Using the PSMA
Rectangular Waveform Core Loss Data

John H. Harris

8 August 2011

Contents

Appendices

List of Tables

List of Figures

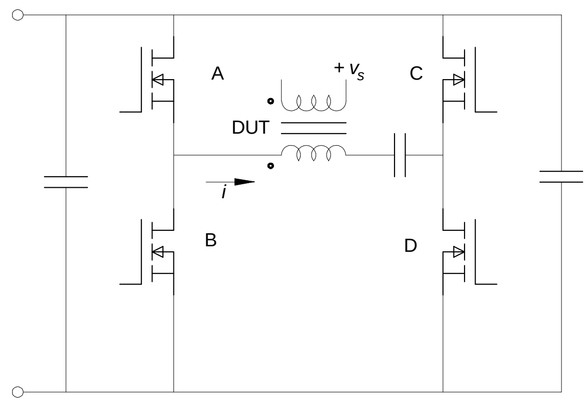


Figure 1: Full-bridge excitation circuit. The device under test (DUT) is a magnetic core.

1 Introduction

The Power Source Manufacturers Association (PSMA) has sponsored a research project to investigate magnetic core loss under excitation with rectangular voltage waveforms. This document is a guide to using the data that were compiled in the course of the research. For a discussion of the findings of the project, see *Testing Core Loss for Rectangular Waveforms, Phase II, Final Report* [3].

2 The Basic Experiment

In the basic experiment, a magnetic core is driven with a full bridge switching circuit (Figure 1). There is a second sense winding for measuring flux, to avoid measuring resistive loss in the drive winding. A blocking capacitor insures zero net DC current. In most tests, the capacitor voltage is negligible, and is only needed to cancel small volt-second mismatched due to timing errors. However, asymmetric waveforms are implemented by using significantly different on-times for positive and negative pulses. The capacitor voltage automatically adjusts to reduce the voltage of the longer pulse in order to have equal volt seconds between positive and negative pulses. Our apparatus is capable of taking measurements in a range of about 5 kHz to 500 kHz. The sample core is oil cooled, usually kept at 80 °C.

We take measurements with an Agilent Technologies DSO 7104A oscilloscope. Current is measured with a Tektronix P 6021 AC current probe. The switching bridge is controlled by a programmable function generator with some additional logic. This determines the frequency and pulse lengths. The bridge power supply voltage determines the voltage amplitude. All three instruments are under the control of a computer, which orchestrates sets of experiments and manages the resulting data (Figure 2). The oscilloscope sweep time is one

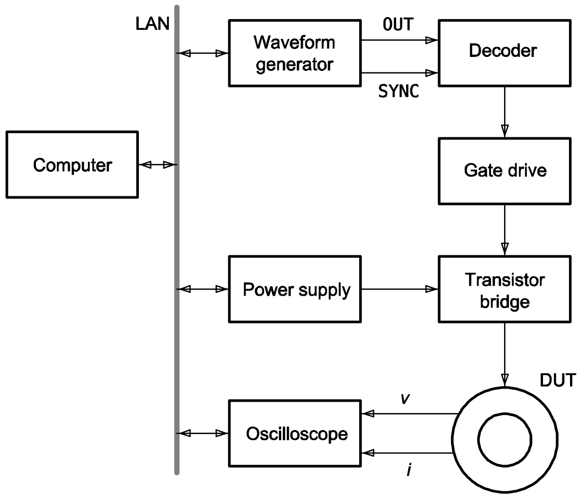


Figure 2: Block diagram of the experimental apparatus. The power supply, drive bridge, and oscilloscope are all controlled by the computer, which also gathers, and analyzes, the resulting data.

period. It averages the information from 512 triggers, and saves voltage and current information at 1000 time-points per period.

We refer to one of these experiment as a *run*—think of a run as single captured waveform. A run takes only a few seconds, but wound cores are normally tested with a batch of runs, called a *run set*, in which frequency, amplitude, and pulse widths are varied from run to run. Run sets always include square wave runs, which characterize the coil under test. Data from more than 4000 runs have been collected and accompany this report. Other wave forms (Appendix A) are also usually included to provide validation of the composite waveform hypothesis (CWH), or for exploring other phenomena.

3 Using the Data

The PSMA Core Loss project has generated a lot of data, and it can be bewildering to a new user. In this section we will take a top-down view of it. The data might reside on a DVD-R disk, or on a web site, but either way, its organization is the same. The data are contained under a single home directory (typically named `data/`),[1] which contains files of general interest, and subdirectories of data. Users with different interests will be looking in different subdirectories. All users should probably read Section 3.1 to become familiar with our naming conventions. Many users will want to look over the listing of the various run

[1]Different operating systems use different symbols for directory path separators. We will use `/` in this document.

sets to choose ones of interest, noting their set identifier (*`setId`*), as described in Section 3.2.

The following paragraphs describe some common use scenarios.

A tour of the data. A good way to become familiar with the data is to choose a run set, and then browse its directory, `sets/`*`setId`*`/` using the operating system's file browser. Take a quick look at the oscilloscope plots in `sets/`*`setId`*`/images/`. For image directories like this one, it may be helpful to set the browser to view thumbnails of the files. Also, try opening `.png` files with an image viewer, such as the one integrated into Microsoft Explorer (right click the file and select Preview). With it, you can interactively move from image to image in the directory.

Many of the interesting plots are in the `sets/`*`setId`*`/results/` directory. Use Table 2 as a quick guide to the plot files. See Section 6 for more information.

Graphical PWM design. Users wanting to characterize a material for design can see conventional plots of loss versus peak flux density and loss versus frequency, as well as Herbert plots and hysteresis plots (Section 6.1).

Numerical PWM design data. Designers wanting to use the two-plane Steinmetz model can take the parameters off the loss-versus-frequency plots, or can find this same data for *all* the sets in the file `steinmetz.dat`, found in the home directory (Section 5.2).

CWH validation. Users interested in the validation experiments for the composite waveform hypothesis can look at pulse-family plots found in `sets/`*`setId`*`/results/`. These image files have names that begin with `expand`, `skew`, `hippo`, or `asym`. The drilled core experiments look for evidence of transient flux migration within a core. Both are discussed in Section 6.2.

Basic research with raw data. Researchers who wish to delve deeper into the data can look at the raw data acquired by the oscilloscope in CSV files found in the `sets/`*`setId`*`/scope/` directories (Section 5.1). Also of interest are the processed data files named `data-`*`setId`*`.dat`, found in the `sets/`*`setId`*`/` directory (Section 5.2).

3.1 Nomenclature

Let us pause for a moment to discuss run, run set, and core *identifiers*. These are short strings used as names for the important objects in this project. In this document we represent them symbolically with

`runId` the run identifier, which uniquely identifies an single experimental run, in which we measure the electrical properties for one waveform on one wound device.

setId the (run) set identifier, which uniquely identifies a group or *set* of runs for a single wound device.

coreId the core identifier, which uniquely identifies a single core design, usually specified by manufacturer and part number, and perhaps material and coating as well. (It does *not* distinguish between different instances of that design, nor does it say anything about how the core is wound.)

These identifiers are arbitrary, however, we do use a few conventions:

- Core IDs usually begin with a few letter that suggest the manufacturer, e.g., `mi` for Magnetics Inc.
- Identifiers end with integers padded with zeros. This preserves sequential order in a lexical sort, such as in a directory listing.
- Set IDs usually start with the core ID, followed by a dash and an integer.
- Run IDs start with the set ID, followed by a dash and a three-digit, zero-padded integer.

For example, ***coreId*** `mi01` is tested in ***setId*** `mi01-1`, which includes ***runId*** `mi01-1-023`, a square wave with a period of 3.2 μs and a peak voltage of 3.2 V.

3.2 Browsing and Retrieving

To get a quick idea of what data are available, two summary reports, found in the home directory, consolidate information from *all* the run sets and cores.[2]

The run set summary report, `setsrept.txt`, contains a subset of the fields from the set generator data file *and* the core data file, for all the runs conducted. Here is a typical entry:

```
mi01-1   First of 3 in constriction experiments.
  Core mi01, sn 1: Magnetics Inc., 42206-TC shape, R material.
  5 turns. 120 data. Waveforms: square skew expand asym.
  Core note: Three items with test data, and mfr assigned sn
    1,2,3. SN 3 was used in Phase~I.
  Set note: Compare with runs sets mi09 and mi10. Core mi001 sn 1
    no longer exists; was used to make core mi009.
```

It contains (line 1) the run set identifier, and a descriptive comment, (2) the core identifier, serial number, manufacturer, shape, and material, (3) the number of turns, number of runs, and a list of waveforms used, (4) the optional run set note, giving more information about the run, and (6) the optional core note giving more information about the core.

[2]Each run set has two files that summarize the experimental set up in detail: the run generator data file, and the core data file (Section 4). These files contain all the details, but are not particularly convenient for browsing because they are scattered across the run set directories. The summary reports are generated from these files.

The core data summary file is similar, and summarizes the information in the core data files. For example, this is the entry for core used in the example above:

```
mi01  Magnetics Inc. 0R-42206-TC. (42206-TC shape, R material)
  Note: Three items with test data, and mfr assigned sn 1,2,3. SN 3
  was used in Phase I.
```

It contains (line 1) the core identifier, manufacturer, part number, shape, material, and (line 2) the optional note.

Once the user has located a run set of interest, he can browse the directory tree or download a zip archive (Section 3.3).

3.3 The Directory Structure

Here is a summary of the directory structure:

`/` (The home directory.) Contains general information, including this manual, and text files listing the cores and run sets.

`/cores/` Contains data files describing the various magnetic cores (Section 4.3).

`/sets/` Contains subdirectories, each containing the data for a single run set, and named for the run set identifier, ***setId***.

`/sets/`*`setId`*`/` Contains various data files describing the run set ***setId***, and subdirectories with raw data and plots. A run set directory typically contains about 50 MB of data, for several hundred runs, each characterized by frequency, peak voltage, and wave shape.

`/sets/`*`setId`*`/scope/` Contains the raw data from the oscilloscope: one CSV file for each run. Each CSV file comprises 1000 data points for one period of a waveform. (Section 5.1.)

`/sets/`*`setId`*`/images/` Contains plots of the scope data for each run, showing voltage and current, as well as power, and energy, all versus time. Several image formats are provided. Using image viewing software, the user can quickly browse the waveforms in the run set (Section 6). The image file formats are discussed in Appendix B.

`/sets/`*`setId`*`/auximages/` Embedded winding sets have an additional image directory (Section 6.2).

`/sets/`*`setId`*`/results/` Contains plot images summarizing the run set, comparing runs in families of curves having like pulse shapes.

`/zips/` Contains zip archive files of the run set directories, for convenient downloading and distribution. The zip files have names like `runs-`*`setId`*`.zip`. Each zip file is self-contained, containing supporting information (including this document) for the user. There is also an archive, `cores.zip`, of the contents of the `/cores/` directory.

Browsing the home directory, one finds `readme.txt`, which mostly directs the user to *this* document, in `user.pdf`. Users familiar with the system will go directly to the summary reports, `runsrept.txt` and `corerept.txt`, for browsing, and identify the set ID of a run set of interest. Then they can find its subdirectory under the `sets/` directory (to browse), or its zip file in the `zips/` directory (to retrieve).

4 The Input Data Files

This section summarizes the files that describe the experiment to the apparatus, for it to execute. They also serve as documentation, and so we have tried to make them human readable, and easy to parse with other software.

Here we give field names along with their meanings. Table 1 gives a summary of all the files. In these files, we use designations to identify the five waveform types (see Appendix A, Figure 5 and Table 3). The `.dat` data files are ASCII text files and can be viewed with an ordinary text editor (Appendix B).

An experimental *run* is a data sample of one waveform shape, with a given period, T, pulse width, t_1, and power supply voltage, V_{PS}, and perhaps other parameters, depending of the type of waveform. Experimental runs are grouped in *families*—several runs that vary one parameter while holding others constant. A *run set* (identified by a ***`setId`***) comprises all the runs (and families of runs) found in a single run set file.

Initially, the experimenter, using a text editor, edits a file called the *set generator* file named `gen-`*`setId`*`.dat`, describing a run set in terms of a few general parameters. This file serves two purposes, (1) to communicate the experimenter's choice of parameters to the run set generator program, and (2) to document the experiment.

The run set generator program reads the set generator file as input, and creates a run set file (named `runs-`*`setId`*`.dat`), that is a table listing each individual run, to be read by the control program that operates the apparatus. Using the control program interactively, the experimenter can execute any single run (for exploratory work, single runs can be edited before running), or more typically, *all* the runs can be executed sequentially and automatically.

The oscilloscope, also under computer control, measures the resulting voltage and current data, and uploads it to the control program, which stores it in files named ***`runId`***`.csv`. These CSV files contain the raw data for the experiment.

All the runs used to characterize a given core can be included in a single run-set file, and the data gathered in about half an hour. Further processing of the raw data is accomplished as an additional process.

The file names, field names, and their interpretation are presented in the following subsections. The file formats are described in Appendix B. We will present detailed information about the run set file first, because a good understanding of it is necessary to understand the subsequent topic, the run generator input file.

Type Directory File	Description
Input	
`/cores/`	
`core-`*`coreId`*`.dat`	The magnetic core data file, with information describing one particular core design.
`/sets/`*`setId`*`/`	
`gen-`*`setId`*`.dat`	The run set generator input data file, instructions for making the input file for the apparatus control program.
`runs-`*`setId`*`.dat`	The output from the run set generator, the input to the apparatus control program. Each record (a line) describes one run.
Raw data	
`/sets/`*`setId`*`/scope/`	
`runId``.csv`	The output from the oscilloscope for a single run.
`runId``.txt`	A configuration log from the oscilloscope for a single run. This file is missing in experiments run after direct (network) downloading was adopted (revision 95).
Results	
`/`	
`steinmetz.dat`	A table of the two-plane Steinmetz parameters for all the run sets.
`/sets/`*`setId`*`/`	
`data-`*`setId`*`.dat`	A table of basic derived data, including the energy loss per period and peak flux. It has the same format as `runs-`*`setId`*`.dat` file, but with six additional columns.

Table 1: Numerical data files. File naming conventions for parameter and data files. *`setId`* refers to the run set identifier, *`runId`* to the run identifier.

4.1 Run Set Data

The run-set input file is read by the apparatus control software. Each record is a white-space-delimited list of the parameters for one run. An experimental run has these parameters:

`runId` The run identifier. This is chosen to be be globally unique. The run-set generator program creates the run ID by appending a dash and a zero-padded integer to the run set identifier.

`coreId` The core sample identifier. This uniquely identifies a given the manufacturer, core geometry, and material. Different parts with the same manufacturer's part number are distinguished by our serial number field, `sn`, found in the set generator file (Section 4.2), but it does not appear in *this* data file—all the runs in a set use the same wound device and hence the same `sn`.[3]

`N` The number of turns on the core. We assume that the number of drive and sense turns are equal.

`period` The period, T, of the drive voltage, in seconds. (Floating point.)

`t1` The pulse width of the first pulse, in seconds. (Floating point.) The "first" pulse is the first of the two in the waveform generator output waveform.

`pinch` The *pinch*, P, it the time between pulse 1 and pulse 2, t_0, expressed as a fraction of the off time for a symmetric waveform:

$$P = 1 - \frac{2t_0}{T - t_1 - t_2}$$

If D is the duty cycle, $0 \leq P \leq 1$ for $D < 1$. $P = 1$ for `expand` waveforms, and is undefined for $D = 1$ (indicated in the file by `-1`). (Floating point.)

`Deq1` A Boolean value, true (1) if the duty cycle $D = 1$, and 0 otherwise. This may seem redundant, but it is needed for clarity and safety, because of the possibility of roundoff errors when comparing floating-point time parameters.

`Vps` It is the voltage requested from the power supply.

`vRange` Full scale voltage range setting for the oscilloscope.

`iRange` Full scale current range setting for the oscilloscope.

`ok` A Boolean value, true (1) if the run produced usable, valid data, falser (0) otherwise. Runs where `ok` = 0 are neither plotted individually, nor used in comparison plots.

The following fields are used by the data reduction programs to group data by families. The nonzero values indicate the plotting sequence:

[3]Early in the Phase I project, `coreId` identified specific cores, not just the manufacturer's part numbers. As the project evolved, serial numbers were added, but the format of this file was not changed, except by adding columns on the left.

`TSet` Zero if it is not a `square` wave. Otherwise, records with the same value have the same period.

`VSet` Zero if it is not a `square` wave. Otherwise, records with the same value have the same V_{PS}.

`expandSet` Zero if it is not an `expand` waveform. Otherwise, records with the same value have the same pulse shape (varying t_0).

`skewSet` Zero if it is not a `skew` waveform. Otherwise, records with the same value have the same pulse shape (varying `pinch`, P).

`asymSet` Zero if it is not a `asym` waveform. Otherwise, records with the same value have the same pulse shape and volt-seconds product for the t_1 pulse.

`hippoSet` Zero if it is not a `hippo` waveform. Otherwise, records with the same value have the same t_1 pulse pairs, varying the zero flux time, t_0.

`probe` If nonzero, the core has an additional singe-turn probe winding. The value is the oscilloscope sensitivity multiplier, used to give better resolution, by approximately adjusting for the single turn probe winding, and the reduced core cross-section area.

Empty lines are ignored by the control software, but are conventionally used to group families.

4.2 Set Generator Data

Run-set table data could be created by hand, using a text editor or spreadsheet program, but this would be tedious and error prone. Instead, the run-set tables were created by the run set generator program. Its input file is a white-space-delimited *dictionary* (Appendix B), and its primary purpose is to serve as a software interface, and to document the experiment.

The fields are:

`setId` The run set ID is used to name the run set data file and, by convention, is also prepended to each run ID.

`comment` The value is a one-line comment that will be printed at the top of the run set data file. This is used to record important experiment information. It serves as a title. It usually contains some spaces, and if so, it is enclosed in braces or quotations marks.

`coreId` The identifier for the core sample.

`sn` Serial number. If there is more than one sample core that otherwise has the same core ID (i.e., the same manufacturer and part number), each it is assigned (either by the manufacturer or the experimenter) a unique serial number.

`N` The number of winding turns.

`types` A list of waveform types to be generated. They can be any of {`square`, `expand`, `skew`, `hippo`, `asym`} (See Appendix A, Figure 5 and Table 3).

`probe` Nonzero if the core has an additional singe-turn probe winding, zero otherwise. Nonzero values are used by the control program as a multiplicative factor to adjust the oscilloscope sensitivity of the `out` channel, for better resolution.

`t1min` Minimum pulse width for `square` wave runs.

`t1max` Maximum pulse width for `square` wave runs.

`VpsMin` Minimum power supply voltage. This is determined by the noise floor.

`VpsMax` Maximum power supply voltage. This is determined by the capabilities of the drive system.

`vRangeMult` Oscilloscope voltage range multiplier.

`iRangeMult` Oscilloscope current range multiplier.

`grid` Spacing of the geometric series, in decilog (Appendix D). The usual value of 2 gives five values of pulse width and peak voltage per decade.

`satLimit` Saturation limit in peak volt-seconds per turn. In general, this number is determined by experimentation. Candidate runs using pulses that exceed this number are not used.

`minVsecPerN` Minimum volt-seconds per turn. This is normally set to 0, so that the minimum is determined by `VpsMin` and `t1min`. It is useful during setup to speed up the process of determining a good value for `satLimit`.

`expandSamples` A white-space-delimited, paired list of V_{PS}/N and t_1 for the pulses used in `expand`, `skew`, and `hippo` runs. These values will be "rounded" to the decilog preferred values (Section D). The values should be chosen so that there will be corresponding `square` pulses, for comparison.

`t0min` The minimum off-time between pulses, t_0, to be used for `expand`, `skew`, and `hippo` runs.

`t0max` For `expand` and `skew` runs, the maximum off-time. `expand` waveforms are symmetric, with the period $T = 2(t_1 + t_0)$. The off time, t_0, increases geometrically, stopping when it exceeds `t0max`. This value is chosen to remove runs that would be rejected because of blocking capacitance slump.

`t0maxHippo` For `hippo` runs, the maximum off time.

`TFactor` For `skew` runs, the ratio of the period, T, to the pulse width, used to determine T, i.e., the period is `TFactor * t1`. The series starts with $t_0 = t_{0\,\mathrm{min}}$, and increases t_0 geometrically until the pulses are approximately symmetric.

`A_L` Inductance per turn squared, in henrys.

`oilTemp` The temperature [°C] of the cooling oil, or `air` if air cooled.

`note` A place to put detailed notes about the run set. This field is not written to the run set data file, but is available to post processing for documenting the experiment.

The set generator data file is still used for handcrafted run set data, but in that case, the `type` field is left empty, to prevent set generator program from writing over the manual run-set data file.

4.3 Core Data File Format

The input file is a white-space-delimited dictionary (Appendix B). Some fields are obsolete and may be empty. The fields are:

`coreId` The identifier for the core sample. This is globally unique.

`mfr` Manufacturer.

`pn` Manufacturer's part number. This often encodes the core shape and material, etc., but we include it anyway, to reference the manufacturer's data. Modified cores have the word `custom` here, and an explanation in the `note` field.

`shape` The manufacturer's designation for the shape, e.g., `E19/8/5`. Modified cores may have the manufacturer's shape designation followed by an asterisk and explanation in the `note` field.

`material` The manufacturer's designation for the magnetic material.

`mu` Relative magnetic permeability.

`maxBvsF` Table of maximum flux density versus frequency. This is a paired list of f, B pairs. The maximum B is the minimum of `Bsat` (defined below), and the peak flux density that produces a core loss power density of `spPmax` (defined below) for sinusoidal excitation, as represented in the manufacturer's design data, in Hz, Gauss.

`Bsat` The saturation peak flux density used in compiling `maxBvsF`. This is a subjective judgment, usually inferred from the B versus H plot, in Gauss.

`vol` The core volume in mm^3.

`spPmax` The maximum specific power density used in compiling `maxBvsF`. It represents a maximum power density for the core, set by heat transfer consideration in a application typical for the core, in $\mathrm{W/cm}^3$.

`le` Effective flux path length, in mm.

`Ae` Effective cross section area, in mm^2.

`Amin` Minimum cross section area, in mm^2.

`m` Mass, in grams.

`gap` Gap length, in mm.

`Nmin` The minimum number of turns to be considered when estimating properties.

`Nmax` The maximum number of turns to be considered when estimating properties.

`A_L` The inductance per N^2, A_L, in henrys.

`note` Any relevant notes.

Fields that are lists are white space delimited. (Appendix E.2 shows a template core data file.)

5 Numerical Results

In Section 4 we presented the input data files and their field names and definitions. Here we present the output: raw data from the oscilloscope, and the results from numerical processing.

5.1 Raw Oscilloscope Data

The CSV format of the oscilloscope data files was dictated by the hardware. The first *two* lines are headings. (Note that they do *not* start with a `#` character, a syntax commonly used to denote "commented-out" lines. For these files, it is best to simply throw out the first two lines.) The first gives the field identifiers, the second, the units of measure. All the data are floating point numbers. The `SYNC` and `OUT` fields are the logic inputs to the waveform decoder and are usually only used for debugging.

`x-axis` Time, in seconds, with respect to the trigger event.

`SYNC` The `sync` signal from the arbitrary waveform generator [volts].

`OUT` The `output` signal from the arbitrary waveform generator [volts]. This channel was also used for the probe winding voltage in the embedded sense winding experiments.

`V` The sense winding voltage probe, V_s. [volts]

`I` The current probe. [amperes]

The raw data have been plotted and image files are found in the `sets/`*`setId`*`/images/` directory. These plots show the oscilloscope $v(t)$ and $i(t)$ (with DC subtracted off), as well as power ($P(t) = vi$) and energy ($E(t) = \int P dt$), for one cycle.

Note that the raw oscilloscope voltage and current data have a net DC offset. If you use these data directly, you may have to subtract off the average values.

5.2 Numerical Results

The basic analysis is done with Matlab$^{\text{TM}}$ and is stored in a file named `data-`*`setId`*`.dat`. The format is the same as the run set data file (Section 4.1), but with additional columns appended to each record:

`Ecyc` Loss energy per period, [J]

`Pavg` Average power. [W]

`Period1` The period, as measured by the oscilloscope. [sec]

`dPhi` Peak flux. [T]

`Ippm` Peak-to-peak current. [A]

The two-plane Steinmetz parameters are found in the file `steinmetz.dat`, in the home directory.[4] Our model fits two planes, for one each for low and high frequencies, with parameter subscripted 1 and 2, respectively. The file has these fields:

`setId` The run set ID.

`mrf` The manufacturer of the core.

`material` The manufacturer's designation for the core material.

`k1` The k_1 parameter.

`alpha1` The α_1 parameter.

`beta1` The β_1 parameter.

`k2` The k_2 parameter.

`akpha2` The α_2 parameter.

`beta2` The β_2 parameter.

`err` The standard error of fit in decibels, i.e., $10\log_{10}$ of the RMS fractional deviation of measured loss from the model.

6 Plots

It is convenient to think of the data plots as being either for *characterization* or *validation*. The former is derived from the square wave data, and is intended to characterize the properties of cores under square wave excitation. The later with non-squarewave excitation, and are intended to help detect deviations from the CWH. Both are summarized in Table 2, and described in more detail in Section 6.1.

6.1 Characterization Plots

There are a number of plots having to do with the characterization (`square`) data. These are found in `sets/`*`setId`*`/results/`:

`conven-`*`setId.ext`* These are conventional plots of core loss versus peak flux, parametrized by frequency.

`herbert-`*`setId.ext`* These are Herbert plots [1, 2], which show core loss versus pulse width, parameterized by peak flux.

[4] Note that this is a *two-plane* model, fit to *square wave* data, and *not* what is conventionally referred to by "Steinmetz approximation," which is for sinusoidal excitations. We use this name because, for each plane, the algebraic form is the same as its namesake.

Directory File	Description
`sets/`*`setId`*`/images/`	
`runId.ext`	A plot of the oscilloscope data for a single run, along with a plot of power and energy, versus time, for one period.
`sets/`*`setId`*`/results/`	
`conven-`*`setId.ext`*	Plots of square wave core loss power, versus peak flux, parameterized by frequency.
`herbert-`*`setId.ext`*	Plots of square wave core loss power, versus pulse width, parameterized by peak flux.
`hyst-`*`setId-n.ext`*	Families of square wave hysteresis curves, grouped by pulse width. *`n`* is the `TSet` family number.
`sf2-`*`setId.ext`*	A plot of two-plane Steinmetz curve fit for the run set.
`asym-`*`setId.ext`*	Plots of `asym` waveform core loss versus t_1/T, grouped in families by t_1 pulse size. The loss for a square wave with the same pulse size is shown for comparison for each family.
`expand-`*`setId.ext`*	Families of `expand` runs showing core loss versus off time. The loss for a square wave with the same pulse size is shown for comparison, for each family.
`hippo-`*`setId.ext`*	Plots of `hippo` waveform core loss versus zero-flux time, t_0, grouped in families by pulse size. The loss for a square wave with the same pulse size is shown for comparison, for each family.
`skew-`*`setId.ext`*	Plots of `skew` waveform core loss versus t_2, grouped in families by t_1 pulse size. The loss for a square wave with the same pulse size for t_1 is shown for comparison, for each family.

Table 2: Plot image files. *`setId`* refers to the run set identifier, *`runId`* to the run identifier. The image file extensions, *`.ext`* are either `.png`, `.eps`, or `.pdf`.

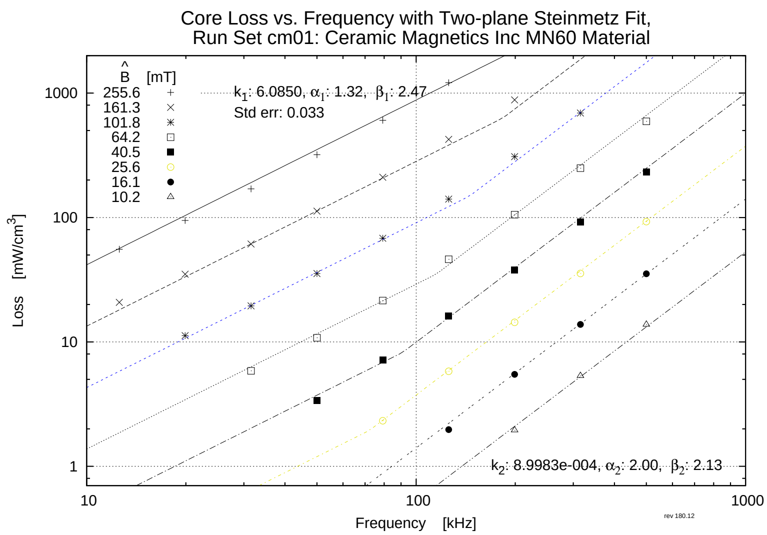


Figure 3: Two-plane Steinmetz curve fit.

`hyst-`*`setId.ext`* Hysteresis plots are presented for the square waveforms. They are presented in families, grouped by period (i.e., constant `TSet`, varying V_{PS}).

Because of ringing in the sense voltage signal, simply plotting V_s versus i gives a messy plot, with the hysteresis loop shape obscured by the ringing. We use boxcar averaging, tuned to the ringing frequency (with a $85.3\,\mu$s window) to drop out the ringing.

`sf2-`*`setId.ext`* These are plots of core loss versus frequency for square wave excitation, parameterized by peak flux density. The two-plane Steinmetz model is also plotted, for comparison. The slope discontinuities in the Steinmetz lines demark the intersection of the two planes.

6.2 Validation Plots

Now let us turn to the validations plots:

`shape-setId.ext` These are plots of the core loss for various validation waveforms, where *`shape`* is one of `expand`, `skew`, `hippo`, or `asym` (see Appendix A). Loss is presented in families of constant pulse shape, varying another parameter for which we would expect, by the CWH, invariant core loss.

`prb-`*`setId.ext`* These plots are provided only for the embedded winding experiments (runs sets `mi11-`*`n`*), and are discussed in detail below.

There are several run sets in which we investigate the flux distribution dynamics using a drilled core with embedded sense windings (also called *probe*

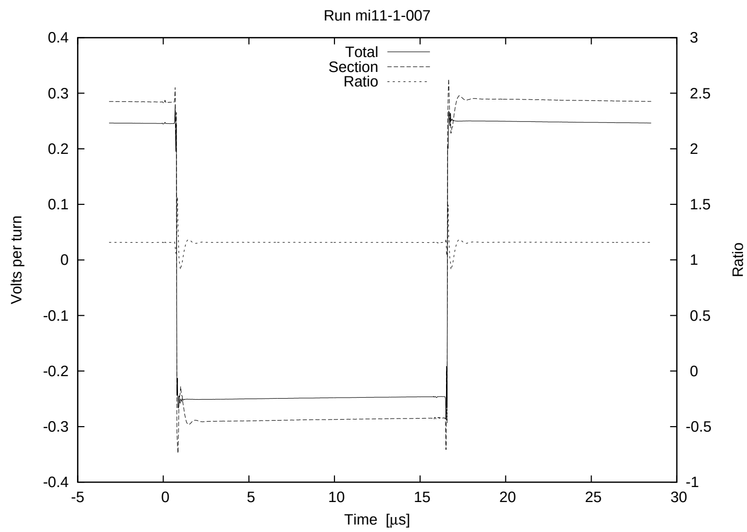


Figure 4: Embedded winding plot.

windings) that measure the flux in a subsection of the total core cross-section area. These run sets have identifiers of the form `mi11-`*n*, where the dash number, *n*, identifies various different probe winding configurations (Appendix C.1). In these experiments, the `OUT` channel of the oscilloscope (Section 5.1) is used to for the embedded winding probe.

The plot files are named `prb-`*`runId.ext`*, and are found in `sets/`*`setId`* `/auximages/`. Three graphics formats are provided. The plots show three values: (1) the full-core sense winding voltage, (2) the embedded winding voltage, and (3) the ratio of the two (Figure 4). To make visual comparison easier, the embedded winding voltage was scaled by NA/A_p, where N is the number of turns around the full cross-section, and A_p is the "probe area," the area of the core enclosed within the sense winding. This ratio would be unity if the flux distribution were uniform. We do not expect the flux distribution be uniform; see [3] for a full discussion.

During the off time in `expand` runs, the ratio signal is overwhelmed by digitization aliasing. Because of this effect we "lift the pen" for the ratio plot in these regions.

Appendix

A Waveforms

There are five excitation voltage waveforms used in this project. The `hippo` waveform is named for its resemblance to hippopotamus dentition.

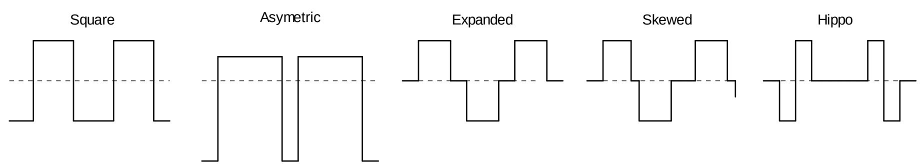


Figure 5: Voltage waveforms used in testing.

Designation	D	$t_1 = t_2$	Symmetric	Comment
`square`	$= 1$	yes	yes	Basic characterization
`skew`	< 1	yes	no	Constant T
`expand`	< 1	yes	yes	Varying T
`hippo`	< 1	yes	yes	Varying T, 0-flux dead time
`asym`	$= 1$	no	no	

Table 3: Waveform type designations.

B File Formats

There a few general types of file format used in the project:

Comma separated value (CSV). This is a standard tabular ASCII text file format, where columns are separated by commas. Files have the `.csv` extension. It was chosen because it was the only ASCII text output file format available from the Agilent oscilloscope. The first line or two of all the CSV files is a header.

Key-value dictionary. This is a particularly nice ASCII text file format for configuration files that is both human- and machine-readable. Each line has zero or more white-space-delimited key-value pairs. The *key* is a unique parameter name; the *value* is the value assigned to it. The layout and order of the pairs is unimportant. If a value is enclosed in quotation marks or braces, it may be empty or contain embedded white space. In this project, these files have the `.dat` extension.

In our data files, we use the number sign (`#`) character for comments. It may be used more than once per line.

White space delimited table. These are ASCII text table files that separate columns with white space—spans of one or more space or tab characters. They are easily read by a human if printed with a monospaced font, such as Courier. Cells that are empty, or contain white space, must be enclosed in quotation marks or braces. Blank lines and lines beginning with `#` are ignored. The latter are often used for headings. These files also have the `.dat` extension.

SQLite database. These are SQLite relational database files that were used in the implementation of the project. These are included in the distribution for the convenience of those who will know how to use them; it will be the easiest way for them to user the numeral data. See `http://www.sqlite.org` for more information.

There are also several image file formats:

Encapsulated Postscript. These are Postscript page description files, suitable for embedding in other documents. Files have the `.eps` extension.

Portable network graphics (PNG). These are raster graphics files, used for the data plots. Files have the `.png` extension. They are useful for browsing.

Portable Document Format (PDF). Adobe's ubiquitous page description format. These are slower to load but use vector graphics, which allows zooming at full resolution.

Gnuplot code. File having the extension `.plt` are Gnuplot code. When opened with Gnuplot (see `http://www.gnuplot.info/`) the user sees a dynamic plot window. Gnuplot can use it to generate scores of different image file formats, on most popular operating systems.

C Embedded Winding Details

In order to investigate the possibility that off-time losses were related to the delayed migration of flux, we used a toroidal core with holes drilled to permit the embedding of single-turn sense windings. The next section describes the nomenclature used to specify the winding geometry. Section C.2 reproduces the detail drawing for that core.

C.1 Subsection Notation

In the `mill-`*x* run sets, we investigate core flux distribution using a ferrite core with embedded sense windings. We fabricated a core drilled with intersecting through-holes. By threading wires through the holes, we can form a single-turn

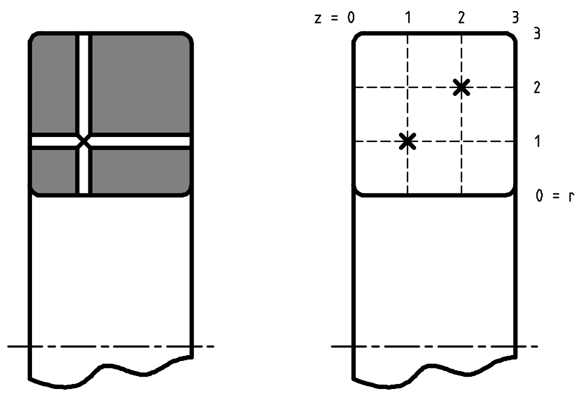


Figure 6: Drilled core and the subsection grid. Section view showing one pair of cross-point holes (left), and the grid used to describe subsection geometry (right). Cross points are show with ×.

sense winding to measure the flux in subsections of the core. By subtracting the signal from different windings, either electrically or numerically, one can infer the flux in *any* of the nine subsections.

The drilled core was fabricated from a toroidal core having a square cross-section, $0.5 \times 0.5\,\mathrm{in}^2$ (a Magnetics Inc. 0F46113TC, our core number `mi11`). The drilled core had the core number `mi11-1`. It has two cross points that divide the core area into nine equal, square subsections (Figure 6).

The different wirings are described in the run set data files, in the `note` fields. In order to name the different flux integration regions, we introduce a grid notation. Grid lines coincide with hole centerlines, and vertices are indexed with z, r pairs. The z direction is parallel to the axis of the toroid, the r direction is radial. By convention, the cross points are located on the $z = r$ diagonal.

Now we can economically designate any rectangular subsection on the grid with its two diagonal corner points, P_1 and P_2, specified with four ordered indices, $\{z_1, r_1, z_2, r_2\}$: the flux Φ_{z_1,r_1,z_2,r_2} is the flux passing through the rectangular area having diagonal corners at $\{z_1, r_1\}$ and $\{z_2, r_2\}$ (e.g., Figure 7). Alternatively, one can think of the four indices as representing the left, bottom, right, and top sides bounding the area, formed either by drilled holes or the outer surfaces of the core. Because our indices are only single digits, we can drop the commas for a more compact notation, e.g., the full cross-section area is A_{0033}.

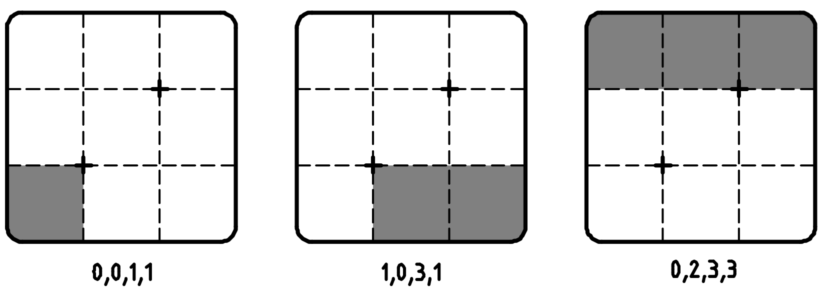
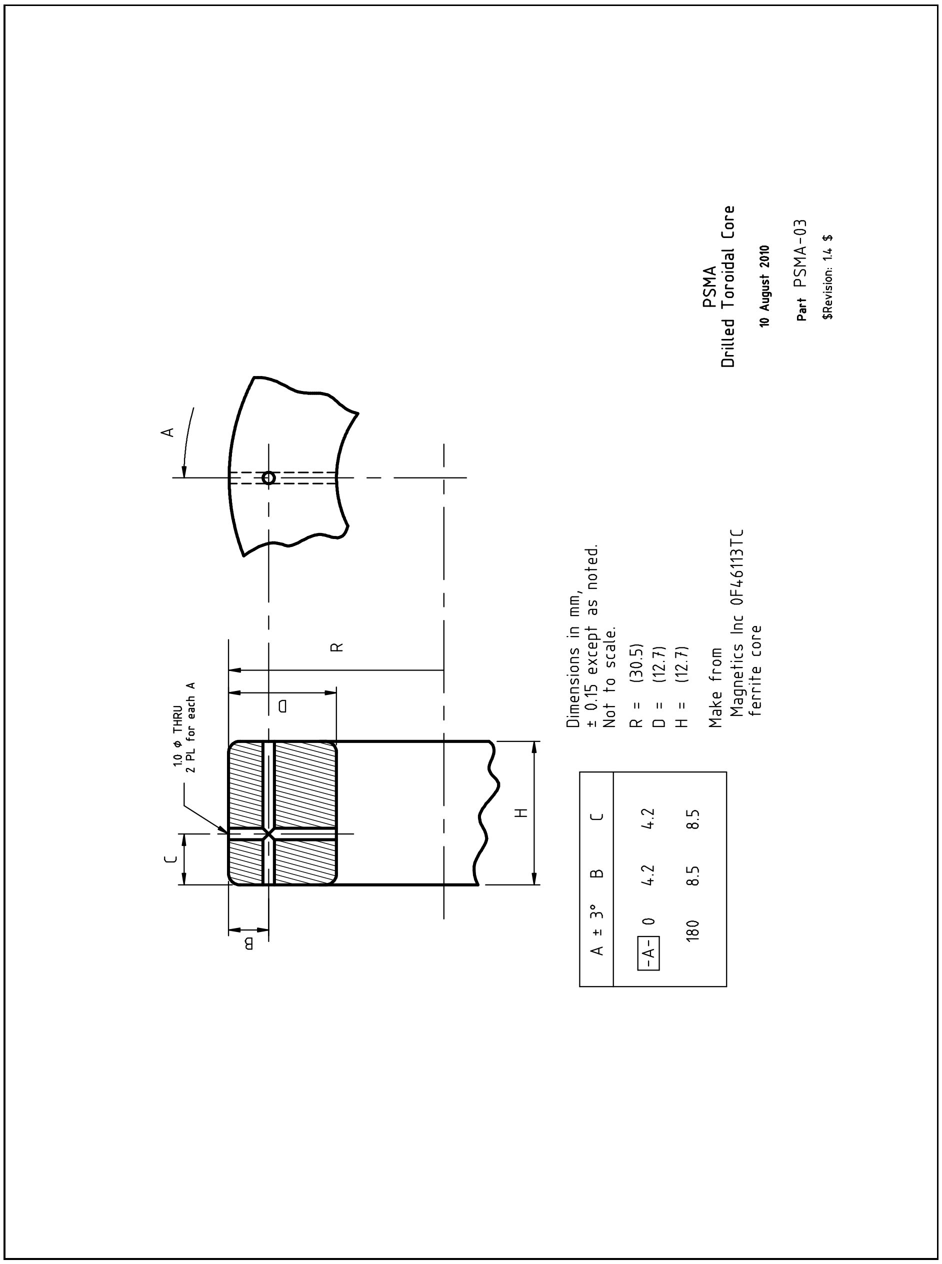


Figure 7: Examples of core cross-section subsections, and their grid designations.

C.2 Drilled Core

This is the mechanical detail drawing for the core used in the embedded winding experiments:



D Preferred Values

In order to make the data easier to plot and use, we use values of voltage-per-turn and pulse width in geometric series. The software application programming interface (API) deals with integer *decilogs*—integer values of $10 \log_{10} x$ (i.e., like decibels, but not restricted to power measurements). Our reference levels are 1 volt per turn, and 1 second. For the present project, points are even-integer decilog values, corresponding to the sequence $x = \{1, 1.58, 2.51, 3.98, 6.31, 10\}$. (These are the the Renard R5 series preferred values, and might be a good choice for industry standard product data presentation.) Note that we chose excitation voltages to be preferred *per-turn* values, so that data can be compared for devices with differing N, so the preferred values above may be hidden behind a factor of N.

E Input Data File Templates

Here we present the run set generator and core date file templates. These are the starting point for the experimentalist, who fills in the blanks. We reproduce them here to demonstrate the file formats.

E.1 Run Set Generator

The starting point for run set generator input files:

"bin/gen-template.dat" 24≡

```
setId   {}
comment {}
coreId  {}
sn      {}
N       0
types   {square skew expand hippo asym}
probe   0
satLimit 32e-6  # {max Vps*t1/N}
minVsecPerN  0  # {min Vps*t1/N}
A_L     1e-6    # H/turn^2
t1min   1e-6
t1max   1e-3
t0min   100e-9
t0max   20e-6
t0maxHippo 50e-6
VpsMin  1.0
VpsMax  16.0
vRangeMult 2.5
iRangeMult 3.0
grid    2       # {decilog grid spacing}
expandSamples {}
TFactor 6.0  # {max T/t1}
oilTemp {}
```

```
note    {}
◇
```

E.2 Core Data

The starting point for making core data files:

"bin/core-template.dat" 25≡

```
coreId  {}
mfr     {}
pn      {}
shape   {}
material {}
coating none
vol     {}
le      {}
Ae      {}
Amin    {}
m       {}
gap     0
mu      {}
Bsat    {}    # Gauss
spPmax  200e-3
Nmin    1
Nmax    50
A_L     1e-9  # Henry/turn^2
maxBvsF {5e3  {}  10e3  {}  25e3 {}  100e3 {}  250e3 {} }
note    {}
◇
```

References

[1] Edward Herbert. User-friendly data for magnetic core loss calculations. Unpublished., August 2008.

[2] Charles R. Sullivan, John H. Harris, and Edward Herbert. Core loss predictions for general PWM waveforms from a simplified set of measured data. In *Applied Power Electronics Conference and Exposition (APEC), 2010 25th Annual IEEE*, pages 1048–1055, February 2010.

[3] Charles R. Sullivan and John H. Harris. Testing core loss for rectangular waveforms, phase II final report. Technical report, Power Sources Manfacturers Association, 2011.