Loughborough University

This item was submitted to Loughborough University as a PhD thesis by the author and is made available in the Institutional Repository (https://dspace.lboro.ac.uk/) under the following Creative Commons Licence conditions.

creative commons

COMMONS DEED

Attribution-NonCommercial-NoDerivs 2.5

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

Attribution. You must attribute the work in the manner specified by the author or licensor.

Noncommercial. You may not use this work for commercial purposes.

No Derivative Works. You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the Legal Code (the full license).

Disclaimer

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/

LOUGHBOROUGH
UNIVERSITY OF TECHNOLOGY
LIBRARY

AUTHOR/FILING TITLE
PIMPERTON, M. G.

ACCESSION/COPY NO.
040040810

VOL. NO. | CLASS MARK

3 JUL 1992
20 MAR 1992
-4 AUG 1992
-1 JUL 1994
1 JUL 1994
-3 JUL 1992
-3 JUL 1992
-2 JUL 1993
30 JUN 1995
-4 DEC 1992
-3 JUL 1992
28 JUN 1996
-2 JUL 1993
17 NOV 1996
-9 JUL 1993
27 JUN 1997
FOR REFERENCE ONLY
-6 MAR 2000
-3 JUL 2000
-8 JAN 2001

040040810 4

The Meatgrinder: an Efficient Current-Multiplying Inductive Energy Storage and Transfer Circuit

by

M G Pimperton B.Sc., AMIEE

A Doctoral Thesis
Submitted in partial fulfilment of the requirements for the award
of the degree of Doctor of Philosophy
of the Loughborough University of Technology

December 1990

Supervisor: Dr V V Vadher
Department of Electronic and Electrical Engineering

(C) by M G Pimperton 1990

Loughborough University
of Technology Library
Date 1/10/91
Class
Acc
No. 40040810

ABSTRACT

The meatgrinder is a high-efficiency inductive energy storage and transfer circuit which may be used to supply high-current pulsed power requirements in applications such as electromagnetic propulsion. It overcomes the inherent 25% efficiency limit when transferring energy between uncoupled inductors and simultaneously provides current multiplication.

An unloaded six-step demonstration circuit has been used to multiply current from 7A to 76A at an efficiency of 44%, and a single-step demonstration circuit has been used to multiply the current in an uncoupled load inductor from 10A to 30A, the efficiency of energy transfer being 31%. Both circuits use power MOSFETs for switching.

These circuits have been used in conjunction with theoretical analysis and computer simulation to study the design and performance of the meatgrinder. Investigations have been carried out in order to confirm the basic theory, to clarify the details of circuit operation, and to provide the information necessary for future feasibility studies.

ACKNOWLEDGEMENTS

My thanks go to Professor I R Smith, my Director of Research, for making this work possible and for suggesting many improvements to the text of the thesis. My thanks also to my supervisor Dr V V Vadher for his guidance, encouragement and constructive appraisal of the thesis. I am very grateful also to Mr H R Stewardson, who provided much guidance in the initial stages of the research.

Assistance with equipment and measurements was provided by Mr J R Rippon and Mr A J Spencer, and Mr P Cherry produced some of the circuit diagrams. Thanks are due to Mr K Gregory for the use of his graph plotting programs. The assistance of the following members of the Department's mechanical workshop is also gratefully acknowledged: Mr P Barrington, Mr A Booton, Mr D Michelson and Mr G Wagg.

I am grateful to Mr P Senior for valuable technical discussions and for much-needed moral support.

For final production of the thesis, the computing, printing and copying facilities were provided by Campbell Scientific Limited.

My grateful thanks go to my wife Fiona, who laboured long and hard at the drawing board to mount and annotate the illustrations.

Finally, this research could not have been completed without the constant love and support of both my wife and friends at Herbert Street Independent Methodist Church.

This thesis is dedicated to my parents.

CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

LIST OF PRINCIPAL SYMBOLS

This list contains symbols which are used frequently in the thesis. Many of these symbols are used with several different subscripts. Where a particular variation does not appear in this list, the specific quantities to which the subscripts refer are defined in the text.

Symbol	Definition	Unit
α	inductance ratio L_1/L_2	-
β	current multiplcation	-
δx	change in quantity x	units of x
η	efficiency	-
η_d	efficiency of decompression	-
η_s	meatgrinder step efficiency	-
η_t	total meatgrinder efficiency	-
η_{td}	total meatgrinder efficiency with decompression	-
η_u	efficiency penalty due to uncoupled load	-
σ	ratio of meatgrinder coil inductance to load inductance	-
a	mean coil radius	cm
b	coil width	cm
c	radial coil thickness	cm
i	instantaneous value of current	A
k	coupling coefficient	-
k'	coupling coefficient with uncoupled load	-
t	time	s
Φ	magnetic flux	Wb
C	capacitance	F
E	energy	J
I	current	A

Symbol	Definition	Unit
K	coupling coefficient (in PSpice input files)	-
L	inductance	H
M	mutual inductance	H
N	number of turns	-
R	resistance	Ω
$R_{DS(ON)}$	saturation drain-source resistance of MOSFET	Ω
V	voltage	V
V_{sw}	voltage across opening switch	V
V_{ind}	induced voltage across previously switched-out coil section	V
$V_{o/c}$	total open-circuit voltage across previously switched-out coil section	V

CHAPTER ONE

BACKGROUND TO RESEARCH

1.1 INTRODUCTION TO PULSED POWER

1.1.1 Definition

Pulsed power generally refers to that area of technology which investigates the generation and application of short bursts of electrical power by means other than direct connection to a.c. or d.c. electrical sources.

The need for a pulsed power system arises when the current or voltage requirements of a load exceed the practical capabilities of available power sources. The solution to such problems can usually be divided into three parts [1,2]:

(a) Drawing the energy required from the source at a rate within the capability of the source.
(b) Storing the energy.
(c) Delivering the energy to the load at the required rate.

1.1.2 Applications

Systems generating instantaneous powers exceeding 1MW are in use or under development in fields such as fusion research, atomic particle research and defence [1,2]. Other applications include welding [3], lightning simulation [4] and fracturing of rock [5].

Work published in the literature concentrates on applications requiring currents of many thousands of ampères. It should be noted, however, that there could also be benefits in applying this technology at more modest current levels.

This could apply, for example, to a load requiring 50A in short bursts, which it draws from a lead-acid battery. Now the ampère-hour capacity of such batteries depends on the discharge rate. If, therefore, the required energy could be drawn from the battery at a lower current (and by implication over a longer period of time) then the battery would be able to supply the load for a longer time before becoming discharged. This could be achieved by interposing an energy storage and delivery circuit between the battery and the load, thus creating a pulsed power system. The system could draw energy from the battery over several milliseconds, the maximum current being restricted to 5A, for example, and deliver the energy to the load in microseconds, with the peak current being the required 50A. There would, of course, be penalties in terms of cost, complexity, size and weight, but this does not preclude such an application being viable in the future.

1.1.3 Important Factors In Pulsed Power

The design or investigation of a pulsed power system can be divided into two main parts. These are:

(a) the parameters which make up the performance specification of the system

(b) the components and techniques used to achieve the specification.

1.1.3.1 Performance

The performance specification is usually given in terms of the following major parameters:

(a) load current - up to megampères [6]
(b) load voltage - up to megavolts [7]
(c) risetime of load current or voltage - down to nanoseconds [8]
(d) efficiency, that is
  (energy delivered to load) / (energy drawn from source)
(e) pulse repetition rate - up to tens of kHz [9].

It is also common to quote figures for energy or instantaneous power.

1.1.3.2 Components and Techniques

The list below indicates a selection of the areas to which attention is paid by workers in this field.

The primary sources of energy may be:

(a) a.c. electrical sources, such as a 3-phase supply from the mains or a dedicated synchronous generator
(b) d.c. electrical sources, such as lead-acid batteries
(c) explosives, propellants or plasmas [1,10].

Energy may be stored:

(a) by charging a capacitor ($E=\frac{1}{2}CV^2$)
(b) in the form of the magnetic field of an inductor ($E=\frac{1}{2}LI^2$)
(c) as kinetic energy in a flywheel ($E=\frac{1}{2}I\omega^2$).

Energy delivery requires switching to reconfigure the circuit appropriately. Devices used to achieve this include:

(a) solid-state semiconductor devices, such as GTOs [9]
(b) explosively-operated opening switches [11,12,13]
(c) spark gaps [9]
(d) various types of gas-filled tubes, such as thyratrons or crossatrons [9]
(e) saturable inductors (magnetic switches) [14].

The technology of pulsed power can be thought of as being divided into energy supply, energy storage and energy delivery. It is important to remember, however, that such divisions are purely arbitrary. This is because a capacitor bank, for example, whilst obviously being a means of storage, is often regarded as the energy source for a system. Or, as another example, an explosive flux compressor [1] may have two sources of energy: a capacitor bank to charge the storage inductor and the chemical energy of the explosive to effect the flux compression and deliver the energy to the load.

Results from pulsed power research are published extensively, and there are many papers available on each of the topics mentioned above. In addition, specialist conferences such as the American IEEE Pulsed Power Conference are held at regular intervals. The proceedings of these conferences often include review papers [2,9,15,16].

1.2 INDUCTIVE ENERGY STORAGE

Inductors offer high energy density (measured in Jkg^{-1} or Jm^{-3}). Their resistance can be reduced by cooling (although this may not always be beneficial [1]). For some applications they may offer the best compromise between energy density, speed and efficiency [17].

When compared to capacitors the energy density of inductors is particularly attractive, as shown in table 1.1 [2]. Zucker [18] compared inductive and capacitive systems by regarding them as transmission lines subject to the same electric field limit. His analysis shows that, in such cases, inductive systems are inherently superior for power density (and by implication energy density).

STORAGE MODE/ DEVICE TYPE	ENERGY DENSITY		TIME SCALE TO DELIVER TO LOAD
	$Jm^{-3} \times 10^{6}$	Jkg^{-1}	
ELECTROSTATIC			
Capacitors	0.01-1	300-500	μs
MAGNETIC/INDUCTORS			
Conventional	3-5		
Cryogenic	10-30	$10^{2}-10^{3}$	ms to μs
Superconducting	20-40		
CHEMICAL			
Batteries	2000	10^{6}	minutes
Explosives	6000	$5x10^{6}$	μs
INERTIAL			
Flywheel	400	$10^{4}-10^{5}$	seconds

Table 1.1 Comparison of Energy Storage Techniques (1981 figures)

However, whereas a capacitor can be discharged by closing a switch onto a load, energy can only be transferred from an inductor by interrupting the current with an opening switch. This is a major problem, and much research is underway both to increase switch capabilities and to find circuit techniques which reduce the duty on the switches. The meatgrinder [20] is one such technique.

1.3 THE MEATGRINDER

1.3.1 Important Features

1.3.1.1 Efficiency

Energy compression theory [19,20] shows that for an energy transfer process to be 100% efficient it must either employ complementary forms of energy or be continuous and incremental.

"Complementary" means, for example, electromagnetic (inductive) and electrostatic (capacitive) energy. Thus when considering energy transfer between uncoupled inductors the 25% efficiency limit (see Appendix A) can theoretically be removed by using an intermediate capacitor. The disadvantage of this method is the size of the capacitor required [21]. Other schemes using kinetic energy as the complementary form have been suggested, but they too suffer from practical problems [17].

The most well-known continuous energy transfer system is the whip [19]. Its behaviour inspired Zucker [20,23], and later Wipf [24], to propose an electromagnetic equivalent. Zucker named his circuit the meatgrinder. Legentil and Rioux [13,21] had already shown the benefit of incremental processes and performed preliminary experiments.

Zucker [20,23] showed that a meatgrinder with an infinite number of coil sections would be 100% efficient when used to transfer energy to an uncoupled load inductor. Thus it was expected that practical circuits with very high efficiencies could be designed.

Lototskii [25] has proposed an alternative high-efficiency inductive energy-transfer circuit. It too aims to provide current multiplication and to transfer energy to uncoupled loads at efficiencies greater than 25%. The circuit appears to consist of several strings of series-connected inductors, the strings being connected in parallel with the load sequentially. The description of operation, however, is not particularly clear and Lototskii states that mutual inductance is a hindrance to the operation of his circuit rather than a help. The paper contains no experimental results and a recent literature search did not reveal any follow-up publications.

1.3.1.2 Current Multiplication

When open-circuiting one inductor into another in the "conventional" manner, the current falls (see Appendix A). In contrast, the basic meatgrinder action always leads to an increase in current for any non-zero coupling coefficient between the coil sections [20].

Transformers may be used to produce this type of current multiplication by first establishing a current in the primary winding with the secondary winding open-circuit, then short-circuiting the secondary winding and immediately open-circuiting the primary winding [11]. However, with an uncoupled inductive load this technique is again subject to the 25% efficiency limit [26], the energy being lost in the primary side opening switch.

1.3.1.3 Pulse Compression

The essence of pulsed power is that energy is delivered to the load much faster than it is drawn from the source. In the meatgrinder, after a relatively slow charge, this pulse compression occurs simultaneously with the current multiplication as the sequential switching proceeds. This means that the load current waveform resembles that shown in figure 1.1.

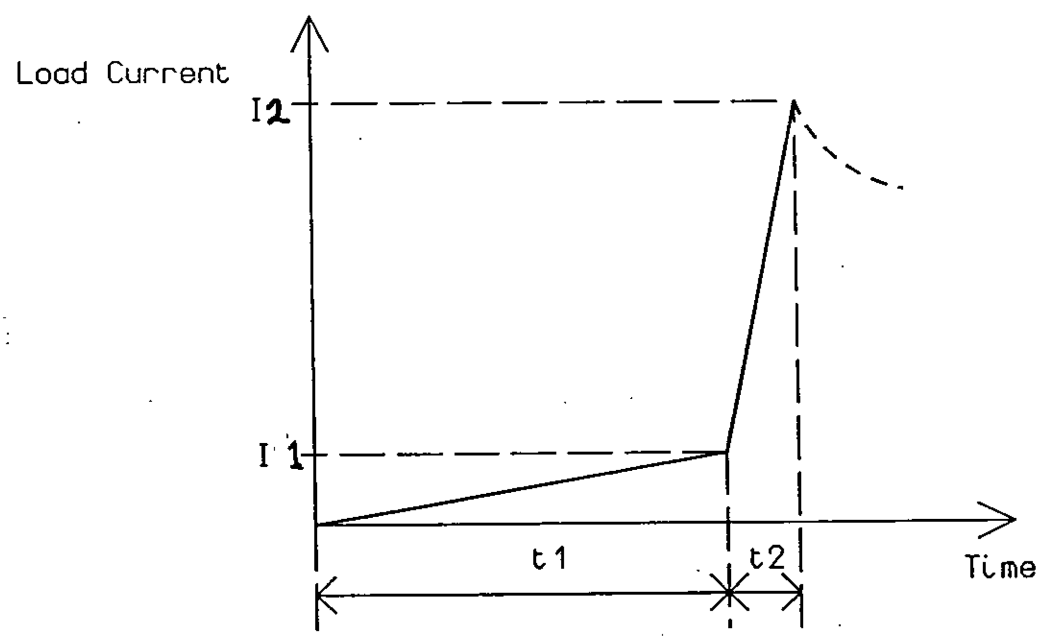


Figure 1.1 Idealised Meatgrinder Load Current Waveform

In figure 1.1:

Current multiplication $= I_2 / I_1$
Charge time $= t_1$
Energy transfer time $= t_2$

and hence it follows that if

Initial energy $= E_1$
Final energy $= E_2$

then

Efficiency $= E_2 / E_1 = \eta$

and therefore

$$\text{Power multiplication} = \frac{E_2}{t_2} \bigg/ \frac{E_1}{t_1} = \eta . \frac{t_1}{t_2}$$

1.3.2 Previous Work

Since Zucker and Long first proposed the meatgrinder, they and their co-workers at the Energy Compression Research Corporation (ECRC), California, have published several papers on the topic [20,23,26-33,35-38]. The papers have recently begun to acknowledge the work of Legentil and Rioux [13,21,22]. Their work in the early 1970s addressed several of the issues currently being studied.

The meatgrinder principle was first demonstrated with a four-step loaded circuit which multiplied current from 15A to 45A at an efficiency of 47.5% [27]. Experiments with a single-step meatgrinder [28] showed that the theory was equally applicable at high currents.

Theoretical designs of meatgrinders for use with electromagnetic (EM) guns have been published [30-33]. These designs include multi-step meatgrinders which provide current multiplication and constant current during launching, and a single-step meatgrinder which works in reverse to recover the energy left in the gun barrel. The energy recovery proposal has a smaller component count than the technique proposed by Ness and Chu [34]. Recent experiments [33] have confirmed the advantage of a single-step meatgrinder over a conventional technique which uses no switching.

A further recent design [38] uses a capacitor to eliminate the opening switch in a single-step circuit. The objective of this is to make "lost" energy recoverable, as proposed by Rioux [13].

Meatgrinders have also been proposed for use with explosive generators [35] and resistive loads such as radar or lasers requiring a pulse of constant current [30,31].

1.4 AIMS OF RESEARCH

Work on the meatgrinder in the UK was first started at LUT. The initial aim of the research was simply to gain familiarity with the concepts and techniques involved. It was recognised that this could best be done by building a working meatgrinder, which would serve both to confirm existing results and to fill in any gaps as regards characterising the circuit behaviour.

This initial work would then pave the way for examining the potential of the meatgrinder in existing proposed applications, and equally in novel applications yet to emerge.

CHAPTER TWO

COIL DESIGN AND CONSTRUCTION FOR A SIX-STEP MEATGRINDER

2.1 SPECIFICATIONS

In designing the experimental meatgrinder, the sole objective was to produce a working circuit which would allow the various principles to be investigated. This was the only basis on which the design parameters were chosen.

2.1.1 Number of Steps

This was fixed at six, i.e. seven meatgrinder coil sections. This number is, of course, somewhat artificial; normally the objective would be to keep the circuit as simple as possible by achieving the required performance in the minimum number of steps.

The circuit to be investigated is therefore as shown in figure 2.1. The last coil section L_7 acts as the load, which in this case is magnetically coupled to the meatgrinder coil. (See Chapter 6 for a discussion of the effect of an uncoupled load.)

2.1.2 Current and Current Multiplication

An arbitrary target of ten was chosen for the overall current multiplication. A current limit of 100A was chosen so that semiconductor switches could be used.

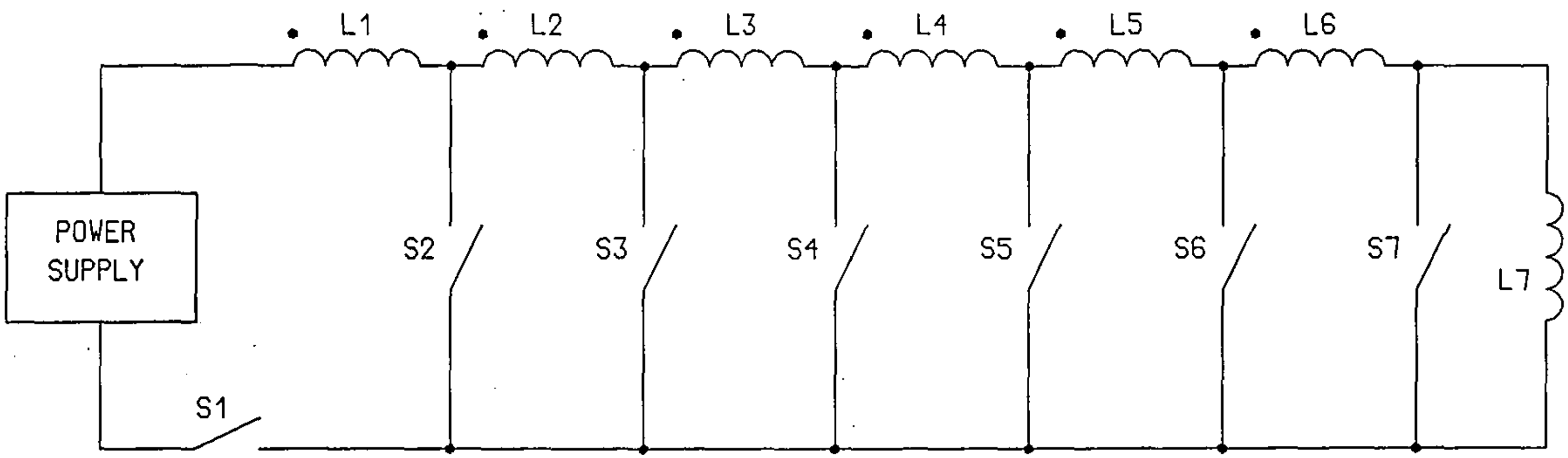


FIGURE 2.1 SIX-STEP MEATGRINDER CIRCUIT

2.1.3 Coil Geometry

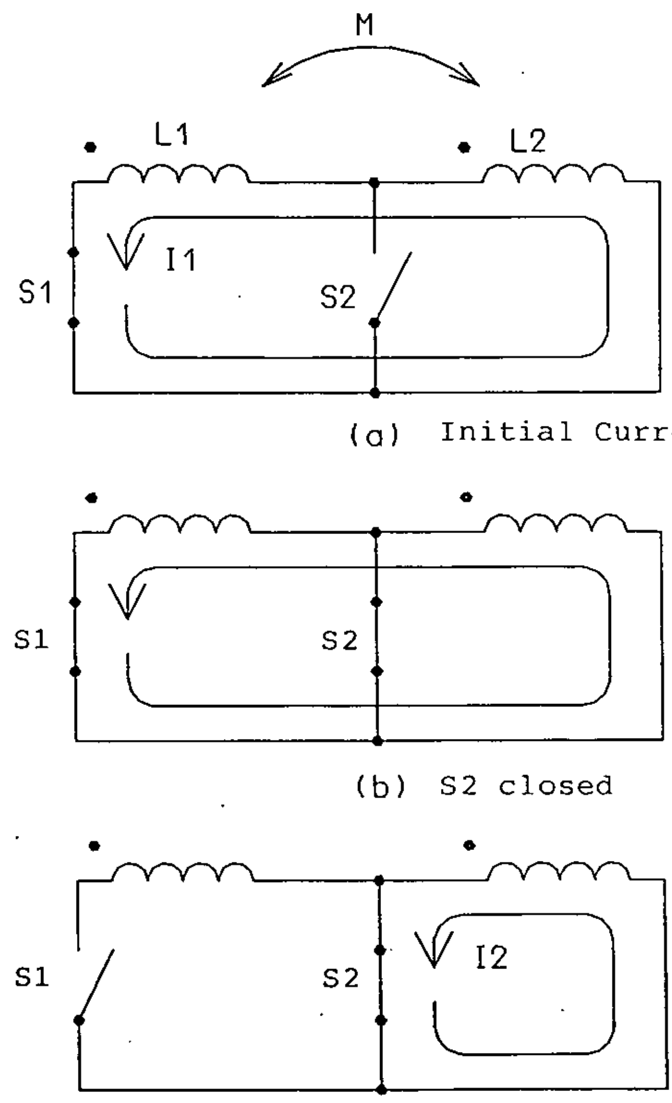
It was decided to wind the coil as a continuous spiral of copper strip, with appropriate tappings to divide it into the seven sections. The spiral strip construction has a number of advantages as it offers low resistance (due to the large cross-sectional area of the copper) and tight magnetic coupling.

A potential disadvantage of the spiral geometry is that it does not confine flux. This means, firstly, that the inductance can be degraded by eddy currents induced in nearby metalwork and, secondly, that the coil can become a source of electromagnetic interference (EMI). For the purposes of this investigation, however, these problems are not particularly important.

2.2 DESIGN FORMULAE

2.2.1 Meatgrinder

Figure 2.2 shows the operation of a single meatgrinder step, in which it is assumed that winding resistance is negligible and that the switches are ideal (i.e., they have zero on-state resistance, infinite off-state resistance and switching between the two states occurs instantaneously). In figure 2.2(a) the initial current I_1 is established in the meatgrinder. Closure of S_2 (figure 2.2(b)) has no effect because the current is constant and there is therefore no voltage in the circuit.



(a) Initial Current Established

(b) S2 closed

(c) S1 opened, energy transferred

Figure 2.2 Basic Meatgrinder Action

Figure 2.2(c) shows how opening S_1 diverts the current into S_2, thus transferring energy from L_1 to L_2 via the mutual inductance M. When the transfer is complete the ratio of the final and initial currents is [23], by flux linkage conservation considerations:

$$\frac{I_2}{I_1} = \frac{L_2+M}{L_2} \qquad (2.1)$$

If the initial and final magnetically stored energies are respectively E_1 and E_2 then

$$E_1 = ½(L_1+L_2+2M)I_1^2$$

and

$$E_2 = ½L_2I_2^2$$

Substituting for I_2 from equation (2.1) gives the efficiency of the energy transfer (η) as

$$\eta = \frac{L_2^2+M^2+2ML_2}{L_2^2+L_1L_2+2ML_2} \qquad (2.2)$$

On substituting $M = k\sqrt{L_1L_2}$ (where k is the coupling coefficient) and dividing by L_2 throughout, equation (2.2) becomes

$$\eta = \frac{1+k^2\alpha+2k\sqrt{\alpha}}{1+\alpha+2k\alpha} = \frac{(1+k\sqrt{\alpha})^2}{1+\alpha+2k\alpha} \qquad (2.3)$$

where $\alpha=L_1/L_2$.

2.2.2 Inductance Calculations

The coil is designed from a formula given by Grover [39]. For a circular coil with a rectangular cross-section the inductance is

$$L = 0.019739(2a/b)N^2aK' \qquad (2.4)$$

where

- a is the mean coil radius in cm
- b is the coil width in cm
- N is the number of turns
- K' is a tabulated correction factor accounting for end and insulation effects (function of a, b and the radial coil thickness c).

Note: Grover also gives a formula specifically for spirals of strip. However, this is not applicable in this case because it refers to coils whose width does not exceed their radial thickness.

The coupling coefficient between coil sections can be obtained to a good approximation by considering the ratio of the cross-sectional areas (see figure 2.3, in which the spacing between coils is exaggerated for clarity).

The area common to both coils is

$$A_1 = \pi r_1^2$$

and the area enclosed by coil 2 is

$$A_2 = \pi r_2^2$$

The coupling coefficient is then approximately

$$k = A_1/A_2 \qquad (2.5)$$

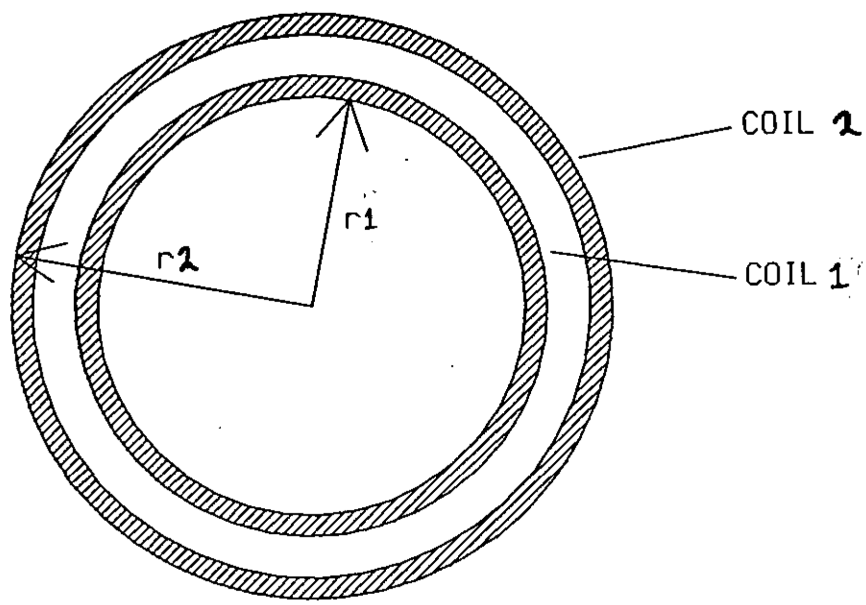


Figure 2.3 Calculation of Coupling Coefficient

This method is only approximate as it takes no account of any flux which links the whole of one coil but only part of the other; it also yields different results according to whether the coil radius used is the minimum, mean or maximum value. These differences are small, however, provided that the radial thickness is small in comparison to the radius of the coil.

2.3 DESIGN PROCEDURE

2.3.1 Notation for Total Inductance

In this work it is often necessary to calculate the total inductance of two or more inductors connected in a series-aiding configuration. These values are derived from the expression

$$\begin{bmatrix}\text{total}\\ \text{inductance}\end{bmatrix} = \begin{bmatrix}\text{sum of}\\ \text{self-inductances}\end{bmatrix} + 2\begin{bmatrix}\text{sum of mutual}\\ \text{inductances}\end{bmatrix}$$

In this thesis, total inductances are denoted as follows:

The total inductance of two inductors L_a and L_b in series is denoted as L_{ab}. Where more than two inductors are connected in series a hyphen is used in the notation to separate the first and last inductors in the chain. Thus the total inductance of four inductors L_c, L_d, L_e and L_f in series is denoted L_{c-f}.

Coupling coefficients are denoted in a similar way. For example, the coefficient of coupling between L_g and the total inductance L_{h-j} is denoted by $k_{g(h-j)}$.

2.3.2 Preliminaries

2.3.2.1 Initial Calculations

The objective of these calculations is to check that the figures being considered for parameters such as the inductance and efficiency of the meatgrinder are of the right order of magnitude.

The first step is to make the simplifying assumption that for each switching step, the coupling coefficient between the coil section to be switched out and the rest of the coil remaining in circuit is 0.8. It is also assumed that each step has the same efficiency η_s so that

$$\eta_t = (\eta_s)^6 \qquad \text{(overall meatgrinder efficiency)}$$

If η_t is arbitrarily chosen to be 70%, then η_s is 94.2%.

The final energy is calculated from the initial energy (arbitrarily set at 50mJ) and the overall efficiency (70%). Since the final current (100A) flows only in the last coil section L_7, the inductance can be calculated from $E = \frac{1}{2}LI^2$ and is found to be 7μH. By applying equation (2.3) to the sixth step, the inductance of L_6 is then obtained as 2.7μH.

Since k_{67} is known to be 0.8, the total inductance L_{67} can be calculated, and since the step efficiency is known the energy before the sixth step can be determined. From these two values the current before the sixth step is found to be 66.9A.

Repeating these stages for steps 5, 4, 3, 2 and 1 yields the figures given in table 2.1. Although arrived at without reference to physical construction, the figures do serve to indicate that the targets for current multiplication and efficiency can be reached.

STEP	CURRENT (A)	ENERGY (mJ)	INDUCTANCE SWITCHED OUT (μH)	INDUCTANCE REMAINING (μH)
-	7	50	-	2029
1	10	47	L_1 = 327	856
2	16	44	L_2 = 138	361
3	23	42	L_3 = 58	152
4	35	39	L_4 = 26	64
5	67	37	L_5 = 6	17
6	100	35	L_6 = 3	L_7 = 7

Table 2.1 Results of Initial Calculations
(for each step k=0.8 and η_s=94%)

2.3.2.2 Choice of Coil Radius and Width

Equation (2.4) shows that increasing the radius reduces the number of turns required for a given inductance. Fewer turns leads to a reduction in radial thickness, and this reduced thickness is also a smaller proportion of the coil radius. The coupling coefficients between the coil sections are thereby increased.

The upper limit on the radius is determined by physical factors such as the method of construction of the coil former and winding of the coil. With this in mind, the inner radius was restricted to 20cm.

The width of the coil was set at 10cm, although it is recognised that this is not necessarily optimum in terms of minimising the coil resistance.

2.3.3 Calculation Sequence

After carrying out the initial calculations described above, the next step is to produce a physical coil design as follows:

2.3.3.1 Design of L_7

This is achieved simply by estimating, to the nearest whole number, the number of turns required to give an inductance of approximately 7μH (as indicated by the initial rough calculations). Equation (2.4) is then applied to find the actual inductance provided by this number of turns. This is 9.3μH from four turns.

2.3.3.2 Design of L_6

It was decided that L_6 should consist of one turn, giving an inductance of 0.6μH. This was in order to make the last switching step small, thereby maximising the efficiency (see equation (2.3)) and minimising the duty on the opening switch during the switching of a relatively high current. (The trade-off in this decision is, of course, that the current multiplication falls. As described below, this is compensated for by higher multiplication in the fifth step.)

Having found the value of L_6, the value of k_{67} is calculated. This subsequently allows the efficiency to be found from equation (2.3). Finally the current before the sixth step (I_6) is calculated.

2.3.3.3 Design of Remaining Coil Sections

The design of the remaining sections is dealt with in the same way as that of the first section, the only difference being that the coupling coefficient calculation becomes longer.

As an example, consider the fifth step, where L_5 is switched out to leave L_6 and L_7 in series as the inductance in circuit. To calculate the transfer efficiency, the coupling coefficient required is that between L_5 and L_{67} (the series combination of L_6 and L_7). In order to maintain a reasonable accuracy in the "ratio of areas" method, $k_{5(67)}$ is calculated by first finding k_{56} and k_{57} separately.

Now the total value of all three inductances is

$$L_{5-7} = L_5+L_6+L_7+2(M_{56}+M_{57}+M_{67})$$

$$= L_5+L_{67}+2(M_{56}+M_{57})$$

which must be the same as

$$L_{5-7} = L_5+L_{67}+2M_{5(67)}$$

so that

$$M_{5(67)} = M_{56} + M_{57} \tag{2.6}$$

and

$$k_{5(67)} = \frac{k_{56}\sqrt{L_5 L_6} + k_{57}\sqrt{L_5 L_7}}{\sqrt{L_5 L_{67}}} \tag{2.7}$$

2.3.4 Design Figures

Design data for the various winding sections are given in table 2.2. The current multiplication figures in table 2.3 show the expected performance without any coil or supply resistance in the circuit.

SECTION	INDUCTANCE (μH)	TURNS	DC RESISTANCE (Ω)
L_1	387.0	25	0.058
L_2	244.8	20	0.045
L_3	134.6	15	0.033
L_4	47.7	9	0.020
L_5	21.0	6	0.013
L_6	0.6	1	0.002
L_7 (load)	9.3	4	0.009

Inner coil radius=20cm

Total radial thickness=1.1cm

Coil width=10cm

Table 2.2 Design Data for Windings

STEP	CURRENT (A)
–	5
1	8
2	12
3	20
4	37
5	80
6	100

Table 2.3 Expected Current Multiplication Using Inductance Figures From Table 2.2

2.4 PHYSICAL CONSTRUCTION OF THE MEATGRINDER

The coil was wound by hand on the rig shown in figure 2.4. The inter-turn insulation is two layers of 10cm-wide "Mylar", each layer being 0.1mm thick. The Mylar was dispensed from the two upper reel holders and the copper from the bottom one. Double-sided adhesive tape was used at regular intervals to secure the Mylar to the copper.

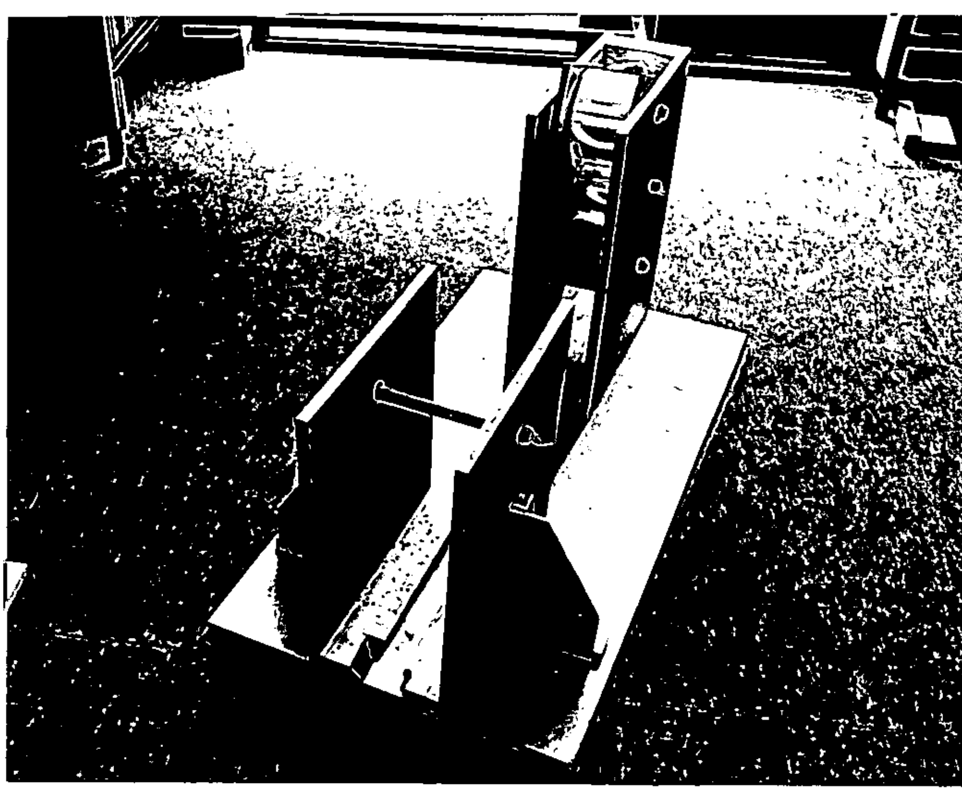


Figure 2.4 Coil Winding Rig

The copper used was 0.1mm thick and 8cm wide. This width ensures that the insulation overlaps properly on both sides. The change in inductance caused by this change of width is only a few percent and is therefore not critical.

A former was constructed from "Darvic" insulating material and was suspended on the winding rig by means of temporary end plates.

Figure 2.5 shows the completed coil. Access to the junctions between sections is provided by copper strips secured to the main winding by silver-loaded heat-cured epoxy adhesive. These are brought out through slots in the end cheek and are clearly visible in figure 2.5(a). The end of L_7 is brought through a slot in the former and connected to a copper ring (figure 2.5(b)) which provides a common return for the switches. The ring is mounted so as to present the minimum cross-sectional area to the flux passing through the centre of the former, and it also has a break at one point on the circumference. Both of these measures are designed to minimise eddy currents in the ring during transients.

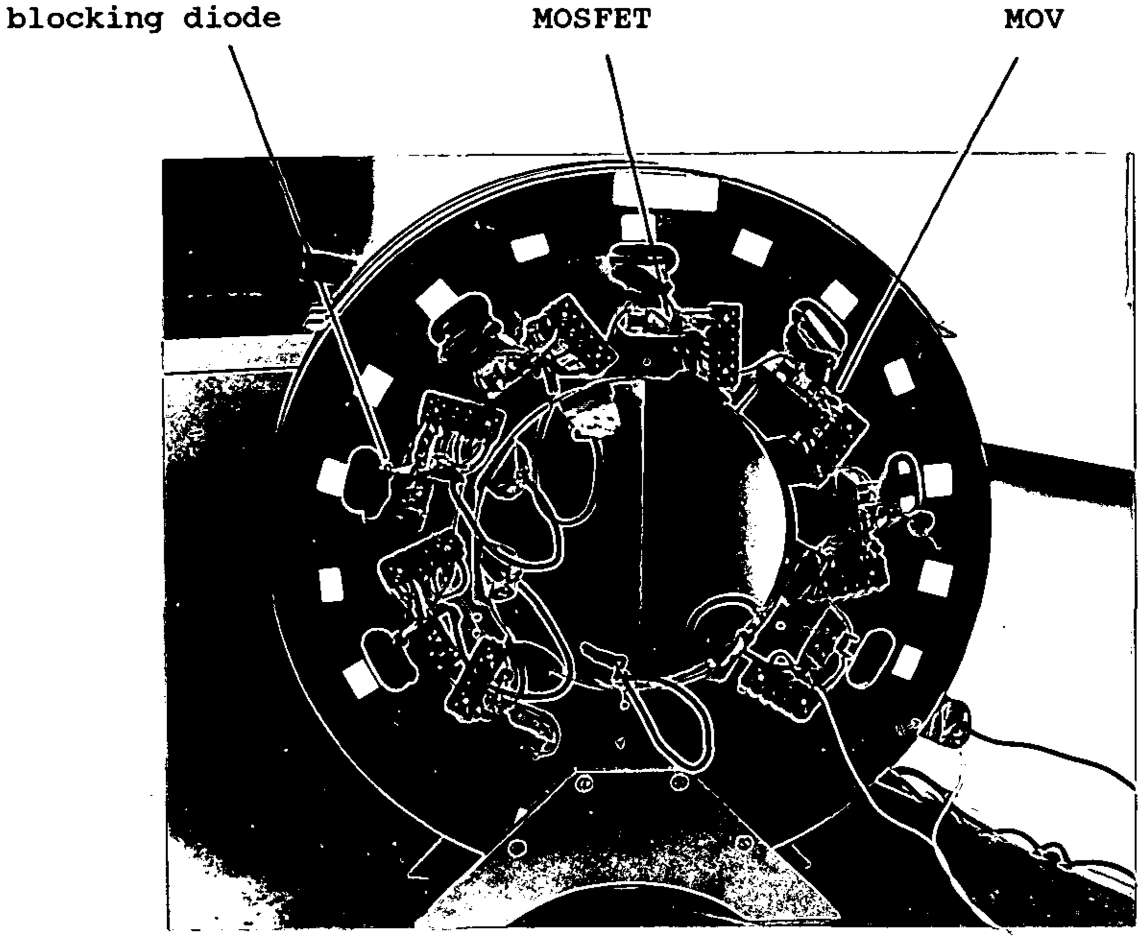


Figure 2.5(a) Completed Coil - Front

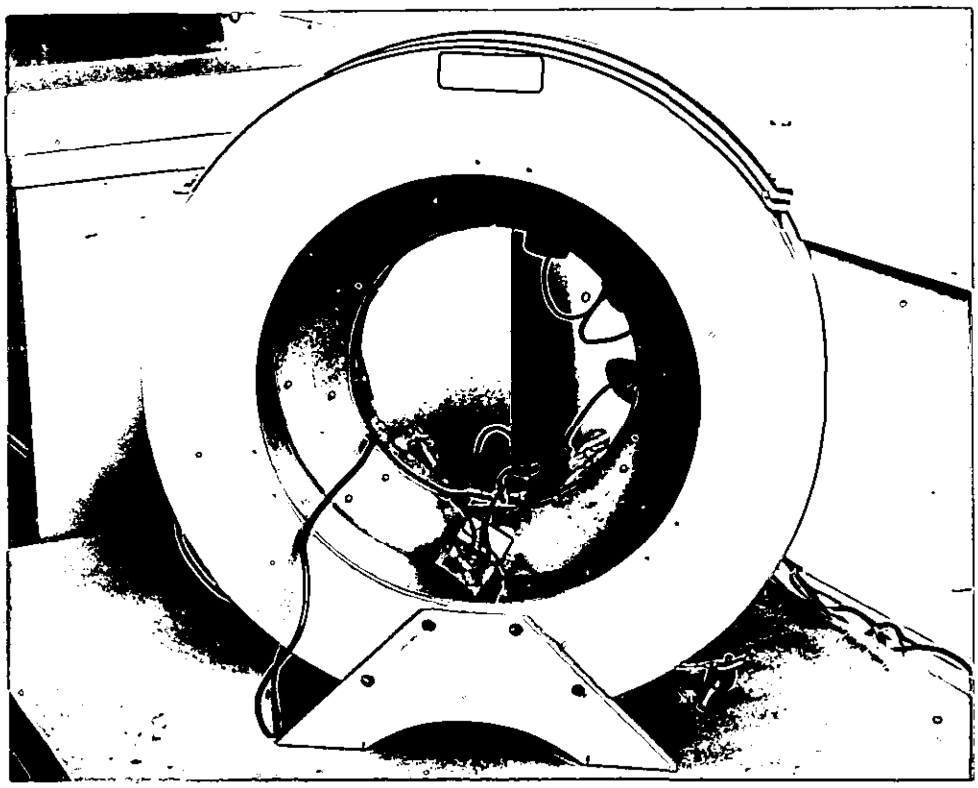


Figure 2.5(b) Completed Coil - Rear

2.5 MEASUREMENTS

Resistance measurements were made at d.c. and inductance measurements at 1kHz. Tables 2.4 to 2.8 show the results and the consequent changes in predicted performance. The method of construction used meant that the coil could not be wound very tightly, which explains the significantly lower coupling coefficients. Other figures are in reasonable agreement with the design values. (Because of the loose winding, only 22 turns were obtained on L_1 instead of the design figure of 25. The inductance is acceptable despite this, because of the increased radius.)

Table 2.4 shows that for the outer coils (which have the largest number of turns), the self-inductance is approximately proportional to the number of turns squared. For example, L_1 has 22 turns whilst L_2 has 20, giving an approximate inductance ratio of $22^2/20^2$=1.2 to 1, compared with the ratio of measured inductances of 403/289=1.4 to 1. Coil section L_3 follows the same approximate law, whereas for sections L_4 to L_7 the correlation is less good. This is probably due to a combination of measurement inaccuracy and the increasing significance of the factors a,b and K' in equation (2.4).

The coupling coefficients given in table 2.5 might be expected to fit approximately with the "ratio of areas" method of calculation described earlier. Since it was not possible to measure the radius of individual coil sections (because of the cheeks on the coil former), it is, however, difficult to verify this. It is clear, though, that the coupling coefficients with very low values (e.g. k_{16}=0.29, k_{17}=0.2) are significantly lower than predicted by even a rough calculation. This can be attributed to measurement inaccuracy (which leads to the need for the adjustments referred to in table 2.5) and the limitations of the "ratio of areas" method, which were described earlier. About half of the coupling coefficients fit with the "$k-k^2$" coupling coefficient model described by Giorgi et al [28].

Table 2.7 lists the inductance remaining in circuit after each energy transfer. It is this inductance, together with the inductance of the section to be switched out, which is used to calculate the current multiplication and efficiency of a meatgrinder switching step.

The performance figures given in table 2.8 show how the expected current multiplication and efficiency are degraded by the lower coupling coefficients.

403	289	168	84	63	6	40
289	289	173	88	57	13	26
168	173	147	78	48	10	33
84	88	78	52	32	5	22
63	57	48	32	24	4	15
6	13	10	5	4	1	2
40	26	33	22	15	2	11

Dotted lines separate inductors which are switched out (L_1 to L_6) from the coupled load L_7.

Table 2.4 Inductance Matrix (Measured Values in μH)

-	0.85	0.69	0.58	0.55	0.29	0.20
0.85	-	0.84	0.71	0.68	0.50	0.44
0.69	0.84	-	0.89	0.80	0.75	0.70
0.58	0.71	0.89	-	0.91	0.74	0.65
0.55	0.68	0.80	0.91	-	0.76	0.70
0.29	0.50	0.75	0.74	0.76	-	0.71
0.20	0.44	0.70	0.65	0.70	0.71	-

Note: Some of the values in this matrix had to be changed in order to make them physically consistent with the neighbouring figures. In other words, k_{12} must be greater than k_{13}, which must be greater than k_{14} and so on. The figures given are those used in computer simulations and give sufficiently accurate results. The inconsistencies are attributable to inaccuracy in measurements.

Table 2.5 Coupling Coefficient Matrix Derived From Table 2.4

$$\begin{bmatrix} 0.071 & & & & & & \\ & 0.054 & & & & & \\ & & 0.037 & & & & \\ & & & 0.021 & & & \\ & & & & 0.013 & & \\ & & & & & 0.004 & \\ & & & & & & 0.011 \end{bmatrix}$$

Table 2.6 Resistance Matrix (Values in Ohms)

STEP	INDUCTANCE REMAINING IN CIRCUIT (μH)
-	$L_{1-7} = 3390$
1	$L_{2-7} = 1721$
2	$L_{3-7} = 718$
3	$L_{4-7} = 244$
4	$L_{5-7} = 77$
5	$L_{67} = 18$
6	$L_{7} = 11$

Table 2.7 Inductance Remaining in Circuit at Each Step

PARAMETER	DESIGN VALUE	REVISED VALUE
I_1	5A	7A
I_2	8A	9A
I_3	12A	14A
I_4	20A	23A
I_5	37A	40A
I_6	80A	80A
I_7	100A	100A
η_1	98%	95%
η_2	98%	94%
η_3	98%	95%
η_4	98%	96%
η_5	98%	92%
η_6	99%	98%
η_t	89%	73%

Table 2.8 Revised Values for Expected Meatgrinder Performance Based on Measured Coil Parameters

CHAPTER THREE

CONTROL ELECTRONICS FOR THE SIX-STEP MEATGRINDER

3.1 INTRODUCTION

This Chapter describes the electronic components and circuits used to provide switching control for the six-step meatgrinder.

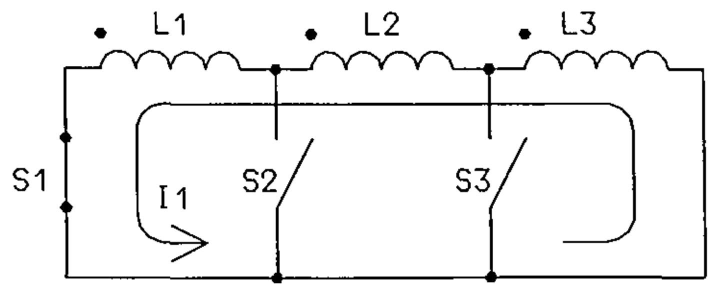
3.2 TIMING CIRCUIT

3.2.1 Specification

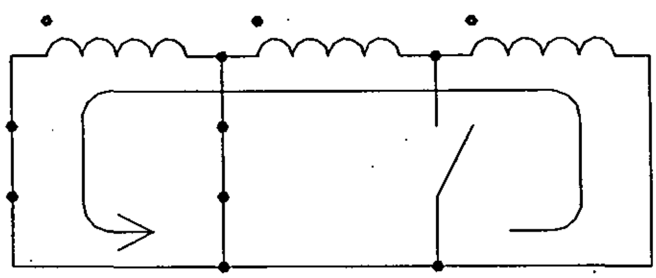
3.2.1.1 Principle

The operation of a two-step meatgrinder can be explained by reference to the circuit shown in figure 3.1, in which each step consists of a "make-before-break" switching sequence. The parameters to be specified are the timing of the "break" relative to the "make", and the timing of the start of one sequence relative to the end of the previous one.

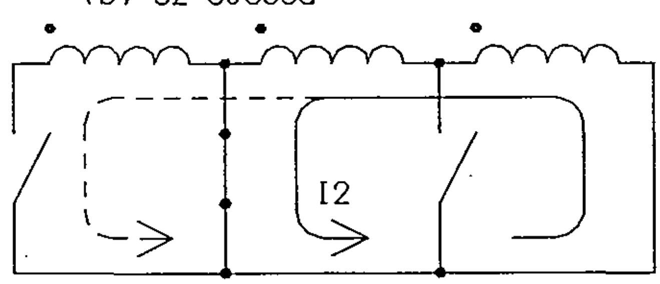
Figure 3.1(a) assumes that a constant current I_1 has been established, so that S_2 can be closed without affecting the circuit [20]. Once S_2 is closed, S_1 can be opened at any time, and since the objective is for the whole process to be rapid, S_1 should be opened with minimum delay.



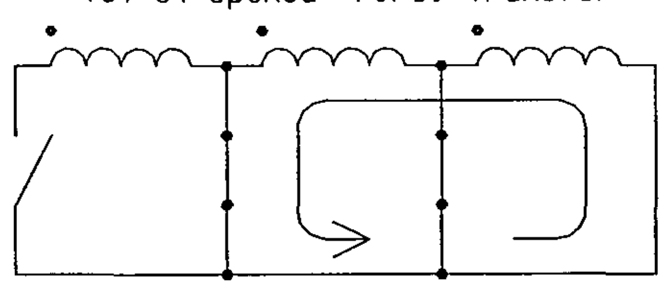
(a) Initial Current Established



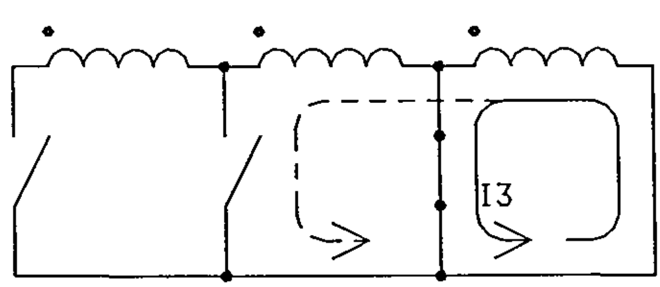
(b) S2 Closed



(c) S1 Opened: First Transfer



(d) S3 Closed



(e) S2 Opened: Second Transfer

Figure 3.1 Operation of Two-Step Meatgrinder

If S_3 is closed before the current in L_1 has fallen to zero, i.e. before the energy transfer from L_1 to L_2 and L_3 is complete, S_2 will open immediately afterwards to initiate the second energy transfer into the final loop. However, current in the largest loop (flowing through L_1, L_2 and L_3) will now also be forced to transfer to the final loop. Since this transfer involves a larger change of inductance it will be less efficient (see equation (2.3)) and so the final current in L_3 will be less than predicted.

In order to avoid this situation, S_3 should not be closed until the previous transfer is complete and the current I_2 has stabilised. There should again be no unnecessary delay, in order that the total energy delivery time is minimised. In addition, any such delay would result in an energy loss due to the resistance present in all practical circuits.

Figure 3.2 is the resultant timing diagram for the two-step circuit. The switching time t_{sw} is the same for both steps (assuming the same type of switch is used in each case). It is a fixed time, determined from the turn-on time of the switches, to ensure that the switch resistance is at its on-state value before the previous switch is opened.

The energy transfer time t_{delay} must be optimised as described above. For a circuit with n steps there are clearly n-1 delays to be optimised.

3.2.1.2 Charging

A practical meatgrinder circuit must initially be connected to a current source. Referring to figure 2.1, in which the source is labelled "Power Supply", the first task of the timing circuit is to close S_1 and to allow the current to build up to the required value before the first energy transfer is initiated. This process is called charging the coil.

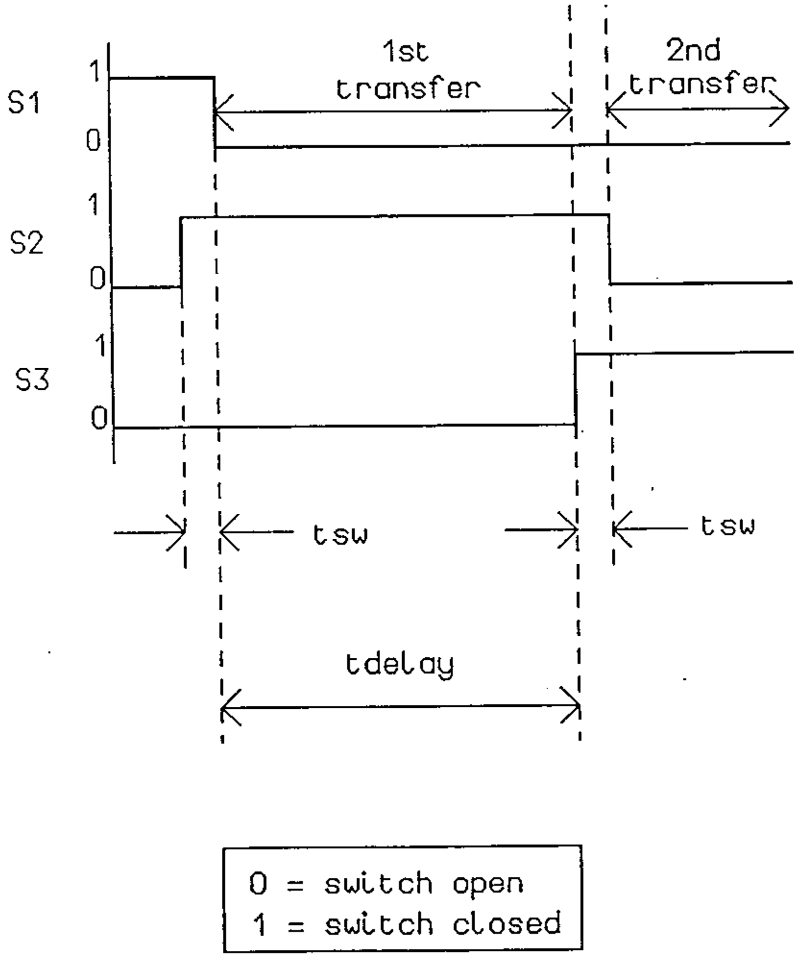


Figure 3.2 Timing Diagram - Two-Step Meatgrinder

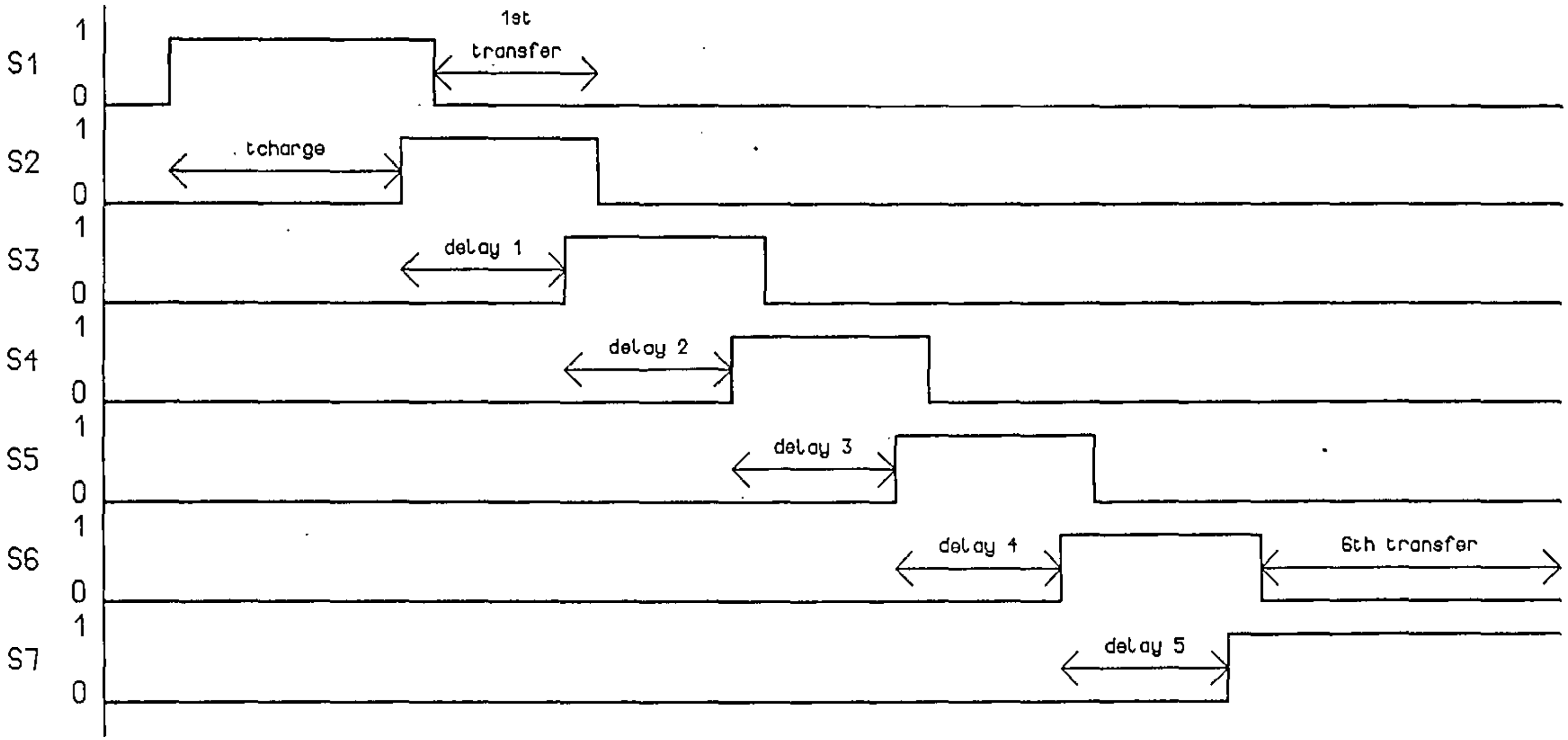
Note: It has been stated that switches should only be closed when the current is constant. For the first sequence, however, this is not necessary because the use of a blocking diode in series with S_2 (not shown in figure 2.1) prevents any unwanted current flow through S_2, even if this is closed whilst the current is still rising (see reference [27] and below).

Based on the above discussion, the timing diagram for a six-step meatgrinder is as shown in figure 3.3. The timing circuit must generate a two-state logic signal for each switch and also provide a means of manually varying the transfer delays so that they can be optimised.

3.2.2 Description of Operation

The physical layout of the timing circuit is shown in figure 3.4. The main board is on the left and the delay extension board on the right. The circuit is supplied from a 10V d.c. regulated power supply and the isolated logic signal outputs are fed out via the ribbon cable connector on the lower edge of the main board.

A detailed circuit description is given in Appendix B; the following description is based on the block diagram of figure 3.5.



NOTE: DELAYS ARE VARIABLE

0 = switch open
1 = switch closed

Figure 3.3 Timing Diagram - Six-Step Meatgrinder

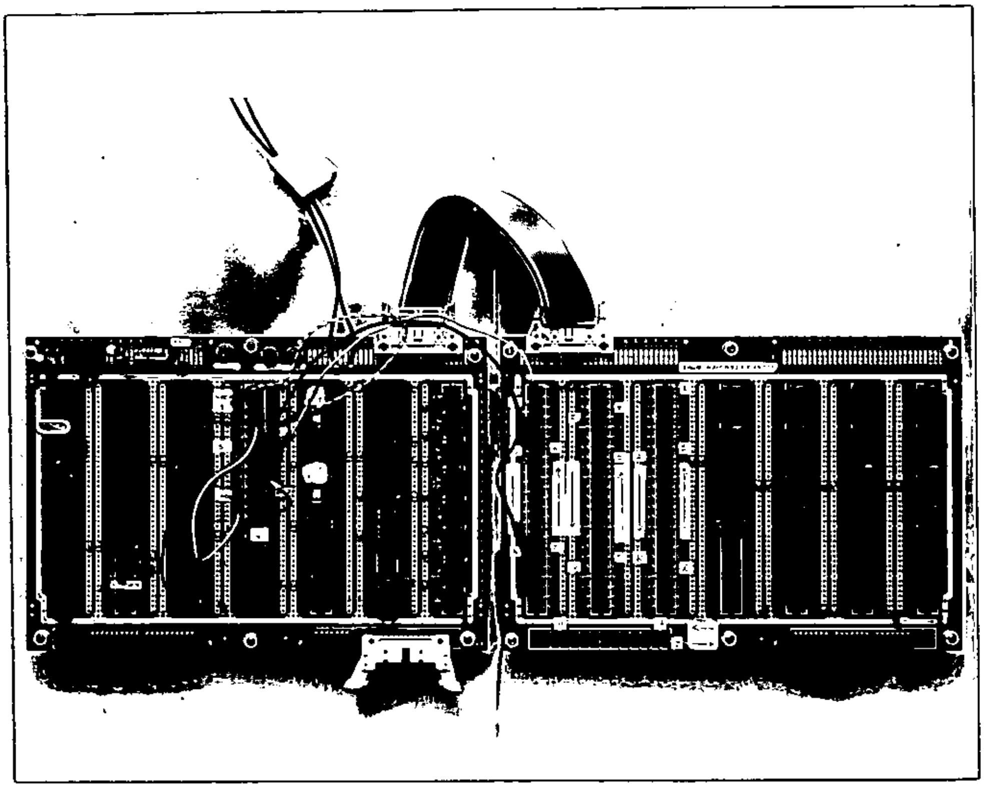


Figure 3.4 Timing Circuit

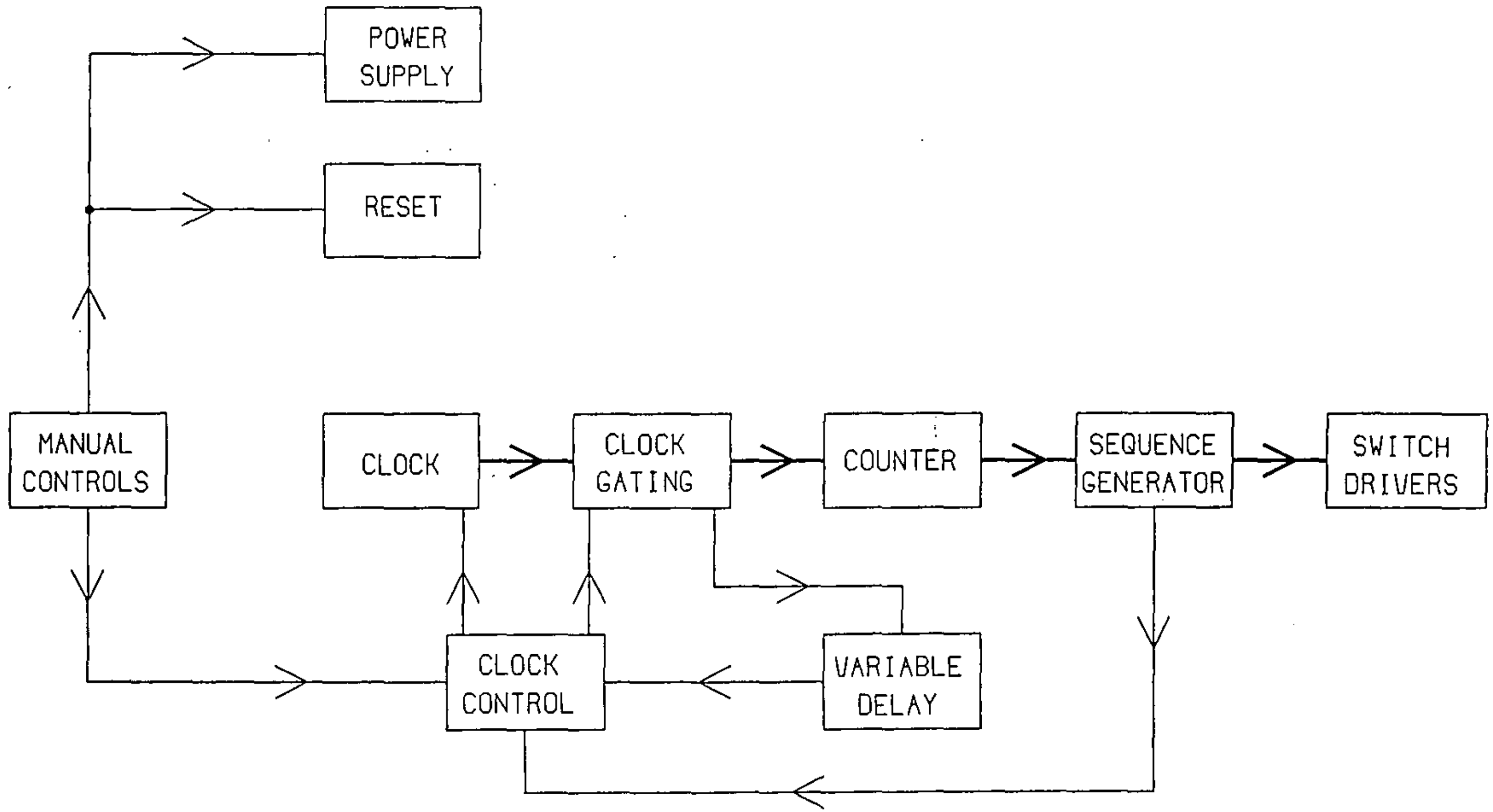


FIGURE 3.5 TIMING CIRCUIT BLOCK DIAGRAM

3.2.2.1 Manual Controls

The manual controls can be seen at the upper edge of the main board in figure 3.4. They are:

(1) A toggle switch to apply power to the circuit.
(2) A Reset pushbutton to initialise the circuit.
(3) A Run pushbutton which initiates the output sequence by turning on the clock via the Clock Control function.

3.2.2.2 Main Signal Path

The main signal path through the block diagram is indicated by the heavy lines in figure 3.5.

Normally the Clock Gating function feeds clock pulses to the Counter. The counter output is fed to the Sequence Generator, which in turn produces the required set of two-state logic signals. The state of each signal corresponds to the switching requirements in the power circuit at that particular time.

When the sequence is complete the Sequence Generator stops the clock via the Clock Control function, with the output signals being frozen in their last state.

3.2.2.3 Variable Delays

Whenever a variable delay is needed the Sequence Generator sends a signal to the Clock Control function. This causes the Clock Gating function to remove the clock pulses from the Counter and to reroute them to the Variable Delay function. The counter then stops counting, thus preventing any further change in the state of the output signals.

The delay ends when the desired number of clock pulses has been received by the Variable Delay function. This number is determined by the position of the flying leads visible in figure 3.4, there being one lead for each delay. After the delay, clock pulses are returned to the Counter and the sequence continues.

3.3 POWER CIRCUIT COMPONENTS

The term "power circuit" is used to distinguish between the actual meatgrinder circuit (as shown in figure 3.6) and the control circuits also described in this Chapter. This section describes the components present in the circuit shown in figure 3.6, other than the coil itself.

3.3.1 Switching Devices

Since the experimental meatgrinder carries relatively low currents (under 100A), switching can be carried out by semiconductor devices. The devices considered were MOSFETs, GTOs and bipolar transistors (BJTs). Each has advantages and disadvantages [40-42], but the MOSFET was chosen for its simple

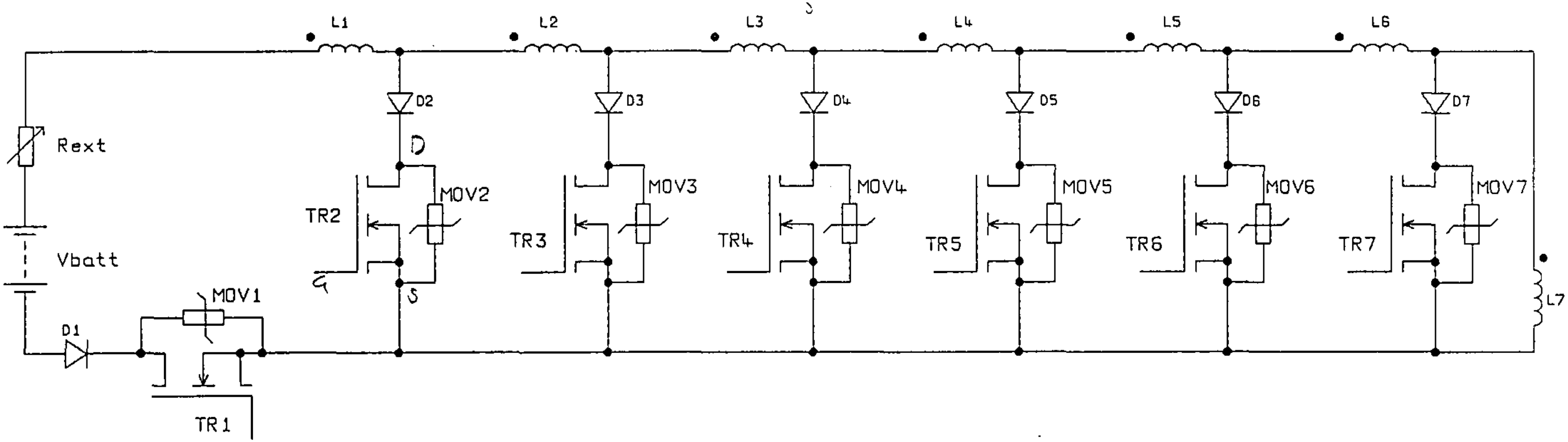


Figure 3.6 Circuit Diagram - Six-Step Meatgrinder

drive requirements, its high switching speed and its ability to work up to the surge current rating without incurring penalties in terms of drive power or on-state resistance.

Although the power level in the six-step meatgrinder is very low, it has been suggested [43] that MOSFETs could be used to obtain an output pulse of up to 750kW from a meatgrinder. Santamaria and Ness [44] have in fact carried out tests on MOSFET arrays designed to switch 700A at 6kV.

3.3.2 Voltage Clamp Devices

A MOSFET is likely to be damaged if its drain-to-source voltage exceeds the breakdown value. Protection against this must normally be provided by an external component, although the latest generation of International Rectifier devices has built-in protection which can be relied upon up to the continuous current rating [45].

Following the example of Giorgi et al [27], metal oxide varistors (MOVs - also known as constant voltage resistors or CVRs) are used where external protection is needed. These are rugged devices available in a wide variety of ratings, although when used at very high currents they can present an explosion hazard [46]. Selection is on the basis of the maximum current and the speed at which current is diverted from the MOSFET into the MOV [47].

Figure 3.7 compares the response of an ideal clamp device with that of a typical MOV (Power Development Z320C). It can be seen that although the voltage pulse is not flat-topped, the current decay is still approximately linear. The main problem is the initial overshoot, which is a function of the rate of current rise in the MOV (i.e. the turn-off time of the protected device) and any stray inductance present in the MOV leads. Devices must be selected conservatively, because manufacturers' data on overshoot does not account for the sub-microsecond turn-off times produced by a MOSFET.

There is an alternative device known as a transient voltage suppressor (TVS). This is a special-purpose zener diode which has a flatter clamping characteristic than the MOV and is not so prone to overshoot [48]. It is recommended for semiconductor protection but in this case no devices with suitable voltage ratings were available.

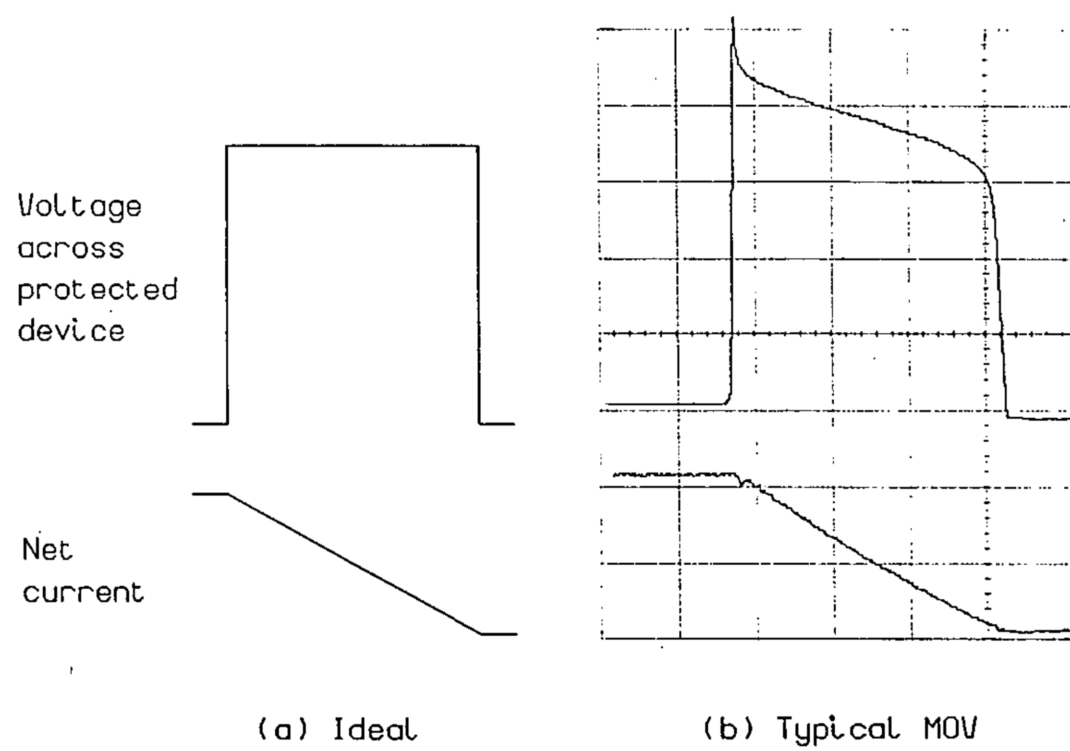


Figure 3.7 MOV Clamping Characteristics

3.3.3 Blocking Diodes

Inherent in the construction of a power MOSFET is the anti-parallel drain-soure diode depicted in figure 3.8. This means that the blocking diodes D_2 to D_7 in figure 3.6 are necessary to prevent the power supply being shorted out during charging. Diode D_1 is simply to protect against reverse connection of the power supply.

During charging, when the coil current is increasing, diodes D_2 to D_7 are reverse biased. This ensures that turning on TR2 does not have any effect on the circuit, the current path only changing when TR1 turns off.

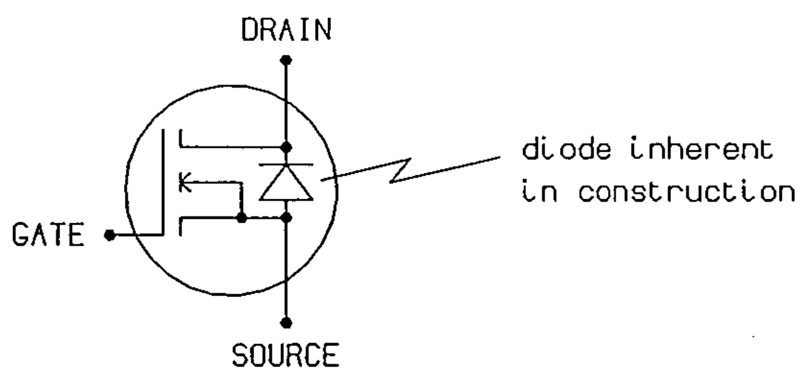


Figure 3.8 Internal Diode in MOSFET

The reverse voltage across any one of the diodes D_2 to D_7 during charging is, of course, a proportion of the power supply voltage, with the highest proportion being across D_2. The diodes need to be rated accordingly. The device used for all seven blocking diodes is the Motorola type MR752, which has a reverse voltage rating of 200V and a non-repetitive surge current rating of 400A.

3.3.4 Power Supply

The power supply consists of two standard 12V car batteries connected in series. The use of these batteries in preference to a mains-powered unit reduces the noise level in the experimental voltage and current waveforms obtained from various parts of the meatgrinder circuit.

The time allowed for charging is programmed into the timing circuit (see Appendix B). The current level at the end of this time can be adjusted by means of the 10Ω potentiometer R_{ext}.

3.3.5 Mounting

The MOSFETs are mounted on steel brackets attached to the coil (see figure 2.6). The MOVs are connected directly across the MOSFETs so as to minimise stray inductance. The blocking diodes can also be seen in figure 2.6, connected between the MOSFETs and the coil.

3.4 INTERFACE CIRCUIT

3.4.1 Specification

An interface circuit is needed to convert the logic signals from the timing circuit into signals suitable for driving the MOSFETs in the power circuit. The MOSFET drive signal voltage must be sufficiently high to keep the on-state resistance down when drain current flows. Current capability is also important, since the rate at which the MOSFET capacitances are charged or discharged affects the switching speed [49,50].

3.4.2 Description of Operation

The physical layout of the interface circuit is shown in figure 3.9. The supply required is 20.5V d.c. (see Appendix B). Input and output connections are via the right- and left-hand ribbon cable connectors respectively.

Figure 3.10 represents a single channel of the circuit. The MOSFET driver is a special-purpose integrated circuit (IC) which draws current from the supply V_{drive}. A complete circuit diagram and further details are given in Appendix B.

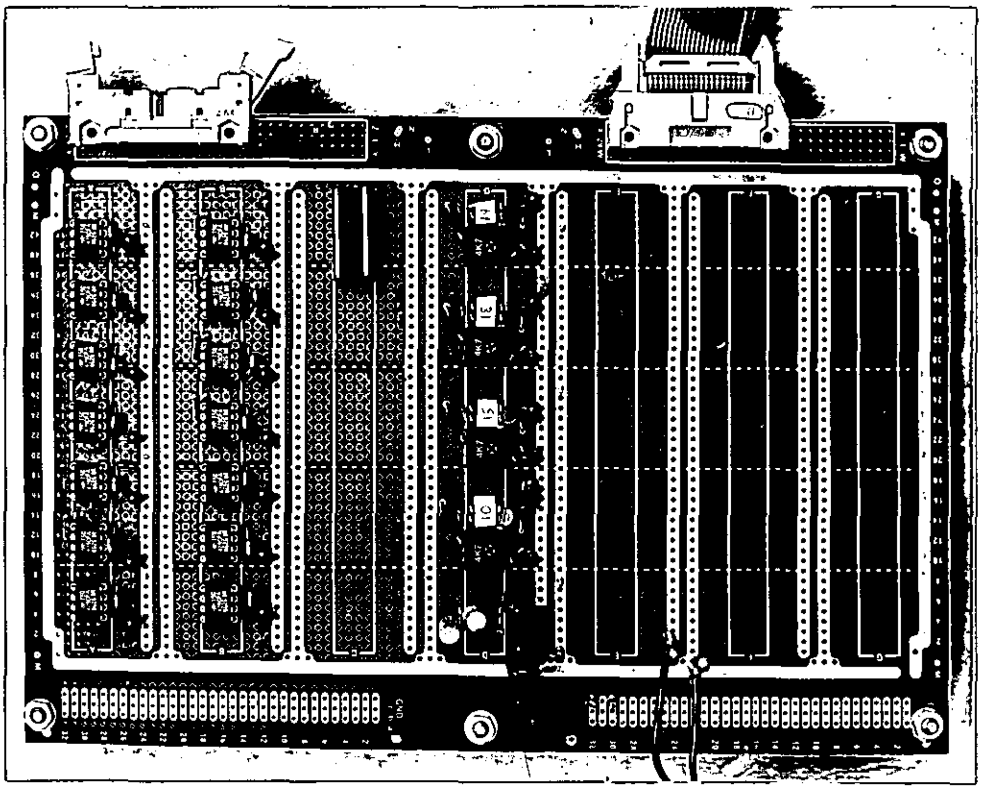


Figure 3.9 Interface Circuit

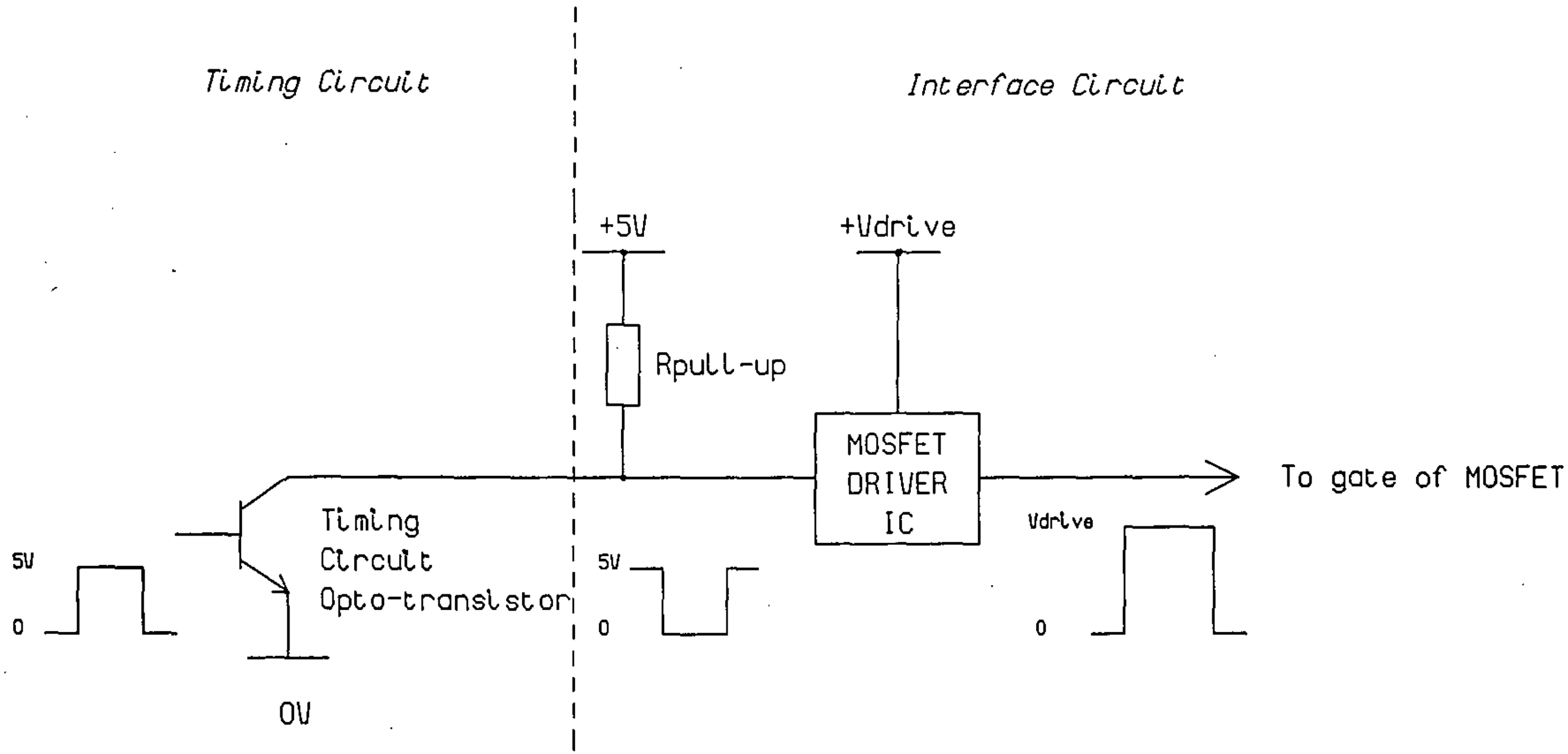


Figure 3.10 Interface Circuit Functional Diagram

CHAPTER FOUR

DISCUSSION OF EXPERIMENTAL RESULTS FROM THE SIX-STEP MEATGRINDER

4.1 EXPERIMENTAL PROCEDURE

4.1.1 Apparatus

Figure 4.1 shows the experimental apparatus. From left to right it comprises:

(a) the control electronics and d.c. supplies (see Chapter 3 and Appendix B)
(b) the meatgrinder coil with the switching components mounted on it (see Chapters 2 and 3)
(c) the Tektronix A6302 current probe (used in conjunction with a x20 multiplier to prevent saturation)
(d) the two 12V batteries
(e) the Gould 1604 20 megasample/second digital storage oscilloscope.

4.1.2 Steps Taken to Reduce Noise

In this context, "noise" refers to any unwanted disturbance on the circuit waveforms. It usually takes the form of spikes or oscillations and is caused by current flow in parasitic (stray) capacitances or pick-up by parasitic inductances [51,52].

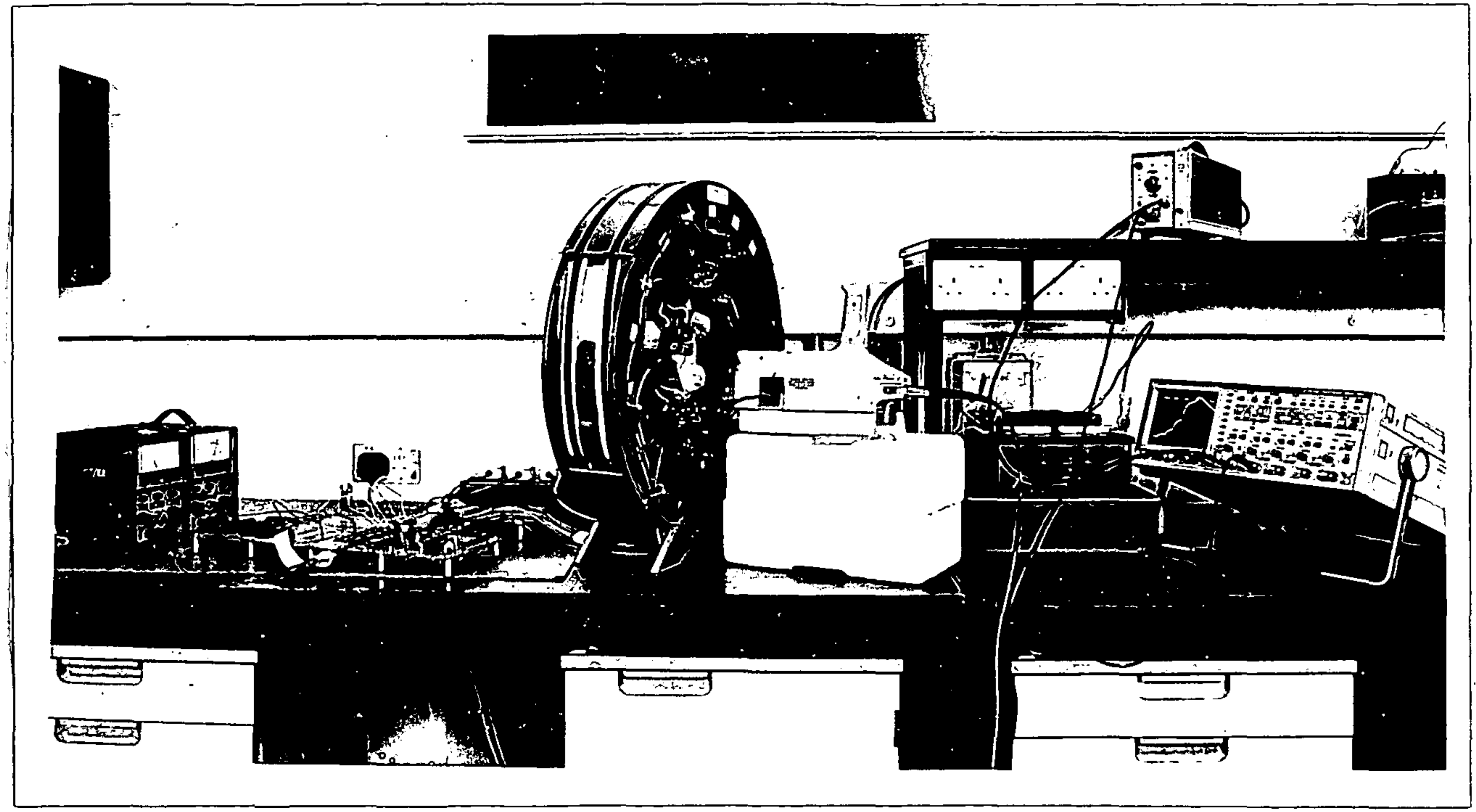
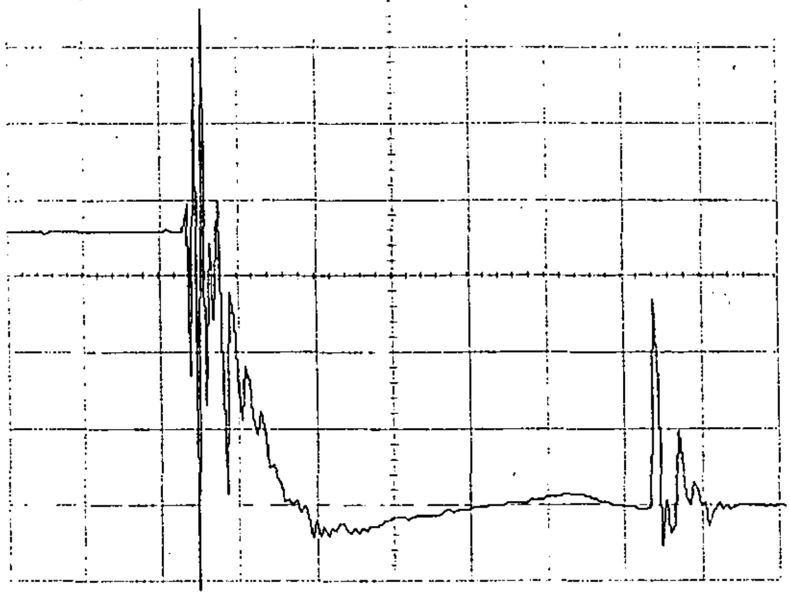


Figure 4.1 Experimental Apparatus

Any experimental results obtained are affected by whether or not the ground end of the coil is earthed. Generally this is beneficial, though not consistently so. Specifically, the results change when the apparatus is moved to a different laboratory, due presumably to differences in the impedances of the mains earth in the different laboratories.

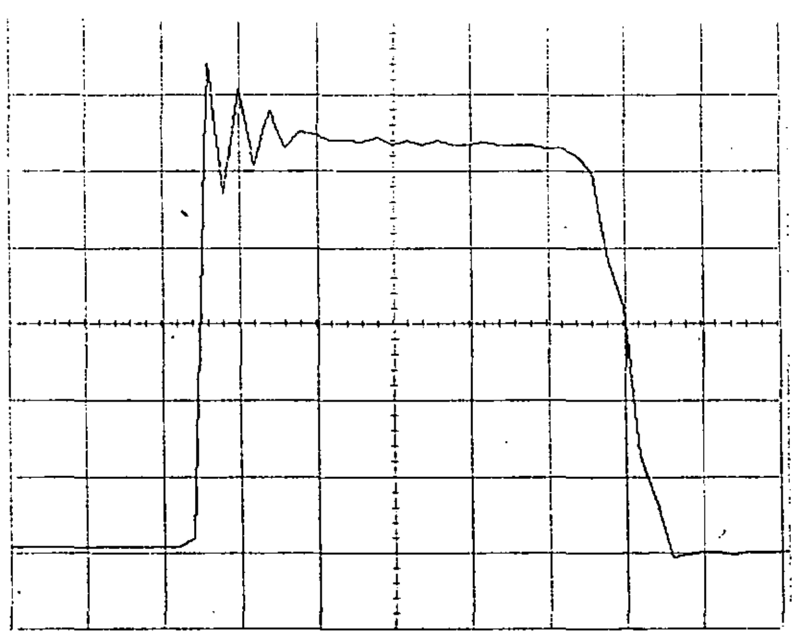
A specific example of the noise generated by parasitic components is shown in figures 4.2 to 4.4, which all relate to the turn-off of current in TR1 (see figure 3.6) at the beginning of the switching sequence. Figure 4.2 shows the current waveform with the gate drive signal connected directly to the gate terminal of TR1. Figure 4.3 is the corresponding drain-source voltage shown on an expanded time scale to illustrate the very rapid rising edge. Figure 4.4 shows the improvement obtained in the noise level when the turn-off is slowed down by the insertion of a 100Ω resistor in series with the gate of TR1. By restricting the rate-of-change of voltage experienced by parasitic capacitors and the rate-of-change of current experienced by parasitic inductors, the noise level may be substantially reduced.

Although each MOSFET requires a separate drive signal, only a single ground return conductor is used. This prevents ground loops [51,52] which can generate large circulating currents, and seriously disturb the circuit operation.



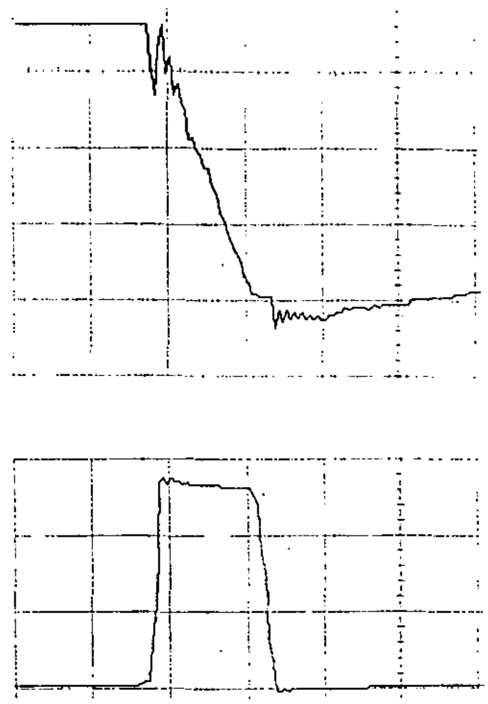
Current 2A/div   Time 2μs/div

Figure 4.2 Noise Generated at Turn-Off of TR1 Current



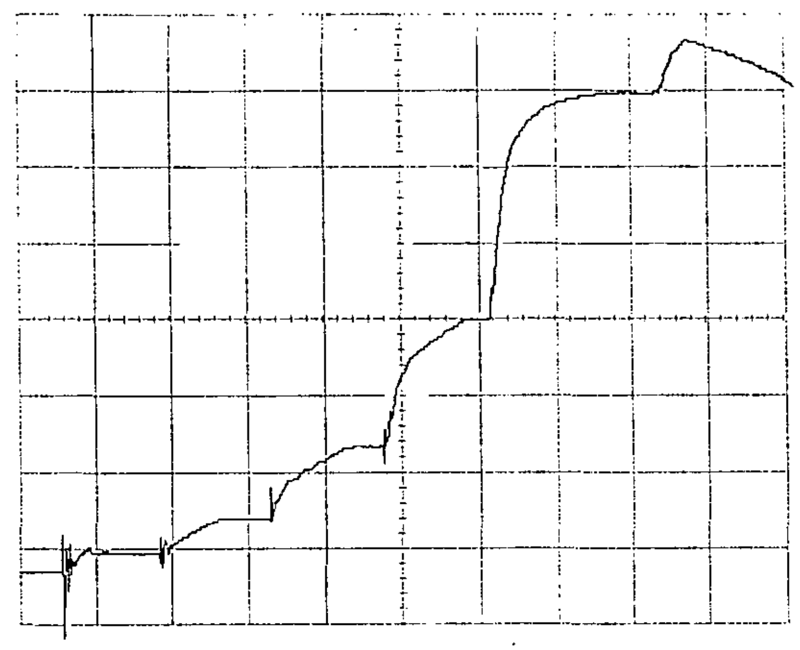
Voltage 100V/div   Time 0.5μs/div

Figure 4.3 Corresponding Voltage Across TR1 at Turn-Off



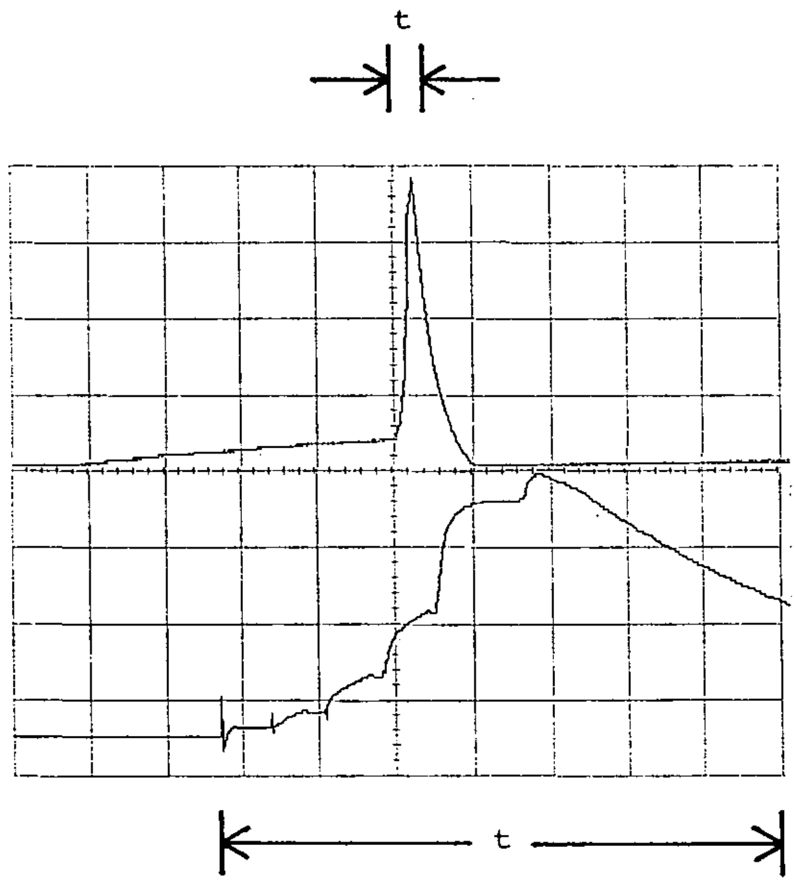
Top: Current 2A/div Time 2μs/div
Bottom: Voltage 200V/div Time 2μs/div

Figure 4.4 Reduction of Noise by Slowing Down Turn-off of TR1



Current 10A/div   Time 10μs/div

Figure 4.5 Current in L_7 Showing Current Multiplication



Top: Current 20A/div Time 400μs/div
Bottom Current 20A/div Time 20μs/div

Figure 4.6 Current in L_7 Showing Pulse Compression

4.2 VERIFICATION OF MEATGRINDER PRINCIPLE

The results shown in figures 4.5 and 4.6 confirm that as each coil section is switched out, the current in the remaining inductance increases, energy having been transferred forwards into fewer coil sections via the mutual inductance.

Figure 4.5 confirms that the current in the final coil section is over ten times greater than the input current to the meatgrinder. The final current (approximately 76A) and the corresponding efficiency (approximately 42%) are lower than predicted, as is discussed in detail in the next section.

Pulse compression is also achieved, as shown by figure 4.6. The input current is raised to 7A (corresponding to 83mJ of stored energy) in 1.7ms, giving an input power of about 49W. Of this initial energy, less than 1mJ is stored in the self inductance of L_7. After the meatgrinder switching, the circuit energy (about 32mJ) is stored entirely in the self inductance of L_7. An energy increase of 31mJ in about 80μs corresponds to an output power of about 387W.

4.3 EFFICIENCY

4.3.1 Comparison With Predicted Performance

From table 4.1 it can be seen that the current and efficiency figures for the experimental meatgrinder deviate from the predicted values only in the last two steps. As described below, further investigation has shown that the cause of this is circuit resistance, the chief component of which is the on-state resistance of the MOSFETs.

STEP	CURRENT (A)		EFFICIENCY (%)	
	PREDICTED	MEASURED	PREDICTED	MEASURED
0	7	7	-	-
1	9	9	95	95
2	14	14	94	94
3	23	23	95	95
4	40	40	96	96
5	80	70	92	70
6	100	76	98	74
			TOTAL 73%	TOTAL 42%

Table 4.1 Performance of Six-Step Meatgrinder

4.3.2 Effect of Resistance

4.3.2.1 Computer Simulation

Much benefit was gained in this research from computer simulation of the meatgrinder circuit. Appendix C describes the "PSpice" simulation package used for this purpose and provides an overview of how results were obtained.

Simulated current waveforms are identified by the component through which the current passes. I(D6), for example, means the current through D6. Simulated voltage waveforms are identified by a node number, so that V(22) means the voltage at node 22 with reference to ground (node zero). Reference should be made to figures C.1 and C.2 as necessary.

(i) Existing Circuit

In order that the simulation results can be relied upon, the first step taken is to model the circuit as built. This process produces the simulated current waveform of figure 4.7, which corresponds to the experimental waveform of figure 4.5. Figures 4.8 and 4.9 are further examples of current waveforms in the circuit which show that predictions produced by the model are reasonably accurate.

The model includes coil capacitance, although the effect of this is not significant. It is represented as a lumped component connected across each coil section, with the values used being simply an estimation of what the true values might be. Adding these capacitances does not significantly affect the current waveforms, although the simulation of the high-frequency ringing in some of the voltage waveforms is improved slightly, as is clear from figure 4.10. Evidently the inclusion of the coil capacitance is not a critical factor in this case.

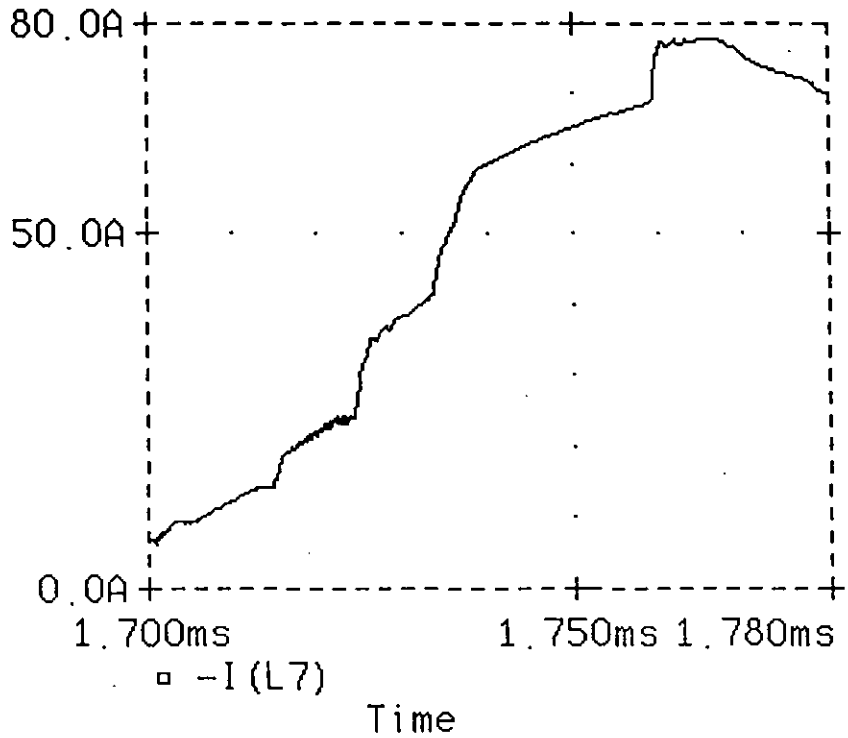
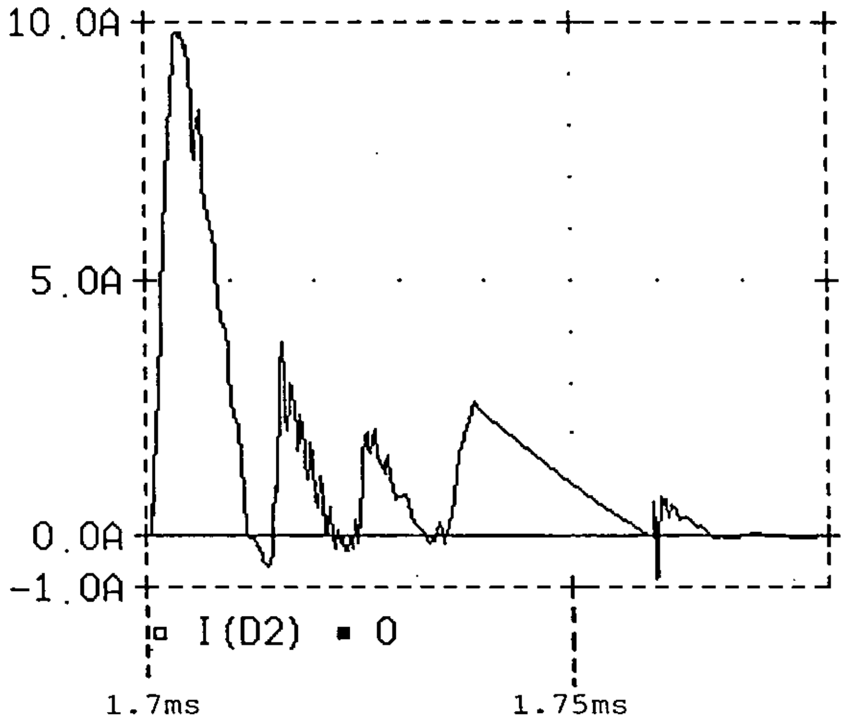
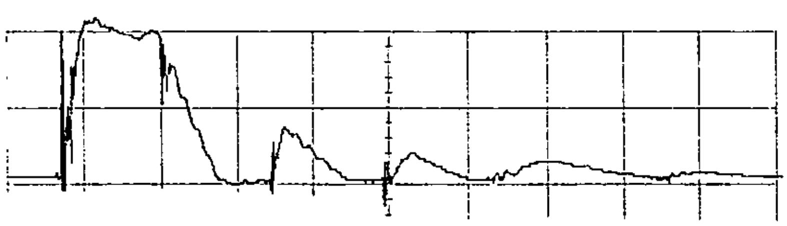


Figure 4.7 Simulated L_7 Current Waveform



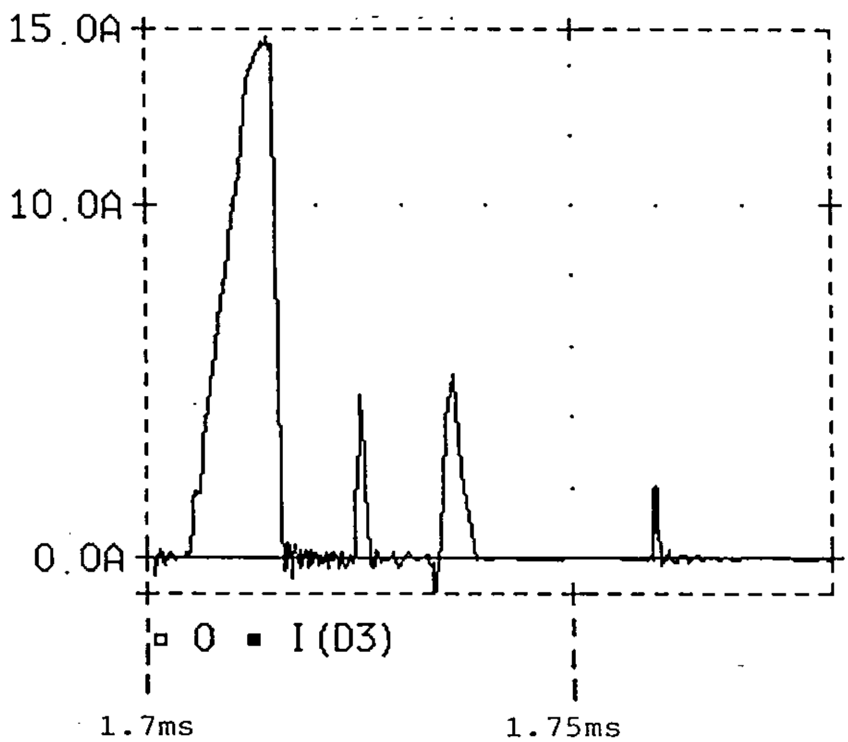
(a) Simulated



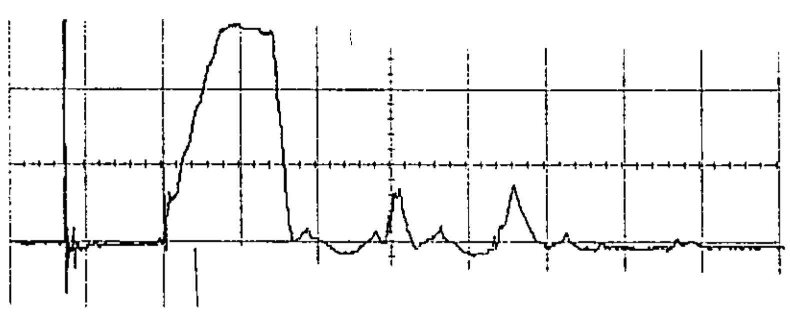
Current 5A/div   Time 10μs/div

(b) Experimental

Figure 4.8 Current in D2



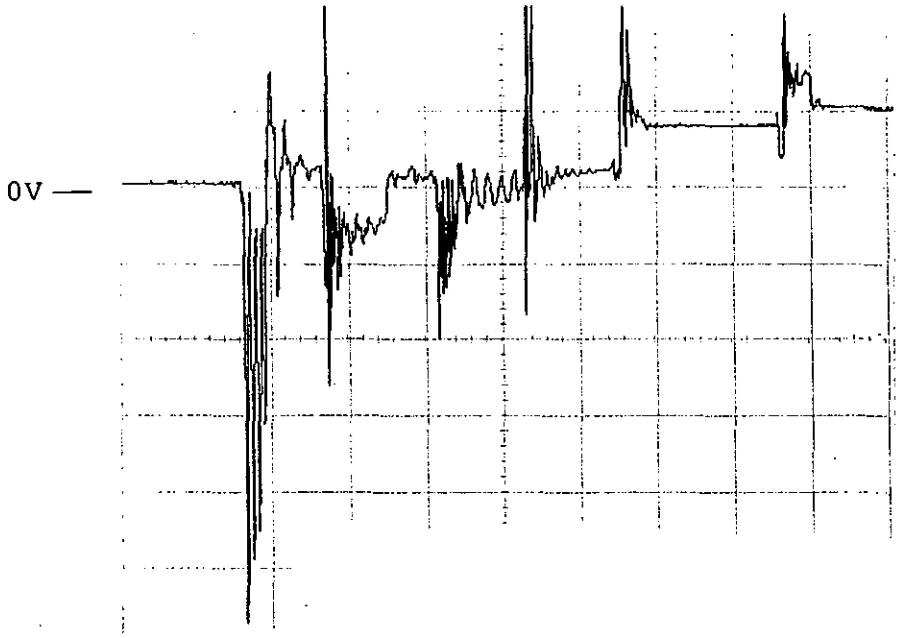
(a) Simulated



Current 5A/div   Time 10μs/div

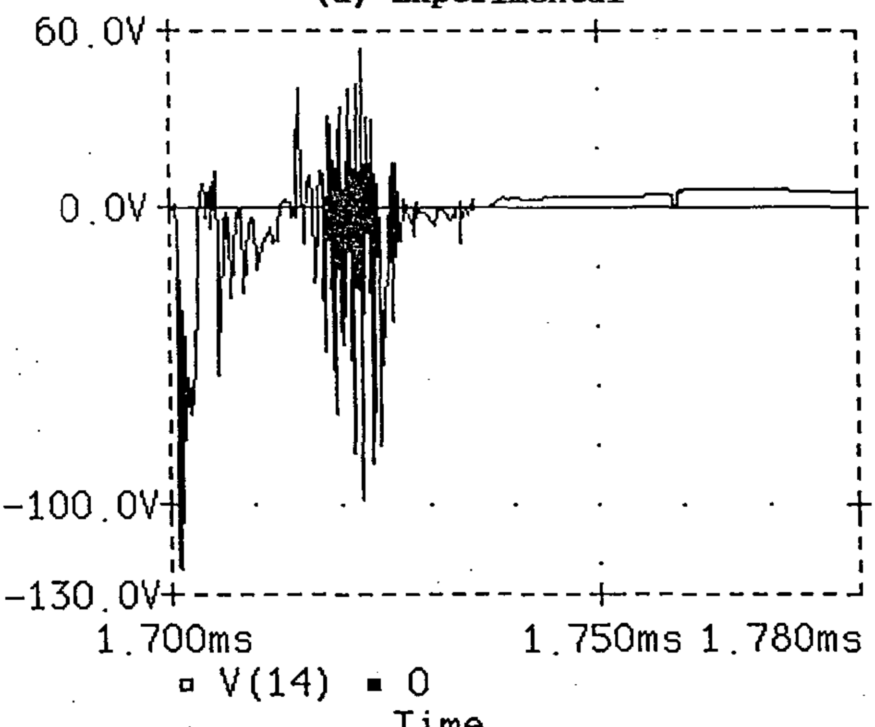
(b) Experimental

Figure 4.9 Current in D3



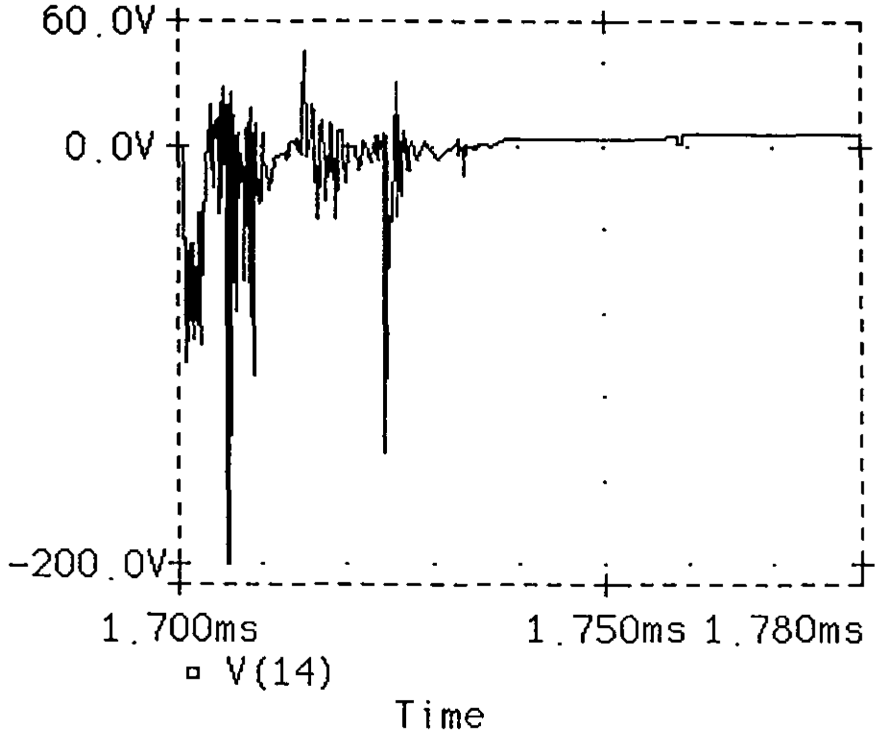
Voltage 10V/div   Time 10μs/div

(a) Experimental



(b) Simulated, Including Coil Capacitance

Figure 4.10 Voltage Across L_7



(c) Simulated, Excluding Coil Capacitance

Figure 4.10 Voltage Across L_7

It was found that the inclusion of the parasitic inductance of the MOSFET and clamp device leads had no discernible effect on the results obtained, but it did significantly increase the simulation run-time.

(ii) Low Resistance Circuit

Figure 4.11 is a simulated result which illustrates the effect of reducing the coil resistance to a negligibly small value. The peak currents at the fifth and sixth steps are increased, but are still not as high as their predicted values. This result may be contrasted with that of figure 4.12, in which the current rises to over 94A - within a few percent of the predicted value of 100A. The waveforms in figure 4.12 are from the idealised circuit model, with no coil resistance and with the transistors replaced by simple switches with extremely low on-state resistance.

It should be noted that the 100A figure itself is purely a nominal figure. The computer program described in Appendix E, which implements the theoretical expression for current multiplication, gives a final current of about 96A.

4.3.2.2 Effect of Reducing Initial Current

Table 4.2 and figure 4.13 show how the current multiplication can be increased by a reduction in the initial current. This phenomenon is a further pointer to the effect of resistance in the circuit and is explained below.

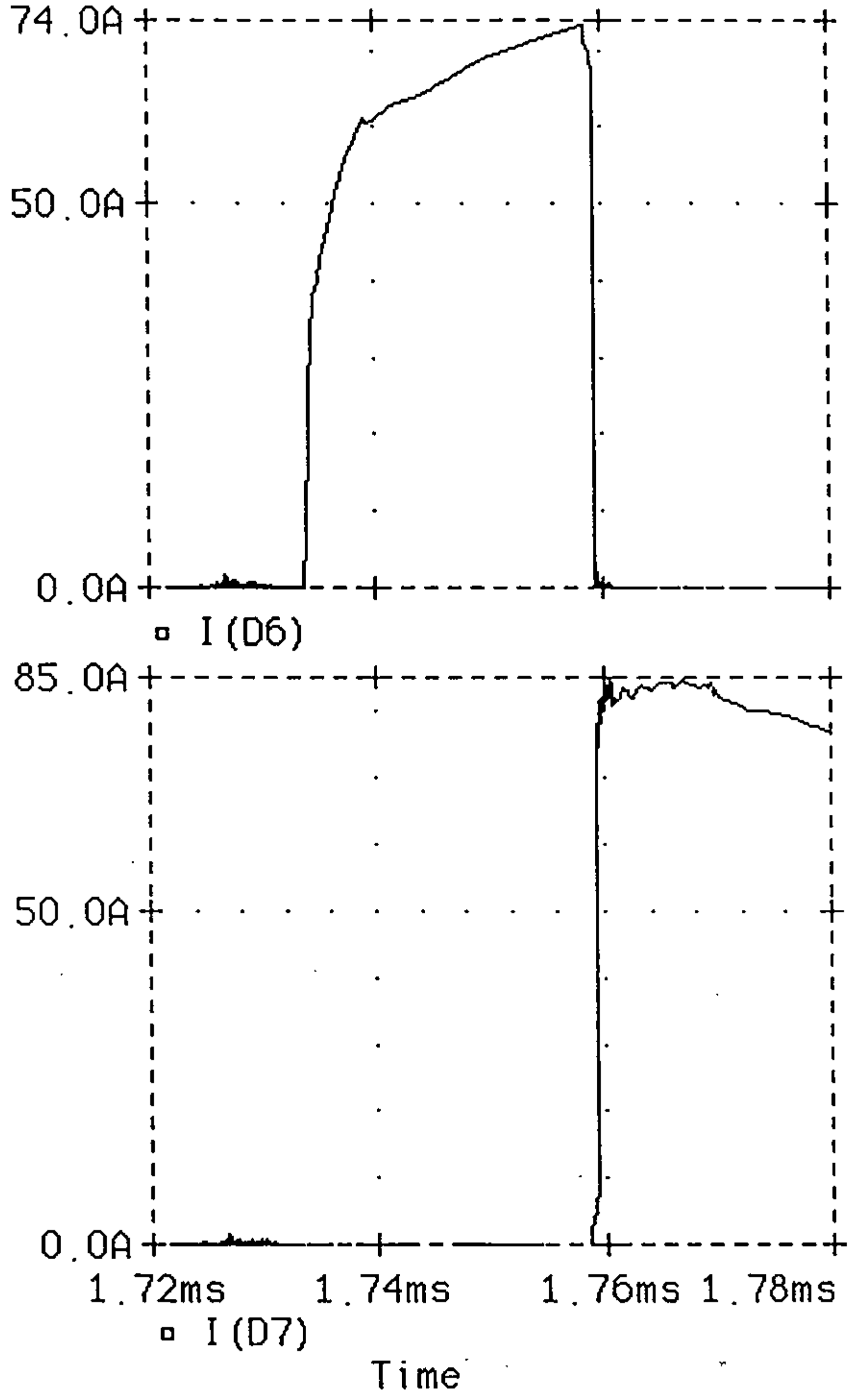


Figure 4.11 Fifth and Sixth Step Currents With Low Coil Resistance

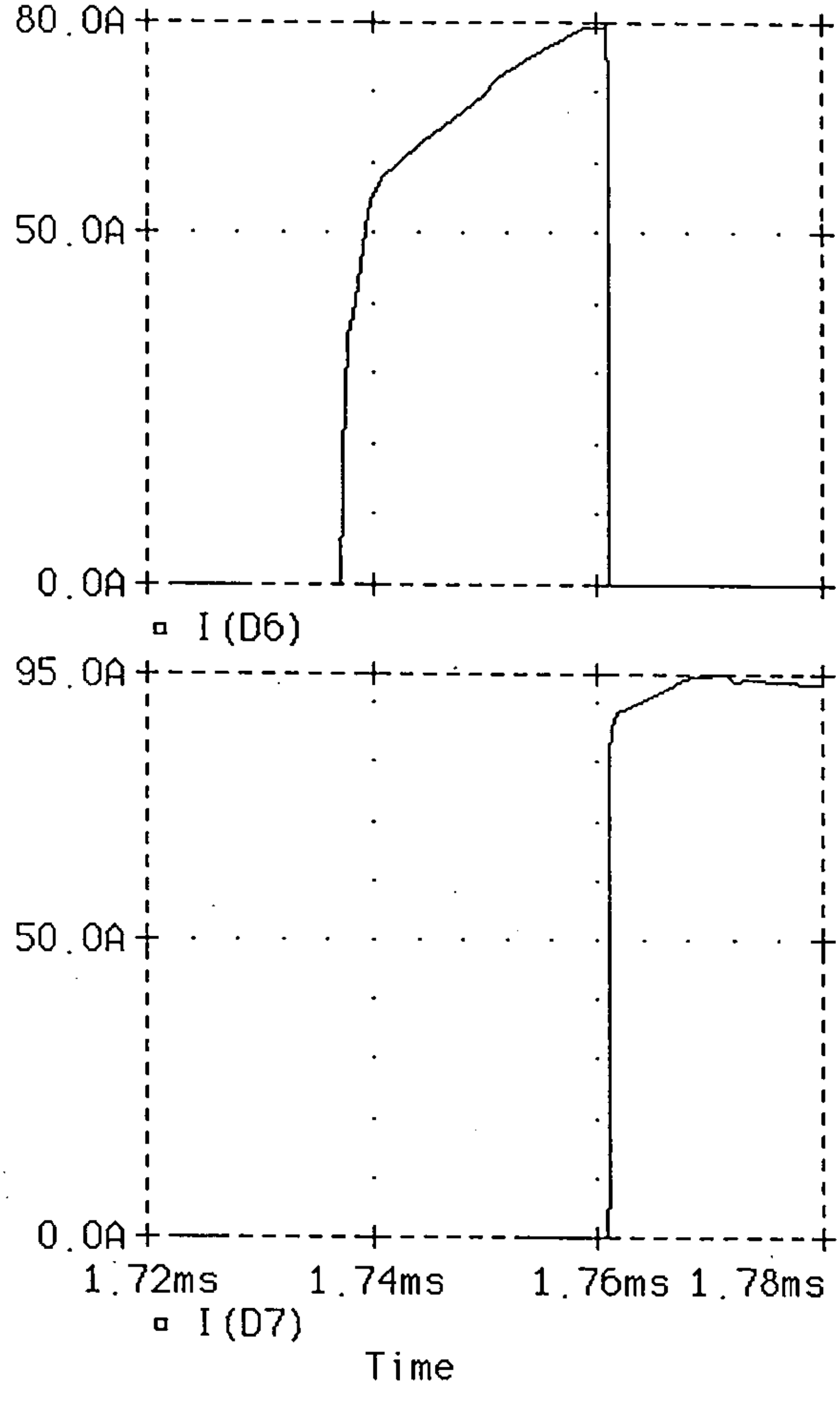
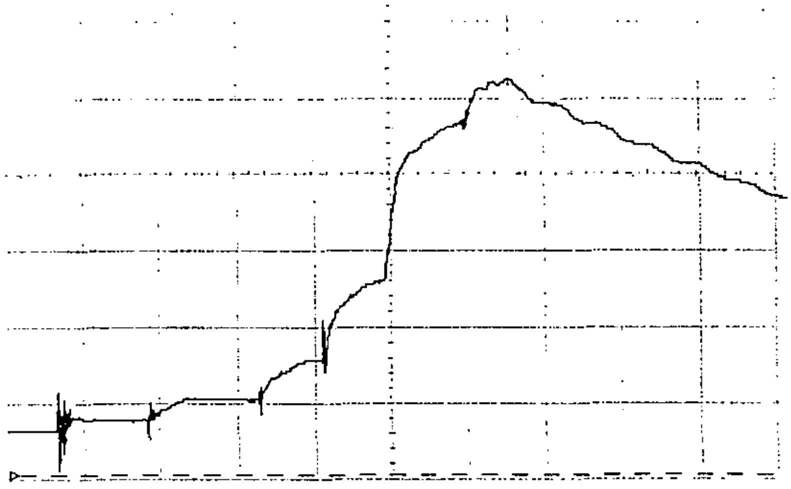
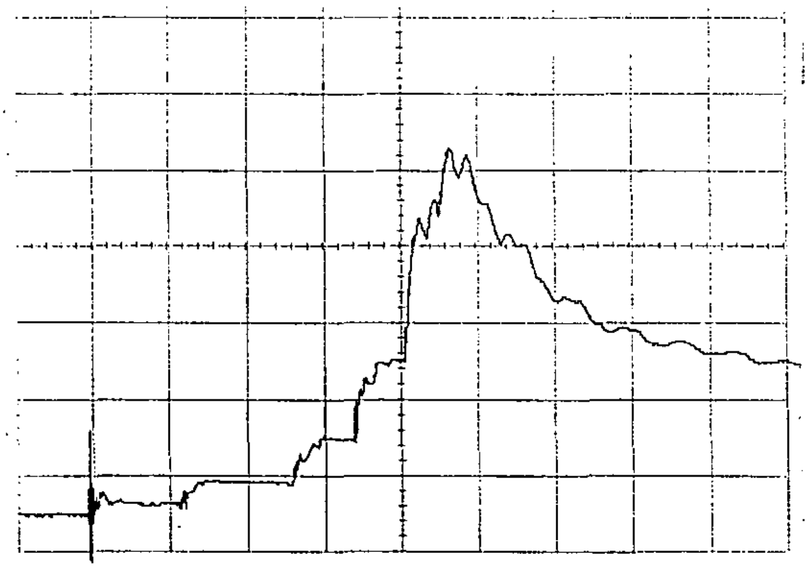


Figure 4.12 Fifth and Sixth Step Currents - Idealised Circuit



Current 10A/div   Time 10μs/div

(a) Initial Current 4A



Current 5A/div   Time 10μs/div

(b) Initial Current 2A

Figure 4.13 Increased Current Multiplication With Lower Initial Current

INITIAL CURRENT (A)	TOTAL CURRENT MULTIPLICATION	TOTAL TRANSFER TIME (μs)
7	10.9	80
4	13.0	60
2	13.5	48

Theoretical current multiplication = 14.3

Table 4.2 Effect of Changing Initial Current

The energy transfer times in the experimental meatgrinder were not minimised, but there is clearly a trend for the process to speed up as the initial current falls. This arises because less time is required to transfer a smaller amount of energy for a given set of switch voltages; this subject is also discussed later in the Chapter.

4.3.2.3 Conclusion

The various simulation and experimental results discussed above show that the current multiplication is degraded by any resistance in the circuit. In this particular instance the effect is apparent only in the last two steps, and it is the MOSFET on-state resistance which is dominant.

In the single-step meatgrinder circuit shown in figure 4.14, the resistor R_{L2} dissipates energy continuously during the transfer of energy (R_{L1} simply adds to the off-state resistance of the opening switch). At the same time, energy is supplied to the closed loop from L_1 by the mutual inductance M. The final current I_2 therefore depends on the net energy stored in L_2, which is the difference between the energy supplied and the energy lost in R_{L2}.

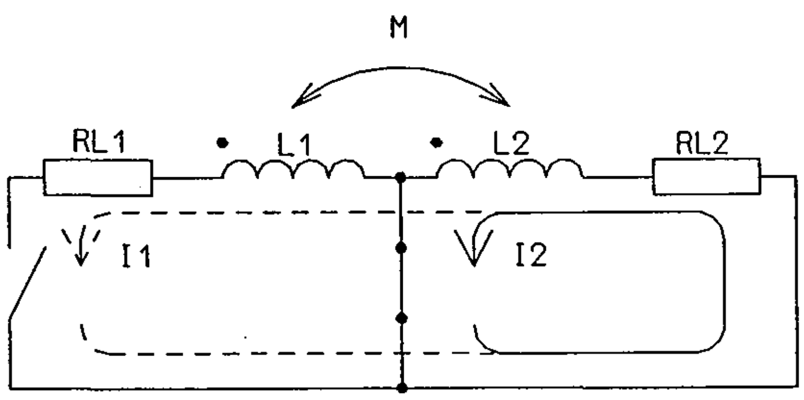


Figure 4.14 Single-Step Meatgrinder With Resistance

Although the meatgrinder current multiplication is independent of time (equation (2.1)), the losses in the resistance are not. Thus in a given resistive circuit the final current will vary according to the speed of the energy transfer. It is in fact possible to design the circuit such that the current actually remains constant throughout the transfer [30,31,43].

In the first four steps of the six-step meatgrinder the resistance losses are insignificant because of the relatively low currents and large time constants. (See table 4.3, which shows how the time constant of the closed loop decreases by a factor of nearly 1000 between the initial state and the fifth step.) In the last two steps the higher current means, however, that these losses become significant, even over a few microseconds. When the initial current is reduced, less time is required for the energy transfer (see table 4.2 and below) and a smaller proportion of the energy is dissipated in the resistance.

STEP	TIME CONSTANT (μs)
0	11 453
1	7 648
2	4 199
3	1 821
4	681
5	175
6	182

(Based on $R_{DS(ON)} = 0.085\Omega$)

Table 4.3 Time Constants of Closed Loops

4.3.3 Energy Transfer Times and Switch Voltages

Having established the importance of the speed of energy transfer, the factors which govern this speed are now discussed.

4.3.3.1 Analysis for Single-Step Circuit

In the two-section coil shown in figure 2.2, the requirement is to determine the time taken for the current in L_1 to fall to zero after S_1 has opened. Zucker et al [23] originally modelled the open state of S_1 as a constant resistance. However, as was observed subsequently [27] the analysis is simpler if S_1 has a constant voltage characteristic in its open state because the current decay is then linear rather than exponential.

In the present research, and in experiments carried out elsewhere [27,28,43], voltage clamp devices have been used which exhibit an almost constant voltage characteristic (see Chapter 3). An analysis based on such a characteristic is therefore both realistic and simple.

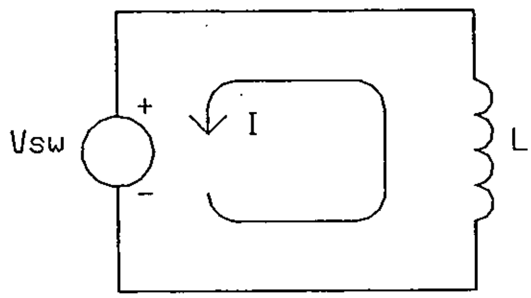


Figure 4.15 Decay of Current in Inductor

In the circuit shown in figure 4.15 the voltage across the opening switch is clearly

$$V_{sw} = L \frac{di}{dt}$$

from which the rate-of-change of the current is

$$\frac{di}{dt} = \frac{V_{sw}}{L} \qquad (4.1)$$

which is a constant. This being so, the time for the current to fall from an initial value I to zero is

$$t_{decay} = \frac{LI}{V_{sw}}$$

Following the example of Zucker [20], the meatgrinder circuit can be represented by the addition of a second loop to the circuit of figure 4.15 and the inclusion of further voltage sources to account for the mutual inductance. Figure 4.16 is thus an equivalent circuit for the single-step meatgrinder of figure 2.2.

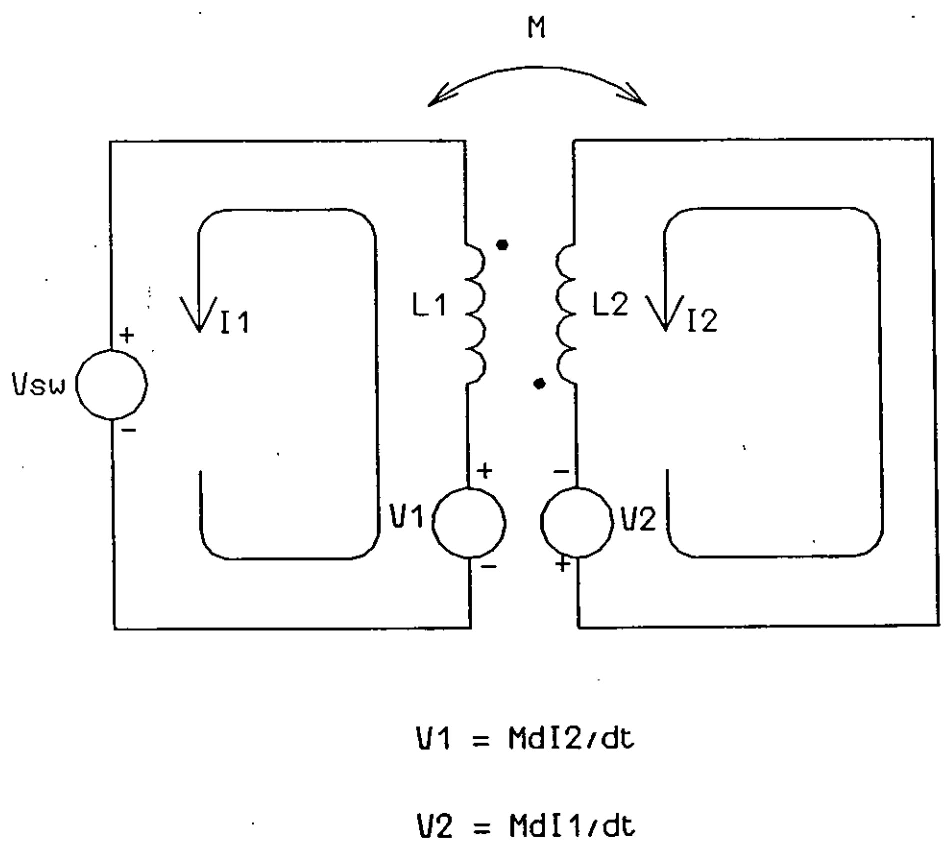


Figure 4.16 Equivalent Circuit of Single-Step Meatgrinder

Summing voltages around the two loops yields

$$V_{sw} = L_1 \frac{di_1}{dt} + M \frac{di_2}{dt} \tag{4.2}$$

$$0 = L_2 \frac{di_2}{dt} + M \frac{di_1}{dt} \tag{4.3}$$

Using equation (4.3) to eliminate di_2/dt from equation (4.2) leads to

$$V_{sw} = \frac{di_1}{dt} [L_1 - (M^2/L_2)]$$

$$= \frac{di_1}{dt} [L_1 (1-k^2)] \tag{4.4}$$

or

$$\frac{di_1}{dt} = \frac{V_{sw}}{L_1 (1-k^2)} \tag{4.5}$$

where k is the coupling coefficient between the self inductances L_1 and L_2.

For a given initial current I_1 and a constant switch voltage V_{sw} the energy transfer time is then

$$t_{trans} = I_1 \frac{L_1 (1-k^2)}{V_{sw}} \qquad (4.6)$$

This expression agrees with Zucker's analysis for the exponential case [23], which leads to a time constant for the decay of I_1 of $L_1(1-k^2)/R$, where R is the relevant circuit resistance.

It will be noticed by comparison with equation (4.1) that the meatgrinder action reduces the apparent inductance "seen" by the opening switch by a factor of $(1-k^2)$, which illustrates well how the efficiency of energy transfer is really a measure of the leakage inductance between L_1 and L_2. This is so because as k increases from zero, the effective inductance $L_1(1-k^2)$ falls, and the energy transfer time for a given switch voltage and initial current therefore also falls. The switch consequently experiences the same change of current in a shorter time and dissipates less energy.

In conclusion, the only way to speed up an energy transfer without changing the inductances is to raise the switch voltage.

4.3.3.2 Behaviour of Multi-Step Circuit

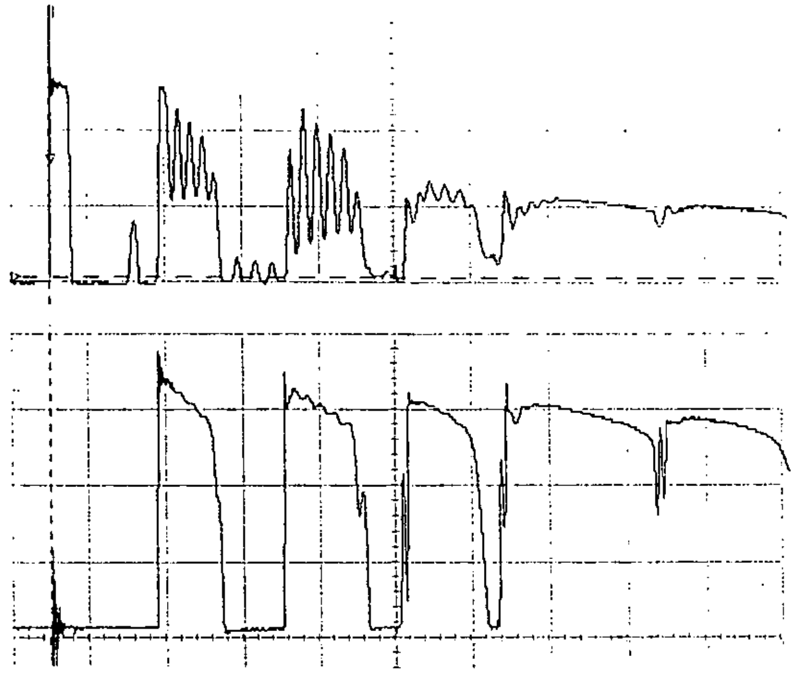
(i) First Energy Transfer

Current and voltage waveforms for the first energy transfer are shown in figure 4.4, and table 4.4 shows that the theoretical and experimental values for the energy transfer time are in good agreement.

(ii) Second Energy Transfer

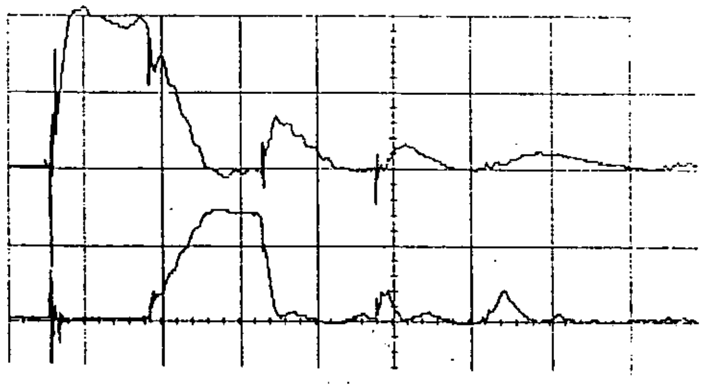
Current and voltage waveforms for the second transfer are shown in figure 4.17. It can be seen that although the TR2 voltage is not perfectly flat-topped the current decay is still reasonably linear.

It should be noted that the time predicted by equation (4.6) is sensitive to the value of the coupling coefficient. Changing this from 0.79 to 0.71, for example, yields a theoretical time of 8.5μs rather than 6.5μs. Given that the coupling coefficients in table 4.4 are derived from several inductance values, there could easily be significant uncertainties of between five and ten percent. If inaccuracies of a few percent in the experimental time values are added to this, the agreement between theoretical and experimental values is again reasonable.



Top: TR1 voltage 200V/div Time 10μs/div
Bottom: TR2 voltage 50V/div Time 10μs/div

(a)



Top: TR2 current 5A/div Time 10μs/div
Bottom: TR3 current 10A/div Time 10μs/div

(b)

Figure 4.17 Waveforms for Second Transfer

STEP	INDUCTANCE SWITCHED OUT (μH)	COUPLING COEFFICIENT	TRANSFER TIME (μs) *	
			THEORETICAL	EXPERIMENTAL
1	403	0.76	2.2^{+}	2.5
2	289	0.79	6.5^{++}	8.5
3	147	0.86	3.5^{++}	11.0
4	52	0.90	1.5^{++}	11.0
5	24	0.86	1.7^{++}	20.0
6	1	0.81	0.2^{+++}	4.0

* using equation (4.6)

\+ based on constant clamp voltage of 550V

++ based on constant clamp voltage of 150V

+++ based on constant clamp voltage of 150V and 70A current

Table 4.4 Transfer Details

It is evident from figure 4.17 that the voltage across TR1 rises again during the second transfer. (The ringing is due to the effect of parasitic components in the MOSFET. This is illustrated by figure 4.20 which shows the simulated waveforms when the MOSFET is replaced by a simple switch (voltage-controlled resistor).)

Clearly this voltage on TR1 is present for the duration of the second transfer and is less than the clamping value of about 550V.

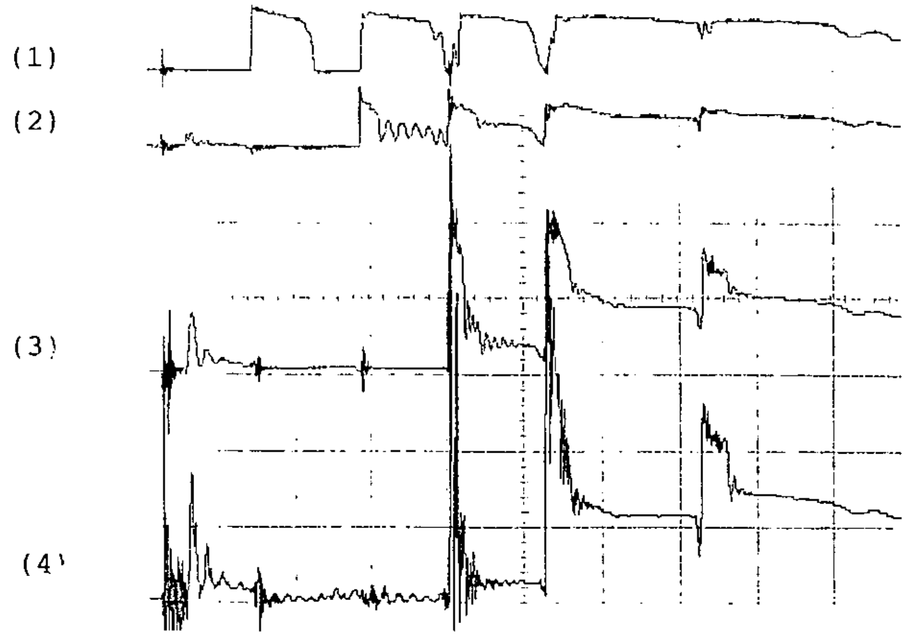
After TR1 has turned off, L_1 is still connected electrically to the remainder of the meatgrinder coil. More importantly, however, it is still magnetically coupled to the remainder of the coil. Therefore, when TR2 turns off and the voltage across L_2 rises, L_1 experiences an induced voltage by transformer action. This induced voltage is added to the voltage already present by virtue of the electrical connection, and the net result (with the small addition of the battery voltage) appears across TR1.

This transformer action is an important phenomenon in the design and operation of multi-step meatgrinders. Its significance in the remaining transfers is discussed below and is analysed in more detail in Chapter 5.

(iii) Remaining Energy Transfers

It can be seen from Table 4.4 that the energy transfer times for the last four steps are much longer than would be expected from the analysis of a single-step meatgrinder. Figure 4.18 shows that the voltage waveforms at turn-off are not simple flat-topped pulses, and this leads to longer energy transfer times (see below). Figure 4.19 shows the corresponding current waveforms.

In order that the phenomena discussed below can be seen more clearly, simulated voltage and current waveforms are given in figures 4.20 and 4.21. These results are obtained from a model which uses simple switches rather than MOSFETs, but does include both coil and switch resistance. Stray coil capacitance is not included. (Although the MOSFETs are eliminated, there is still some high-frequency ringing, due probably to diode capacitance.)



All traces: Time 10μs/div

(1) TR2 voltage 200V/div (2) TR3 voltage 200V/div

(3) TR4 voltage 50V/div (4) TR5 voltage 20V/div

(a)

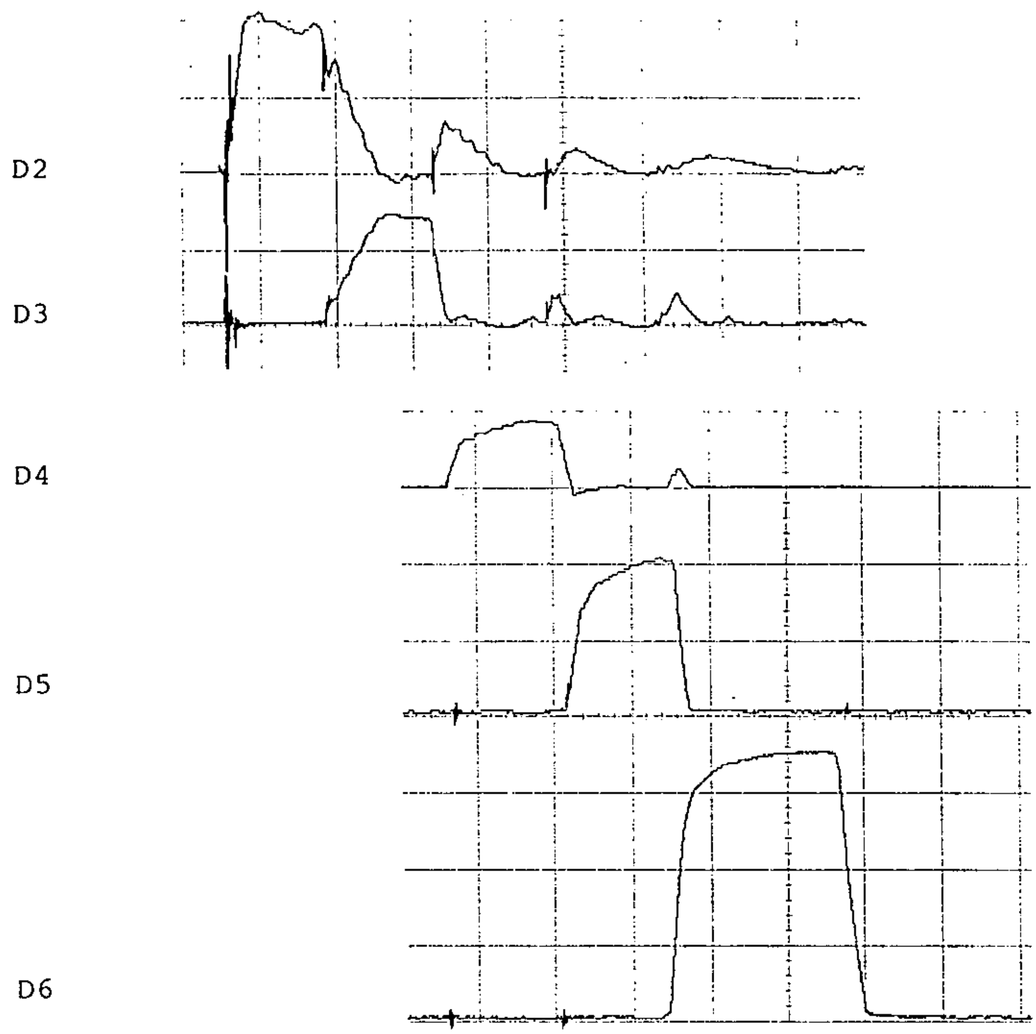
All traces: Time 10μs/div

(1) TR2 voltage 200V/div (2) TR5 voltage 50V/div

(3) TR6 voltage 10V/div (4) TR7 voltage 10V/div

(b)

Figure 4.18 Voltage Waveforms



All traces: Time 10μs/div

D2 5A/div D3 10A/div D4 27.5A/div
D5 20A/div D6 20A/div

Figure 4.19 Current Waveforms

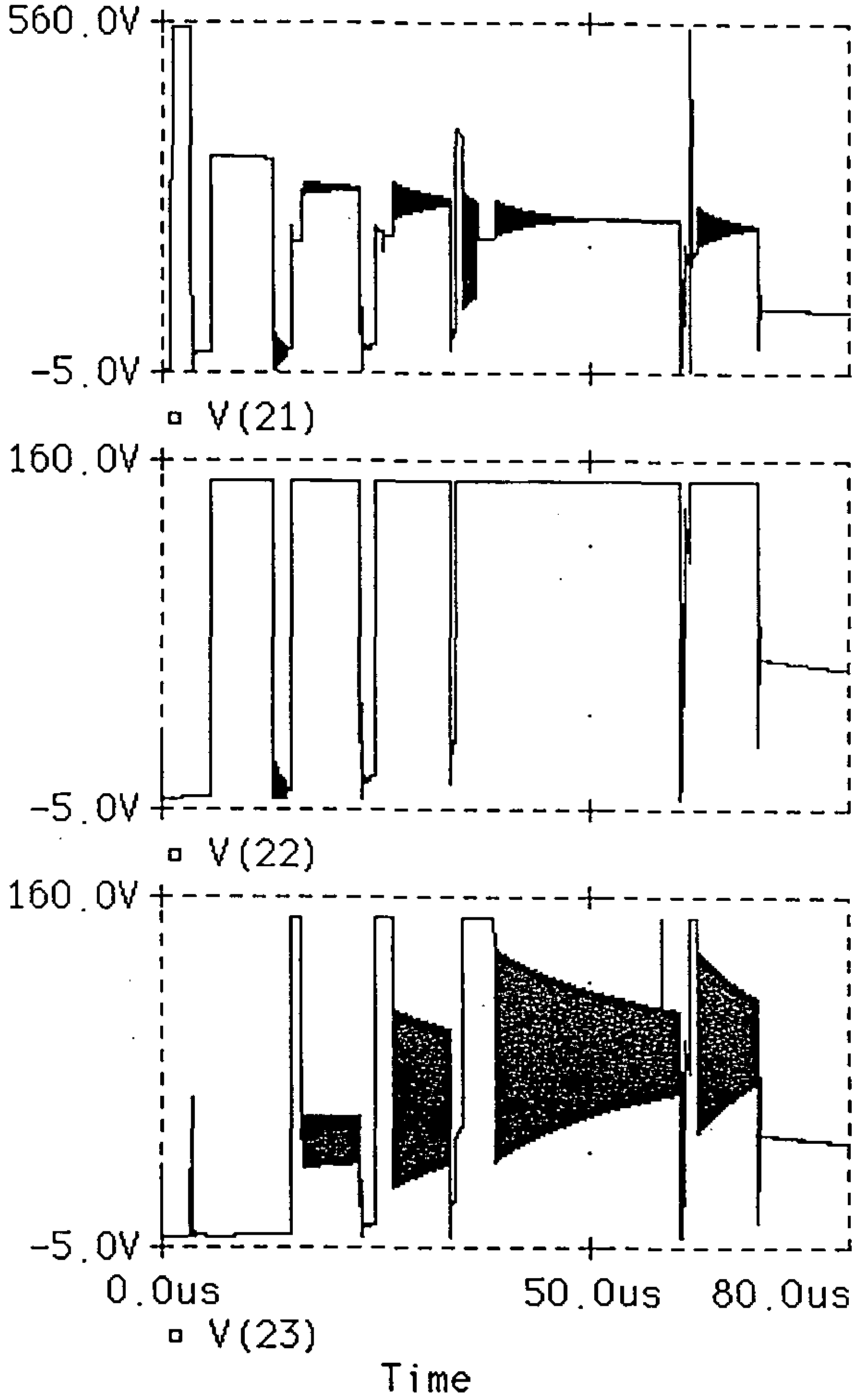


Figure 4.20 Simulated Voltage Waveforms

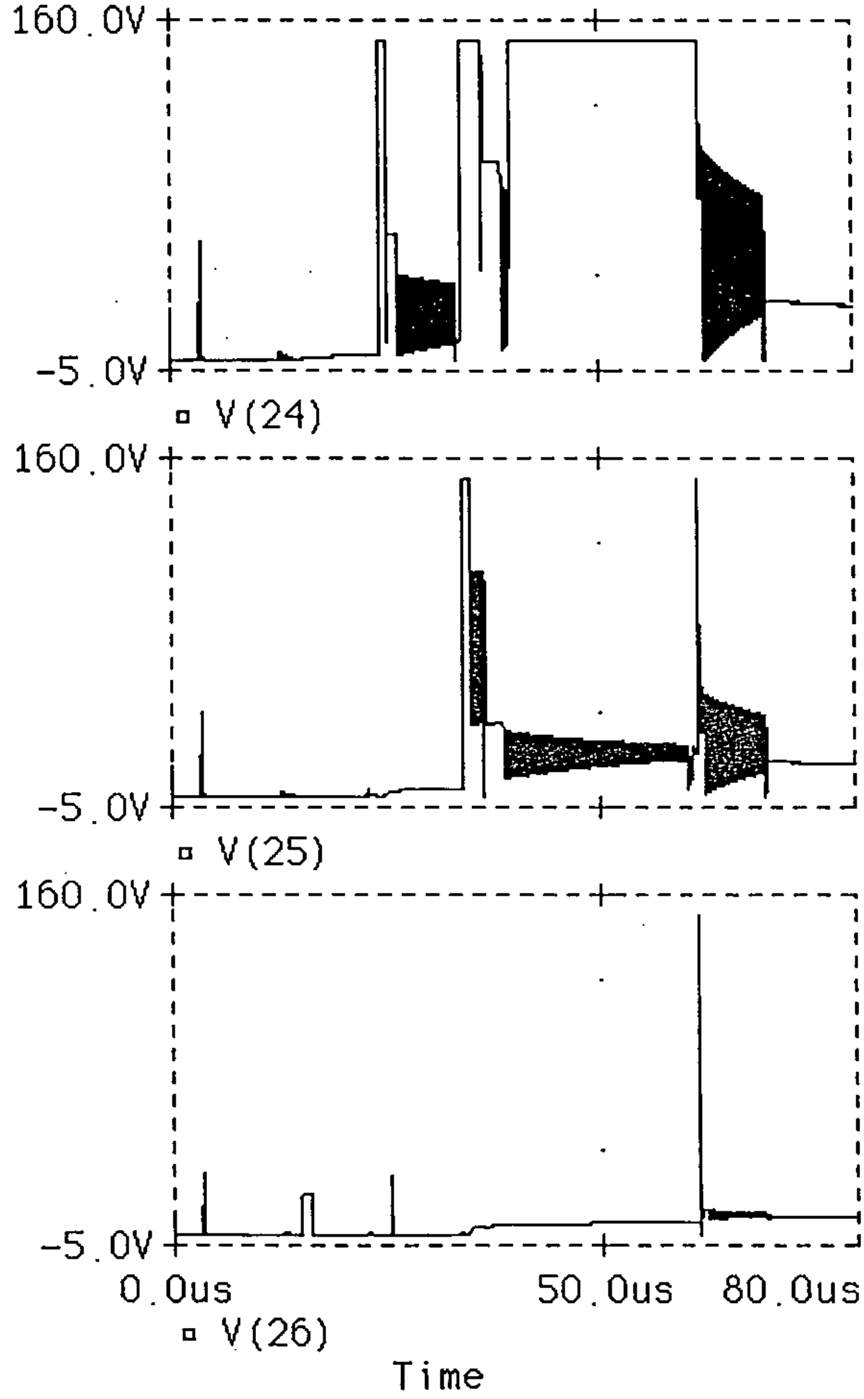


Figure 4.20 Simulated Voltage Waveforms (continued)

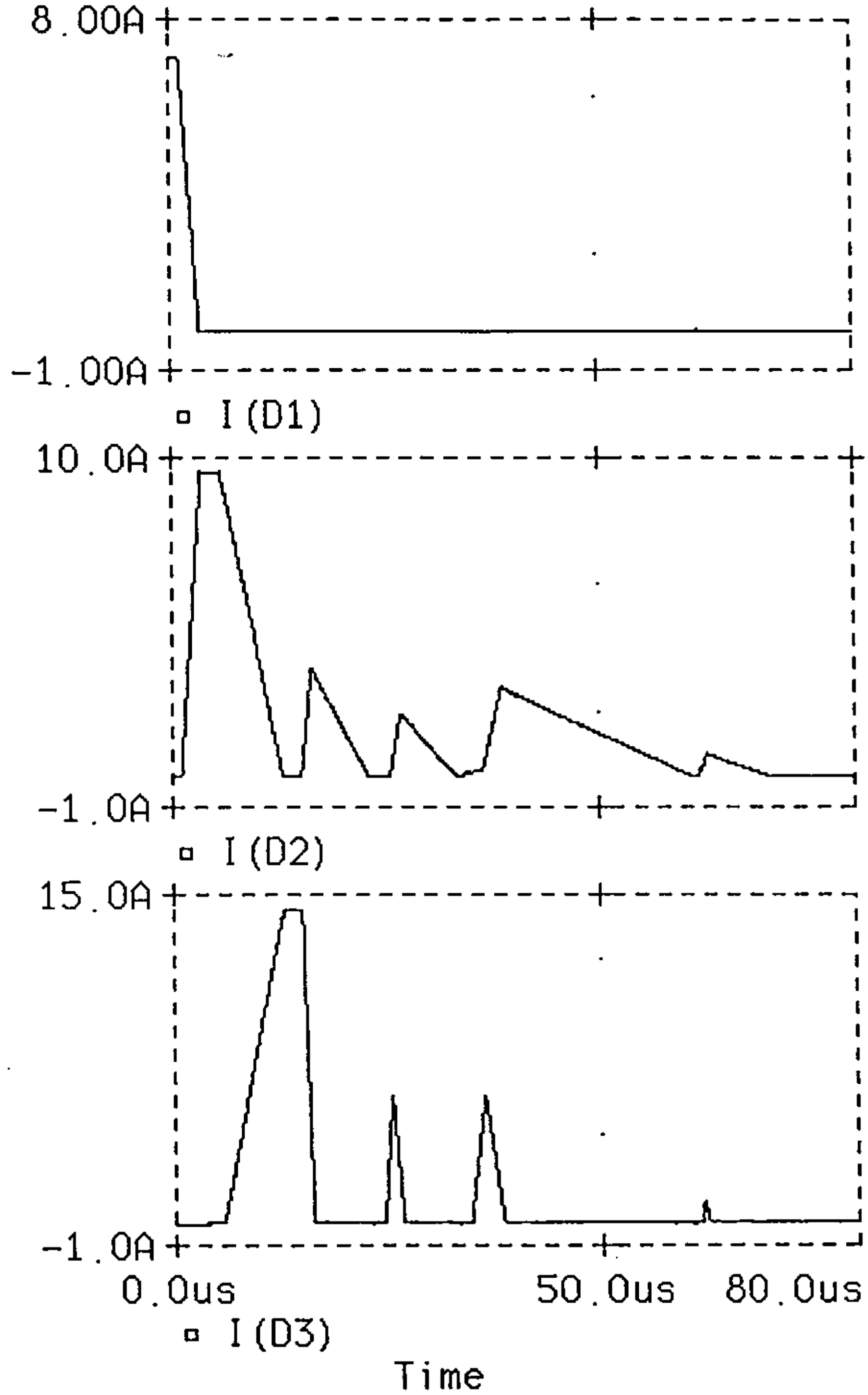


Figure 4.21 Simulated Current Waveforms

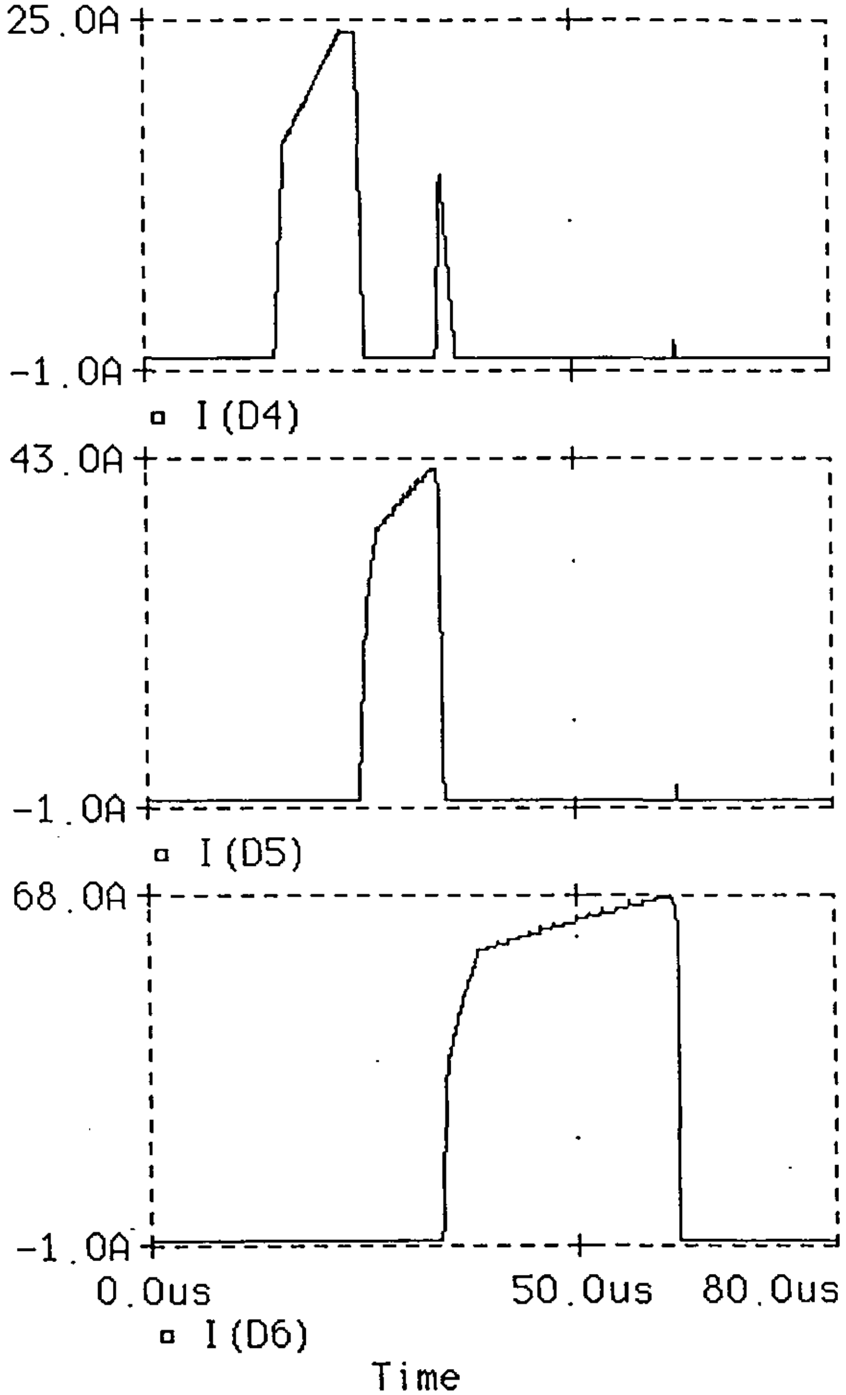


Figure 4.21 Simulated Current Waveforms (continued)

In addition to the transformer action described above, another phenomenon occurs in the multi-step meatgrinder, in that during each of the last four energy transfers, the voltage across TR2 reaches the clamp level (150V) and current flows through L_2 via the clamp device across TR2. Taking the third energy transfer as an example, it will be seen that the current in TR4 has two distinct rates of rise. Corresponding to this are two distinct voltages across TR3, which is the opening switch for the third energy transfer. The voltage across TR3 is at its clamp value for only a small proportion of the energy transfer time, and as a consequence the transfer takes longer. (A detailed analysis of this process is given later, in Chapter 5.)

This phenomenon is repeated during each of the remaining transfers, with the voltage across TR2 remaining at its clamp level and current flowing in inductors which have previously been switched out. When the voltage across TR3, for example, is at its lower value, it is determined not by the clamp device connected across TR3 but by the clamp device connected across a different transistor, which has been forced to clamp by the transformer action of the meatgrinder. This phenomenon will therefore be referred to in this thesis as "transformer action clamping".

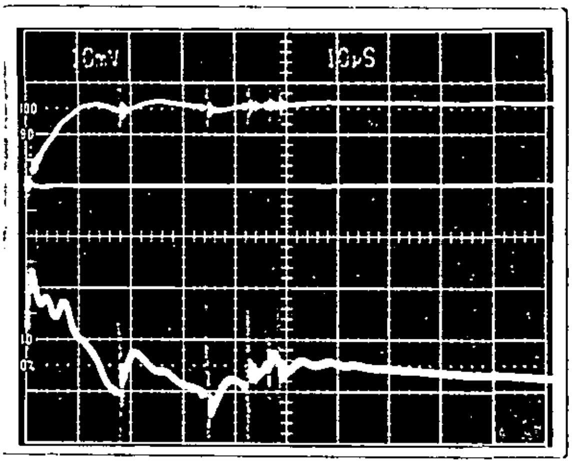
(iv) Effect of Switch Voltages on Energy Transfer Times

It was shown previously that where no current flows in previously switched-out inductors, the energy transfer time is controlled by the clamp voltage of the opening switch. When transformer action clamping occurs the clamp voltages of any previous switches are also signifcant.

In this case it would be desirable to speed up the fifth and sixth energy transfers in order to reduce the losses in the circuit resistance, thereby increasing the final current. Changing TR1 from a 200V IRF250 (protected by a MOV which actually clamps at about 150V) to a 500V IRF450 (which actually avalanches at about 550V) did not discernibly increase the final current, but it did reduce the energy transfer times. This would obviously be expected for the first energy transfer. However, the second transfer is also speeded up because the voltage across TR2 is allowed to remain at its clamp level for the entire duration of the transfer (see figure 4.20). Previously it experienced the transformer action clamping now experienced only by TR3 and subsequent MOSFETs.

Voltage
100V/div

Current
5A/div



Time 10μs/div

Figure 4.22 TR1 Voltage and Current With 200V Device for TR1

Figure 4.22 illustrates how using the lower voltage device for TR1 meant that TR1 was the "controlling" device for subsequent transfers. (Note: the waveforms in figure 4.22 are inaccurate because of measurement problems at the time. The rise time of the voltage, for example, should be much less than is indicated. However, the waveforms do serve to illustrate the point being made.) It can be seen that the voltage across TR1 remains at its clamp level throughout the process. With the higher voltage device used for TR1, it is the voltage across TR2 which remains at its clamp level during each transfer.

TR3 experiences transformer action clamping in both cases. With the lower voltage device used for TR1, the transformer action clamp voltage is 150V across TR1, i.e. about 126V (150V less the battery voltage) at the end of L_1. With the higher voltage device, the clamp voltage is 150V across TR2, i.e. 150V at the junction of L_1 and L_2. Since the transformation ratios are unchanged, this corresponds to a higher voltage at the junction of L_2 and L_3, i.e. across TR3.

Similar reasoning applies to subsequent transfers. It is clear, however, that the increased speed of the fifth and sixth transfers is insufficient to increase the final current obtained. Figure 4.20 shows that the clamped voltages across TR5 and TR6 during the fifth and sixth energy transfers respectively are very low - of the order of 10V. This is because of the high turns ratios by the time these transfers are reached. As a result, a large increase in the voltage at the end of L_1, for example, results in only a small increase in the voltage at, for example, the junction of L_5 and L_6.

Computer simulation was used to test the effect of making both TR2 and TR3 500V devices. The corresponding results are shown in figures 4.23 and 4.24. Transistor TR1 once again becomes the "controlling" device and the final current is nearly 90A. Figure 4.25 shows that the voltage across TR5 during the fifth transfer is increased significantly by the increased clamp voltages of TR2 and TR3. The voltage across TR6 during the sixth transfer is not significantly altered, but by implication it could be increased by raising the switch voltages even further.

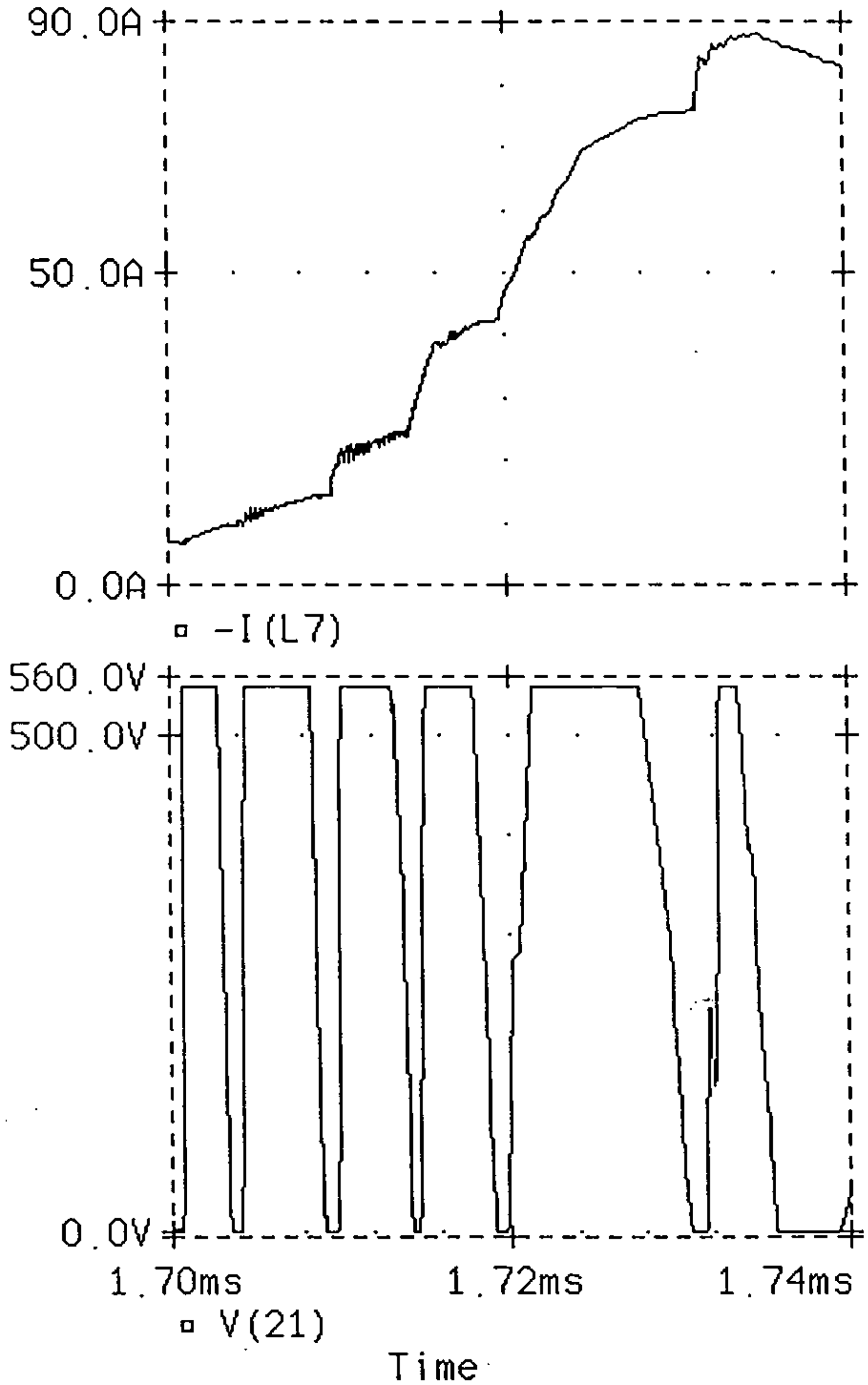


Figure 4.23 Effect of Making TR2 and TR3 500V Devices

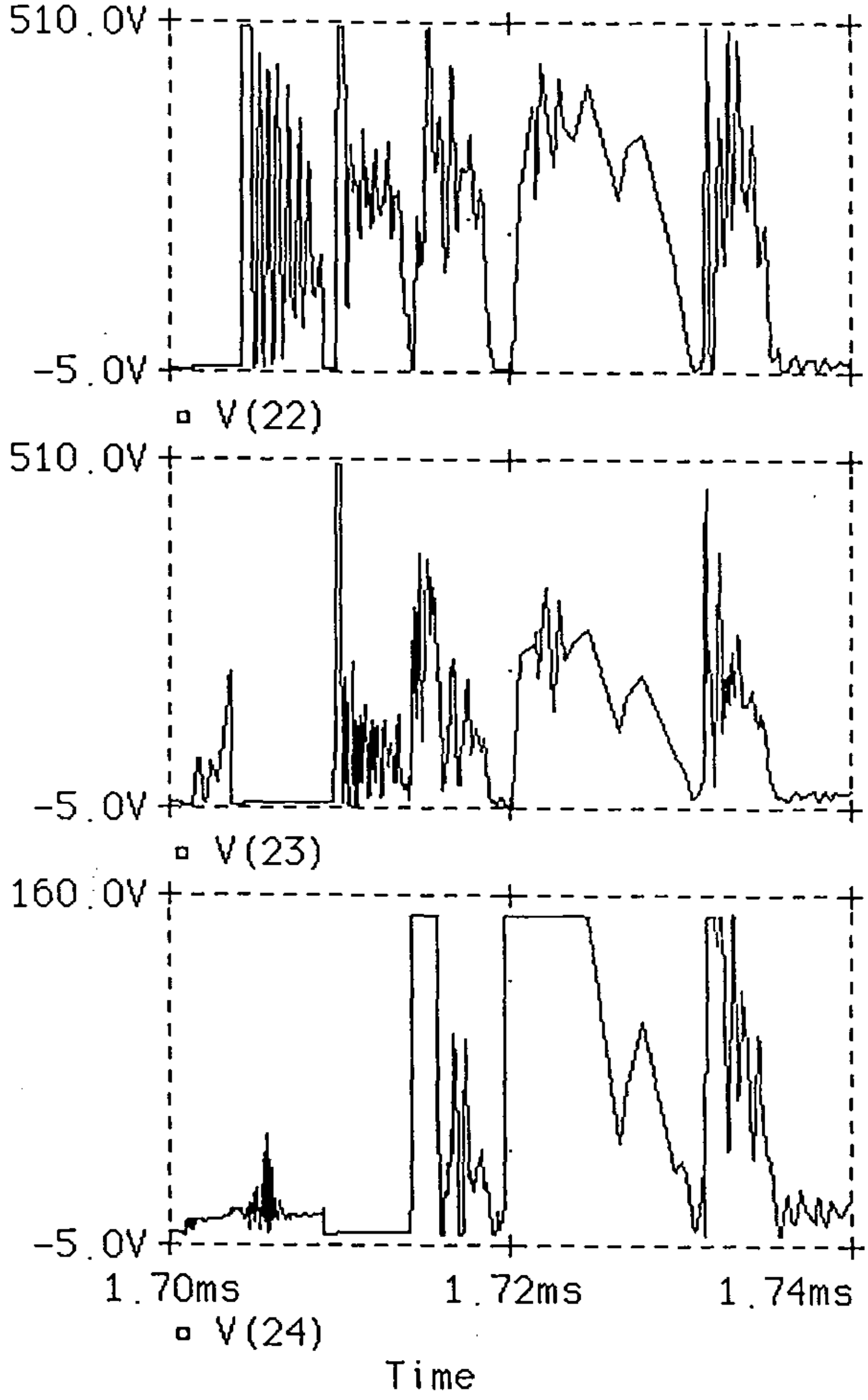


Figure 4.24 Simulated Voltage Waveforms With 500V Devices for TR2 and TR3

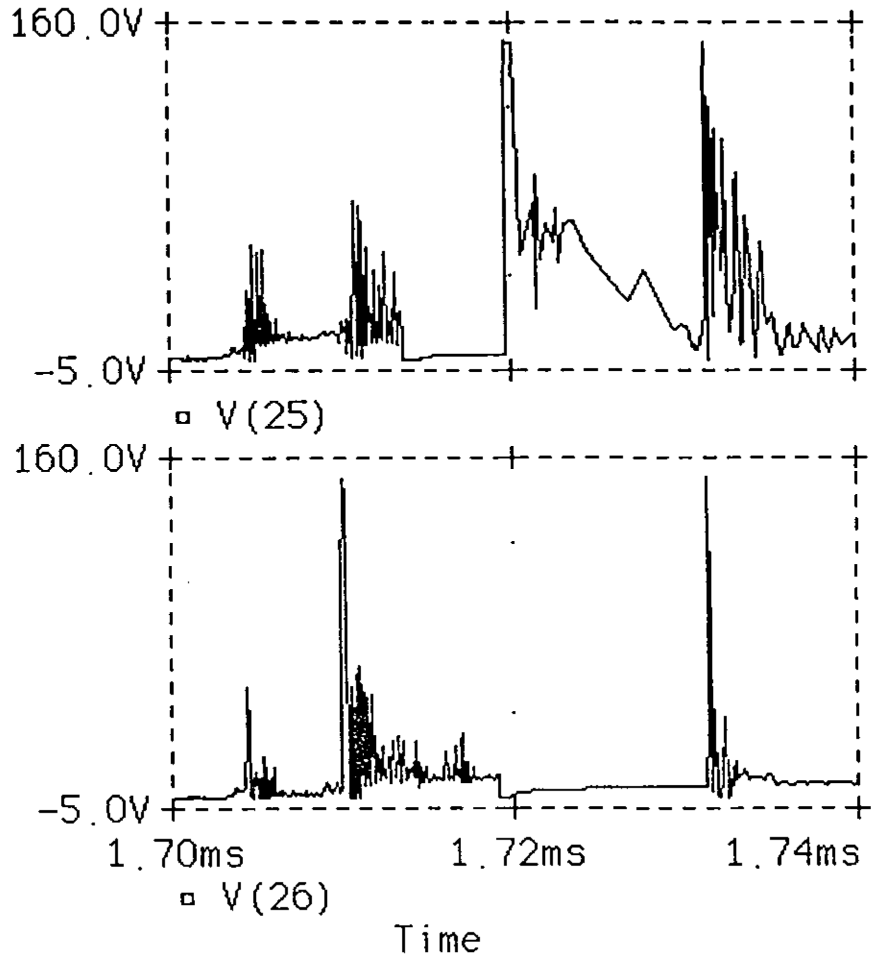
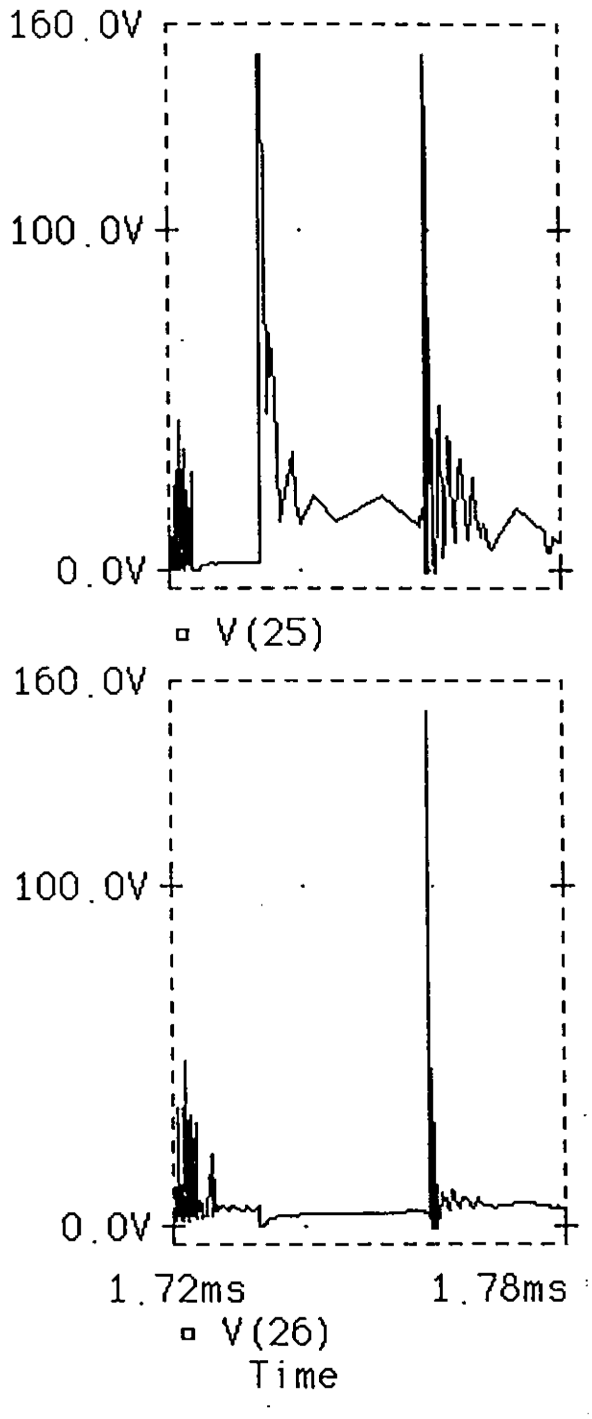
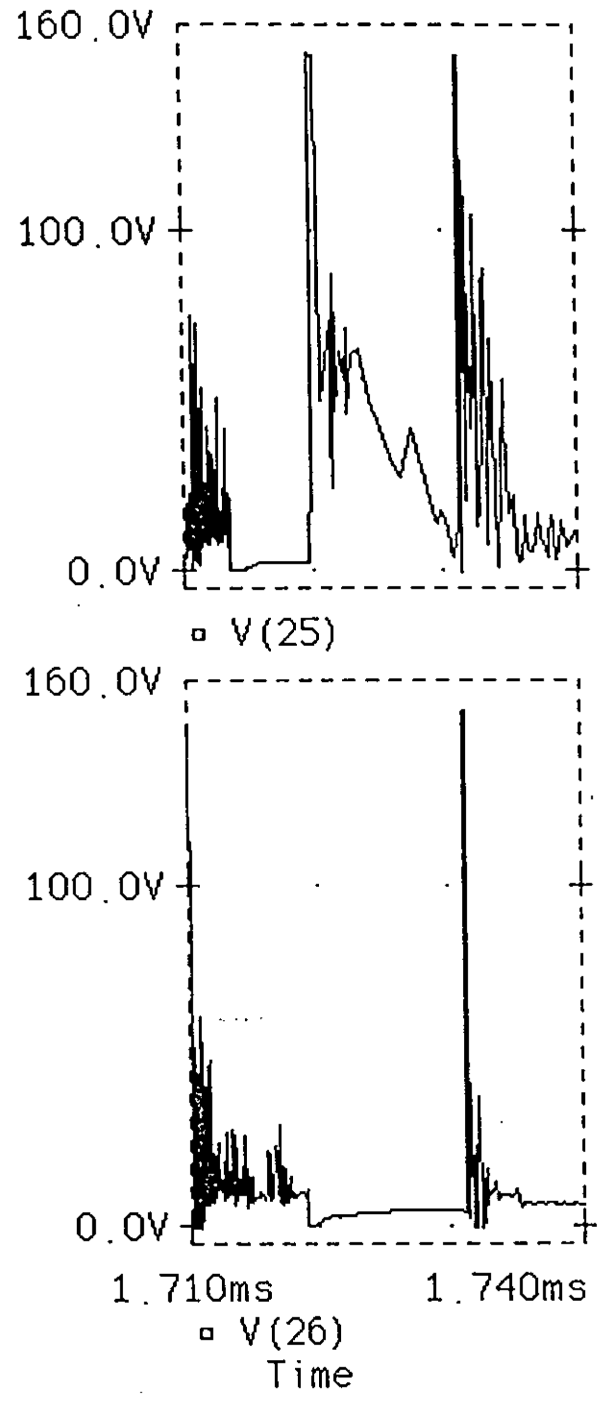


Figure 4.24 Simulated Voltage Waveforms With 500V Devices for TR2 and TR3 (continued)



(a) With IRF450 for TR1 Only



(b) With IRF450 for TR1, TR2 and TR3

Figure 4.25 Switch Voltages During Fifth and Sixth Transfers

The conclusion to be drawn from this discussion is that in a multi-step circuit, just as in the simple case, the energy transfer times are dictated by the switch voltages. The difference in a multi-step circuit is that the result is affected by the clamp voltage of previously-opened switches, in addition to the clamp voltage of the opening switch itself. This topic has been addressed by other workers [13,27,30,43] and is discussed further in Chapter 5.

4.4 VOLTAGES ACROSS IN-CIRCUIT INDUCTORS

4.4.1 Theory

4.4.1.1 With Uncoupled Load

The single-step meatgrinder shown in figure 4.26 has a magnetically uncoupled load L_{LOAD}. (The meatgrinder is assumed to be charged, and the current source is omitted for clarity.)

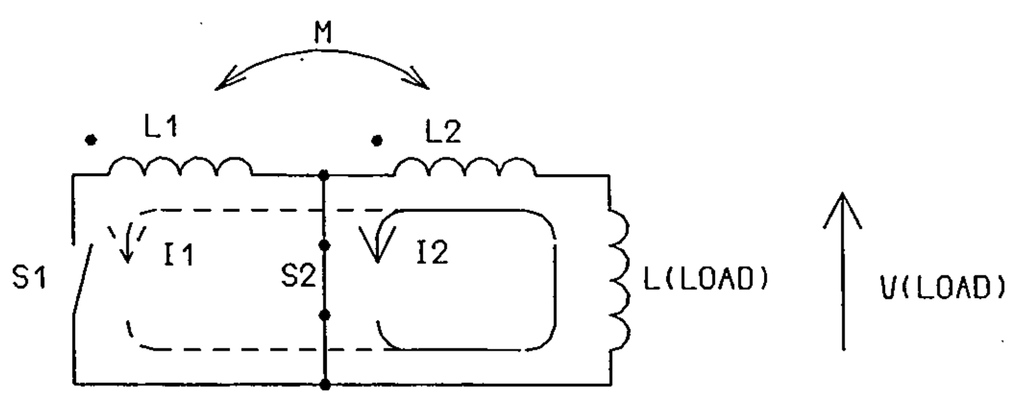


Figure 4.26 Single-Step Meatgrinder With Uncoupled Load

The voltage across the load is given by

$$V_{LOAD} = L_{LOAD} \frac{dI}{dt} \qquad (4.7)$$

Substituting for dI/dt from equation (4.5) gives

$$V_{LOAD} = V_{sw} \frac{L_{LOAD}}{L_1(1-k'^2)} \qquad (4.8)$$

where k' is the coupling coefficient modified to account for L_{LOAD} (see Appendix D).

If there is no circuit resistance, the voltage across L_2 is at all times equal and opposite to that across L_{LOAD}.

The change in flux linkage experienced by L_{LOAD} is obtained from

$$\delta(N\Phi)_{LOAD} = \int_0^{\delta t} V_{LOAD}\, dt \qquad (4.9)$$

where δt is the energy transfer time.

The change in flux linkage experienced by L_2 is similarly

$$\delta(N\Phi)_{L2} = \int_0^{\delta t} V_{L2}\, dt \qquad (4.10)$$

and since $V_{L2} = -V_{LOAD}$ then

$$\delta(N\Phi)_{LOAD} = -\delta(N\Phi)_{L2} \qquad (4.11)$$

This confirms that flux linkage is conserved around the closed loop. The initial and final flux linkages are

$$(N\Phi)_1 = I_1(L_2+M_{12}+L_{LOAD}) \tag{4.12}$$

$$(N\Phi)_2 = I_2(L_2+L_{LOAD}) \tag{4.13}$$

and it can be seen that the current increase is balanced out by the loss of the effect of the mutual inductance.

4.4.1.2 With Coupled Load

(i) Coil With Three Sections

If the load is actually the last section of the meatgrinder coil (see figure 4.27), the same principle applies, as illustrated by the following example:

Let the parameters of the circuit in figure 4.27 be

$$L_1 = L_2 = 10\mu H,\ L_3 = 100\mu H,\ k_{12} = 0.9,\ k_{23} = 0.5,\ k_{13} = 0.3$$

This means that

$$M_{12} = 9\mu H,\ M_{13} = 9.49\mu H,\ M_{23} = 15.81\mu H$$

and that

$$L_{23} = 141.62\mu H,\ L_{1-3} = 188.59\mu H$$

Therefore

$$k_{1(23)} = 0.49,\ M_{1(23)} = 18.44\mu H$$

By applying equation (2.1), the final current for an initial current of 10A is found to be 11.3A. This leads to the figures shown in table 4.5.

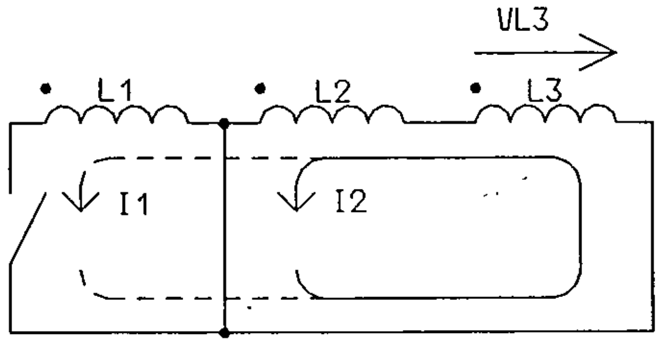


Figure 4.27 Three-Section Meatgrinder

FLUX LINKAGE WITH L_2			FLUX LINKAGE WITH L_3		
INITIAL	FINAL	CHANGE	INITIAL	FINAL	CHANGE
348	292	-56	1253	1309	+56

Flux linkages in weber-turns x 10^{-6}

Table 4.5 Example Showing Conservation of Flux Linkage in Circuit of Figure 4.27

Flux linkage is again conserved. Since there is a change of flux linkage there must again be a voltage at the junction of L_2 and L_3 during the transfer.

The direction of the induced voltage is governed by Lenz's law as before. In this example L_3 experiences an increase in flux linkage, just as it would do if it were an uncoupled load. Therefore, for the direction of current flow shown in figure 4.27, the polarity of the voltage is as indicated.

The generalised expression for the change in flux linkage experienced by L_3 is

$$\delta(N\Phi)_{L3} = I_2(L_2+M_{23}) - I_1(L_2+M_{12}+M_{23}) \tag{4.14}$$

Applying equation (2.1) for the three-section case gives

$$I_2 = I_1 \frac{L_2+L_3+2M_{23}+M_{12}+M_{13}}{L_2+L_3+2M_{23}} \tag{4.15}$$

and eliminating I_2 from equation (4.14) yields

$$\delta(N\Phi)_{L3} = I_1 \left[\frac{M_{13}(L_2+M_{23}) - M_{12}(L_3+M_{23})}{L_2+L_3+2M_{23}} \right] \tag{4.16}$$

If the transfer of energy occurs at a constant rate in a time δt, the voltage is

$$V_{L3} = \frac{\delta(N\Phi)_{L3}}{\delta t} = \frac{I_1}{\delta t}\left[\frac{M_{13}(L_2+M_{23}) - M_{12}(L_3+M_{23})}{L_2+L_3+2M_{23}}\right] \qquad (4.17)$$

It can be seen that the quantity in brackets can be positive, negative or even zero depending on the values of the different inductances.

(A more general expression, which would also apply to non-linear transfers, would use di_1/dt in place of $I_1/\delta t$.)

(ii) Coil With Four or More Sections

When the inductance in the closed loop consists of more than two sections the voltage across any one section can be found by applying the technique described above.

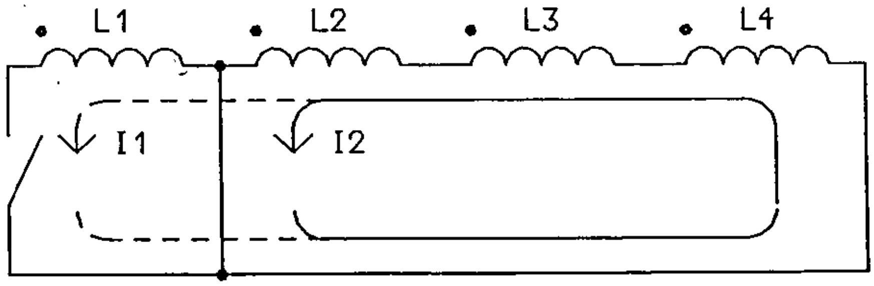


Figure 4.28 Four-Section Meatgrinder

For example, analysing the circuit of figure 4.28 gives

$$V_{L2} = \frac{di_1}{dt}\left[\frac{M_{12}(L_3+L_4+2M_{34}+M_{23}+M_{24})-(M_{13}+M_{14})(L_2+M_{23}+M_{24})}{L_2+L_3+L_4+2(M_{23}+M_{24}+M_{34})}\right]$$

(4.18)

which is similar to equation (4.17) and can again be positive, negative or zero.

As the number of meatgrinder sections increases, the equation becomes longer. This can be avoided, however, if several sections are treated collectively, thereby reducing the circuit to that of figure 4.27. The process is outlined below:

For an n-section meatgrinder, with L_1 being switched out, the total inductance in the closed loop is L_{2-n}. If the voltage across a particular section, say L_x, is required, the total inductance of the remaining sections, L_y, is

$$L_y = L_{2-n} - \left[L_x + \sum_{a=2}^{a=(x-1)} M_{ax} + \sum_{a=(x+1)}^{a=n} M_{ax}\right] \qquad (4.19)$$

that is, it is the total inductance less the self inductance of L_x, less all the mutual inductances associated with L_x. Thus the (n-1) sections which form the closed loop can now be considered as two sections L_x and L_y.

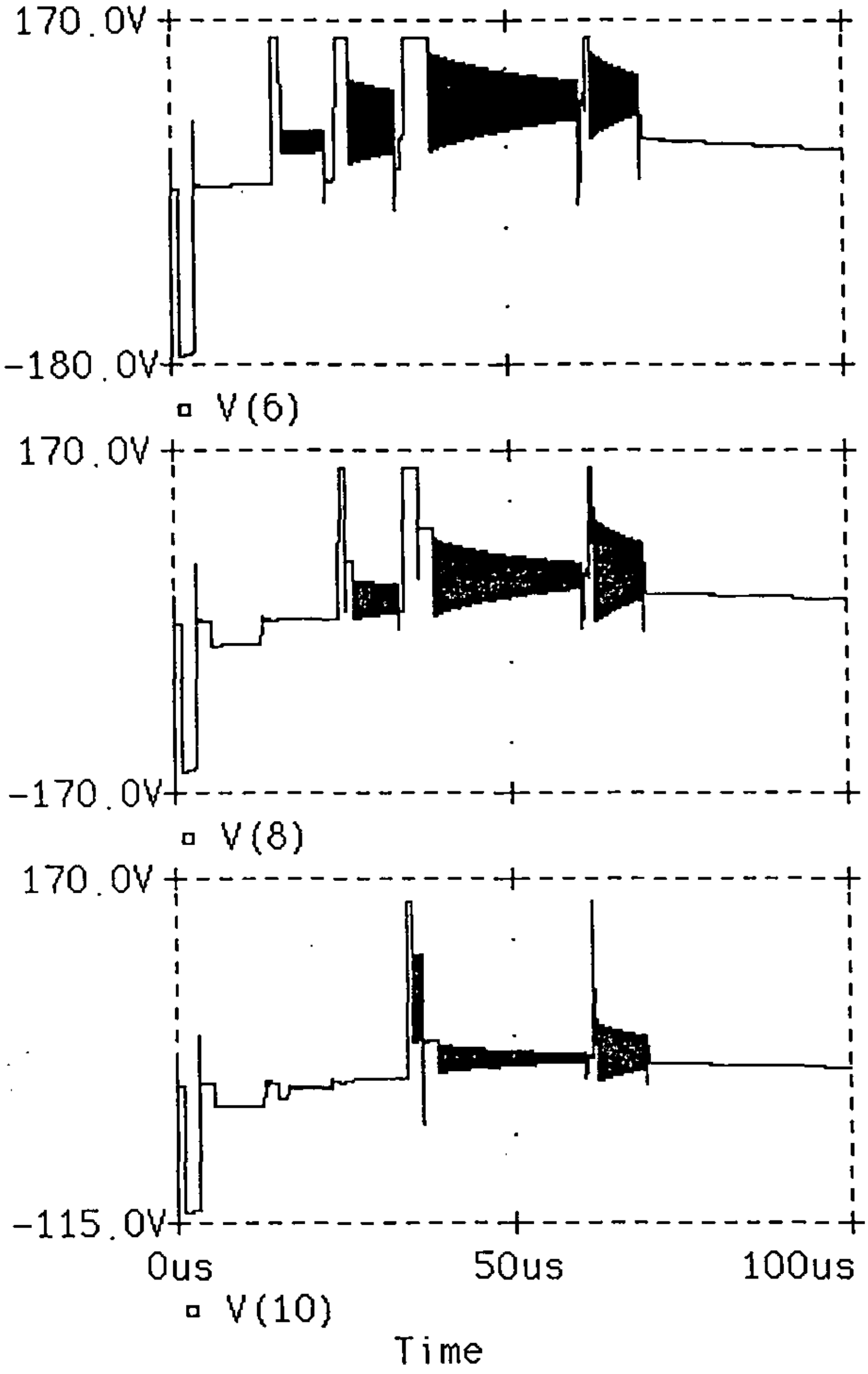


Figure 4.29 Simulated In-Circuit Inductance Voltages

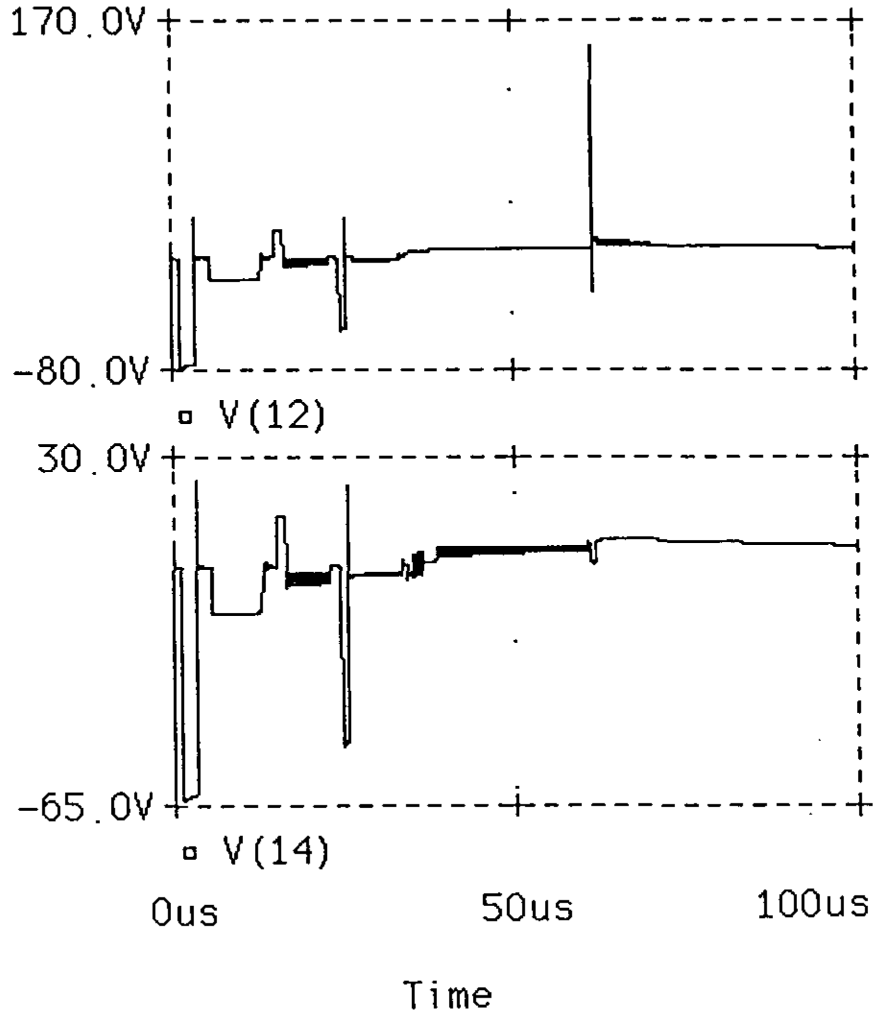


Figure 4.29 Simulated In-Circuit Inductance Voltages (continued)

If the voltage across the last coil section L_n is required, L_y is replaced by $L_{2-(n-1)}$. In other words, to find the voltage across L_4 in figure 4.28, L_2 and L_3 are treated collectively as L_{2-3}.

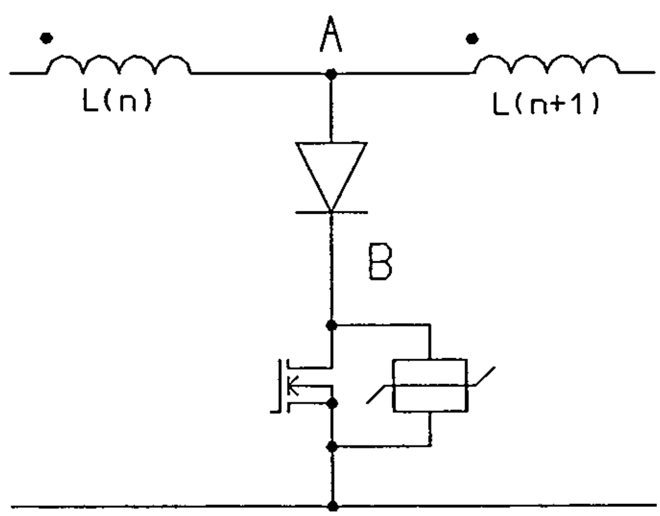
4.4.2 Six-Step Circuit Waveforms

According to the theory above, an "in-circuit inductance voltage" is expected at each of the junctions between coil sections (see figure 3.6), except for that between L_1 and L_2 because this junction is either shorted to ground (first energy transfer) or out of circuit (remaining transfers). The simulated waveforms given in figure 4.29 show these voltages. The effect is particularly noticeable during the first transfer.

It will be noted that the voltage waveforms given previously (figure 4.20, for example) do not show this effect because they are on the cathode side of the blocking diode (see figure 4.30). A negative voltage on the anode reverse biases the diode and is therefore not seen at the cathode.

The voltage across L_7 during the first energy transfer is about 65V, which agrees with the figure obtained by working out the flux linkage change and dividing by the energy transfer time. (The figure was obtained from a Fortran program which, although useful in other respects, was largely supplanted for this type of calculation by the "PSpice" simulation program as the research progressed. The Fortran program is described in Appendix E.)

Experimental voltage waveforms for the six-step meatgrinder are given in figure 4.31. The expected "in-circuit inductance" voltages can be seen, although these are somewhat obscured by noise. The noise is significant because it increases the maximum reverse voltage experienced by the blocking diodes:



Negative voltage at A not seen at B

Figure 4.30 Reverse Biasing of Blocking Diode

In the simulated waveforms of figure 4.29 the peak negative voltage at the junction of L_2 and L_3 is about 180V. This represents the peak reverse voltage across the blocking diode D3.* Figure 4.31 indicates, however, that the ringing causes this reverse voltage to reach almost 240V, which is the peak non-repetitive reverse voltage rating of the MR752 diode.

* see figure 3.6

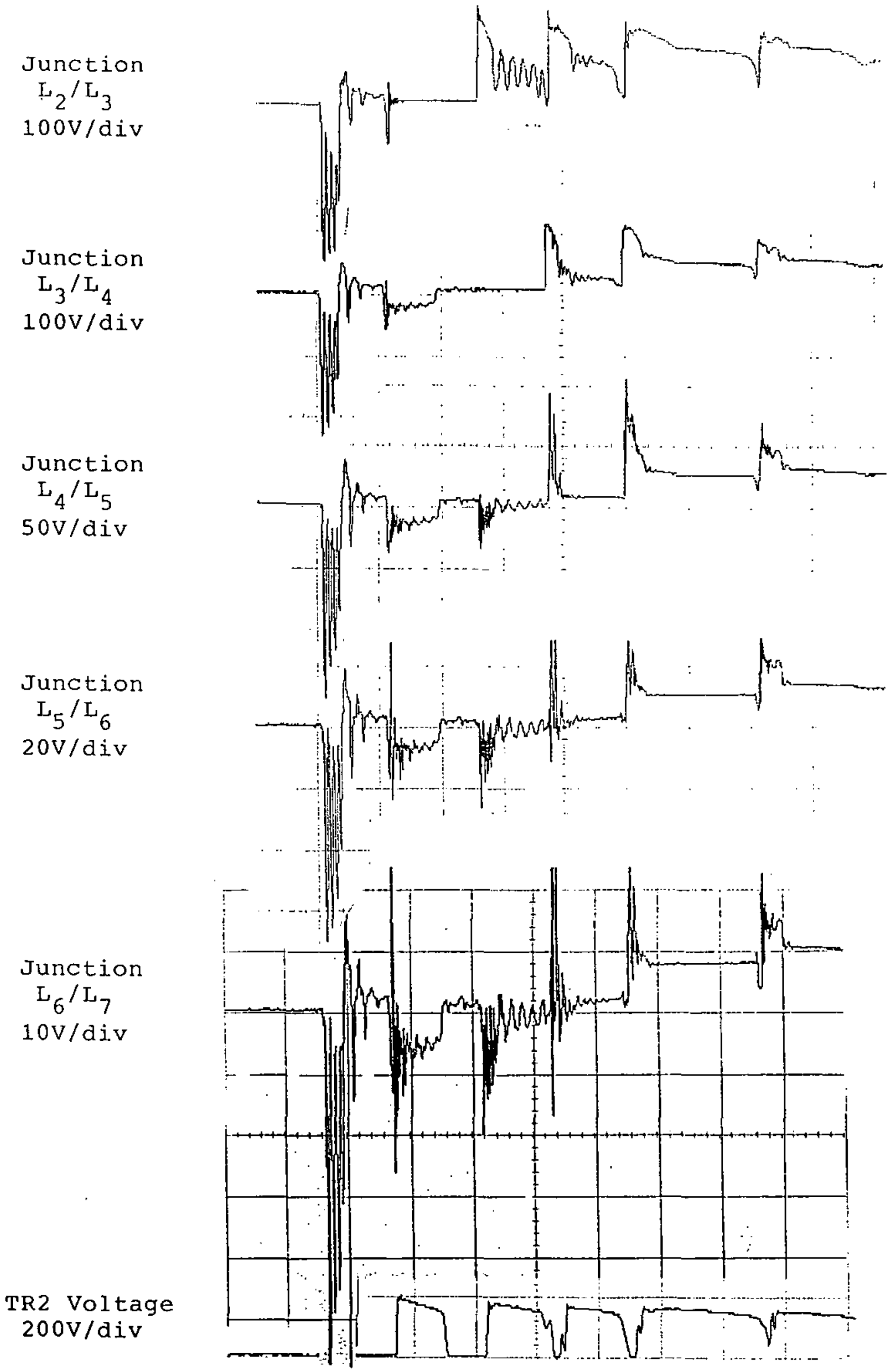


Figure 4.31 Experimental Waveforms: In-Circuit Inductance Voltages

All traces: Time 10μs/div

Figure 4.32 is the simulated waveform from the model which uses MOSFETs rather than simple switches. This predicts that the ringing will cause a reverse voltage greater than 180V. In fact the voltage is clamped at 200V, which is the value used for the breakdown voltage in the MR752 component model. (PSpice does not model destructive breakdown, but assumes that zener action takes place when the breakdown voltage is exceeded. Also, it is not possible to distinguish between a repetitive and a non-repetitive rating as only one value can be specified.)

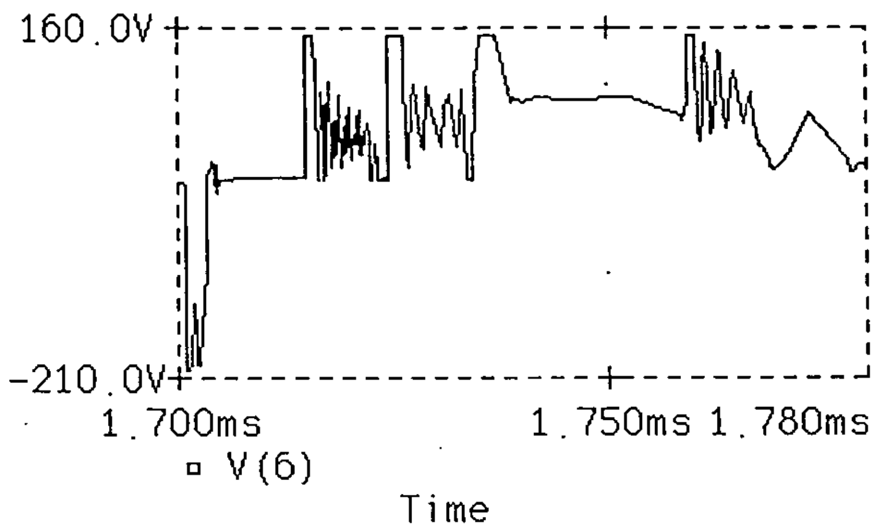


Figure 4.32 Simulated Voltage at Junction of L_2 and L_3 Including Parasitic Effects

This illustrates how it can be beneficial both to simulate an idealised circuit (in order that the phenomenon under investigation is not obscured by noise), and to simulate the behaviour of real components in the same circuit (in order to assess the significance of noise).

4.4.3 Possible Breakdown of Blocking Diodes

The discussion above raises two points. Firstly, it seems that in the experimental meatgrinder circuit (see figure 3.6), diodes D3 and D4 are only surviving by chance, since their reverse voltage approaches the maximum rating. This means that it is not simply the voltage experienced during charging which must be considered when selecting such diodes.

Secondly, if the diodes were capable of non-destructive breakdown, the consequences of such breakdown would need to be examined. In such a case the voltage across the diode is clamped and current flows through the device. Since the clamping action is caused by an induced voltage related to the voltage across the opening switch, this can again be referred to as transfomer action clamping (TAC).

4.4.4 Terminology

For clarity, the phenomenon discussed earlier relating to previously switched-out inductors will be referred to as external TAC (ETAC). Similarly, the phenomenon relating to in-circuit inductors will be referred to as internal TAC (ITAC).

ETAC is due to clamping by the devices across the MOSFETs. It has already been shown that in-circuit inductance voltages may be of either polarity. There is thus no reason why ITAC cannot occur in this mode, meaning that a further sub-division of ITAC is required to distinguish between the two possible modes. ITAC due to the MOSFET clamp device will be referred to as positive ITAC (since current then flows in the same direction as it does when the MOSFET is turned on), whilst that due to the blocking diodes (or the devices performing that function) will be referred to as negative ITAC.

Further analysis of TAC, together with a discussion of its implications, is given in Chapter 5.

CHAPTER FIVE

ANALYSIS OF THE EFFECTS OF TRANSFORMER ACTION CLAMPING

5.1 INTRODUCTION

The meatgrinder relies on magnetic coupling between coil sections in order to transfer energy forwards as each section is switched out. As a system of coupled coils the circuit can be regarded as a transformer, although it is not usually described in such terms. Consequently, a changing flux associated with one section will generate induced voltages across other sections, and corresponding currents will flow whenever a closed path exists. This will be referred to in this thesis as transformer action.

Transformer action in the meatgrinder has been demonstrated both by computer simulation and by practical experimentation. The results of this work were presented in Chapter 4, along with a certain amount of discussion and theoretical analysis.

This Chapter builds on the material already given in Chapter 4. It is shown that transformer action clamping (TAC) is detrimental to the meatgrinder efficiency only in the sense that it slows down the energy transfer process, thereby allowing greater resistive losses. The Chapter also includes a discussion of design implications in terms of maximising the speed of energy transfer and protecting the circuit components.

5.2 EXTERNAL INDUCED VOLTAGES WITHOUT CLAMPING

5.2.1 Theory

In the circuit of figure 5.1, L_1 and L_2 are the coil sections involved in the energy transfer process, and L_3 is the section previously switched out. (This notation means that the coil sections are not numbered consecutively from left to right, but is used so that L_1 and L_2 are the sections still in circuit. Note also that L_2 and L_3 could be the total inductance of several sections in series.)

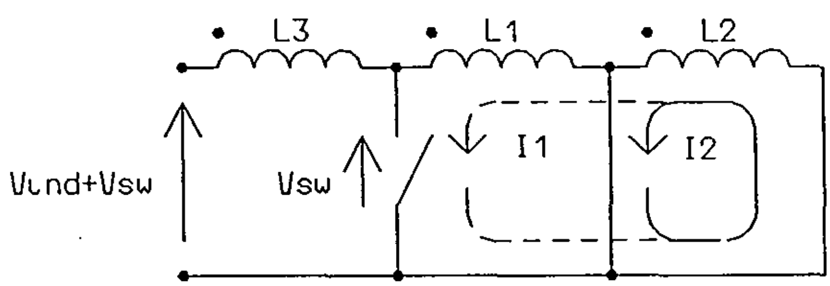
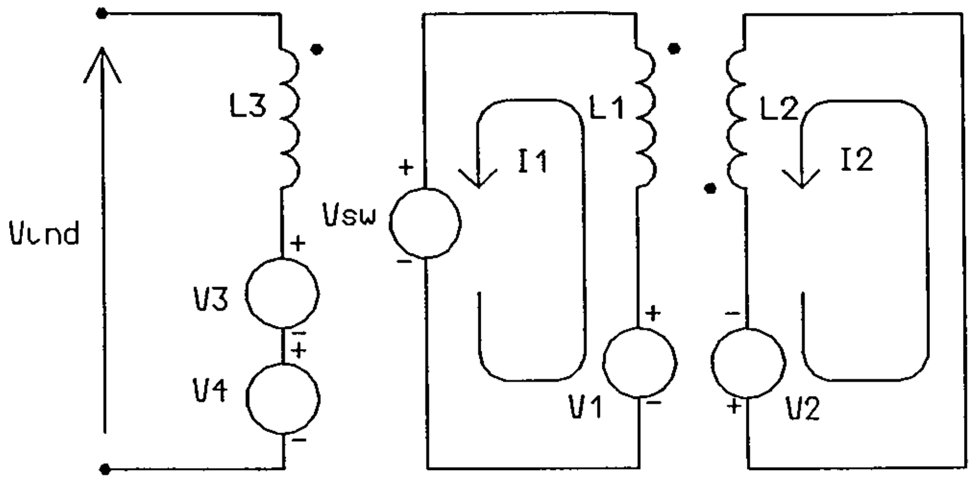


Figure 5.1 Meatgrinder Circuit Showing External Induced Voltage

The voltage across the opening switch is V_{sw}, whilst V_{ind} is the voltage induced across L_3 by transformer action during the energy transfer. The electrical connection means that the open-circuit end of L_3 rises to a voltage (V_{ind}+V_{sw}).

The relationship between V_{ind} and V_{sw} can be found by analysing the equivalent circuit shown in figure 5.2. This is an extension of figure 4.16, with an additional loop added to account for L_3 and extra voltage sources to account for the extra mutual inductances. It is assumed that no current is flowing in L_3.



V1 = M12 dI2/dt

V2 = M12 dI1/dt

V3 = M13 dI1/dt

V4 = M23 dI2/dt

Figure 5.2 Equivalent Circuit of Figure 5.1

Summing voltages around the L_1 and L_2 loops yields the expressions given in equations (4.2) and (4.3) respectively. Summing voltages around the L_3 loop yields

$$V_{ind} = M_{13} \frac{di_1}{dt} + M_{23} \frac{di_2}{dt} \qquad (5.1)$$

Using equation (4.3) to eliminate di_2/dt from equation (5.1) leads to

$$V_{ind} = \left[M_{13} - M_{23} \frac{M_{12}}{L_2} \right] \frac{di_1}{dt} \tag{5.2}$$

or

$$V_{ind} = \sqrt{L_1 L_3}\,(k_{13} - k_{12}k_{23}) \frac{di_1}{dt} \tag{5.3}$$

Finally, substituting for di_1/dt from equation (4.5) gives

$$V_{ind} = V_{sw} \sqrt{\frac{L_3}{L_1}} \left[\frac{k_{13} - k_{12}k_{23}}{1 - k_{12}^2} \right] \tag{5.4}$$

Thus, because L_3 is connected electrically to L_1, the total open-circuit voltage produced is

$$V_{o/c} = V_{sw} \left[\sqrt{\frac{L_3}{L_1}} \left[\frac{k_{13} - k_{12}k_{23}}{1 - k_{12}^2} \right] + 1 \right] \tag{5.5}$$

5.2.2 Verification

Equation (5.5) was implemented in the Fortran program "Ideal" described in Appendix E. As the sample output shows, the program predicts an open-circuit voltage of 318V with respect to ground at the end of L_1 (that is, the end not connected to L_2) during the second energy transfer in the six-step meatgrinder.

The experimental waveform of this same voltage was shown in figure 4.17(a). Although the mean value of the voltage can only be estimated, it does appear to be approximately of the right order.

The simulated waveform V(21) in figure 4.20 is free from oscillation. The open-circuit induced voltage as measured from this waveform is 320V - very close indeed to the theoretical value from the "Ideal" program.

5.3 INVESTIGATION OF EXTERNAL TRANSFORMER ACTION CLAMPING

5.3.1 Introduction

As described in Chapter 4, external transformer action clamping (ETAC) occurs in the last four energy transfers of the six-step meatgrinder. The current waveforms in figure 4.19 clearly show the induced currents flowing in previously switched-out coil sections as the MOV across TR2 breaks down. (Some current also flows via the MOVs across TR3 and TR4.)

Giorgi [27] was obviously aware of ETAC. He states that in his experiment he obtained the same current multiplication when ETAC occurred as when it was somehow deliberately prevented. In later work with Long et al [43], regarding an induced voltage in a previously switched-out section, it is stated that,

> "This voltage could cause a current to flow....which would cause energy to flow backward in the system and reduce the efficiency."

An analysis was carried out in an attempt to clarify whether or not ETAC degrades efficiency. The analysis is for an ideal circuit and is described below.

5.3.2 Algebraic Analysis

Figure 5.3 shows the circuit to be considered. It will be assumed that the first energy transfer has been completed and that ETAC occurs during the second transfer.

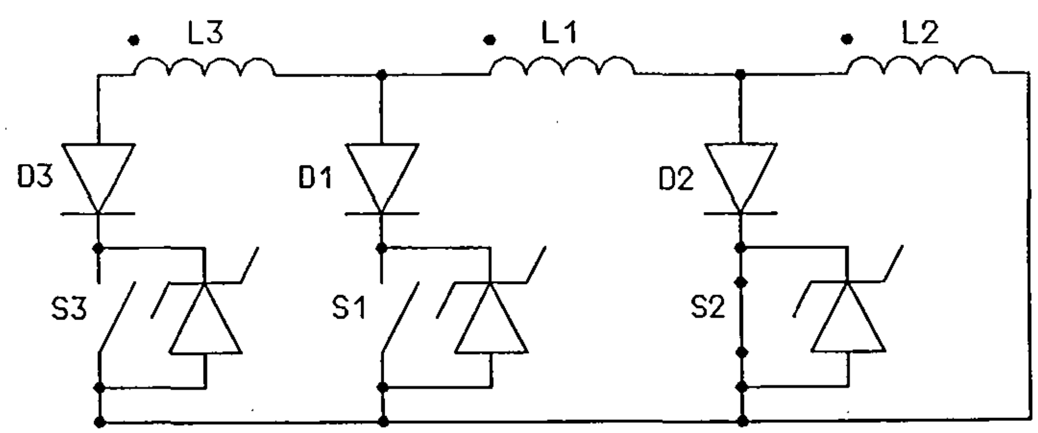


Figure 5.3 Two-Step Meatgrinder With Notation as Used for ETAC Analysis

Assuming that voltage clamping is ideal, the waveforms for the second transfer are as presented in figure 5.4. (Waveforms like this can be seen, for example, in the third energy transfer in the six-step meatgrinder - see figures 4.18 to 4.21.) The transfer can conveniently be divided into two parts, referred to as phase one and phase two respectively. It can be seen from figure 5.4 that phase one ends when the current in the clamp device across the opening switch has fallen to zero. Until this instant, the voltage across the opening switch is constrained to be the clamp voltage of the clamp device across it. When the clamp device no longer carries current, the voltage falls to a value dictated by the clamp voltage across S3 and the inductances of the meatgrinder coil.

Phase two of the energy transfer is complete when the current in D3 reaches zero. (It should be remembered that throughout the process, current flow in L_3 or D3 is via the clamp device across S3, rather than via the switch itself.)

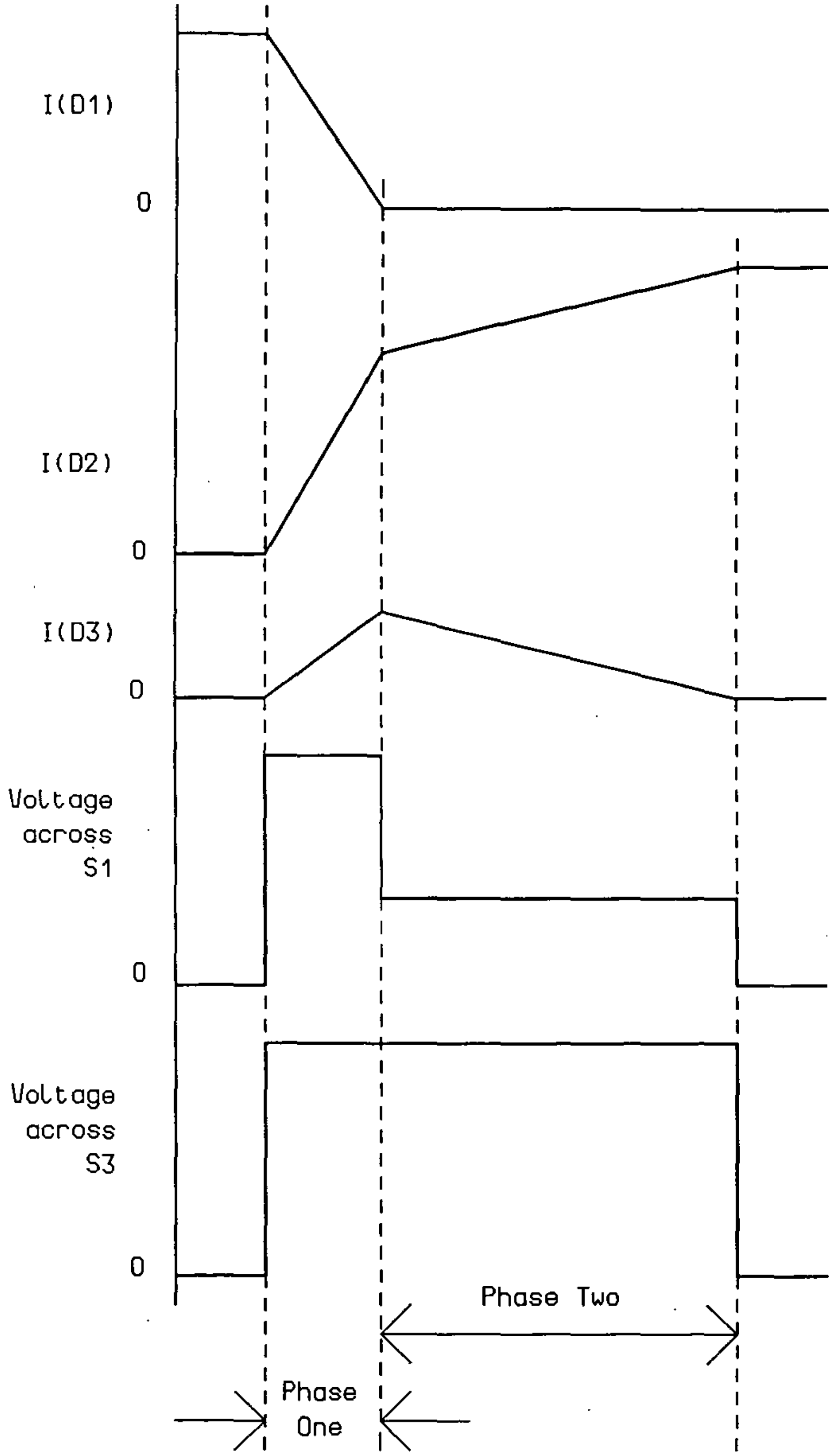
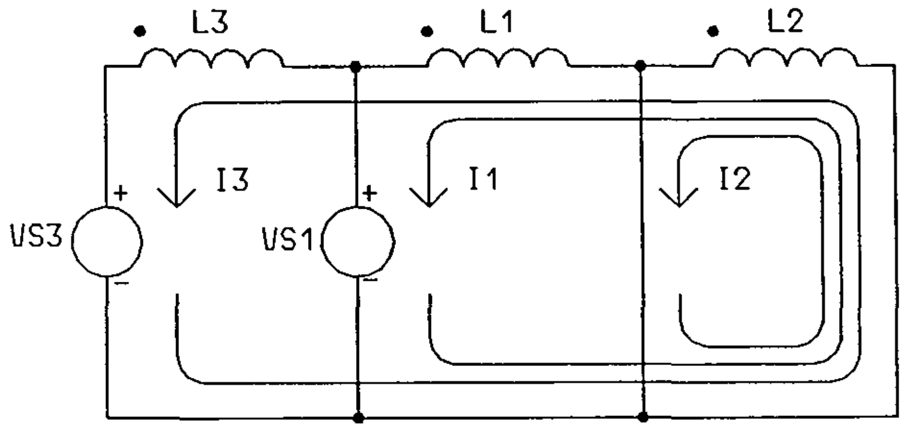


Figure 5.4 Typical Waveforms for Second Transfer of Two-Step Meatgrinder

5.3.2.1 Phase One

In the first phase, current flows in all three inductors, as indicated in figure 5.5. The voltages across S1 and S3 are both at their clamp values and are represented as voltage sources.



(Blocking diodes omitted for clarity)

Figure 5.5 Current Flow During Phase One of ETAC

The current paths form three overlapping loops, and when the voltages are summed around each of these, the expressions obtained are:

(i) Loop including L_2 only:

$$(L_2+M_{12})\frac{di_1}{dt} + L_2\frac{di_2}{dt} + (L_2+M_{12}+M_{23})\frac{di_3}{dt} = 0 \qquad (5.6)$$

(ii) Loop including both L_1 and L_2:

$$(L_1+L_2+2M_{12})\frac{di_1}{dt} + (L_2+M_{12})\frac{di_2}{dt} + (L_1+L_2+2M_{12}+M_{13}+M_{23})\frac{di_3}{dt}$$

$$+ V_{S1} = 0 \qquad (5.7)$$

(iii) Loop including L_1, L_2 and L_3:

$$(L_1+L_2+2M_{12}+M_{13}+M_{23})\frac{di_1}{dt} + (L_2+M_{12}+M_{23})\frac{di_2}{dt}$$

$$+ (L_1+L_2+L_3+2M_{12}+2M_{13}+2M_{23})\frac{di_3}{dt} + V_{S3} = 0 \qquad (5.8)$$

Equations (5.6), (5.7) and (5.8) respectively can be abbreviated as

$$Ax_1 + Bx_2 + Cx_3 = 0 \qquad (5.9)$$

$$Dx_1 + Ax_2 + Ex_3 + F = 0 \qquad (5.10)$$

$$Ex_1 + Cx_2 + Hx_3 + J = 0 \qquad (5.11)$$

where

$x_1 = di_1/dt$

$x_2 = di_2/dt$

$x_3 = di_3/dt$

$A = L_2+M_{12}$

$B = L_2$

$C = L_2+M_{12}+M_{23}$

$D = L_1+L_2+2M_{12}$

$E = L_1+L_2+2M_{12}+M_{13}+M_{23}$

$F = V_{S1}$

$H = L_1+L_2+L_3+2M_{12}+2M_{13}+2M_{23}$

$J = V_{S3}$

and they are simultaneous equations which can be manipulated to solve for the unknowns x_1, x_2 and x_3.

(i) Solution for x_1 (di_1/dt)

Eliminating x_2 and x_3 from equations (5.9) to (5.11) leads to

$$[(CE-AH)(A^2-BD) - (CD-AE)(AC-BE)]x_1 - [(CF-AJ)(AC-BE) + BF(CE-AH)] = 0 \qquad (5.12)$$

The algebra required to convert this expression back to the original notation is extensive but not complex, and it is not reproduced here. The final result, adopting some further shorthand, is

$$\frac{di_1}{dt} = \frac{V_{S3}X - V_{S1}Y}{L_1L_3\ (2k_{12}k_{13}k_{23} - k_{13}^2 - k_{12}^2 - k_{23}^2 + 1)} \qquad (5.13)$$

where

$$X = \sqrt{L_1L_3}(k_{13} - k_{12}k_{23}) + L_1(1 - k_{12}^2)$$

$$Y = 2\sqrt{L_1L_3}(k_{13} - k_{12}k_{23}) + L_1(1 - k_{12}^2) + L_3(1 - k_{23}^2)$$

An initial check on the validity of this expression can be carried out by making V_{S3} equal to the open-circuit value given by equation (5.5). For this purpose, equation (5.5) can be rearranged as

$$V_{o/c} = V_{sw}\left[\frac{\sqrt{L_1L_3}(k_{13} - k_{12}k_{23}) + L_1(1 - k_{12}^2)}{L_1(1 - k_{12}^2)}\right] \qquad (5.14)$$

and inserting this result into equation (5.13) gives

$$\frac{di_1}{dt} = \frac{V_{S1}}{L_1(1 - k_{12}^2)}\ [X^2 - YL_1(1 - k_{12}^2)]\ /\ W \qquad (5.15)$$

where

$$W = L_1L_3(2k_{12}k_{13}k_{23} - k_{13}^2 - k_{12}^2 - k_{23}^2 + 1)$$

The definitions of X and Y are as given for equation (5.13). By expanding the expression

$$X^2 - YL_1(1 - k_{12}^2)$$

it can be shown that

$$X^2 - YL_1(1 - k_{12}^2) + W = 0$$

and thus that

$$\frac{di_1}{dt} = - \frac{V_{S1}}{L_1(1 - k_{12}^2)} \qquad (5.16)$$

which is identical to equation (4.5), where there is no ETAC.

(ii) Solution for x_2 (di_2/dt)

The unknown x_2 is found in a manner similar to that for x_1 above, using equations (5.9) to (5.11). Extensive manipulation of the expressions obtained leads to a final result of

$$\frac{di_2}{dt} = \frac{V_{S3}[M-L] + V_{S1}[J+K+L-M]}{WL_2(1 - k_{12}^2)} \qquad (5.17)$$

where

$$J = L_2 L_3 (1 - k_{12}^2 - k_{23}^2 + k_{12}^2 k_{23}^2)$$

$$K = L_3 \sqrt{L_1 L_2} (k_{12} - k_{13} k_{23} - k_{12}^3 + k_{12}^2 k_{13} k_{23})$$

$$L = L_2 \sqrt{L_1 L_3} (k_{13} - k_{12} k_{23} - k_{12}^2 k_{13} + k_{12}^3 k_{23})$$

$$M = L_1 \sqrt{L_2 L_3} (k_{23} - k_{12} k_{13} - k_{12}^2 k_{23} + k_{12}^3 k_{13})$$

$$W = L_1 L_3 (2k_{12} k_{13} k_{23} - k_{13}^2 - k_{12}^2 - k_{23}^2 + 1)$$

(iii) Solution for x_3 (di_3/dt)

Starting from equations (5.9) to (5.11), x_3 may be found in terms of x_1 as

$$x_3 = - \frac{V_{S1} + L_1(1 - k_{12}^2)x_1}{X} \tag{5.18}$$

where X is as defined for equation (5.13).

The validity of this equation can be confirmed by setting x_1 (di_1/dt) to the value it assumes when ETAC does not occur (as given by equation (5.16)). When this is done x_3 becomes zero, because there is then no current flow in the switched-out loop. Completing the solution of equations (5.13) and (5.18) gives

$$\frac{di_3}{dt} = - \left[\frac{V_{S3} L_1 (1 - k_{12}^2) - V_{S1} X}{W} \right] \tag{5.19}$$

with X and W again as previously defined.

(iv) Final Current Values

At the end of phase one, i_1 has fallen to zero, and for a linear energy transfer the duration of phase one δt_1 can therefore be found from equation (5.13). With the initial value of i_1 denoted by I_1, the result obtained is

$$\delta t_1 = I_1 \frac{W}{V_{S3}X - V_{S1}Y} \tag{5.20}$$

The changes in i_2 and i_3 during phase one may now be found by substituting equation (5.20) into equations (5.17) and (5.19) respectively. This leads to

$$\delta i_2 = \left[\frac{V_{S3}[M-L] + V_{S1}[J+K+L-M]}{V_{S3}[L+N] - V_{S1}[2L+J+N]} \right] I_1 \tag{5.21}$$

where J, K, L and M are as defined for equation (5.17) and
$N = L_1 L_2 (1 + k_{12}^4 - 2k_{12}^2)$

and

$$\delta i_3 = - \left[\frac{V_{S3}L_1(1 - K_{12}^2) - V_{S1}X}{V_{S3}X - V_{S1}Y} \right] I_1 \tag{5.22}$$

(v) Numerical Examples

Equations (5.21) and (5.22) may be tested by comparing the values they provide for the changes in i_2 and i_3 during phase one of an energy transfer with the values obtained from a circuit simulation using PSpice (see Appendix C). To achieve this, the equations are implemented as Fortran routines in the program "Ideal" (see Appendix E).

The first example is a two-step meatgrinder (see figure 5.3) designed purely for test purposes. The PSpice input file giving the circuit parameters is given in Appendix C.

The results for three different cases are given in table 5.1, with the corresponding current and voltage waveforms being shown in figures 5.6 to 5.8. Table 5.1 also includes the results of a similar test carried out for the third energy transfer of the (simulated) six-step meatgrinder. The waveforms for this test are given in figure 5.9.

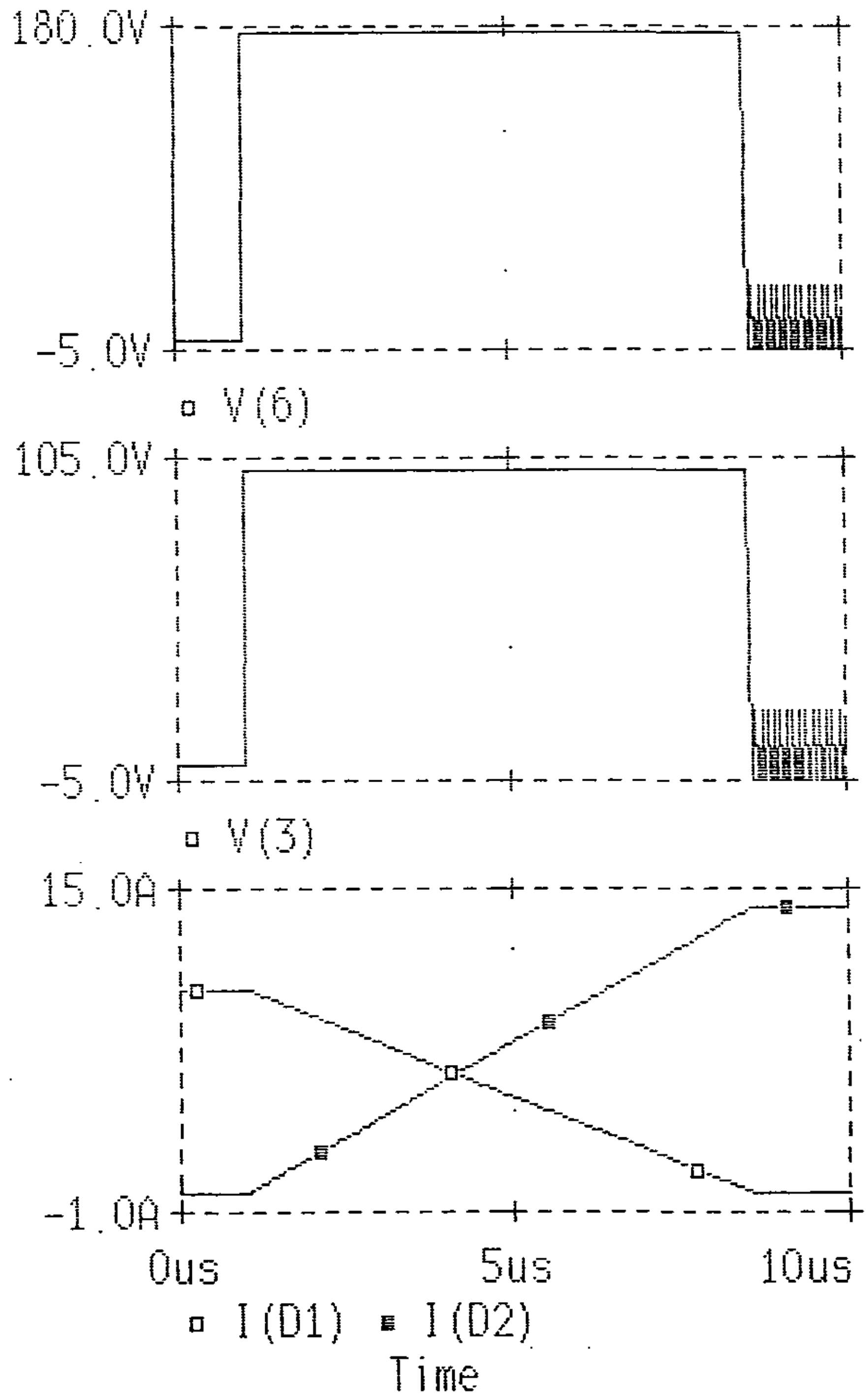


Figure 5.6 Simulated Waveforms for ETAC Investigation: V_{S3} = 175.4V

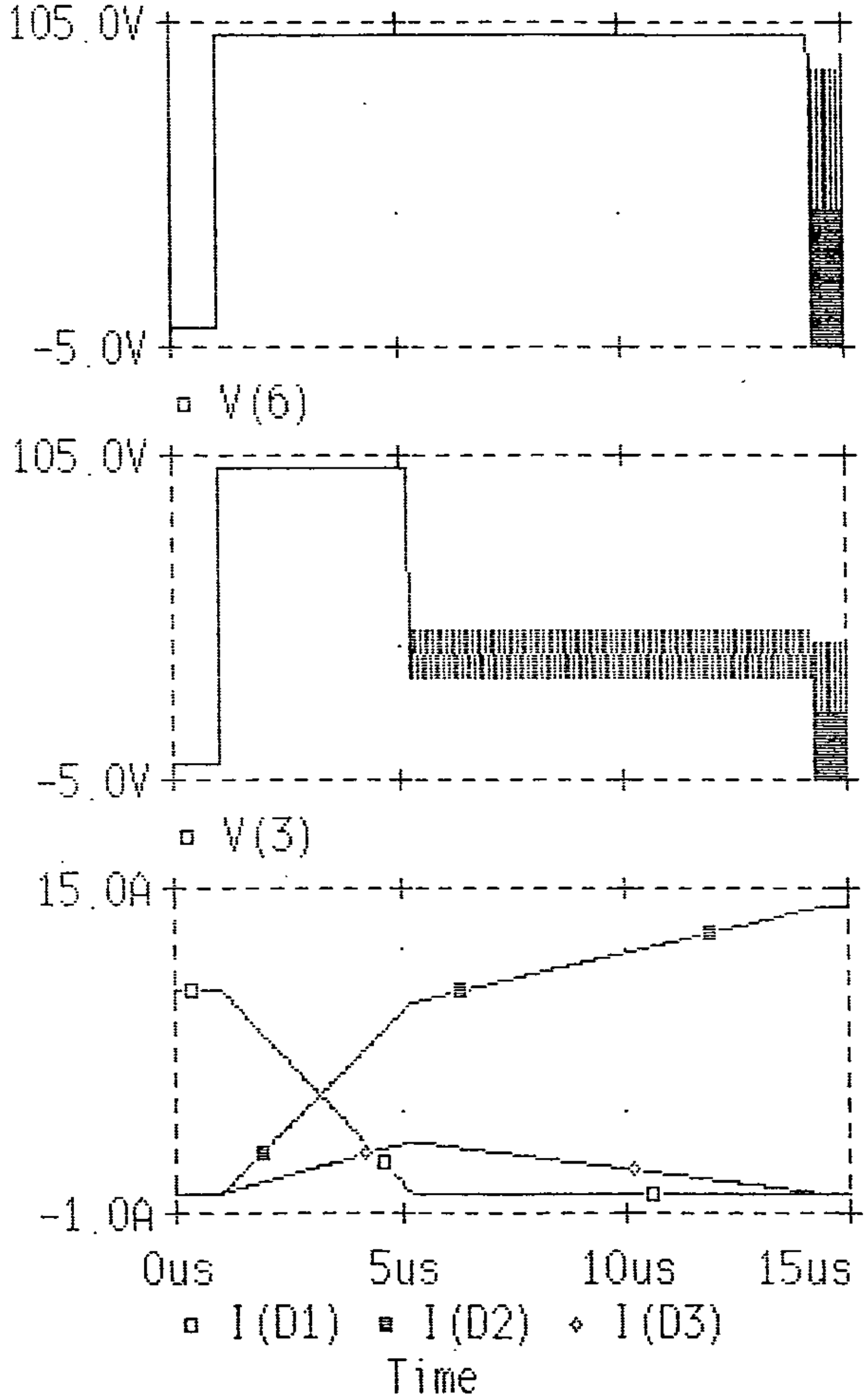


Figure 5.7 Simulated Waveforms for ETAC Investigation: V_{S3} = 100V

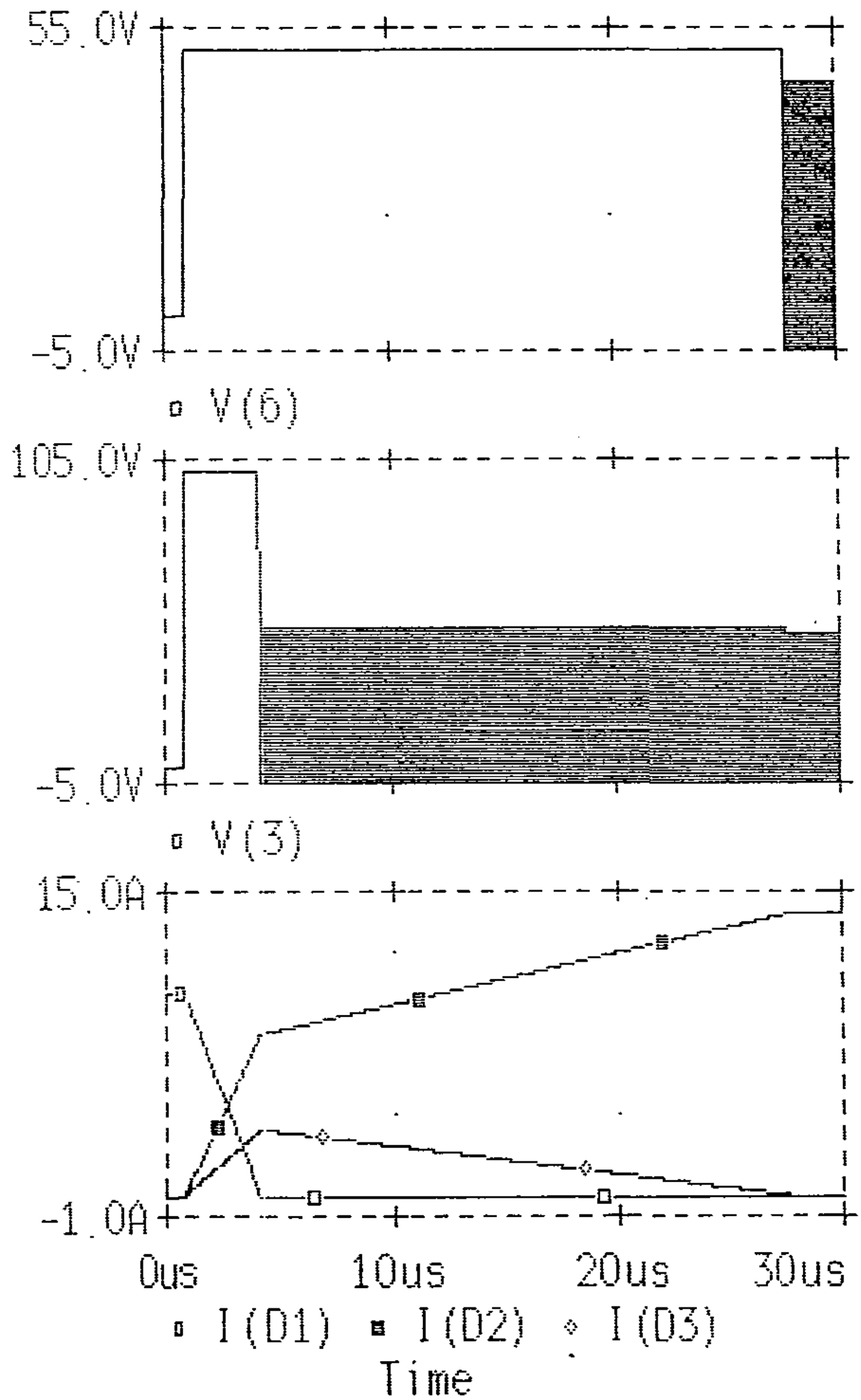


Figure 5.8 Simulated Waveforms for ETAC Investigation: V_{S3} = 50V

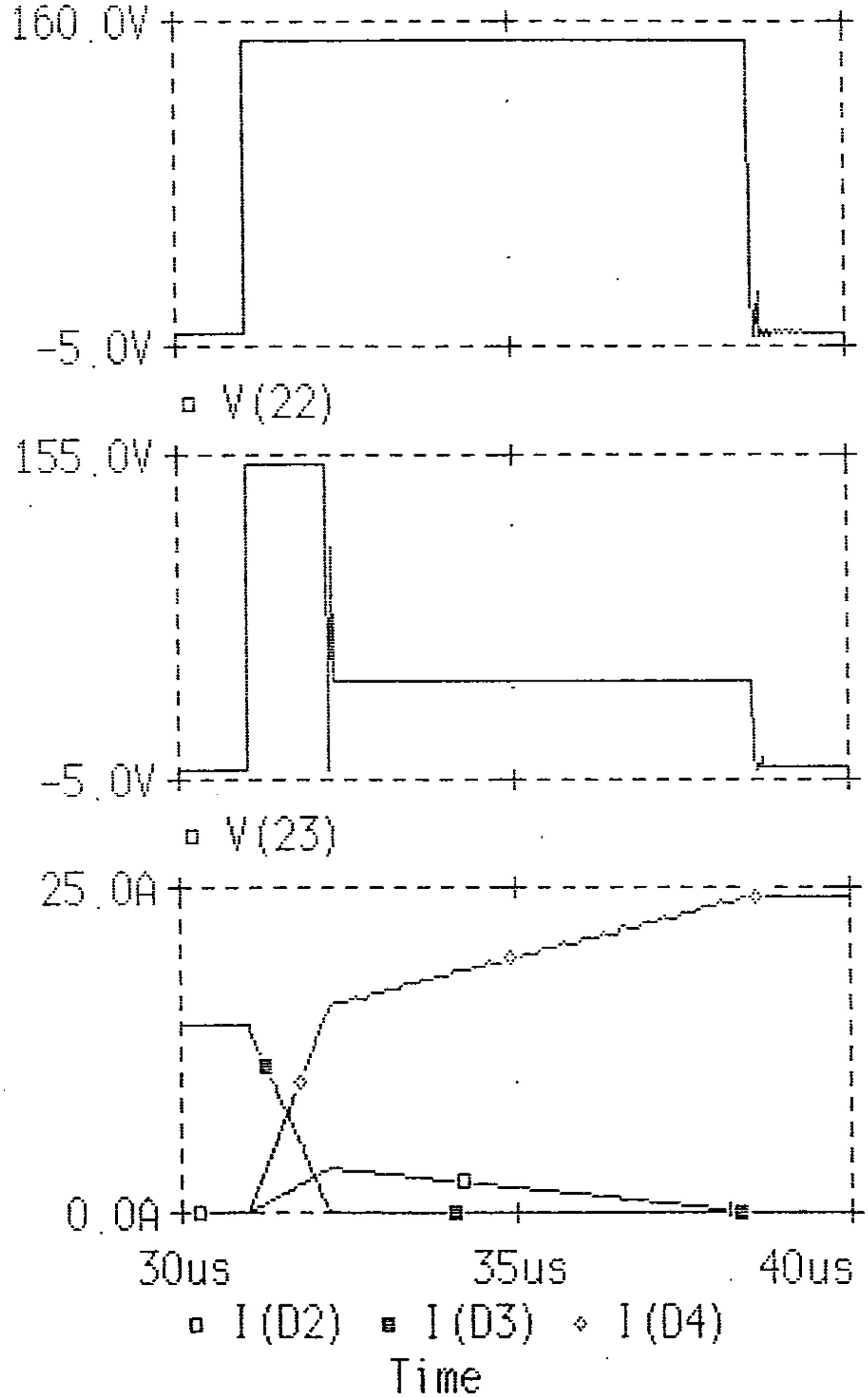


Figure 5.9 Simulated Waveforms From Third Energy Transfer of Six-Step Meatgrinder

CLAMPING VOLTAGE V_{S3} (V)	PHASE ONE DURATION (μs)		FINAL VALUE OF i_2 AFTER PHASE ONE (A)		FINAL VALUE OF i_3 AFTER PHASE ONE (A)	
	THEORY	SIM'N	THEORY	SIM'N	THEORY	SIM'N
175.4*	7.52	7.57	14.08	14.04	0.00	0.00
100.0	4.18	4.22	9.35	9.37	2.52	2.52
50.0	3.24	3.26	8.00	8.00	3.24	3.23
150.0**	1.16	1.26	16.19	16.02	3.40	3.38

(SIM'N=SIMULATION)

* open-circuit value calculated from equation (5.5)

** result from third transfer of six-step meatgrinder

Table 5.1 Comparison of Results from Theoretical Equations and Computer Simulation

5.3.2.2 Phase Two

In phase two the circuit is as shown in figure 5.10. Clearly this is the same configuration as for a normal energy transfer except that the initial value of i_2 is now not zero.

For convenience, let the total inductance of L_3 and L_1 be denoted as L_a, that is

$$L_a = L_1 + L_3 + 2M_{13}$$

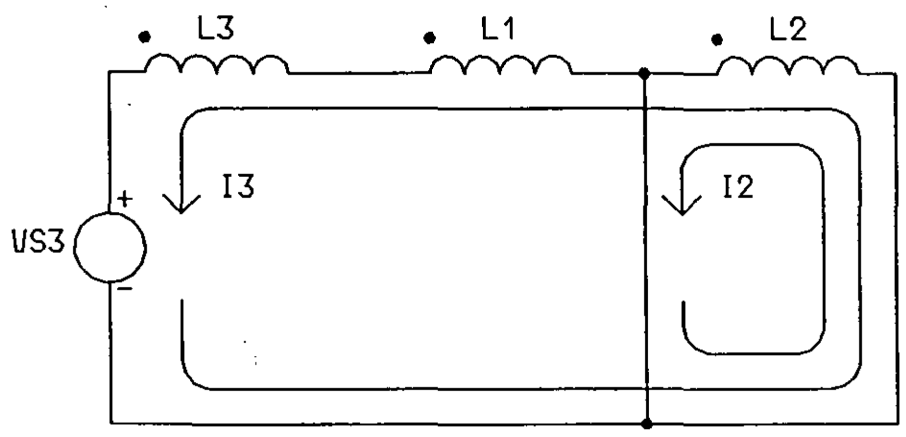


Figure 5.10 Current Flow During Phase Two of ETAC

Summing voltages around the two loops of figure 5.10 yields

$$Ax_3 + Bx_2 = 0 \quad (5.23)$$

$$Cx_3 + Ax_2 + D = 0 \quad (5.24)$$

where

$x_2 = di_2/dt$

$x_3 = di_3/dt$

$A = L_2 + M_{a2}$

$B = L_2$

$C = L_a + L_2 + 2M_{a2}$

$D = V_{S3}$

Note also that $M_{a2} = M_{12} + M_{32}$.

Eliminating x_2 from equations (5.23) and (5.24) gives

$$x_3 = \frac{di_3}{dt} = - \frac{V_{S3}}{L_a(1 - k_{a2}^2)} \tag{5.25}$$

From equation (5.23), if the initial value of i_3 is I_3 and the transfer is linear, the change in i_2 during phase two is

$$\delta i_2 = - \left[\frac{L_2 + M_{a2}}{L_2} I_3 \right] \tag{5.26}$$

which is of the same form as equation (2.1). The minus sign indicates that i_2 rises as i_3 falls, and the value of I_3 is given by equation (5.22).

5.3.2.3 **Total Current Increase**

The final value of i_2 is found by adding the separate increases given by equations (5.21) and (5.26). This is again an exercise which is laborious rather than complex.

Appendix F gives a sample of the working necessary to obtain the final result. The Appendix also describes how the same task was subsequently performed in a fraction of the time using REDUCE, a computer algebra package.

The final result from the algebraic manipulation is

$$\frac{I_2}{I_1} = \frac{L_2 + M_{12}}{L_2} \qquad (5.27)$$

where

I_1 = initial value of i_1

I_2 = final value of i_2

which is identical to the current multiplication when ETAC does not occur. Although this may seem surprising, it is logical when considered in terms of the conservation of flux linkage.

Equation (5.27) may be derived by applying the principle of flux linkage conservation to the loop containing L_2 only. This loop remains closed throughout the transfer and therefore flux linkage must be conserved. The final result depends only on the initial and final current paths and is unaffected by any intermediate state such as occurs with ETAC.

5.3.3 Computer Simulation

Table 5.1 above gives results which refer only to phase one of the second energy transfer in a simulated two-step meatgrinder (see circuit diagram, figure 5.3). (The first transfer is of no interest in this case because ETAC cannot occur.)

The same simulated circuit was used to test the validity of equation (5.27). Table 5.2 gives the maximum current obtained in L_2, which in all but the first test is the current at the end of phase two. In the first test there is only one "phase" because ETAC does not occur.

CLAMPING VOLTAGE V_{S3} (V)	DOES ETAC OCCUR?	MAXIMUM CURRENT (A)
175.4	no	14.04
100.0	yes	14.00
50.0	yes	13.92
25.0	yes	13.76

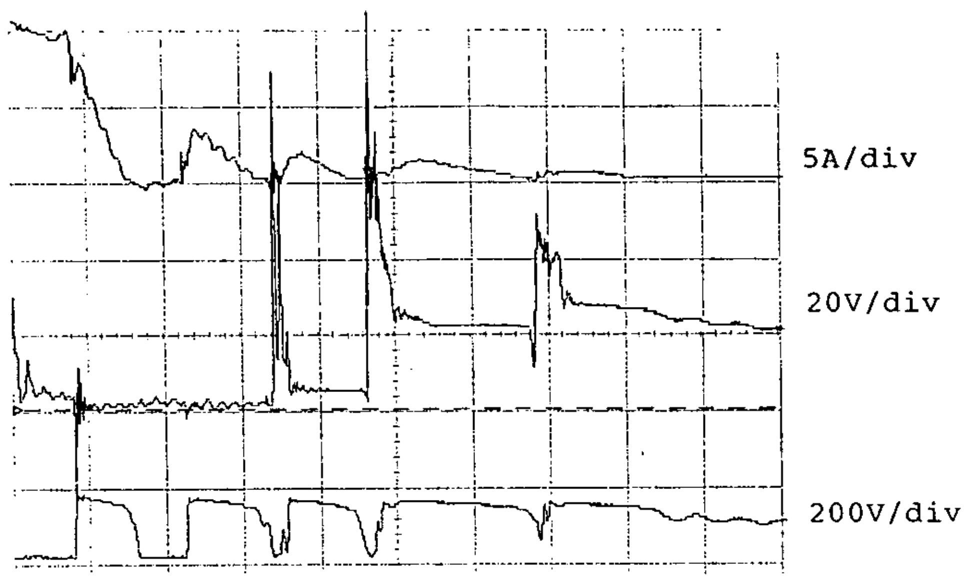
Table 5.2 Maximum Current in Simulated Two-Step Meatgrinder

It can be seen from table 5.2 that the final current is the same in each case, to within two percent, which confirms the validity of equation (5.27).

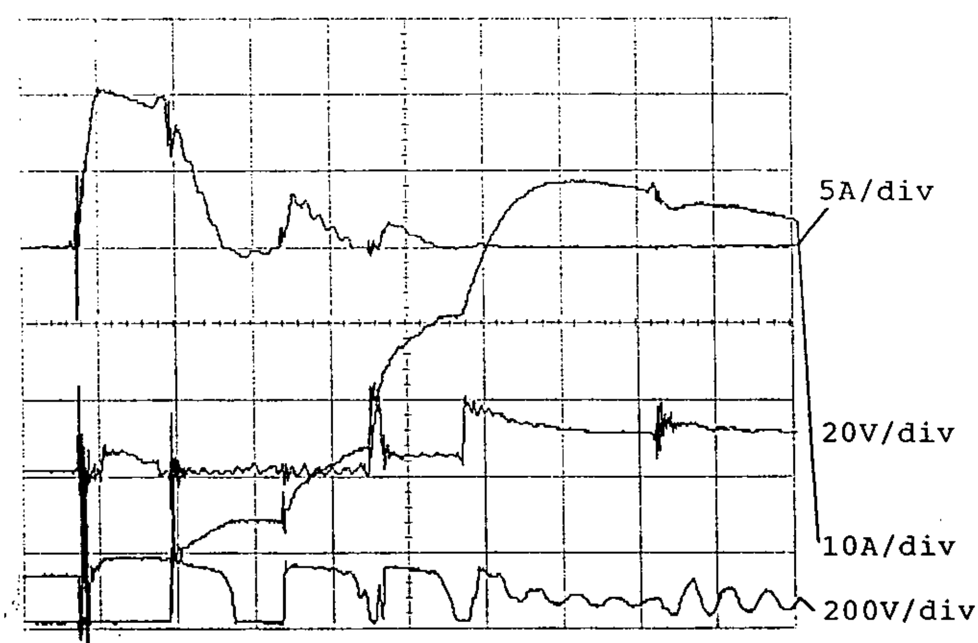
5.3.4 Experimental Results

The effect of preventing ETAC during the fifth step of the experimental six-step meatgrinder was investigated.

Figure 5.11(a) shows voltage and current waveforms in the unmodified circuit. It can be seen that for each energy transfer, the voltage across TR2 rises to its clamp value and current flows back through MOV2 as ETAC occurs.



(a) No extra clamp applied



(b) With 10V zener diode across TR5

Figure 5.11 Prevention of ETAC During Fifth Step of Six-Step Meatgrinder

All traces: Time 10μs/div

Figure 5.11(b) shows the effect of deliberately clamping the voltage across TR5 to a very low value with a 10V zener diode. There is now no current flow back through MOV2 during the fifth and sixth transfers; the fifth transfer becomes an "ordinary" one with no ETAC, and whilst ETAC again occurs during the sixth transfer, it is controlled by the voltage across TR5.

Clearly, however, the current multiplication at the fifth step is reduced, with the current rising to less than 60A compared to 70A previously. This is attributable to the effect of resistance, as described in Chapter 4. Comparing figures 5.11(a) and 5.11(b) it will be seen that the voltage across TR5 is lower throughout the transfer. Therefore, again as explained in Chapter 4, a longer time is required to transfer the energy. Whilst energy is transferred forward more slowly than before, the circuit resistance dissipates energy through copper loss at the same rate as before. Thus allowing extra time has no beneficial effect because the current levels off and then begins to fall.

The simulated two-step circuit referred to earlier has no coil resistance, but the constraints of the simulation program mean that there has to be a non-zero switch resistance in the on-state. (Specifically, the maximum value of R_{off}/R_{on} is 10^{12}.) This accounts for the very gradual fall in maximum current as the clamp voltage is reduced, because, as figures 5.6 to 5.8 show, reducing the clamp voltage increases the total time necessary for the energy transfer. The energy dissipated in the resistance increases because of the increased transfer time, thereby reducing the efficiency of the meatgrinder.

5.4 INVESTIGATION OF INTERNAL TRANSFORMER ACTION CLAMPING

It has been shown previously that ETAC does not degrade the efficiency of a meatgrinder because it has no effect on the initial and final flux linkages. If the same were true of ITAC, then it too would have no effect on the final current obtained.

5.4.1 Simulation of Negative ITAC

To investigate the effect of negative ITAC, the circuit of figure 5.12 was simulated, using the PSpice input file given in Appendix C. Zener diode DZ4 allows for simulation of negative ITAC when switched in by S4. This circuit arrangement (rather than simply replacing D3 with a zener diode) was found to be necessary in order to force PSpice to generate the correct initial conditions.

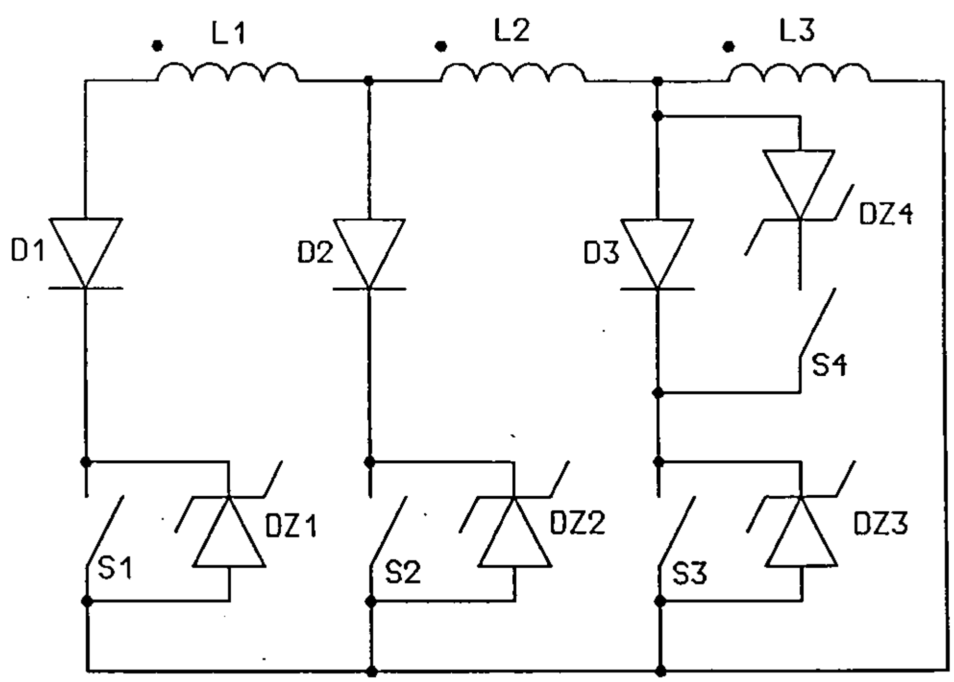


Figure 5.12 Simulated Two-Step Meatgrinder for Investigation of ITAC

Figure 5.13 shows the results when there is no ITAC. It can be seen that node 3 goes 150V negative during the first transfer and that the final current after the second transfer is 20A.

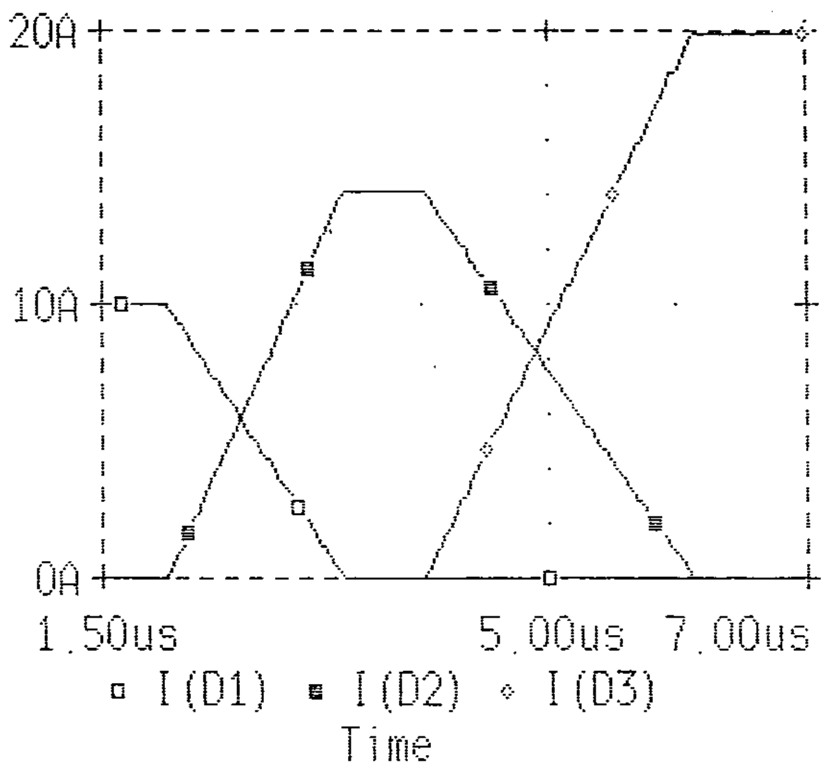
To force ITAC to occur, DZ4 is given a breakdown voltage of 50V and switched in when the first transfer starts. Figure 5.14(a) shows the negative current in DZ4 rising to a peak and then falling to zero. When it has done so, the current in D2 is at the same level as in the non-ITAC case - just over 14A.

It is clear that, as with ETAC, the transfer now occurs in two phases. As before, the first phase ends when the current in the first loop (that is, the loop with the opening switch) has fallen to zero. At the end of the second phase there is no current in DZ4 and the current path is identical to the non-ITAC case. Thus the same flux linkage considerations described for ETAC apply. The current multiplication is consequently unchanged.

The second transfer proceeds as normal, leading to a final current of 20A.

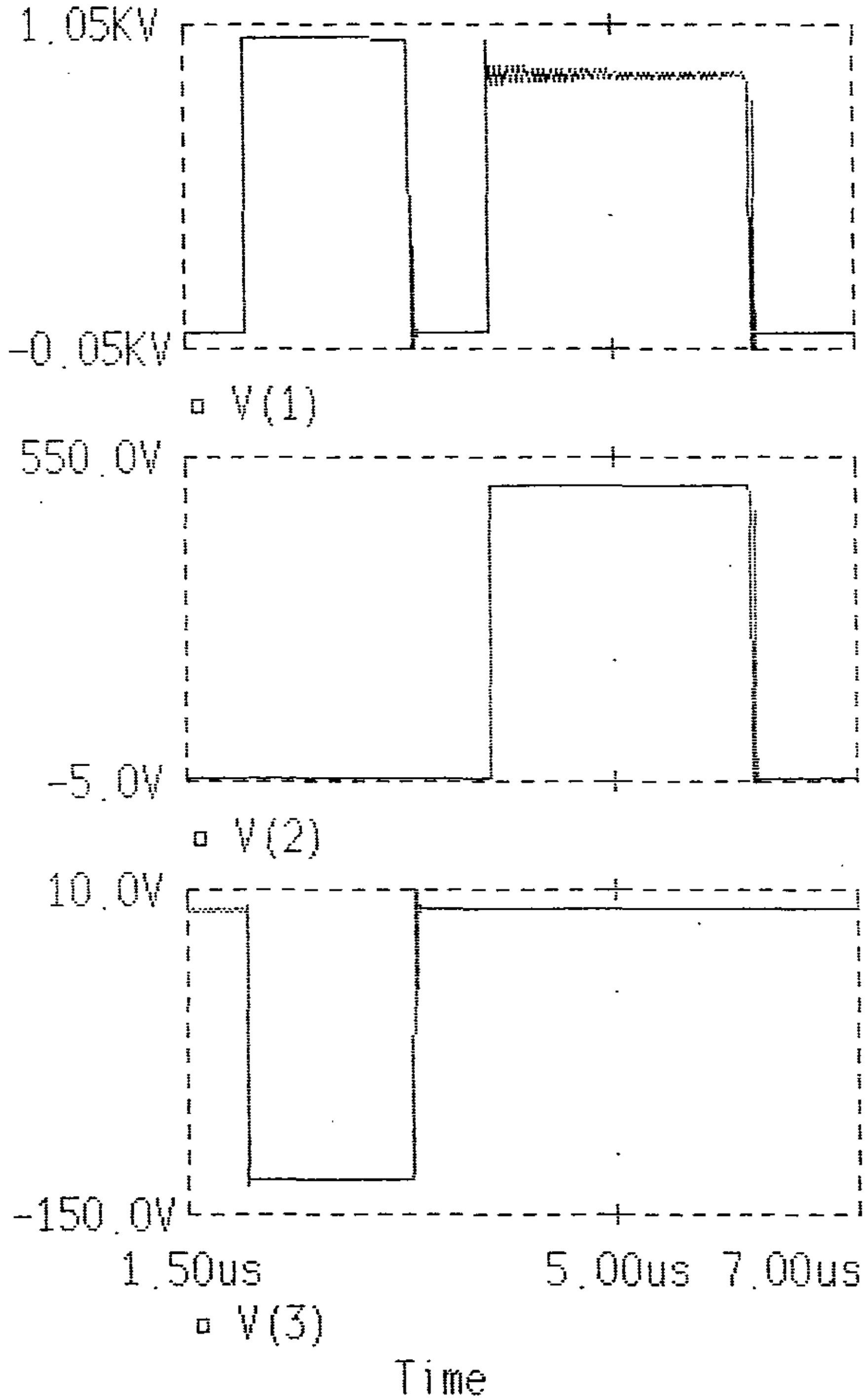
It will be noted from figure 5.14(a) that at the end of phase one of the first energy transfer, the current in D2 is higher than it is at the end of phase two (17A compared to 14A). This is because the clamp current through DZ4 also flows through D2.

A third simulation was carried out in which the second energy transfer was initiated at the end of phase one of the first transfer. The results are shown in figure 5.15, from which it can be seen that the final current is significantly less than 20A. This is not a resistance effect, however, because the whole process now takes less time. The reduced efficiency must therefore be due to the current still flowing in DZ4 when the



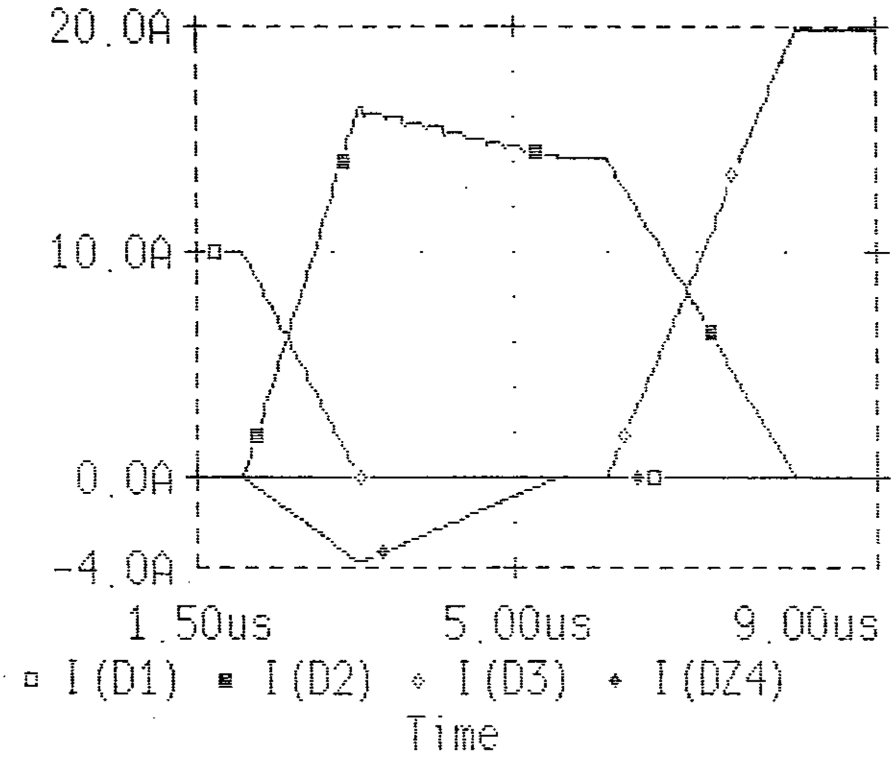
(a) Currents

Figure 5.13 Simulated Two-Step Meatgrinder - No ITAC



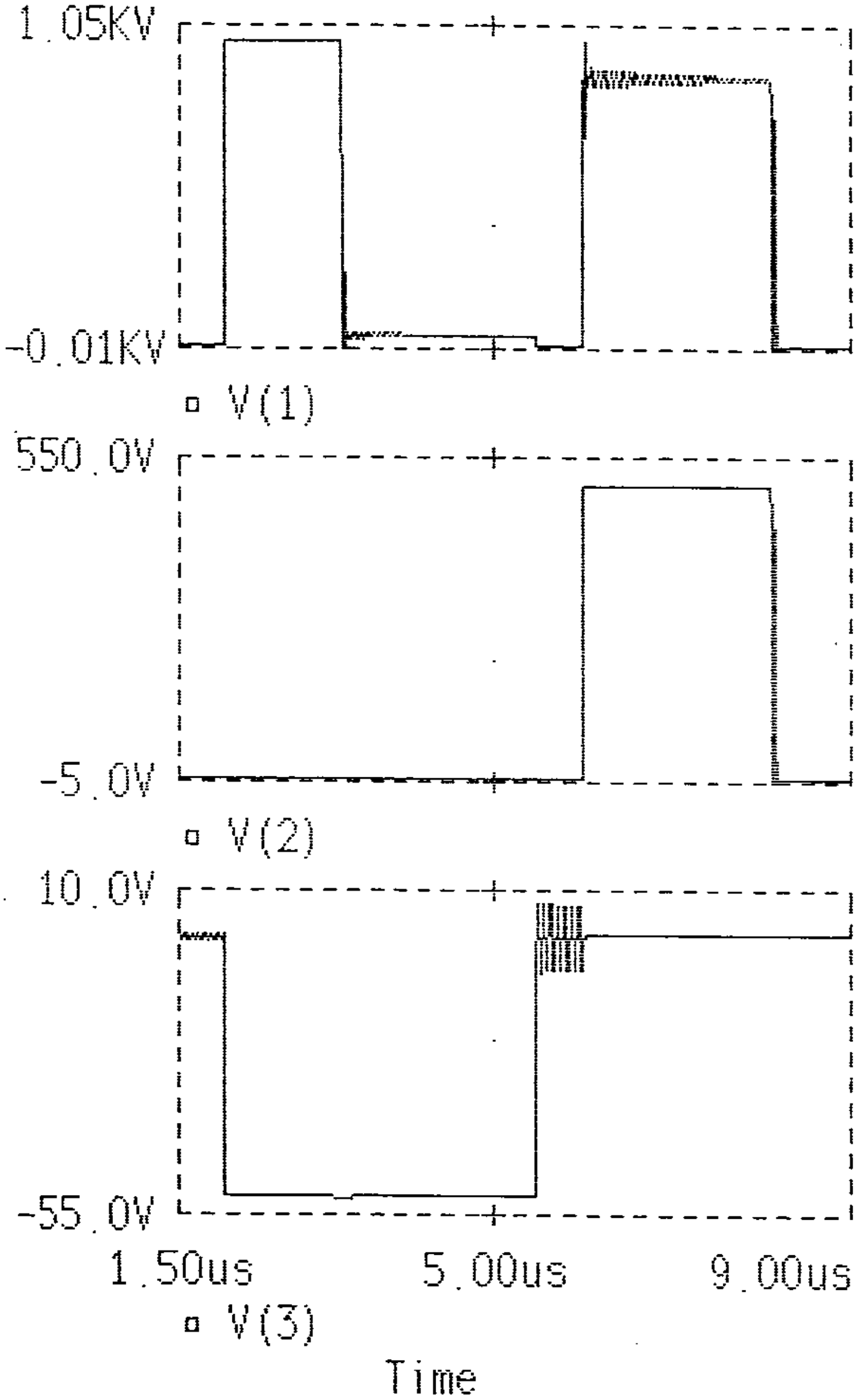
(b) Voltages

Figure 5.13 Simulated Two-Step Meatgrinder - No ITAC



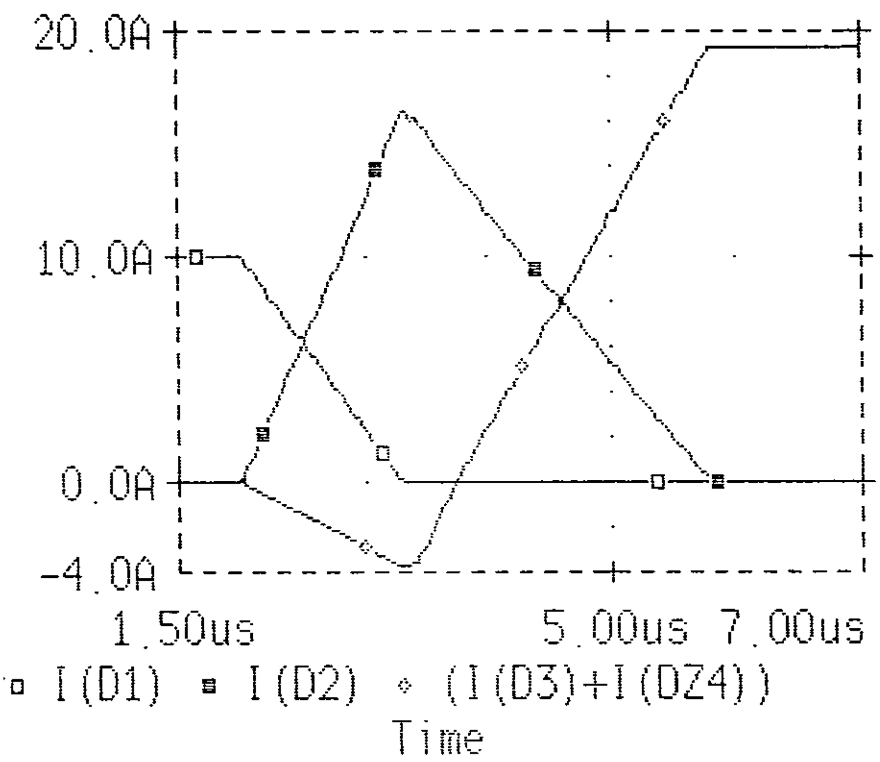
(a) Currents

Figure 5.14 Simulated Two-Step Meatgrinder With ITAC



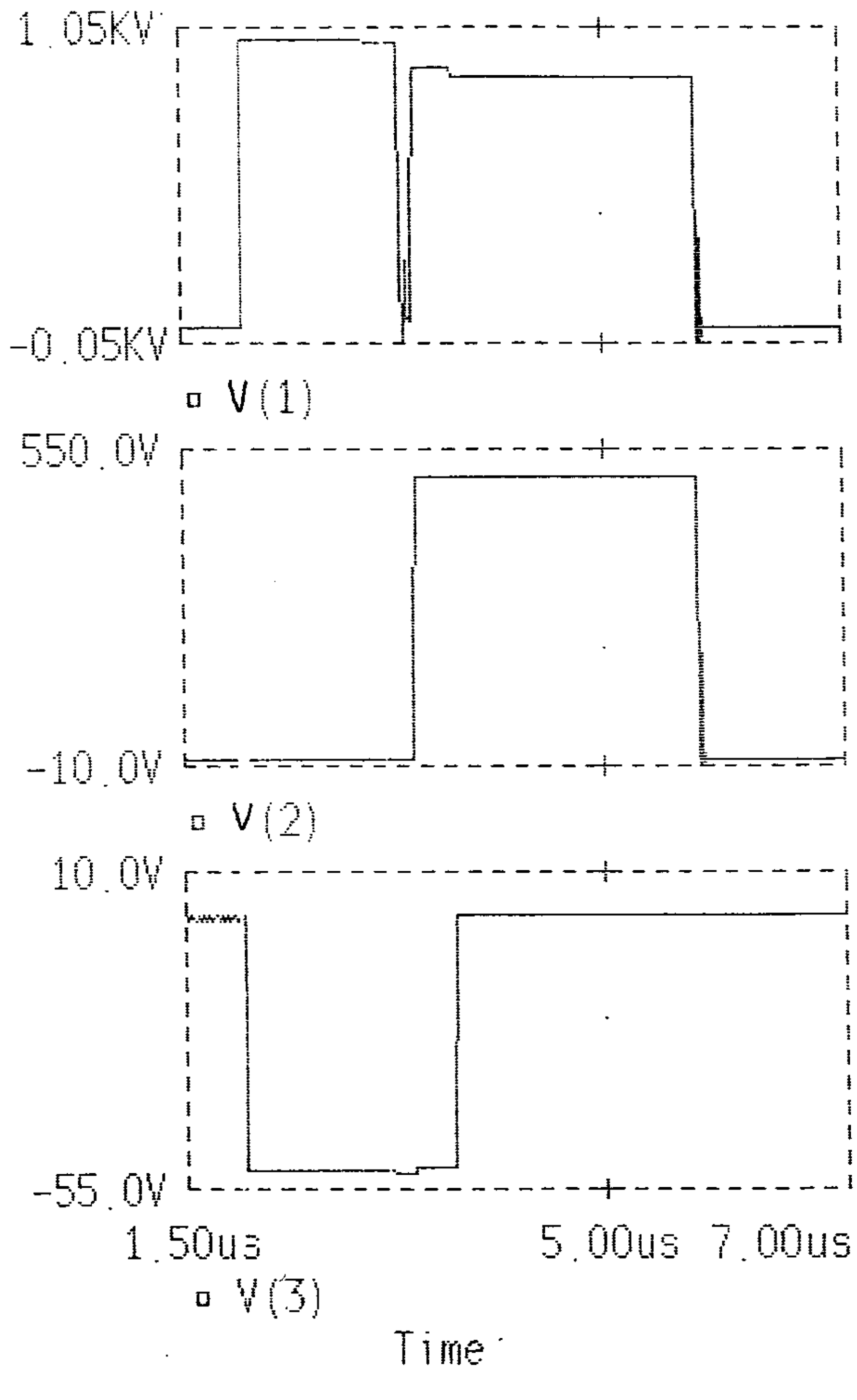
(b) Voltages

Figure 5.14 Simulated Two-Step Meatgrinder With ITAC



(a) Currents

Figure 5.15 Simulated Two-Step Meatgrinder With ITAC and Incorrect Timing for Second Transfer



(b) Voltages

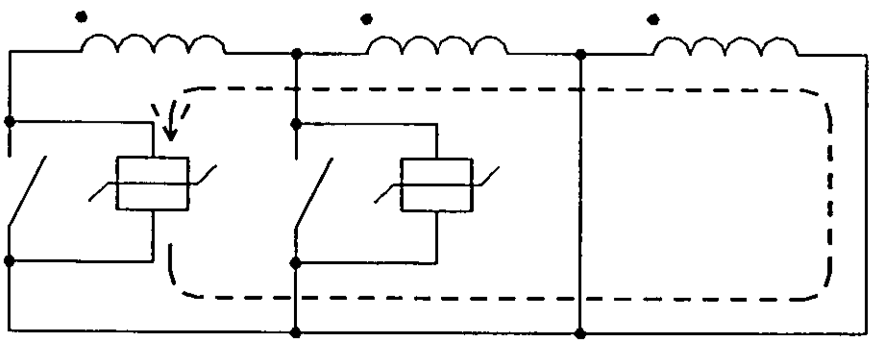
Figure 5.15 Simulated Two-Step Meatgrinder With ITAC and Incorrect Timing for Second Transfer

energy transfer is initiated. Effectively this means that the first energy transfer is incomplete when the second one starts and this leads to additional energy loss. (See Chapter 3 for the first discussion of this principle.)

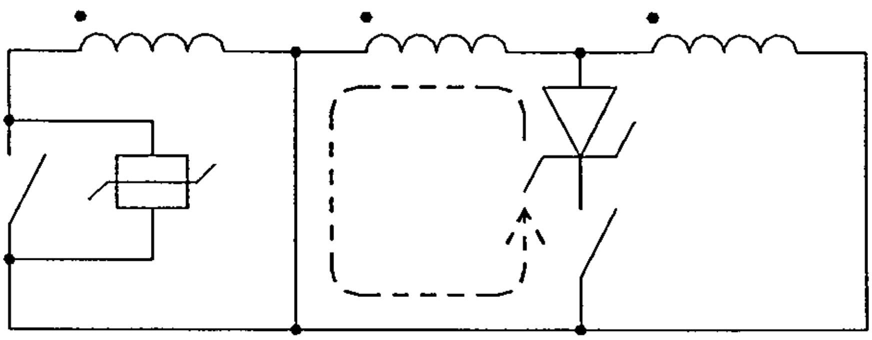
5.4.2 Positive ITAC

Positive ITAC can occur if an in-circuit inductance voltage causes the clamp device across one of the switches to break down. Clamp current would then flow around the loop indicated in figure 5.16(c).

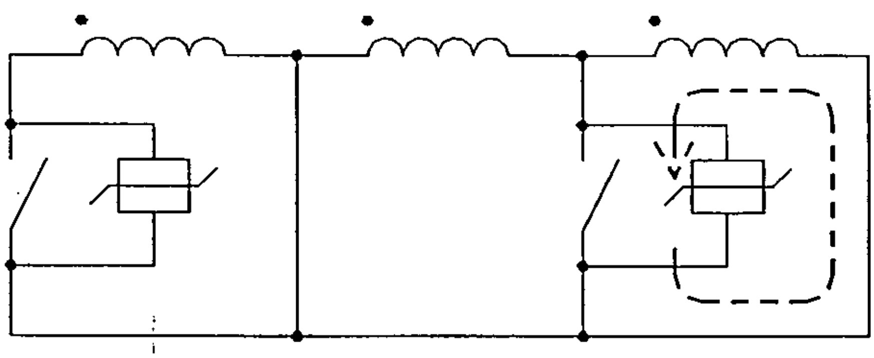
This phenomenon has not been investigated in detail because the principles discussed previously for ETAC and negative ITAC also apply to positive ITAC. In other words the energy transfer occurs in two phases, the second phase being complete when the clamp current has fallen to zero. At this point the circuit conditions are the same as in the non-ITAC case, and the current multiplication is identical.



(a) ETAC



(b) Negative ITAC



(c) Positive ITAC

Note: For clarity, only relevant diodes etc,. are shown. Arrows indicate path of clamp current.

Figure 5.16 Clamp Current Paths Related to TAC

5.5 EFFECT OF TRANSFORMER ACTION CLAMPING ON TRANSFER TIME

5.5.1 Analysis for ETAC

The analysis below again refers to the two-step meatgrinder shown in figure 5.3, although the result is applicable to any meatgrinder energy transfer where ETAC can occur. Much of the notation used is as defined previously, but in addition t_1, t_2 and t_3 are defined as:

t_1 = duration of second transfer with no ETAC
t_2 = duration of phase one of second transfer with ETAC
t_3 = duration of phase two of second transfer with ETAC

Assuming a linear energy transfer, t_1 is found from equation (5.16) to be

$$t_1 = I_1 \frac{L_1(1 - k_{12}^2)}{V_{S1}} \tag{5.28}$$

where I_1 is the initial value of i_1.

Similarly, t_2 follows from equation (5.13) and is

$$t_2 = I_1 \frac{-W}{V_{S3}X - V_{S1}Y} \tag{5.29}$$

(The minus sign results from the fact that $\delta i_1 = -I_1$ during phase one.)

Equation (5.25) gives t_3 in terms of I_3 as

$$t_3 = I_3 \frac{L_a(1 - k_{a2}^2)}{V_{S3}} \tag{5.30}$$

where I_3 is the value of i_3 at the beginning of phase two.

Now L_a was defined previously in section 5.3.2.2 as

$$L_a = L_1 + L_3 + 2M_{13} \tag{5.31}$$

and k_{a2} is the coupling coefficient between L_a and L_2. The constant term k_{a2} can be eliminated from equation (5.30) by considering the mutual inductance M_{a2}, which is, of course,

$$M_{a2} = M_{12} + M_{32} \tag{5.32}$$

where M_{12} and M_{32} are the mutual inductances between L_1 and L_2, and between L_3 and L_2 respectively.

Expressing the mutual inductances in equation (5.32) in terms of coupling coefficients and self inductances (e.g. $M_{12} = k_{12}\sqrt{L_1 L_2}$) yields an expression for k_{a2}. It is found subsequently that

$$1 - k_{a2}^2 = \frac{Y}{L_a} \tag{5.33}$$

where Y is as defined for equation (5.13).

Equation (5.30) can now be expanded, with the value of I_3 being obtained from equation (5.22) (the minus sign is unnecessary, since this refers to the change in I_3). The result obtained is

$$t_3 = I_1 \left[\frac{V_{S3}L_1(1 - k_{12}^2) - V_{S1}X}{V_{S3}X - V_{S1}Y} \right] \frac{Y}{V_{S3}} \qquad (5.34)$$

Adding equations (5.29) and (5.34) shows that the total transfer time when ETAC occurs is

$$t_2 + t_3 = \frac{I_1}{V_{S3}} \left[\frac{V_{S3}B - V_{S1}XY}{V_{S3}X - V_{S1}Y} \right] \qquad (5.35)$$

where

$$B = YL_1(1 - k_{12}^2) - W$$

In deriving equation (5.16) it was noted that

$$YL_1(1 - k_{12}^2) - W = X^2$$

and it therefore follows that equation (5.35) can be rewritten as

$$t_2 + t_3 = \frac{XI_1}{V_{S3}} \left[\frac{V_{S3}X - V_{S1}Y}{V_{S3}X - V_{S1}Y} \right]$$

$$= \frac{XI_1}{V_{S3}} \tag{5.36}$$

The ratio of the total transfer times in the ETAC and non-ETAC cases is found, by dividing equation (5.36) by equation (5.28), to be

$$\frac{t_2 + t_3}{t_1} = \frac{V_{S1}}{V_{S3}} \left[\frac{X}{L_1(1 - k_{12}^2)} \right] \tag{5.37}$$

The definition of X, as given for equation (5.13), is

$$X = \sqrt{L_1L_3}(k_{13} - k_{12}k_{23}) + L_1(1 - k_{12}^2)$$

and this enables equation (5.37) to be rewritten as

$$\frac{t_2 + t_3}{t_1} = \frac{V_{S1}}{V_{S3}} \left[\sqrt{\frac{L_3}{L_1}} \left[\frac{k_{13} - k_{12}k_{23}}{1 - k_{12}^2} \right] + 1 \right] \tag{5.38}$$

Finally, reference to equation (5.5) shows that

$$\frac{t_2 + t_3}{t_1} = \frac{V_{o/c}}{V_{S3}} \tag{5.39}$$

where $V_{o/c}$ is the open-circuit voltage across S3 in the non-ETAC case and V_{S3} is the voltage at which a clamp device across S3 breaks down, thereby allowing ETAC to occur.

If ETAC is able to occur, $V_{o/c}$ must by definition be greater than V_{S3}. The transfer time when ETAC occurs will therefore always be greater than when it does not.

It will be noted that equation (5.39) is only meaningful if the clamp voltage of the clamp device across S_3 is less than $V_{o/c}$.

5.5.2 Principle

5.5.2.1 ETAC

The equation for the voltage induced in an inductive circuit, if neither the inductance nor the current is constant, is

$$V = \frac{d(LI)}{dt}$$

from which the time for a linear change in the flux linkage LI is

$$\delta t = \frac{\delta(LI)}{V} \tag{5.40}$$

Equation (5.40) shows that the time required to bring about a given change in flux linkage is inversely proportional to the voltage causing the change. Such a relationship is expressed by equation (5.39) because the change in flux linkage is identical whether ETAC occurs or not. In both cases, the change in inductance is the same because the same coil section is switched out, and in both cases the current multiplication is the same, as was demonstrated in section 5.3.

Equation (5.36) shows that when ETAC occurs, the energy transfer time is controlled by the clamp voltage of S3. Hence it is this voltage which is to be regarded as "causing the change", even though the transfer is initiated by opening S1.

Figure 5.4 shows that the clamp device across S3 (which is in series with D3) has current flowing through it for the whole of the transfer, which is the distinguishing characteristic marking out the "controlling" voltage. Thus when there is no TAC, it is the voltage across the opening switch itself which controls the transfer time because it is the clamp device across this switch which carries current for the whole transfer. (See the previous discussion in Chapter 4.)

5.5.2.2 ITAC

It has been shown that with regard to current multiplication, the same considerations apply to ITAC as to ETAC. This is also true for the effect of ITAC on transfer time, since the change in flux linkage is the same whether ITAC occurs or not. For a given opening switch voltage, therefore, the transfer time is again controlled by (i.e. is inversely proportional to) the voltage across the device which carries current for the whole transfer.

For the example circuit of figure 5.12, DZ4 is the clamp device which carries current for the whole transfer when ITAC occurs (see figure 5.14(a)). Thus, if other parameters remain unchanged, the transfer time is inversely proportional to the clamp voltage of DZ4 (see below).

If positive ITAC were to occur (see figure 5.16(c)), the same reasoning would again apply. For example, in the circuit of figure 5.12, the transfer time would be controlled by the breakdown voltage of DZ3.

5.5.3 Simulation Results

Simulations based on the circuits of figures 5.3 and 5.12 were used to obtain transfer time figures for three cases of ETAC and three cases of ITAC respectively. The corresponding transfer time in each circuit with no TAC was also obtained. Figures 5.17 and 5.18 show the theoretical values of the ratio $(t_2 + t_3)/t_1$ for the transfers considered in both circuits. The curves are derived by assuming that

$$\frac{t_2 + t_3}{t_1} = \frac{V_{o/c}}{V_c} . \qquad (5.41)$$

where V_c is the clamp voltage of the device causing TAC, and $V_{o/c}$ is the corresponding unclamped voltage. Equation (5.41) is a generalised form of equation (5.39), and applies to both ETAC and ITAC.

The transfer times given by the simulations are marked on figures 5.17 and 5.18, and it can be seen that they lie on the theoretical curve. This shows that equation (5.41) is correct.

5.5.4 Conclusion

The occurrence of TAC in any mode slows down the energy transfer process, with the new transfer time being inversely proportional to the clamp voltage of the device causing the TAC. This confirms Giorgi's observation [27] that when ETAC occurred in his circuit, he obtained the same current multiplication but an increased transfer time.

In practical circuits an increased transfer time may in fact reduce the efficiency because of the resistance effect (see Chapter 4). The statement made by Long et al [43] to the effect that ETAC could reduce efficiency is thus also valid.

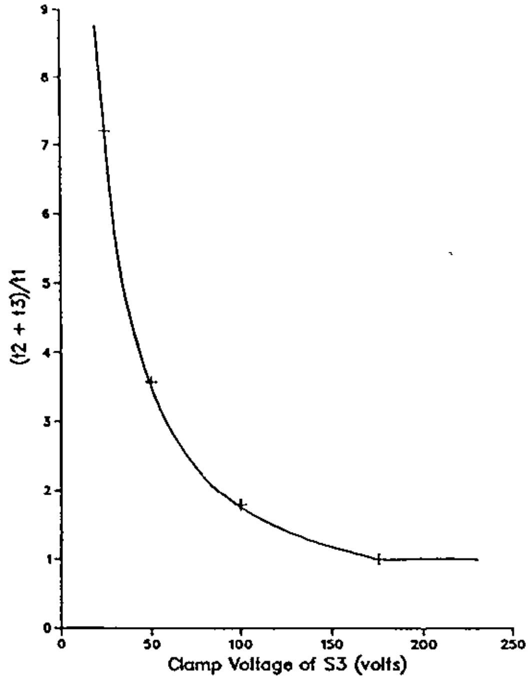


Figure 5.17 Variation of Transfer Time Ratio - Simulation of ETAC

(Symbols indicate simulation results)

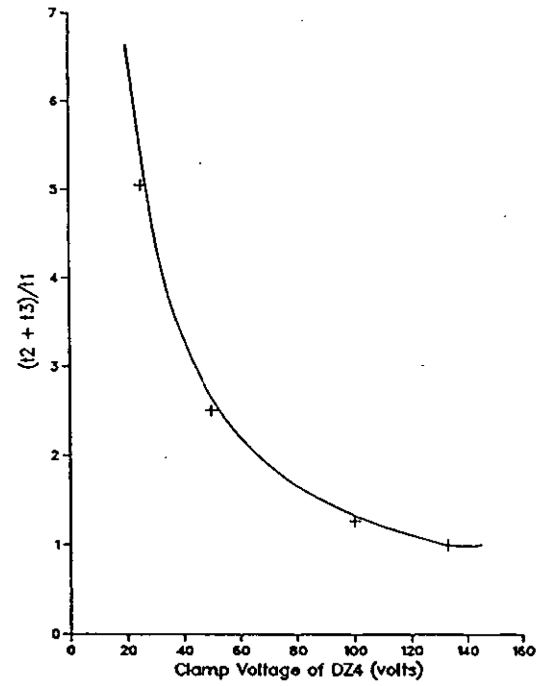


Figure 5.18 Variation of Transfer Time Ratio - Simulation of ITAC

(Symbols indicate simulation results)

5.6 DESIGN IMPLICATIONS OF TRANSFORMER ACTION CLAMPING

5.6.1 Introduction

It has been shown that for a given opening switch voltage, the energy transfer process is completed most rapidly if TAC does not occur. Although slowing down the transfer can increase the resistive losses, as described above, of equal importance is the fact that even if the same energy is delivered to the load, the increased time of delivery reduces the output power.

It could be said that there is an advantage to TAC, in that the energy dissipated during a transfer is shared between two or more clamp devices, thereby reducing the demand on each individual device. Although this could be important at high energy levels, it will generally be far less significant than the drawbacks outlined above. Thus the approach normally would be to attempt to minimise the effect of TAC.

5.6.2 ETAC

To avoid ETAC during any given transfer it is first necessary to calculate (or obtain by simulation) all external induced voltages for the opening switch voltage to be used. It must then be ensured that in the practical circuit, all clamp devices are rated sufficiently high to prevent breakdown. This can, however, be difficult to achieve in a practical circuit because of the cumulative effect of the induced voltages.

In the six-step meatgrinder, for example, it was necessary to clamp TR5 to about 10V in order to prevent TR2 causing ETAC during the fifth transfer (see above). Recalling that TR2 clamps at about 160V, this means that if the voltage across TR5 was allowed to rise to 100V (in order to obtain reasonable transfer speed), then TR2 would have to withstand about 1600V without breaking down if ETAC was to be prevented. The problem then would be to find a switch with the combination of high breakdown voltage, low on-state resistance, and sufficient current-carrying and current-breaking capacity. Long et al [43] comment that the voltage ratios involved tend to lead to "impractical results", i.e. the open-circuit induced voltages are so high that it is not possible to find suitable switches.

5.6.2.1 Compound Switch

As a posssible solution to obtaining the desired switch characteristics, Lindner [30] proposed the compound switch shown in figure 5.19. The switch element S1 has the required current capacity and on-state resistance but only a relatively low breakdown voltage. Both elements are closed whilst that particular branch is conducting, and S1 is opened to effect the energy transfer in the normal way. Once the current has fallen to zero, S2 is opened. Element S2 cannot break current but has a sufficiently high breakdown voltage to prevent ETAC.

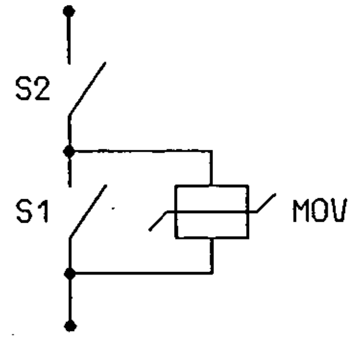


Figure 5.19 Compound Switch to Prevent ETAC

5.6.2.2 Series Switches

Long et al [43] have demonstrated an alternative approach which reduces the external induced voltage problem. This involves placing the opening switches in series with the coil sections, as illustrated in figure 5.20.

Initially all the series switches S1, S3 etc. are closed and all the parallel switches S2, S4 etc. open. Operation then proceeds as follows: close S2 - open S1 - close S4 - open S3...close Sn - open Sn-1. The difference between this and the normal method of operation is that the series switches break the electrical connection between the coil sections. Thus, although each switched-out coil section still experiences an induced voltage, there is no cumulative effect and the breakdown voltage requirement is thereby reduced.

The disadvantage of this method of switching is that since the on-state resistances of the series switches add up, they must be very low. In addition, twice as many switches are required as in the normal circuit configuration. (Long et al [43] present a circuit diagram in which the closing switches S2, S4 etc. are replaced by diodes. They do not, however, explain how multiple current paths are avoided in such a circuit. In figure 5.20, for example, if S2, S4 etc. are replaced by diodes, then when S1 is opened, current will flow simultaneously in several loops, rather than flowing only in the loop containing S2.)

If ETAC cannot be avoided, it is clear that the principle to be followed is simply to use switches with as high a voltage rating as is practicable.

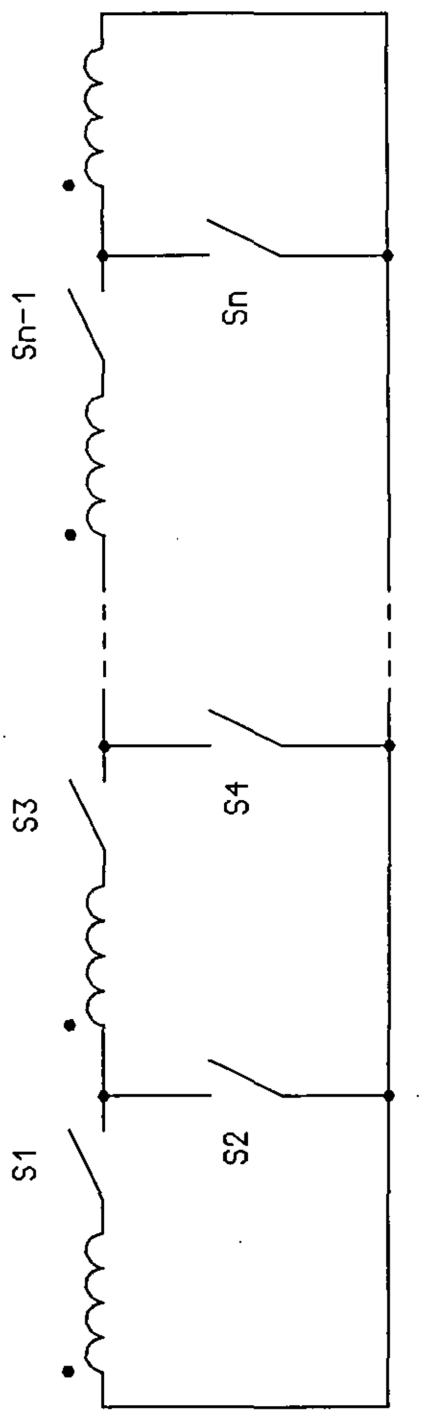


Figure 5.20 Meatgrinder With Series Switches

5.6.3 ITAC

Of the authors referred to in this research, only Legentil and Rioux [13] acknowledge the possibility of ITAC. However, as has been reiterated several times, ITAC is in principle the same phenomenon as ETAC. The same design principle applies: that is, that in order to avoid or minimise the speed reduction caused by ITAC, voltage ratings should be as high as practicable.

5.6.3.1 Component Protection

As described previously in Chapter 4, negative ITAC means that the blocking diodes must be capable of non-destructive breakdown. Even if ITAC is not expected, the unpredictable effects of noise mean that protection is still required.

The ordinary diode cannot simply be replaced because zener-type devices tend to have insufficient forward current rating. It is also often the case that no data is given for turn-on time for forward conduction. Therefore an arrangement such as that shown in figure 5.21 is required.

The protection device could be a zener diode, a transient voltage suppressor or a non-linear resistor such as a MOV (see component descriptions in Chapter 3). The purpose of the series diode is to prevent forward conduction via the protection device.

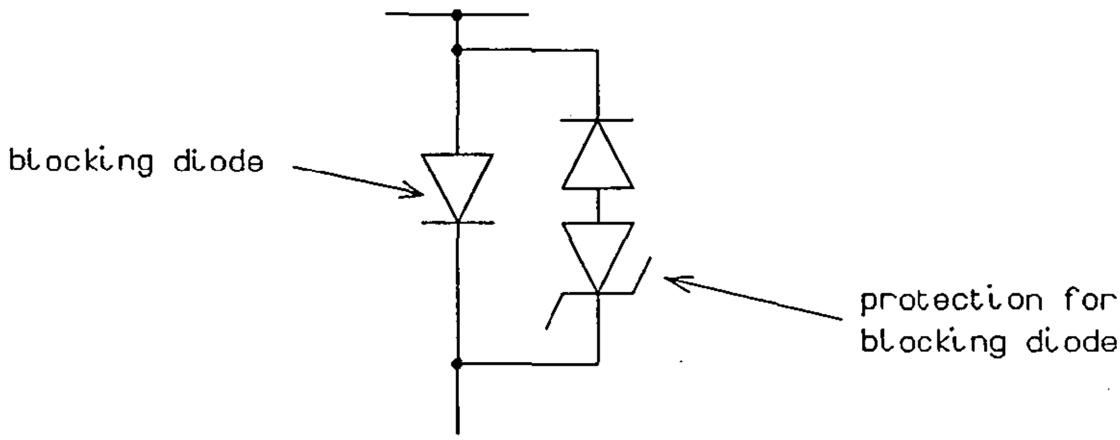


Figure 5.21 Protection for Blocking Diode

Protection should also be provided for the last switch in a multi-step circuit (such as TR7 in the six-step meatgrinder), even though it does not break current. This is to protect the switch from either predicted positive ITAC or unexpected noise spikes.

CHAPTER SIX

OPTIMISATION OF MEATGRINDER DESIGNS

6.1 INTRODUCTION

There will be many possible meatgrinder designs which can meet a given performance specification. This Chapter provides an introduction to the area of optimisation; that is, finding the best design possible rather than simply one which works.

A typical design requirement could be to minimise the number of steps required to produce a given current multiplication and efficiency. As this is a very broad problem, the approach taken was to examine the more specific question of maximising the efficiency for a given number of steps and a fixed current multiplication. Solving this problem should provide data which can be used to meet the more usual type of design requirement.

The single-step meatgrinder is considered first. This implementation is both simple and potentially very useful [26,28,32,33,35]. For the case of a series-connected uncoupled load it is shown that the efficiency of energy transfer may be maximised by the correct choice of meatgrinder inductances. Experimental work which supports this finding is described. It is further shown that the use of a decompression switch [27] to short out the load during charging of the meatgrinder coil does not change these optimum inductance values.

The possibility of optimisation is further demonstrated for the case of a two-step circuit.

6.2 ANALYSIS OF UNLOADED SINGLE-STEP MEATGRINDER

Figure 6.1 shows an unloaded single-step meatgrinder, the operation of which was introduced in Chapter 2. From Chapter 2, the current multiplication β is

$$\beta = \frac{I_2}{I_1} = \frac{L_2+M}{L_2} \tag{6.1}$$

and the step efficiency is

$$\eta_s = \frac{1+k^2\alpha+2k\alpha}{1+\ \alpha\ +2k\alpha} = \frac{(1+k\sqrt{\alpha})^2}{1+\ \alpha+\ 2k\alpha} \tag{6.2}$$

where $\alpha=L_1/L_2$.

From equation (6.1) the inductance ratio α can be expressed as

$$\alpha = \frac{(\beta-1)^2}{k^2} \tag{6.3}$$

since $M = k\sqrt{L_1L_2}$. Substituting equation (6.3) into equation (6.2) then yields

$$\eta_s = \frac{k^2\beta^2}{\beta^2+(k^2-1)(2\beta-1)} \tag{6.4}$$

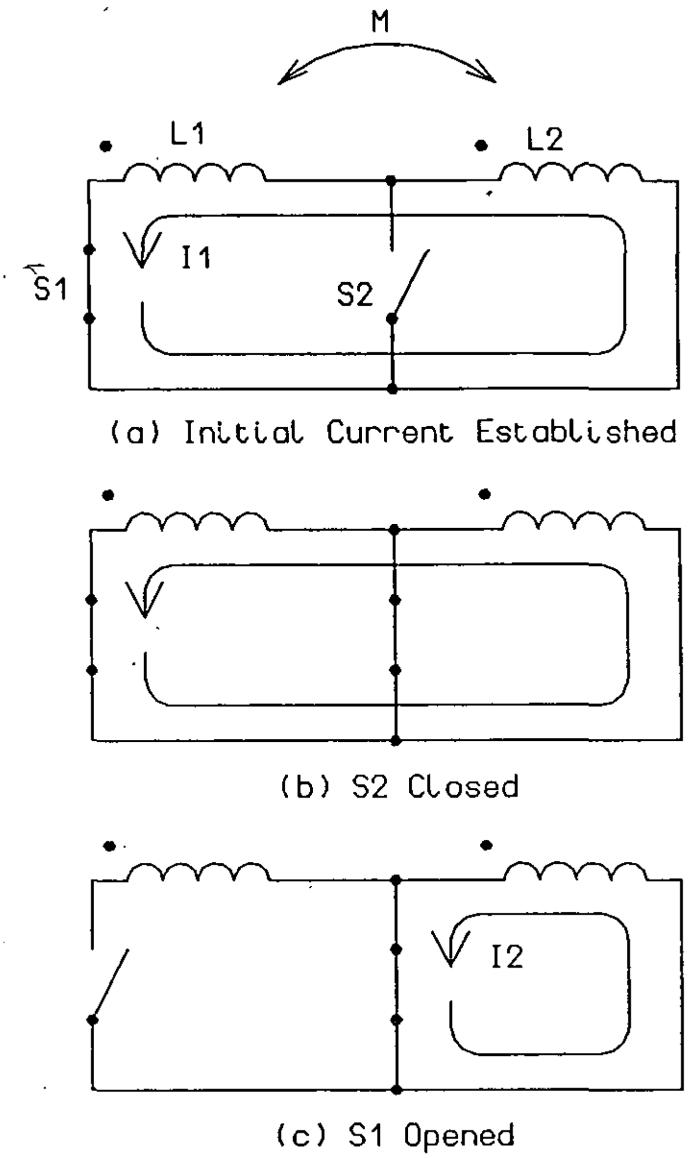


Figure 6.1 Operation of Unloaded Single-Step Meatgrinder

6.3 ANALYSIS OF LOADED SINGLE-STEP MEATGRINDER

The addition of an uncoupled load inductance in series with L_2 (figure 6.2) does not affect the operation of the circuit. However, the coupling coefficient k must be modified to account for the load. As described in Appendix D, this leads to the result

$$k' = k \left[\frac{L_2}{L_2 + L_{LOAD}} \right]^{\frac{1}{2}} \tag{6.5}$$

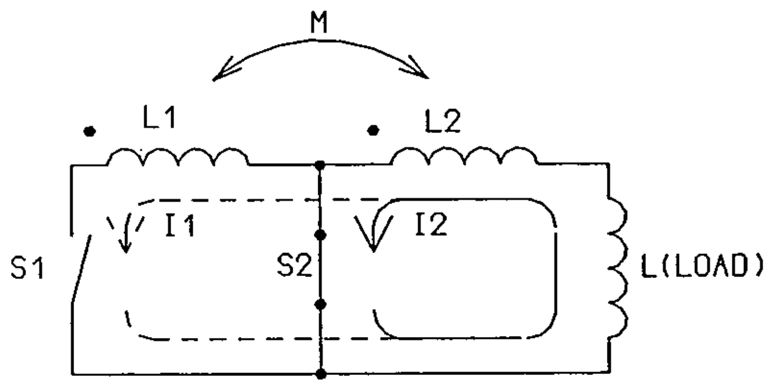


Figure 6.2 Single-Step Meatgrinder With Uncoupled Load

In addition to calculating the step efficiency, account must also be taken of the fact that some of the final circuit energy remains stored in L_2. This leads to a further efficiency penalty η_u, where

$$\eta_u = \frac{L_{LOAD}}{L_2 + L_{LOAD}} \tag{6.6}$$

Now, let the inductance ratios L_1/L_{LOAD} and L_2/L_{LOAD} be referred to as σ_1 and σ_2 respectively. Equation (6.5) shows that as σ_2 increases, the value of k' approaches that of k. This means that for a given value of current multiplication β, the step efficiency η_s improves. The uncoupled load penalty η_u, however, simultaneously becomes smaller (equation (6.6)). The optimisation requirement is to determine the net effect on the total efficiency η_t, where

$$\eta_t = \eta_s \eta_u \tag{6.7}$$

To analyse the effect of σ_2 on η_t, equation (6.7) is expanded. The step efficiency η_s is given by equation (6.4), but with k' in place of k. If η_u is expressed in terms of σ_2, then multiplying the expressions for η_s and η_u yields

$$\eta_t = \frac{k^2\beta^2\sigma_2}{\sigma_2^2[\beta^2-2\beta(1-k^2)+(1-k^2)] + \sigma_2[2\beta^2-2\beta(1-k^2)-2\beta+(1-k^2)+1] + [\beta^2-2\beta+1]} \tag{6.8}$$

Differentiating this result with respect to σ_2 yields

$$\frac{d\eta_t}{d\sigma_2} = \frac{k^2\beta^2 A - k^2\beta^2\sigma_2(dA/d\sigma_2)}{A^2} \tag{6.9}$$

where

$$A = [\beta^2(\sigma_2+1) + (k^2\sigma_2-\sigma_2-1)(2\beta-1)][1+\sigma_2]$$

The stationary points of equation (6.8) are located by equating its derivative to zero. This leads to a non-imaginary value of the inductance ratio σ_2 of

$$\sigma_{2(stat)} = \frac{\beta - 1}{[\beta^2 - 2\beta(1-k^2) + (1-k^2)]^{\frac{1}{2}}} \qquad (6.10)$$

The nature of this stationary point may be determined from the following observations:

(a) Differentiation of equation (6.8) shows that the function for σ_2 has only one stationary point in the region of interest between zero and infinity. This means that the stationary point cannot be a local minimum or maximum and must either be an absolute maximum, an absolute minimum or a point of inflexion.

(b) When $\sigma_2=0$, $\eta_t=0$ (except for the special case $\beta=1$, when η_t is indeterminate, but this case is not of interest because it has no physical meaning).

(c) As σ_2 tends to infinity, η_t tends to zero.

(d) It follows from (b) and (c) that the function has a lower value at the extremes of the region of interest than it does at the stationary point. Hence the stationary point is an absolute maximum.

Equation (6.10) thus provides the optimum value of σ_2, for which the energy transfer efficiency η_t is maximised.

An example of the variation of η_t with σ_2 for given values of current multiplication and coupling coefficient is shown in figure 6.3. As σ_2 increases, the increase in η_s is initially much more significant than the decrease in η_u and the overall efficiency η_t rises rapidly. The function is such that as the peak efficiency is approached, the curve begins to level off

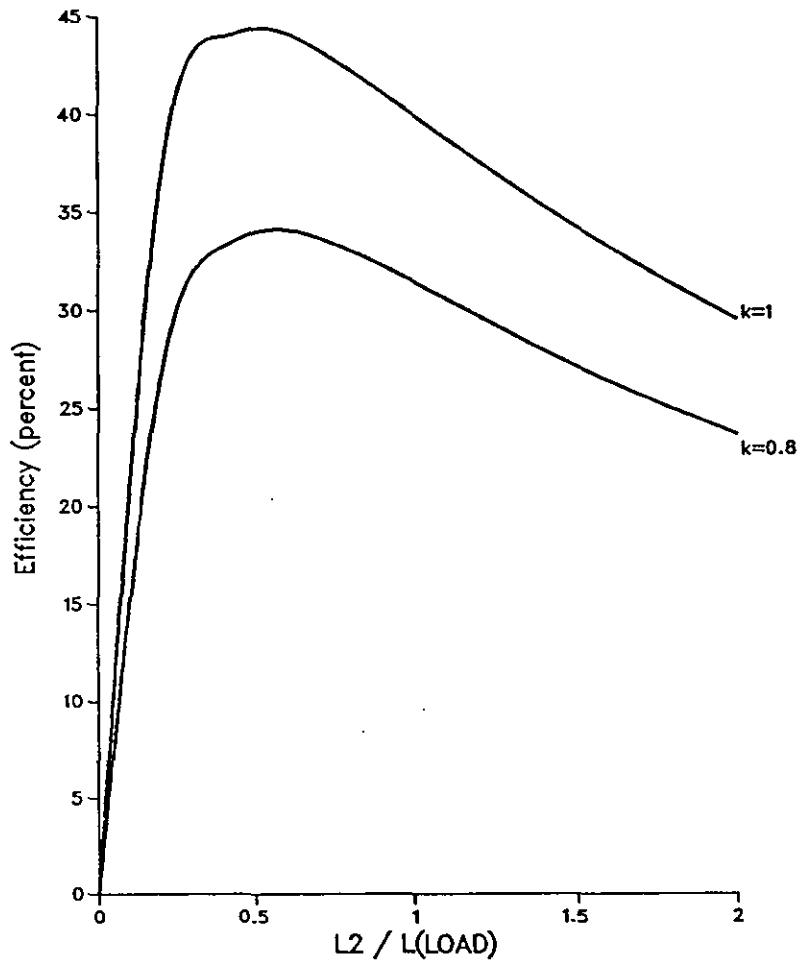


Figure 6.3 Example Showing Variation of Efficiency With L_2/L_{LOAD} (Multiplication = 2, No Decompression)

before subsequently rising to a slightly higher peak. The peak is not sharply defined, σ_2 values either side of the true optimum only affecting the efficiency by one or two percent. As σ_2 increases further, η_t rolls off slowly as the decrease in η_u becomes more significant.

The analysis is completed by determining σ_1 in terms of σ_2, so that both meatgrinder self inductances are known in terms of the load inductance. This is achieved by modifying equation (6.3) to account for the load inductance so that it becomes

$$\frac{L_1}{L_2+L_{LOAD}} = \frac{(\beta-1)^2}{k'^2} \tag{6.11}$$

Substituting equation (6.5) into equation (6.11) gives

$$\frac{L_1}{L_{LOAD}} = \sigma_1 = \frac{(\beta-1)^2}{k^2}.\frac{(1+\sigma_2)^2}{\sigma_2} \tag{6.12}$$

Figures 6.4(a) to 6.4(c) are sample curves derived from the above equations. They indicate at a glance the capability of an ideal single-step meatgrinder circuit.

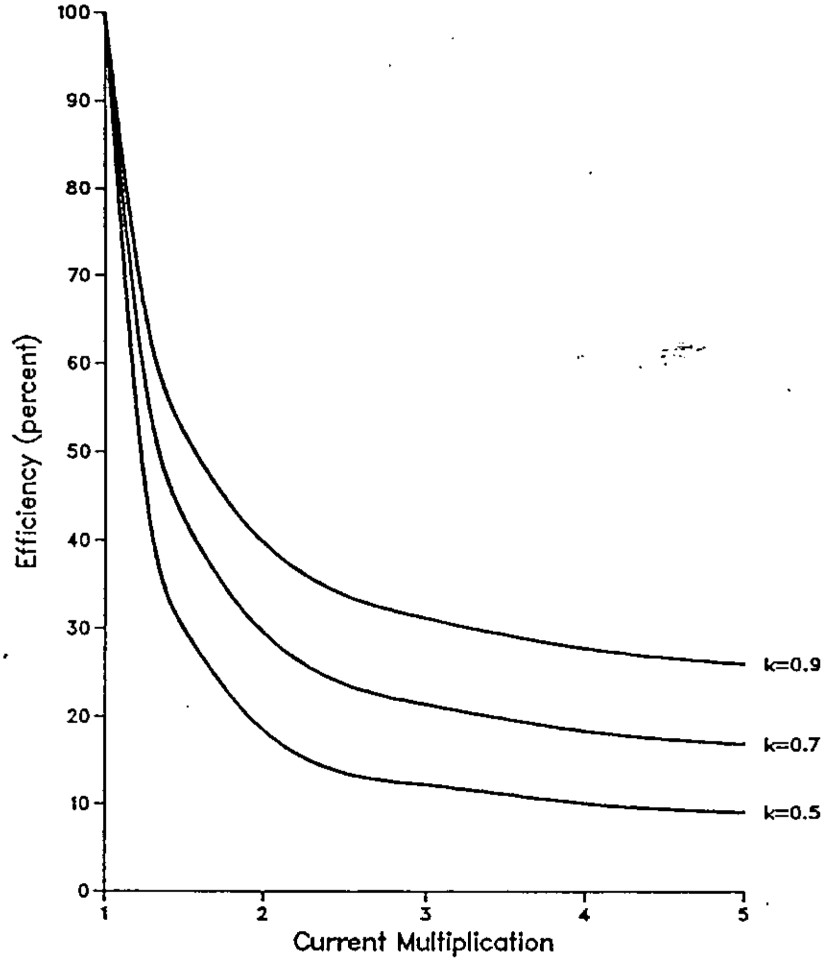


Figure 6.4(a) Single Step Transfer Efficiency at Optimal Conditions (Uncoupled Load, No Decompression)

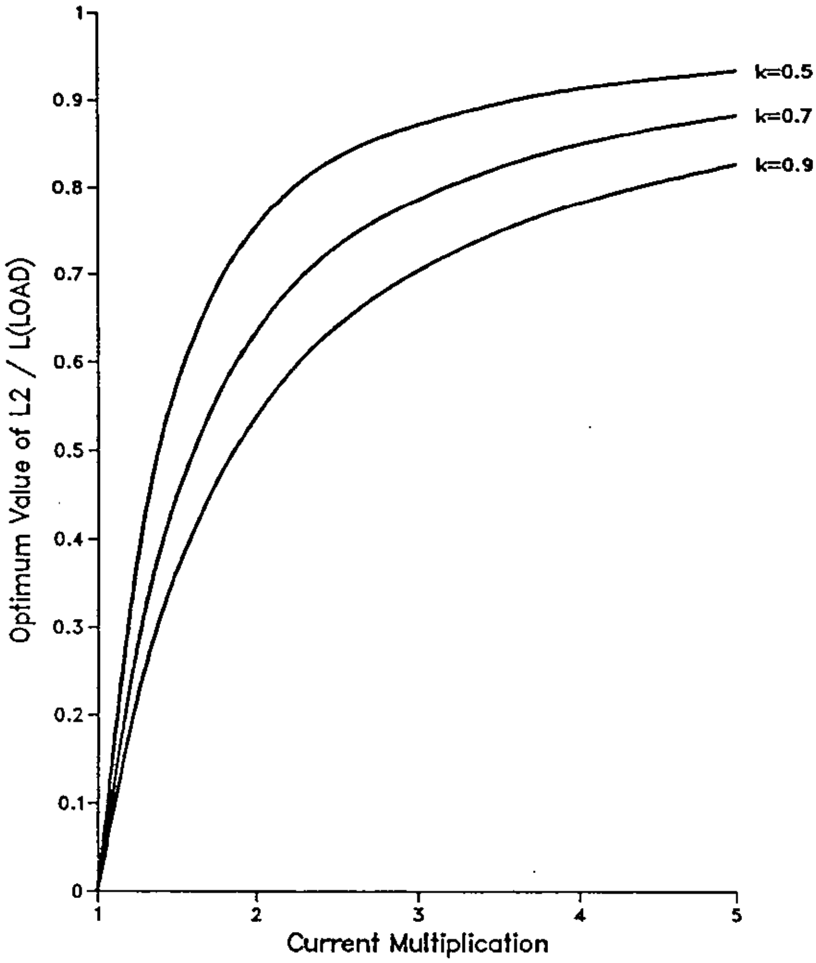


Figure 6.4(b) Optimum Values of L_2/L_{LOAD} for Single Step, No Decompression

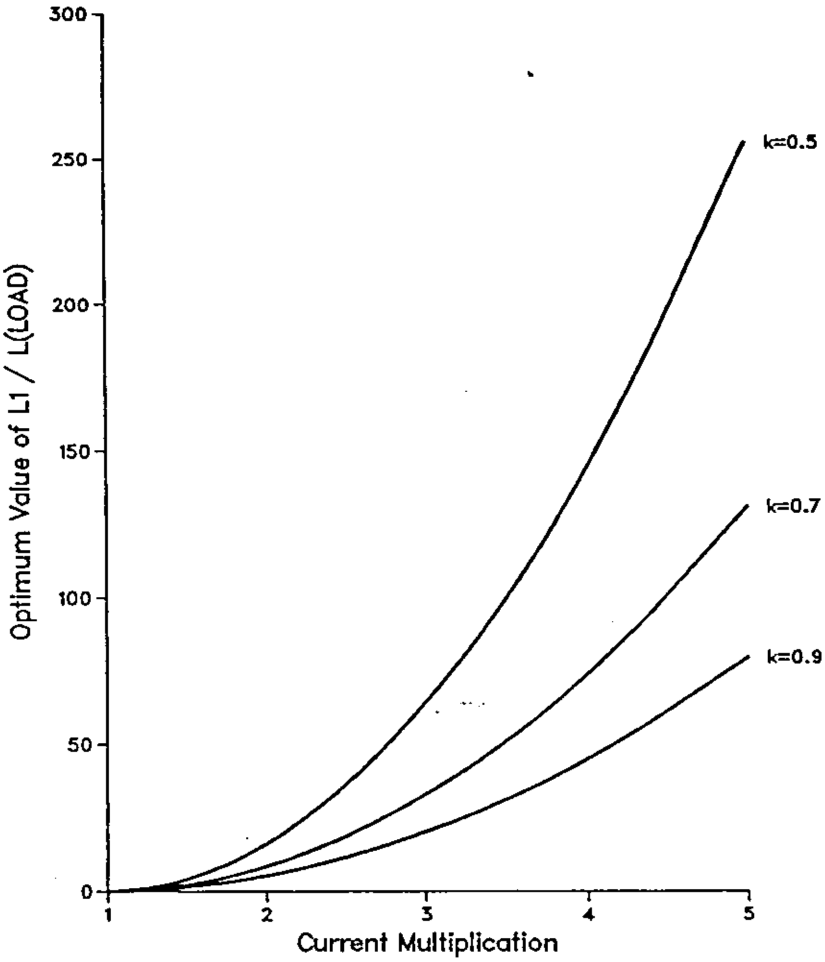
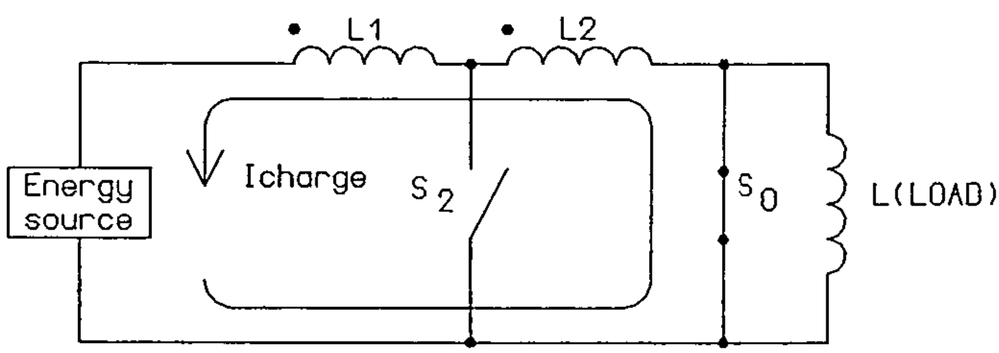


Figure 6.4(c) Optimum Values of L_1/L_{LOAD} for Single Step, No Decompression

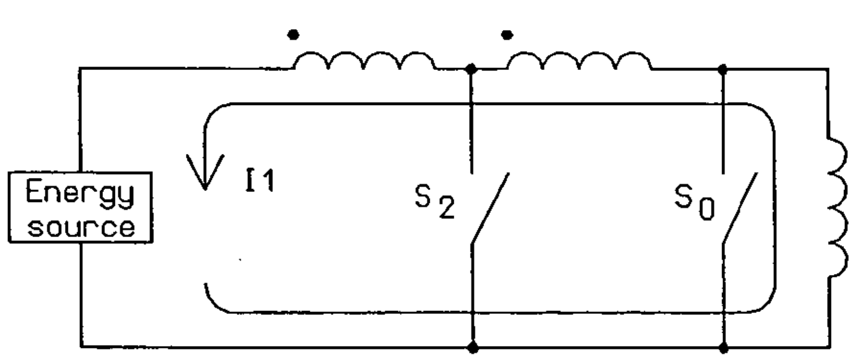
6.4 ANALYSIS OF SINGLE-STEP MEATGRINDER WITH DECOMPRESSION

6.4.1 Introduction

In some cases it is undesirable for load current to flow whilst the meatgrinder is storing energy from the source, since this process may take a relatively long time. In such cases a switch (referred to as a decompression switch) is used to short out the load during charging. Once the desired current has been reached, the switch is opened in order to bring the load into circuit (see figure 6.5); operation then proceeds as before.



(a) Charging



(b) Load Brought into Circuit by Opening Decompression Switch

Figure 6.5 Use of Decompression Switch

"Decompression" means that the flux is initially generated by the current in a single inductor; a second inductor is then brought into circuit and generation of the flux is divided between the two. The process leads to both a reduction in current and a loss of energy in the switch.

Defining the decompression efficiency η_d as the ratio of the total circuit energy after decompression to the initial circuit energy, and the decompression current ratio β_d as the ratio of the corresponding currents, it follows that

$$\beta_d = \eta_d = \frac{L_T}{L_T + L_{LOAD}} \qquad (6.13)$$

where L_T is the total meatgrinder inductance.

6.4.2 Mathematical Analysis

The decompression current ratio β_d simply serves to indicate the initial charge current necessary to give the required initial load current. The meatgrinder action again multiplies the current in the load by a factor β.

The overall efficiency with decompression η_{td} is given by

$$\eta_{td} = \eta_s \eta_u \eta_d = \eta_t \eta_d \qquad (6.14)$$

where the other symbols have their previous meaning.

The total efficiency η_t may be derived as described previously. By expanding L_T and using equation (6.13) to substitute for L_1, the decompression efficiency η_d may be expressed in terms of β

and σ_2. The total efficiency with decompression η_{td} may subsequently be expressed as

$$\eta_{td} = \frac{k^2\beta^2\sigma_2}{A} \cdot \frac{(A-k^2\sigma_2)}{A} \qquad (6.15)$$

Again, it is the stationary points of this function which are of interest. Differentiating equation (6.15) with respect to σ_2 and equating the derivative to zero leads to the condition:

$$[A-2k^2\sigma_{2d}] \cdot [Ak^2\beta^2-k^2\beta^2\sigma_{2d}(dA/d\sigma_2)] = 0 \qquad (6.16)$$

where σ_{2d} is the stationary point value of σ_2 with decompression.

The values of σ_{2d} are found by equating each of the two brackets in turn to zero. It can be seen that the second bracket then yields exactly the same result as the non-decompression case (equation (6.10)). Substituting for A in the first bracket and equating to zero leads to a quadratic in σ_2 the roots of which are imaginary and therefore not of interest.

There is therefore again only one stationary point of interest, and as before it is a maximum. This shows that the optimum value of σ_2 is the same with or without decompression.

Figure 6.6 refers to the same example as figure 6.3 and shows how decompression degrades the efficiency without shifting the point at which the maximum occurs.

Figure 6.7 is derived from figure 6.4(a) by multiplying each efficiency value by η_d. It can be seen that there is a peak in the curve for the case $k_{12}=0.9$. This is a consequence of the fact that as the current multiplication rises, the meatgrinder

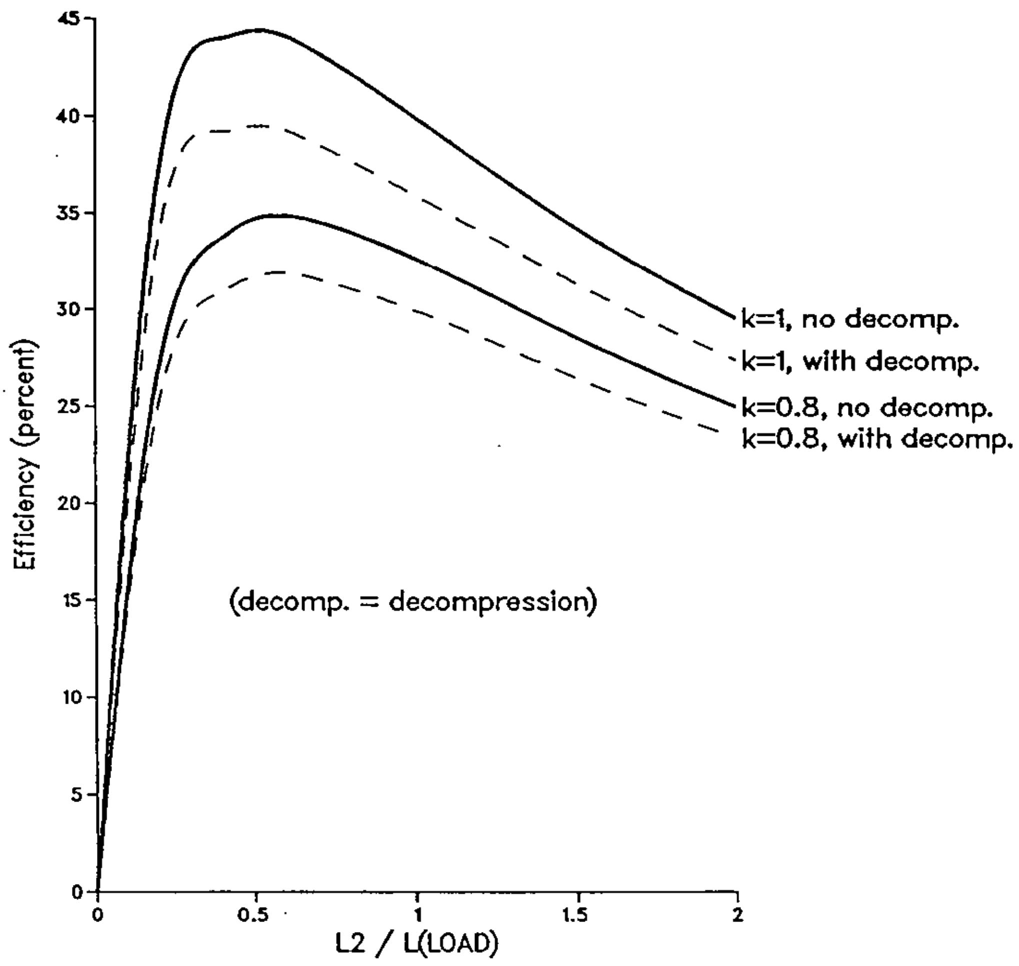


Figure 6.6 Example Showing Optimum Efficiency of a Single Step (Current Multiplcation = 2)

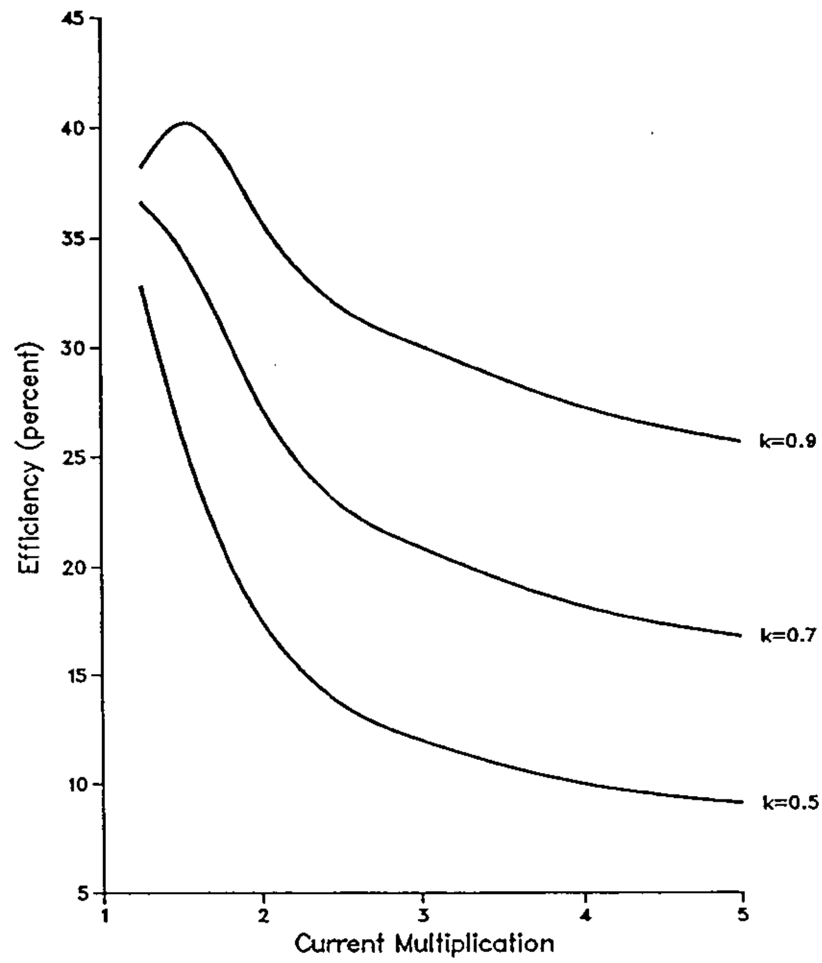


Figure 6.7 Single Step Transfer Efficiency at Optimal Conditions (Uncoupled Load, With Decompression)

efficiency η_t falls (see figure 6.4(a)), whereas the decompression efficiency η_d rises because the meatgrinder inductance becomes larger relative to the load inductance. The shape of the optimal efficiency curve therefore depends on the net effect of these two trends.

6.5 EXPERIMENTAL RESULTS FOR A SINGLE-STEP MEATGRINDER

6.5.1 Objective

The theory given above shows that, for a given coupling coefficient between the meatgrinder coil sections, the same current multiplication can be achieved with many different sets of meatgrinder inductances. Although the current multiplication is the same, however, the efficiency is different in each case. An optimum set of inductances exists which maximises the efficiency.

The purpose of the experiments discussed below is to provide a demonstration of this principle in operation.

6.5.1.1 Specification

The case arbitrarily chosen for the demonstration has the following parameters:

Current multiplication = 3
Coupling coefficient = 0.9
Load inductance = 100μH

Figure 6.8 shows the theoretical efficiency curve for this case, as derived from equation (6.8). The three representative values of σ_2 (i.e. L_2/L_{LOAD}) chosen for the experiments are indicated. Also shown are the three experimental results (see discussion below).

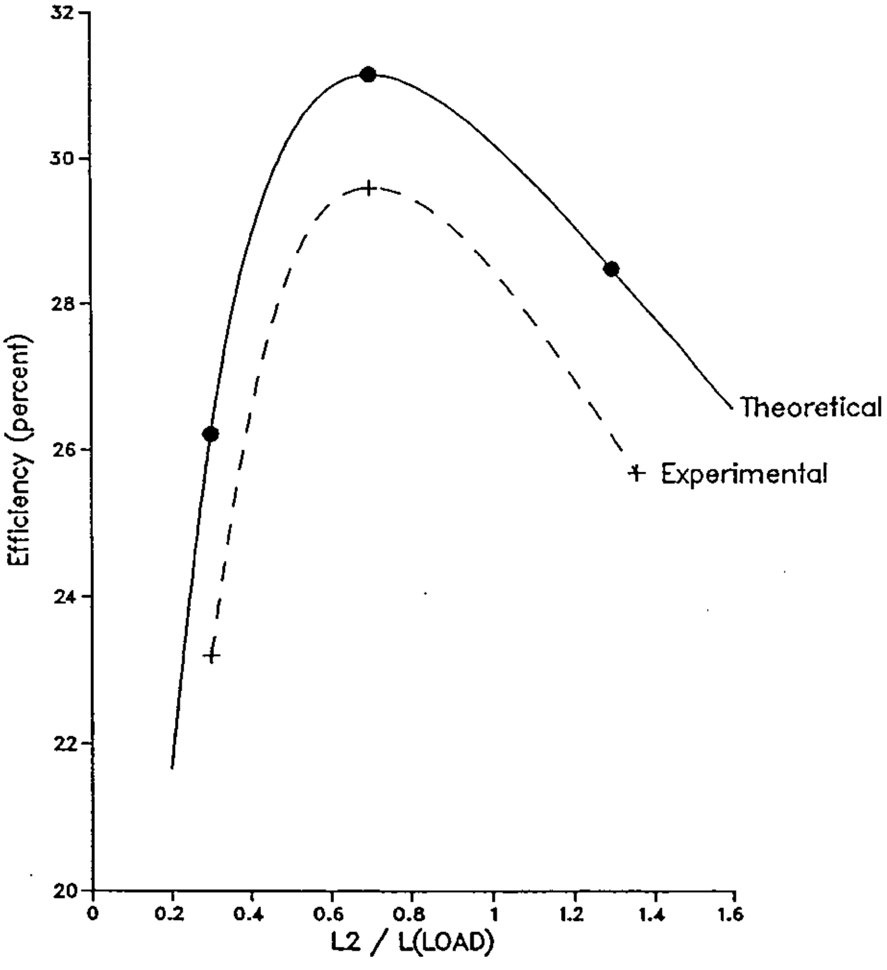


Figure 6.8 Values for Multiplication = 3, k = 0.9, No Decompression

An initial current of 10A was selected so that switching could again be carried out with MOSFETs. (The advantages of MOSFETs were described in Chapter 3.)

6.5.2 Coil Design

6.5.2.1 Choice of Geometry

It was decided that the meatgrinder should consist of two concentric single-layer solenoids. Concentric solenoids have been successfully used in other meatgrinder work [27,28], and the single-layer type are relatively simple to design and construct. High magnetic coupling can be obtained by making the coils large, so that the difference in cross-sectional areas is small. Even if the two inductances are very different, a high coupling coefficient can still be obtained by using different wire or strip sizes to produce coils of roughly equal width.

6.5.2.2 Design Procedure

The inductance formula used is that given by Welsby [53]. It approximates the coil to a cylindrical current sheet, and is appropriate where the turns are close together and the radial thickness of the winding is small compared to the coil radius.

The formula is

$$L = \frac{a^2N^2}{b} \cdot \frac{1}{1 + 0.9(a/b) - 0.02(a/b)^2} \cdot 4\pi^2 \times 10^{-3} \ \mu H \qquad (6.17)$$

where

a = coil radius in cm

b = coil width in cm

N = number of turns

In order to minimise constructional difficulties, the radius of the outer coil (L_1) was restricted to 25cm. A convenient nominal coil width of 10cm was chosen. Equation (6.17) was then applied on a trial and error basis in order to find the number of turns needed to give the required inductance. The wire diameter or strip width required was given by the coil width divided by the number of turns.

In each of the three cases, L_1 and L_2 were designed to have the same width.

The load inductor was designed in a similar manner.

6.5.2.3 Mechanical Construction

Coil formers were constructed from wood and "Darvic" insulating material. Winding was carried out manually by suspending the former on a lathe and securing the winding with polyester adhesive tape at regular intervals.

The meatgrinder coils are shown in figure 6.9. The outer coil L_1 is on the right and is wound with enamelled copper wire, whilst the inner coil L_2, on the left, is wound with copper strip insulated with polyester film. In this particular case, L_2 was not wound as a single layer, the required inductance being obtained by trial and error. This was due to the required srip width being unavailable.

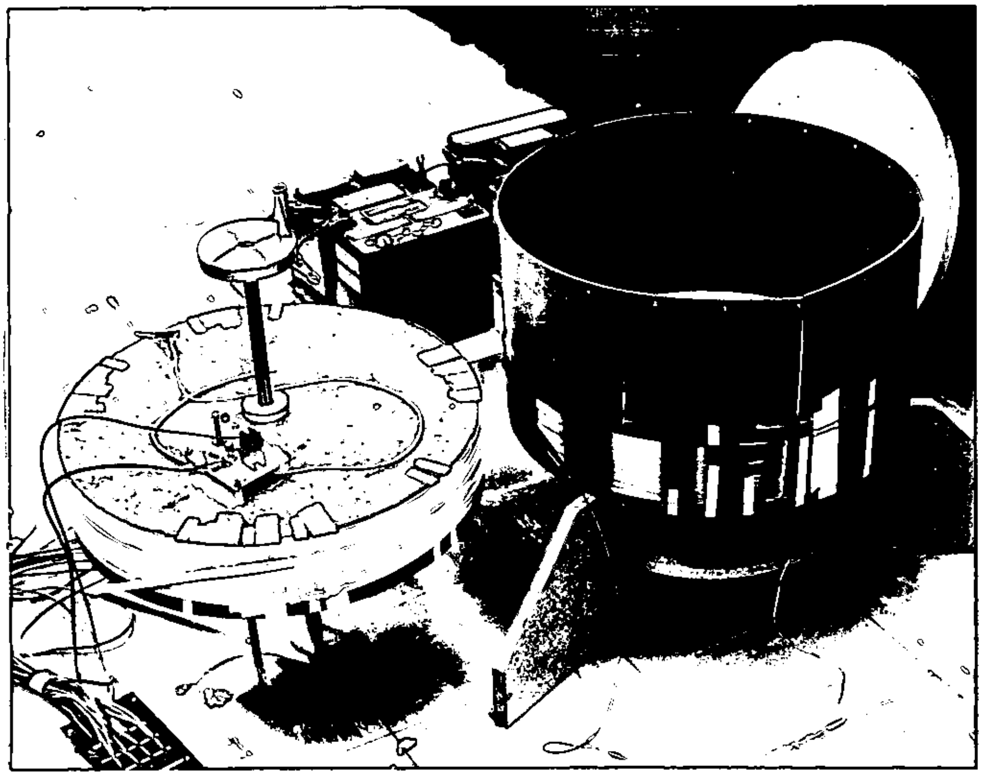


Figure 6.9 Coils for Single-Step Meatgrinder

With L_2 fitting snugly inside L_1, adjustment of the coupling coefficient is provided by a screw arrangement constructed so as to enable L_2 to be moved along the common axis of the two coils. To make the arrangement functional, the threaded rod and handle assembly is first removed from L_2. L_2 is then placed inside L_1, with the legs underneath L_2 sliding through holes in the base of L_1. The lid shown just behind L_1 is then screwed in place and the threaded rod re-inserted. The rod screws into a threaded plate on the base of L_2, and the handle on the end of the rod bears down on the top lid. Thus when the handle is turned, L_2 is forced to move relative to L_1.

Figure 6.10 shows the assembled meatgrinder connected to the load coil.

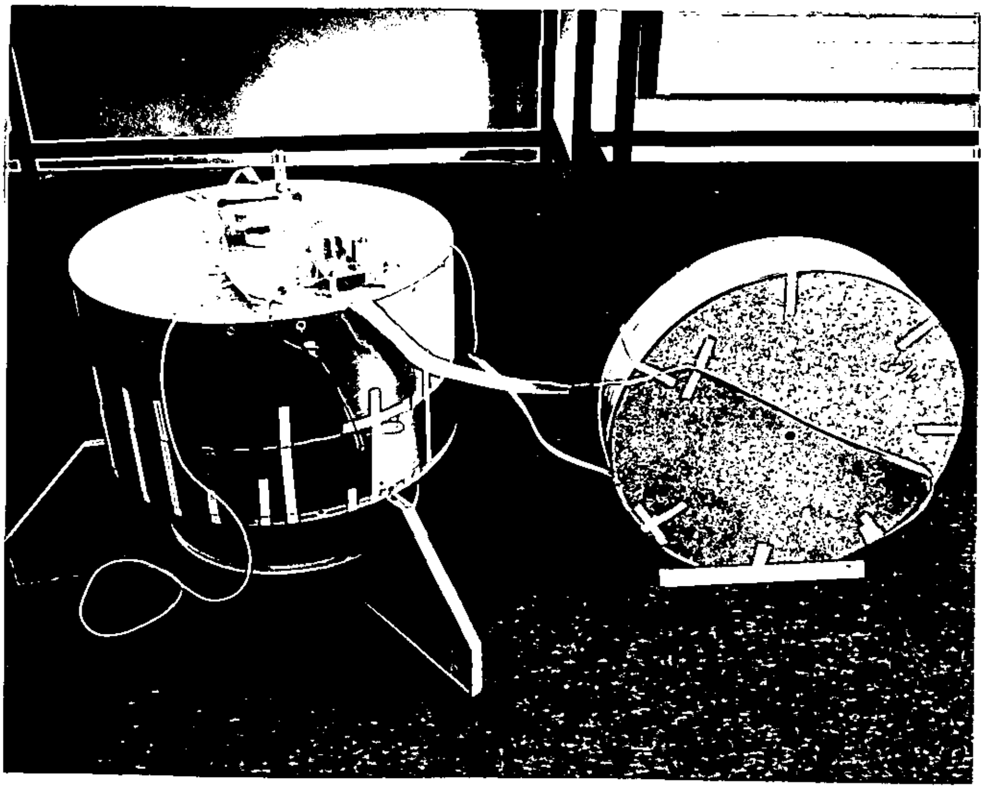


Figure 6.10 Single-Step Meatgrinder With Load Coil

6.5.2.4 Measurements

The coil inductances were measured at 10kHz (the highest frequency available on the instrument used), care being taken to keep the coils well away from stray metalwork. The physical dimensions produced were not exactly as designed, but the results were nevertheless satisfactory (see tables 6.1 and 6.2).

EXPERIMENT NUMBER	L_1	L_2
1	59 turns 1.6mm dia. wire	6.25 turns 15mm strip
2	48.5 turns 1.6mm dia. wire	12.5 turns 8mm strip (2 layers)
3	52 turns 2mm dia. wire	9 turns 8mm strip

L_1 radius = 25cm
L_2 radius = 24.1cm
Load coil: radius 25cm, wound with 11 turns of 8mm strip

Table 6.1 Winding Details

EXPT. NO.	L_1 (μH)		L_2 (μH)		L_{12} (μH)	k
	DESIGN	MEAS'D	DESIGN	MEAS'D	(MEAS'D)	
1	2782	2740	30	30	3283	0.89
2	2039	2016	70	70	2740	0.87
3	2009	2012	130	136	2950	0.77

(MEAS'D = MEASURED)

Note: L_{12} is the total inductance of L_1 and L_2 in series.

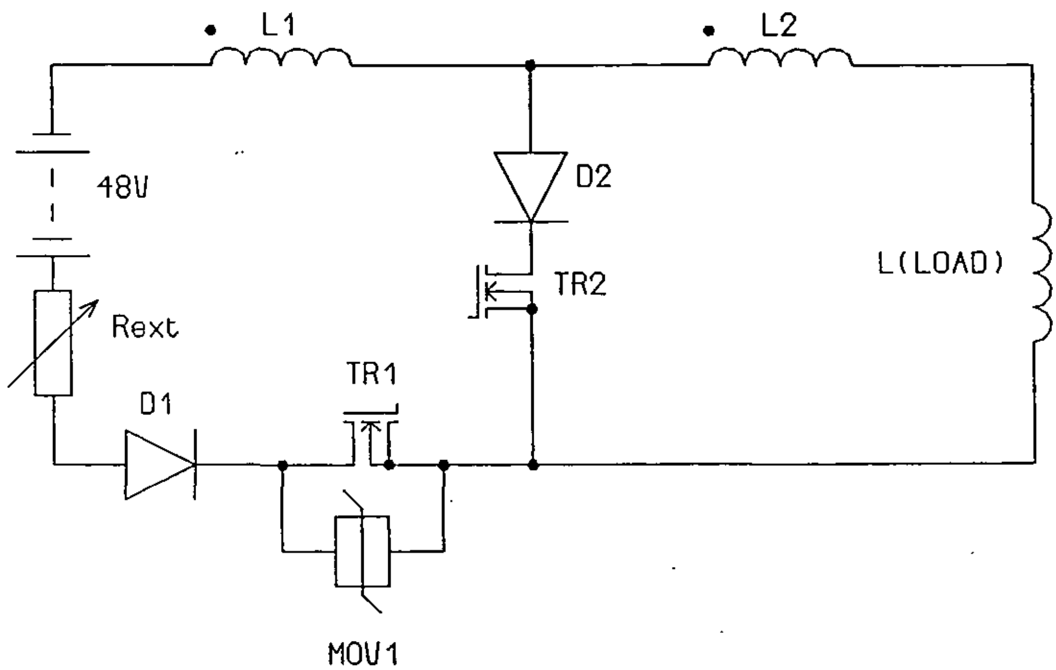
Table 6.2 Inductance Figures

The coupling coefficient for experiment 3 is rather low because an error in positioning the L_1 winding meant that the coils could not overlap sufficiently to reduce the leakage flux to the required level. The experiment was not repeated because the current multiplication was still satisfactory. (It should be remembered that the value of coupling coefficient obtained is highly sensitive to small changes in the inductance figures. In this particular case, for example, an increase of only 4% in the total inductance would raise the coupling coefficient to 0.88.)

6.5.3 Other Circuit Components

Figure 6.11 shows the circuit diagram for the single-step meatgrinder experiments; the electronic components can be seen mounted on the small circuit board in figure 6.9.

Transistor TR1 is a high voltage (1000V) device with a 2Ω on-state resistance. A circuit simulation showed that because



COMPONENTS NOT SHOWN: 18V gate-source zener diodes for TR1 and TR2

13A fuse in series with R_{ext}

TR1: International Rectifier IRFPG50

TR2: International Rectifier IRFP044

D1, D2: Motorola MR756

MOV1: Power Development Z320C

Figure 6.11 Circuit Diagram for Single-Step Meatgrinder

of the circuit resistance, an opening switch voltage of about 800V was necessary to obtain a final current of 30A. The high on-state resistance is unimportant for the purposes of the experiment because it simply increases the energy dissipated during charging. The clamp device MOV1 is necessary to restrict the drain-source voltage during turn-off when the MOSFET is operating above its continuous current rating [45].

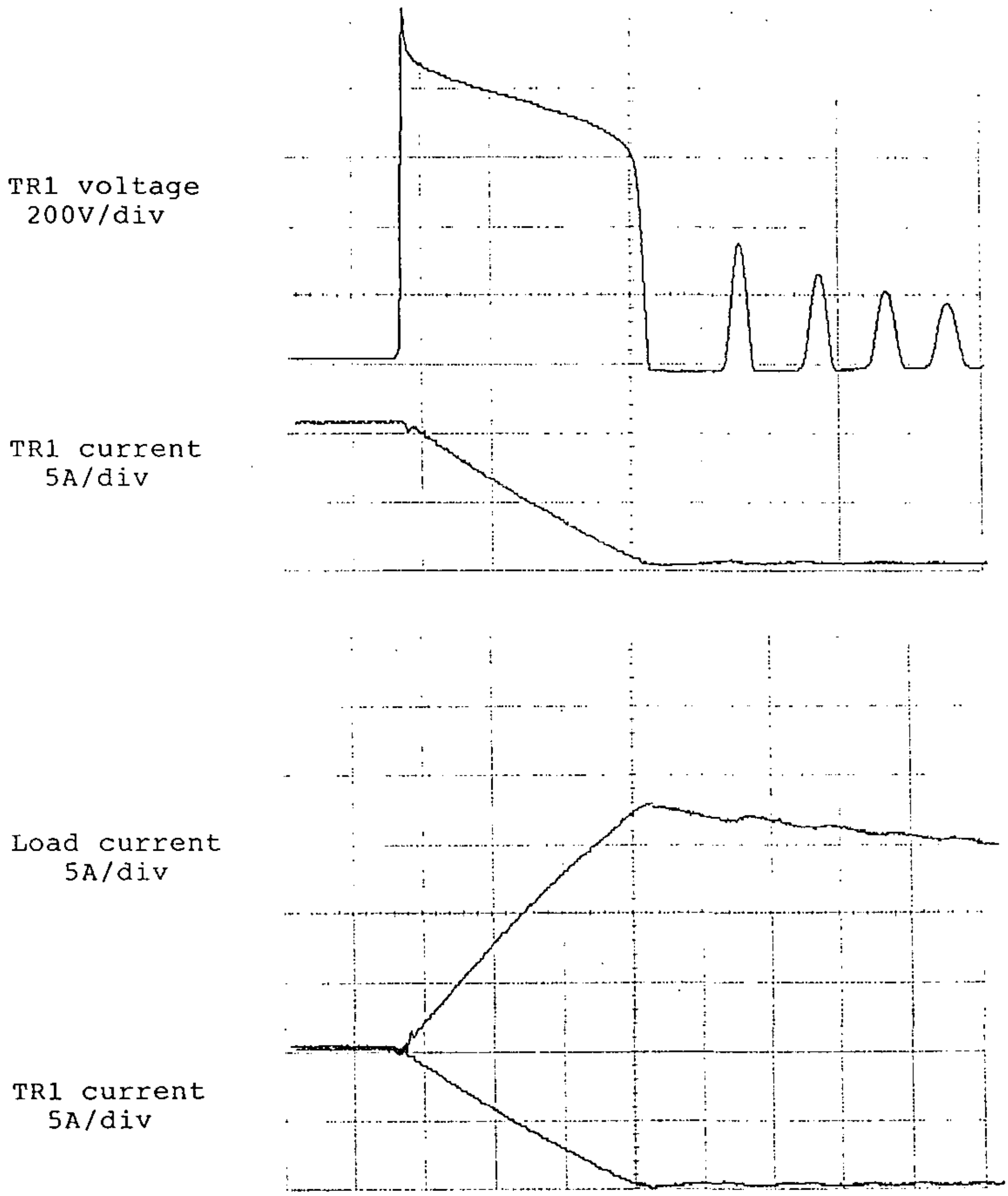
By contrast, transistor TR2 has a low on-state resistance (0.02Ω) so that the loop resistance is kept to a minimum for the energy transfer. Although this device does not break current the possibility of unexpected voltage spikes still has to be considered. No external drain-source voltage limiting is required, however, because TR2 operates within its continuous current rating and is therefore able to self-avalanche if necessary.

The transistors are driven by the same timing and drive circuits as were used for the six-step meatgrinder (see Chapter 3 and Appendix B). The "TR1" and "TR2a" outputs are used to drive TR1 and TR2 respectively; the other outputs are not required. The same charge time (1.7ms) is used and, with a 48V supply, adjustment of R_{ext} allows the current after this time to be set to 10A. The power supply consists of four 12V car batteries.

6.5.4 Results

6.5.4.1 First Experiment

Figure 6.12 shows that the circuit operates as expected: the voltage on TR1 is clamped by MOV1, and the current in L_{LOAD} rises to about 28A - quite close to the predicted value of 30A.



All traces: Time 10μs/div

Figure 6.12 Waveforms for First Experiment

As in the six-step meatgrinder, noise was reduced (and the final current slightly increased) by inserting a 100Ω resistor in series with the gate to slow down the turn-off of TR1 (see figure 6.13).

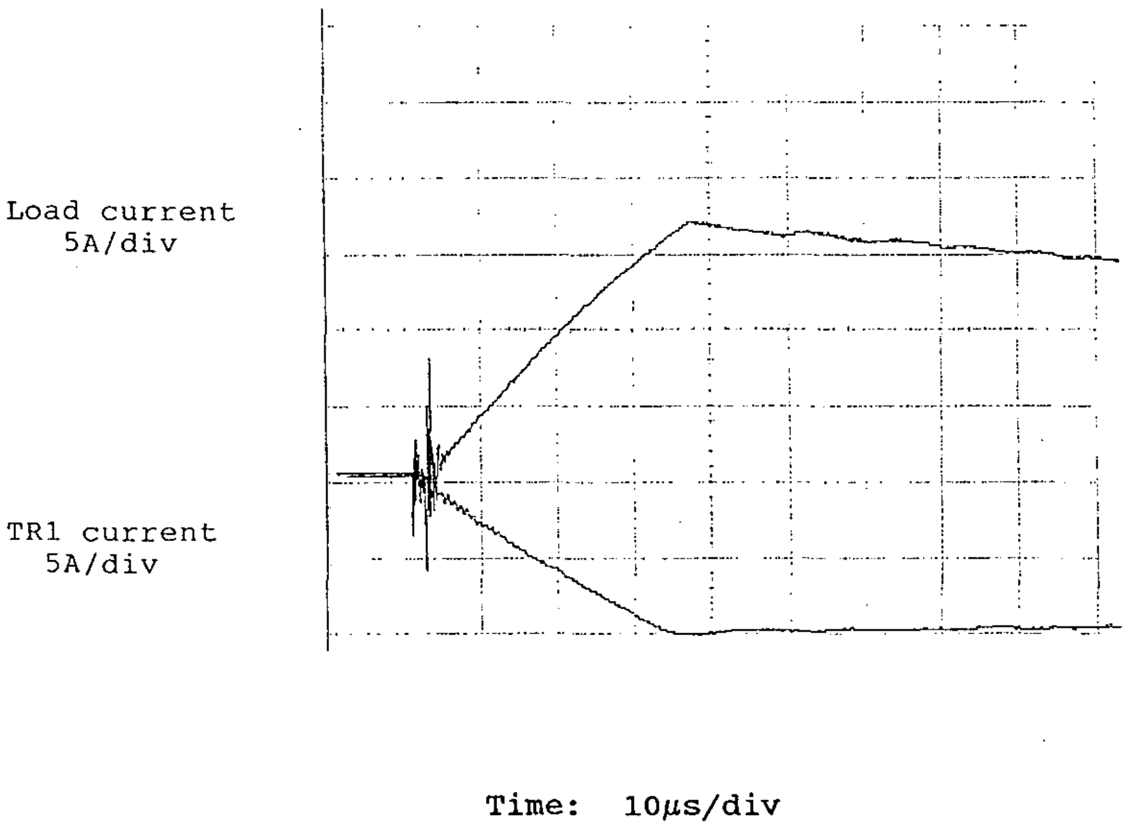
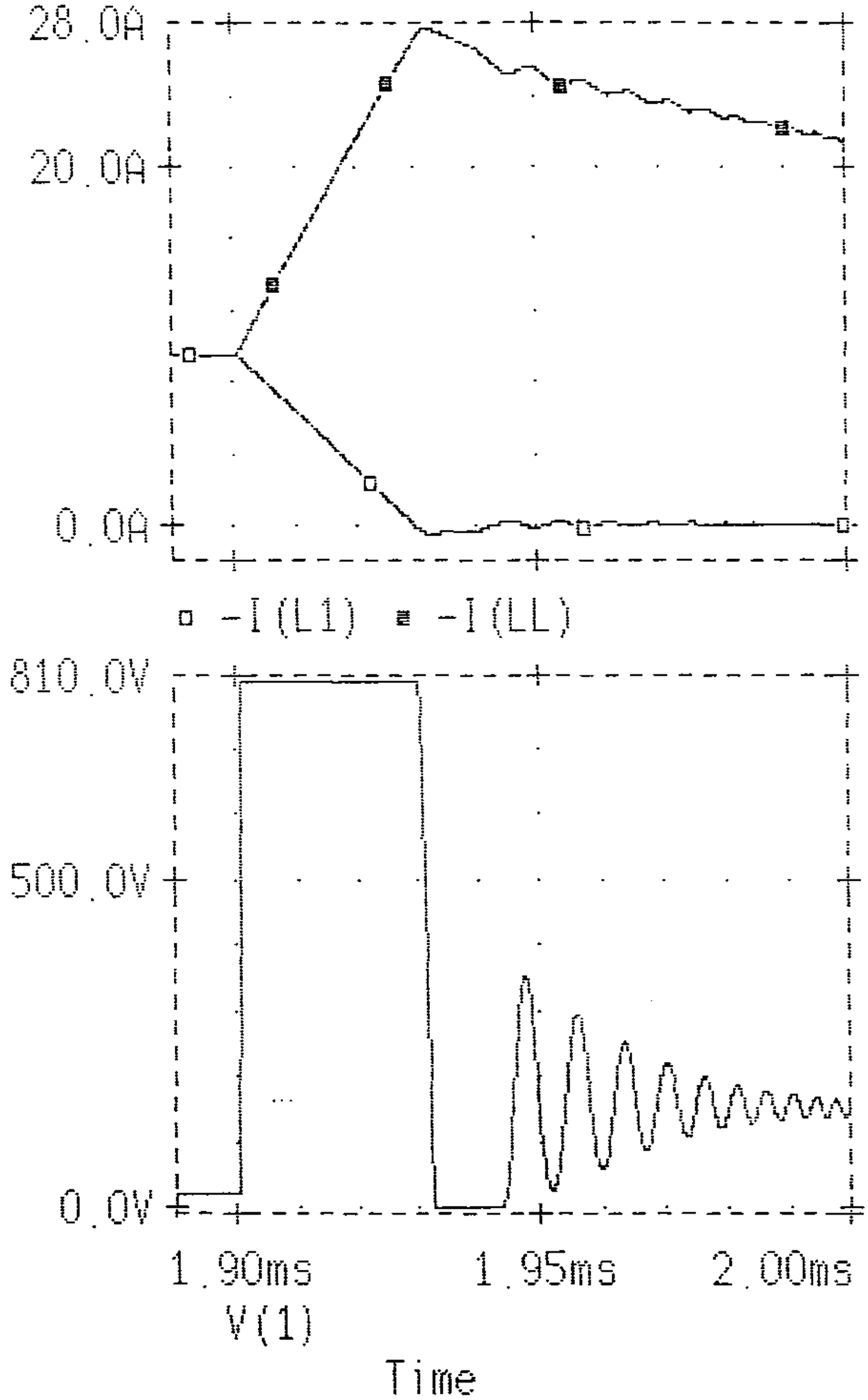


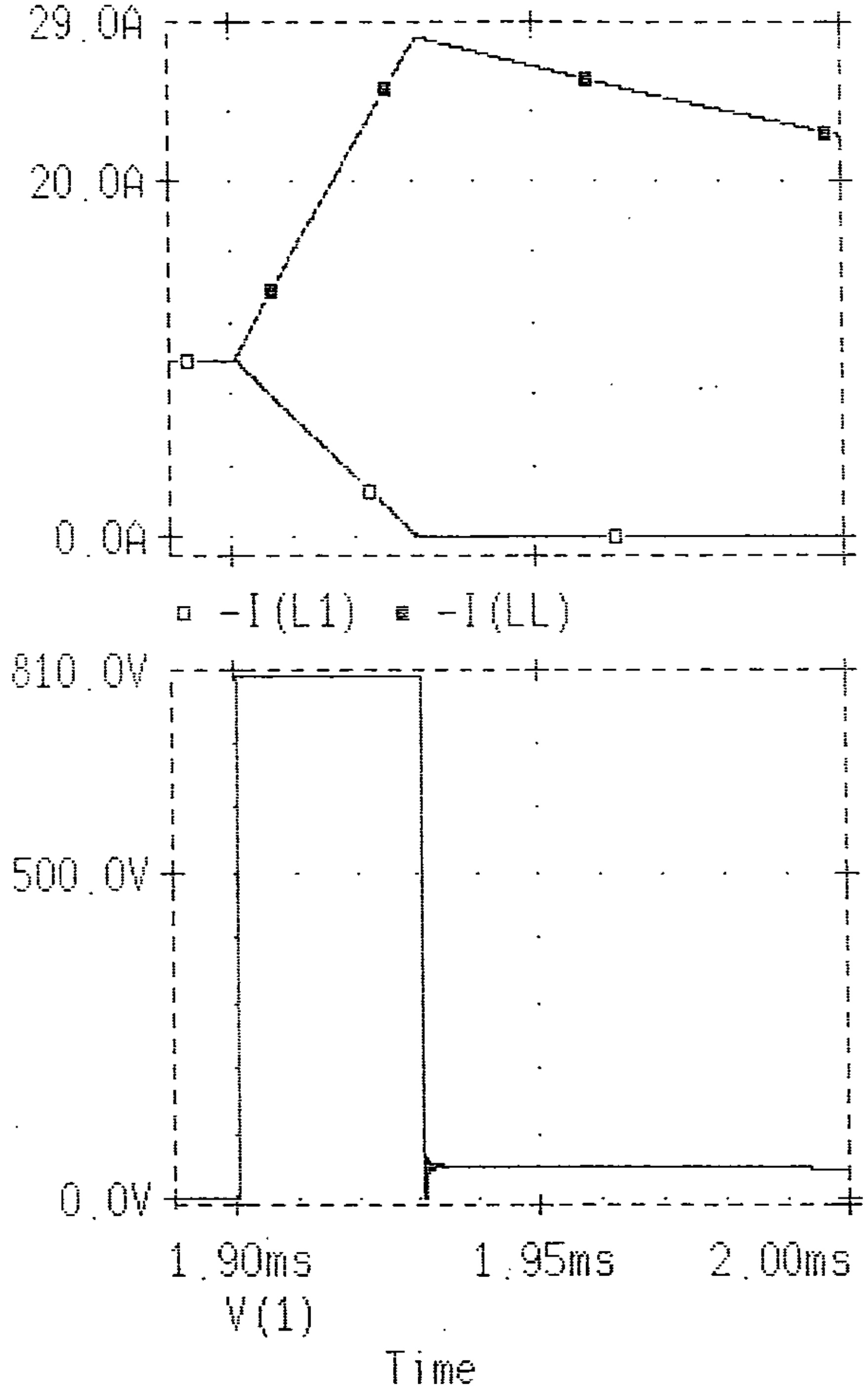
Figure 6.13 Current Waveforms Without Additional Gate Resistor

The oscillations present in the voltage waveform are due to the non-ideal behaviour of the MOSFET, as shown by the simulated waveforms of figure 6.14. These waveforms compare the behaviour of a MOSFET with that of an ideal switch.



(a) MOSFET

Figure 6.14 Waveforms Showing Non-Ideal Behaviour of MOSFET



(b) With Ideal Switch in Place of MOSFET

Figure 6.14 Waveforms Showing Non-Ideal Behaviour of MOSFET

6.5.4.2 Second and Third Experiments

Figures 6.15 and 6.16 show waveforms corresponding to figure 6.12 for the second and third experiments respectively. As expected, the current multiplication is approximately 3 in each case.

Table 6.3 gives the efficiency figures for all three experiments; these values are also indicated on figure 6.8.

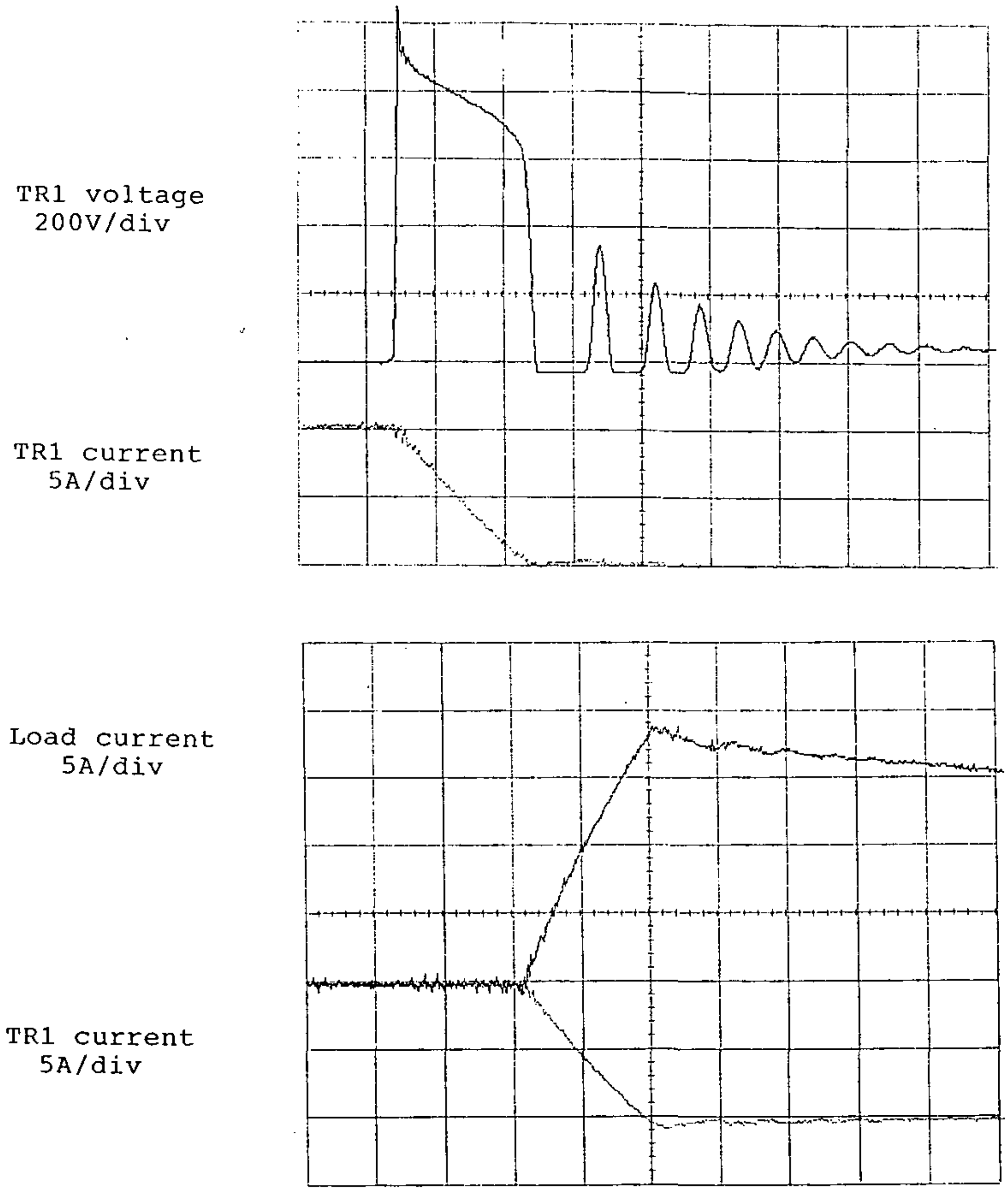
EXPT. NO.	INITIAL CIRCUIT ENERGY (mJ)	FINAL LOAD ENERGY (mJ)	EFFICIENCY (%)
1	169.2	39.2	23.2
2	142.0	42.1 *	29.6
3	152.5	39.2	25.7

* final current was slightly higher in experiment 2

Table 6.3 Efficiency figures for Single-Step Experiments

6.5.5 Comments

The experiments served firstly as a further demonstration of the meatgrinder principle, this time with an uncoupled load. The results showed clearly that a given current multiplication in the load can be achieved with different sets of meatgrinder inductances, only one of which maximises the efficiency. In this case it is the second experiment which corresponds to the maximum efficiency indicated in figure 6.8.



All traces: Time 10μs/div

Figure 6.15 Waveforms for Second Experiment

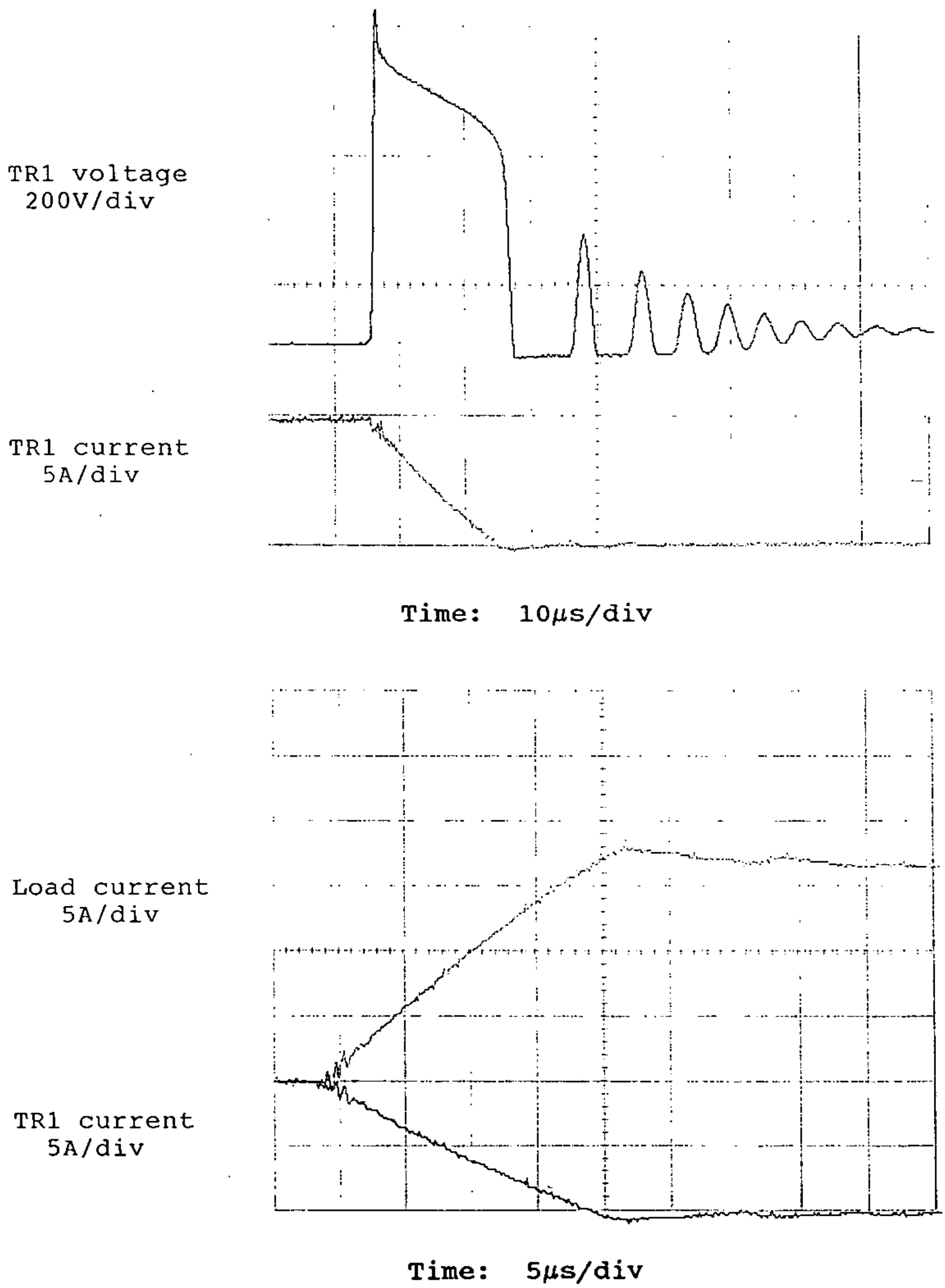


Figure 6.16 Waveforms for Third Experiment

Figure 6.8 shows that the trend of the experimental results is the same as that of the theoretical values. The efficiencies are lower because the coupling coefficient was less than 0.9 and because energy is lost in the circuit resistance. (To reduce this loss it would be necessary to increase the speed of energy transfer by using a higher switch voltage.)

In conclusion, the experimental results clearly support the theoretical analysis presented earlier. This includes the third experiment because although the coupling coefficient was a little low, the current multiplication was not noticeably affected. In the light of the first two experiments, it is reasonable to assume that even if the correct total inductance were to be obtained, the performance would not be significantly different.

6.6 OPTIMISATION STUDY FOR A TWO-STEP MEATGRINDER

6.6.1 Introduction

It has been shown that for a single-step circuit with a given current multiplication and coupling coefficient, there is a unique optimum design which maximises the efficiency. It is logical to consider next whether or not such an optimum design exists for a circuit with more than one step.

With more than one step, any analysis will clearly be more complex. There will be at least three coupling coefficients, and the overall current multiplication is achieved in two or more steps.

In early work [23,27], the team at ECRC adopted the approach of assuming an equal efficiency per step, although this was not presented as an optimum design. Later [28], however, they presented curves showing the "maximum efficiency for a given number of steps", although they did not indicate how they were derived.

The objective of this section is simply to show one approach which could be used in starting a general optimisation study for multi-step meatgrinders.

6.6.2 Problem to be Studied

This study considers a loaded two-step meatgrinder without decompression (see figure 6.17). The coupling coefficients are constrained according to the model proposed by Giorgi et al [28]. In this model the coupling coefficient is k between adjacent coil sections, k^2 between sections separated by one other section, and so on. In a two-step meatgrinder this gives

$$k_{12} = k$$

$$k_{23} = k$$

$$k_{13} = k^2$$

(In a three-step, four-section circuit, this model would give, for example, $k_{14} = k^3$.)

Giorgi states that his model fits well with measured and calculated values for real coil designs. In fact, in the present work about one half of the coupling coefficients for the six-step meatgrinder fit with the model (see table 2.5). The model is referred to subsequently in this thesis as the $k\text{-}k^2$ model.

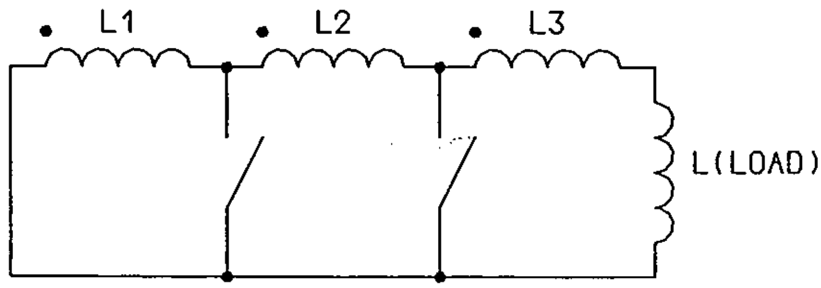


Figure 6.17 Loaded Two-Step Meatgrinder

Any sensible values could be used for the coupling coefficients, but the $k\text{-}k^2$ model seems a reasonable starting point. Such a starting point is even more important in studies of circuits with more than two steps; a six-step circuit, for example, has 21 coupling coefficients.

For the two-step circuit, let the overall current multiplication be β, and let the first and second step current multiplications be β_1 and β_2 respectively. Thus

$$\beta = \beta_1\beta_2 \qquad (6.18)$$

6.6.2.1 Approach

Initially, a generalised algebraic analysis (as carried out for the single-step circuit) was attempted. It did not seem possible, however, to obtain universal expressions for an optimum set of meatgrinder inductances. It was therefore decided to adopt a less general approach and to consider certain specific numerical examples.

Using the additional notation

$$L_1/L_{LOAD} = \sigma_1$$

$$L_2/L_{LOAD} = \sigma_2$$

$$L_3/L_{LOAD} = \sigma_3$$

the procedure followed was:

(1) Choose values for the overall current multplication β and the coupling coefficient k.
(2) Choose a value for β_2 (any value such that $1 < \beta_2 < \beta$).
(3) Choose a value for σ_3. This gives the value of σ_2 since β_2 is already fixed.
(4) Find the value of σ_1 from the values of β_1, σ_2 and σ_3 (β_1 is fixed by β and β_2).
(5) Calculate the individual step efficiencies and the overall efficiency.

These five steps yield a set of meatgrinder inductances and the associated values of efficiency. These inductances are only one possible way of achieving the step current multiplications β_1 and β_2, and β_1 and β_2 are in turn just one way of achieving the overall current multiplication β. The analysis can therefore be continued as follows:

(1) Leave β_1 and β_2 unchanged and find efficiency values for several different sets of σ_1, σ_2 and σ_3, i.e. repeat the procedure above from step (3) onwards.

(2) Divide β up differently, i.e. choose a new value for β_2, and then repeat the entire analysis as before.

In order to carry out some examples of this type of analysis, a Fortran program was written.

6.6.3 Computer Program

The program "mgeff_01" ("meatgrinder_efficiency_01") is listed in Appendix E. To reiterate, the program has two loops and operates as follows:

1. β and k are fixed (although they may be altered by editing the program)
2. Read and validate upper and lower limits and step sizes for both β_2 and σ_3
3. For each value of β_2, step through all the values of σ_3, calculating the corresponding values of σ_2, σ_1, η_{s1}, η_{s2} and η_t in every case.
4. Stop

An example of output from the program is given below. The amount of data in this example is very small; normally the figures would be fed to a plotting program to produce the type of curves given later in this Chapter.

The equations used by mgeff_01 are based on expressions already given in this or earlier Chapters of the thesis.

6.6.3.1 Example Output

mgeff_01

```
++++++++++++++++++++++++++++++++++++++++++++
+                                          +
+               MGEFF_01                   +
+                                          +
+  Optimisation investigation:             +
+  Two-step ideal meatgrinder with         +
+  uncoupled load and no decompression.    +
+                                          +
+   mod.1 28.9.88               mgp 9.88   +
++++++++++++++++++++++++++++++++++++++++++++
```

GAPLOT filename?   test

X2 lower limit?   1.2
X2 upper limit?   1.8
X2 step size?   0.2
r3 lower limit?   0.1
r3 upper limit?   0.5
r3 step size?   0.1

X1= 1.667 X2= 1.200

r3	r2	r1	eff1	eff2	efftot	Result No.
0.100	0.988	3.924	71.5	57.0	40.7	1
0.200	0.588	4.001	67.8	63.5	43.0	2
0.300	0.460	4.184	66.6	63.2	42.1	3
0.400	0.400	4.370	66.4	61.0	40.5	4
0.500	0.367	4.555	66.5	58.4	38.8	5

X1= 1.429 X2= 1.400

r3	r2	r1	eff1	eff2	efftot	Result No.
0.100	3.951	2.702	88.2	33.0	29.2	6
0.200	2.351	2.238	86.7	43.4	37.7	7
0.300	1.839	2.163	85.9	46.9	40.3	8
0.400	1.600	2.182	85.5	47.6	40.7	9
0.500	1.469	2.236	85.3	47.0	40.1	10

X1= 1.250 X2= 1.600

r3	r2	r1	eff1	eff2	efftot	Result No.
0.100	8.890	1.590	95.2	22.6	21.6	11
0.200	5.290	1.175	94.8	32.3	30.6	12
0.300	4.139	1.069	94.5	36.6	34.6	13
0.400	3.600	1.039	94.4	38.3	36.2	14
0.500	3.306	1.040	94.3	38.8	36.5	15

X1= 1.111 X2= 1.800

r3	r2	r1	eff1	eff2	efftot	Result No.
0.100	15.804	0.498	98.9	17.4	17.2	16
0.200	9.404	0.346	98.8	25.9	25.6	17
0.300	7.358	0.303	98.7	30.2	29.8	18
0.400	6.400	0.287	98.7	32.3	31.9	19
0.500	5.878	0.283	98.7	33.1	32.7	20

6.6.4 Results

This section gives results which show how the program can be used to investigate a particular design requirement. The parameters chosen are for illustration only, but could equally well represent a real design problem.

In figures 6.18 to 6.21, X is used for the overall current multiplication of the meatgrinder (β) and X2 for the current multiplication of the second step (β_2). (The different notation was necessitated by the limitations of the computer used to produce the curves.) Each curve represents the variation of efficiency with σ_3 for a particular value of β_2. The corresponding values of σ_1 and σ_2 are available in the output file, and either one could be used as the independent variable for the efficiency plots instead of σ_3.

Figures 6.18 to 6.20 are results for the same values of β and β_2, but with different sets of coupling coefficients. To distinguish between the curves, it should be remembered that the step 1 efficiency is always highest for the highest value of β_2. The symbols enable the corresponding curves on the other two graphs to be identified.

In all three cases the trends are similar, except that when k=0.5 (figure 6.18), the maximum overall eficiency does not increase as β_2 falls, as it does for the other values of k. This phenomenon could be investigated further, although with efficiencies of less than 10%, it is unlikely to be of interest.

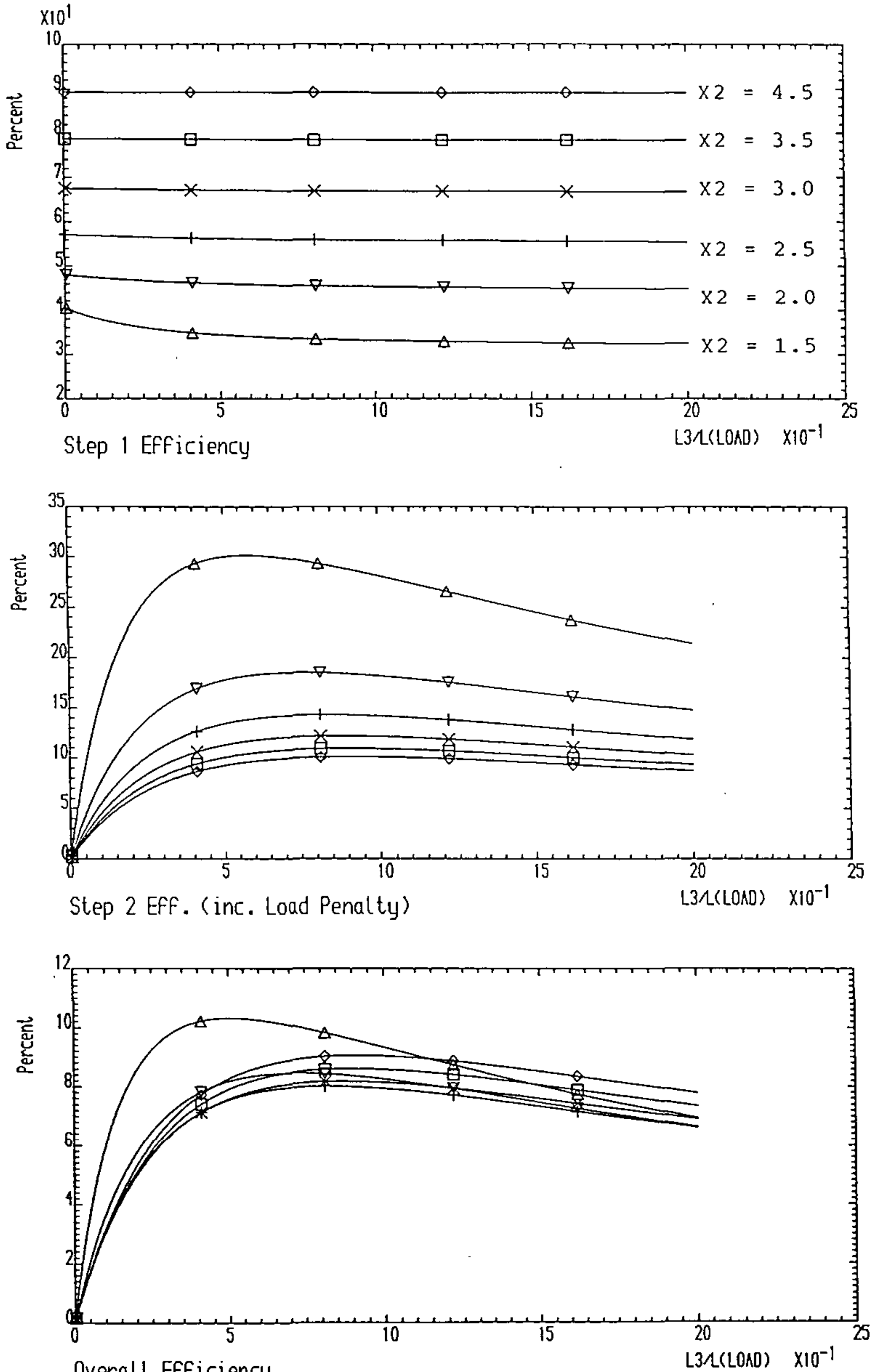


FIGURE 6.18 Two-Step Meatgrinder Efficiency For:
X=5, X2: 1.5, 2.0, 2.5, 3.0, 3.5, 4.5; k12=k23=0.5, k13=0.25

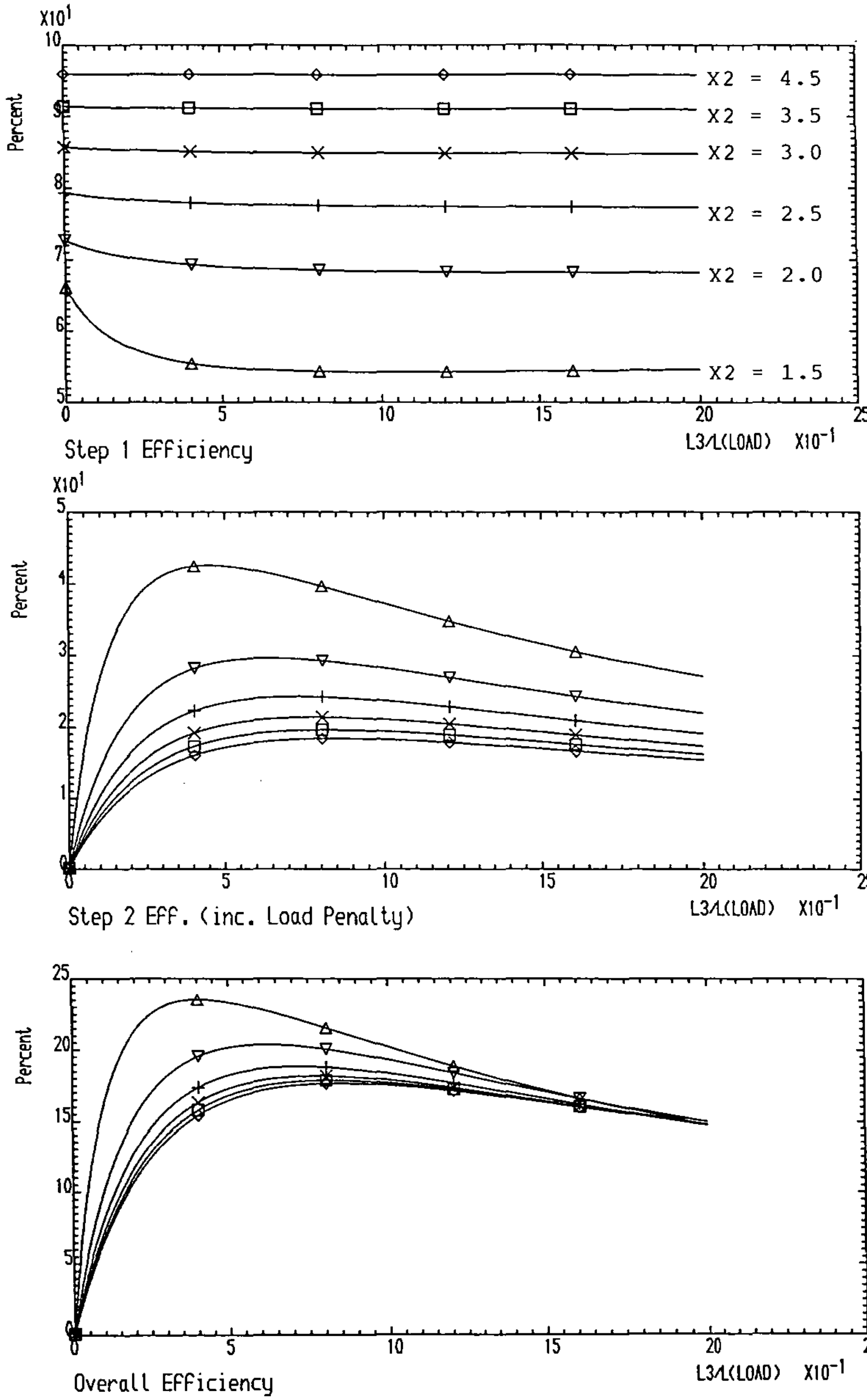


FIGURE 6.19 Two-Step Meatgrinder Efficiency for:
X=5, X2: 1.5, 2.0, 2.5, 3.0, 3.5, 4.5; k12=k23=0.7, k13=0.49

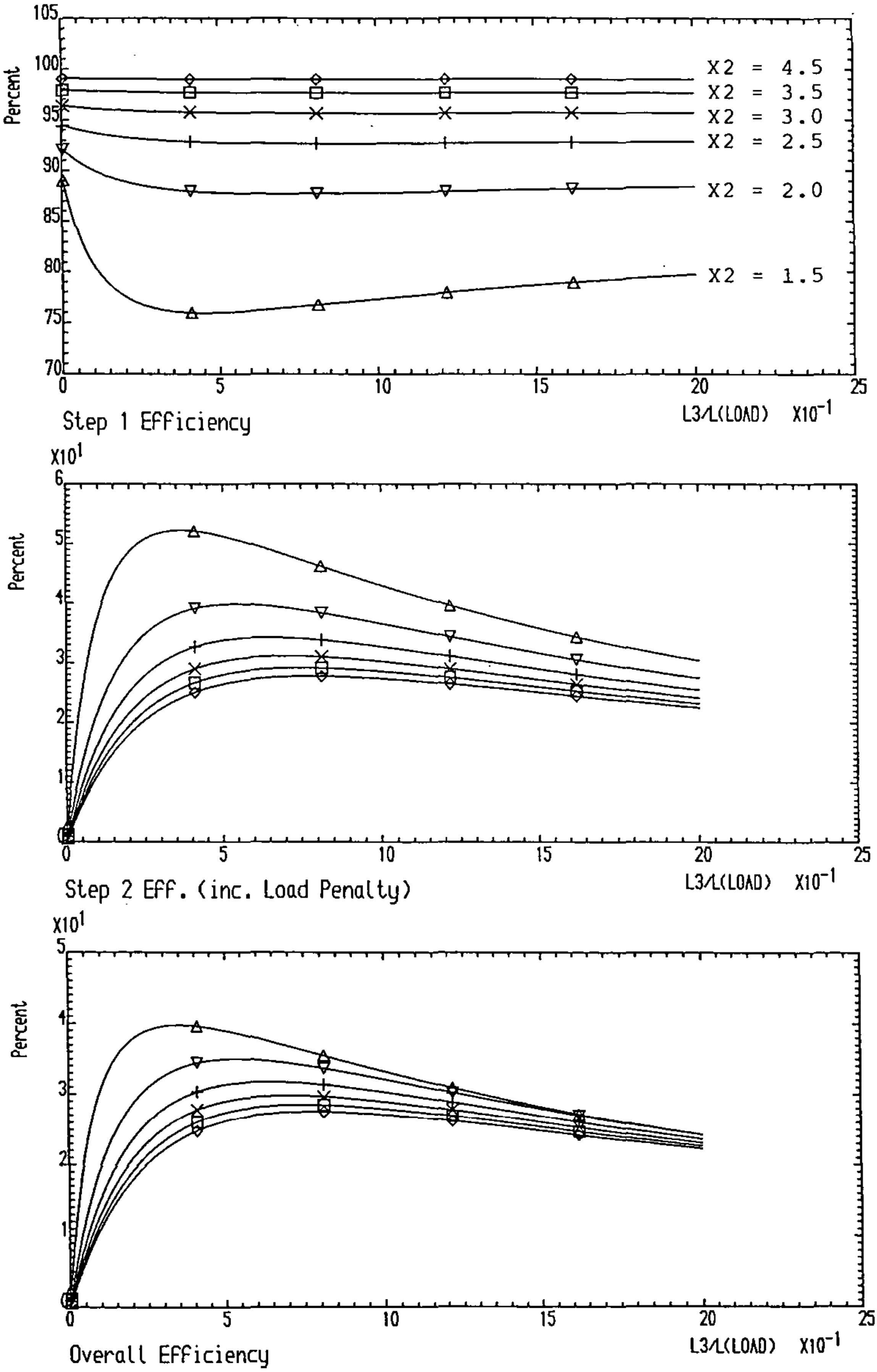


FIGURE 6.20 Two-Step Meatgrinder Efficiency for:
X=5, X2: 1.5, 2.0, 2.5, 3.0, 3.5, 4.5; k12=k23=0.9, k13=0.81

The overall efficiency is significantly higher in the other two cases (figures 6.19 and 6.20). As β_2 falls, and more of the current multiplication is achieved in the first step, the maximum overall efficiency rises consistently. To find out how far this trend continues, the case when k=0.9 is analysed further by running the program with different values of β_2.

The results are shown in figure 6.21. In this case, the lowest value of β_2 (1.1) actually corresponds to the lowest maximum overall efficiency. It is worth noting that the efficiency curve for that particular case exhibits two maxima: the overall maximum which occurs first at a low value of σ_3, followed by a local maximum at about σ_3=4.5. This emphasises the importance of covering a wide range of σ_3 values.

To find the best value of β_2, the maximum efficiency from each curve is plotted on a separate graph. The optimum value can be seen clearly in figure 6.22. To achieve this maximum requires the following inductance ratios:

$\sigma_1 = 41$
$\sigma_2 = 1$
$\sigma_3 = 0.33$

To implement this design it would be necessary to find a physical coil design which produced these ratios whilst maintaining the required coupling coefficients.

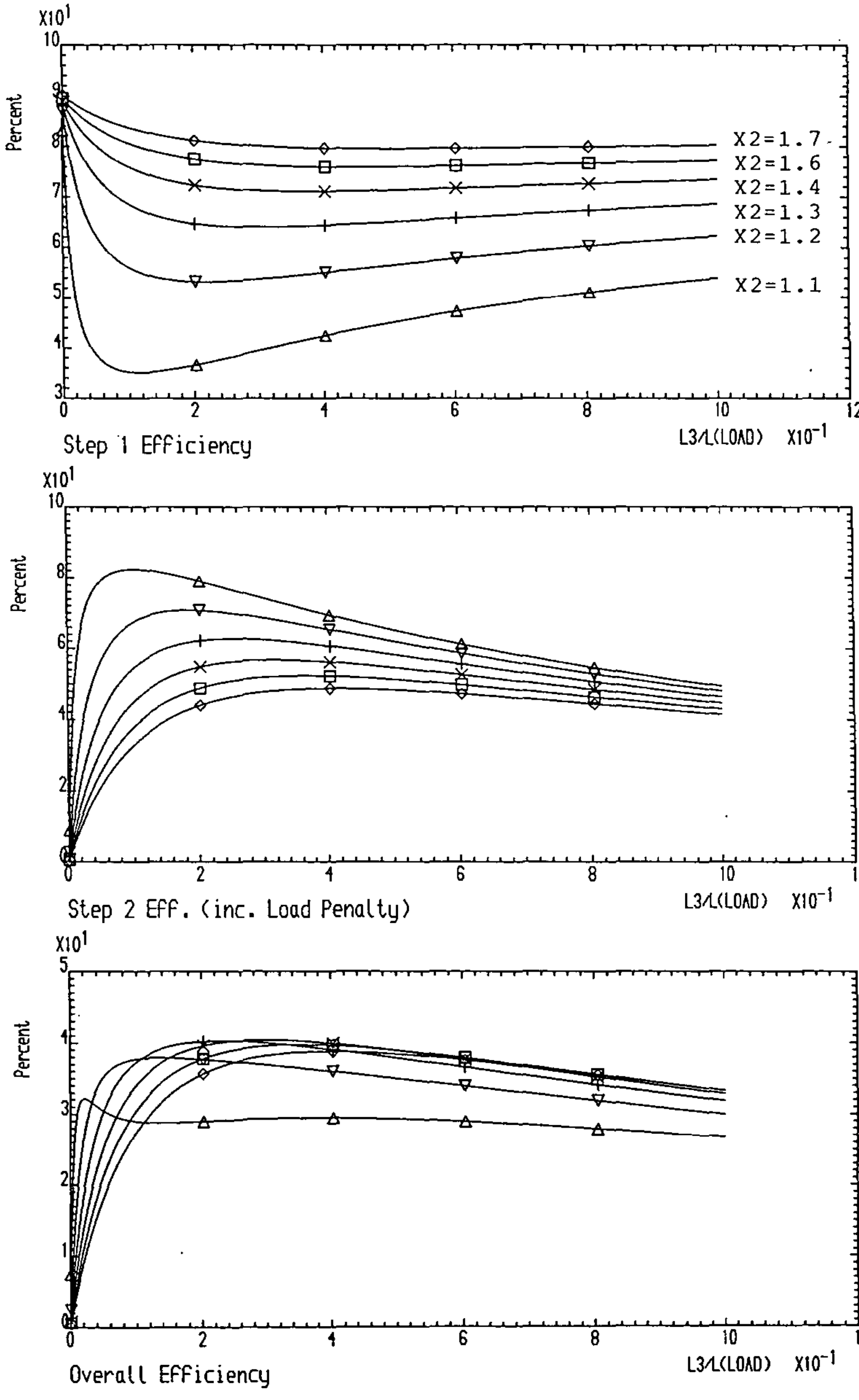


FIGURE 6.21 Two-Step Meatgrinder Efficiency For:
X=5, X2: 1.1, 1.2, 1.3, 1.4, 1.6, 1.7; k12=k23=0.9, k13=0.81

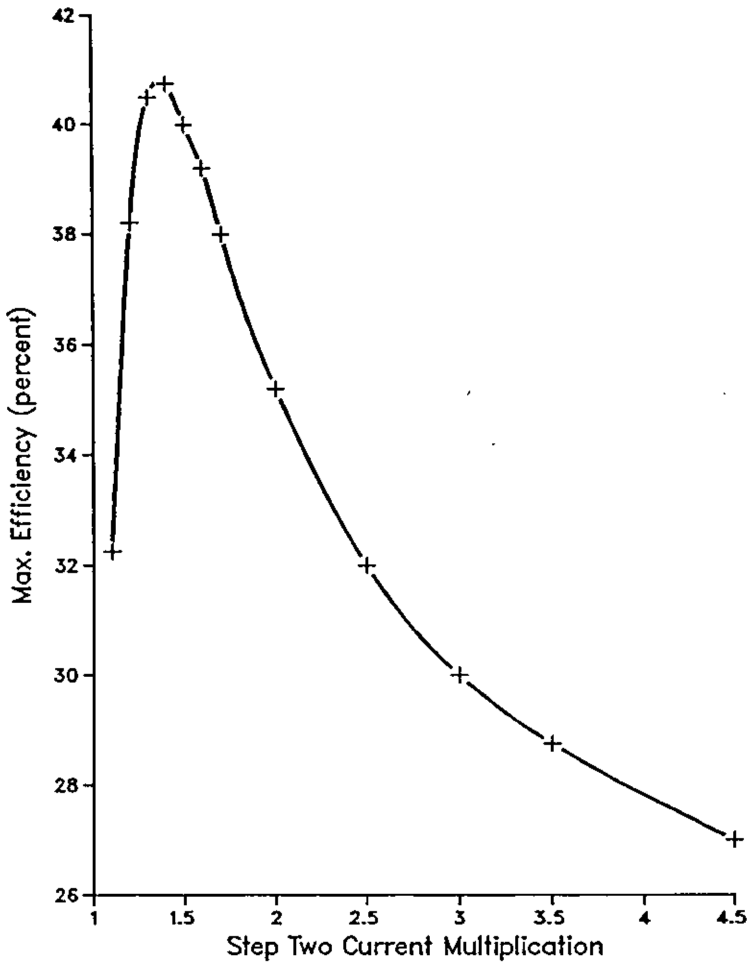


Figure 6.22 Example of Optimum Condition in Two-Step Meatgrinder:

$X = 5$

$k_{12} = 0.9$

$k_{23} = 0.81$

No decompression

6.6.5 Comments

This section has shown how optimisation might be achieved by fixing some of the variables and then investigating the variation of efficiency empirically rather than analytically. It seems likely that this method could also be used to investigate circuits with three or more steps.

One method of optimising the design of multi-variable systems is called factorial design [54]. The technique involves varying parameters one at a time, two at a time and so on, and examining the effect on the system output. This method has been applied to the study of electrical machines [55,56] but does not appear to be appropriate for multi-step meatgrinder circuits. The reason for this is that if the overall current multiplication is fixed, then inductances, for example, cannot be varied one at a time. In other words, the variables are interdependent. It should be noted, however, that this research has not addressed factorial design in detail, and its use in future optimisation work is therefore not excluded.

CHAPTER SEVEN

CONCLUSIONS

7.1 ACHIEVEMENTS OF THE RESEARCH

The work described in this thesis constitutes a thorough investigation of the major principles associated with the meatgrinder concept. A firm foundation has now been laid for further research on this topic.

The initial objective stated in Chapter 1 was to gain familiarity with the concepts and techniques involved in the meatgrinder idea. This has been achieved. The fundamental circuit characteristics are now clearly understood, which leaves the way open for more detailed studies into the many possible applications of this type of circuit.

It has been confirmed that the meatgrinder concept offers a means of transferring energy between uncoupled inductors at efficiencies greater than 25%. The higher efficiency reduces the demand on both the power supply and the opening switches, thereby simplifying the design of a pulsed power system. It has also been verified that this technique of transferring energy between magnetically-coupled coils provides current multiplication. This avoids the need for conventional transformers, which are also subject to the 25% efficiency limit when transferring energy to an uncoupled load [25].

For a sufficiently large number of meatgrinder coil sections, the theoretical efficiency tends towards 100%. In practice the efficiency is limited by the coupling coefficients which can be obtained, the number of stages used and the circuit resistance, but there appears to be no reason why circuits should not be built which operate at efficiencies of well over 75%.

The effect of induced voltages in the meatgrinder has been studied at length in this research. Of particular importance is the fact that the induced voltages can be sufficiently high to cause voltage protection (clamp) devices to break down. This is the phenomenon referred to in the thesis as transformer action clamping (TAC). Previous work on this topic [13,27,43] has been confirmed and extended. A mathematical proof has been presented which shows that the theoretical current multiplication is unaffected by the occurrence of external transformer action clamping (ETAC - when the clamping is due to voltages induced across previously switched-out coil sections). It has been shown that this is simply an instance of the principle of the conservation of flux linkage, and that the same principle applies to internal transformer action clamping (ITAC - when the clamping is due to voltages induced across coil sections which are still in the circuit).

It has further been demonstrated that the occurrence of TAC always increases the transfer time, thereby leading to additional energy loss if the circuit resistance is significant. This has led to two general design requirements: firstly, that the resistance should be minimised (it is worth noting that in the low-current experiments carried out for this research, and in similar low-current work carried out elsewhere [43], it is the on-state resistance of the switches, rather than the coil resistance, which dominates), and secondly that the higher the voltages in the system are allowed to rise, the greater will be the energy transfer speed.

The work carried out on optimisation has shown that a single-step meatgrinder can be designed so as to maximise the efficiency for a given current multiplication. Such optimisation will be of vital importance in applications such as the proposed 100TW experiment [35], and will enable this simple implementation of the meatgrinder concept to be exploited to the full.

As far as multi-step circuits are concerned, the research has demonstrated one possible approach to optimisation. The results indicate that for a given set of constraints (for example, the overall current multiplication, the individual step current multiplications, and the coupling coefficients), it will again be possible to produce designs which maximise the efficiency.

The use of a commercial circuit simulation package has clearly been of great value. This will continue to be the case where the principles of circuit operation are being studied, but may become less appropriate as other factors need to be included in the simulation. These factors could include transient behaviour, parasitic components, or the characteristics of high-energy switches; these factors are not available in existing commercial packages.

Computer simulation needs to be complemented by experimental results, and the low-current experiments carried out have shown the appropriateness of power MOSFETs for switching. They are simple to drive and, because they carry current for such a short time, need no heatsinks. At present it seems likely that other switching techniques (see Chapter 1) will be necessary for anything other than demonstration circuits, although MOSFETs (or other semiconductor devices) may be of value if used in multiple arrays [44].

7.2 SUGGESTIONS FOR FURTHER WORK

7.2.1 Optimisation

Further work is required on design methods for multi-step meatgrinders. There are two major aspects to this problem.

The first aspect is the type of analysis shown in Chapter 6, where the approach is to find inductance values (or ratios) without reference to physical implementation. Studies could be carried out for circuits with two, three or more steps, covering a wide range of coupling coefficients and current multiplications. The results of such studies should provide a guide as to performance capabilities, which can then act as a starting point for more detailed investigations.

The second aspect concerns the physical implementation of coil designs. A physical design must produce the required inductances and coupling coefficients. This in itself can present significant problems, even laying aside the initial problem of choosing which geometry to use. This is because, as other authors have observed [57,58], inductance formulae usually yield an inductance value for a given set of dimensions. The designer, on the other hand, needs to know what dimensions to use to obtain a given inductance. The problem is compounded when the aim is to design a multi-section coil with given coupling coefficients between the sections. Computers can be of great asssistance [36], but there will often be an element of trial and error in the procedure.

Choice of geometry is a problem facing all coil designers, and there is consequently an abundance of literature on this topic [57-74]. The first major choice is between geometries such as the toroid, which largely confine flux to within the volume of the coil, and geometries such as the solenoid which do not. The choice is not always obvious, and several authors have published papers [59-61] which include comparisons between coil types on the basis of other factors such as resistance, coupling coefficient and mechanical forces.

Literature can be found on most different coil geometries, such as solenoid [58], spiral [62], coaxial [63] and toroidal [64]. In particular, a toroid known as a cage coil has been investigated in detail at Loughborough [65-68] and is worthy of investigation as a meatgrinder coil because of its flux-confining property and its ease of construction in comparison to more conventional toroids.

An area which has been largely untackled in this research is that of the transient current distribution in coils when they are subjected to very fast voltage or current pulses. As the currents and voltages involved increase, and the pulse rise times are reduced, it will become necessary to account for the fact that the resistance, stray capacitance, and even the inductance of a coil all vary with frequency.

The principles governing these variations are described in several texts on electromagnetism [69-71], and authors have also considered the a.c. response of specific coil geometries [63,65,72,73]. Work is required, however, to turn this information on the steady-state a.c. response into knowledge of the single-shot response. This may not be very straightforward: Grover [39], commenting simply on a.c. response, states:

> "...the high frequency resistance and inductance of coils cannot be accurately calculated and should be measured at the desired frequencies."

and Zowarka [74], recognising the difficulty of predicting dynamic behaviour, adopts the approach of building physical scale models to verify parameters.

Note also that the effect of mechanical forces during transients will need to be studied.

The experimental results obtained in this research agree well with computer simulations, despite measurements having been made at a relatively low frequency (1kHz for the six-step meatgrinder, 10kHz for the single-step meatgrinder). This is a good indication that transient effects have not been very significant to date. It is likely, however, that an understanding of transient effects will be important in constructing full-scale meatgrinder circuits.

7.2.2 Reverse Operation

When the meatgrinder was proposed [20] it was introduced as a reversible circuit. This refers to the fact that because the efficiency can theoretically be 100%, the circuit could be operated "forwards" (i.e. in the normal manner) and then "backwards" (i.e. re-introducing coil sections, thereby causing the current to fall), leaving the circuit in its original state with the same amount of energy.

Practical circuits are not 100% efficient, but the possibility of operating the circuit in reverse still remains (see figure 7.1).

There appears to be no published work on this aspect of the meatgrinder operation, other than the proposal to use a single-step circuit for recovering the energy left in the barrel of an electromagnetic gun [32].

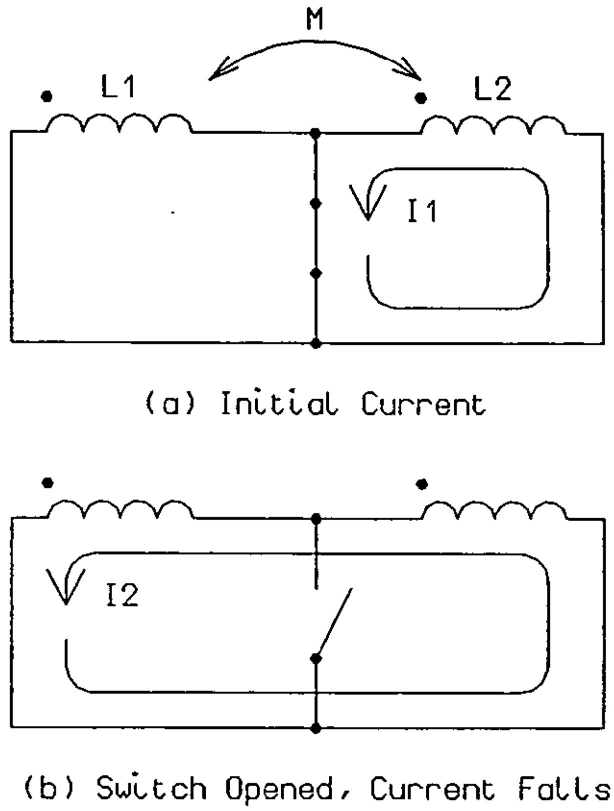


Figure 7.1 Reverse Operation of Meatgrinder

A meatgrinder changes inductance as well as current. It may be possible to utilise this fact in inductance matching applications. Figure 7.2 illustrates how this might work, with the objective being to transfer energy from the energy source to L_{LOAD}.

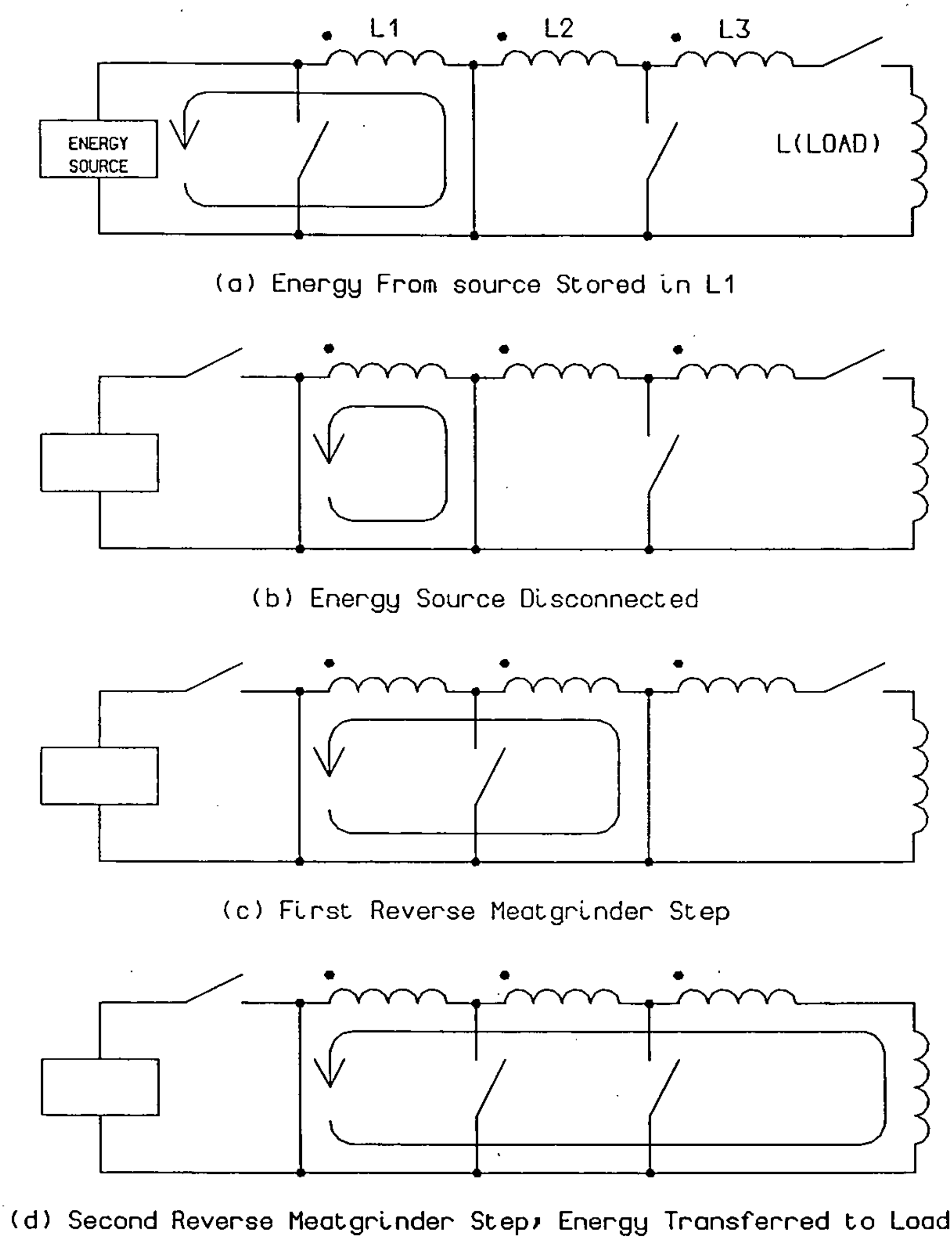


Figure 7.2 Inductance Matching by Reverse Operation of Meatgrinder

If, for example, the energy source is an explosive generator, the voltage it produces exists for only a short time, and the energy which can be extracted in this time will vary according to the inductance connected to it. L_1 can be made as small as necessary in order to extract the required amount of energy, the meatgrinder then being operated as shown in order to transfer energy to the final load. With the circuit in the condition shown in figure 7.2(d) there will, of course, still be energy stored in the inductances L_1 and L_2. It may be possible to subsequently operate the meatgrinder in the forward or normal mode, thus switching L_1 and L_2 out of the circuit and transferring more energy to the load.

Investigations of this and any other possible use of reverse operation will require analysis, computer simulation and experimentation.

7.2.3 Reduction of Induced Voltages

In Chapter 5 it was concluded that in a multi-step meatgrinder with the usual parallel switches, the switch voltage ratings need to be as high as possible in order to maximise the speed of energy transfer. This approach will lead to satisfactory solutions as long as appropriate switches can be obtained and all the circuit components are suitably insulated. It is inevitable, however, that some implementations will be impractical because of very high induced voltages.

The use of series switches, as discussed in Chapter 5, is one possible solution [43]. It does, however, have the drawback of the extra switch resistance. It is suggested that work could be carried out in order to find an alternative solution.

The essence of the problem is that once a coil section is switched out it serves no further purpose, but still remains magnetically coupled to the rest of the circuit. Equation (5.5) shows that the induced voltage could be made smaller either by reducing the inductance of the switched-out section or by reducing the coupling coefficient between this section and the rest of the coil.

Coupling coefficients can be changed by mechanically altering the relative orientation of the coils. This could be investigated, but would probably only be of value for applications working in the millisecond/second time regime. In addition, it should be noted that the process of changing the mutual inductances would itself induce voltages in the circuit.

Aboltin'sh [75] has indicated that the inductance of a coil can be varied by the nearby presence of a semiconductor, the variation being achieved by the current flow in that semiconductor. It is possible that this technique could be used to reduce the inductance of a switched-out section. A further possibility is the use of an additional winding to generate an equal and opposite induced voltage. Such a "cancellation winding" would be wound in the opposite sense to the main winding and would be switched in when required. This approach would admittedly add considerable complexity to the meatgrinder, thereby making it appreciably less attractive as a pulsed power device.

7.2.4 Secondary Topics

This section discusses two areas which could usefully be studied in conjunction with the more important topics referred to above.

7.2.4.1 Coil Width Optimisation

In Chapter 2 it was noted that the width of the spiral strip coil used for the six-step meatgrinder was not optimised to minimise the resistance. Future designs using this design could benefit from such an optimisation study.

7.2.4.2 Timing Circuit

The timing circuit designed for this research serves its purpose well and has been used for both six-step and single-step meatgrinders. Should there be a need for more outputs or longer delays, however, it would become rather unwieldy.

A more elegant design could provide simple expansion of the number of outputs and a more compact method of programming delays. It may be appropriate to use a microprocessor, which would also offer the potential of implementing more sophisticated functions. These could include sensing the meatgrinder currents to enable the microprocessor to optimise the delays over a number of shots.

7.3 THE FUTURE OF THE MEATGRINDER

Analysis and successful experiments have shown the potential of the meatgrinder in a range of pulsed power applications [23,26-36,38,43]. A member of the team at the Energy Compression Research Corporation has observed [76], however, that the technique has not attracted as much interest as might have been expected.

One reason for this may be the apparent complexity of multi-step circuits. There may be a perception that any benefits offered by the circuit would be outweighed by the need for multiple switches and multiple coils. If this is the case, then more emphasis needs to be placed on the value of the single-step circuit [26,28,32,33,35]. If the use of this circuit became an established technique, researchers would be more willing to investigate the advantages of adding more steps. In this respect, it is clear also that optimisation studies are important because they will enable the design of multi-step circuits to be systemised and made less daunting.

The potential of the meatgrinder has neither been fully explored nor fully exploited. With its high efficiency, its current multiplication and its ability to work in reverse, there are potentially many areas of application in pulsed power technology. Perhaps in time the meatgrinder will take its place in this field alongside the Marx generator and the simple transformer. Alternatively, it could be that the concept will find a niche in an area as yet unconsidered, in one of the "novel applications" spoken of in Chapter 1.

REFERENCES

(1) WELDON, W.F., "Pulsed Power Packs a Punch", IEEE Spectrum March 1985, pp 59-66

(2) ROSE, M.F., "Techniques and Applications of Pulsed Power Technology", Proc. 16th Intersociety Energy Conversion Engineering Conf. ("Technologies For The Transition"), Atlanta 1981, Vol 1, pp 146-154

(3) YABLOCHNIKOV, B.A., "An Inductor for Magnetic Pulse Welding of Thin-Walled Large Diameter Pipes", Automatic Welding (GB) Vol 36(1), 1983, pp 50-53

(4) HERBERT, J.L. et al, "Pulsed Power Applications in Aircraft Qualification Testing", Proc. 6th IEEE Pulsed Power Conf. 1987, pp 326-329

(5) GRAY, E.W., MOENY, W.M. et al, "Pulsed Power Fracturing of Rock", Proc. 6th IEEE Pulsed Power Conf. 1987, pp 330-331

(6) FRAZIER, G.B., ASHBY , S.R. et al, "EAGLE: A 2TW Pulsed Power Research Facility", Proc. 3rd IEEE Pulsed Power Conf. 1981, pp 8-14

(7) NEAU, E.L., "COMET: A 6MV 400kJ Magnetically-Switched Pulsed Power Module", Proc. 4th IEEE Pulsed Power Conf. 1983, pp 246-250

(8) HARRISON, J., MILLER, R. et al, "Design of Very Fast Rise and Fall Time Low Impedance Megavolt Pulse Generators for Laser Excitation", Proc. 1st Int. Conf. on Energy Storage, Compression and Switching 1974, pp 51-56

(9) LEVY, S., FARBER, J. et al, "Advances in Repetitive Pulsed Power Technology", Proc. 17th IEEE Power Modulator Symposium 1986, pp 9-20

(10) FOWLER, C.M., ZIMMERMAN, E.L. et al, "Railguns Powered by Explosive Driven Flux Compression Generators", IEEE Trans. Magnetics Vol 22(6), 1986, pp 1475-1480

(11) TRIPOLI, G.A. et al, "Design and Operation of a 0 to 4MJ Current Multiplier Inductive Store With Multiple Exploding Foil Switching", Proc. 5th IEEE Pulsed Power Conf. 1985, pp 22-25

(12) SALGE, J., BRAUNSBERGER, U. et al, "Circuit Breaking by Exploding Wires in Magnetic Energy Storage Systems", Proc. 1st Int. Conf. on Energy Storage, Compression and Switching 1974, pp 477-480

(13) LEGENTIL, M., RIOUX, C., "Caracteristiques des Transferts Inductifs à Haut Rendement par Procede Dissipatif", ("Characteristics of a High Efficiency Inductive Transfer System Using a Dissipative Transfer Procedure"), Revue de Physique Appliquee Vol 11(2), 1976, pp 337-342

(14) BALL, S.E., BURKES, T.R., "Saturable Inductors as High-Power Switches", Proc. 3rd IEEE Pulsed Power Conf. 1981, pp 269-272

(15) SARJEANT, W.J., ROHWEIN, G.J., "A Review of Repetitive Pulse Power Component Technology", Proc. 4th IEEE Pulsed Power Conf. 1983, pp 631-637

(16) BELL, W.R., "An Electrical Engineer's Perspective of High Voltage Pulsed Power", Proc. Universities Power Engineering Conf., Sunderland Polytechnic 1987, paper 12/06

(17) WIPF, S.L., "Reversible Energy Transfer Between Inductances", Proc. 1st Int. Conf. on Energy Storage, Compression and Switching 1974, pp 469-475

(18) ZUCKER, O., "Fundamental Concepts of Energy Compression", Proc. 2nd Int. Conf. on Energy Storage, Compression and Switching 1978, pp 55-73

(19) ZUCKER, O., BOSTICK, W.H., "Theoretical And Practical Aspects of Energy Storage And Compression", Proc. 1st Int. Conf. on Energy Storage, Compression and Switching 1974, pp 71-93

(20) ZUCKER, O., LONG, J., "The Meatgrinder: a Reversible Inductive Storage and Transfer System", Proc. 4th IEEE Pulsed Power Conf. 1983, pp 375-379

(21) RIOUX, C., "Theorie Simplifiee des Procedes de Transfert Inductif à Haut Rendement", ("Simplified Theory of High Efficiency Inductive Transfer Procedures"), Revue de Physique Appliquee Vol 10(2), 1975, pp 75-79

(22) ALLANO, S., RIOUX, C. et al, "Study of a Non-Dissipative Transfer System and Tests on Related Vacuum Switches", Proc. 6th IEEE Pulsed Power Conf. 1987, pp 449-452

(23) ZUCKER, O. et al, "The Meatgrinder: Theoretical and Practical Limitations", IEEE Trans. Magnetics Vol 20(2), 1984, pp 391-394

(24) WIPF, S.L., "High-Current Pulses From Inductive Stores", Proc. 9th Symp. on Engineering Problems of Fusion Research 1981, pp 1442-1446

(25) LOTOTSKII, A.P., "On the Effectiveness of Magnetic Energy Transmission From Inductive Stores", Electric Technology USSR (USA) No. 2, 1985, pp 119-124

(26) GIORGI, D., LINDNER, K. et al, "Enhancing the Transfer of Inductive Energy to an Imploding Plasma Load With a Single-Step Meatgrinder Circuit", IEEE Trans. Magnetics Vol 23(3), 1987, pp 1913-1918

(27) GIORGI, D., LINDNER, K. et al, "Proof of Principle Experiment of the Meatgrinder: an Inductive Energy Storage and Transfer Circuit", Proc. 5th IEEE Pulsed Power Conf. 1985, pp 295-298

(28) GIORGI, D., LONG, J. et al, "New High-Current Meatgrinder Experiments", IEEE Trans. Magnetics Vol 22(6), 1986, pp 1485-1488

(29) LINDNER, K., LONG, J. et al, "Fuse Opening Switch Performance Enhancement", Proc. 6th IEEE Pulsed Power Conf 1987, pp 565-568

(30) LINDNER, K., LONG, J. et al, "A Meatgrinder Circuit for Energising Resistive and Varying Inductive Loads (EM Guns)", IEEE Trans. Magnetics Vol 22(6), 1986, pp 1591-1596

(31) LINDNER, K., LONG, J. et al, "Novel Inductive Energy Storage and Transfer Circuit for Energising Resistive and Varying Inductive Loads (EM Guns)", Proc. 17th Power Modulator Symp. Seattle USA 1986, pp 305-310

(32) ZUCKER, O., LONG, J. et al, "Theory, Experiments and Applications of Meatgrinder Circuits", Proc. 4th Conf. Megagauss Field Generation and Related Topics 1986, pp 609-618

(33) LINDNER, K., LONG, J. et al, "Enhanced Energy Transfer to an EM Gun", Proc. 6th IEEE Pulsed Power Conf. 1987, pp 749-752

(34) NESS, R.M., CHU, E.Y., "Multicycle Resonant Energy Recovery for Inductive Energy Applications", Proc. 6th IEEE Pulsed Power Conf. 1987, pp 38-41

(35) ZUCKER, O., LONG, J. et al, "The Generation of 100TW With an Explosive Generator-Driven Single-Step Meatgrinder", Proc. 4th Conf. Megagauss Field Generation and Related Topics 1986, pp 619-629

(36) GIORGI, D., LINDNER, K. et al, "The Design and Analysis of a Multistage Meatgrinder Circuit", Proc. 5th IEEE Pulsed Power Conf. 1985, pp 615-618

(37) LONG, J., LINDNER, K. et al, "Analysis and Comparison of Circuits Undergoing a Change of Inductance via Continuous Sequential Switching", Proc. 4th Conf. Megagauss Field Generation and Related Topics 1986, pp 593-607

(38) GIORGI, D., HELEVA, H. et al, "The Ringer: An Efficient, High Repetition Rate Circuit for Electromagnetic Launchers", IEEE Trans. Magnetics Vol 25(1), 1989, pp 203-206

(39) GROVER, F.W., "Inductance Calculations", Instrument Society of America 1973

(40) YOUDAN, P., "How the Semiconductor War was Won", Electrical Review Vol 217(13), 1985, pp 38-42

(41) CARLISLE, B.H., "GTOs Challenge SCRs and Darlingtons", Machine Design Vol 57(267), 1985, pp 70-74

(42) MOORE, J., "The Continuing Relevance of the Bipolar Transistor", Electronic Engineering Vol 56(694), 1984, pp 79-87

(43) LONG, J., GRIFFIN, A. et al, "Experimental Demonstration of Novel All-Inductive Modulator", Energy Compression Research Corp. report, 1988

(44) SANTAMARIA, G.T., NESS, R.M., "High Power Switching Using Power FET Arrays", Proc. 6th IEEE Pulsed Power Conf. 1987, pp 161-164

(45) GRANT, D., "HEXFET III: A New Generation of Power MOSFETs", HEXFET Power MOSFET Designer's Manual HDB-4, pp I5-I19, International Rectifier Corp. 1987

(46) DE LORENZI, A., GAIO, E. et al, "Performances of ZnO Varistors in Parallel Operation", Proc. 6th IEEE Pulsed Power Conf. 1987, pp 623-626

(47) GENERAL ELECTRIC CO., "Transient Voltage Suppression", General Electric Co. 1978

(48) GENERAL SEMICONDUCTOR INDUSTRIES INC., "Product Data Book", 10th edition, General Semiconductor Industries Inc.

(49) CLEMENTE, S., "Gate Drive Characteristics and Requirements for Power HEXFETs", HEXFET Power MOSFET Designer's Manual HDB-4, pp I67-I74, International Rectifier Corp. 1987

(50) CLEMENTE, S., PELLY, B.R. et al, "Understanding HEXFET Switching Performance", HEXFET Power MOSFET Designer's Manual HDB-4, pp I85-I98, International Rectifier Corp. 1987

(51) MORRISON, R.,"Grounding and Shielding Techniques in Instrumentation ", Wiley Interscience 1986 (3rd Edition)

(52) JONES, B.K., "Electronics for Experimentation and Research", Prentice-Hall International 1986

(53) WELSBY, V.G.,"The Theory and Design of Inductance Coils", Macdonald 1964

(54) KEMPTHORNE, O.,"The Design and Analysis of Experiments", Wiley 1952

(55) SMITH, I.R., HAMILL, B., "Effect of Parameter Variations on Induction Motor Transients", Proc. IEE Vol 120(12), 1973, pp 1489-1492

(56) KETTLEBOROUGH, J.G. et al, "Simulation of a Dedicated Aircraft Generator Supplying a Heavy Rectified Load", Proc. IEE Part B Vol 130(6), 1983, pp 431-435

(57) BERTINI, S. et al, "Optimised Design of Air-Core Inductors for Applications in Power Electronics", Proc. IASTED Int. Symp., Lugano, Switzerland 1987, pp 27-29

(58) MURGATROYD, P.N., "The Brooks Inductor: a Study of Optimal Solenoid Cross Sections", Proc. IEE Part B, Vol 133(5), 1986, pp 309-314

(59) MURGATROYD, P.N., "Some Optimum Properties for Air-Cored Chokes", IEE Journal on Electrical Power Applications, Vol 1(4), 1978, pp 117-118

(60) MURGATROYD, P.N., "Electromagnetic Fields Around Power System Inductors and Possible Methods of Reducing Them", Proc. IEE Colloquium on EMI and Cardiac Pacemakers, 1985, pp 6/1-6/6

(61) SADEDIN, D.R., "Geometry of a Pulse Transformer for Electromagnetic Launching", IEEE Trans. Magnetics, Vol 20(2), 1984, pp 381-384

(62) REEVES, R., "Air-Cored Foil-Wound Inductors", Proc. IEE, Vol 125(5), 1978, pp 460-464

(63) SPANN, M.L., PRATAP, S.B., "Fabrication of a Compact Storage Inductor for Railguns", IEEE Trans. Magnetics, Vol 20(2), 1984, pp 215-218

(64) MURGATROYD, P.N. et al, "Economic Designs for Single-Layer Toroidal Inductors", Proc. IEE Part B, Vol 132(6), 1985, pp 315-318

(65) MURGATROYD, P.N. et al, "The Toroidal Cage Coil", Proc. IEE Part B, Vol 127(4), 1980, pp 207-214

(66) MURGATROYD, P.N., "Optimised Designs of Toroidal Cage Coils", Proc. IEE Part B, Vol 128(4), 1981, pp 195-200

(67) MURGATROYD, P.N., "Some Optimum Shapes for Toroidal Inductors", Proc. IEE Part B, Vol 129(3), 1982, pp 168-176

(68) BELAHRACHE, D., "Studies of Air-Cored Toroidal Inductors", M.Phil. Thesis, Loughborough University of Technology, 1987

(69) RAMO, S., WHINNERY, J.R. et al, "Fields and Waves in Communications Electronics", Wiley 1965

(70) PARTON, J.E., OWEN, S.J., "Applied Electromagnetics", Macmillan 1975

(71) POPOVIC, B.D., "Introductory Engineering Electromagnetics", Addison-Wesley 1971

(72) EVANS, P.D., AUGLA, K.H., "High-Frequency Losses in Multiturn Foil-Wound Air-Cored Inductors", Proc. IEE Part A, Vol 134(1), 1987, pp 31-36

(73) EL-MISSIRY, M.M., "Calculation of Current Distribution and Optimum Dimensions of Foil-Wound Air-Cored Reactors", Proc. IEE, Vol 124(11), 1977, pp 1073-1077

(74) ZOWARKA, R.C., "Physical Scale Modelling to Verify Storage Inductor Parameters", IEEE Trans. Magnetics, Vol 20(2), 1984, pp 219-222

(75) ABOLTIN'SH, E.E., "Inductance Coil With a Semiconductor", Radio Engineering and Electronic Physics (USA), Vol 26(5), 1981, pp 99-102

(76) Dr.O.Zucker, Energy Compression Research Corporation, in informal conversation with a member of staff from LUT, July 1989

(77) LENK, J.D., "Handbook of Digital Electronics", Prentice-Hall 1981

(78) WATSON, J., "Cost-Effective Electronic Construction", Macmillan 1985

(79) PHILIPS COMPONENTS LTD., "Technical Handbook Book 4: ICs Part 5: High-Speed CMOS Logic HC/HCT Family", Philips Components Ltd. 1988

(80) TUINENGA, P.W., "SPICE: A Guide To Circuit Simulation and Analysis Using PSpice", Prentice-Hall 1988

(81) MICROSIM CORP., "PSpice User's Guide", MicroSim Corp. 1989

(82) MONTEITH, D.O., SHAW, W.S., "Using PC-Based SPICE for Power System Design and Analysis", Proc. Power Conversion International, 1987, p 336-352

(83) PRIGOZY, S.,"Novel Applications of SPICE in Engineering Education", IEEE Trans. Education Vol 32(1), 1989, pp 35-38

(84) HU, C., KI, F.K., "Toward a Practical Computer Aid For Thyristor Circuit Design", Proc. Power Electronics Specialists Conf., 1980, pp 174-179

(85) AMARASINGHE, K.A., MANNING, C.D., "Computer Simulation of a High Frequency Zero-Current Switching Quasi-Resonant Flyback Converter", Int. Journal of Electronics Vol 67(1), 1989, pp 161-170

(86) ELLIS, T.M.R., "Structured Fortran", University of Sheffield Computing Services 1980

(87) LEDGARD, H.F., CHMURA, L.J., "Fortran With Style: Programming Proverbs", Hayden 1978

(88) KREITZBERG, C.B. et al, "The Elements of Fortran Style: Techniques for Effective Programming", Harcourt Brace Jovanovich Inc. 1972

(89) DAVENPORT, J.H., SIRET, Y. et al, "Computer Algebra", Academic Press 1988

(90) MANCHESTER COMPUTER CENTRE, "REDUCE Package For Algebraic Computations", CMS Note 007 July 1988

(91) SEWARD, L.R., "REDUCE User's Guide For IBM 360 and Derivative Computers: Version 3.3", RAND Corporation, Santa Monica USA, July 1987

(92) Private communication with Dr.D.Harper, Computer Algebra Support Officer, Liverpool University

APPENDIX A

ENERGY TRANSFER BETWEEN UNCOUPLED INDUCTORS

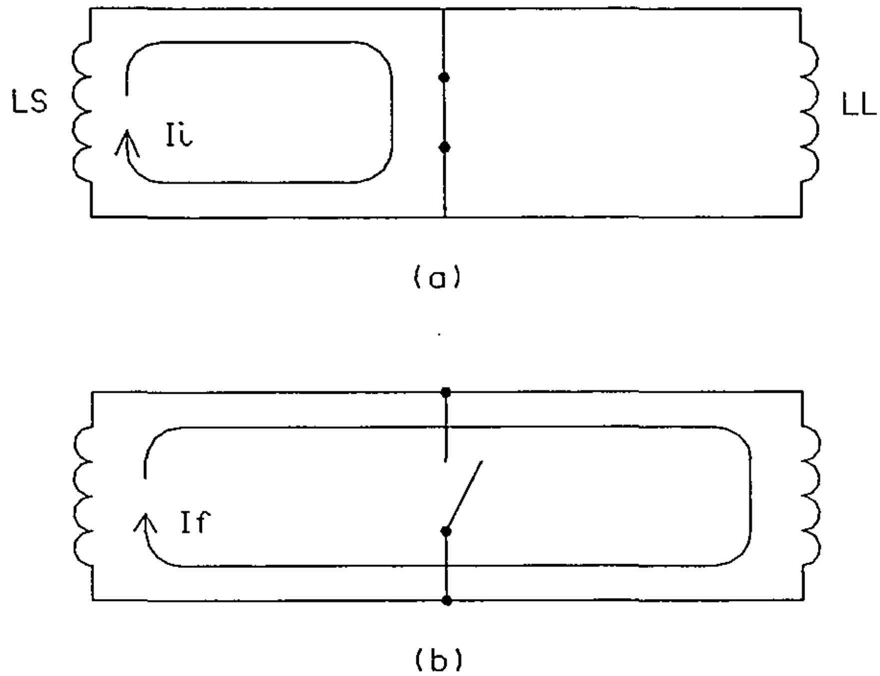


Figure A.1 Energy Transfer Between Uncoupled Inductors

By reference to figure A.1, it can be seen that energy transfer is accomplished by open-circuiting the source inductor L_S into the load inductor L_L. In this process the current in L_S falls from an initial value I_i to a final value I_f, and energy is lost in the opening switch.

The initial and final flux linkages in the two inductors are

$$(N\Phi)_i = L_S I_i$$

and

$$(N\Phi)_f = (L_S + L_L) I_f$$

respectively. Since flux linkage is conserved around a closed, loss-free loop, these two must be equal. The final current is therefore

$$I_f = I_i \frac{L_S}{L_S + L_L} \tag{A.1}$$

The initial energy in the circuit is

$$E_i = \tfrac{1}{2} L_S I_i^2 \tag{A.2}$$

and the final energy in L_L only is

$$E_L = \tfrac{1}{2} L_L I_f^2 \tag{A.3}$$

Dividing equation (A.3) by equation (A.2) and substituting for I_f from equation (A.1), it follows that the efficiency is

$$\eta = \frac{L_L L_S}{(L_L + L_S)^2} \tag{A.4}$$

By differentiating equation (A.4) with respect to L_S it is found that η_{max} occurs when the source and load inductances are equal. On substituting $L_S = L_L$ into equation (A.4), η_{max} is obtained as 25%.

Of the initial energy stored in L_s, 50% is dissipated in the switch, 25% is transferred to L_L and 25% remains in L_S.

From equations (A.1) and (A.2) it can be shown that the energy dissipated in the switch is

$$E_{dis} = \frac{L_L}{L_S + L_L} \tag{A.5}$$

This expression shows that as the source inductance increases relative to the load inductance, the energy dissipated in the switch falls. The energy transferred to L_L also falls, however, because more energy remains in L_S.

APPENDIX B

ELECTRONIC CIRCUIT DETAILS

B.1 INTRODUCTION

This Appendix describes the electronic circuits used in the meatgrinder project. It is intended to enable their operation to be understood and to facilitate future modifications. In reading this Appendix, reference should be made to Chapter 3, in which the circuits were introduced and their purposes described. The circuits were designed with the aid of references [77] to [79].

B.2 TIMING CIRCUIT

B.2.1 Introduction

The timing circuit is physically divided between two circuit boards, the two circuits being referred to as the main circuit and the delay extension circuit. Figures B.1 and B.2 are circuit diagrams of the main circuit and the delay extension circuit respectively. Table B.1 gives details of the ICs whilst figures B.3 to B.5 are "maps" of the ribbon cable connectors.

Most of the ICs are from the 74HCxx high-speed CMOS family, which offers low power consumption and better noise immunity than standard TTL [79].

PLANS TAKEN FROM HERE

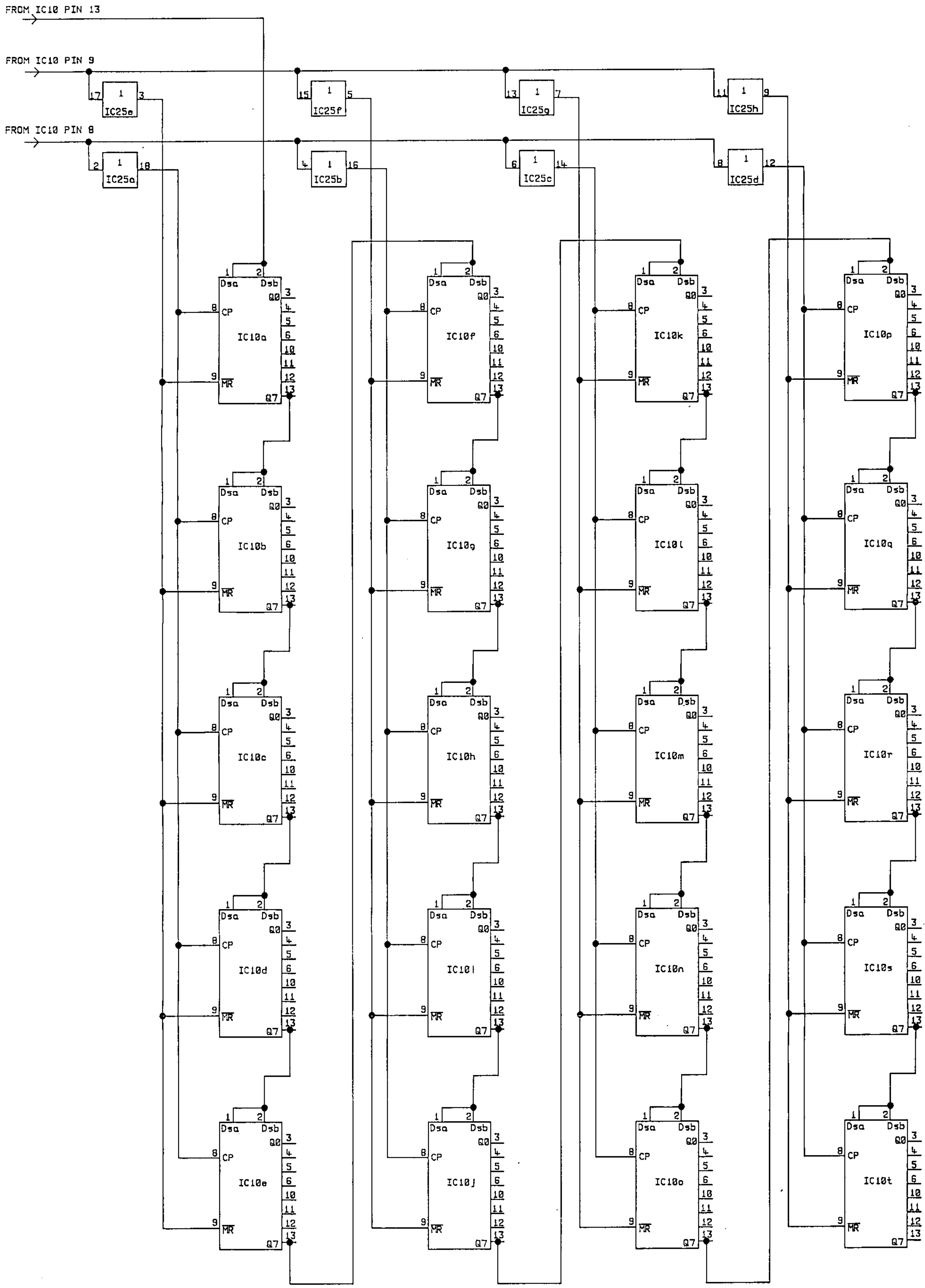


FIGURE B.2 DELAY EXTENSION CIRCUIT

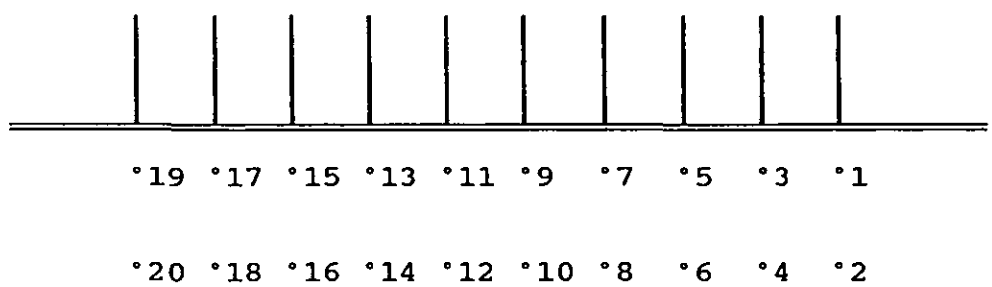
AS OF 27/07/89

The clock runs at just over 2MHz, giving a minimum time between output states of 477ns, or roughly 0.5μs. This was selected to give a good balance between flexibility during experiments and ease of circuit design. (As the clock rate increases in digital circuits the design becomes more difficult because the propagation delays of the gates become comparable with the clock period.)

Provision is made for up to seven variable delays, although experiments on the six-step meatgrinder used only five of these. Similarly, the devices TR2a, TR3a, TR4a, TR5a, TR6a, TR7a and TR8 were not used in experiments, but their drive signals are used internally by the timing circuit logic.

IC NUMBER	TYPE	DESCRIPTION
1	7805	+5V regulator
2	ICM72091PA	clock generator (Intersil)
3,11,12,13	74HC08	2-input AND gate x 4
4,5,6	74HC161	4-bit synchronous counter
7,19	74HC02	2-input NOR gate x 4
8	74HC32	2-input OR gate x 4
9,10,10a-10t	74HC164	8-bit SIPO shift register
14,15	74HC04	hex inverter
16	74HC4002	4-input NOR gate x 2
17,18	HN27C64G-15	8k x 8 150ns EPROM
20,21,25	74HC244	octal line driver
22	74HC123	dual monostable
23,24	74HC273	D-type flip-flop x 8
OP1a-7b	HCPL-2631	dual optocoupler

Table B.1 Description of Timing Circuit ICs

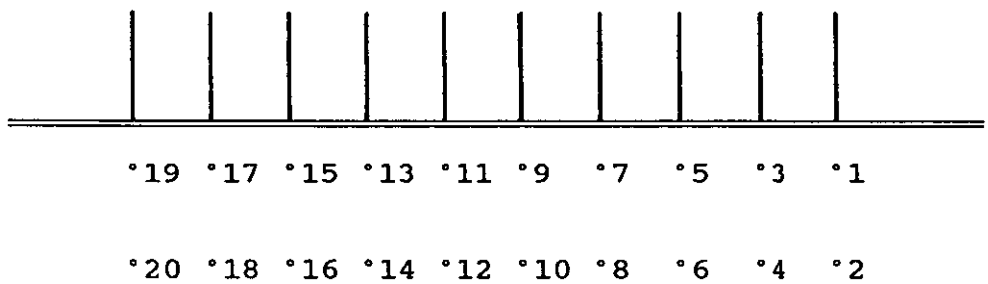


(viewed from underside of circuit board)

KEY:

1	GND	6	TR8	11	TR4	16	TR6a
2	Vcc	7	TR2	12	TR4a	17	TR7
3	spare	8	TR2a	13	TR5	18	TR7a
4	spare	9	TR3	14	TR5a	19	spare
5	TR1	10	TR3a	15	TR6	20	spare

Figure B.3 Timing Circuit Ribbon Cable Connector: Output to Interface Circuit

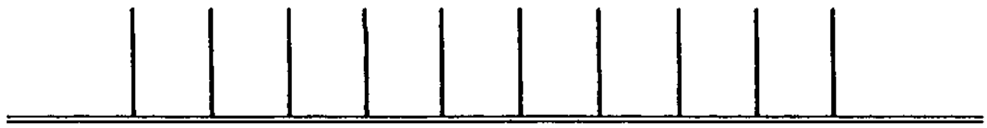


(viewed from underside of circuit board)

KEY:

1	GND	6	cascade	11	spare	16	spare
2	Vcc	7	spare	12	spare	17	spare
3	reset	8	spare	13	spare	18	spare
4	clock	9	spare	14	spare	19	spare
5	spare	10	spare	15	spare	20	spare

Figure B.4 Timing Circuit Ribbon Cable Connector: Output to Delay Extension Board



°19 °17 °15 °13 °11 °9 °7 °5 °3 °1

°20 °18 °16 °14 °12 °10 °8 °6 °4 °2

(viewed from underside of circuit board)

KEY:

1	GND	6	cascade	11	spare	16	spare
2	Vcc	7	spare	12	spare	17	spare
3	reset	8	spare	13	spare	18	spare
4	clock	9	spare	14	spare	19	spare
5	spare	10	spare	15	spare	20	spare

Figure B.5 Timing Circuit Ribbon Cable Connector: Input to Delay Extension Board

B.2.2 Stored Switching Sequence

ICs 17 and 18 are EPROMs which store the output sequence as a series of eight-bit digital words. Twelve address lines are used, giving access to 4096 memory locations.

The timing diagram used to programme the EPROMs is shown in figure B.6. Chapter 3 explains the principle behind the diagram.

The first stage is charging, which begins when TR1 turns on. This occupies the first 3542 memory locations, which corresponds to approximately 1.7ms. (The choice of charge time was arbitrary, the requirement being simply to leave plenty of memory space to programme the meatgrinder switching.)

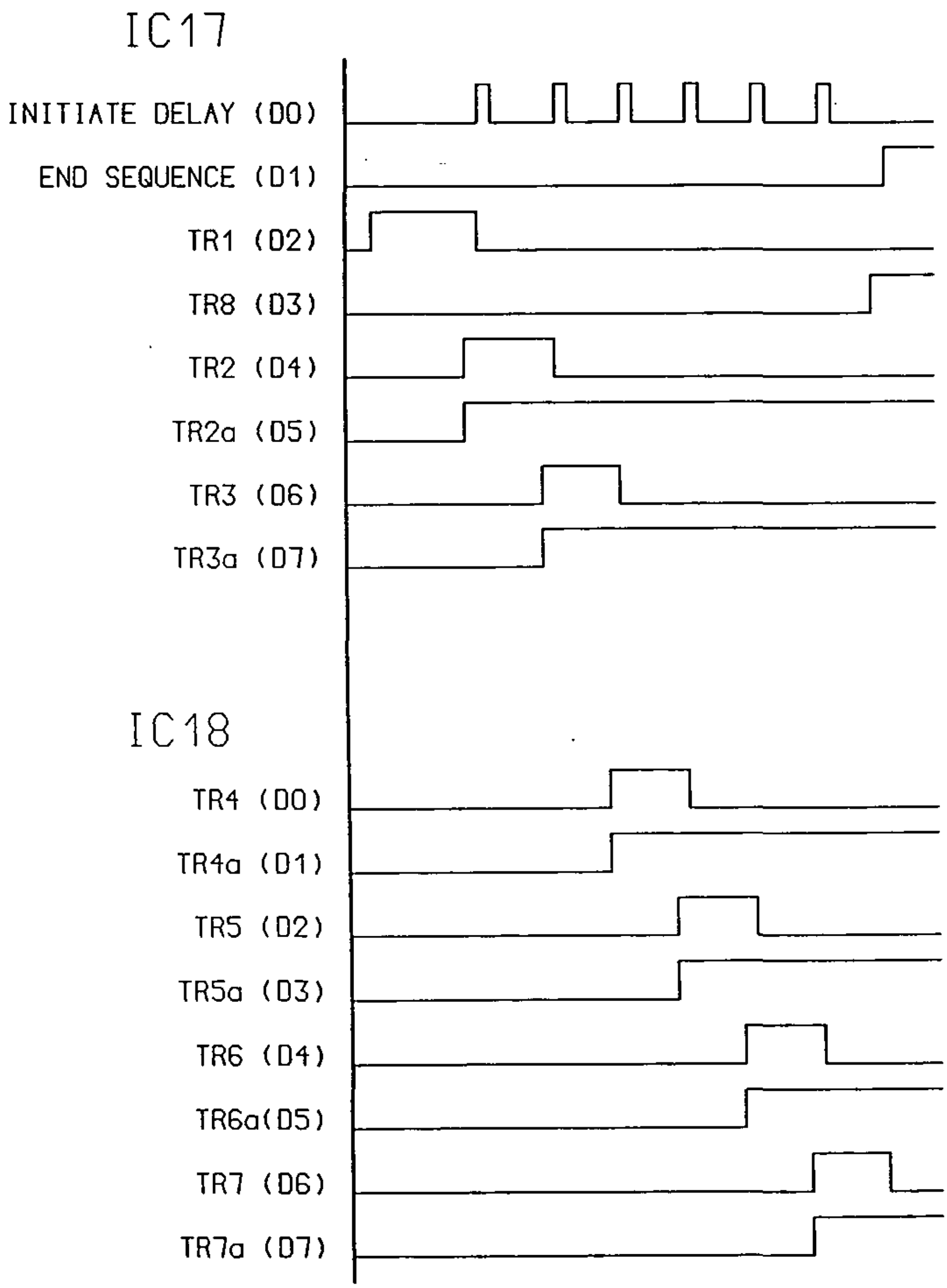


Figure B.6 Timing Diagram for EPROM Programming (Not to Scale)

To obtain an initial current of 7A the expression for exponential charging is applied:

$$I(t) = I_m(1-e^{-t/T})$$

where

$$T = L/R$$

$$L = L_{coil} \qquad (L_{coil} = 3390\mu H)$$

$$R = R_{coil}+R_{ext} \qquad (R_{coil} = 0.28\Omega)$$

and

$$I_m = \frac{V_s}{R}$$

where V_s is the supply voltage.

Thus for V_s=24V and R_{ext}=2Ω, for example, the current after 1.7ms is

$$I = \frac{24}{2.28}(1-e^{-(2.28x1700/3390)})$$

$$= 7.17A$$

The meatgrinder switching begins when TR2 turns on. The "turn-on-before-turn-off" sequence described in Chapter 3 then proceeds using devices TR3, TR4, TR5, TR6, TR7 and TR8 (note: TR8 was not used in the experiments).

The output for TR2a goes high at the same time as that for TR2, but stays high when TR2 turns off. A similar comment applies to TR3a,TR4a TR5a,TR6a and TR7a. These outputs are used internally by the delay select logic (see below) but are also available externally to drive switches if required.

The INITIATE DELAY output is pulsed high once per energy transfer to trigger a user-adjusted variable delay.

Finally, END SEQUENCE goes high at the end of the sequence to ensure that the outputs remain frozen in their final state.

The digital words to be programmed are obtained by referring to figure B.6 and reading vertically upwards for each memory location, counting a high state as 1 and a low state as 0. Thus the first word required in IC17 is 00000100 (binary) or 04 (hexadecimal). Similarly the first word in IC18 is 00000000.

Each word is repeated as many times as required: 3542 times for the first word in IC17, for example. It will be recalled that delays are required during the switching sequence to allow each energy transfer to finish before the next one starts. For each transfer a certain amount of fixed delay has been programmed in, with the remainder being obtained from the variable delay circuit. This is due to the way the circuit design evolved. The programmed-in delays could easily be removed if, for example, more charge time was required. The entire delay would then be provided by the variable delay circuit.

Tables B.2 and B.3 are extracts from typical listings of the contents of ICs 17 and 18. Re-programming to change the contents was carried out according to the requirements of the experiments.

ADDRESS	HEX DATA															
0000	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04
0010	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0DC0	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04
0DD0	04	04	04	08	38	31	30	30	30	30	30	30	30	30	30	30
0DE0	30	30	30	30	30	30	30	30	30	30	30	30	30	F0	E1	E0
0DF0	E0	E0	E0	E0	E0	E0	E0	E0	E0	E0	E0	E0	E0	E0	E0	E0
0E00	E0	E0	E0	A1	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0
0E10	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0EA0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0

Table B.2 Extracts From Typical Contents Listing: IC17

ADDRESS	HEX DATA															
0000	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0DF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0E00	00	00	03	03	03	03	03	03	03	03	03	03	03	03	03	03
0E10	03	03	03	0F	0E	0E	0E	0E	0E	0E	0E	0E	3E	3A	3A	3A
0E20	3A	3A	3A	3A	3A	FA	EA	EA	AA	AA	AA	AA	AA	AA	AA	AA
0E30	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0EA0	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA

Table B.3 Extracts From Typical Contents Listing: IC18

B.2.3 Description of Operation

B.2.3.1 Initialisation and Start-Up

Power is first applied via SW1, which illuminates LED1. C3 and R1 provide the POWER UP signal, a 5V spike with a time constant of 47ms. The spike occurs whenever C3 charges up: either when power is first applied, or if the "Reset" pushbutton PB2 is momentarily pressed, thereby discharging C3 via R22. D2 limits any negative spike to approximately 0.4V to protect the ICs.

POWER UP (or its inverted equivalent) has the following functions:

(a) Reset the shift register (ICs 9,10 and 10a to 10t), the counter (ICs 4,5 and 6), the monostable (IC22a) and the EPROM output latches (ICs 23 and 24) to zero. This is done by the inverted version of POWER UP.
(b) Reset flip-flop B (FFB) via IC8d to disable the clock.
(c) Reset flip-flop A (FFA) via IC8a to direct clock pulses to the counter (ICs 4,5 and 6) when the clock is enabled.

Momentarily pressing the "Start" pushbutton PB1 sets FFB, thus enabling the clock. Clock signals are gated to the counter via IC3a. The counter consists of three four-bit synchronous counters in cascade, these being used in preference to a ripple counter to prevent any timing problems. As the first count is zero, the highest count is 4095 (twelve bits = 4096 states).

B.2.3.2 Output

Each clock pulse received by the counter increments the output. This in turn addresses the next memory location in the two EPROMs. During the actual change of address the state of the EPROM outputs cannot be predicted because the address lines all change state at slightly different speeds. The result is that the outputs can momentarily assume an arbitrary state totally unrelated to the desired output.

To solve this problem the outputs are latched by ICs 23 and 24. These are D-type latches which are clocked with the same clock edge as the counters. Thus on each clock edge the data "clocked in" and read to the latch outputs is actually that from the previous address (see figure B.7, in which the duration of the indeterminate state is exaggerated for clarity).

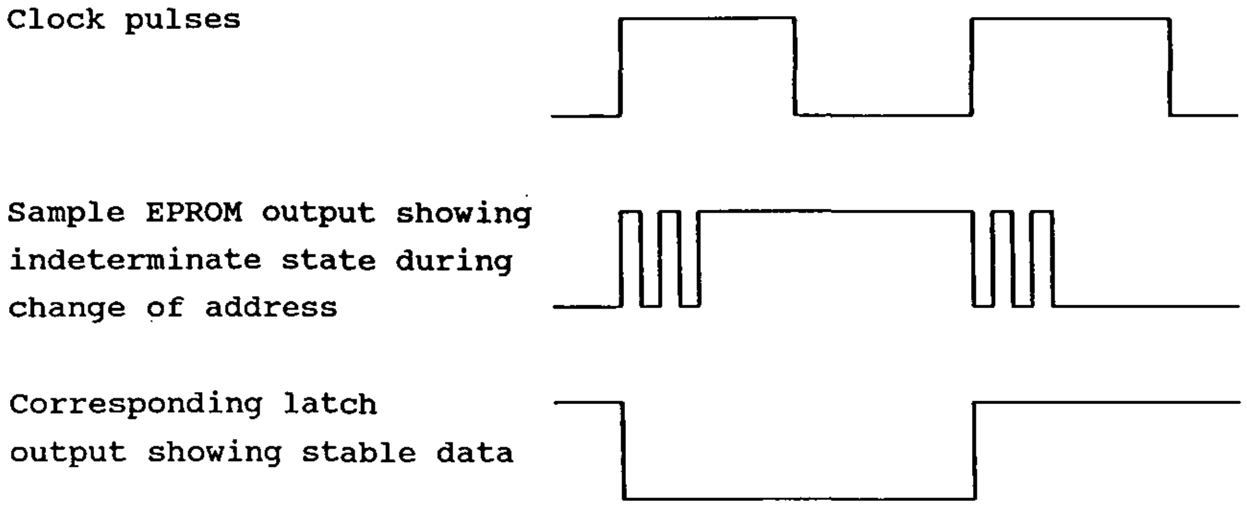


Figure B.7 Illustration of EPROM Output Latch Operation

Non-inverting buffers ICs 20a to 21f provide the current for high-speed optocouplers OP1 to OP7. The optocouplers protect the circuit from noise which may be fed back from power circuitry. Should it be required, they also provide electrical isolation to enable switches to be driven with respect to different ground references.

B.2.3.3 Variable Delays

(i) INITIATE DELAY Signal

As described in Chapter 3, a variable delay is required each time a switch opens to initiate an energy transfer. The

INITIATE DELAY output from IC17 triggers a monostable (IC22a) which in turn applies a "set" pulse to FFA. This has the effect of diverting clock pulses from the counter to the shift register.

Note that INITIATE DELAY has to be converted to a pulse, otherwise the "reset" pulse to FFA at the end of the delay would simply turn off the clock pulses to the shift register without re-directing them to the counter.

(ii) Delay Circuit

The delay circuit (shift register) is made up of twenty-two eight-bit serial-in-parallel-out shift registers in cascade. The devices on the delay extension board (ICs 10a to 10t) are buffered by ICs 25a to 25h in order to stay within fan-out limitations.

At the start of a delay all outputs are low, a reset pulse having been applied from IC13d either at power-up or at the end of a previous delay. Each clock pulse received causes the data at the serial input (IC9 pins 1 and 2) to be read in and read out to the least significant parallel output (IC9 pin 3). The data previously on pin 3 is shifted to pin 4, that previously on pin 4 to pin 5 and so on. Since there is a permanent high on the serial input, the effect is to cause each output to go high in turn.

The delays are set by linking each delay line to a shift register output. If, for example, DELAY 2 (IC12b pin 5) is linked to IC10 pin 6, the second delay will end when enough clock pulses have been received to cause IC10 pin 6 to go high. (Detection of when a particular delay is required is achieved by the delay select logic - see below.)

(ii)(a) Available Range of Delay

If the enabled delay line is connected to IC9 pin 3, then as soon as the first clock pulse is received by the shift register, the delay select logic generates END OF DELAY, which in turn puts a high on IC7b pin 5. It is likely that the INITIATE DELAY pulse from IC22b will still be present and therefore that there will also be a high on IC7a pin 3. This leads to a "0-0" output from FFA.

The next state depends on which of the inputs to FFA first goes low, and this is purely a function of the propagation delays in the system. If END OF DELAY goes low first (as it will once the shift register has been reset) the effect will be to direct clock pulses to the shift register once more. The other possibility is that the clock is correctly re-routed to the counter, which will receive two clock edges separated only by propagation delays. It is therefore best to avoid using this shift register output.

Using IC9 pin 4 gives zero delay, with each subsequent shift register output adding one clock cycle to the delay. Thus the maximum delay is 174 cycles or 83μs.

(iii) Delay Select Logic

The delay select logic serves two purposes. It firstly detects which delay is required according to the point in the switching sequence. It then detects when that particular delay is complete.

From figure B.6 it can be seen that when the first delay is required, DRIVE TR2a is high whereas DRIVE TR3a is low. Similarly, when the second delay is required, DRIVE TR3a is high and DRIVE TR4a low. This pattern is repeated for the remaining delays and is the basis of delay selection.

As an example consider the first delay. When this is required, both inputs to IC3c are high and therefore the output SELECT DELAY 1 is high. Examination of figure B.6 shows that all other SELECT DELAY signals are low. Thus of the AND gates ICs 12a to 13c, only IC12a is enabled. This means that only DELAY 1 has any effect on the output from the delay select logic.

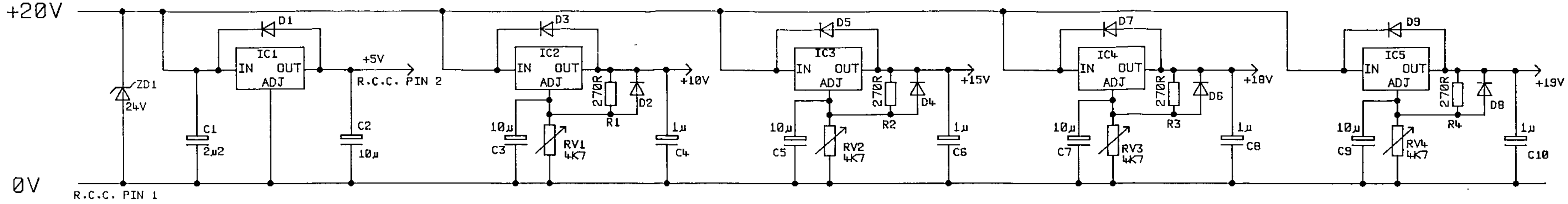
ICs 16a,16b,15e,15f and 19c form an OR gate. When the selected delay line goes high the output of the OR gate goes high, thus causing the shift register to be reset and the clock to be re-routed to the counter. (The length of the END OF DELAY pulse depends only on propagation delays; this is a design weakness, but no problems have resulted from it.)

B.2.3.4 End of Sequence

When the END SEQ signal goes high FFB is reset via IC8d, thus disabling the clock. In addition LED2 is illuminated via IC14b and TR9. The EPROM latch outputs finish with the data in address 4095 whilst the counter returns to zero.

FIGURE B.8 INTERFACE CIRCUIT

N.B. AT SOME POINT 0V MUST BE CONNECTED TO POWER GROUND
R.C.C. = RIBBON CABLE CONNECTOR



COMPONENT DETAILS

IC1		7805	+5V REGULATOR
IC2-IC5		317L	ADJUSTABLE 100mA REGULATOR
IC6-IC19	Teledyne	TSC429CPA	MOSFET DRIVER
D1-D9		1N914	
D10-D14		BAT85	SCHOTTKY DIODE

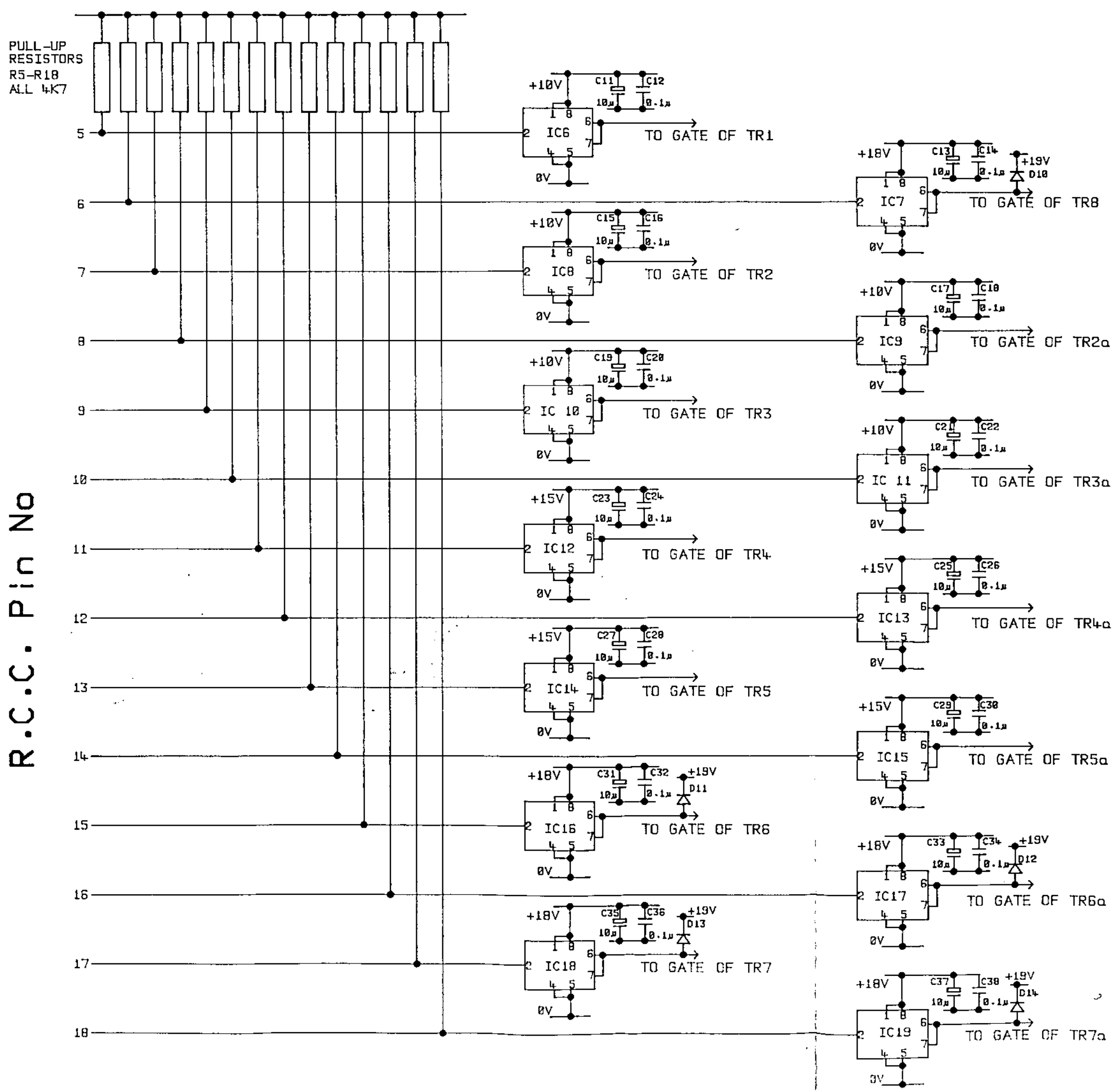
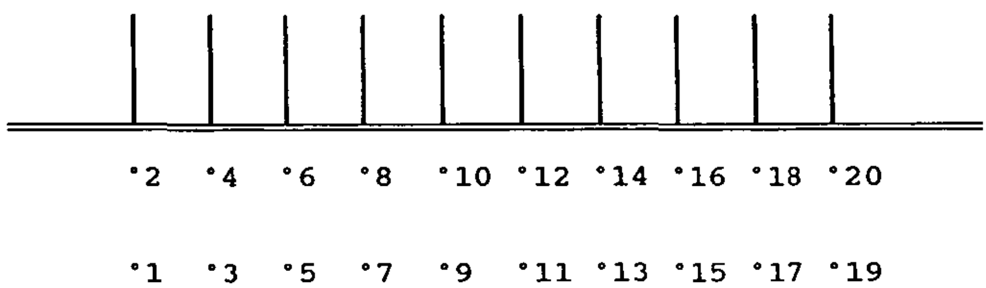


FIGURE B.8 INTERFACE CIRCUIT

AS AT 3/7/89

B.3 INTERFACE CIRCUIT

Figure B.8 is a circuit diagram of the interface circuit and figures B.9 and B.10 are "maps" of the ribbon cable connectors.

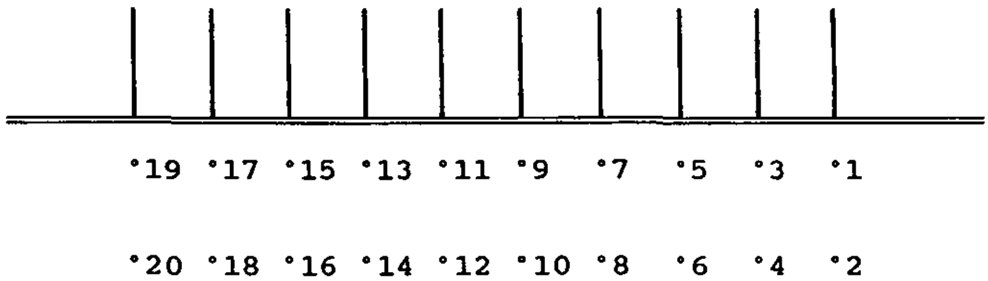


(viewed from underside of circuit board)

KEY:

1	GND	6	TR8	11	TR4	16	TR6a
2	Vcc	7	TR2	12	TR4a	17	TR7
3	spare	8	TR2a	13	TR5	18	TR7a
4	spare	9	TR3	14	TR5a	19	spare
5	TR1	10	TR3a	15	TR6	20	spare

Figure B.9 Interface Circuit Ribbon Cable Connector: Input From Timing Circuit



(viewed from underside of circuit board)

KEY:

1	GND	6	TR2a	11	TR5	16	TR7a
2	spare	7	TR3	12	TR5a	17	+19V
3	TR1	8	TR3a	13	TR6	18	spare
4	TR8	9	TR4	14	TR6a	19	spare
5	TR2	10	TR4a	15	TR7	20	spare

Figure B.10 Interface Circuit Ribbon Cable Connector: Output to Power Circuit

The incoming 20V d.c. supply is used to provide the following non-isolated regulated d.c. supplies using individual regulators:

(1) A 5V supply for the optocouplers in the timing circuit. The optocoupler outputs are open collector, and so pull-up resistors R5-R18 are provided to enable the signals to swing between 0V and 5V.

(2) 10V, 15V and 18V supplies for the MOSFET drivers ICs 6 to 19. Each driver is connected to the appropriate supply according to the gate drive voltage required.

(3) A 19V supply used for MOSFET gate-source protection where the nominal gate drive voltage is 18V. Diodes such as D10 are used to connect the driver output to the 19V supply, thus preventing the output voltage exceeding 19V plus the forward bias voltage of the diode. D10-D14 are Schottky diodes with a low forward bias voltage of about 0.3V.

For the other drive voltages, gate-source protection is provided by ordinary zener diodes. However, the tolerance on an 18V zener diode is too great to guarantee limiting the gate-source voltage to 20V.

The MOSFET drivers have an inverting action, but this is cancelled out by the inverting action of the open collector outputs of the timing circuit optocouplers. Thus as the inputs to the optocouplers swing between 0V and 5V the outputs of the drivers swing between 0V and the appropriate gate drive voltage. The drivers can produce very rapid MOSFET switching as they are capable of sourcing or sinking up to 6A.

Note: The incoming supply must be at least 20.5V to ensure proper regulation of the 18V and 19V supplies. Raising the input voltage further should be avoided because it increases the dissipation in the other regulators, particularly the 5V one. Obviously this could be dealt with by mounting the regulators on heatsinks but in the longer term it would be preferable to find a more elegant circuit design.

APPENDIX C

COMPUTER SIMULATION USING PSPICE

C.1 INTRODUCTION

PSpice [80] is a simulation package which may be used to test the response of a circuit without working on the circuit hardware itself. Several different types of analysis may be carried out, including a.c. response, d.c. response, transient response and noise analysis.

The package is derived from the SPICE family of mainframe programs. SPICE is an acronym for Simulation Program with Integrated Circuit Emphasis, but PSpice is equally suitable for simulation of power circuits [81,82]. Version 4.0 also includes simulation of digital circuits.

Reference [83] provides other examples of the use of PSpice.

The simulation work described in this thesis was carried out using version 3.3 on an IBM-compatible personal computer; versions are also available for use on other types of computer [81].

In addition to the references quoted, new users will benefit from viewing the disk-based "slide show" included with version 4.0. This provides a comprehensive introduction to all aspects of the package.

C.2 METHOD OF USE

This section contains only an overview of how PSpice is used. See the quoted references for further details.

C.2.1 Input

The circuit is described by first allocating a number to each node, the ground reference always being node zero. The input file is then created using a text editor, and consists of lines describing the components and indicating the nodes to which they are connected. Further lines are required to specify the analyses required, the output required and various other options.

C.2.1.1 Models

Components such as transistors and diodes must clearly have their characteristics fully described if their behaviour is to be simulated. The collection of numerical values which serves this purpose is known as a component model. A diode model, for example, must include values for junction capacitance, saturation current and reverse breakdown voltage.

The equivalent circuits and associated parameters used in PSpice are described in reference [81]. The default parameter values can be used to simulate the response of a typical device (a typical diode, for example), but the resulting model will not relate to any specific device (a 1N4002 diode, for example). More often the user will either specify one of the pre-defined models in the PSpice component library (such as the IRF250 MOSFET) or custom-build a model using the "Parts" program in conjunction with data sheet values. This latter method was used to model the Motorola MR752 diode.

Devices not covered by any of the above methods can be modelled by means of a user-defined subcircuit. This technique has been used successfully to model SCRs [84] and GTOs [85].

C.2.2 Output

Results are available in numerical and graphical form. Graphical output is produced by the "Probe" program which acts as the equivalent of an oscilloscope. Examples of this type of output appear in several Chapters in this thesis.

C.3 USES IN RESEARCH

Computer simulation has been used in this research to:

(a) simulate circuits which have already been built and tested. This may be done either to confirm the suitability of the simulation program for a particular purpose or to validate the input file used to represent the circuit.

(b) vary circuit parameters to test their significance. This could be used, for example, to check whether a stray capacitance is affecting the output or to make a coil more nearly ideal by reducing its resistance to a very low value.

(c) assess the effect of changing to a different device (a different MOSFET, for example) without the need to do it physically.

(d) check the validity of theoretical analysis.

(e) avoid the need for theoretical analysis. In some cases it is quicker to find a result by trial and error than to calculate it.

(f) measure quantities such as clamp device currents, which are physically unaccessible in the real circuit.

C.4 RUN TIME

As described in Chapter 1 the operation of the meatgrinder consists of a relatively long charge time followed by a much shorter time of switching and energy transfer. In order to avoid losing detail in the latter part of the process (which is the part of most interest) it is necessary to impose an upper limit on the internal time step used by PSpice. This is because the built-in time step control can cause aliasing [81] when a circuit runs freely, i.e. without a driving voltage or current source. The meatgrinder forms such a circuit once the power supply is disconnected. This means that during charging, whilst PSpice would normally allow the time step to rise to 36μs, for example, it may be restricted to only 100ns. Consequently, out of a run time of an hour it may only be the final five or ten minutes which provide information of interest.

The simplest way to alleviate this problem is to use a faster computer. It has been found, for example, that the same simulation runs two-and-a-half times faster on a PC running an 80386 processor at 16MHz than on a similar machine running an

80286 processor at 8MHz. Clock speeds of up to 33MHz are now becoming available.

A much more satisfactory solution, however, would be to find a way of running the first part of the simulation without the time step limit. Variation of the time step limit during a run is not available in PSpice at present. It would therefore be necessary to run the two parts of the simulation separately and to have a means of using the results of the first part to establish initial conditions for the second part.

C.4.1 Use of Initial Condition Facility

When using PSpice it is possible to specify the initial current in an inductor. This is done by using the "IC=" statement in the inductor specification and the "UIC" (Use Initial Conditions) qualifier in the .TRAN statement.

It has been found that in certain cases this facility can be used to avoid the need for simulation of the charging phase. It should be noted, however, that this has only been successful in specific simple cases. This means, for example, that there must be no capacitors across the coils and that diodes must be ideal (that is, the built-in default diode model must be used). Obviously this limits the usefulness of the technique. Furthermore, it has been found that a flaw in PSpice necessitates an adjustment of the value used in the "IC=" statement. The next sections describe this problem and its solution.

C.4.1.1 Problem Description

The adjustment is only required when dealing with series-connected inductors which are mutually coupled. To demonstrate this, consider the two following PSpice descriptions for a series L-R circuit:

(Note: These circuit descriptions are in the form required by PSpice. Numbers such as "1 0", "1 2" and so on are node numbers. Also, "u" must be used in place of "μ".)

Circuit 1

```
L1   1   0   100uH   IC=10A

R    1   0   10
```

Circuit 2

```
L1   1   2   50uH   IC=10A
L2   2   0   50uH   IC=10A

R    1   0   10
```

Transient analysis yields the same result in both cases; an initial current of 10A decaying exponentially.

If now the circuit is changed to:

Circuit 3

```
L1    1    2    25uH   IC=10A
L2    2    0    25uH   IC=10A

k12   L1   L2   0.9999

R     1    0    10
```

this gives the same total inductance (L1+L2+2M12) but an initial current of only 5A.

As the coupling coefficient is changed the initial current produced varies as shown in table C.1.

COUPLING COEFFICIENT	INITIAL CURRENT (A)
0.1	9.091
0.3	7.692
0.5	6.667
0.7	5.882
0.9	5.263

Table C.1 Variation of Initial Current For Circuit 3

C.4.1.2 Solution

It appears that PSpice calculates the initial total flux linkage from the inductor specifications alone, regardless of whether or not there is any coupling. If coupling exists, PSpice subsequently adds the mutual inductance and adjusts the initial current to give the same total flux linkage.

Example:-

```
L1   1   2   25uH  IC=10A
L2   2   0   25uH  IC=10A

k12  L1  L2  0.9999
```

(a) Total flux linkage = 2 x 25E-6 x 10 = 500E-6
(b) Total inductance = 25 + 25 + (2 x 25) = 100μH
(c) Hence initial current = 500 / 100 = 5A

Thus the desired current value can be obtained by calculating the value to use in the "IC=" statement, I(ic), from:

$$I(ic) = I(desired) \times \frac{\text{total inductance}}{\text{total self inductance}} \quad (C.1)$$

This holds for any number of series sections.

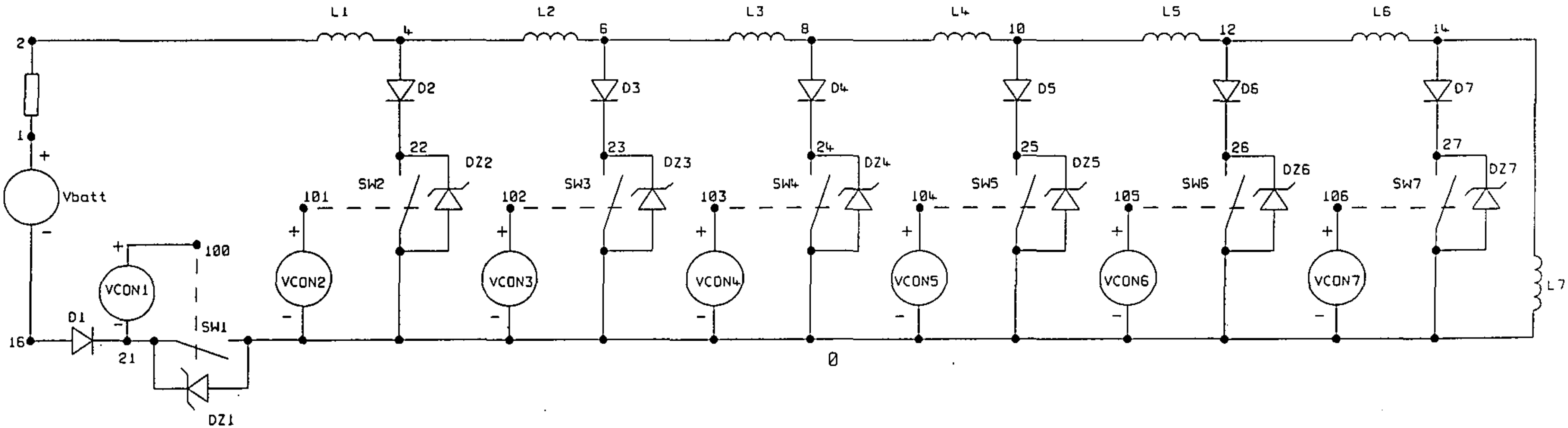


Figure C.1 PSpice Circuit For Idealised Six-Step Meatgrinder

C.5 EXAMPLE INPUT FILES

C.5.1 Idealised Six-Step Meatgrinder Using Initial Condition Technique

(The circuit diagram from which this input file is derived is shown in figure C.1.)

```
IDEAL MEATGRINDER: NO CHARGE (IC=-25.096A)

*coil
L1  2   4    403.0uH   IC=-25.096A
L2  4   6    289.0uH   IC=-25.096A
L3  6   8    147.0uH   IC=-25.096A
L4  8   10    52.0uH   IC=-25.096A
L5  10  12    24.0uH   IC=-25.096A
L6  12  14     0.9uH   IC=-25.096A
L7  14  0     11.5uH   IC=-25.096A

K12  L1 L2  0.85
K13  L1 L3  0.69
K14  L1 L4  0.58
K15  L1 L5  0.55
K16  L1 L6  0.29
K17  L1 L7  0.2
K23  L2 L3  0.84
K24  L2 L4  0.71
K25  L2 L5  0.68
K26  L2 L6  0.5
K27  L2 L7  0.44
K34  L3 L4  0.89
K35  L3 L5  0.8
```

```
K36   L3 L6   0.75
K37   L3 L7   0.7
K45   L4 L5   0.91
K46   L4 L6   0.74
K47   L4 L7   0.65
K56   L5 L6   0.76
K57   L5 L7   0.7
K67   L6 L7   0.71

*supply
Vbatt   16 1   PULSE (0 24V 0 0 0 1 2)
Rext    1  2   2.25  ;use to set initial current

*blocking diodes
D1   16 21   IDEAL
D4   8  24   IDEAL
D5   10 25   IDEAL
D6   12 26   IDEAL
D7   14 27   IDEAL

*clamps (represented by zeners)
DZ1   0 21   CLAMP1
DZ2   0 22   CLAMP2
DZ3   0 23   CLAMP2
DZ4   0 24   CLAMP2
DZ5   0 25   CLAMP2
DZ6   0 26   CLAMP2
DZ7   0 27   CLAMP2

*switches
S1   21 0   100 0   SW
S2   22 0   101 0   SW
S3   23 0   102 0   SW
S4   24 0   103 0   SW
```

```
S5   25 0   104 0   SW
S6   26 0   105 0   SW
S7   27 0   106 0   SW

VCON1   100 0   PULSE (10V 0      1.0us   0ns 0ns    1s      2s)
*delay to see I1
VCON2   101 0   PULSE (0 10V      1.0us   0ns 0ns   10.0us   1s)
*off at 11
VCON3   102 0   PULSE (0 10V     11.0us   0ns 0ns   20.0us   1s)
*off at 31
VCON4   103 0   PULSE (0 10V     31.0us   0ns 0ns   20.0us   1s)
*off at 51
VCON5   104 0   PULSE (0 10V     51.0us   0ns 0ns   20.0us   1s)
*off at 71
VCON6   105 0   PULSE (0 10V     71.0us   0ns 0ns   30.0us   1s)
*off at 101
VCON7   106 0   PULSE (0 10V    101.0us   0ns 0ns    0.5s    1s)
*stays on

*models
.MODEL  IDEAL     D     (VJ=1E-6)
*Trr defaults to 0 and Bv to infinity
.MODEL  CLAMP1    D     (BV=500V)
.MODEL  CLAMP2    D     (BV=150V)
.MODEL  SW      VSWITCH (Ron=1E-6 Roff=1E6 Von=10V Voff=0)
.OPTIONS   LIMPTS=20000 ITL4=20 ITL5=0 RELTOL=0.01 ABSTOL=1E-3
+VNTOL=1E-3
.TRAN 10ns 120us 0.0us  50ns UIC
.PROBE  I(L7) I(L6) I(L5) I(L4) I(L3) I(L2) I(L1)
+       I(D1) I(D2) I(D3) I(D4) I(D5) I(D6) I(D7)
+       V(2)  V(4)  V(6)  V(8)  V(10) V(12) V(14)
+       V(21) V(22) V(23) V(24) V(25) V(26) V(27)
+       V(100)
.END
```

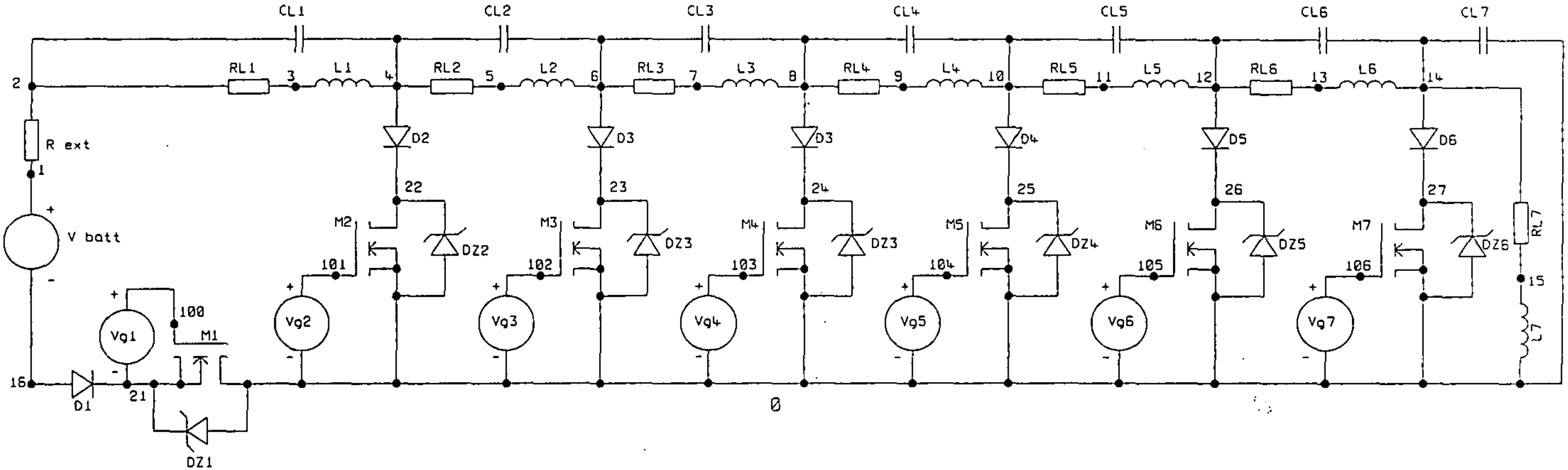


Figure C.2 PSpice Circuit For Six-Step Meatgrinder With MOSFET and Diode Models

C.5.2 Six-Step Meatgrinder With MOSFET and Diode Models

(The circuit diagram from which this input file is derived is shown in figure C.2.)

```
MEATGRINDER WITH MOSFETs AND ARBITRARY CAPACITANCES

*coil
L1   3   4    403.0uH
L2   5   6    289.0uH
L3   7   8    147.0uH
L4   9   10    52.0uH
L5   11  12    24.0uH
L6   13  14     0.9uH
L7   15  0     11.5uH

K12   L1  L2   0.85
K13   L1  L3   0.69
K14   L1  L4   0.58
K15   L1  L5   0.55
K16   L1  L6   0.29
K17   L1  L7   0.2
K23   L2  L3   0.84
K24   L2  L4   0.71
K25   L2  L5   0.68
K26   L2  L6   0.5
K27   L2  L7   0.44
K34   L3  L4   0.89
K35   L3  L5   0.8
K36   L3  L6   0.75
K37   L3  L7   0.7
K45   L4  L5   0.91
K46   L4  L6   0.74
```

```
K47   L4 L7   0.65
K56   L5 L6   0.76
K57   L5 L7   0.7
K67   L6 L7   0.71

RL1   2   3    0.071
RL2   4   5    0.054
RL3   6   7    0.037
RL4   8   9    0.021
RL5   10  11   0.013
RL6   12  13   0.004
RL7   14  15   0.011

*coil capacitances
CL1   2   4    50pF
CL2   4   6    50pF
CL3   6   8    40pF
CL4   8   10   30pF
CL5   10  12   20pF
CL6   12  14   2pF
CL7   14  16   10pF

*supply
Vbatt   16  1   PULSE (0 24V 0 0 0 1 2)
Rext    1   2   1.95 ;use to set initial current

*blocking diodes
D1   16  21   MR752
D2   4   22   MR752
D3   6   23   MR752
D4   8   24   MR752
D5   10  25   MR752
D6   12  26   MR752
D7   14  27   MR752
```

```
*clamps (represented by zeners)
DZ1   0 21   CLAMP1
DZ2   0 22   CLAMP2
DZ3   0 23   CLAMP2
DZ4   0 24   CLAMP2
DZ5   0 25   CLAMP2
DZ6   0 26   CLAMP2
DZ7   0 27   CLAMP2

*transistors (gate terminals 100,101......)
M1   21 100 0 0   IRF450
M2   22 101 0 0   IRF250
M3   23 102 0 0   IRF250
M4   24 103 0 0   IRF250
M5   25 104 0 0   IRF250
M6   26 105 0 0   IRF250
M7   27 106 0 0   IRF250

Vg1  100 0  PULSE (0 10V    0      100ns 100ns 1700.5us 1s)
*Tcharge+overlap
Vg2  101 0  PULSE (0 10V 1700.0us 100ns 100ns   5.0us  1s)
*off at 1705
Vg3  102 0  PULSE (0 10V 1704.5us 100ns 100ns  10.0us  1s)
*off at 1714.5
Vg4  103 0  PULSE (0 15V 1714.0us 100ns 100ns  10.0us  1s)
*off at 1724
Vg5  104 0  PULSE (0 15V 1723.5us 100ns 100ns  10.0us  1s)
*off at 1733.5
Vg6  105 0  PULSE (0 18V 1733.0us 100ns 100ns  26.0us  1s)
*off at 1759
Vg7  106 0  PULSE (0 18V 1758.5us 100ns 100ns   0.5s   1s)
*stays on
```

```
*models
.MODEL MR752 D (Is=609.4f Rs=2.534m N=1 Xti=3 Eg=1.11 Bv=200
+               Ibv=100u Cjo=1.379n Vj=.75 M=.8 Fc=.5 Tt=6.059u)
.MODEL  CLAMP1    D     (BV=550V)
.MODEL  CLAMP2    D     (BV=150V)
.MODEL IRF450 NMOS(Level=3 Gamma=0 Delta=0 Eta=0 Theta=0 Kappa=0
+            Vmax=0 Xj=0 Tox=100n Uo=600 Phi=.6 Rs=32.16m
+            Kp=20.69u W=2.1 L=2u Vto=3.415 Rd=.2606 Rds=1.6MEG
+            Cbd=1.732n Pb=.8 Mj=.5 Fc=.5 Cgso=4.125n Cgdo=50.59p
+            Rg=4.582 Is=1E-30)
.MODEL IRF250 NMOS(Level=3 Gamma=0 Delta=0 Eta=0 Theta=0 Kappa=0
+            Vmax=0 Xj=0 Tox=100n Uo=600 Phi=.6 Rs=26.49m
+            Kp=20.56u W=1.5 L=2u Vto=3.435 Rd=32.81m Rds=640K
+            Cbd=3.095n Pb=.8 Mj=.5 Fc=.5 Cgso=4.271n Cgdo=181.4p
+            Rg=6.931 Is=44.42n)

.OPTIONS   LIMPTS=10000 ITL5=0 RELTOL=0.01 ABSTOL=1E-3
+          VNTOL=1E-3

.TRAN 20ns 1810us 1700us
.PROBE  I(L7) I(L6) I(L5) I(L4) I(L3) I(L2) I(L1)
+       I(D1) I(D2) I(D3) I(D4) I(D5) I(D6) I(D7)
+       V(2)  V(4)  V(6)  V(8)  V(10) V(12) V(14) V(16)
+       V(21) V(22) V(23) V(24) V(25) V(26) V(27)
.END
```

C.5.3 Two-Step Meatgrinder Used for Investigation of ETAC

(The circuit diagram from which this input file is derived is shown in figure 5.3.)

3 SECTIONS; V_{S3}=100V

```
L1   3   4   100uH   IC=-14.9A
L2   4   0   150uH   IC=-14.9A
L3   6   3   200uH

K12  L1  L2  0.5
K13  L1  L3  0.6
K23  L2  L3  0.4

*clamps
Vsw     1   0   PULSE (0 100V 1us 0 0 1 2)
R1      2   1   1E-9 ; just to avoid voltage loop
VS3     7   8   PULSE (0 100V 1us 0 0 1 2)

*diodes
D1   3   2   DIODE
D2   4   5   DIODE
D3   6   7   DIODE

*switches
S1   5   0   100   0   SW
S2   8   0   101   0   SW

VCON1  100   0   PULSE(0 10V  1us 0 0 1 2)
VCON2  101   0   PULSE(0 10V  1us 0 0 1 2);brings VS3 into circuit
```

```
*models
.MODEL DIODE    D
.MODEL  SW   VSWITCH (Ron=1E-6 Roff=1E6 Von=10V Voff=0)

.TRAN 10ns 15us 0  50ns   UIC
.PROBE
.END
```

C.5.4 Two-Step Meatgrinder Used for Investigation of ITAC

(The circuit diagram from which this input file is derived is shown in figure 5.12.)

```
3 SECTIONS;Bv(DZ4)=50V

L1   1   2   200uH   IC=-19.57A
L2   2   3   100uH   IC=-19.57A
L3   3   0   150uH   IC=-19.57A

K12   L1   L2   0.6
K13   L1   L3   0.4
K23   L2   L3   0.5

*diodes
D1   1   4   DIODE
D2   2   5   DIODE
D3   3   6   DIODE
```

```
*zener clamps
DZ1   0   4   CLAMP1
DZ2   0   5   CLAMP2
DZ3   0   6   CLAMP3
DZ4   3   7   CLAMP4

*switches
S1   4   0   100   0   SW
S2   5   0   101   0   SW
S3   6   0   102   0   SW
S4   7   6   103   0   SW

VCON1   100   0   PULSE(10V   0    2us   0 0 1      2)
VCON2   101   0   PULSE(0    10V   2us   0 0 1      2)
VCON3   102   0   PULSE( 0   10V   1     0 0 1      2)
VCON4   103   0   PULSE( 0   10V   2us   0 0 1      2)

*models
.MODEL DIODE      D
.MODEL CLAMP1     D (BV=1000V)
.MODEL CLAMP2     D (BV=500V)
.MODEL CLAMP3     D (BV=100V)
.MODEL CLAMP4     D (BV=50V)
.MODEL  SW   VSWITCH (Ron=1E-6 Roff=1E6 Von=10V Voff=0)

.OPTIONS RELTOL=0.01  ABSTOL=1E-3 VNTOL=1E-3 ITL5=0 ITL4=20
.TRAN 5ns 8us 1.5us  25ns   UIC
.PROBE
.END
```

APPENDIX D

EFFECT OF UNCOUPLED LOAD ON COUPLING COEFFICIENT

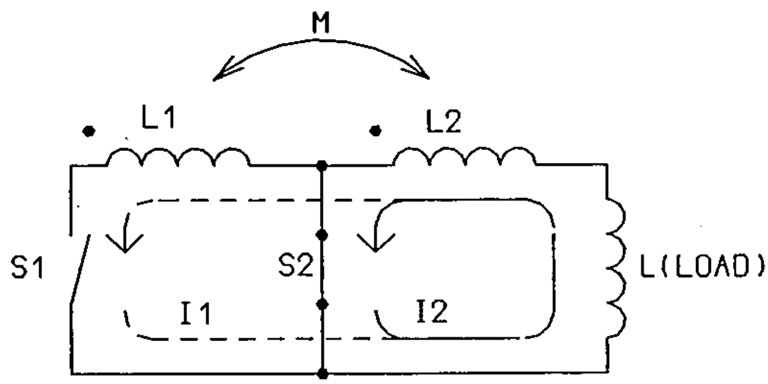


Figure D.1 Single-Step Meatgrinder With Uncoupled Load

Figure D.1 shows a single-step meatgrinder with an uncoupled load in series with the second coil section L_2. Although the addition of such a load does not affect the operation of the circuit, the final inductance becomes

$$L_2' = L_2 + L_{LOAD} \tag{D.1}$$

and the coupling coefficient must be modified as follows:

The total circuit inductance L_0 is

$$L_0 = L_1 + L_2' + 2M_{12}' \tag{D.2}$$

Since the load is magnetically uncoupled, it has no effect on the mutual inductance in the circuit. Hence

$$M_{12}' = M_{12}$$

or

$$k'\sqrt{L_1 L_2'} = k\sqrt{L_1 L_2}$$

and therefore

$$k' = k\sqrt{L_2/L_2'}$$

from which

$$k' = k \left[\frac{L_2}{L_2 + L_{LOAD}} \right]^{\frac{1}{2}} \qquad \text{(D.3)}$$

The modified coupling coefficient k' may now be used in expressions developed for the unloaded circuit, with L_2 being replaced by $L_2 + L_{LOAD}$.

APPENDIX E

FORTRAN PROGRAMS USED IN THE RESEARCH

E.1 IDEAL MEATGRINDER CALCULATIONS

The program "Ideal" was written with the intention of providing a general-purpose tool for investigation of the meatgrinder concept. The calculations for multi-step circuits did prove useful, but the program was replaced by use of the PSpice simulation package (see Appendix C). PSpice is ideal for simulating the meatgrinder circuit, and is able to cope with phenomena such as transformer action clamping, which "Ideal" does not include.

The other function of "Ideal" is to evaluate certain algebraic expressions, rather like a sophisticated programmable calculator. The expressions relate to the theoretical investigation of transformer action clamping described in Chapter 5.

The following pages contain a listing of the program followed by some example outputs.

```
c     ========================================
c     PROGRAM TITLE: IDEAL MEATGRINDER CALCULATIONS
c     WRITTEN BY: M.G.PIMPERTON
c     DATE WRITTEN: MARCH 1988
c     WRITTEN FOR: DEPT. OF ELECTRONIC AND ELECTRICAL ENGINEERING.
c                  LOUGHBOROUGH UNIVERSITY OF TECHNOLOGY
c
c     PROGRAM INTENT:
c
c     THE PROGRAM IS INTENDED FOR USE IN INVESTIGATIONS OF
c     THE MEATGRINDER CONCEPT.
c
c     THE ALGORITHMS ARE BASED ON ANALYSIS OF THE IDEAL CIRCUIT: NO
c     RESISTANCE. NO CAPACITANCE. IDEAL SWITCHES.  FURTHERMORE. TWO
c     SIMPLIFYING ASSUMPTIONS ARE MADE:
c
c             (a) THE SWITCHES HAVE IDEAL VOLTAGE CLAMPS ACROSS THEM
c                 SO THAT THE TRANSFERS ARE LINEAR. I.E. di/dt IS
c                 CONSTANT.
c             (b) THE MULTISTAGE CALCULATION TAKES NO ACCOUNT OF TRANSFORMER
c                 ACTION CLAMPING.
c                 THIS ASSUMPTION MEANS THAT THE TRANSFER TIME DEPENDS ONLY
c                 ON THE CLAMP VOLTAGE OF THE OPENING SWITCH IN QUESTION.
c                 IT IS RECOGNISED THAT THIS LIMITS THE USEFULNESS OF SOME OF
c                 THE CALCULATIONS.
c
c
c     INPUT/OUTPUT DEVICES:  0 INTERACTIVE TERMINAL
c                            1 DATAFILE
c                            2 RESULTS FILE
c
c
c     UNITS COMMONLY USED:
c
c     inductance   microhenries
c     time         microseconds
c     voltage      kilovolts
c     energy       joules
c     current      amps
c
c
c     NOTE ON GLOBAL VARIABLES:
c
c     THE PROGRAM PERFORMS TWO DIFFERENT TASKS: CALCULATIONS FOR MULTI-
c     STEP MEATGRINDER CIRCUITS. WHICH USE DATAFILE DATA AND ARE
c     AUTOMATIC. AND CALCULATIONS IN WHICH
c     THE COMPUTER IS USED AS A SOPHISTICATED CALCULATOR.
c
c     THE MULTI-STEP CALCULATIONS CONSIST OF TWO MAIN FUNCTIONS:
c     DATAFILE EDITING. AND ACTUAL CALCULATION.  THE CALCULATION
c     FUNCTION TREATS THE DATAFILE ITEMS
c     AS CONSTANT AND AS SUCH THEY ARE IN COMMON BLOCKS.  DURING EDITING.
c     HOWEVER. THEY ARE PASSED AS ARGUMENTS.
c
c
c
c     THE MAIN PROGRAM PRESENTS THE AVAILABLE CHOICES AND DIRECTS TO
c     ONE OF THE TWO LEVEL 2 ROUTINES.
c     ========================================
      program ideal

      implicit double precision (a-h,o-z)
      call sfc(">udd>EL>MGPimperton>mg>mgideal>ideal")

```

STC

```
      integer secopt,sec
      parameter (maxopt=3)

      common/option/mnopt

      data intro,main,sec/1,1,2/
c     ======================================
c     SET TERMINAL
c     ======================================
      open(0,defer=.true.,prompt=.false.)

      call mesage (0,intro)
 10   call menu (main,mnopt)
      if (mnopt.eq.maxopt) goto 50
 20   call menu (sec,secopt)

      if (secopt.eq.1) then
         call data
         goto 20
      else if (secopt.eq.2) then
         call rncalc
         goto 20
      else
         goto 10
      endif
c     ======================================
c     RESET TERMINAL
c     ======================================
 50   open(0,defer=.false.)

      stop
      end

c     *** LEVEL 2 ROUTINES ***

c     ======================================
c     "data" SIMPLY PROVIDES A CHOICE BETWEEN EDITING AN EXISTING FILE
c     OR CREATING A BRAND NEW ONE.
c
c     CALLING ROUTINE: main program
c
c     (This routine is rather superfluous but cannot be removed without
c     altering the structure of the rest of the program.)
c     ======================================
      subroutine data

      implicit double precision (a-h,o-z)
      integer datmod
      logical newdat

      data datmod/1/

      call menu2 (datmod,newdat)

      if (newdat) then
         call datnew
      else
         print*,"Modify existing data:"
         call mdexst
      endif

      return
      end
```

```
c       ========================================
c       "rncalc" CONNECTS TO A SUITABLE DATAFILE, CALLS THE APPROPRIATE
c       CALCULATION ROUTINE, THEN GIVES THE OPTION TO RUN AGAIN, WITH
c       A DIFFERENT DATAFILE IF REQUIRED.
c
c       CALLING ROUTINE: main program
c       ========================================
        subroutine rncalc
c       ========================================
c       TYPE DECLARATIONS AND DATA ARE FOR MENU PURPOSES.
c       THE CHARACTER VARIABLE "caller" IS REQUIRED TO ENABLE "oldfle"
c       TO SAY WHERE IT WAS CALLED FROM WHEN PRINTING ERROR MESSAGES.
c       ========================================
        implicit double precision (a-h,o-z)
        integer rerun
        character*10 caller
        logical again,nwfile

        common/option/mnopt

        data rerun,newfle/2,3/
        data caller/"runcalc"/
c       ========================================
c       GET FILE NAME, CHECK IT IS  SUITABLE AND CONNECT TO IT.
c       ========================================
   10   call oldfle (caller,x)
c       ========================================
c       CALL APPROPRIATE CALC. ROUTINE.
c       ========================================
   20   if (mnopt.eq.1) then
           call calc1
        else if (mnopt.eq.2) then
           call calc2
        endif
c       ========================================
c       RUN AGAIN?
c       ========================================
        call menu2 (rerun,again)
        if (again) then
           call menu2 (newfle,nwfile)
           if (nwfile) then
              close(1)
              goto 10
           else
c       ========================================
c       THE READ STATEMENT IS USED TO POSITION THE FILE CORRECTLY
c       (AS THOUGH "oldfle" HAD BEEN CALLED).   "x" IS JUST A DUMMY NAME.
c       ========================================
              rewind(1)
              read(1) x
              goto 20
           endif
        endif

        close(1)
        return
        end

c       **** LEVEL 3 ROUTINES ****

c       ========================================
c       "datnew" CREATES A NEW DATAFILE AND INTERACTIVELY READS
c       APPROPRIATE DATA ACCORDING TO THE MAIN OPTION.
c
```

```
c     CALLING ROUTINE: data
c     ========================================
      subroutine datnew

      implicit double precision (a-h,o-z)
      parameter(maxns=50)

      integer create
      double precision ind(maxns),idl
      character caller*10

      common/option/mnopt

      dimension cf(maxns,maxns),vs(maxns)

      data idl,create/1.1,2/
      data caller/"datnew"/
c     ========================================
c     CREATE NEW FILE
c     ========================================
      call newfle (caller)
c     ========================================
c     GET DATA
c     ========================================
      if (mnopt.eq.1.or.mnopt.eq.2) then
c     ========================================
c     GET THE FOLLOWING:  no. of coil sections
c                         coil inductances
c                         coupling factors
c                         switch voltages (for use when load voltage is
c                                          not specified)
c                         load voltage (for specifying a maximum when
c                                       appropriate)
c     ========================================
         ns=nsect(maxns)
         call getind (ns,ind)
         call getcf (ns,cf)
         call gtswvt (ns,vs)
         call ldvolt(vload)
c     ========================================
c     WRITE DATA; "idl" IS AN IDENTIFICATION NO. TO SHOW THAT THE FILE IS
c     SUITABLE FOR THAT PARTICULAR OPTION.
c     ========================================
         write(1) idl
         write(1) ns
         write(1) (ind(i),i=1,ns),((cf(j,i),j=1,ns),i=1,ns),
     +             (vs(i),i=1,ns),vload


      endif

      call mesage (mnopt,create)

      close(1)
      return
      end

c     ========================================
c     "mdexst" CONNECTS TO AN EXISTING DATAFILE, VERIFIES ITS
c     SUITABILITY AND ACCEPTS MODIFICATIONS.
c
c     CALLING ROUTINE: data
c     ========================================
      subroutine mdexst
```

```
      implicit double precision (a-h,o-z)
      parameter(maxns=50)

      double precision ind(maxns),idnum
      character caller*10,reply*1

      common/option/mnopt

      dimension cf(maxns,maxns),vs(maxns)

      data modify/3/
      data caller/"modexist"/
c     ======================================
c     CONNECT TO A FILE.  "idnum" IS RETURNED SO THAT IT CAN BE WRITTEN
c     AFTER REWINDING.
c     ======================================
      call oldfle (caller,idnum)
c     ======================================
c     PRINT PRESENT DATA AND ACCEPT MODIFICATIONS.
c     ======================================
      if (mnopt.eq.1.or.mnopt.eq.2) then
c     ======================================
c     FIRST ITEM TO BE READ IS "ns" BECAUSE "idnum" WAS READ BY
c     "oldfle".
c     ======================================
         read(1) ns
         read(1) (ind(i),i=1,ns),((cf(j,i),i=1,ns),j=1,ns),
     +             (vs(i),i=1,ns),vload

         call modind (ns,ind)
         call modcf (ns,cf)
         call mdswvt (ns,vs)

         write(0,*)
         write(0,*)"Max. load voltage: ",vload,"kV    Modify?  "
         read*,reply
         if (reply.eq."y") call ldvolt(vload)
c     ======================================
c     WRITE BACK TO FILE
c     ======================================
         rewind(1)
         write(1) idnum
         write(1) ns
         write(1) (ind(i),i=1,ns),((cf(j,i),i=1,ns),j=1,ns),
     +               (vs(i),i=1,ns),vload

      endif

      call mesage (mnopt,modify)

      close(1)
      return
      end

c     ======================================
c     "calc1" IS THE MAIN CALCULATION ROUTINE FOR OPTION 1.  THE
c     ROUTINES IT CALLS ARE LABELLED "1" OR "12" ETC. TO ALLOW
c     FOR FURTHER OPTIONS.
c
c     THE PROCEDURE IT FOLLOWS IS:
c
c         1.  ESTABLISH THE LOAD AND THE OPERATION MODES REQUIRED.
c         2.  CALCULATE THE INITIAL ENERGY AND CURRENT.
```

```
c         3.  CALL A SEQUENCE OF CALCULATION ROUTINES ACCORDING TO
c             WHETHER OPERATION IS FORWARD OR REVERSE.
c         4.  CALCULATE THE REMAINING SINGLE-VALUED QUANTITIES.
c         5.  PRINT THE RESULTS.
c
c     VARIABLE DICTIONARY:
c
c     indvtk    coupling factors relevant to calculation of induced
c               voltages (2 values for each node)
c     ldvfix    flag to indicate whether there is a maximum load voltage
c     curr      current
c     en        energy
c     cumind    cumulative inductances relevant to transfers
c     transk    coupling factors relevant to transfers
c     lthree    array of "L3" values for use in calculating induced
c               voltages (1 value for each node)
c     cmult     current multiplication at each step
c     stpefy    step efficiency
c     effind    effective inductance for calculating transfer dynamics
c     swvolt    results array for switch voltages (may have same values
c               as "vs" array)
c     loadv     results array for load voltage at each step (some elements
c               may contain the pre-specified value)
c     trtime    transfer time
c     indced    induced voltages at each step
c     pnalty    efficiency penalty due to the fact that not all the
c               final energy is in the load (fwd: uncoupled load only;
c               rev: any load)
c     totefy    (final load energy)/(initial stored energy)
c     totmlt    final current/initial current
c     power     instantaneous load power:  (energy delivered)/(tot. transfer
c                                                                     time)
c
c     CALLING ROUTINE:  rncalc
c     ========================================
      subroutine calc1

      implicit double precision (a-h,o-z)
      double precision ind,indvtk,lthree,indced,loadv
      integer dir,dcmprs,fixldv,see,reswr
      logical fwd,decomp,ldvfix,review,rwrite

      parameter(maxns=50)

      common/coil1/ind(maxns),cf(maxns,maxns)
      common/volt1/vs(maxns),vload

      dimension curr(maxns+1),en(maxns+1),cumind(maxns),transk(maxns-1)
     +          ,lthree(maxns-2,maxns-2),indvtk(2,maxns-2,maxns-2),
     +          cmult(maxns),stpefy(maxns),effind(maxns),
     +          swvolt(maxns),loadv(maxns),trtime(maxns),
     +          indced(maxns-2,maxns-2)

      data dir,dcmprs,fixldv,see,reswr/4,5,6,7,8/
c     ========================================
c     READ DATA.  FILE IS CONNECTED AND POSITIONED AT "ns".
c     ========================================
      read(1) ns
      read(1) (ind(i),i=1,ns),((cf(j,i),i=1,ns),j=1,ns),
     +        (vs(i),i=1,ns),vload
c     ========================================
c     NORMALISE DATA SO THAT ALL CALCULATION EXPRESSIONS ARE FREE FROM
c     ADJUSTMENT FACTORS ("DE-NORMALISE" FOR RESULTS DISPLAY).
c     ========================================
```

```
      do 20,i=1,ns
         ind(i)=ind(i)*1E-6
         vs(i)=vs(i)*1E3
20    continue
      vload=vload*1E3
c     ========================================
c     GET MODES.  DECOMPRESSION IS ONLY AVAILABLE IN FORWARD.
c     ========================================
      call menu2 (dir,fwd)
      call load(uload)
c     ========================================
c     NORMALISE LOAD
c     ========================================
      uload=uload*1E-6

      if (fwd.and.uload.ne.0.0) then
         call menu2 (dcmprs,decomp)
      else
         decomp=.false.
      endif

      call menu2 (fixldv,ldvfix)
c     ========================================
c     INITIAL CONDITIONS
c     ========================================
      call initl (ns,uload,fwd,decomp,curr(1),en(1))
c     ========================================
c     MAIN CALCULATIONS
c     ========================================
      if (fwd) then
         call fdadjl (ns,cumind,transk,lthree,indvtk)

         call fwdl1 (ns,decomp,uload,cumind,transk,
     +               curr,cmult,en,stpefy)

         call fwdl2 (ns,decomp,ldvfix,uload,curr,cumind,
     +               transk,lthree,indvtk,
     +               effind,swvolt,loadv,trtime,indced)

      else
ctemp         call rdadjl (ns,cumind,transk,lthree,indvtk)
ctemp         call revl1 (ns,uload,cumind,transk,
ctemp     +                curr,cmult,en,stpefy)
ctemp         call revl2 (ns,ldvfix,uload,curr,transk,lthree,indvtk,
ctemp     +                effind,swvolt,loadv,trtime,indced)
      print*,"rdadjl,revl1,revl2 all called"                    TEMP
      endif

      call miscl (ns,fwd,decomp,uload,curr(1),curr(ns+1),en(1),en(ns+1),
     +                                                     trtime,
     +            pnalty,totefy,totmlt,power)
c     ========================================
c     RESULTS
c     ========================================
10    call pdatal (fwd,uload,ns)
      call rsltl1 (ns,decomp,curr,cmult,en,stpefy,pnalty,totefy,totmlt,
     +             power)
      call rsltl2 (ns,decomp,fwd,swvolt,loadv,trtime,indced)

      call menu2 (see,review)
      if (review) goto 10

ctemp      call menu2 (reswr,rwrite)

```

```
      return
      end

c     ========================================
c     "calc2" IS THE CONTROLLING ROUTINE FOR THE EXPRESSION EVALUATION
c     OPTION.  IT FIRST READS THE NECESSARY DATA FROM THE FILE AND THEN
c     PRESENTS THE APPROPRIATE MENU.  "calc1" AND "calc2" ARE MUTUALLY
c     EXCLUSIVE.
c
c     CALLING ROUTINE: rncalc
c     ========================================
      subroutine calc2

      implicit double precision(a-h,o-z)
      double precision ind
      integer calc,calopt

      parameter (maxns=50)
      data calc/3/

      common/coil1/ind(maxns),cf(maxns,maxns)
c     ========================================
c     READ DATA.  FILE IS CONNECTED AND POSITIONED AT "ns".
c     ========================================
      read(1) ns
      read(1) (ind(i),i=1,ns),((cf(j,i),i=1,ns),j=1,ns)
c     ========================================
c     NORMALISE DATA FOR CALCULATIONS
c     ========================================
      do 20,i=1,ns
         ind(i)=ind(i)*1E-6
 20   continue
c     ========================================
c     PRINT MENU AND CALL ROUTINES
c     ========================================
 10   call menu (calc,calopt)

      if (calopt.eq.1) then
         call calc21 (ns)
         goto 10
      else if (calopt.eq.2) then
         call calc22
         goto 10
      else if (calopt.eq.3) then
         call calc23
         goto 10
      endif

      return
      end


c     **** LEVEL 4 ROUTINES ****


c     ========================================
c     "nsect" GETS THE NUMBER OF SECTIONS IN THE MEATGRINDER COIL.
c     THE MINIMUM HAS TO BE 2 TO ALLOW ANY TRANSFERS.
c
c     CALLING ROUTINE: datnew
c     ========================================
      function nsect(maxns)

      implicit double precision (a-h,o-z)
```

```
      write(0,*)
      write(0,*)
10    write(0,*)"No. of coil sections?  "
      read*,n
      if (n.lt.2.or.n.gt.maxns) goto 10

      nsect=n

      return
      end

c     ======================================
c     "getind" FILLS THE COIL INDUCTANCE ARRAY.
c
c     CALLING ROUTINE: datnew
c     ======================================
      subroutine getind (ns,ind)

      implicit double precision (a-h,o-z)
      parameter(maxns=50)

      double precision ind(maxns),indmin
c     ======================================
c     MIN. VALUE (0.1nH) IS AN ARBITRARY CHOICE.
c     ======================================
      parameter(indmin=1E-4)

      write(0,*)
      write(0,*)
      print*,"COIL INDUCTANCES (microhenries):"
      write(0,*)
      do 10,i=1,ns
5        write(0,*)"L",i,"="
         read*,ind(i)
         if (ind(i).lt.indmin) goto 5
10    continue

      return
      end

c     ======================================
c     "getcf" FILLS THE NON-DIAGONAL ELEMENTS OF THE
c     COUPLING FACTOR ARRAY.  THIS ARRAY IS MADE SYMMETRICAL
c     FOR CONVENIENCE.  THE LIMITS ENSURE REASONABLE VALUES
c     AND PREVENT ANY REAL ARITHMETIC PROBLEMS.
c
c     CALLING ROUTINE: datnew
c     ======================================
      subroutine getcf (ns,cf)

      implicit double precision (a-h,o-z)
      parameter(maxns=50,cfmin=0.1,cfmax=0.99)

      dimension cf(maxns,maxns)

      write(0,*)
      write(0,*)
      print*,"COUPLING FACTORS:"
      write(0,*)
      do 10,j=1,ns-1
         do 10,i=j+1,ns
5           write(0,*)"k(",j,"_",i,")="
            read*,cf(j,i)
```

```
                if (cf(j,i).lt.cfmin.or.cf(j,i).gt.cfmax) goto 5

                cf(i,j)=cf(j,i)
 10     continue

        return
        end

c       ========================================
c       "gtswvt" ("get_switch_voltages") FILLS THE SWITCH VOLTAGE
c       ARRAY; THE DECOMPRESSION SWITCH VOLTAGE IS IN THE FIRST
c       ELEMENT.
c
c       THE OPTIONS SELECTED DETERMINE WHICH OF THE VALUES IN THIS
c       ARRAY ARE ACTUALLY USED IN CALCULATIONS.
c
c       MIN. VALUE (10V) IS ARBITRARY.
c
c       CALLING ROUTINE: datnew
c       ========================================
        subroutine gtswvt(ns,vs)

        implicit double precision (a-h,o-z)
        parameter(maxns=50,vsmin=0.001)

        dimension vs(maxns)

        write(0,*)
        write(0,*)
        print*,"CLAMP VOLTAGES (kV):"
        write(0,*)
        print*,"Decompression switch:"
        write(0,*)

 10     write(0,*)"Vsw.decomp="
        read*,vs(1)
        if (vs(1).lt.vsmin) goto 10

        write(0,*)
        print*,"Meatgrinder switches:"
        write(0,*)
        do 20,i=2,ns
 15        write(0,*)"Vsw",i-1,"="
           read*,vs(i)
           if (vs(i).lt.vsmin) goto 15
 20     continue

        return
        end

c       ========================================
c       "ldvolt" GETS THE MAXIMUM LOAD VOLTAGE.  THIS VALUE IS
c       USED IN ACCORDANCE WITH THE SELECTED OPTIONS AND
c       OPERATING MODES.
c
c       MIN. VALUE 10V.
c
c       CALLING ROUTINES: datnew
c                         mdexst
c       ========================================
        subroutine ldvolt(vload)

        implicit double precision (a-h,o-z)
        double precision minldv
```

```
      parameter(minldv=0.01)

      write(0,*)
      write(0,*)
10    write(0,*)"Max. load voltage (kV)?  "
      read*,vload
      if (vload.lt.minldv) goto 10

      return
      end

c     ========================================
c     "modind" ACCEPTS MODIFICATIONS TO THE COIL INDUCTANCE ARRAY.
c
c     CALLING ROUTINE: mdexst
c     ========================================
      subroutine modind(ns,ind)

      implicit double precision (a-h,o-z)
      double precision ind,indmin

      parameter(maxns=50,indmin=1E-4)

      dimension ind(maxns)

      write(0,*)
      write(0,*)
      print*,"COIL INDUCTANCES (microhenries):"
      write(0,*)
      do 10 i=1,ns
         write(0,*)"L",i,"=",ind(i)
10    continue

20    write(0,*)
      write(0,*)"Subscript of value to be changed (0 to quit)?  "
      read*,n
      if (n.lt.0.or.n.gt.ns) goto 20
      if (n.ne.0) then
30       write(0,*)"L",n,"="
         read*,ind(n)
         if (ind(n).lt.indmin) goto 30
         goto 20
      endif

      return
      end

c     ========================================
c     "modcf" ACCEPTS MODIFICATIONS TO THE COUPLING FACTOR ARRAY.
c
c     CALLING ROUTINE: mdexst
c     ========================================
      subroutine modcf(ns,cf)

      implicit double precision (a-h,o-z)
      parameter(maxns=50,cfmin=0.0,cfmax=0.99)

      dimension cf(maxns,maxns)

      write(0,*)
      write(0,*)
      print*,"COUPLING FACTORS:"
      write(0,*)
```

```
      do 10,j=1,ns-1
         do 10,i=j+1,ns
            write(0,*)"k(",j,"_",i,")=",cf(j,i)
10    continue
c     ========================================
c     THE ODD-LOOKING STRUCTURE IS USED SO THAT THE USER
c     ONLY HAS TO ENTER A SINGLE 0 TO QUIT.
c     ========================================
20    write(0,*)
      write(0,*)"Value to be changed:"
25    write(0,*)"1st subscript (0 to quit)?  "
      read*,m
      if (m.lt.0.or.m.gt.ns) goto 25

      if (m.ne.0) then
30       write(0,*)"2nd subscript (0 to quit)?  "
         read*,n
         if (n.lt.0.or.n.gt.ns) goto 30

         if (n.ne.0) then
c     ========================================
c     DIAGONAL ELEMENTS ARE NOT DEFINED
c     ========================================
            if (n.eq.m) then
               goto 20

            else
40             write(0,*)"k(",m,"_",n,")="
               read*,cf(m,n)
               if (cf(m,n).lt.cfmin.or.cf(m,n).gt.cfmax) goto 40

               cf(n,m)=cf(m,n)
               goto 20
            endif
         endif
      endif

      return
      end

c     ========================================
c     "mdswvt" ("mod_switch_voltages") ACCEPTS MODIFICATIONS
c     TO THE SWITCH VOLTAGE ARRAY.
c
c     CALLING ROUTINE: mdexst
c     ========================================
      subroutine mdswvt(ns,vs)

      implicit double precision (a-h,o-z)
      parameter(maxns=50,vsmin=0.001)

      dimension vs(maxns)

      write(0,*)
      write(0,*)
      print*,"CLAMP VOLTAGES (kV):"
      write(0,*)
      write(0,*)"1  Vsw.decomp=",vs(1)

      do 10,i=2,ns
         write(0,*)i,"  Vsw",i-1,"=",vs(i)
10    continue

      write(0,*)
```

```
20    write(0,*)"Ref. no. of value to be changed (0 to quit)?  "
      read*,n
      if (n.lt.0.or.n.gt.ns) goto 20

      if (n.ne.0) then
         if (n.eq.1) then
30          write(0,*)"Vsw.decomp="
            read*,vs(1)
            if (vs(1).lt.vsmin) goto 30
         else
40          write(0,*)"Vsw",n-1,"="
            read*,vs(n)
            if (vs(n).lt.vsmin) goto 40
         endif

         goto 20
      endif

      return
      end

c     ========================================
c     "load" GETS THE UNCOUPLED LOAD.
c
c     CALLING ROUTINE: calc1
c     ========================================
      subroutine load (uload)

      implicit double precision (a-h,o-z)

      write(0,*)
10    write(0,*)"Uncoupled load (microhenries)?  "
      read*,uload

      if (uload.eq.0.0) then
         print*,"Last section of meatgrinder coil will be taken as load.
     +          "
      endif

      return
      end

c     ========================================
c     "init1" PROVIDES THE INITIAL CURRENT AND ENERGY.
c
c     HAVING READ THE INITIAL CURRENT, THE ENERGY IS SIMPLY
c     CALCULATED ACCORDING TO THE OPERATION MODES SELECTED.
c
c     VARIABLE DICTIONARY:
c
c     mincur   minimum current (arbitrary limit)
c     mtgndr   total meatgrinder inductance
c     induct   inductance in which initial energy resides
c
c     CALLING ROUTINE:  calc1
c     ========================================
      subroutine init1 (ns,uload,fwd,decomp,
     +                  curr1,en1)

      implicit double precision (a-h,o-z)
      double precision ind,mincur,mtgndr,induct
      logical fwd,decomp

      parameter(maxns=50,mincur=0.01)
```

```
      common/coil1/ind(maxns),cf(maxns,maxns)

      write(0,*)
10    write(0,*)"Initial current (amps)?  "
      read*,curr1
      if (curr1.lt.mincur) goto10

      mtgndr=totind(1,ns)

      if (.not.fwd) then
         induct=ind(1)
      else if (fwd.and.decomp) then
         induct=mtgndr
      else
         induct=mtgndr+uload
      endif

      en1=0.5*induct*curr1*curr1

      return
      end

c     ========================================
c     "fdadj1" ("forward_data_adjust1") TAKES THE RAW COIL DATA
c     AND PRODUCES ARRAYS WHICH CAN MORE EASILY BE APPLIED TO THE
c     VARIOUS FORMULAE.  THESE ARRAYS ARE DEFINED IN THE INTRO. TO
c     "calc1".
c
c     CALLING ROUTINE: calc1
c     ========================================
      subroutine fdadj1 (ns,cumind,transk,lthree,indvtk)

      implicit double precision (a-h,o-z)
      double precision ind,indvtk,lthree,mut12,mut13

      parameter(maxns=50)

      common/coil1/ind(maxns),cf(maxns,maxns)
c     ========================================
c     ARRAY SIZES:
c
c     cumind    same as no. of coil sections because last element
c               is just a single inductance, i.e. ind(ns)
c
c     transk    same as no. of switching steps, i.e. ns-1
c
c     lthree    there are ns-2 switching steps which produce induced
c               voltages (fwd or rev), with a maximum of ns-2 nodes
c               involved
c
c     indvtk    calculation of the induced voltage at any node requires three
c               coupling factors, one of which is the "normal" one
c               involved in the energy transfer; hence two more have to
c               be calculated, and the array has an extra dimension
c
c     NOTE: ARRAYS HAVE TO BE DIMENSIONED TO "maxns" TO ENSURE
c     CONSISTENCY BETWEEN ALL ROUTINES.
c     ========================================
      dimension cumind(maxns),transk(maxns-1),lthree(maxns-2,maxns-2),
     +          indvtk(2,maxns-2,maxns-2)
c     ========================================
c     CALC. CUM. INDUCTANCES FOR FORWARD TRANSFERS
c     ========================================
```

```
      do 10,i=1,ns
         cumind(i)=totind(i,ns)
10    continue
c     ========================================
c     CALC. TRANSFER COUPLING FACTORS ACCORDING TO:
c
c     k12=(L12-L1-L2)/(2*sqrt[L1*L2])
c     ========================================
      do 20,j=1,ns-1
         transk(j)=(cumind(j)-cumind(j+1)-ind(j))/(2*sqrt(cumind(j+1)
     +                                                  *ind(j)))
20    continue

c     ========================================
c     LOOP TO CALCULATE "lthree" AND "indvtk" VALUES.
c
c     "i" IS THE VOLTAGE-INDUCING  STEP NUMBER (N.B. THE FIRST STEP WHICH
c     PRODUCES AN INDUCED VOLTAGE IS ACTUALLY THE SECOND SWITCHING STEP).
c
c     "j" IS THE NODE NUMBER DURING STEP "i".  NODE 1 IS NEAREST TO THE
c     SECTION BEING SWITCHED OUT.  FOR EXAMPLE, IN THE SECOND
c     VOLTAGE-PRODUCING STEP, L3 IS SWITCHED OUT (REGARDED AS "L1" IN
c     THE GENERAL FORMULA). NODE 1 IS THE JUNCTION BETWEEN L2 & L1
c     AND NODE 2 IS THE END OF L1.
c
c     FOR EACH NODE, "L3" IS THE TOTAL NON-PARTICIPATIVE INDUCTANCE,
c     AND THIS LEADS TO THE EXPRESSION FOR lthree(j,i).
c     ========================================
      i=1
      j=1
      do 40,i=1,ns-2
         do 40,j=1,i
            lthree(j,i)=totind(i-j+1,i)
c     ========================================
c     indvtk(1,j,i) HOLDS WHAT IS REGARDED AS "k31" IN THE GENERAL FORMULA.
c
c     "L3" HAS JUST BEEN CALCULATED, "L1" IS THE SECTION BEING SWITCHED
c     OUT (ind(i+1)), AND THE TOTAL IS AGAIN FOUND BY "totind".
c     ========================================
            temp=sqrt(lthree(j,i)*ind(i+1))
            indvtk(1,j,i)=(totind(i-j+1,i+1)-lthree(j,i)-ind(i+1))/
     +                    (2*temp)
c     ========================================
c     indvtk(2,j,i) HOLDS "k32" AND SINCE "L3" (THE NON-PARTICIPATIVE
c     INDUCTANCE WHICH EXPERIENCES THE INDUCED VOLTAGE) AND "L2" (THE
c     INDUCTANCE REMAINING IN CIRCUIT AFTER THE TRANSFER) ARE NOT
c     ADJACENT, THE TOTAL OF THREE INDUCTANCES MUST BE USED TO
c     EVALUATE IT.
c
c     "L2" IS cumind(i+2), "L123" IS A "totind" FUNCTION AND mut12, mut13
c     ARE TWO OF THE THREE MUTUALS.  "M32" IS USED TO FIND "k32".
c
c     NOTICE THE USE OF transk(i+1) AS "k12".
c     ========================================
            mut12=transk(i+1)*sqrt(ind(i+1)*cumind(i+2))
            mut13=indvtk(1,j,i)*temp

            top=totind(i-j+1,ns)-ind(i+1)-cumind(i+2)-lthree(j,i)-
     +                                              2*(mut12+mut13)
            bottom=2*sqrt(lthree(j,i)*cumind(i+2))

            indvtk(2,j,i)=top/bottom
40    continue

```

```
      return
      end

c     ========================================
c     "fwd11" PRODUCES THE FIRST BATCH OF RESULTS ARRAYS.
c
c     CALLING ROUTINE: calc1
c     ========================================
      subroutine fwd11 (ns,decomp,uload,cumind,transk,
     +                  curr,cmult,en,stpefy)

      implicit double precision (a-h,o-z)
      double precision ind,ltwo,mutual,mtgndr
      logical decomp

      parameter(maxns=50)

      common/coil1/ind(maxns),cf(maxns,maxns)
c     ========================================
c     ARRAY SIZES:
c
c     THERE ARE ns-1 TRANSFER STEPS AND POSSIBLY A DECOMPRESSION STEP.
c     THE ENERGY AND CURRENT ARRAYS MUST ALSO STORE THE INITIAL VALUES.
c     ========================================
      dimension cumind(maxns),transk(maxns-1),curr(maxns+1),
     +          cmult(maxns),en(maxns+1),stpefy(maxns)
c     ========================================
c     DECOMPRESSION STEP FIRST
c     ========================================
      if (decomp) then
         mtgndr=cumind(1)
c        ========================================
c        CURRENT RATIO FOR DECOMP. IS LIKE A REVERSE STEP WITH k=0.
c        ========================================
         cmult(1)=mtgndr/(mtgndr+uload)
         curr(2)=curr(1)*cmult(1)
c        ========================================
c        CALCULATION OF ENERGY ALWAYS USES I*I RATHER THAN I**2
c        (MORE EFFICIENT).  ENERGY IN JOULES.
c        ========================================
         en(2)=0.5*(mtgndr+uload)*curr(2)*curr(2)
         stpefy(1)=en(2)/en(1)
      else
c        ========================================
c        IF NO DECOMP., THESE VALUES ARE NEEDED FOR OTHER CALCULATIONS
c        BUT ARE NOT PRINTED AT THE RESULTS STAGE.
c        ========================================
         cmult(1)=1.0
         curr(2)=curr(1)
         en(2)=en(1)
         stpefy(1)=1.0
      endif

c     ========================================
c     TRANSFER STEPS
c     ========================================
      do 10,i=1,ns-1
c        ========================================
c        BASIC FORMULA IS I2/I1=(L2+M12)/L2.
c
c        ltwo WILL BE ACCURATE WHATEVER THE VALUE OF uload, WHEREAS
c        THE ADJUSTMENT FACTOR FOR THE COUPLING FACTOR MAY NOT BE
c        EVALUATED TO EXACTLY ONE IF uload=0.0.
c        ========================================
```

```
         ltwo=cumind(i+1)+uload

         if (uload.ne.0.0) then
            cfact=transk(i)*sqrt(cumind(i+1)/ltwo)
         else
            cfact=transk(i)
         endif

         mutual=cfact*sqrt(ind(i)*ltwo)

         cmult(i+1)=(ltwo+mutual)/ltwo

         curr(i+2)=curr(i+1)*cmult(i+1)
         en(i+2)=0.5*ltwo*curr(i+2)*curr(i+2)
         stpefy(i+1)=en(i+2)/en(i+1)
 10   continue

      return
      end

c     ========================================
c     "fwd12" CALCULATES THE TRANSFER DYNAMICS QUANTITIES,
c     i.e. EFFECTIVE INDUCTANCE, LOAD VOLTAGE (IF NOT SPECIFIED)
c     OR SWITCH VOLTAGE (IF LOAD VOLTAGE SPECIFIED), AND
c     TRANSFER TIME (ASSUMING LINEAR TRANSFER).
c
c     THE CALCULATIONS ARE COMPLICATED BY THE NEED TO CONSIDER
c     CAREFULLY WHETHER THE LOAD VOLTAGE IS FIXED AND WHETHER
c     THE LOAD IS COUPLED OR UNCOUPLED.  THE ROUTINE IS STRUCTURED
c     TO MAKE THE LOGIC EASY TO FOLLOW AND CONEQUENTLY THERE
c     ARE SOME INEFFICIENCIES, i.e. REPEAT TESTS OR CALCULATIONS.
c
c     REMEMBER THAT THE ARRAYS loadv AND swvolt ARE THOSE WHICH
c     APPEAR IN THE RESULTS.
c
c     CALLING ROUTINE: calc1
c     ========================================
      subroutine fwd12 (ns,decomp,ldvfix,uload,curr,cumind,
     +                  transk,lthree,indvtk,
     +                  effind,swvolt,loadv,trtime,indced)

      implicit double precision (a-h,o-z)
      double precision ind,lthree,indvtk,loadv,indced,mtgndr,mutld
      logical decomp,ldvfix

      parameter(maxns=50)

      common/coil1/ind(maxns),cf(maxns,maxns)
      common/volt1/vs(maxns),vload
c     ========================================
c     THE ARRAY mutld (IF EVALUATED) CONTAINS, FOR EACH STEP, THE TWO
c     MUTUAL INDUCTANCES REQUIRED TO CALCULATE THE VOLTAGE ON AN
c     UNCOUPLED LOAD.  AS THERE IS NO DECOMPRESSION AND THE LAST
c     TRANSFER PRODUCES ZERO VOLTAGE ON A COUPLED LOAD, THERE ARE
c     ns-2 STEPS REQUIRING SUCH MUTUALS.
c     ========================================
      dimension curr(maxns+1),cumind(maxns),transk(maxns-1),
     +          lthree(maxns-2,maxns-2),indvtk(2,maxns-2,maxns-2),
     +          effind(maxns),swvolt(maxns),loadv(maxns),trtime(maxns),
     +          indced(maxns-2,maxns-2),mutld(2,maxns-2)
c     ========================================
c     INITIALISE LOAD VOLTAGE AND SWITCH VOLTAGE ARRAYS
c     ========================================
      if (ldvfix) then
```

```
            do 10,i=1,ns
               loadv(i)=vload
 10         continue
         else
            do 20,j=1,ns
               swvolt(j)=vs(j)
 20         continue
         endif
c        ========================================
c        THIS TEST APPLIES WHETHER LOAD VOLTAGE IS FIXED OR NOT.
c        ========================================
         if (uload.eq.0.0) then
            call cldmut (ns,cumind,mutld)
         endif
c        ========================================
c        IN THIS CASE, THE ARRAY ELEMENTS CONTAINING THE DECOMPRESSION
c        VALUES DO NOT HAVE TO BE DEFINED IF THERE IS NO DECOMPRESSION
c        (THEY ARE JUST NOT PRINTED AT THE RESULTS STAGE).
c
c        DECOMPRESSION IS A REVERSE STEP WITH k=0.
c        ========================================
         if (decomp) then
            mtgndr=totind(1,ns)
            effind(1)=(mtgndr*uload)/(mtgndr+uload)
c           ========================================
c           IN DECOMPRESSION, SWITCH VOLTAGE=LOAD VOLTAGE.  THIS BLOCK ASSIGNS
c           WHICHEVER ONE HAS NOT BEEN DEFINED.
c           ========================================
            if (ldvfix) then
               swvolt(1)=vload
            else
               loadv(1)=swvolt(1)
            endif

            trtime(1)=(effind(1)*(curr(1)-curr(2)))/swvolt(1)
         endif
c        ========================================
c        MAIN LOOP
c        ========================================
         call fwd121 (ns,ldvfix,uload,curr,cumind,transk,mutld,
     +                effind,swvolt,loadv,trtime)

         call fndcel (ns,uload,cumind,transk,lthree,indvtk,swvolt,
     +                indced)

         return
         end

c        ========================================
c        "miscl" ("miscellaneous_calculations1") PRODUCES FOUR SINGLE
c        RESULTS.
c
c        CALLING ROUTINE: calcl
c        ========================================
         subroutine miscl (ns,fwd,decomp,uload,curr1,curr2,en1,en2,trtime,
     +                     pnalty,totefy,totmlt,power)

         implicit double precision (a-h,o-z)
         double precision ind,mtgndr,mutual
         logical fwd,decomp

         parameter(maxns=50)

         common/coil1/ind(maxns),cf(maxns,maxns)
```

```
      dimension trtime(maxns)
c     ========================================
c     NOT ALL THE FINAL ENERGY RESIDES IN THE LOAD
c     ========================================
      if (fwd) then

         if (uload.ne.0.0) then
            pnalty=uload/(uload+ind(ns))
         else
            pnalty=1.0
         endif

      else

         mtgndr=totind(1,ns)

         if (uload.ne.0.0) then
            pnalty=uload/(uload+mtgndr)
         else
c           ========================================
c           COUPLED LOAD IN REVERSE: THE ENERGY ASSOCIATED WITH
c           THE LAST SECTION OF THE MEATGRINDER [ind(ns)] IS STORED IN THE
c           SELF INDUCTANCE AND IN THE MUTUAL INDUCTANCE OF
c           THAT SECTION WITH EVERY OTHER SECTION.
c
c           (NO PENALTY FOR A COUPLED LOAD IN FORWARD.)
c           ========================================
            totmut=0.0
            do 10,i=1,ns-1
               mutual=cf(i,ns)*sqrt(ind(i)*ind(ns))
               totmut=totmut+mutual
 10         continue

            pnalty=(ind(ns)+totmut)/mtgndr
         endif

      endif
c     ========================================
c     en1,curr1,en2,curr2 ARE THE INITIAL AND FINAL ENERGIES AND CURRENTS.
c     ========================================
      totefy=(en2/en1)*pnalty

      totmlt=curr2/curr1
c     ========================================
c     TOTAL TRANSFER TIME IS NEEDED TO CALCULATE INSTANTANEOUS LOAD
c     POWER.  FIRST ELEMENT OF TIME ARRAY ONLY ADDED IF THERE WAS
c     DECOMPRESSION.
c     ========================================
      time=0.0
      do 20,j=2,ns
         time=time+trtime(j)
 20   continue

      if (decomp) then
         time=time+trtime(1)
      endif

      power=(en2*pnalty)/time

      return
      end

c     ========================================
```

```
c     "pdatal" ("print_data1") GIVES A REMINDER OF THE MOST IMPORTANT
c     DATA BEFORE THE RESULTS ARE PRINTED.
c
c     CALLING ROUTINE: calc1
c     ========================================
      subroutine pdatal (fwd,uload,ns)

      implicit double precision (a-h,o-z)
      double precision ind
      logical fwd
      character*40 fname

      parameter(maxns=50)

      common/coil1/ind(maxns),cf(maxns,maxns)

      write(0,*)
      write(0,*)
      print*,"* * * * * * * * * * * * * * * * * * *"
      write(0,*)
      write(0,*)

      inquire(1,name=fname)

      print*,"Datafile: ",fname

      if (fwd) then
         print*,"Forward operation"
      else
         print*,"Reverse operation"
      endif

      if (uload.eq.0.0) then
         write(0,100) ind(ns)*1E6
 100     format(/" Coupled load=",f9.2," microhenries")
      else
         write(0,200) uload*1E6
 200     format(/" Uncoupled load=",f9.2," microhenries")
      endif

      return
      end

c     ========================================
c     "rslt11" PRINTS THE RESULTS GENERATED BY "fwd11" AND "misc1".
c
c     CALLING ROUTINE: calc1
c     ========================================
      subroutine rslt11 (ns,decomp,curr,cmult,en,stpefy,pnalty,totefy,
     +                   totmlt,power)

      implicit double precision (a-h,o-z)
      logical decomp

      parameter(maxns=50,nlines=22)

      dimension curr(maxns+1),cmult(maxns),en(maxns+1),stpefy(maxns+1)

      open(0,defer=.false.)

      write(0,*)
      print*,"(A) CURRENT/ENERGY/EFFICIENCY:"
c     ========================================
c     TABLE HEADING
```

```
c     ========================================
      write(0,100)
100   format(/"Step",5x,"New Current(A)",2x,"Current Mult.",2x,
     +       "New Energy(J)",4x,"Step Efficiency(%)")
c     ========================================
c     INITIAL CONDITIONS
c     ========================================
      write(0,200) curr(1),en(1)
200   format(2x,"0",7x,f11.2,11x,"-",5x,f15.6,13x,"-")
c     ========================================
c     DECOMPRESSION
c     ========================================
      if (decomp) then
         write(0,300) curr(2),cmult(1),en(2),stpefy(1)*100.0
300      format("Decomp.",3x,f11.2,2x,f11.2,4x,f15.6,11x,f5.1)
      endif
c     ========================================
c     TRANSFERS
c     ========================================
      do 10,i=1,ns-1
c        ========================================
c        CHECK IF NEW HEADING NEEDED
c        ========================================
         if (mod(i,nlines).eq.0) write(0,100)

         write(0,400) i,curr(i+2),cmult(i+1),en(i+2),stpefy(i+1)*100.0
400      format(i3,7x,f11.2,2x,f11.2,4x,f15.6,11x,f5.1)

10    continue

      write(0,500) pnalty*100.0
500   format(/"Percent of final energy in load=",f5.1,"%")

      write(0,600) totmlt
600   format(/"Total Current Multiplication: x",f7.1)

      nefy=anint(totefy*100.0)
      write(0,700) nefy
700   format(/"TOTAL EFFICIENCY: ",i3,"%")

      write(0,800) power*1E-3
800   format(/"INSTANTANEOUS LOAD POWER: ",f10.3,"kW")

      open(0,defer=.true.)

      return
      end

c     ========================================
c     "rslt12" PRINTS THE TRANSFER DYNAMICS RESULTS, CALLING A
c     SEPARATE ROUTINE (res121) FOR THE INDUCED VOLTAGES.
c
c     CALLING ROUTINE: calc1
c     ========================================
      subroutine rslt12 (ns,decomp,fwd,swvolt,loadv,trtime,indced)

      implicit double precision (a-h,o-z)
      double precision loadv,indced
      logical decomp,fwd

      parameter(maxns=50,nlines=22)

      dimension swvolt(maxns),loadv(maxns),trtime(maxns),
     +          indced(maxns-2,maxns-2)
```

```
      open(0,defer=.false.)

      write(0,*)
      write(0,*)
      print*,"(B) TRANSFER DYNAMICS:"
      print*,"(B)(i) Switching Voltages & Times:"
c     ========================================
c     TABLE HEADING
c     ========================================
      write(0,100)
100   format(/"Step",4x,"Load Voltage(kV)",2x,"Switch Voltage(kV)",3x,
     +       "Transfer Time(microseconds)")
c     ========================================
c     DECOMPRESSION
c     ========================================
      if (decomp) then
         write(0,200) loadv(1)*1E-3,swvolt(1)*1E-3,trtime(1)*1E6
200      format("Decomp.",3x,f9.3,9x,f9.3,15x,f14.5)
      endif
c     ========================================
c     TRANSFERS
c     ========================================
      do 10 i=1,ns-1
c        ========================================
c        NEW HEADING?
c        ========================================
         if (mod(i,nlines).eq.0) write(0,100)

         write(0,300) i,loadv(i+1)*1E-3,swvolt(i+1)*1E-3,
     +                trtime(i+1)*1E6
300      format(i3,5x,f11.5,9x,f9.3,15x,f14.5)

10    continue

      write(0,*)
      print*,"(B)(ii) Induced Voltages(kV):"

      call res121 (ns,fwd,indced)

      open(0,defer=.true.)

      return
      end

c     ========================================
c     "calc21" DEALS WITH A USER-DIRECTED TOTAL INDUCTANCE CALC..
c
c     CALLING ROUTINE: calc2
c     ========================================
      subroutine calc21 (ns)

      implicit double precision(a-h,o-z)
      integer start,finish
c     ========================================
c     VALIDATE start,finish
c     ========================================
      write(0,*)
10    write(0,*)"No. of start inductance?  "
      read*,start

      if (start.lt.1.or.start.gt.ns) goto 10

20    write(0,*)"No. of finish inductance?  "
```

```
      read*,finish

      if (finish.lt.1.or.finish.gt.ns) goto 20
      if (start.gt.finish) goto 10
c     ========================================
c     RESULT
c     ========================================
      total=totind(start,finish)

      write(0,100) total*1E6
100   format(/"Total inductance is ",f10.3," microhenries"/)

      return
      end

c     ========================================
c     "calc22" DEALS WITH USER-DIRECTED COUPLING FACTOR CALCULATIONS.
c
c     CALLING ROUTINE: calc2
c     ========================================
      subroutine calc22

      implicit double precision(a-h,o-z)
c     ========================================
c     "L1" AND "L2" ARE UNRELATED TO SPECIFIC DATA FILES - THEY ARE JUST
c     GENERAL AS IN
c                 k12=(L12-L1-L2)/(2*sqrt[L1.L2]).
c     ========================================
      write(0,*)
10    write(0,*)"L1="
      read*,one
      write(0,*)"L2="
      read*,two
      write(0,*)"Ltot="
      read*,total
c     ========================================
c     DATA IS ONLY VALIDATED TO PREVENT CRASH DUE TO NEGATIVE OR ZERO
c     SQUARE ROOT, I.E. YOU CAN STILL GET A NONSENSE ANSWER.
c     ========================================
      if (one.le.0.0.or.two.le.0.0) goto 10

      cfact=(total-one-two)/(2*sqrt(one*two))

      write(0,100) cfact
100   format(/"Coupling factor is ",f4.2,/)

      return
      end

c     ========================================
c     "calc23" CALCULATES THE FOLLOWING, AS APPLIED TO THE FIRST PHASE OF
c     BACK CLAMPING:
c
c     rate      di1/dt
c     ratio2    (value of i2 when i1=0)/(i1max)
c     ratio3    i3max/i1max
c
c     DICTIONARY:
c
c     temp1-3,x,y,top,bottom    temporary variables used in calc. of "rate"
c     c1-c10,a-e                temporary variables used in calc. of "ratio2"
c
c
c     CALLING ROUTINE: calc2
```

```
c     ========================================
      subroutine calc23

      implicit double precision(a-h,o-z)
      double precision k12,k13,k23
c     ========================================
c     INPUT DATA AND VALIDATE (TO AN EXTENT).
c     ========================================
      write(0,*)
      print*,"Enter voltages in volts, inductances in microhenries:"
 10   write(0,*)
      write(0,*)"Vback_clamp="
      read*,vback
      write(0,*)"Vswitch="
      read*,vswtch
      if (vback.lt.0.0.or.vswtch.lt.0.0) goto 10

 20   write(0,*)
      write(0,*)"L1="
      read*,one
      write(0,*)"L2="
      read*,two
      write(0,*)"L3="
      read*,three
      if (one.le.0.0.or.two.le.0.0.or.three.le.0.0) goto 20

 30   write(0,*)
      write(0,*)"k12="
      read*,k12
      write(0,*)"k13="
      read*,k13
      write(0,*)"k23="
      read*,k23
      if (k12.lt.0.0.or.k13.lt.0.0.or.k23.lt.0.0) goto 30
      if (k12.gt.1.0.or.k13.gt.1.0.or.k23.gt.1.0) goto 30
c     ========================================
c     NORMALISE
c     ========================================
      one=one*1E-6
      two=two*1E-6
      three=three*1E-6
c     ========================================
c     CALCULATE rate & ratio3
c     ========================================
      temp1=(sqrt(one*three))*(k13-(k12*k23))
      temp2=one*(1-(k12*k12))
      temp3=three*(1-(k23*k23))

      x=temp1+temp2
      y=(2.0*temp1)+temp2+temp3

      top=(vback*x)-(vswtch*y)
      bottom=(one*three)*((2.0*k12*k13*k23)-(k13*k13)-(k12*k12)
     +        -(k23*k23)+1.0)

      rate=top/bottom
      ratio3=((vback*one*(1-(k12*k12)))-(vswtch*x))/top
c     ========================================
c     CALCULATE ratio2
c     ========================================
      c1=two  *three
      c2=three*sqrt(one*two)
      c3=two  *sqrt(one*three)
      c4=one  *sqrt(two*three)
```

```
      c5=one  *two

      c6 =1   -(k12*k12)    -(k23*k23)      +(k12*k12*k23*k23)
      c7 =k12 -(k13*k23)    -(k12*k12*k12)  +(k12*k12*k13*k23)
      c8 =k13 -(k12*k23)    -(k12*k12*k13)  +(k12*k12*k12*k23)
      c9 =k23 -(k12*k13)    -(k12*k12*k23)  +(k12*k12*k12*k13)
      c10=1   -(2.0*k12*k12)                +(k12*k12*k12*k12)

      a=c1*c6
      b=c2*c7
      c=c3*c8
      d=c4*c9
      e=c5*c10

      ratio2=-((vback*(d-c))+(vswtch*(a+b+c-d)))/
     +        ((vback*(c+e))-(vswtch*((2.0*c)+a+e)))
c     ========================================
c     RESULTS
c     ========================================
      print*,"* * * * * * * * * * * * * * *"

      write(0,100) rate*1E-6
100   format(/"di1/dt=",f12.5," amps/microsecond")
      write(0,200) ratio2
200   format(/"(i2 at i1=0)/i1max=",f7.4,/)
      write(0,300) ratio3
300   format("(i3 at i1=0)/i1max=",f7.4,/)

      print*,"* * * * * * * * * * * * * * *"

      return
      end


c     **** LEVEL 5 ROUTINES ****


c     ========================================
c     "cldmut" ("coupled_load_mutual_inductances") GENERATES AN ARRAY
c     CONTAINING, FOR EACH TRANSFER EXCEPT THE LAST, THE TWO MUTUAL
c     INDUCTANCES NECESSARY TO CALCULATE THE VOLTAGE ON A COUPLED LOAD
c     (THE LAST SECTION OF THE MEATGRINDER COIL).  THESE ARE:
c
c     m13: BETWEEN THE INDUCTANCE BEING SWITCHED OUT AND THE LOAD
c     m23: BETWEEN THE REMAINING IN-CIRCUIT INDUCTANCE AND THE LOAD
c
c     m23 IS FOUND FROM THE TOTAL IN-CIRCUIT INDUCTANCE AND m13 FROM THE
c     TOTAL OF THE THREE INDUCTANCES INVOLVED (m12 IS A REQUIRED
c     INTERMEDIATE VALUE).
c
c     (NOTE: VALUES ARE NOT PRODUCED FOR THE LAST STEP BECAUSE BY DEFINITION
c     vload=0.  SINCE, HOWEVER, m13 IS NOT ZERO DURING THIS STEP, THIS LEADS
c     TO A NON-ZERO VALUE FOR dltflx IN "fwd121".  THIS DOES NOT ACTUALLY
c     MATTER BECAUSE THIS VALUE OF dltflx IS NOT USED IN ANY CALCULATION.)
c
c     CALLING ROUTINE: fwd12
c     ========================================
      subroutine cldmut (ns,cumind,mutld)

      implicit double precision (a-h,o-z)
      double precision ind,mutld,ltwo,m23,m12,m13

      parameter(maxns=50)

```

```
      common/coil1/ind(maxns),cf(maxns,maxns)

      dimension cumind(maxns),mutld(2,maxns-2)

      do 10,i=1,ns-2
         ltwo=totind(i+1,ns-1)

         m23=(cumind(i+1)-ltwo-ind(ns))/2
         m12=(totind(i,ns-1)-ind(i)-ltwo)/2

         m13=(cumind(i)-ind(i)-ltwo-ind(ns)-(2*(m12+m23)))/2

         mutld(1,i)=m13
         mutld(2,i)=m23
 10   continue

      return
      end

c     ========================================
c     "fwd121" WORKS OUT,FOR EACH TRANSFER, THE EFFECTIVE INDUCTANCE,
c     APPROPRIATE VOLTAGE AND TRANSFER TIME.  THE TIME IS WORKED OUT
c     FROM WHICHEVER VOLTAGE IS GIVEN AND THE OTHER VOLTAGE IS THEN
c     CALCULATED FROM THE TIME.
c
c     ONCE THE EFFECTIVE INDUCTANCE HAS BEEN CALCULATED, THE BASIC
c     DIVISION IN THE STRUCTURE IS BETWEEN A MAXIMUM LOAD VOLTAGE
c     DETERMINING THE SWITCH VOLTAGE AND A SPECIFIED SWITCH VOLTAGE
c     DETERMINING THE LOAD VOLTAGE.  WITHIN THOSE TWO CASES THE LOAD
c     MAY BE UNCOUPLED OR COUPLED; IN THE LATTER CASE, THE LAST LOAD VOLTAGE
c     MUST BE ZERO, EVEN IF THE USER HAS ASKED FOR HIS SPECIFIED
c     MAXIMUM TO BE USED.
c
c     N.B. THE CHANGE OF CURRENT EXPERIENCED BY THE OPENING SWITCH IS
c     EQUAL TO THE INITIAL CURRENT FOR THAT STEP, I.E. curr(i+1),
c     WHEREAS THAT EXPERIENCED BY ANY LOAD IS "deltai",
c     I.E. curr(i+2)-curr(i+1).
c
c     CALLING ROUTINE: fwd12
c     ========================================
      subroutine fwd121 (ns,ldvfix,uload,curr,cumind,transk,mutld,
     +                   effind,swvolt,loadv,trtime)

      implicit double precision (a-h,o-z)
      double precision mutld,loadv,ind
      logical ldvfix

      parameter(maxns=50)

      common/coil1/ind(maxns),cf(maxns,maxns)
      common/volt1/vs(maxns),vload

      dimension curr(maxns+1),cumind(maxns),transk(maxns-1),
     +          mutld(2,maxns-2),effind(maxns),swvolt(maxns),
     +          loadv(maxns),trtime(maxns)

      do 10,i=1,ns-1

         deltai=curr(i+2)-curr(i+1)

c        ========================================
c        dltflx (delta_flux) IS THE FLUX-LINKAGE CHANGE EXPERIENCED BY
c        A COUPLED LOAD.  THE SAME PRINCIPLE IS USED FOR THE UNCOUPLED
c        LOAD, EXCEPT THAT THE EXPRESSION BECOMES THE SIMPLE
```

```
c          v=Ldi/dt.
c
c          dltflx CAN BE POSITIVE OR NEGATIVE.  A POSITIVE VALUE IS
c          REGARDED AS CORRESPONDING TO A "POSITIVE" LOAD VOLTAGE. I.E. ONE
c          IN THE SAME DIRECTION AS THAT ALWAYS EXPERIENCED BY AN
c          UNCOUPLED LOAD.  NOTE. HOWEVER. THAT SUCH A "POSITIVE" VOLTAGE
c          ACTUALLY MEANS OPPOSING CURRENT FLOW.I.E. THE NON-GROUND END
c          GOES NEGATIVE W.R.T. GROUND IN THE NORMAL CIRCUIT CONFIGURATION.
c
c          ========================================
           if (uload.eq.0.0) then
              cfact=transk(i)
c             ========================================
c             THE VALUE OF dltflx PRODUCED FOR THE LAST STEP IS NOT USED
c           . IN CALCULATION.
c             ========================================
              dltflx=(curr(i+2)*(ind(ns)+mutld(2,i)))-(curr(i+1)*(ind(ns)+
     +                                      mutld(2,i)+mutld(1,i)))

           else
              cfact=transk(i)*sqrt(cumind(i+1)/(cumind(i+1)+uload))
           endif

           effind(i+1)=ind(i)*(1-(cfact*cfact))

c          ========================================
c          MAIN "IF" BLOCK.
c          ========================================
           if (ldvfix) then

              if(uload.ne.0.0) then
                 trtime(i+1)=uload*deltai/vload
                 swvolt(i+1)=effind(i+1)*curr(i+1)/trtime(i+1)
              else
c             ========================================
c             LAST STEP IS DIFFERENT IF LOAD IS COUPLED
c             ========================================
                 if (i.ne.ns-1) then
c       ========================================
c       TIME CANNOT BE NEGATIVE BUT SIGN OF VOLTAGE INDICATES
c       DIRECTION.  OBVIOUSLY A NEGATIVE SWITCH VOLTAGE IS
c       MEANINGLESS. AND THIS THEREFORE HAS IMPLICATIONS
c       FOR FIXING THE VOLTAGE ON A COUPLED LOAD. SINCE THE
c       REAL INTENTION IS TO SPECIFY A MAXIMUM.
c       ========================================
                    time=dltflx/vload
                    trtime(i+1)=abs(time)
                    swvolt(i+1)=effind(i+1)*curr(i+1)/time
                 else
                    trtime(ns)=effind(ns)*curr(i+1)/vs(ns)
                    swvolt(ns)=vs(ns)
                    loadv(ns)=0.0
                 endif
              endif

           else if (.not.ldvfix) then

              trtime(i+1)=effind(i+1)*curr(i+1)/swvolt(i+1)

              if (uload.ne.0.0) then
                 loadv(i+1)=uload*deltai/trtime(i+1)
              else
                 if (i.ne.ns-1) then
c                ========================================
```

```
c             THIS QUANTITY CAN LEGITIMATELY BE NEGATIVE.
c             =======================================
                 loadv(i+1)=dltflx/trtime(i+1)
              else
                 loadv(ns)=0.0
              endif
           endif

        endif

 10   continue

      return
      end

c     =======================================
c     "fndcel" ("forward_induced_voltages1") CALCULATES THE INDUCED
c     VOLTAGE AT EACH NON-PARTICIPATIVE NODE (I.E. THOSE IN THE
c     SWITCHED OUT OR "BACK" LOOPS).  THE CORRECT DATA HAVING BEEN
c     PROVIDED BY OTHER ROUTINES, THE CALCULATION IS SIMPLE.
c
c     NOTE THAT TO GET THE NODE VOLTAGE W.R.T. GROUND, THE SWITCH
c     VOLTAGE MUST BE ADDED TO ACCOUNT FOR THE ELECTRICAL CONNECTION.
c
c     REFER ALSO TO COMMENTS FOR "fdadj1" ETC..
c
c     CALLING ROUTINE: fwd12
c     =======================================
      subroutine fndcel (ns,uload,cumind,transk,lthree,indvtk,swvolt,
     +                   indced)

      implicit double precision (a-h,o-z)
      double precision ind,lthree,indvtk,indced,k12,k31,k32

      parameter(maxns=50)

      common/coil1/ind(maxns),cf(maxns,maxns)

      dimension cumind(maxns),transk(maxns-1),lthree(maxns-2,maxns-2),
     +          indvtk(2,maxns-2,maxns-2),swvolt(maxns),
     +          indced(maxns-2,maxns-2)
c     =======================================
c     INITIALISE FOR PRINTING PURPOSES
c     =======================================
      do 10,i=1,ns-2
         do 10,j=1,ns-2
            indced(j,i)=0.0
 10   continue

      i=1
      j=1
      do 20,i=1,ns-2
         do 20,j=1,i

            if (uload.ne.0.0) then
               factor=sqrt(cumind(i+2)/(cumind(i+2)+uload))
               k12=transk(i+1)*factor
               k32=indvtk(2,j,i)*factor
            else
               k12=transk(i+1)
               k32=indvtk(2,j,i)
            endif

            k31=indvtk(1,j,i)
```

```
c              =======================================
c              REFER TO WRITTEN NOTES FOR FORMULA
c              =======================================
               a=sqrt(lthree(j,i)/ind(i+1))
               b=(k31-(k12*k32))/(1-(k12*k12))

               indced(j,i)=swvolt(i+2)*((a*b)+1)
 20      continue

         return
         end

c        ======================================
c        "res121" PRINTS THE INDUCED VOLTAGES FOR BOTH DIRECTIONS OF
c        OPERATION.  THE FOLLOWING SHOULD BE REMEMBERED:
c
c        1.  THE ARRAY IS TRIANGULAR, THE NO. OF NODES INCREASING WITH
c            STEP NO. IN FORWARD, DECREASING WITH STEP NO. IN REVERSE.
c
c        2.  THE VOLTAGE-INDUCING STEP NO. ("i") DIFFERS FROM THE ACTUAL
c            TRANSFER NO. BY ONE.
c
c        3.  FOR EACH DIFFERENT STEP, THE SAME NODE NO. CORRESPONDS
c            TO A DIFFERENT CIRCUIT JUNCTION; HENCE THE NEED FOR THE
c            BACKWARD COUNTING (I.E. BECAUSE, IN FWD, L1 IS ALWAYS
c            THE LAST NODE, AS OPPOSED TO ALWAYS BEING NODE 1).  THE
c            EXPRESSIONS USED TO GIVE THE CORRECT LABELS ALSO RESULT
c            FROM THIS FACT.
c
c        CALLING ROUTINE: rslt12
c        ======================================
         subroutine res121 (ns,fwd,indced)

         implicit double precision (a-h,o-z)
         double precision indced
         logical fwd

         parameter(maxns=50)

         dimension indced(maxns-2,maxns-2)

         if (fwd) then
            write(0,*)
            print*,"No induced voltages for Step 1"

            do 10,i=1,ns-2
               write(0,*)
               print*,"Step ",i+1
c              =======================================
c              "j" IS THE NODE NO..
c              =======================================
               do 10,j=i,1,-1

                  if (j.eq.i) then
                     print*,"L1:      ",indced(j,i)*1E-3
                  else
                     print*,"L",i-j,"/L",i-j+1,":    ",indced(j,i)*1E-3
                  endif
 10         continue
c        ======================================
c        REVERSE
c        ======================================
         else

```

```
          do 20,i=1,ns-2
             write(0,*)
             print*,"Step ",i

             do 20,j=(ns-1-i),1,-1

                if (j.eq.ns-1-i) then
                   print*,"L",ns,":       ",indced(j,i)*1E-3
                else
                   print*,"L",i+j+1,"/L",i+j+2,":    ",indced(j,i)*1E-3
                endif
 20       continue

          write(0,*)
          print*,"No induced voltages for step ",ns-1

       endif

       return
       end



c      ****  CALCULATION ROUTINES CALLED BY MORE THAN ONE LEVEL  ****


c      ========================================
c      "totind" TAKES AN INDUCTANCE ARRAY AND ITS CORRESPONDING
c      COUPLING FACTOR ARRAY AND CALCULATES THE TOTAL INDUCTANCE
c      OF ANY SUB-SET OF ADJACENT INDUCTANCES.  THE SUB-SET IS
c      DELINEATED BY "start" AND "finish", AND MAY CONSIST OF
c      A SINGLE INDUCTANCE.
c
c      CALLING ROUTINES: init1
c                        fdadj1
c                        fwd11
c                        fwd12
c                        cldmut
c
c
c      ========================================
       function totind (start,finish)

       implicit double precision (a-h,o-z)
       double precision ind,mutual
       integer start,finish

       parameter(maxns=50)

       common/coil1/ind(maxns),cf(maxns,maxns)

       totslf=0.0
       do 10,i=start,finish
          totslf=totslf+ind(i)
 10    continue
c      ========================================
c      THE LOOP TO CALCULATE MUTUALS IS NOT EXECUTED
c      IF start=finish.
c      ========================================
       totmut=0.0
       do 20,k=start,finish-1
          do 20,j=k+1,finish
             mutual=cf(k,j)*sqrt(ind(k)*ind(j))
             totmut=totmut+(2*mutual)
```

```
   20 continue

      totind=totslf+totmut

      return
      end

c     **** I/O ROUTINES (LEVEL A) ****

c     ========================================
c     "mesage" PRINTS AN INFORMATIVE MESSAGE ACCORDING TO THE
c     MESSAGE NUMBER.
c
c     CALLING ROUTINES: main program
c                       newcreate
c                       modexist
c     ========================================
      subroutine mesage(refnum,msgnum)
      integer refnum

      write(0,*)
      if (msgnum.eq.1) then
         write(0,*)
         write(0,*)
         print*,"        ++++++++++++++++++++++++++++++++++++++++++"
         print*,"        +  IDEAL MEATGRINDER CALCULATION PROGRAM  +"
         print*,"        +                                        +"
         print*,"        +  mgp  1.2  7.88                        +"
         print*,"        ++++++++++++++++++++++++++++++++++++++++++"
         write(0,*)
         write(0,*)
         write(0,*)

      else if (msgnum.eq.2) then
         print*,"+++ Option ",refnum," datafile created."

      else if (msgnum.eq.3) then
         print*,"+++ Option ",refnum," datafile modified."

      endif
      return
      end

c     ========================================
c     "menu" PRINTS A MENU ACCORDING TO THE MENU NUMBER, READS A CHOICE
c     AND RETURNS IT AS AN INTEGER.
c
c     CALLING ROUTINES: main program
c                       calc2
c     ========================================
      subroutine menu(mennum,choice)
      integer choice
      write(0,*)

      if (mennum.eq.1) then
         print*,"1  Automatic multistage calculation"
         print*,"2  User-directed expression evaluation"
         print*,"3  Stop"
   10    write(0,*)"Enter choice:  "
         read*,choice
         if (choice.lt.1.or.choice.gt.3), goto 10

      else if (mennum.eq.2) then
         print*,"1  Edit data"
```

```
         print*,"2  Run calculation"
         print*,"3  Return to main menu"
20       write(0,*)"Enter choice:  "
         read*,choice
         if (choice.lt.1.or.choice.gt.3) goto 20

      else if (mennum.eq.3) then
         print*,"1  Total inductance"
         print*,"2  Coupling factor"
         print*,"3  Back clamping analysis"
         print*,"4  Quit"
30       write(0,*)"Enter choice:  "
         read*,choice
         if (choice.lt.1.or.choice.gt.4) goto 30

      endif
      return
      end

c     ========================================
c     "menu2" ASKS A QUESTION ACCORDING TO THE MENU NUMBER, READS THE REPLY
c     AND RETURNS IT TO THE CALLING PROGRAM AS A LOGICAL VALUE.
c
c     CALLING ROUTINES: data
c                       rncalc (x2)
c                       calc1
c     ========================================
      subroutine menu2 (mennum,flag)

      logical flag
      character*1 reply

      write(0,*)
      if (mennum.eq.1) then
         write(0,*)"Create new datafile?  "
      else if (mennum.eq.2) then
         write(0,*)"Run again?  "
      else if (mennum.eq.3) then
         write(0,*)"New datafile?  "
      else if (mennum.eq.4) then
         write(0,*)"Forward operation?  "
      else if (mennum.eq.5) then
         write(0,*)"Decompression?  "
      else if (mennum.eq.6) then
         write(0,*)"Use max. load voltage?  "
      else if (mennum.eq.7) then
         write(0,*)"Review results?  "
      else if (mennum.eq.8) then
         write(0,*)"Write to results file?  "
      endif

      read*,reply

      if (reply.eq."y") then
         flag=.true.
      else
         flag=.false.
      endif

      return
      end

c     ========================================
c     "oldfle" ("get_old_file") READS A FILENAME AND CONNECTS TO
```

```
c     THE FILE. IT CHECKS FOR TWO ERROR CONDITIONS: OPENING ERROR
c     AND UNSUITABLE FILE. EITHER ERROR CAUSES THE REQUEST TO BE
c     REPEATED UNLESS THE MAXIMUM ERROR COUNT IS EXCEEDED, IN
c     WHICH CASE THE PROGRAM STOPS COMPLETELY.
c
c     VARIABLE DICTIONARY: opnerr   file_opening_error_indicator
c                          caller   routine which called oldfle
c                          wrong    wrong_file_type_flag
c                          mxfler   max_file_errors
c                          flopen   file_opening_error
c                          fltype   file_type_error
c
c     CALLING ROUTINES: rncalc
c                       mdexst
c     ========================================
      subroutine oldfle (caller,idnum)

      implicit double precision (a-h,o-z)
      parameter(maxerr=5)

      integer errors,opnerr,flopen,fltype
      double precision idnum
      logical wrong
      character caller*10,fname*20

      common/option/mnopt

      data mxfler,flopen,fltype/1,2,3/

      errors=0

10    write(0,*)
      write(0,*)"Datafile name?  "
      read*,fname
c     ========================================
c     OPEN FILE
c     ========================================
      open(1,file=fname,iostat=opnerr,status="old",form="unformatted")
      if (opnerr.ne.0) then
         errors=errors+1
         if (errors.eq.maxerr) then
            call errmsg (caller,1,mxfler)
            write(0,*)
            goto 50
         endif
         call errmsg (caller,1,flopen)
         goto 10
      endif
c     ========================================
c     VERIFY TYPE
c     ========================================
      read(1) idnum
      call chkfle (idnum,wrong)
      if (wrong) then
         errors=errors+1
         if (errors.eq.maxerr) then
            call errmsg (caller,2,mxfler)
            write(0,*)
            goto 50
         endif
         call errmsg (caller,1,fltype)
         close(1)
         goto 10
      endif
```

```
      return
50    open(0,defer=.false.)
      stop
      end
c     ========================================
c     "newfle" ("get_new_file") CREATES A NEW DATAFILE.  THE PROGRAM
c     STOPS IF THE MAX. NO. OF OPENING ERRORS IS EXCEEDED.
c
c     VARIABLE DICTIONARY: opnerr  file_open_error_indicator
c                          mxfler  max._file _errors
c                          flopen  file_open_error
c
c     CALLING ROUTINES: datnew
c     ========================================
      subroutine newfle (caller)

      implicit double precision (a-h,o-z)
      integer errors,opnerr,flopen
      character caller*10,fname*20

      parameter(maxerr=5)

      data mxfler,flopen/1,2/

      errors=0
c     ========================================
c     CONNECT TO FILE
c     ========================================
10    write(0,*)
      write(0,*)"Name of new datafile?  "
      read*,fname
      open(1,file=fname,iostat=opnerr,status="new",form="unformatted")

      if (opnerr.ne.0) then
         errors=errors+1
         if (errors.eq.maxerr) then
            call errmsg (caller,1,mxfler)
            write(0,*)
            goto 50
         endif
         call errmsg (caller,1,flopen)
         goto 10
      endif

      return
50    open(0,defer=.false.)
      stop
      end

c     **** I/O ROUTINES (LEVEL B) ****

c     ========================================
c     "errmsg" PRINTS AN ERROR MESSAGE IDENTIFIED BY THE INITIATING
c     ROUTINE AND A REF. NUMBER WHICH DISTINGUISHES BETWEEN
c     DIFFERENT OCCURENCES OF THE SAME ERROR WITHIN THAT ROUTINE.
c
c     CALLING ROUTINES: oldfle
c                       newfle
c     ========================================
      subroutine errmsg (caller,refnum,errnum)

      integer refnum,errnum
      character caller*10,text*50
```

```
      if (errnum.eq.1) then
         text="Max. file errors exceeded: program stop."
      else if (errnum.eq.2) then
         text="File open failed."
      else if (errnum.eq.3) then
         text="File unsuitable for this option."
      endif

      write(0,*)
      print*,"**** ",caller,refnum,"  ",text

      return
      end

c     ========================================
c     "chkfle" CHECKS THAT A DATAFILE IS SUITABLE FOR USE WITH THE
c     OPTION SELECTED.
c
c     "idnum" IS READ FROM THE FILE AND COMPARED TO THE EXPECTED
c     VALUE.  THESE VALUES ARE STORED IN THE ARRAY "ident".
c     ========================================
      subroutine chkfle (idnum,wrong)

      logical wrong
      double precision idnum,ident(10)

      common/option/mnopt

      data ident(1),ident(2)/1.1,1.1/

      if (idnum.ne.ident(mnopt)) then
         wrong=.true.
      else
         wrong=.false.
      endif

      return
      end
```

```
+++++++++++++++++++++++++++++++++++++++++
+  IDEAL MEATGRINDER CALCULATION PROGRAM  +
+                                         +
+  mgp  1.2  7.88                         +
+++++++++++++++++++++++++++++++++++++++++




1  Automatic multistage calculation
2  User-directed expression evaluation
3  Stop
Enter choice:  1

1  Edit data
2  Run calculation
3  Return to main menu
Enter choice:  1

Create new datafile?  n
Modify existing data:

Datafile name?  mg1


COIL INDUCTANCES (microhenries):

L1=403.0
L2=289.0
L3=147.0
L4=52.0
L5=24.0
L6=0.9
L7=11.5

Subscript of value to be changed (0 to quit)?  0


COUPLING FACTORS:

k(1_2)=0.85
k(1_3)=0.69
k(1_4)=0.58
k(1_5)=0.55
k(1_6)=0.29
k(1_7)=0.2
k(2_3)=0.84
k(2_4)=0.71
k(2_5)=0.68
k(2_6)=0.5
k(2_7)=0.44
k(3_4)=0.89
k(3_5)=0.8
k(3_6)=0.75
k(3_7)=0.7
k(4_5)=0.91
k(4_6)=0.74
k(4_7)=0.65
k(5_6)=0.76
k(5_7)=0.7
k(6_7)=0.71

Value to be changed:
1st subscript (0 to quit)?  0
```

```
CLAMP VOLTAGES (kV):

1  Vsw.decomp=1.0
2  Vsw1=0.55
3  Vsw2=0.15
4  Vsw3=0.15
5  Vsw4=0.15
6  Vsw5=0.15
7  Vsw6=0.15

Ref. no. of value to be changed (0 to quit)?  0

Max. load voltage: 5.0kV    Modify?  n

+++ Option 1 datafile modified.

1  Edit data
2  Run calculation
3  Return to main menu
Enter choice:  2

Datafile name?  mg1

Forward operation?  y

Uncoupled load (microhenries)?  0
Last section of meatgrinder coil will be taken as load.

Use max. load voltage?  n

Initial current (amps)?  7


* * * * * * * * * * * * * * *


Datafile: >udd>EL>MGPimperton>mg>mgideal>mg1
Forward operation

Coupled load=    11.50 microhenries


(A) CURRENT/ENERGY/EFFICIENCY:
```

Step	New Current(A)	Current Mult.	New Energy(J)	Step Efficiency(%)
0	7.00	-	0.081460	-
1	9.55	1.36	0.077068	94.6
2	14.31	1.50	0.071862	93.2
3	24.46	1.71	0.068627	95.5
4	42.67	1.74	0.064907	94.6
5	80.80	1.89	0.055390	85.3
6	96.85	1.20	0.053933	97.4

```
Percent of final energy in load=100.0%

Total Current Multiplication: x   13.8

TOTAL EFFICIENCY:  66%

INSTANTANEOUS LOAD POWER:      3.028kW
```

(B) TRANSFER DYNAMICS:

(B)(i) Switching Voltages & Times:

Step	Load Voltage(kV)	Switch Voltage(kV)	Transfer Time(microseconds)
1	0.06476	0.550	2.28116
2	0.01261	0.150	7.27140
3	0.00242	0.150	3.01475
4	0.03660	0.150	2.02844
5	0.00986	0.150	2.97368
6	0.00000	0.150	0.24041

(B)(ii) Induced Voltages(kV):

No induced voltages for Step 1

Step 2

L1: 0.317862384617421227

Step 3

L1: 0.643778829426506347

L1/L2: 0.37571145595952722

Step 4

L1: 0.920574403088797163

L1/L2: 0.565962050321608928

L2/L3: 0.325435708554778516

Step 5

L1: 1.45967735721775622

L1/L2: 0.92887476489132329

L2/L3: 0.544839254072142929

L3/L4: 0.342601097616620486

Step 6

L1: 4.14332887174678882

L1/L2: 3.196023202965606

L2/L3: 2.17917142311552343

L3/L4: 1.20113487031438651

L4/L5: 0.560806463155683678

Review results? n

Run again? y

New datafile? n

Forward operation? y

Uncoupled load (microhenries)? 10

Decompression? n

Use max. load voltage? n

Initial current (amps)? 7

* * * * * * * * * * * * * * *

Datafile: >udd>EL>MGPimperton>mg>mgideal>mg1
Forward operation

Uncoupled load=   10.00 microhenries

(A) CURRENT/ENERGY/EFFICIENCY:

Step	New Current(A)	Current Mult.	New Energy(J)	Step Efficiency(%)
0	7.00	-	0.081705	-
1	9.53	1.36	0.077281	94.6
2	14.22	1.49	0.071980	93.1
3	23.88	1.68	0.068298	94.9
4	39.48	1.65	0.063362	92.8
5	61.68	1.56	0.051299	81.0
6	68.23	1.11	0.050049	97.6

Percent of final energy in load= 46.5%

Total Current Multiplication: x   9.7

TOTAL EFFICIENCY: 28%

E.2 TWO-STEP MEATGRINDER EFFICIENCY INVESTIGATION

The program "mgeff_01" was written as part of the optimisation investigation described in Chapter 6.

The following pages contain a listing of the program.

Note: The Fortran programs were written with the aid of references [86] to [88].

```
c       ========================================
c       mgeff_01 INVESTIGATES THE EFFICIENCY OF A TWO-STEP IDEAL
c       MEATGRINDER  WITH AN UNCOUPLED LOAD AND NO DECOMPRESSION WITH
c       A VIEW TO FINDING OPTIMAL INDUCTANCE VALUES W.R.T. THE LOAD
c       INDUCTANCE.
c
c       SINCE THERE HAS NOT BEEN ANY OBVIOUS RIGOROUS MATHEMATICAL
c       SOLUTION, AN EXPERIMENTAL, STATISTICAL-TYPE INVESTIGATION WILL
c       BE DONE.  THIS PROGRAM IS THE FIRST STEP IN THE INVESTIGATION:
c       THERE MAY BE OTHER PROGRAMS- HENCE THE NUMBERING.
c
c       THE OVERALL CURRENT MULTIPLICATION x AND THE THREE COUPLING FACTORS
c       ARE FIXED, AND VARIOUS x1:x2 COMBINATIONS ARE TRIED.  WITHIN EACH SUCH
c       COMBINATION THE EFFICIENCY IS CALCULATED AS A FUNCTION OF r3 (L3/Lload).
c
c
c       VARIABLE DICTIONARY:
c
c       x       overall multiplication  I3/I1
c       x1      I2/I1
c       x2      I3/I2
c       r1      L1/Lload
c       r2      L2/Lload
c       r3      L3/Lload
c       r23     (L2+L3+2M23)/Lload =L23/Lload
c       cf123   coupling beteween L1 and L23
c       eff1    efficiency of first m/g step
c       eff2    (eff. of 2nd m/g step)*(uncoupled load penalty)
c       enld2   (load en. after 1st step)/(initial load en.)
c       enld3   (load en. after 2nd step)/(initial load en.)
c
c       mgp 15.9.88
c       ========================================
        program mgeff_01

        implicit double precision (a-h,o-z)
        double precision k12,k13,k23

        parameter (maxlin=21)

        data intro/1/
c       ========================================
c       OVERALL MULTIPLICATION AND k VALUES
c       ========================================
        data x,k12,k13,k23 /5.0,0.9,0.81,0.9/

        open(0,defer=.true.,prompt=.false.)

        call mesage(intro)
        call ofile(nsets)
c       ========================================
c       GET SWEEP LIMITS/STEP SIZES
c       ========================================
        call swplim(x,x2min,x2max,deltx2,r3min,r3max,deltr3)

        nrslts=0
c       ========================================
c       CALCULATION LOOPS.  N.B. UPPER LIMIT OF DO VARIABLE IS INCREASED
c       BY HALF THE INCREMENT TO PREVENT MISCALCULATION OF TRIP COUNT DUE
c       TO REAL ARITHMETIC ROUNDING ERRORS.
```

```
c
c     TABLE HEADING IS PRINTED EVERY TIME x2 CHANGES AND WHENEVER THERE
c     WOULD OTHERWISE BE NO HEADING VISIBLE ON THE SCREEN.
c     ======================================
      do 20,x2=x2min,(x2max+(deltx2/2.0)),deltx2
         x1=x/x2
         call headg(x1,x2)

         do 10,r3=r3min,(r3max+(deltr3/2.0)),deltr3

            nrslts=nrslts+1
            if (mod(nrslts,maxlin).eq.0) call headg(x1,x2)

            r2=ratio1(x2,k23,r3)

            call adjust(r2,r3,k12,k13,k23,
     c                  r23,cf123)

            r1=ratio1(x1,cf123,r23)

            eff1=effy1(x1,cf123,r23)
            eff2=effy2(x2,k23,r3)
            efftot=eff1*eff2
            enld2=x2*x2
            enld3=x*x
c           ========================================
c           RESULTS
c           ========================================
            write(0,100) r3,r2,r1,(eff1*100.0),(eff2*100.0),
     c                            (efftot*100.0),nrslts

100         format(2x,f9.3,2x,f9.3,2x,f9.3,3x,f5.1,3x,f5.1,4x,f5.1,6x,
     c             i5)
c           ========================================
c           GAPLOT REQUIRES SINGLE PRECISION DATA
c           ========================================
            write(1) sngl(r3),sngl(r2),sngl(r1),sngl(eff1*100.0),
     c               sngl(eff2*100.0),sngl(efftot*100.0)
c           ========================================
c           nsets IS FOR PLOTTING PURPOSES FROM THE OUTPUT FILE.
c           ========================================
            nsets=nsets+1

10       continue

20    continue

      rewind(1)
      write(1) nsets
      close(1)
      open(0,defer=.false.)

      stop
      end


c     ***  LEVEL 2 ROUTINES  ***


c     ========================================
```

```
c      mesage PRINTS A MESSAGE ACCORDING TO A MESSAGE NUMBER.
c
c      CALLING ROUTINES:  main program
c      ========================================
       subroutine mesage(msgnum)

       write(0,*)
       if (msgnum.eq.1) then
          write(0,*)
          write(0,*)
          print*,"        +++++++++++++++++++++++++++++++++++++++++++"
          print*,"        +                                         +"
          print*,"        +                 MGEFF_01                +"
          print*,"        +                                         +"
          print*,"        +  Optimisation investigation:            +"
          print*,"        +  Two-step ideal meatgrinder with        +"
          print*,"        +  uncoupled load and no decompression.   +"
          print*,"        +                                         +"
          print*,"        +   mod.1 28.9.88              mgp 9.88   +"
          print*,"        +++++++++++++++++++++++++++++++++++++++++++"
          write(0,*)
          write(0,*)
       endif

       return
       end

c      ========================================
c      ofile CONNECTS TO A BINARY RESULTS FILE.  nsets IS THE NO. OF SETS
c      OF DATA (REQUIRED BY K.G. PLOTTING PROGRAMS).
c
c      CALLING ROUTINE:  main program
c      ========================================
       subroutine ofile(nsets)

       integer opnerr,file
       character*20 filenm

       data file/1/

 10    write(0,*)
       write(0,*)"GAPLOT filename?    "
       read*,filenm

       open(1,file=filenm,iostat=opnerr,status="new",mode="out",
      c                  form="unformatted",binary stream=.true.)

       if (opnerr.ne.0) then
          call errmsg(file)
          goto 10
       endif

       write(1) 0,6,3
       nsets=0

       return
       end

c      ========================================
c      swplim READS THE LIMITS AND STEP SIZES FOR THE TWO LOOPS
```

```
c      INTERACTIVELY.  THE LIMITS OF x2 ARE ALWAYS 1 AND x.
c
c      CALLING ROUTINE:  main program
c      ========================================
       subroutine swplim(x,x2min,x2max,deltx2,r3min,r3max,deltr3)

       implicit double precision(a-h,o-z)

       write(0,*)
5      write(0,*)"X2 lower limit?   "
       read*,x2min
       if (x2min.le.1.0.or.x2min.ge.x) goto 5

6      write(0,*)"X2 upper limit?   "
       read*,x2max
       if (x2max.le.x2min.or.x2max.ge.x) goto 6

10     write(0,*)"X2 step size?     "
       read*,deltx2
       if (deltx2.le.0.0.or.deltx2.gt.(x2max-x2min)) goto 10

20     write(0,*)"r3 lower limit?   "
       read*,r3min
       if (r3min.le.0.0) goto 20

30     write(0,*)"r3 upper limit?   "
       read*,r3max
       if (r3max.le.r3min) goto 30

40     write(0,*)"r3 step size?     "
       read*,deltr3
       if (deltr3.le.0.0.or.deltr3.gt.(r3max-r3min)) goto 40

       return
       end

c      ========================================
c      headg REPRINTS THE VALUES OF X1 AND X2 AND A FRESH TABLE HEADING.
c
c      CALLING ROUTINE: main program
c      ========================================
       subroutine headg(x1,x2)

       implicit double precision(a-h,o-z)

       print*,"______________________________________________________________"

       write(0,100) x1,x2
100    format(/"X1=",f7.3,2x,"X2=",f7.3/)
       write(0,200)
200    format(8x,"r3",9x,"r2",9x,"r1",5x,"eff1",4x,"eff2",4x,"efftot",
     c        6x,"Result No.")

       return
       end

c      ========================================
c      adjust MAKES STEP 1 DATA SUITABLE FOR USE BY ratio1 TO FIND r1.
c
c      CALLING ROUTINE: main program
```

```
c     ========================================
      subroutine adjust(r2,r3,k12,k13,k23,
     c                          r23,cf123)

      implicit double precision(a-h,o-z)
      double precision k12,k13,k23

      r23=r2+r3+(2.0*k23*sqrt(r2*r3))

      cf123=((k12*sqrt(r2))+(k13*sqrt(r3)))/sqrt(r23)

      return
      end

c     ========================================
c     ratio1 CALCULATES "L1/L(LOAD)", WHERE L1 IS THE INDUCTANCE
c     SWITCHED OUT.
c
c     CALLING ROUTINE: main program
c     ========================================
      function ratio1(mult,cfact,indrat)

      implicit double precision(a-h,o-z)
      double precision mult,indrat

      ratio1=(((mult-1.0)**2.0)*((indrat+1.0)**2.0))/
     c                          (cfact*cfact*indrat)

      return
      end

c     ========================================
c     effy1 FINDS THE STEP EFFICIENCY, NOT INCLUDING THE UNCOUPLED
c     LOAD PENALTY.
c
c     CALLING ROUTINE: main program
c     ========================================
      function effy1 (mult,cfact,indrat)

      implicit double precision (a-h,o-z)
      double precision mult,indrat,mltsqd

      mltsqd=mult*mult
      factor=1.0-(cfact*cfact)

      top=indrat*cfact*cfact*mltsqd

      temp1=mltsqd-(2.0*mult*factor)+factor
      temp2=mltsqd-(2.0*mult)+1.0

      bottom=(indrat*temp1)+temp2

      effy1=top/bottom

      return
      end

c     ========================================
c     effy2 CALCULATES THE STEP EFFICIENCY, INCLUDING THE UNCOUPLED
c     LOAD PENALTY.
```

```
c
c     CALLING ROUTINE: main program
c     ======================================
      function effy2 (mult,cfact,indrat)

      implicit double precision(a-h,o-z)
      double precision mult,mltsqd,indrat

      mltsqd=mult*mult
      factor=1.0-(cfact*cfact)

      top=indrat*cfact*cfact*mltsqd

      temp1=mltsqd-(2.0*mult*factor)+factor
      temp2=(2.0*mltsqd)-(2.0*mult*factor)-(2.0*mult)+factor+1.0
      temp3=mltsqd-(2.0*mult)+1.0

      bottom=(indrat*indrat*temp1)+(indrat*temp2)+temp3

      effy2=top/bottom

      return
      end


c     ***  LEVEL 3 ROUTINES  ***


c     ======================================
c     errmsg PRINTS AN ERROR MESSAGE ACCORDING TO AN ERROR NUMBER.
c
c     CALLING ROUTINE: ofile
c     ======================================
      subroutine errmsg (errnum)

      integer errnum

      if (errnum.eq.1) then
         print*,"*** File open failed"
      endif

      return
      end
```

APPENDIX F

USE OF COMPUTER ALGEBRA PACKAGE

F.1 INTRODUCTION

This Appendix relates to the investigation of ETAC described in Chapter 5.

The requirement is to determine the total current increase in the final meatgrinder coil section. This is done by adding the results of equations (5.21) and (5.26), which are reproduced below as equations (F.1) and (F.2).

For phase one:

$$\delta i_2 = \left[\frac{V_{S3}[M-L] + V_{S1}[J+K+L-M]}{V_{S3}[L+N] - V_{S1}[2L+J+N]} \right] I_1 \qquad \text{(F.1)}$$

where

$$J = L_2 L_3 \quad (1 - k_{12}^2 - k_{23}^2 + k_{12}^2 k_{23}^2)$$

$$K = L_3 \sqrt{L_1 L_2} \quad (k_{12} - k_{13} k_{23} - k_{12}^3 + k_{12}^2 k_{13} k_{23})$$

$$L = L_2 \sqrt{L_1 L_3} \quad (k_{13} - k_{12} k_{23} - k_{12}^2 k_{13} + k_{12}^3 k_{23})$$

$$M = L_1 \sqrt{L_2 L_3} \quad (k_{23} - k_{12} k_{13} - k_{12}^2 k_{23} + k_{12}^3 k_{13})$$

$$N = L_1 L_2 \quad (1 + k_{12}^4 - 2k_{12}^2)$$

For phase two:

$$\delta i_2 = -\frac{L_2 + M_{12} + M_{23}}{L_2} I_3 \tag{F.2}$$

Equation (F.2) may be expressed in terms of I_1 by using equation (5.22) to substitute for I_3. The result is

$$\delta i_2 = -\left[\frac{L_2 + M_{12} + M_{23}}{L_2}\right]\left[\frac{V_{S3}L_1(1 - k_{12}^2) - V_{S1}X}{V_{S3}X - V_{S1}Y}\right] I_1 \tag{F.3}$$

where

$$X = \sqrt{L_1L_3}(k_{13} - k_{12}k_{23}) + L_1(1 - k_{12}^2)$$

$$Y = 2\sqrt{L_1L_3}(k_{13} - k_{12}k_{23}) + L_1(1 - k_{12}^2) + L_3(1 - k_{23}^2)$$

F.2 EVALUATION BY HAND

This section gives a sample of the working required to add equations (F.1) and (F.3). The exercise can be regarded as adding two fractions a/b and c/d. The result is

$$\frac{a}{b} + \frac{c}{d} = \frac{ad + bc}{bd} \tag{F.4}$$

If equations (F.1) and (F.3) are divided by I_1 on both sides, then the right-hand sides both have the form

$$\frac{V_{S3}\theta_1 + V_{S1}\theta_2}{V_{S3}\theta_3 + V_{S1}\theta_4}$$

Therefore each of the three products given by equation (F.4) has the form

$$V_{S3}^2\theta_5 + V_{S3}V_{S1}\theta_6 + V_{S1}^2\theta_7$$

and there is a total of nine coefficients to evaluate.

Consider the product "bd". This is

$$bd = (V_{S3}[L+N] - V_{S1}[2L+J+N])\ (V_{S3}L_2X - V_{S1}L_2Y) \qquad (F.5)$$

Thus

$$bd = V_{S3}^2[L+N]L_2X - V_{S3}V_{S1}(L_2Y[L+N] + L_2X[2L+J+N]) + V_{S1}^2L_2Y[2L+J+N] \qquad (F.6)$$

The coefficient of V_{S3}^2 in equation (F.6) has four terms.

First term:

$$L_1L_2L_3(k_{13}^2 - 2k_{12}k_{13}k_{23} - k_{12}^2k_{13}^2 + k_{12}^2k_{23}^2 - k_{12}^4k_{23}^2 + 2k_{12}^3k_{13}k_{23})$$

Second term:

$$L_1 L_2 \sqrt{L_1 L_3}(k_{13} + k_{12}^4 k_{13} - 2k_{12}^2 k_{13} - k_{12} k_{23} - k_{12}^5 k_{23} + 2k_{12}^3 k_{23})$$

Third term:

$$L_1 L_2 \sqrt{L_1 L_3}(k_{13} - k_{12} k_{23} - 2k_{12}^2 k_{13} + 2k_{12}^3 k_{23} + k_{12}^4 k_{13} - k_{12}^5 k_{23})$$

Fourth term:

$$L_1^2 L_2 (1 - 3k_{12}^2 + k_{12}^4 - k_{12}^6)$$

Adding the above four terms the V_{S3}^2 coefficient is

$$L_1 L_2^2 \{ L_3 [k_{13}^2 - 2k_{12} k_{13} k_{23} - k_{12}^2 k_{13}^2 + k_{12}^2 k_{23}^2 + 2k_{12}^3 k_{13} k_{23} - k_{12}^4 k_{23}^2] + 2\sqrt{L_1 L_3}[k_{13} + k_{12}^4 k_{13} - 2k_{12}^2 k_{13} - k_{12} k_{23} - k_{12}^5 k_{23} + 2k_{12}^3 k_{23}] + L_1 [1 - 3k_{12}^2 + 3k_{12}^4 - k_{12}^6] \}$$

Similarly, the coefficient of $V_{S3} V_{S1}$ is the sum of six terms and that of V_{S1}^2 the sum of nine terms.

This process is repeated for the two products "ad" and "bc", which are then added to form the numerator of the final result.

The final stage is to compare the numerator and denominator, extracting and cancelling any common factors. This is again tackled in three stages, considering the coefficients of V_{S3}^2, $V_{S3}V_{S1}$ and V_{S1}^2. In each case the ratio is the same and so the final result is

$$\frac{\delta i_2}{I_1} = - \left[\frac{L_2 + M_{12}}{L_2} \right] \tag{F.7}$$

F.3 USE OF "REDUCE" COMPUTER ALGEBRA PACKAGE

Computer programs exist for manipulating algebraic expressions [89]. One such program was used to carry out the task described above.

The REDUCE computer algebra package [90,91] was used on an Amdahl 5890 mainframe to generate the output given at the end of this Appendix. The input file is listed first, the expressions J, K, L, M, N, X and Y corresponding to those defined in Chapter 5. The expressions INCR1 and -INCR2 correspond tq equations (F.1) and (F.3) divided by I_1.

The first three lines (L1: = LL1**2 etc.) are necessary because REDUCE does not simplify square roots very well [92]. These lines define LL1, LL2 and LL3 to be the square roots without actually using the square root function.

The results produced come after the three lines

```
INCR1;
INCR2;
INCR1-INCR2;
```

and it can be seen that if the final result is multiplied top and bottom by LL2 (i.e. $\sqrt{L_2}$) the result is identical to equation (F.7).

```
0NON-CONTIGUOUS.
 SLISP : 1466352 BYTES
-
1
     REDUCE 3.3, 15-Jan-88 ...
     L1:=LL1**2$

     L2:=LL2**2$

     L3:=LL3**2$


     X:=   (LL1*LL3) * (K13 - K12*K23) + L1*(1 - K12**2)$

     Y :=2*(LL1*LL3) * (K13 - K12*K23) + L1*(1 - K12**2) + L3*(1 - K23**2)$


     J:=L2*L3*              (1    - K12**2   - K23**2        + K12**2 * K23**2)$

     K:=L3 * (LL1*LL2) * (K12 - K13*K23 - K12**3         + K12**2 * K13*K23)$

     L:=L2 * (LL1*LL3) * (K13 - K12*K23 - K12**2 * K13 + K12**3 * K23)$

     M:=L1 * (LL2*LL3) * (K23 - K12*K13 - K12**2 * K23 + K12**3 * K13)$


     N:=L1*L2 * (1 + K12**4 - 2*K12**2)$



     INCR1TOP   :=VB * (M-L) + VSW * (J   + K+L-M)$

     INCR1BOT   :=VB * (L+N) - VSW * (2*L + J+N)$

     INCR2TOP1:=L2 + K12*(LL1*LL2) + K23*(LL2*LL3)  $

     INCR2TOP2:=VB*L1 * (1 - K12**2) - VSW*X  $

     INCR2BOT1:=L2  $

     INCR2BOT2:=VB*X - VSW*Y  $


     INCR1:=INCR1TOP / INCR1BOT  $

     INCR2:=(INCR2TOP1 / INCR2BOT1) * (INCR2TOP2 / INCR2BOT2) $


     ON EXP;

     ON GCD;

     INCR1;

                      2
  - (LL3*(VSW*K23 *LL3*LL2 + VSW*K23*K12*LL2*LL1 + VSW*K23*K13*LL3*LL1 + VSW


           VSW*K12*LL3*LL1 - VSW*K13*LL2*LL1 - VSW*LL3*LL2 - VB*K23*K12*LL2*
```

```
           2                                  2    2
     *LL1   + VB*K13*LL2*LL1))/(LL2*(VSW*K23 *LL3   + 2*VSW*K23*K12*LL3*L

                      2           2                                   2    2
    *LL3*LL1 - VSW*LL3   - VSW*LL1   - VB*K23*K12*LL3*LL1 - VB*K12 *LL1   +

INCR2;

           2         2                  2
(LL1*(VSW*K23 *K12*LL3   + 2*VSW*K23*K12 *LL3*LL1 + VSW*K23*K12*LL3*LL2 - VS

                 3    2           2
    LL1 + VSW*K12 *LL1   + VSW*K12 *LL2*LL1 - VSW*K12*K13*LL3*LL1 - VSW*K1

                       2                                     3    2
    *LL2*LL1 - VB*K23*K12 *LL3*LL1 + VB*K23*LL3*LL1 - VB*K12 *LL1   - VB*K

                           2    2                                     2
    *LL2*LL1))/(LL2*(VSW*K23 *LL3   + 2*VSW*K23*K12*LL3*LL1 + VSW*K12 *LL1

       2          2                                 2    2
    LL3   - VSW*LL1   - VB*K23*K12*LL3*LL1 - VB*K12 *LL1   + VB*K13*LL3*LL1

INCR1-INCR2;

    K12*LL1 + LL2
 - ---------------
        LL2

QUIT;
*** END OF RUN
```

APPENDIX G

CONFERENCE PUBLICATION

"Optimum Design Criteria For a Single-Step Meatgrinder"

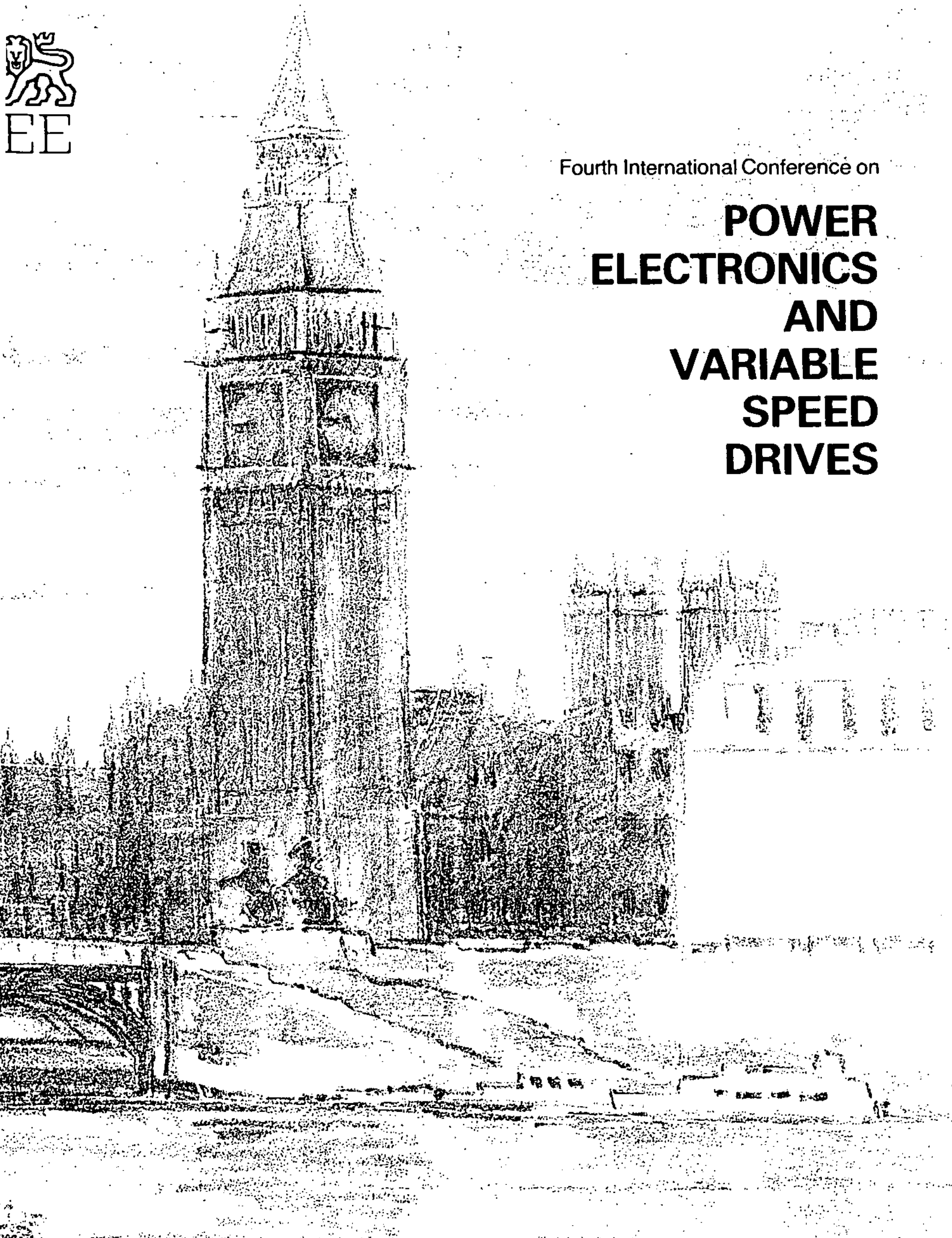
by

M G Pimperton, V V Vadher and I R Smith

Presented at the Fourth IEE International Conference on Power Electronics and Variable Speed Drives, Savoy Place, London, 17-19 July 1990.

Fourth International Conference on

POWER ELECTRONICS AND VARIABLE SPEED DRIVES



Conference Publication Number 324

Fourth International Conference on

Power Electronics and Variable-Speed Drives

17-19 July 1990

Organised by the
Power Division of the Institution of Electrical Engineers

in association with the
Institute of Electrical Engineers of Japan
Institute of Electrical and Electronics Engineers Inc
(Industry Applications Society & Industrial Electronics Society)
Institution of Engineers of Australia
Institution of Engineers of Ireland
Société des Electriciens et des Electroniciens
South African Institute of Electrical Engineers

Venue
The Institution of Electrical Engineers, Savoy Place, London WC2, UK

OPTIMUM DESIGN CRITERIA FOR A SINGLE-STEP MEATGRINDER

M.G. Pimperton, V V Vadher and I R Smith

Loughborough University of Technology, UK

INTRODUCTION

The principle behind the meatgrinder is that of dividing a storage inductor into many mutually-coupled sections; such that switching out each section in turn enables high efficiency energy transfer to be achieved [1,2]. In fact, as the number of coil sections approaches infinity the circuit efficiency approaches 100% [1].

Giorgi et al [3] have demonstrated the soundness of this principal with a five-section coil. Experimental results have been obtained at Loughborough which confirm the theory. Designs for multi-step meatgrinders for specific applications have also been proposed [4,5,6].

If the meatgrinder coil is divided into just two mutually-coupled sections the circuit is simplified and becomes easier to analyse. Such a circuit has been used to further demonstrate the validity of the meatgrinder theory [7]. It has also been recognised, however, that a single-step meatgrinder may have value in its own right in some applications [8-10]. The study of a single step provides a logical starting point for a more general investigation into the design of multi-step circuits.

ANALYSIS OF UNLOADED SINGLE-STEP MEATGRINDER

Figure 1 shows an unloaded single-step meatgrinder. The initial current flows through L_1 and L_2 (figure 1(a)) and the transfer is effected by first closing S_2 (which has no effect on the current flow-figure 1(b)) and then opening S_1 (figure 1(c)). The current multiplication β is given by [2]

$$\beta = \frac{I_2}{I_1} = \frac{L_2+M}{L_2} \tag{1}$$

where $M = k\sqrt{L_1 L_2}$.

If the initial and final energies are E_1 and E_2 respectively then

$$E_1 = \frac{1}{2}(L_1+L_2+2M)I_1^2$$

and

$$E_2 = \frac{1}{2}L_2 I_2^2$$

Substituting for I_2 from equation (1), the efficiency of the step, η_s is

$$\eta_s = \frac{E_2}{E_1} = \frac{1+k^2\alpha+2k\alpha}{1+\alpha+2k\alpha} = \frac{(1+k\sqrt{\alpha})^2}{1+\alpha+2k\alpha} \tag{2}$$

where $\alpha = L_1/L_2$.

From equation (1) the inductance ratio α can be expressed as

$$\alpha = \frac{(\beta-1)^2}{k^2} \tag{3}$$

and substituting equation (3) into equation (2) yields

$$\eta_s = \frac{k^2\beta^2}{\beta^2+(k^2-1)(2\beta-1)} \tag{4}$$

ANALYSIS OF LOADED SINGLE-STEP MEATGRINDER

The addition of an uncoupled load inductance in series with L_2 (figure 2) does not affect the operation of the circuit. However, the final inductance becomes

$$L_2' = L_2+L_{LOAD} \tag{5}$$

and the coupling factor must be modified as follows:

The total circuit inductance L_o is

$$L_o = L_1+L_2'+2M_{12}' \tag{6}$$

Since the load is magnetically uncoupled, it has no effect on the mutual inductance. Hence

$$M_{12}' = M_{12}$$

$$\text{or } k'\sqrt{L_1 L_2'} = k\sqrt{L_1 L_2}$$

and therefore

$$k' = k\sqrt{L_2/L_2'}$$

$$= k\left[\frac{L_2}{L_2+L_{LOAD}}\right]^{1/2} \tag{7}$$

In addition to calculating the step efficiency, account must also be taken of the fact that some of the final circuit energy is stored in L_2. This leads to a further efficiency penalty η_u, where

$$\eta_u = \frac{L_{LOAD}}{L_2+L_{LOAD}} \tag{8}$$

ow, let the ratios L_1/L_{LOAD} and L_2/L_{LOAD} be eferred to as σ_1 and σ_2 respectively. quation (7) shows that, as σ_2 increases, the alue of k' approaches that of k. This means hat for a given current multiplication β the tep efficiency η_s improves. However, at the ame time, the uncoupled load penalty η_u gets maller (equation (8)). The requirement is o determine the net effect on the total fficiency η_t, where

$$\eta_t = \eta_s \, \eta_u \qquad (9)$$

o analyse the effect of σ_2 on η_t, equation 9) is expanded. η_s is given by equation 4), but with k' in place of k. η_u can be xpressed in terms of σ_2. Multiplying the wo expressions together yields

$$t = \frac{k^2\beta^2\sigma_2}{\sigma_2^{\,2}[\beta^2-2\beta(1-k^2)+(1-k^2)] + \sigma_2[2\beta^2-2\beta(1-k^2)-2\beta+(1-k^2)+1] + [\beta^2-2\beta+1]} \qquad (10)$$

ifferentiating the result with respect to σ_2 ields

$$\frac{d\eta_t}{d\sigma_2} = \frac{k^2\beta^2A - k^2\beta^2\sigma_2(dA/d\sigma_2)}{A^2} \qquad (11)$$

here

$$A = [\beta^2(\sigma_2+1) + (k^2\sigma_2-\sigma_2-1)(2\beta-1)][1+\sigma_2]$$

tationary points of the function are located y equating the derivative to zero, which ields a non-imaginary value of

$$\sigma_2 = \frac{\beta-1}{[\beta^2 - 2\beta(1-k^2) + (1-k^2)]^{1/2}} \qquad (12)$$

he nature of this stationary point may be etermined by making the following bservations:

quation (12) represents the optimum value of $_2$, for which the efficiency η_t is maximised.

n example of the variation of η_t with σ_2 is hown in figure 3.

he analysis is completed by finding σ_1 in erms of σ_2, so that both the meatgrinder nductances are known in terms of the load nductance. This can be done by modifying quation (3) to account for the load, so that t becomes

$$\frac{L_1}{L_2+L_{LOAD}} = \frac{(\beta-1)^2}{k'^2} \qquad (13)$$

ubstituting equation (7) into equation (13) ives

$$\frac{L_1}{L_{LOAD}} = \sigma_1 = \frac{(\beta-1)^2}{k^2}\,\frac{(1+\sigma_2)^2}{\sigma_2} \qquad (14)$$

Figures 4(a) to 4(c) are sample curves derived from the above equations. They may be used to indicate at a glance the capability of an ideal single-step meatgrinder circuit.

ANALYSIS OF SINGLE-STEP MEATGRINDER WITH DECOMPRESSION

Introduction

In some cases it is undesirable for load current to flow whilst the meatgrinder is storing energy from the source, since this process may take a relatively long time. In such cases a decompression switch is used to short out the load during charging. Once the desired current has been reached the switch is opened in order to bring the load into circuit (see figure 5); operation then proceeds as before.

"Decompression" means that the flux is initially generated by the current in a single inductor; a second inductor is then brought into circuit and the generation of the flux is divided between the two. The process leads to both a reduction in current and a loss of energy in the switch.

Defining the decompression efficiency η_d as the ratio of the total circuit energy after decompression to the initial circuit energy, and the decompression current ratio β_d as the ratio of the corresponding currents, it is found that

$$\beta_d = \eta_d = \frac{L_T}{L_T+L_{LOAD}} \qquad (15)$$

where L_T is the total meatgrinder inductance.

Mathematical Analysis

The current multiplication in the load due to meatgrinder action is still β; as β_d simply serves to indicate the necessary initial charge current to give the required initial load current.

The overall efficiency with decompression η_{td} is given by

$$\eta_{td} = \eta_s \, \eta_u \, \eta_\sigma = \eta_t \, \eta_d \qquad (16)$$

where the other symbols have their previous meaning.

The total efficiency η_t may be derived as described in section 2. η_d may be expressed in terms of β and σ_2 by expanding L_T and using equation (15) to substitute for L_1. η_{td} may subsequently be expressed as

$$\eta_{td} = \frac{k^2\beta^2\sigma_2}{A}\,\frac{(A-k^2\sigma_2)}{A} \qquad (17)$$

Again, it is the stationary points of this function which are of interest. Differentiating equation (17) with respect to σ_2 and equating the derivative to zero leads to the following condition:

$$[A-2k^2\sigma_2]\,[Ak^2\beta^2-k^2\beta^2\sigma_2\,(dA/d\sigma_2)] = 0 \qquad (18)$$

where σ_2 is the stationary point value with decompression.

The values of σ_2 are found by equating each of the two brackets in turn to zero. It can be seen that applying this to the second bracket yields exactly the same result as the non-decompression case (equation (12)). Substituting for A in the first bracket and equating to zero leads to a quadratic in σ_2 the roots of which are imaginary and therefore not of interest.

Thus there is again only one stationary point of interest, and as before it is a maximum. This shows that the optimum value of σ_2 is the same with or without decompression.

Figure 6 refers to the same example as figure 3 and shows how decompression degrades the efficiency without shifting the point at which the maximum occurs. Figure 7 is derived from figure 4(a) by multiplying each efficiency value by η_d.

EXPERIMENTAL RESULTS

The purpose of these experiments was to provide a demonstration of the principle described above. The case arbitrarily chosen for the demonstration has the following parameters:

Current multiplication = 3
Coupling coefficient = 0.9
Load inductance = 100 μH

Figure 8 is the efficiency curve for this case and is derived from equation (10). The three representative values of σ_2 (i.e. L_2/L_{LOAD}) chosen for the experiments are indicated.
An initial current of 10A was selected in order that switching could be carried out with power MOSFETs, for ease of drive circuit design and switching speed compared with GTO's and bipolar transistors.

Coil Design

The meatgrinder consists of two concentric single-layer solenoids. Concentric solenoids have been successfully used in other meagrinder work [3,7], and the single-layer type is relatively simple to design and construct. High magnetic coupling can be obtained by making the coils large, so that the difference in cross-sectional areas is small. Even if the two inductances are very different, a high coupling coefficient can still be obtained by using different wire or strip sizes to produce coils of roughly equal width.

The inductance formula used is that given by Welsby [11]. It approximates the coil to a cylindrical current sheet, and is appropriate where the turns are close together and the radial thickness of the winding is small compared to the coil radius.

The formula is

$$L = \frac{r^2N^2}{b} \cdot \frac{4\pi^2 \times 10^{-3}}{1 + 0.9(r/b) - 0.02(r/b)^2} \ \mu H \qquad (19)$$

where
r = coil radius in cm
b = coil width in cm
N = number of turns

In order to minimise construction difficulties, the radius of the outer coil (L_1) was restricted to 25 cm. A nominal coil width of 10 cm was chosen. Equation (19) was then applied on a trial and error basis in order to find the number of turns needed to give the required inductance. The wire diameter or strip width required was given by coil width/number of turns.

In each of the three cases, L_1 and L_2 were designed to have the same width.

The load inductor was designed in a similar manner.

Mechanical Construction

The meatgrinder coils are shown in figure 9. L_1 is on the right and is wound with enamelled copper wire, whilst L_2, on the left, is wound with copper strip insulated with polyester film. In this particular case, L_2 was not wound as a single layer, the required inductance being obtained by trial and error. This was due to the required strip width not being available.

L_2 fits inside L_1. Adjustment of the coupling coefficient is provided by a screw arrangement constructed so as to enable L_2 to be moved up or down along the common axis of the two coils. To make the arrangement functional, the threaded road and handle assembly is first removed from L_2. L_2 is then placed inside L_1, the legs underneath L_2 sliding through holes in the base of L_2, and the handle on the end of the rod bears down on the top lid. Thus when the handle is turned, L_2 is forced to move relative to L_1.

Figure 10 shows the assembled meatgrinder connected to the load coil, and table 1 gives the inductance values for the three experiments.

The coupling coefficient for experiment 3 is rather low because an error in positioning the L_1 winding meant that the coils could not overlap to obtain k = 0.9. The experiment was not repeated because the performance was still satisfactory. (It should be remembered that the value of coupling coefficient obtained is highly sensitive to small changes in the inductance figures. In this particular case, for example, an increase of only 4% in the total inductance would raise the coupling coefficient to 0.88).

Other Circuit Components

Figure 11 is the circuit diagram for the experiments: the electronic components can be seen mounted on the small circuit board in figure 9.

TR1 is a high voltage (1000V) device with a 2Ω on-state resistance. A circuit simulation showed that because of the circuit resistance, an opening switch voltage of about 800V would be necessary to obtain a final current of 30A. The high on-stage resistance is unimportant for the purposes of the experiment because it simply increases the energy dissipated during charging. MOV1 is necessary to restrict the drain-source voltage during switching off for protection when operating above its continuous current rating.

TABLE 1 - Inductance figures

Expt. No.	L_1 (μH)		L_2 (μH)		L_1+L_2+2M (μH)	k
	Design	Meas'd	Design	Meas'd	(Meas'd)	
1	2782	2740	30	30	3283	0.89
2	2039	2016	70	70	2740	0.87
3	2009	2012	130	136	2950	0.77

(Meas'd = Measured)

By contrast, TR2 has a low on-state resistance (0.02Ω) so that the loop resistance is kept to a minimum for the energy transfer: TR2 does not break current but the possibility of unexpected voltage spikes still has to be considered. No external drain-source voltage limiting is required, however, because TR2 operates within its continuous current rating and is therefore able to self-avalanche if necessary.

The transistors are driven by a digital timing circuit. A charge time of 1.7 ms is used, and with a 48V supply, adjustment of R_{ext} allows the current after this time to be set to 10A. The power supply consists of four 12V car batteries.

Results

First experiment. Figure 12 shows how the circuit operates as expected: the voltage on TR1 is clamped by MOV1, and the current in L_{LOAD} rises to about 28A - quite close to the predicted value of 30A. Noise was reduced (and the final current slightly increased) by slowing down the turn-off of TR1 (see figure 13). This was achieved by inserting a 100Ω resistor in series with the gate.

The oscillations in the voltage waveform are due to the non-ideal behaviour of the MOSFET as shown by the simulated waveforms of figure 14. The behaviour of an ideal switch is shown in figure 15 for comparison.

Second and third experiments. Figures 16 and 17 are the waveforms corresponding to figure 12 for the second and third experiments respectively. The current multiplication is approximately 3 in each case, as expected.

Comments

The results show that a given current multiplication in the load can be achieved with different sets of meatgrinder inductances, only one of which maximises the efficiency. In this case it is the second experiment which corresponds to the maximum efficiency indicated in figure 8.

Table 2 shows the measured energy levels and the efficiencies obtained from the single step meatgrinder. Figure 8 also shows the experimental efficiency curve for comparison purposes. The small differences between the experimental and theoretical curves are due to (a) the coupling coefficient of the experimental meatgrinder is not 0.9, the figure used for predictions and (b) the theoretical model neglects the effect of winding resistances.

TABLE 2

Expt. No.	Initial Circuit Energy (mJ)	Final Load Energy (mJ)	Efficiency %
1	169.2	39.2	23.2
2	142.0	42.1	29.6
3	152.5	39.2	25.7

CONCLUSIONS

Theoretical analysis and experimental results have been presented which show that a simple technique exists for optimising the design of a single-step meatgrinder feeding an uncoupled load. It has further been shown that the optimum values for the two meatgrinder inductances are unaffected by the inclusion of a decompression switch. The work has shown how energy can be transferred between uncoupled inductors at efficiencies greater than 25%. The higher efficiency reduces the demand on both the power supply and the opening switches, thereby simplifying the design of a pulse power system.

The meatgrinder has the added advantage of providing current multiplication which avoids the need for conventional transformers which are also subject to the 25% efficiency limit when transferring energy to an uncoupled load [8].

A single-step meatgrinder has been successfully used to launch projectiles from an electromagnetic gun [10], and an experiment has been proposed [9] in which a single-step circuit would be used to generate a load power pulse of 100 TW. Optimisation of the meatgrinder design will be vital in such large-scale experiments.

REFERENCES

1. Zucker, O., Long, J., "The Meatgrinder : a reversible inductive storage and transfer system", Proc, 4th IEE Pulsed Power Conf. 1983, p.375.

2. Zucker, O. et al, "The meatgrinder: Theoretical and practical limitations", IEEE Trans. Magnetics March 1984, p.391.

3. Giorgi, D., Lindner, K. et al, "Proof of principle experiment of the meatgrinder: an inductive energy storage and transfer circuit", Proc. 5th IEEE Pulsed Power Conf. 1985, p.295.

Lindner, K., Long, J et al, "A meatgrinder circuit for energising resistive and varying inductive loads (EM Guns)", IEEE Trans. Magnetics Vol. 22 No. 6 November 1986, p. 1591.

Zucker, O., Long, J. et al, "Theory, experiments and applications of meatgrinder circuits", Proc. 4th Conf. Megagauss field generation and related topics 1986, p.609.

Giorgi, D., Lindner, K. et al, "The design and analysis of a multistage meatgrinder circuit", Proc. 5th IEEE Pulsed Power Conf. 1985, p.615.

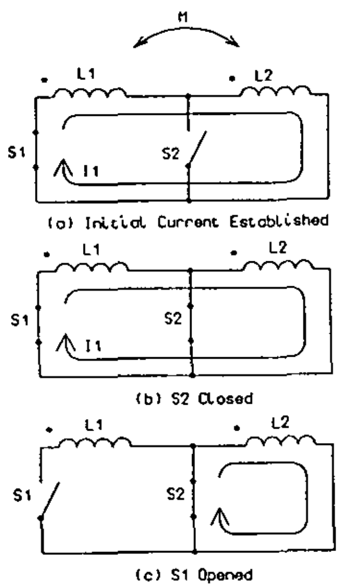
Giorgi, D., Long, J. et al, "New high-current meatgrinder experiments", IEEE Trans, Magnetics Vol. 22 No. 6 November 1986, p.1485.

8. Giorgi, D., Lindner, K. et al, "Enhancing the transfer of inductive energy to an imploding plasma load with a single-step meatgrinder circuit", IEEE Trans. Magnetics Vol. 23, No. 3 May 1987, p.1913.

9. Zucker, O., Long, J. et al, "The generation of 100TW with an explosive generator-driven single-step meatgrinder", Proc. 4th Conf. Megagauss Field Generation and Related Topics 1986, p.619.

10. Lindner, K., Long, J. et al, "Enhanced Energy Transfer to an EM Gun", Proc. 6th IEEE Pulsed Power Conf. 1987, p.749.

11. Welsby, V.G., "The Theory and Design of Inductance Coils", Macdonald 1964.



gure 1 Operation of Unloaded Single-Step Meatgrinder

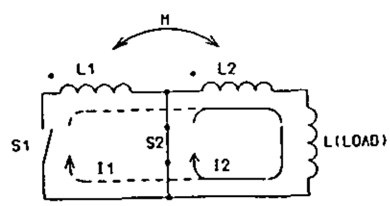


Figure 2 Single-Step Meatgrinder With Uncoupled Load

Figure 3 Example Showing Variation of Efficiency With L2/L(LOAD) (Multiplication=2; single-step, no decompression)

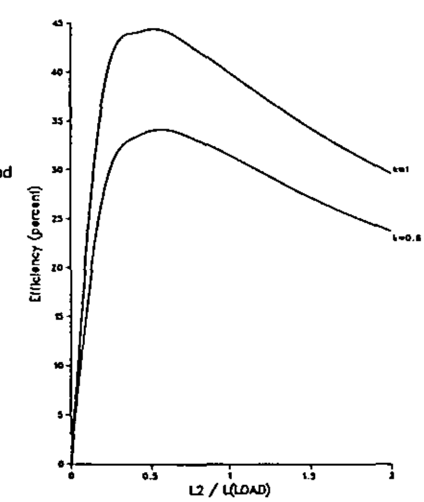


Figure 4(a) Single Step Transfer Efficiency at Optimal Conditions (Uncoupled Load, No Decompression)

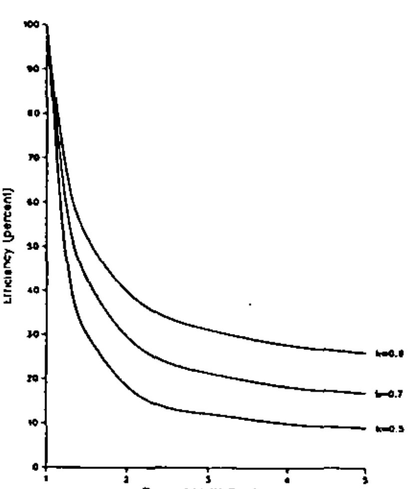


Figure 4(b) Optimum Values of L2/L(LOAD) for Single Step, No Decompression

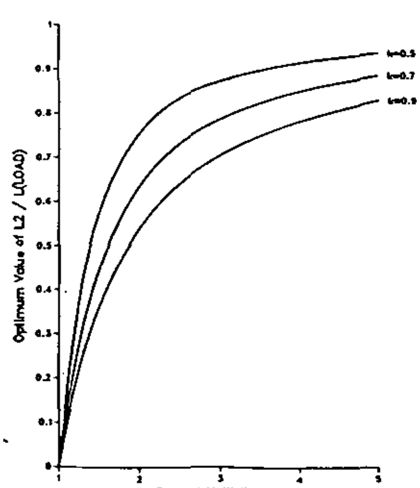
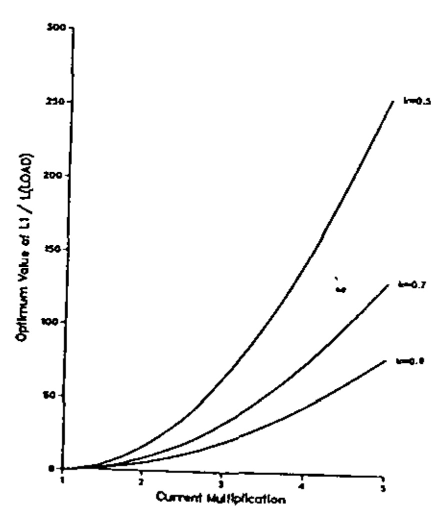
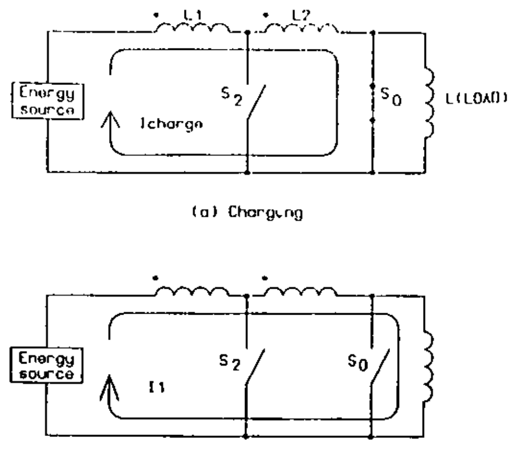


Figure 4(c) Optimum Values of L1/L(LOAD) for Single Step, No Decompression





Load Brought into Circuit by Opening Decompression Switch

Figure 5 Use of Decompression Switch

Figure 6 Example Showing Optimum Efficiency Of a Single Step (Current multiplication=2)

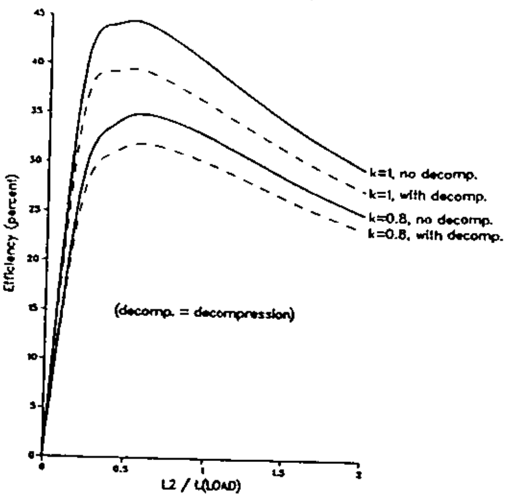
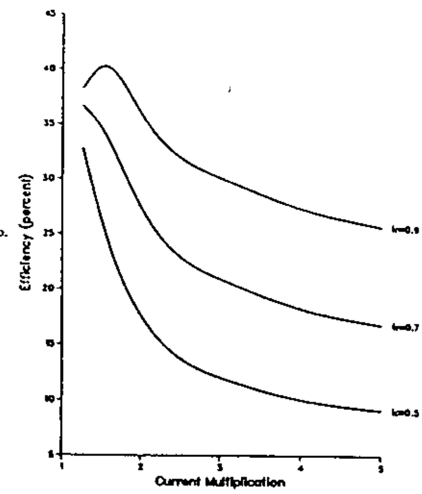
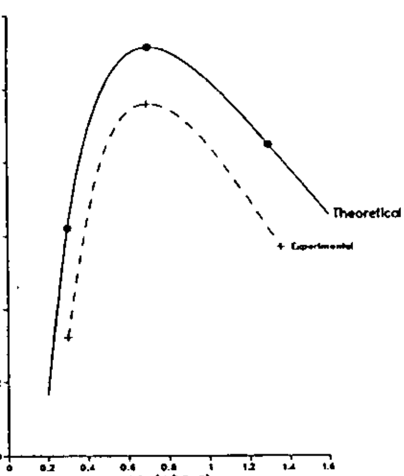


Figure 7 Single Step Transfer Efficiency at Optimal Conditions (Uncoupled Load, With Decompression)



8 Values for Multiplication=3, k=0.9, No Decompression



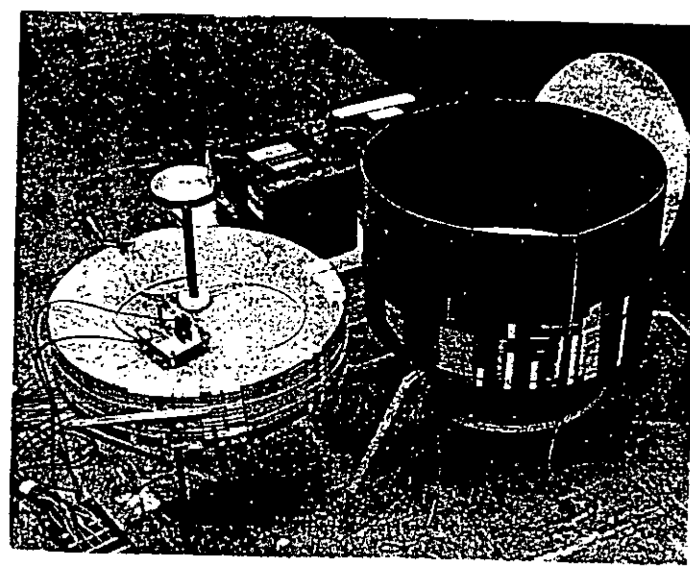


Figure 9 Single step meatgrinder coils

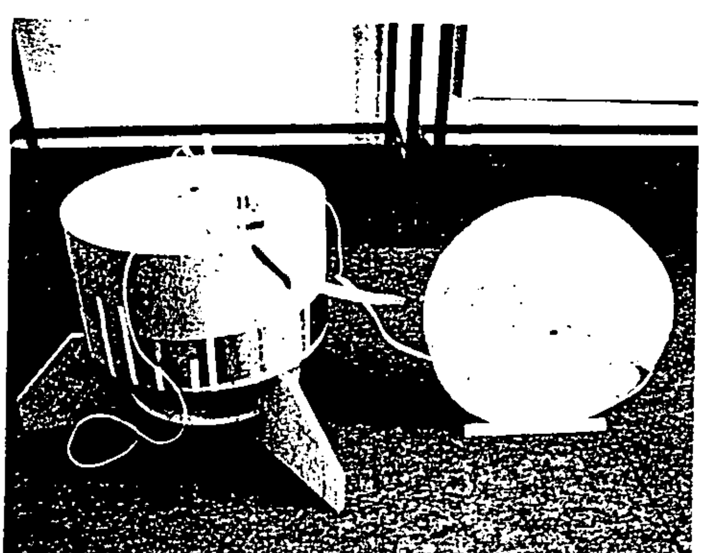
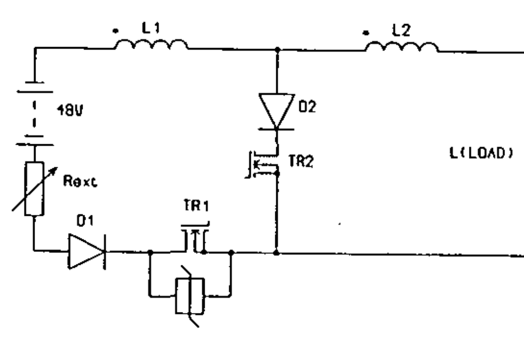


Figure 10 Single step meatgrinder and load coil arrangement



COMPONENTS NOT SHOWN: 18V gate-source zener diodes for TR1 and TR2
13A fuse in series with Rext

TR1: International Rectifier IRFPG50
TR2: International Rectifier IRFP044
D1,D2: Motorola MR756
MOV1: Power Development Z320C

Figure 11 Circuit Diagram for Single-Step Meatgrinder

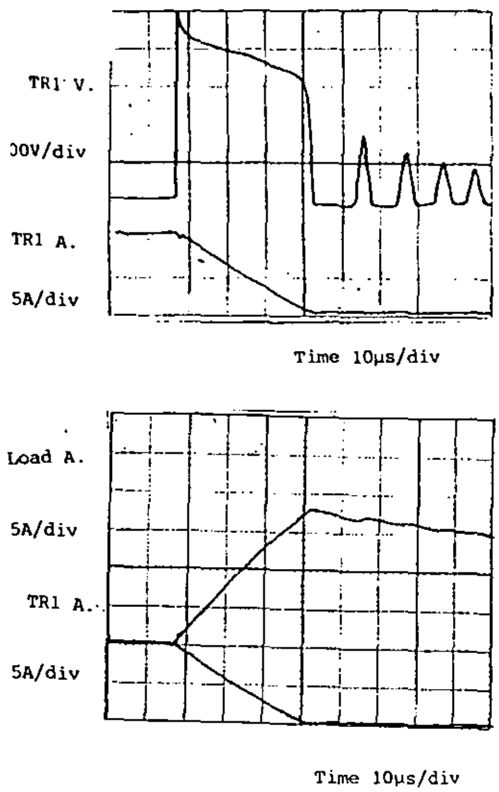


Figure 12 Waveforms for Experiment No.1

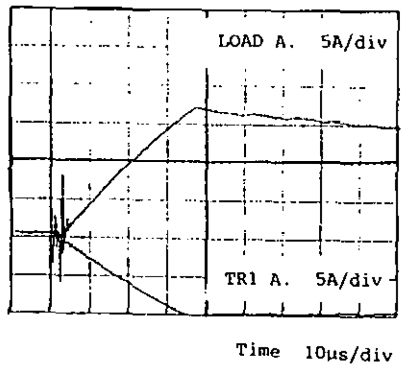


Figure 13 Current waveforms without additional gate resistor

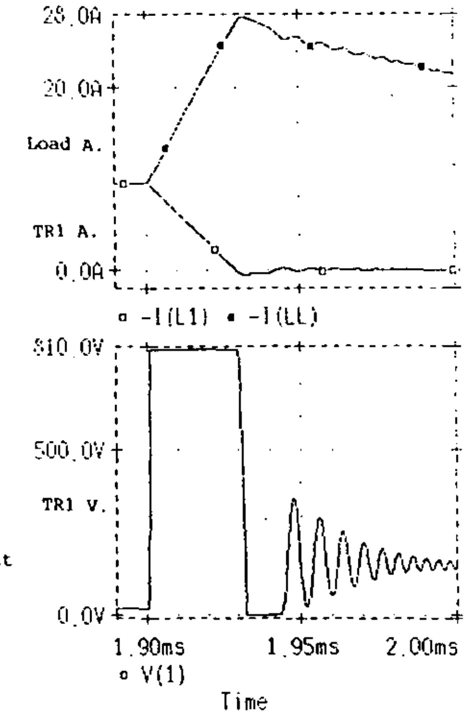


Figure 14 Waveforms showing non-ideal behaviour of TR1 - MOV1 Combination

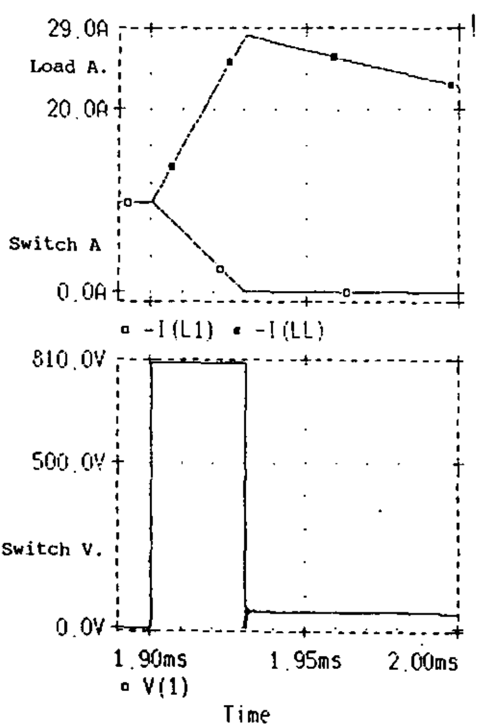


Figure 15 Waveforms showing behaviour of an ideal switch replacing TR1 in figure 11.

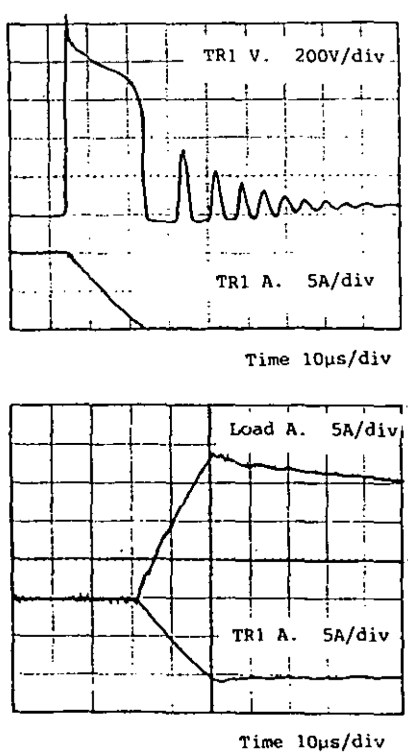


Figure 16 Waveforms for Experiment No.2

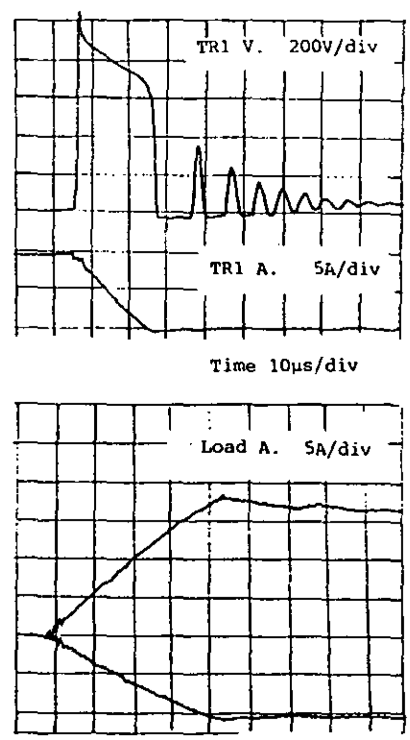


Figure 17 Waveforms for Experiment No.3