cm
    '// to the right of the left-most extent of the force field.
    '// Last iteration has Z0 = 0.22 meters, which is 2 cm to
```

```
    '// the right of the force peak.  Iterations are separated
    '// by 0.5 cm.
    For I As Int32 = 1 To 24 Step 1
        Z0 = (ZmaxF + D) - (I * 0.005)
        RowNum = RowNum + 1
        With objExcelWS
            .Cells(RowNum, 1) = FormatNumber(Z0, 4)
            .Cells(RowNum + 25, 1) = FormatNumber(Z0, 4)
        End With
        '/////////////////////////////////////////////////////////
        '// Main loop to step through the number of layers in the coil.
        For J As Int32 = 1 To 10 Step 1
            Nlayers = J
            ' Calculate the resistance and inductance.
            BuildTheCoil(Nlayers)
            ' Calculate the circuit resistance and inductance.
            R = Resr + Rsw + Rcoil
            L = Lcoil
            ' Execute the run.
            IntegrateOneRun(Nlayers, Z0, False, 0)
            ' Save the results.
            With objExcelWS
                ' The slug's kinetic energy is saved in the first table.
                .Cells(RowNum, 2 + J) = FormatNumber(ESend, 2)
                ' The recoverable energy is saved in the second table.
                .Cells(RowNum + 25, 2 + J) = FormatNumber(ECend + ELend, 3)
            End With
        Next J
    Next I
    '///////////////////////////////////////////////
    '// Display the details of the run with ten layers.
    Nlayers = 10
    Z0 = 0.31
    BuildTheCoil(Nlayers)
    R = Resr + Rsw + Rcoil
    L = Lcoil
    IntegrateOneRun(Nlayers, Z0, True, 100)
    ' Save and close the output file.
    objExcelWB.Save()
    objExcelWB.Close()
    objExcel.Quit()
    ' Alert the user to completion.
    MsgBox("All finished.")
End Sub

'//////////////////////////////////////////////////////////////////////////////
'//////////////////////////////////////////////////////////////////////////////
'//
'//                                Integrate one run
'//
'//////////////////////////////////////////////////////////////////////////////
'//////////////////////////////////////////////////////////////////////////////

Public Sub IntegrateOneRun( _
    ByVal Nlayers As Int32, ByVal Z0 As Double, _
    ByVal SaveResults As Boolean, ByVal RowNum As Int32)
    ' The two principal independent variables are passed as.  All other
    ' variables are global variables.  If SaveResults is True, then the
```

```
' results are saved to the Excel file.  Argument RowNum specifies
' the row in the spreadsheet at which these interim results are
' to be save.
' If SaveResults is False, then the integration is ended when the
' slug's position at the end of a time step is less than ZmaxF - D,
' that is, when the slug has passed through the right edge of the
' force field.
' If SaveResults is True, then the integration is ended when the
' slug's position at the end of a time step is less than -Hcoil/2,
' that is, when the slug leaves the coil entirely.  This allows the
' output to show what happens after the slug is gone.
'////////////////////////////////////////////////
'// Write the header for the output, if requested.
If (SaveResults = True) Then
    With objExcelWS
        .Cells(RowNum + 1, 1) = "Ending"
        .Cells(RowNum + 2, 1) = "Time"
        .Cells(RowNum + 3, 1) = "(s)"
        .Cells(RowNum, 2) = "Ending"
        .Cells(RowNum + 1, 2) = "Slug's"
        .Cells(RowNum + 2, 2) = "Position"
        .Cells(RowNum + 3, 2) = "(m)"
        .Cells(RowNum, 3) = "Ending"
        .Cells(RowNum + 1, 3) = "Slug's"
        .Cells(RowNum + 2, 3) = "Speed"
        .Cells(RowNum + 3, 3) = "(m/s)"
        .Cells(RowNum, 4) = "Starting"
        .Cells(RowNum + 1, 4) = "Slug's"
        .Cells(RowNum + 2, 4) = "Acceleration"
        .Cells(RowNum + 3, 4) = "(m/s^2)"
        .Cells(RowNum + 1, 5) = "Ending"
        .Cells(RowNum + 2, 5) = "Charge"
        .Cells(RowNum + 3, 5) = "(C)"
        .Cells(RowNum, 6) = "Ending"
        .Cells(RowNum + 1, 6) = "Total"
        .Cells(RowNum + 2, 6) = "Current"
        .Cells(RowNum + 3, 6) = "(A)"
        .Cells(RowNum, 7) = "Spatial"
        .Cells(RowNum + 1, 7) = "Force"
        .Cells(RowNum + 2, 7) = "Factor"
        .Cells(RowNum + 3, 7) = "(N/A^2)"
        .Cells(RowNum, 8) = "Ending"
        .Cells(RowNum + 1, 8) = "Capacitor's"
        .Cells(RowNum + 2, 8) = "Energy"
        .Cells(RowNum + 3, 8) = "(J)"
        .Cells(RowNum, 9) = "Ending"
        .Cells(RowNum + 1, 9) = "Inductor's"
        .Cells(RowNum + 2, 9) = "Energy"
        .Cells(RowNum + 3, 9) = "(J)"
        .Cells(RowNum, 10) = "Ending"
        .Cells(RowNum + 1, 10) = "Cumulative"
        .Cells(RowNum + 2, 10) = "Heat"
        .Cells(RowNum + 3, 10) = "(J)"
        .Cells(RowNum, 11) = "Ending"
        .Cells(RowNum + 1, 11) = "Slug's"
        .Cells(RowNum + 2, 11) = "Energy"
        .Cells(RowNum + 3, 11) = "(J)"
        .Cells(RowNum, 12) = "Ending"
```

```
        .Cells(RowNum + 1, 12) = "Total"
        .Cells(RowNum + 2, 12) = "Energy"
        .Cells(RowNum + 3, 12) = "(J)"
        .Cells(RowNum, 13) = "Starting"
        .Cells(RowNum + 1, 13) = "Capacitor's"
        .Cells(RowNum + 2, 13) = "Voltage"
        .Cells(RowNum + 3, 13) = "(C)"
        .Cells(RowNum, 14) = "Starting"
        .Cells(RowNum + 1, 14) = "Inductor's"
        .Cells(RowNum + 2, 14) = "Voltage"
        .Cells(RowNum + 3, 14) = "(V)"
        .Cells(RowNum, 15) = "Starting"
        .Cells(RowNum + 1, 15) = "Resistors's"
        .Cells(RowNum + 2, 15) = "Voltage"
        .Cells(RowNum + 3, 15) = "(V)"
        .Cells(RowNum, 16) = "Starting"
        .Cells(RowNum + 1, 16) = "Slug's"
        .Cells(RowNum + 2, 16) = "Voltage"
        .Cells(RowNum + 3, 16) = "(V)"
        .Cells(RowNum, 17) = "Starting"
        .Cells(RowNum + 1, 17) = "Voltage"
        .Cells(RowNum + 2, 17) = "Error"
        .Cells(RowNum + 3, 17) = "(V)"
    End With
End If
'//////////////////////////////////////////////////////
'// Write the starting data to the file, if requested.
If (SaveResults = True) Then
    RowNum = RowNum + 4
    With objExcelWS
        .Cells(RowNum, 1) = T           ' Time
        .Cells(RowNum, 2) = Z0          ' Ending slug's position
        .Cells(RowNum, 3) = S0          ' Ending slug's speed
        .Cells(RowNum, 4) = 0           ' Starting slug's acceleration
        .Cells(RowNum, 5) = QC0         ' Ending charge
        .Cells(RowNum, 6) = 0           ' Ending current
        .Cells(RowNum, 7) = ""          ' Spatial force factor
        .Cells(RowNum, 8) = 0.5 * C * VC0 * VC0 ' Ending capacitor's energy
        .Cells(RowNum, 9) = 0           ' Ending inductor's energy
        .Cells(RowNum, 10) = 0          ' Ending cumulative heat
        .Cells(RowNum, 11) = 0          ' Ending slug's energy
        .Cells(RowNum, 12) = 0.5 * C * VC0 * VC0 ' Ending total energy
        .Cells(RowNum, 13) = VC0        ' Starting capacitor's voltage
        .Cells(RowNum, 14) = VC0        ' Starting inductor's voltage
        .Cells(RowNum, 15) = 0          ' Starting resistor's voltage
        .Cells(RowNum, 16) = 0          ' Starting slug's voltage
        .Cells(RowNum, 17) = 0          ' Starting voltage error
    End With
End If
'//////////////////////////////////////////
'// Set things up for the first iteration.
T = 0
Tlastdisplay = -1000 ' Dummy previous time for displaying results
Tlastsave = -1000    ' Dummy previous time for saving results
' For the slug:
Zstart = Z0
Sstart = S0
' For the energies:
```

```
QCstart = QC0
ECstart = 0.5 * C * VC0 * VC0
ERstart = 0
ELstart = 0
ESstart = 0
ETstart = ECstart + ERstart + ELstart + ESstart
' For the current:
Istart = 0
'/////////////////////////////////
'// Main loop for the integration.
Do
    ' Increment the time to the end of the next time step.
    T = T + delT
    ' Estimate the ending position of the slug - Equation (30).
    ZendEst = Zstart + (Sstart * delT)
    ' Find the spatial force factor at the beginning of the time step.
    F_start = InterpolateForceField(Zstart)
    ' Estimate the spatial force factor at the end of the time step.
    F_endEst = InterpolateForceField(ZendEst)
    ' Calculate the average force factor - Equation (31).
    F_avg = 0.5 * (F_start + F_endEst)
    ' Calculate the slug's acceleration at the start - Equation (28'Nlayers).
    Astart = Nlayers * Nlayers * Istart * Istart * F_avg / M
    ' Calculate the slug's speed at the end - Equation (29').
    Send = Sstart + (Astart * delT)
    ' Calculate the slug's position at the end - Equation (32).
    Zend = Zstart + (Sstart * delT) + (0.5 * Astart * delT * delT)
    ' Calculate the second derivative of the charge - Equation (27'Nlayers).
    ' Note the slug always consumes positive power.
    d2Qdt2start = (Istart * R) + (Istart * (Nlayers * Nlayers * F_avg * Sstart))
    d2Qdt2start = d2Qdt2start - (QCstart / C)
    d2Qdt2start = d2Qdt2start / L
    ' Calculate the current at the end - Equation (33).
    Iend = Istart - (d2Qdt2start * delT)
    ' Calculate the charge at the end - Equation (34).
    QCend = QCstart - (Istart * delT) + (0.5 * d2Qdt2start * delT * delT)
    '//////////////////////
    '// Energy calculations
    ' Calculate the ending energy in the capacitor - Equation (35).
    ECend = 0.5 * QCend * QCend / C
    ' Calculate the heat dissipated during the time step - Equation (36).
    delER = (Istart * Istart) - (Istart * d2Qdt2start * delT) + _
        (d2Qdt2start * d2Qdt2start * delT * delT / 3)
    delER = R * delER * delT
    ' Calculate the cumulative heat at the end - Equation (37).
    ERend = ERstart + delER
    ' Calculate the ending energy in the inductor - Equation (38).
    ELend = 0.5 * L * Iend * Iend
    ' Calculate the ending energy of the slug - Equation (39).
    ESend = 0.5 * M * Send * Send
    ' Calculate the total ending energy.
    ETend = ECend + ERend + ELend + ESend
    '///////////////////////
    '// Voltage calculations
    ' Calculate the starting and ending voltages on the capacitor.
    VCstart = QCstart / C
    ' Calculate the starting and ending voltages over the resistor.
    VRstart = R * Istart
```

```
            ' Calculate the starting and ending voltages over the inductor.
            VLstart = -L * d2Qdt2start
            ' Calculate the starting and ending voltages over the slug - Equation
(23Nlayers).
            VSstart = Nlayers * Nlayers * Istart * Sstart * F_start
            ' Calculate the starting and ending voltage errors.
            '///////////////////////////////////////////////////
            '// If it is time, then save the data for this step.
            If (SaveResults = True) Then
                If ((T - Tlastsave) >= delTsave) Then
                    RowNum = RowNum + 1
                    With objExcelWS
                        .Cells(RowNum, 1) = T             ' Time
                        .Cells(RowNum, 2) = Zend          ' Ending slug's position
                        .Cells(RowNum, 3) = Send          ' Ending slug's speed
                        .Cells(RowNum, 4) = Astart        ' Starting slug's acceleration
                        .Cells(RowNum, 5) = QCend         ' Ending charge
                        .Cells(RowNum, 6) = Iend          ' Ending current
                        .Cells(RowNum, 7) = F_avg         ' Spatial force factor
                        .Cells(RowNum, 8) = ECend         ' Ending capacitor's energy
                        .Cells(RowNum, 9) = ELend         ' Ending inductor's energy
                        .Cells(RowNum, 10) = ERend        ' Ending cumulative heat
                        .Cells(RowNum, 11) = ESend        ' Ending slug's energy
                        .Cells(RowNum, 12) = ETend        ' Ending total energy
                        .Cells(RowNum, 13) = VCstart      ' Starting capacitor's voltage
                        .Cells(RowNum, 14) = VLstart      ' Starting inductor's voltage
                        .Cells(RowNum, 15) = VRstart      ' Starting resistor's voltage
                        .Cells(RowNum, 16) = VSstart      ' Starting slug's voltage
                        .Cells(RowNum, 17) = ErrVstart    ' Starting voltage error
                    End With
                    Tlastsave = T
                End If
            End If
            '////////////////////////////////////////////////////
            '// Check if we should stop integrating.
            '// Case A: Not saving results.
            If (SaveResults = False) Then
                If (Zend <= Zfinished) Then
                    Exit Do
                End If
            Else
                '// Case B: Interim results are being saved.
                If (Zend <= (-0.5 * Hcoil)) Then
                    RowNum = RowNum + 1
                    With objExcelWS
                        .Cells(RowNum, 1) = T             ' Time
                        .Cells(RowNum, 2) = Zend          ' Ending slug's position
                        .Cells(RowNum, 3) = Send          ' Ending slug's speed
                        .Cells(RowNum, 4) = Astart        ' Starting slug's acceleration
                        .Cells(RowNum, 5) = QCend         ' Ending charge
                        .Cells(RowNum, 6) = Iend          ' Ending current
                        .Cells(RowNum, 7) = F_avg         ' Spatial force factor
                        .Cells(RowNum, 8) = ECend         ' Ending capacitor's energy
                        .Cells(RowNum, 9) = ELend         ' Ending inductor's energy
                        .Cells(RowNum, 10) = ERend        ' Ending cumulative heat
                        .Cells(RowNum, 11) = ESend        ' Ending slug's energy
                        .Cells(RowNum, 12) = ETend        ' Ending total energy
                        .Cells(RowNum, 13) = VCstart      ' Starting capacitor's voltage
```

```
                .Cells(RowNum, 14) = VLstart    ' Starting inductor's voltage
                .Cells(RowNum, 15) = VRstart    ' Starting resistor's voltage
                .Cells(RowNum, 16) = VSstart    ' Starting slug's voltage
                .Cells(RowNum, 17) = ErrVstart  ' Starting voltage error
            End With
            Exit Do
        End If
    End If
    '////////////////////////////////////////////////////////////
    '// If it is time, then display interim results to the user.
    If ((T - Tlastdisplay) >= delTdisplay) Then
        labelDisplay.Text = _
            "Numbers of layers = " & Str(Nlayers) & vbCrLf & _
            "Starting position = " & Str(Z0) & vbCrLf & _
            "Time = " & Str(T) & vbCrLf & _
            "ZendEst = " & Str(ZendEst) & vbCrLf & _
            "F_avg = " & Str(F_avg) & vbCrLf & _
            "Astart = " & Str(Astart) & vbCrLf & _
            "Send = " & Str(Send) & vbCrLf & _
            "Zend = " & Str(Zend) & vbCrLf & _
            "d2Qdt2start = " & Str(d2Qdt2start) & vbCrLf & _
            "Istart = " & Str(Istart) & vbCrLf & _
            "QCstart = " & Str(QCstart) & vbCrLf & _
            "Iend = " & Str(Iend) & vbCrLf & vbCrLf & _
            "ECend = " & Str(ECend) & vbCrLf & _
            "ERend = " & Str(ERend) & vbCrLf & _
            "ELend = " & Str(ELend) & vbCrLf & _
            "ESend = " & Str(ESend) & vbCrLf & _
            "ETend = " & Str(ETend) & vbCrLf & vbCrLf & _
            "VCstart = " & Str(VCstart) & vbCrLf & _
            "VRstart = " & Str(VRstart) & vbCrLf & _
            "VLstart = " & Str(VLstart) & vbCrLf & _
            "VSstart = " & Str(VSstart) & vbCrLf & _
            "ErrVstart = " & Str(ErrVstart)
        labelDisplay.Refresh()
        Tlastdisplay = T
    End If
    '//////////////////////////////////
    '// Update for the next iteration.
    Zstart = Zend
    Sstart = Send
    Istart = Iend
    QCstart = QCend
    ECstart = ECend
    ELstart = ELend
    ERstart = ERend
    ESstart = ESend
    ETstart = ETend
Loop
End Sub

'////////////////////////////////////////////////////////////////////////////////////
'////////////////////////////////////////////////////////////////////////////////////
'//
'//                                 Build the coil
'//
'////////////////////////////////////////////////////////////////////////////////////
'////////////////////////////////////////////////////////////////////////////////////
```

```
Public Sub BuildTheCoil(ByVal Nlayers As Int32)
    ' This function takes the physical parameters of the coil and calculates
    ' its resistance and inductance.  All of the variables except the number
    ' of layers are global variables.
    ' Step #1: Calculate the resistance.
    LenWire = 0
    For I As Int32 = 1 To Nlayers Step 1
        LenWire = LenWire + _
            (NTurns * 2 * Math.PI * (Rcore + ((Nlayers - 0.5) * Dwire)))
    Next I
    Rcoil = RPerM * LenWire
    ' Step #2: Calculate the inductance.
    RADavg = Rcore + (0.5 * Nlayers * Dwire)
    AREAavg = Math.PI * RADavg * RADavg
    Lcoil = 4 * Math.PI * 0.0000001 * _
        (NTurns * Nlayers) * (NTurns * Nlayers) * AREAavg / Hcoil
End Sub

'/////////////////////////////////////////////////////////////////////////////////////
'/////////////////////////////////////////////////////////////////////////////////////
'//
'//                      Interpolation in the force field table
'//
'/////////////////////////////////////////////////////////////////////////////////////
'/////////////////////////////////////////////////////////////////////////////////////

Public Function InterpolateForceField(ByVal Z As Double) As Double
    ' This function returns the interpolated magnitude of the force field at
    ' for the abscissa passed as the argument.  The returned value will be
    ' negative unless the given abscissa lies outside the force field, in
    ' which case the function returns zero.
    Dim IndexToLeft As Int32
    Dim Slope As Double
    ' Check if the slug is positioned outside the force field, to the left.
    If (Z > Ftable(1, 1)) Then
        InterpolateForceField = 0
        Exit Function
    End If
    ' Check if the slug is positioned outside the force field, to the right.
    If (Z < Ftable(FnumZ, 1)) Then
        InterpolateForceField = 0
        Exit Function
    End If
    ' Find the index of the abscissa in the force field table which lies just
    ' to the left of the given abscissa.  Note that this routine does not
    ' assume that the points in the table are equally-spaced.  It assumes only
    ' that they are in order of decreasing values, that is, each point is
    ' closer to the origin than the preceding point.
    For I As Int32 = 1 To (FnumZ - 1) Step 1
        If ((Z <= Ftable(I, 1)) And (Z > Ftable(I + 1, 1))) Then
            IndexToLeft = I
            Exit For
        End If
    Next I
    ' Let us check for the trivial case, where we are exactly at a data point.
    If (Z = Ftable(IndexToLeft, 1)) Then
        ' By chance, we are exactly at a data point.
```

```
            InterpolateForceField = Ftable(IndexToLeft, 2)
            Exit Function
        End If
        ' We deal here with the general case, where we are between two data points.
        Try
            Slope = (Ftable(IndexToLeft + 1, 2) - Ftable(IndexToLeft, 2)) / _
                (Ftable(IndexToLeft + 1, 2) - Ftable(IndexToLeft, 1))
        Catch ex As Exception
            MsgBox("Error in the force field table.  Two abscissas are the same.")
            InterpolateForceField = 0
            Exit Function
        End Try
        InterpolateForceField = Ftable(IndexToLeft, 2) + _
            (Slope * (Z - Ftable(IndexToLeft, 1)))
    End Function

End Class
```