1

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service

Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

**THEORETICAL AND EXPERIMENTAL INVESTIGATIONS
OF COINCIDENCES IN POISSON DISTRIBUTED PULSE
TRAINS AND SPECTRAL DISTORTION CAUSED BY
PULSE PILEUP**

by

Quentin Bristow, B.Sc., P. Eng.

A Thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfilment of
the requirements for the degree of

Doctor of Philosophy

Department of Electronics

Carleton University
Ottawa, Ontario

27th March 1990

Copyright
1990 Quentin Bristow

National Library
of Canada

Bibliothèque nationale
du Canada

Canadian Theses Service

Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

ISBN 0-315-60449-2

1.3.6.4	Derivation of $f_x (\tau < (t_1 + t_2 + t_3) < 2\tau)$	25
CHAPTER 2 Development of Model Theory for Probability Measurements		27
2.0	Introduction	27
2.1	Single Coincidence Measurements	28
2.1.1	Sub-Event Probabilities	28
2.1.2	Complications in the Measurement of Overlapping Event Probabilities	29
2.2	Logic Circuit Models	30
2.2.1	Logical Relationships for Circuit I	34
2.2.2	Logical Relationships for Circuit II	37
2.3	Applications of Markov Process Theory to Circuit State Probabilities	39
2.3.1	The Steady State Markov Matrix Equation	42
2.3.2	Markov Analysis for Circuit I	43
2.3.3	Markov Analysis for Circuit II	46
2.4	A Simplified Markov Analysis for Random Event Detection Logic Circuits	47
2.4.1	Circuit I: Simplified Markov Analysis	47
2.4.1.1	Circuit I as a Canonical Model for Multiple Event Detection	49
2.4.2	Circuit II: Simplified Markov Analysis	49
2.4.3	Summary of Simplified Markov Analysis for Random Event Detection	51
2.5	Transition Rates by Calculation for Fixed-number Pulse Strings	52
2.5.1	Transition Rate Predictions for Strings with Three or More Pulses	56

ABSTRACT

Part one of this two-part study is concerned with the multiple coincidences in pulse trains from X-ray and gamma radiation detectors which are the cause of pulse pileup. A sequence of pulses with inter-arrival times less than τ , the resolving time of the pulse-height analysis system used to acquire spectra, is called a multiple pulse string. Such strings can be classified on the basis of the number of pulses they contain, or the number of resolving times they cover. The occurrence rates of such strings are derived from theoretical considerations. Logic circuits were devised to make experimental measurements of multiple pulse string occurrence rates in the output from a NaI(Tl) scintillation detector over a wide range of count rates. Markov process theory was used to predict state transition rates in the logic circuits, enabling the experimental data to be checked rigorously for conformity with those predicted for a Poisson distribution. No fundamental discrepancies were observed.

Part two of the study is concerned with a theoretical analysis of pulse pileup and the development of a discrete correction algorithm, based on the use of a function to simulate the coincidence spectrum produced by partial sums of pulses. Monte Carlo simulations, incorporating criteria for pulse pileup inherent in the operation of modern ADC's, were used to generate pileup spectra due to coincidences between two pulses, (1st order pileup) and three pulses (2nd order pileup), for different semi-Gaussian pulse shapes. Coincidences between pulses in a single channel produced a basic probability density function spectrum which can be regarded as an impulse response for a particular pulse shape. The use of a flat spectrum (identical count rates in all channels) in the simulations, and in a parallel theoretical analysis, showed the 1st order pileup distorted the spectrum to a linear ramp with a pileup tail. The correction algorithm was successfully applied to correct entire spectra for 1st and 2nd order

pileup; both those generated by Monte Carlo simulations and in addition some real spectra acquired with a laboratory multichannel analysis system.

ACKNOWLEDGEMENTS

I am most grateful to my supervisor, Prof. R.G. Harrison, who has spared no effort on many occasions in making arrangements to accommodate my part-time modus operandi, and the rather unusual thesis topic that came with it, when others might have wilted.

My most sincere thanks to Ms. Kathy Mooney, for her endless patience and superb skill in transcribing the many pages of difficult and complex mathematical notation with an uncanny precision which never ceases to amaze me.

I am indebted to John Glynn of the Geological Survey of Canada Data Systems Group for many helpful discussions on various aspects of probability theory, a subject of which he has a formidable grasp.

Last and most of all I wish to thank my wife Lucille, to whom this work is dedicated, for her unflagging support and inspiration and for the many sacrifices she has made over the long years which it has taken.

**THEORETICAL AND EXPERIMENTAL INVESTIGATIONS OF
MULTIPLE PULSE STRING PROBABILITIES IN POISSON DISTRIBUTED PULSE
TRAINS**

CONTENTS

PART ONE

	<u>Page</u>
OVERVIEW	1
CHAPTER 1 Theoretical Development of Multiple Pulse Strings	5
1.1 Definition of Multiple Pulse Strings	5
1.2 Fixed-number Pulse Strings	7
1.2.1 Relative Probabilities of Fixed-number Pulse Strings	7
1.2.2 Absolute Probabilities of Fixed-number Pulse Strings	9
1.3 Fixed-interval Pulse Strings	10
1.3.1 Equivalence of Fixed-number and Fixed-interval Strings	11
1.3.2 Analytical Basis for Determination of Fixed-interval String Rates and Number of Pulses Per String	12
1.3.3 Relative Contributions of an n-Pulse Fixed-number String to Fixed-interval Strings	14
1.3.4 Fixed-interval String: Special Case When $m = 1$	16
1.3.5 Defining Probability Density Functions for Fixed-interval Strings	18
1.3.6 Calculation of Probability Density Functions for Fixed-interval Strings of order $m > 1$	20
1.3.6.1 Derivation of $f_x(\tau < (t_1 + t_2) < 2\tau)$	22
1.3.6.2 Verification for $P(\tau < (t_1 + t_2) < 2\tau)$	23
1.3.6.3 Derivation of $f_x(2\tau < (t_1 + t_2 + t_3) < 3\tau)$	24

1.3.6.4	Derivation of $f_x (\tau < (t_1 + t_2 + t_3) < 2\tau)$	25
CHAPTER 2 Development of Model Theory for Probability Measurements		27
2.0	Introduction	27
2.1	Single Coincidence Measurements	28
2.1.1	Sub-Event Probabilities	28
2.1.2	Complications in the Measurement of Overlapping Event Probabilities	29
2.2	Logic Circuit Models	30
2.2.1	Logical Relationships for Circuit I	34
2.2.2	Logical Relationships for Circuit II	37
2.3	Applications of Markov Process Theory to Circuit State Probabilities	39
2.3.1	The Steady State Markov Matrix Equation	42
2.3.2	Markov Analysis for Circuit I	43
2.3.3	Markov Analysis for Circuit II	46
2.4	A Simplified Markov Analysis for Random Event Detection Logic Circuits	47
2.4.1	Circuit I: Simplified Markov Analysis	47
2.4.1.1	Circuit I as a Canonical Model for Multiple Event Detection	49
2.4.2	Circuit II: Simplified Markov Analysis	49
2.4.3	Summary of Simplified Markov Analysis for Random Event Detection	51
2.5	Transition Rates by Calculation for Fixed-number Pulse Strings	52
2.5.1	Transition Rate Predictions for Strings with Three or More Pulses	56

2.5.2	Calculation of "Failure" Pulse Rate in State B	58
2.5.3	Derivation of Final Equations for Transition Rates	60
2.5.3.1	Circuit I: Transition Rates	61
2.5.3.2	Circuit II: Transition Rates	62
2.5.4	Numerical Evaluation of Transition Rates	63
CHAPTER 3 Measurement of Multiple Pulse String Occurrence Rates		64
3.0	Review of Sub-Event Probabilities to be Measured	64
3.1	Logic Circuit Implementation	65
3.1.2	State Sequence Control Logic	65
3.1.3	Parameter Inputs and Logic Algorithms for Circuits I and II	68
3.1.4	Circuits I and II in Integrated Circuit Form	72
3.2	Experimental Details	73
3.2.1	Pulse Widths	73
3.2.2	Dead Time Measurements	76
3.2.3	Simultaneous Measurements of Fixed-number and Fixed-Interval Strings	77
CHAPTER 4 Data Analysis and Interpretation		79
4.0	Overview	79
4.1	Correction for Finite Pulse Widths	80
4.2	Fixed-number Pulse String Data from Circuit I Measurements	81
4.2.2	Predicted Transition Rate Data	82
4.2.3	Comparison of Experimental and Predicted Data	82
4.3	Simulated ADC Dead Time Measurements	89

4.4	Fixed-number Pulse String Data From Circuit II Measurements	91
4.5	Fixed-interval String Data from Circuit II Measurements	97
4.6	Simultaneous Fixed-number and Fixed-interval Measurement Data	101
4.7	Conclusions	104

PART TWO

	Page
OVERVIEW	105
CHAPTER 5 A Theoretical Analysis of Pileup, Development of a Discrete Correction Algorithm and Monte Carlo Models	107
5.0 Introduction	107
5.1 The Coincidence Spectrum of Two Pulses	108
5.1.2 The Theoretical Coincidence Probability Density Function	113
5.2 Analysing the Spectral Distortion due to Pulse Pileup	115
5.3 A Discrete Summation Algorithm to Simulate Pulse Pileup	120
5.3.1 Correction for 1st Order Pileup using the Algorithm in Reverse	121
5.4 The Effect of Pileup on a Flat Spectrum	124
5.4.1 A Theoretical Analysis of Spectral Pileup	124
5.4.2 A Flexible Matching Function to Represent $f(\text{PU}, I, J)$	129
5.5 Pileup Simulation using Monte Carlo Technique	134
5.5.1 Redefining the Resolving Time τ	136
5.6 Application of Correction Algorithm to Monte Carlo Pileup Simulations	140
5.6.1 Generation of a Monte Carlo Flat Spectrum with Pileup	140
5.6.2 Generation of Monte Carlo Spectra with Gaussian Peaks	144
5.6.3 Correction of Pileup in a Spectrum with Gaussian Peaks	146
5.6.4 Errors and Noise Sources due to the Correction Algorithm	149

CHAPTER 6	Extension of the Theory and Correction Algorithm to Include 2nd Order Pileup	151
6.0	Introduction	151
6.1	2nd Order Pulse Pileup From Monte Carlo Simulations	151
6.2	Analysis of the "Forbidden Region" and 2nd Order Impulse Response	153
6.3	Coincidences Within and Between Channels	157
6.3.1	Single Coincidences	157
6.3.2	Multiple Coincidences	160
6.4	Modification of the Correction Algorithm to Include 2nd Order Pileup	161
6.5	Fraction of Pulses Moved or Lost Due to Nth Order Pileup	165
6.6	Values of τ_F , τ_R from 2nd Order Pileup Impulse Responses	168
6.7	Effect of 2nd Order Pileup on a Flat Spectrum	171
6.7.2	Non-uniform Distribution of Single Pulses Surviving Coincidences	173
6.7.3	Correction of a Flat Spectrum with 2nd Order Pileup	177
CHAPTER 7	Correction of Spectra Acquired with a Multichannel Analysis System	181
7.0	Introduction	181
7.1	Flat Spectrum Generator, Concept and Design	182
7.2	Experimental Data from the Hardware Flat Spectrum Generator	186
7.2.2	Coincidences Below the Lower Level Discriminator (LLD)	193
7.3	Acquisition and Correction of a Real Radiation Spectrum	196
7.3.1	Experimental Measurements of System Values τ_F and τ_R	197
7.3.2	Application of τ_F , τ_R Data to Correction of a ^{137}Cs Spectrum	202

CHAPTER 8 SUMMARY, CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK	204
8.0 Summary of Contributions arising from this Study	204
8.1 Conclusions	206
8.2 Recommendations for Future Work	207

APPENDICES

1-A	Examples of Fixed-Number Pulse String Data - Circuit I
1-B	Predicted Data for Fixed-Number Pulse Strings
2A	Examples of Fixed-Number Pulse String Data - Circuit II
2B	Predicted Data for Fixed-Number Pulse Strings - Circuit III
3	Examples of Fixed-Interval String Data
4A	Examples of Relative Probability Data
4B	Relative Probability Data Generated by Numerical Convolution
5	Summary of Previous Work References

LIST OF FIGURES

Figure

- 1.1 Typical random pulse sequence with multiple pulse strings identified.
- 1.2 Detailed illustration of fixed-interval strings.
- 1.3 Convolution of truncated probability density function.
- 1.4 Multiple convolution process.

- 2.1 Block diagrams for logic Circuits I and II.
- 2.2 Examples of overlapping coincidence events.
- 2.3 Responses of Circuits I and II to a sequence with overlaps.
- 2.4 Circuit I as a Markov chain.
- 2.5 Circuit II as a Markov chain.

- 3.1 State sequence logic used in Circuits I and II.
- 3.2 Circuit I implemented in Erasable-Programmable-Logic-Device.
- 3.3 Circuit II implemented in Erasable-Programmable-Logic-Device.
- 3.4 Block diagram of entire circuitry required for coincidence measurements.
- 3.5 Photograph showing Experimental Arrangement.

- 4.1 to 4.5 Experimental vs predicted transition rates for fixed-number pulse strings: Circuit I data.
- 4.6 to 4.10 Experimental vs predicted transition rates for fixed-number pulse strings: Circuit II data.
- 4.11 Experimental vs predicted occurrence rates for fixed-interval strings: Circuit II data.

- 5.1.1A, B Triangular pulse coincidences.
- 5.1.2 Sequence of coalescing semi gaussian pulses.
- 5.1.3 Coincidence spectra for different pulse shapes pulses of equal amplitude.
- 5.1.4 Computed Impulse Responses, showing count rate dependency.
- 5.2.1A Pulse A, B, C, contributions when in coincidence with M.
- 5.2.1B General case for a pulse in channel I in coincidence with M.
- 5.4.1 Effect of algorithm on a flat spectrum.
- 5.4.2 Formulation of continuous function algorithm.
- 5.4.3 Computed pileup in a flat spectrum together with that of Fig. 5.4.1.
- 5.4.4 Flexible analytical function to represent $f(\text{PU})$.
- 5.4.5 Logic for criteria of pulse pileup simulation.
- 5.6.1 Comparison of Monte Carlo and discrete algorithm Impulse responses.
- 5.6.2. Correction of flat spectrum and distortion produced by algorithm.
- 5.6.3 Over-correction of flat spectrum.
- 5.6.4 Monte Carlo simulation of pileup in a spectrum with three peaks.
- 5.6.5 Expanded segment of Fig. 5.6.4 showing efficacy of correction.

- 6.1.1 2nd Order Impulse Response channel 125.
- 6.2.1A Convolution of uniform response with no forbidden region.
- 6.2.1B Convolution of uniform response with forbidden region.
- 6.2.2 Convolution of Monte Carlo 1st order impulse response and convolution of matching function.

- 6.4.1 2nd Order Impulse Response with line segments superimposed.
- 6.7.1 2nd order pileup in a flat spectrum.
- 6.7.2 Non-uniform distribution of surviving single pulses in a flat spectrum.
- 6.7.3 Pulse-pairs with unequal pulse amplitudes.
- 6.7.4 Correction of 2nd order pileup in a flat spectrum.

- 7.1.1 Hardware Flat Spectrum Generator block diagram.
- 7.2.1 Pulse shape and a one millisecond pulse train from the Flat-Spectrum Generator.
- 7.2.2 Typical Impulse Response from Flat Spectrum Generator.
- 7.2.3 Examples of spectra from the Flat Spectrum Generator.
- 7.2.4 Correction of spectra from the Flat Spectrum Generator.
- 7.2.5 Error introduced by a high setting of the LLD.
- 7.3.1 Comparison of pulse shape from scintillation detector.
- 7.3.2 Impulse Response of laboratory MCA and matching $f(\text{PU})$ function.
- 7.3.3 Correction of a ^{137}Cs spectrum for 1st and 2nd order pileup.

**THEORETICAL AND EXPERIMENTAL INVESTIGATIONS OF
COINCIDENCES IN POISSON DISTRIBUTED PULSE TRAINS AND
SPECTRAL DISTORTION CAUSED BY PULSE PILEUP**

OVERVIEW

The acquisition of radiation spectra from various types of energy dispersive nuclear radiation detectors has become a distinct and unique field of scientific endeavour over the forty odd years since its inception. It involves the real time sorting of electrical pulses according to amplitude which have inter-arrival times governed by a Poisson distribution.

The fidelity of the acquisition process is degraded by two major, and various minor interference sources. The two major ones are (a) the dead-time (generated by the acquisition electronics, particularly the analogue to digital converter (ADC), while a pulse amplitude is being measured) which leads to an uncertainty in the estimated true input count rate, and (b) the random summing whereby more than one pulse occurs within the resolving time of the acquisition electronics leading to a false amplitude measurement, a phenomenon known as pulse pileup.

The problem of pulse pileup is essentially a manifestation of the Poisson statistical distribution of the pulse inter-arrival times. No matter how small the resolving time of the signal conditioning electronics, there will never be a hardware solution to the problem of distinguishing between a single pulse and the near perfect superimposition of two or more pulses (not necessarily of the same amplitude). The cause is thus a basic limitation imposed by a random process over which there is no control. The effect is to degrade the

spectrum in such a way that significant errors occur when these spectra are used as a basis for elemental analysis e.g. following neutron activation of a sample of material.

This study is divided into two parts. Part one is an account of theoretical and experimental investigations into the phenomenon of coincidences in Poisson distributed pulse trains, which are the cause of pulse pileup. Part two deals in detail with the problem of pulse pileup and describes experimental work to induce pileup and a statistically-based correction algorithm which has been developed to eliminate a substantial part of the spectral distortion which it causes.

The theoretical part of the pulse train study, which is covered in Chapter 1, defines two categories of coincidences which are called **Fixed-number Pulse Strings** and **Fixed-interval Pulse Strings**, where the term "Pulse String" means a sequence of pulses with separations less than a given time interval τ . The equivalence of these two classifications of coincidences, and their respective probabilities of occurrence, are discussed and analyzed in detail.

Chapter 2 describes the development of model theory, which provides the basis for the experimental measurements detailed in Chapter 3. This model theory involves the development of logic circuits to detect multiple pulse strings, and the application of Markov process theory to determine the correct occurrence rates of the various types of coincidences.

Chapter 3 provides an account of the experimental work. This involved using the logic circuits, which were implemented using Programmable Array Logic and controlled by a minicomputer, to measure occurrence rates of the various classes of coincidences occurring in a pulse train generated by a scintillation detector and a radioactive source. The minicomputer used for the data acquisition was

programmed to provide occurrence rate data directly, based on the application of the Markov process theory.

Chapter 4 is a discussion of the experimental data and a comparison with theoretically derived occurrence rate data based on the theory developed in Chapters 1 and 2. The close agreement between the experimental and theoretical results confirms that the theoretical analysis is sound and lends incidental support to the assumption of a Poisson distribution for radioactive decay processes, an assumption which has been debated in the literature from time to time.

The pulse pileup study begins at Chapter 5. This opens with a detailed discussion of the shape of the coincidence spectrum obtained when two pulses coincide to produce partial and total sum pulses. A theoretical expression for the gain in a channel due to pileup is derived which can be evaluated as a discrete function, and in simple cases as a continuous function. It can be used to produce or remove pileup distortion in a model spectrum. The implementation of a correction/distortion algorithm is described which is verified by application to pileup spectra generated by Monte Carlo simulations. The use of a flat spectrum as a model in pulse pileup simulations leads to some interesting results. Consideration of the way in which modern ADC's analyse pulse height leads to a re-definition of the pulse resolving time τ as a two-component parameter.

Chapter 6 extends the theoretical considerations of Chapter 5 to cover 2nd order pulse pileup, i.e. that involving three pulses in coincidence. The Monte Carlo simulation criteria and the correction/distortion algorithm are modified to produce and correct for this type of distortion. It is shown however that it is of much less importance as a source of error than 1st order pileup (two pulses only in coincidence).

Chapter 7 is an account of two experiments which establish the equivalence of the simulations and what occurs in the acquisition of spectra with a real laboratory multichannel analyser using a modern high-speed ADC. It is shown that a hardware flat-spectrum generator, designed to generate real data in a form directly comparable with those obtained in the Monte Carlo simulations, produces results which conform closely with them. The second experiment demonstrates that a spectrum acquired at a count rate high enough to produce significant pileup, can be corrected using the correction algorithm to remove the distortion caused by it.

Appendix 5 is a Summary Review of previous work in the field covered by this study and contains a bibliography, (by no means all-inclusive), which is the source of the numbered references which appear throughout the main text.

CHAPTER 1 THEORETICAL DEVELOPMENT OF MULTIPLE PULSE STRINGS

1.1 DEFINITION OF MULTIPLE PULSE STRINGS

Poisson distributed pulse trains can be grouped into single pulses and two different classes of strings according as the spaces between pulses (inter-arrival times) are greater-than-or-equal-to, or less-than some arbitrary time interval τ . Figure 1B illustrates these multiple pulse strings.

Figure 1.1(a): Typical random pulse sequence. (b) Inter-arrival times $> \tau$ identified which separate pulses into strings. (c) Fixed-number pulse strings preceded and followed by spaces $> \tau/2$ account for all pulses without overlapping intervals.

The two classes of strings have a common element in that both are characterized by the criteria that the strings consist of pulses having consecutive inter-arrival times $< \tau$, and are preceded and followed by spaces $> \tau$. The first class of string consists of fixed-numbers of pulses covering all the positive integer numbers. The second and potentially more useful class consists of strings occurring in fixed time intervals, the intervals being all of the positive integer multiples of τ .

It has been shown by *Tenney*⁵⁸ that the occurrence rate of the fixed-number pulse strings can be expressed as an infinite series which is a simple geometric progression. The occurrence rate of the fixed-interval strings, and the number of pulses per string, can only be obtained from a theoretical derivation which is considerably more complex. A computer evaluation of the formula derived produces predicted data which is in good agreement with experimentally determined string occurrence rates.

Pulse trains from nuclear radiation detectors are widely assumed to have a Poisson distribution, with inter-arrival times having an exponential distribution. (Background material relevant to the statistical theory is to be found in references 85-89.) For a mean input pulse rate N the probability that an inter-arrival time t will be $< \tau$ is given by integration of the probability density function, Ne^{-Nt} , from 0 to τ :

$$P(t < \tau) = \int_0^{\tau} Ne^{-Nt} dt = 1 - e^{-N\tau} \quad (1.1)$$

alternatively, the probability that an inter-arrival time t will be $> \tau$ is found by integrating Ne^{-Nt} from τ to ∞ :

$$P(t > \tau) = \int_{\tau}^{\infty} Ne^{-Nt} dt = e^{-N\tau} \quad (1.2)$$

In dealing with multiple pulse strings it is a point of some importance that the beginning of a string is marked by a pulse. The theory which is developed below is based on the number of inter-arrival times in a given interval rather than the number of pulses.

1.2 FIXED-NUMBER PULSE STRINGS

A fixed-number pulse string of order n is a train of n pulses having inter-arrival times $< \tau$ which is preceded and followed by spaces $> \tau$. It is shown below in a derivation that takes a different approach from that of *Tenney*⁵⁸, that the probability P_n of the occurrence of such a string is given by:

$$P_n = e^{-2z} (1 - e^{-z})^{n-1}$$

where $z = N\tau$, N being the true mean pulse arrival rate. When $n = 1$ this reduces to:

$$P_1 = e^{-2z}$$

which says that the probability of a single pulse preceded and followed by spaces $> \tau$ is e^{-2z} , a result obtained by *Wytttenbach*²⁴.

A typical sequence is shown in Figure 1.1(b). In order to avoid the problem of overlapping events the strings are shown in Figure 1.1(c) with spaces at each end $> \tau/2$. This simplifies the computation of the occurrence probability of the strings while still preserving the minimum separation between two strings as an interval $> \tau$. If single pulses are included, then the probabilities must sum to unity. It should be emphasized that these will be relative and not absolute probabilities.

1.2.1 RELATIVE PROBABILITIES OF FIXED-NUMBER PULSE STRINGS

Single Pulses

This is the probability of two consecutive inter-arrival times each $> \tau/2$. Hence

$$P(\text{single pulse}) = e^{-N\tau/2} \cdot e^{-N\tau/2} = e^{-N\tau}$$

Two Pulses

The probability of two pulses separated by an inter-arrival time $< \tau$ with leading and trailing intervals $> \tau/2$.

$$P(2 \text{ pulses}) = e^{-N\tau/2} \cdot (1 - e^{-N\tau}) \cdot e^{-N\tau/2} = e^{-N\tau} (1 - e^{-N\tau}) \quad (1.1)$$

Three Pulses

The probability of three-pulses with two consecutive inter-arrival times $< \tau$ and leading and trailing spaces $> \tau/2$.

$$P(3 \text{ pulses}) = e^{-N\tau/2} \cdot (1 - e^{-N\tau})^2 \cdot e^{-N\tau/2} = e^{-N\tau} \cdot (1 - e^{-N\tau})^2 \quad (1.2)$$

n Pulses

The probability of n pulses with $(n-1)$ consecutive inter-arrival times $< \tau$, with leading and trailing intervals $> \tau/2$.

$$P(n \text{ pulses}) = e^{-N\tau/2} \cdot (1 - e^{-N\tau})^{(n-1)} \cdot e^{-N\tau/2} = e^{-N\tau} \cdot (1 - e^{-N\tau})^{(n-1)} \quad (1.3)$$

The sum of the relative probabilities of all strings (which accounts for the entire pulse train) is thus:

$$P(\text{all strings}) = \sum_{n=1}^{\infty} e^{-N\tau} (1 - e^{-N\tau})^{(n-1)}$$

This is a geometric progression with first term $a = e^{-N\tau}$ and common ratio $r = (1 - e^{-N\tau})$ for which the sum to infinity is given by:

$$S_{\infty} = \frac{a}{1-r} = \frac{e^{-N\tau}}{1 - (1 - e^{-N\tau})} = 1$$

The relative probabilities thus sum to unity as they should.

1.2.2 ABSOLUTE PROBABILITIES OF FIXED-NUMBER PULSE STRINGS

The division of strings by means of half-spaces is an artificial one. In practice the only way to recognize and detect real strings is to look for a full space at each end (an inter-arrival time $>\tau$). Obviously the strings overlap to the extent that they share these spaces. We can say however that there is one space per string or event, i.e. the number of spaces is also the number of strings. The occurrence rate of spaces (inter-arrival times $>\tau$) is given by $N.e^{-N\tau}$ where N is the mean input pulse rate. The occurrence rate of each string is thus given by the total event rate, $N.e^{-N\tau}$, multiplied by the relative probability as given in (1.3).

Pulses/String	Relative Prob. $\times N.e^{-N\tau}$	= String Occurrence Rate
1	$e^{-N\tau}$	$N.e^{-2N\tau}$
2	$e^{-N\tau}(1-e^{-N\tau})$	$N.e^{-2N\tau}(1-e^{-N\tau})$
3	$e^{-N\tau}(1-e^{-N\tau})^2$	$N.e^{-2N\tau}(1-e^{-N\tau})^2$
n	$e^{-N\tau}(1-e^{-N\tau})^{(n-1)}$	$N.e^{-2N\tau}(1-e^{-N\tau})^{(n-1)}$

The occurrence rate of a fixed-number pulse string having n pulses is thus given by:-

$$\text{Occurrence rate} = N.e^{-2N\tau}(1-e^{-N\tau})^{(n-1)} \quad (1.4)$$

which is the result obtained by *Tenney*⁵⁸.

It will be seen in Part II of this study, that equation (1.4) is of fundamental importance in the theory of pulse pileup.

An independent check on the validity of the occurrence rates as derived above can be made by calculating the sum of all pulses in the various strings and verifying that the total pulse rate is equal to N , the input pulse

rate. The sum of occurrence rates for each string times the number of pulses in the string should thus be equal to N .

$$\text{Total pulse rate} = \sum_{n=1}^{\infty} n \cdot N \cdot e^{-2N\tau} (1 - e^{-N\tau})^{(n-1)}$$

This is an arithmetic-geometric progression of the type $a + 2ar + 3ar^2 + \dots nar^{(n-1)}$

which has a sum to infinity given by:

$$S_{\infty} = \frac{a}{(1-r)^2} = \frac{N \cdot e^{-2N\tau}}{(1 - (1 - e^{-N\tau}))^2} = N$$

This verifies that the assumptions involved in developing the theory are at least self-consistent.

1.3 FIXED-INTERVAL PULSE STRINGS

These are defined as multiple pulse strings with consecutive inter-arrival times $< \tau$ occurring in fixed time intervals which are integer multiples of τ . A space $> \tau$ precedes the pulse which marks the start of the $n\tau$ fixed-interval. The space $> \tau$ at the end of the string must have its origin in the final sub-interval τ of the $n\tau$ interval. In other words there must be at least one pulse in the final sub-interval, and any inter-arrival time after that which is $> \tau$ ends the string.

Figure 1.2(A): A 1τ Fixed-interval string. (B) An $n\tau$ Fixed-interval String.

Figure 1.2 shows two examples of a fixed-interval string. In figure 1.2(A) the string is classified as a $1-\tau$ or single interval string. It begins with a space $>\tau$; the space is ended by the arrival of a pulse, (the "marker pulse"), following which is a series of pulses with inter-arrival times $<\tau$. However before an interval τ (measured from the marker pulse) has elapsed, an inter-arrival time begins which is $>\tau$, and it ends the string. Figure 1.2(B) shows an $n\tau$ fixed-interval string. In this case the pulses arriving after the marker pulse maintain inter-arrival times $<\tau$ until the n th sub-interval, (the sub-interval covering time $(n-1)\tau$ from the marker pulse to $n\tau$), at that point an inter-arrival time $>\tau$ begins which ends the string.

1.3.1 EQUIVALENCE OF FIXED-NUMBER AND FIXED-INTERVAL STRINGS

All multiple pulse strings consist of pulse trains with successive spacings $<\tau$ and are delineated by spaces $>\tau$. Each string has a dual classification. The number of pulses it contains determines its order as a fixed-number pulse string, the number of τ intervals it covers determines its order as a fixed-interval string.

The theory covering the properties of the fixed-number pulse strings has been discussed in the previous sections and is relatively straightforward. The prediction of fixed-interval string occurrence rates and the number of pulses per string is rather more complicated. It is simplified somewhat by working on the principle that a fixed-interval string of order m , (i.e. one with $m\tau$ intervals), is synthesized from a group of fixed-number strings. That group cannot include strings with less than $(m + 1)$ pulses, since there must be one marker pulse at the start of a fixed-interval string, and at least m pulses in the $m\tau$ intervals.

It is shown below that this equivalence is useful in verifying that the analytical solution for fixed-interval string occurrence rates and determination of pulses per string is correct.

1.3.2 ANALYTICAL BASIS FOR DETERMINATION OF FIXED-INTERVAL STRING RATES AND NUMBER OF PULSES PER STRING

The undistorted distribution of inter-arrival times is exponential, however the pulses which form the multiple pulse strings have by definition inter-arrival times which are $< \tau$. Evidently therefore they have a distribution and probability density function which is distorted. The actual probability density function is shown in Figure 1.3(a). The undistorted one would be given by:

$$f_x = \lambda e^{-\lambda t}$$

where λ is the input pulse rate. Because there are no inter-arrival times $> \tau$ however, the exponential is truncated at $t = \tau$ as shown in Figure 1.3(a). This then is the distorted probability density function for the inter-arrival times of pulses in the multiple pulse strings.

The probability that there are n pulses in an $m\tau$ fixed-interval string is the probability that the sum of the $(n-1)$ inter-arrival times involved falls in the m^{th} sub interval of the string.

Now the probability density function of a random variable which is the sum of n other random variables is found by the convolution of the n individual density functions involved (see e.g. *Papoulis*⁸⁹):

$$f_x(\text{sum}) = f_x^{(1)} * f_x^{(2)} * f_x^{(3)} * \dots * f_x^{(n)}$$

Thus the probability density function for n pulses in an $m\tau$ string is found by an $(n-1)$ fold convolution of the truncated exponential density function shown in Figure 1.3(a), since n pulses have $(n-1)$ inter-arrival times.

Figure 1.3 (a): Truncated probability density function for inter-arrival times $< \tau$. **(b)** Convolution of truncated function with itself produces in **(c)** the probability density function of the sum of two inter-arrival times each $< \tau$.

The overall occurrence probability for the $m\tau$ fixed-interval string is thus the sum of the probabilities of there being $(m + 1)$, $(m + 2)$, $(m + 3)$ etc., pulses given by:

$$p(m\tau \text{ string}) = p(m + 1) + p(m + 2) + p(m + 3) + \dots$$

where $p(m + 1)$, $p(m + 2)$, etc., are the individual probabilities determined as described above.

The probable number of pulses in the $m\tau$ string is then a weighted average of these probabilities:

$$\text{Pulses in } m\tau \text{ string} = \frac{(m + 1) \cdot p(m + 1) + (m + 2) \cdot p(m + 2) + (m + 3) \cdot p(m + 3) + \dots}{p(m + 1) + p(m + 2) + p(m + 3) + \dots} \quad (1.5)$$

1.3.3 RELATIVE CONTRIBUTIONS OF AN n -PULSE FIXED-NUMBER STRING TO FIXED-INTERVAL STRINGS

An n -pulse fixed-number string can qualify as an $m\tau$ fixed-interval string for any $m < (n - 1)$. For example a 4-pulse fixed-number string could qualify as a 1τ , 2τ or 3τ fixed-interval string. In fact it must qualify as one or other of these, so that the relative probabilities of these three outcomes must add to unity.

The relative probability that an n -pulse fixed-number string will qualify as an $m\tau$ fixed-interval string can be written as a conditional probability expression (see e.g. *Papoulis*⁸⁹):

$$P(A/B) = \frac{P(A, B)}{P(B)}$$

where $P(A)$ = probability (sum of $(n - 1)$ inter-arrival times falls in m th interval)

$P(B)$ = probability $((n - 1)$ consecutive inter-arrival times each $< \tau$)

The joint probability, $P(A, B)$, is that there are $(n-1)$ inter-arrival times with their sum in the m^{th} interval, and that each one of them is $< \tau$. This is determined by the convolution of the truncated exponential with itself $(n-1)$ times as discussed previously.

The probability, $P(B)$, that there are $(n-1)$ consecutive inter-arrival times each $< \tau$ was a condition dealt with in the theory of the fixed-number pulse strings (see section 1.2.1) and is given by:

$$P(B) = (1 - e^{-N\tau})^{(n-1)}$$

where N as before is the input pulse rate. the notation is simplified by making the substitution which was adopted in Section 1.1:

$$N\tau = z$$

and later where λ is used for the input rate:

$$\lambda\tau = z$$

Hence

$$P(B) = (1 - e^{-z})^{(n-1)}$$

It is convenient to introduce some shorthand to denote the relative probabilities as defined above:

$P'(n=4 \rightarrow m=3)$ = "Relative probability that a 4-pulse fixed-number string qualifies as a 3τ fixed-interval string".

Thus if t_1, t_2, t_3 etc., are the successive inter-arrival times:

$$P'(n=4 \rightarrow m=3) = \frac{P(2\tau < (t_1 + t_2 + t_3) < 3\tau)}{(1 - e^{-z})^3} \quad (1.6)$$

$$P'(n=4 \rightarrow m=2) = \frac{P(\tau < (t_1 + t_2 + t_3) < 2\tau)}{(1 - e^{-z})^3} \quad (1.7)$$

$$P'(n=4 \rightarrow m=1) = \frac{P(t_1 + t_2 + t_3 < \tau)}{(1 - e^{-\lambda \tau})^3} \quad (1.8)$$

Since there are no other possibilities in this example for the 4-pulse fixed-number string to qualify as a fixed-interval string, the three relative probabilities should sum to unity in this case.

Similarly the notation $P(n=4 \rightarrow m=3)$ is defined as the absolute probability that a 3τ fixed-interval string has exactly 4 pulses in it whose inter-arrival times are each $< \tau$, as opposed to some other number of pulses, and that it therefore is also a 4-pulse fixed-number string.

1.3.4 FIXED-INTERVAL STRING: SPECIAL CASE WHEN $m=1$

When $m=1$ the fixed-interval string is a single interval τ . In this case the undistorted exponential probability density function for pulse inter-arrival times is valid. The reason is that only times which are $< \tau$ are involved when the fixed-interval itself is of length τ .

The probability density function for the sum of k inter-arrival times in time t is then given by the k -fold convolution of the density function.

$$f_x = \lambda e^{-\lambda t}$$

which gives:

$$f_x(\text{sum } k) = \lambda \frac{(\lambda t)^{(k-1)}}{(k-1)!} \cdot e^{-\lambda t}$$

The probability that this sum is in the interval 0 to τ is then

$$P(\text{sum } k < \tau) = \int_0^{\tau} f_x(\text{sum } k) dt$$

The result, after making the substitution $z = \lambda \tau$, is given by

$$P(\text{sum } k < \tau) = 1 - \left\{ e^{-z} + ze^{-z} + \frac{z^2}{2} e^{-z} + \dots + \frac{z^{(k-1)}}{(k-1)!} \cdot e^{-z} \right\}$$

The terms in the bracket are the familiar probabilities of exactly 0, exactly 1, exactly 2, etc., inter-arrival times in time τ . The probability that the sum of k inter-arrival times is $< \tau$ thus specifically excludes the probabilities of any number less than k , i.e. all those in the bracket.

The probability that a single interval string has at least n pulses, i.e. that $(n-1)$ inter-arrival times are $< \tau$, is thus given by

$$P(n \text{ in } \tau) = 1 - e^{-z} \sum_{k=0}^{n-2} \frac{z^k}{k!} \quad (1.9)$$

The relative probability that an n -pulse string will fit is given by:

$$P'(n \text{ in } \tau) = \frac{1 - e^{-z} \sum_{k=0}^{n-2} \frac{z^k}{k!}}{(1 - e^{-z})^{(n-1)}}$$

Clearly when $n=2$ the single interval fixed-interval string is the only possibility so that the relative probability should be unity. Setting $n=2$ in the formula does indeed give this result.

The occurrence probability for single interval strings, from the previous discussion, will be given by:

$$P(m=1 \text{ string}) = \sum_{n=2}^{\infty} \left\{ 1 - e^{-z} \sum_{k=0}^{n-2} \frac{z^k}{k!} \right\}$$

The probable number of pulses in the single interval string will be given by:

$$\text{Pulses/string } m=1 = \frac{\sum_{n=2}^{\infty} n \left\{ 1 - e^{-z} \sum_{k=0}^{n-2} \frac{z^k}{k!} \right\}}{\sum_{n=2}^{\infty} \left\{ 1 - e^{-z} \sum_{k=0}^{n-2} \frac{z^{(k-1)}}{(k-1)!} \right\}} \quad (1.10)$$

1.3.5 DEFINING PROBABILITY DENSITY FUNCTIONS FOR FIXED-INTERVAL STRINGS

The density functions for the various sums of inter-arrival times involve the convolution of two or more individual density functions. There are several approaches to this problem:

- (i) The use of Laplace transforms
- (ii) Solution of the classical convolution integral
- (iii) Numerical convolution on a digital computer.

Since the evaluation of string occurrence rates involves sums to infinity, a computer solution is required at some stage in any event. The Laplace transform approach while elegant, does not provide any insight into the physical processes involved. Accordingly, in the analysis which follows, the convolution integral is used to evaluate the density functions for the sums of up to 3 inter-arrival times with the aid of Figures 1.3 and 1.4. This rather basic approach to the problem provides a reasonably clear illustration of the effect of the truncated exponential density function on the convolution process.

If one were to generate the density function for the sum of two inter-arrival times ($t_1 + t_2$) with no restriction on the lengths of the times involved, one would use the convolution integral in a conventional way:

Figure 1.4 (a): A second convolution of the truncated function of Figure 1.3(A) with the density function produced by the first convolution (Figure 1.3(C)), generates the density function for the sum of three inter-arrival times each $< \tau$ shown in (b). (c) and (d) illustrate the convolution straddling two different density functions in order to generate the triple sum in the interval τ to 2τ .

$$f_x(t_1 + t_2, t) = \int_0^t \lambda e^{-\lambda(t-x)} \cdot \lambda e^{-\lambda x} \cdot dx$$

This amounts to taking the original density function, turning it around to form a mirror image, shifting it by an arbitrary interval t along the axis, and doing a point-by-point multiplication with the original function. The points involved in the multiplication are denoted by the dummy variable x . These products are summed over the limits $x = 0$ to $x = t$ and the result is the value of the new density function for $(t_1 + t_2)$ at t .

This procedure however will not work for the truncated density function shown in Figure 1.3(a). The product of the two functions only exists in the shaded area shown in Figure 1.3(b), i.e. between the limits $x = (t-\tau)$ to $x = \tau$. Furthermore as the "mirror" function moves from left to right the new density function it produces is being mapped out between $t = \tau$ and $t = 2\tau$ and is not valid outside this range.

Clearly this means that the probability density functions for the various sums of inter-arrival times to fall into predetermined intervals, 2τ , 3τ ... $m\tau$ etc., must be obtained by repeated piecewise convolution of the original truncated exponential shown in Figure 1.3(a), with segments generated as described above.

1.3.6 CALCULATION OF PROBABILITY DENSITY FUNCTIONS FOR FIXED-INTERVAL STRINGS OF ORDER $m > 1$

In this section analytical solutions are derived for the probability density functions in the intervals τ to 2τ and 2τ to 3τ for the sums of two and three inter-arrival times, $(t_1 + t_2)$ and $(t_1 + t_2 + t_3)$.

These probability density functions are denoted by:

$$f_x(\tau < (t_1 + t_2) < 2\tau)$$

$$f_x(2\tau < (t_1 + t_2) < 3\tau)$$

$$f_x(\tau < (t_1 + t_2 + t_3) < 2\tau)$$

$$f_x(2\tau < (t_1 + t_2 + t_3) < 3\tau)$$

From these density functions, the probabilities are then calculated as:

$$P(\text{sum}) = \int_{(m-1)\tau}^{m\tau} f_x(\text{sum})$$

It is also shown that the relative probability expressions derived analytically for a given sum to be in the various intervals add to unity, thereby demonstrating that the derivations are self consistent. For example the 4-pulse fixed-number string (with 3 inter-arrival times) can only fit into fixed-interval strings of order $m=1$, $m=2$ and $m=3$. The relative probabilities for these outcomes must therefore add to unity.

Calculation of the density functions for strings of order $m>3$ simply involves repeated application of the piecewise convolution technique which is developed below for the first three orders. The process is not difficult - merely tedious, and is the sort of calculation which a digital computer can do in seconds. Accordingly a FORTRAN program was written to digitize the truncated exponential shown in Figure 1.4(a), and to perform a numerical convolution repeatedly to produce the segmented probability density functions required. It then integrates the segments for the various sums to produce predicted probabilities for the occurrence rates of the fixed-interval strings of any order m .

1.3.6.1 DERIVATION OF $f_x(\tau < (t_1 + t_2) < 2\tau)$ and $P(\tau < (t_1 + t_2) < 2\tau)$

This requires that the density function $\lambda e^{-\lambda t}$ be convolved with itself once, but such that the process is limited to the shaded portion of Figure 1.3b. This means putting limits $(t-\tau)$ to τ , rather than 0 to t as would be the case for a non-truncated density function. Accordingly we have:

$$f_x(\tau < (t_1 + t_2) < 2\tau) = \int_{t-\tau}^{\tau} \lambda e^{-\lambda(t-x)} \cdot \lambda e^{-\lambda x} dx$$

$$= \int_{t-\tau}^{\tau} \lambda^2 e^{-\lambda t} dx = \left[\lambda^2 x e^{-\lambda t} \right]_{t-\tau}^{\tau}$$

$$f_x(\tau < (t_1 + t_2) < 2\tau) = \lambda^2 e^{-\lambda t} (2\tau - t)$$

The probability that a 3-pulse fixed-number string will be in a 2τ fixed-interval string is the probability that the sum of the two inter-arrival times involved, $(t_1 + t_2)$, will fall in the final interval, i.e. in the interval τ to 2τ . We now have the probability density function for this event, so that its probability is given by:

$$P(\tau < (t_1 + t_2) < 2\tau) = \int_{\tau}^{2\tau} f_x(\tau < (t_1 + t_2) < 2\tau) dt = \int_{\tau}^{2\tau} \lambda^2 e^{-\lambda t} (2\tau - t) dt$$

This integral can be simplified by the substitution $x = \lambda t$. Hence:

$$\begin{aligned} \int_{\tau}^{2\tau} \lambda^2 e^{-\lambda t} \cdot (2\tau - t) dt &= \int_{\lambda\tau}^{2\lambda\tau} 2(\lambda\tau) e^{-x} \cdot dx - \int_{\lambda\tau}^{2\lambda\tau} x e^{-x} \cdot dx \\ &= (2\lambda\tau) \left\{ e^{-\lambda\tau} - e^{-2\lambda\tau} \right\} - \left\{ (\lambda\tau) e^{-\lambda\tau} - 2(\lambda\tau) e^{-2\lambda\tau} - (e^{-\lambda\tau} - e^{-2\lambda\tau}) \right\} \end{aligned}$$

If now we make the substitution $\lambda\tau = z$, to be consistent with the substitution $N\tau = z$ made previously, then the above expression reduces to:

$$\begin{aligned} P(\tau < (t_1 + t_2) < 2\tau) &= 2z \left\{ e^{-z} - e^{-2z} \right\} - \left\{ e^{-z}(1+z) - e^{-2z}(1+2z) \right\} \\ &= e^{-z}(z-1) + e^{-2z} \end{aligned} \quad (1.11)$$

This then is the probability that a 3-pulse fixed-number string will fit into a 2τ fixed-interval string.

1.3.6.2 VERIFICATION OF DERIVATION FOR $P(\tau < (t_1 + t_2) < 2\tau)$

There are only two fixed-interval strings into which a 3-pulse fixed-number string will fit, i.e. a 2τ string or the single interval string. The relative probabilities of these two outcomes must therefore add to unity, i.e.:

$$P'(n=3 \rightarrow m=1) + P'(n=3 \rightarrow m=2) = 1$$

Now we know from equation (1.9) that:

$$P'(n=3 \rightarrow m=1) = \frac{1 - e^{-z} \sum_{k=0}^1 \frac{z^k}{k!}}{(1 - e^{-z})^2} = \frac{1 - e^{-z}(1+z)}{(1 - e^{-z})^2}$$

Hence:

$$P'(n=3 \rightarrow m=2) = 1 - \frac{1 - e^{-z}(1+z)}{(1 - e^{-z})^2} = \frac{e^{-z}(z-1) + e^{-2z}}{(1 - e^{-z})^2}$$

The numerator is identical with the absolute probability expression derived in equation (1.11), so that the derivation used there was valid.

1.3.6.3 DERIVATION OF $f_x(2\tau < (t_1 + t_2 + t_3) < 3\tau)$ and $P(2\tau < (t_1 + t_2 + t_3) < 3\tau)$

Reference to Figure 1.4(A) shows the convolution process required to generate this density function. The original truncated exponential is convolved with the density function generated previously, $f_x(\tau < (t_1 + t_2) < 2\tau)$, between limits $(t-\tau)$ to 2τ . The required density function is thus given by:

$$f_x(2\tau < (t_1 + t_2 + t_3) < 3\tau) = \int_{t-\tau}^{2\tau} \lambda e^{-\lambda(t-x)} \cdot \lambda^2 e^{-\lambda x} \cdot (2\tau - x) dx$$

The solution follows a similar pattern to the one presented in Section 1.3.6.1, and is given by:

$$f_x(2\tau < (t_1 + t_2 + t_3) < 3\tau) = \lambda^3 e^{-\lambda t} \left\{ \frac{9}{2} t^2 - 3\lambda t + t^2/2 \right\}$$

The probability that the sum of the three inter-arrival times is in the region 2τ to 3τ , is the probability that a 4-pulse fixed-number string will also qualify as a 3τ fixed-interval string.

$$P(2\tau < (t_1 + t_2 + t_3) < 3\tau) = \int_{2\tau}^{3\tau} f_x(2\tau < (t_1 + t_2 + t_3) < 3\tau) \cdot dt$$

Again the solution is straight forward and after the limits are inserted and the substitution $\lambda\tau = z$ is made the result is:

$$P(n=4 \rightarrow m=3) = P(2\tau < (t_1 + t_2 + t_3) < 3\tau) = e^{-2z}(1 - z + z^2/2) - e^{-3z}$$

The relative probability of this event, given that we have the 4-pulse string, is given by:

$$P'(n=4 \rightarrow m=3) = \frac{e^{-2z}(1-z+z^2/2) - e^{-3z}}{(1-e^{-z})^3} \quad (1.12)$$

1.3.6.4 DERIVATION OF $f_x(\tau < (t_1 + t_2 + t_3) < 2\tau)$ and of $P(\tau < (t_1 + t_2 + t_3) < 2\tau)$

These functions give the probability that a 4-pulse string (with 3 inter-arrival times) will also qualify as a 2τ string, i.e. the probability that the sum of the inter-arrival times falls in the final interval τ to 2τ of the 2τ fixed-interval string.

Reference to Figure 1.4(c) shows that the convolution involves the two different density functions for the sum of two inter-arrival times in the segments 0 to τ , and τ to 2τ .

The integral required is the sum of areas A and B at a given value of t . It can be obtained by separate convolutions of the truncated exponential with the two density functions; the first over the limits $(t - \tau)$ to τ , and the second over the limits τ to t .

$$\text{Area A} = \int_{t-\tau}^{\tau} \lambda e^{-\lambda(t-x)} \cdot \lambda^2 x e^{-\lambda x} dx = I_1$$

$$\text{Area B} = \int_{\tau}^t \lambda e^{-\lambda(t-x)} \cdot \lambda^2 e^{-\lambda x} \cdot (2\tau - x) dx = I_2$$

$$f_x(\tau < (t_1 + t_2 + t_3) < 2\tau) = I_1 + I_2 = \lambda^3 e^{-\lambda t} \left\{ 3\tau t - t^2 - 3\tau^2/2 \right\}$$

Integration of this density function over the limits τ to 2τ leads to the result:

$$\begin{aligned}
 P(n=4 \rightarrow m=2) &= P(\tau < (t_1 + t_2 + t_3) < 2\tau) \\
 &= (e^{-z} - e^{-2z})(z^2/2 + z - 2) + 2ze^{-2z}
 \end{aligned} \tag{1.13}$$

and the relative probability of the 4-pulse string qualifying as a 2τ string as:

$$P'(n=4 \rightarrow m=2) = \frac{(e^{-z} - e^{-2z})(z^2/2 + z - 2) + 2ze^{-2z}}{(1 - e^{-z})^3}$$

From equation (1.9) the relative probability that the 4-pulse fixed-number string will qualify as a single interval string is given by:

$$P'(n=4 \rightarrow m=1) = \frac{1 - e^{-z}(1 + z + z^2/2)}{(1 - e^{-z})^3}$$

We now have the relative probabilities:

$$P'(n=4 \rightarrow m=3) ; P'(n=4 \rightarrow m=2) ; P'(n=4 \rightarrow m=1)$$

from equations (1.12), (1.13) and (1.6) respectively. Since these are the only possible outcomes for a 4-pulse fixed-number string, they should add to unity. Addition of the numerators gives a sum of $1 - 3e^{-z} + 3e^{-2z} - e^{-3z}$ which is the denominator $(1 - e^{-z})^3$. This confirms that the convolution procedure is analytically self consistent; the proportions are also in close agreement with experimental data which will be presented and discussed in a later chapter.

CHAPTER 2 DEVELOPMENT OF MODEL THEORY FOR PROBABILITY MEASUREMENTS

2.0 INTRODUCTION

The theory governing the probabilities of coincidences and multiple pulse strings was developed in the previous chapter. This chapter is devoted to the development of practical tools and techniques for making experimental measurements of these probabilities.

The accurate and reliable measurement of random events is an inherently difficult task, requiring methods which are amenable to step by step verification. Logic circuits were devised to record compound events and their efficacy was tested initially on uncomplicated events for which the occurrence probabilities were well known. In the course of this exercise it became evident that the necessity to allow for overlapping events could result in ambiguities in circuit operation causing erroneous results even though the circuit was performing in accord with the design criteria.

As a result of this experience logic circuits (even apparently trivial ones) were analyzed in a rigorous manner using Markov process theory. In this way the probabilities of a logic circuit being in any one of the various states that it must go through in order to record a specified event, could be predicted unambiguously by measurements of the state transition rates. These in turn could be equated with the transition probabilities predicted from the Poisson properties of the random pulse sequence.

Examples are presented of two circuits designed to measure the probability of occurrence of a simple single coincidence; i.e. the event that a pair of pulses with separation $< \tau$ is preceded and followed by spaces $> \tau$. Both circuits are capable of making the measurement but a Markov analysis shows that while in one case a simple measurement of the success rate as indicated by a single output is sufficient, in the other circuit additional parameters must also be recorded. The underlying

Markov theory is reviewed and its application to the problem of these random event measurements is explained in detail. A streamlined version of the Markov analysis is developed which simplifies the procedure involved in generating formulae for experimental measurements using logic circuits. Theoretical expressions are derived which allow transition rates for the two circuits to be predicted for comparison with experimental data at various input rates.

2.1 SINGLE COINCIDENCE MEASUREMENTS

A single coincidence is defined for the purposes of these measurements as a pair of pulses with a separation $< \tau$ which is preceded and followed by a time interval or space which is $> \tau$ where τ is the time within which pulse arrivals are regarded as being coincident. This event is actually included in the class of multiple pulse strings with fixed-numbers of pulses. In this case the number is two. Because it is the simplest event in the class it is used as an example to demonstrate the theory which is developed in the following sections. It is therefore treated as being separate from the multiple string measurements.

2.1.1 SUB-EVENT PROBABILITIES

The compound event to be considered is composed of three pulse inter-arrival times, each with a well known probability of occurrence. If we denote the compound event as (A, B, C) where A, B and C are the individual sub events, then the probability of sub event A, $P(A)$, is the probability that the first inter-arrival time is $> \tau$.

$$\text{i.e. } P(A) = P(t_1 > \tau)$$

$$\text{and } P(B) = P(t_2 < \tau)$$

$$P(C) = P(t_3 > \tau)$$

Since inter-arrival times in a Poisson distributed pulse train are statistically independent the probability of the compound event is given by:

$$P(A, B, C) = P(A).P(B).P(C).$$

The inter-arrival times also have an exponential distribution given by:

$$P(T < t) = 1 - e^{-Nt}$$

or $P(T > t) = e^{-Nt}$

From which

$$P(ABC) = e^{-N\tau} \cdot (1 - e^{-N\tau}) \cdot e^{-N\tau}$$

$$P(ABC) = e^{-2N\tau} \cdot (1 - e^{-N\tau}) \quad (2.1)$$

2.1.2 COMPLICATIONS IN THE MEASUREMENT OF OVERLAPPING EVENT PROBABILITIES

The probability of the single coincidence event, as given by equation (2.1), seems straightforward enough - until one attempts to make a real-world measurement of it. The question then arises as to what the sample space is, since one cannot define probabilities in isolation. Equation (2.1) is based implicitly on the relative frequency definition of probability, which is given by:

$$P(\text{Event}) = (\text{No. of Successes})/(\text{No. of Attempts})$$

Equation (1.4) then tells us that the occurrence rate of the (single coincidence) event will be given by:

$$\text{Occurrence rate (Event)} = N \cdot P(\text{Event})$$

where N is the input pulse rate.

The single coincidence event is preceded and followed by spaces $> \tau$ which may be shared with other single coincidences. The events may

therefore overlap. This can complicate the design of circuits devised to record successes and attempts. One of the objects of the experimental work in this instance was to make experimental measurements of all the types of multiple pulse strings, (of which the single coincidence is the simplest example), and to demonstrate that this could be accomplished by two different circuit design approaches.

The two circuits are discussed at the block diagram level in the next section. One of them completely ignores the possibility of overlapping events, and as a result the observed success rate for detection of single coincidences is less than the true occurrence rate for these events. Other parameters have to be measured which allow the number of attempts and hence the true probability $P(ABC)$ of the event to be computed. The second circuit is designed to recognize and detect overlapping events and the observed success rate is shown to be the true occurrence rate of the events it is measuring.

The advantage of the second circuit is that only one counter is required to make the measurements, while the first one requires a minimum of two counters making two simultaneous measurements. However the more pedantic approach used with the first circuit can easily be extended to the measurement of more complex overlapping compound events involving any number of sub-events. It is also easier to implement from a circuit design viewpoint.

2.2 LOGIC CIRCUIT MODELS

This section describes the two logic circuits which were designed to record the successes and/or failures of each sub-event in a specified compound random event occurring in a pulse train from a random source. The description is in terms of basic models of the type often referred to as "State Machines". The two circuits, or models, are essentially variants of the same basic concept and are referred to hereafter as "Circuit I" and "Circuit II".

The basic circuit was designed to monitor the pulse train from a random source and to generate a logic pulse whenever the compound event (A, B, C) occurs. The circuit has three states A, B, C, corresponding to three consecutive inter-arrival times. A transition from A to B occurs if the first inter-arrival time is $>\tau$ where τ is a preset time interval generated by a counter and high frequency clock. A transition from B to C occurs if the next inter-arrival time is $<\tau$, and a transition from C occurs a time τ after entering state C. If no pulse arrives while the circuit is in state C, then a "success" (S) logic output goes true. The block diagrams for the two versions of this circuit, I and II, are shown in Figure 2.1.

In Circuit I a reset to state A is made whenever the criterion for an individual event A or B or C is not met. These failures are denoted by F_A , F_B and F_C respectively. The circuit included outputs for the measurement of the number of occurrences of these, as well as of the successes S indicating the occurrence of the complete event (A, B, C). This circuit thus ignores the fact that the final space $>\tau$ in a single coincidence might be the initial space for another one.

Circuit II is a modified version of Circuit I designed to ensure that overlapping events (A, B, C) are properly recorded. An example of such an overlap is shown in the sequence of four pulses of Figure 2.2(a). In this sequence there are two single coincidences which meet the three criteria and which could result in false conversions by an ADC. A second overlap example is shown in the sequence of Figure 2.2(b). There are two single coincidences, one in segment 3 and the other in segment 5. The first pulse would move Circuit I from state A to B (segment 2). With the circuit in state B (segment 2), a transition to state A would occur because of the failure to meet the B criterion (arrival time for next pulse $<\tau$). Failure to meet the A criterion (segment 3) would then result in the first single coincidence being unregistered. Only the second one would be recorded where the A, B, C, criteria are met (segments 4, 5 and 6).

Figure 2.1: Block diagrams for logic Circuits I and II. A, B, C, are the three states through which circuits must progress in order to record a successful event "S". Failure to meet criterion for moving to next state results in outputs F_a , F_b or F_c with a return to the state indicated.

Figure 2.2 (a): Coincidence events defined as space pulse-pair space (A, B, C) overlap. **(b)** A single pulse event and two coincidences overlap. Circuit I would miss the first coincidence.

Figure 2.3 (a): Response of Circuit I to random sequence with four overlapping coincidence events, two are missed. **(b)** Response of Circuit II to same sequence, all are detected.

The reason for including Circuit I in the experimental measurements will become apparent in subsequent Sections of this Chapter. It will be shown that a relatively simple analysis based on Markov process theory can be used to derive correct results from this model, despite the inability to detect overlapping events. It will also be demonstrated that the design concept for Circuit I allows it to be extended to any number of states for measurement of more complicated compound random events, whereas Circuit II by and large is a specific single-purpose design.

The block diagram for Circuit II shows that a reset to state B is made if (i) an event (ABC) is recorded, or (ii) if a failure in state B occurs. In both these cases a period of τ will already have elapsed since the previous pulse arrival, which is equivalent to meeting the state A criterion. This modification enables overlapping events to be recognized, thus ensuring that all bona-fide single coincidences in a pulse train are recorded.

In both circuits the transitions are triggered before the prescribed time τ has elapsed for a given state if the arrival of a pulse within that time meets the criterion for success or failure. The implications of this are shown in Figure 2.3(a), (b), where the states of the two circuits for the same random pulse sequence are illustrated.

In both circuits the τ time interval generator is reset on each pulse arrival. Both circuits remain idle once the state A criterion is met, until the arrival of the next pulse triggers a transition to state B.

It can be seen from the two sequences that Circuit II would detect all four events (A, B, C) in the sequence, while Circuit I would miss two of them.

2.2.1 LOGICAL RELATIONSHIPS FOR CIRCUIT I

The block diagram for Circuit I in Figure 2.1 shows arrows pointing into each state A, B and C, and other arrows pointing away from each state. The sum of the quantities (F_a , F_b , etc.) associated with each arrow

pointing into a state, including arrows that leave and return to the same state, represents the number of times the state is entered. In the case of state A this sum is $F_a + F_b + F_c + S$. Let these sums be denoted by A_n , B_n , and C_n for the three states. Then if a random pulse train is applied for a time long enough for steady-state conditions to become established, the following relationships will apply:

$$A_n = F_a + F_b + F_c + S \quad (2.2)$$

$$B_n = F_b + F_c + S = A_n - F_a \quad (2.3)$$

$$C_n = F_c + S = A_n - F_a - F_b \quad (2.4)$$

It is also apparent from the original criteria for the simple single coincidence that the numbers of input pulses required to produce one of each of them are as follows:

Event	No. of Pulses
F_a	1
F_b	1
F_c	3
S	2

If the average input pulse rate is N and if f_a , f_b , f_c , S are now used to denote occurrence rates, or more accurately transition rates, then

$$N = (f_a + f_b + 3f_c + 2S) \quad (2.5)$$

We thus have a relation between N , the input pulse rate and the state transition rates. However it is only valid for the particular event (the simple single coincidence) which is being used here to demonstrate the basic theory. If now A_n , B_n and C_n are used to denote individual state transition rates, rather than numbers of times a state is entered, then the overall transition rate M is given by the sum of (2.2), (2.3) and (2.4):

$$M = A_n + B_n + C_n = f_a + 2f_b + 3f_c + 3S \quad (2.6)$$

This overall transition rate was arrived at without any consideration of the number of pulses involved in the event and thus holds true for Circuit I no matter what the nature of the event detected.

Hence from (2.5) and (2.6)

$$M = N + f_b + S \quad (2.7)$$

Thus the overall transition rate exceeds the input pulse rate by $(f_b + S)$. Inspection of Figure 2.3(a), the response of Circuit I to a random pulse sequence, shows how this comes about. f_b represents unsuccessful transitions from state B, and S represents successful transitions from state C. In both cases Figure 2.3(a) shows that state transitions $B \rightarrow A$ and $C \rightarrow A$ occur without pulses. There are thus 12 transitions in Figure 2.3(a), but only 10 pulses. This is in accord with the original criteria and accounts for the excess of transition rates over input pulse rate shown by equation (2.7).

The relative probabilities of the circuit being in any one of the three states can now be determined experimentally by measurements of the transition rates f_a , f_b , f_c and S , and computed according to:

$$P(A) = \frac{A_n}{M} = \frac{f_a + f_b + f_c + S}{f_a + 2f_b + 3f_c + 3S} \quad (2.8)$$

$$P(B) = \frac{B_n}{M} = \frac{f_b + f_c + S}{f_a + 2f_b + 3f_c + 3S} \quad (2.9)$$

$$P(C) = \frac{C_n}{M} = \frac{f_c + S}{f_a + 2f_b + 3f_c + 3S} \quad (2.10)$$

where $P(A) + P(B) + P(C) = 1$

2.2.2 LOGICAL RELATIONSHIPS FOR CIRCUIT II

Circuit II has at least one thing in common with Circuit I, the numbers of pulses required to produce the transitions f_a , f_b , f_c and S are the same, i.e. 1, 1, 3, and 2 respectively for the single coincidence, hence the same equation (2.5) applies in relating the input pulse rate N to the transition rates.

The transition rates by inspection of the block diagram in Figure 2.1 for this circuit are given by:

$$\begin{aligned}
 A_n &= f_a + f_c \\
 B_n &= A_n + f_b + S - f_a = f_b + f_c + S \\
 C_n &= B_n - f_b = f_c + S \\
 M &= A_n + B_n + C_n \\
 M &= f_a + f_b + 3f_c + 2S
 \end{aligned} \tag{2.11}$$

However from the previous discussion we see that the equation for M is identical to equation (2.5) for the input pulse rate, so that for this particular event

$$M = N$$

Hence the relative probabilities can be computed from the experimental data as

$$\begin{aligned}
 P(A) &= A_n/N = (f_a + f_c)/N \\
 P(B) &= B_n/N = (f_b + f_c + S)/N \\
 P(C) &= C_n/N = (f_c + S)/N
 \end{aligned}$$

It is important to point out here that for a given input pulse rate N , the individual values for f_a , f_b , f_c and S as measured by the two circuits will be different, but the value of N arrived at by evaluating the expression:

$$N = (f_a + f_b + 3f_c + 2S)$$

using data from either circuit will always be correct, for the simple single coincidence case now being used as an example.

The fundamental operation of the two circuits has been explored in some detail from first principles using the simple single coincidence, (a fixed-number pulse string of order $n = 2$), as an example. This has resulted in the expressions A_n/M , B_n/M and C_n/M which give the probabilities of the circuits being in states A, B or C. These expressions are in terms of the transition rates (parameters which can be measured experimentally) f_a , f_b , f_c and S . A significant parameter in this analysis is the total transition rate M . The expression for this in each case has been shown to be independent of the nature of the event being detected, since it was not derived on the basis of the number of input pulses required to produce the transitions f_a , f_b , f_c and S . It is shown in a later section that M is a parameter which is needed to predict these transition rates, starting only with the event probabilities in terms of the exponential functions.

We still do not however have a link between the observed success rates S for each circuit and the occurrence rate for the single coincidence event we are using as an example. It is shown in the next section that this missing link is provided by the application of Markov process theory to the operation of the two circuits. The state probability expressions derived above in terms of the transition rates come out again in the Markov analysis, but only as the result of some very lengthy and tedious matrix algebra. It is shown in a subsequent section that a simplified (but still rigorous) Markov analysis can be used to generate formulae for event occurrence rates using the measured transition rates, without the necessity for a tedious matrix solution.

2.3 APPLICATIONS OF MARKOV PROCESS THEORY TO CIRCUIT STATE PROBABILITIES

Markov process theory is applicable to situations where the stochastic process involved is "memoryless", with the probabilities of events being independent. The successive states of the logic Circuits I and II are related to events in a Poisson process which meets the criterion for a Markov process. (See references 79, 80 and 81 for texts which cover the theory of Markov chains.)

Circuits I and II block diagrams can be redrawn in terms of equivalent Markov chains with the connecting links redefined as transition probabilities as shown in Figures 2.4 and 2.5. These probabilities are known from the original criteria. They are listed below in Tables 2.1 and 2.2 for both the measured parameters f_a , f_b , f_c and S and for the theoretical ones as derived from the exponential distribution of inter-arrival times. The guiding principle in assigning transition probabilities is that the sum of probabilities for leaving a state to go to another state, or to return to the same state, will be unity.

As an example consider the probability of leaving state A in Circuit I. From Figure 2.1 we see that there are just two routes, the one which leaves (and returns) with transition rate f_a , and the one which goes to state B with transition rate $(f_b + f_c + S)$. The total transition rate for leaving state A is thus $(f_a + f_b + f_c + S)$. Since the probabilities for leaving must sum to unity, the proportions for the two possible routes are $f_a/(f_a + f_b + f_c + S)$ for the first one, and $(f_b + f_c + S)/(f_a + f_b + f_c + S)$ for the second one.

CIRCUIT I AS A MARKOV CHAIN

Figure 2.4

Table 2.1: Circuit I Transition Probabilities

$P_{aa} = 1 - P_{ab} = 1 - e^{-Z}$	$f_a / (f_a + f_b + f_c + S)$
$P_{ab} = e^{-Z}$	$(f_b + f_c + S) / (f_a + f_b + f_c + S)$
$P_{ac} = 0$	0
$P_{ba} = 1 - P_{bc} = 1 - (1 - e^{-Z}) = e^{-Z}$	$f_b / (f_b + f_c + S)$
$P_{bb} = 0$	0
$P_{bc} = 1 - e^{-Z}$	$(f_c + S) / (f_b + f_c + S)$
$P_{ca} = P_{ca}(F) + P_{ca}(S) = (1 - e^{-Z}) + e^{-Z} = 1$	$f_c / (f_c + S) + S / (f_c + S) = 1$
$P_{cb} = 0$	0
$P_{cc} = 0$	0

CIRCUIT II AS A MARKOV CHAIN

Figure 2.5

Table 2.2: Circuit II Transition Probabilities

$P_{aa} = 1 - P_{ab} = 1 - e^{-Z}$	$f_a / (f_c + f_a)$
$P_{ab} = e^{-Z}$	$f_c / (f_c + f_a)$
$P_{ac} = 0$	0
$P_{ba} = 0$	0
$P_{bb} = 1 - P_{bc} = 1 - (1 - e^{-Z}) = e^{-Z}$	$f_b / (f_b + f_c + S)$
$P_{bc} = 1 - e^{-Z}$	$(f_c + S) / (f_b + f_c + S)$
$P_{ca} = 1 - e^{-Z}$ (fails state C criterion)	$f_c / (f_c + S)$
$P_{cb} = e^{-Z}$ (successful exit from state C)	$S / (f_c + S)$
$P_{cc} = 0$	0

2.3.1 THE STEADY STATE MARKOV MATRIX EQUATION

Markov theory predicts that a system which starts in state i has a probability $\phi_{ij}(n)$ of reaching state j after n steps which is given by

$$\phi_{ij}(n) = \sum_k P_{kj} \cdot \phi_{ik}(n-1)$$

where $\phi_{ik}(n-1)$ is the probability of reaching state k after $(n-1)$ steps and P_{kj} is the probability of a transition from state k to state j .

This can be recast as a matrix equation:

$$\underline{\phi}(n) = \underline{\phi}(n-1) \cdot \underline{P}$$

where $\underline{P}$ is the matrix of transition probabilities, and $\underline{\phi}(n)$, $\underline{\phi}(n-1)$ are row

matrices whose elements are the probabilities of the system being in states 1, 2, ..., m , where m is the total number of possible states. $\underline{\phi}(n)$ is the row matrix for these probabilities after n steps and $\underline{\phi}(n-1)$ is the row matrix for the probabilities before the n th step.

As $n \rightarrow \infty$, $\underline{\phi}(n-1) \rightarrow \underline{\phi}(n)$ as the system assumes steady state conditions. When this condition is reached, then the probabilities of the system being in a given state are found as the solution of the equation:

$$\underline{\phi} = \underline{\phi} \underline{P}$$

Denoting the probabilities for the circuits to be in state A, B or C as π_a , π_b , π_c , then for steady state conditions, i.e. after a large number of pulses has been applied, we can write the matrix equation:

$$(\pi_a \ \pi_b \ \pi_c) = (\pi_a \ \pi_b \ \pi_c) \begin{bmatrix} P_{aa} & P_{ab} & P_{ac} \\ P_{ba} & P_{bb} & P_{bc} \\ P_{ca} & P_{cb} & P_{cc} \end{bmatrix}$$

The probabilities π_a , π_b , and π_c are the same ones as those derived from the measured parameters and denoted as $P(A)$, $P(B)$ and $P(C)$ in equations (2.8), (2.9) and (2.10). The matrix equation can be solved by inserting the values for P_{aa} , P_{ab} , etc., from Tables 2.1 or 2.2 in terms of either the exponential expressions, or the measured parameters. The latter solution is tedious but leads to the same results as were obtained in equations (2.8), (2.9) and (2.10) by considering the probabilities as A_n/M , B_n/M and C_n/M .

2.3.2 MARKOV ANALYSIS FOR CIRCUIT I

Substituting for P_{aa} , P_{ab} , etc., now in terms of the exponential expressions,

for Circuit I we have (from Table 2.1):

$$(\pi_a \ \pi_b \ \pi_c) = (\pi_a \ \pi_b \ \pi_c) \begin{bmatrix} 1-e^{-Z} & e^{-Z} & 0 \\ e^{-Z} & 0 & 1-e^{-Z} \\ 1 & 0 & 0 \end{bmatrix}$$

This gives three equations:

$$\pi_a = \pi_a (1-e^{-Z}) + \pi_b e^{-Z} + \pi_c \quad (2.12)$$

$$\pi_b = \pi_a e^{-Z} \quad (2.13)$$

$$\pi_c = \pi_b (1-e^{-Z}) \quad (2.14)$$

They are linearly dependent since the determinant of the P matrix has a column with identical elements (by adding cols. 2 and 3 to col. 1). Equation (2.12) is the sum of (2.13) and (2.14) and yields no new information. However, the individual probabilities must add to unity, i.e.

$$\pi_a + \pi_b + \pi_c = 1$$

$$\text{Hence } \pi_a (1 + e^{-Z} + e^{-Z} (1-e^{-Z})) = 1$$

This leads to the solutions which are shown below, together with those in terms of the measured parameters for reference.

$$\pi_a = \frac{1}{1 + 2e^{-z} - e^{-2z}} = \frac{f_a + f_b + f_c + S}{f_a + f_b + 3f_c + 3S}$$

$$\pi_b = \frac{e^{-z}}{1 + 2e^{-z} - e^{-2z}} = \frac{f_b + f_c + S}{f_a + f_b + 3f_c + 3S}$$

$$\pi_c = \frac{e^{-z}(1 - e^{-z})}{1 + 2e^{-z} - e^{-2z}} = \frac{f_c + S}{f_a + f_b + 3f_c + 3S}$$

We are now in a position to relate the observed success rate for Circuit I to the actual occurrence rate of the single coincidence events (ABC) in the incoming pulse train. The S transition, which when fed to a suitable counter is used as a measure of the observed success rate, occurs whenever a successful exit is made from state C. The probability of this transition is denoted by $P_{ca}(S)$ in Table 2.1 and it signifies that a complete coincidence event has been detected. We have denoted the real probability of a single coincidence event as $P(ABC)$ and it is given in equation (2.1) as

$$P(ABC) = e^{-2z}(1 - e^{-z})$$

Let us denote the probability of this same event as measured by Circuit I as $P'(ABC)$. This will be the probability of the circuit being in state C, times the probability of a successful exit (transition) from that state, which is:

$$P'(ABC) = \pi_c \cdot P_{ca}(S)$$

From Table 2.1 we have $P_{ca}(S) = e^{-z}$, and we have π_c from the solutions above, hence:

$$P'(ABC) = \pi_c \cdot P_{ca}(S) = \frac{e^{-z} \cdot (1 - e^{-z}) \cdot e^{-z}}{1 + 2e^{-z} - e^{-2z}} = \frac{P(ABC)}{1 + 2e^{-z} - e^{-2z}} \quad (2.15)$$

This is not the result predicted from equation (2.1) and the reason is that Circuit I does not recognize overlapping events. It is apparent however that if this result is divided by π_a , then the correct result is obtained.

In terms of the measured parameters this becomes:

$$P'(ABC) = \pi_c \cdot P_{ca}(S) = \frac{f_c + S}{f_a + f_b + 3f_c + 3S} \cdot \frac{S}{f_c + S} = \frac{S}{f_a + f_b + 3f_c + 3S}$$

$$\pi_a = \frac{1}{1 + 2e^{-x} - e^{-2x}} = \frac{f_a + f_b + f_c + S}{f_a + f_b + 3f_c + 3S}$$

when $P'(ABC)$ is divided by π_a we obtain:

$$\begin{aligned} P'(ABC) \cdot \frac{1}{\pi_a} &= \frac{S}{f_a + f_b + 3f_c + 3S} \cdot \frac{f_a + f_b + 3f_c + 3S}{f_a + f_b + f_c + S} \\ &= \frac{S}{f_a + f_b + f_c + S} = \frac{\text{Successes}}{\text{Attempts}} = P(ABC) \end{aligned}$$

The correct probability, $P(ABC)$, can thus be obtained from Circuit I experimental measurements using the expression:

$$P(ABC) = \frac{S}{f_a + f_b + f_c + S}$$

The key point here is that the correct result can be obtained from Circuit I if the right sample space is used. The implementation of Circuit I is significantly simpler from a circuit design viewpoint than it is for Circuit II.

2.3.3 MARKOV ANALYSIS FOR CIRCUIT II

For Circuit II the Markov steady state matrix equation is

$$(\pi_a \ \pi_b \ \pi_c) = (\pi_a \ \pi_b \ \pi_c) \begin{bmatrix} 1-e^{-Z} & e^{-Z} & 0 \\ 0 & e^{-Z} & 1-e^{-Z} \\ e^{-Z} & 1-e^{-Z} & 0 \end{bmatrix}$$

Again the three equations will be linearly dependent and will require the condition $\pi_a + \pi_b + \pi_c = 1$ for a unique solution.

From this:

$$\pi_a = (\pi_a + \pi_c) (1-e^{-Z})$$

$$\pi_b = e^{-Z} (\pi_a + \pi_b + \pi_c) \quad \pi_b (1-e^{-Z}) = e^{-Z} (\pi_a + \pi_c)$$

$$\pi_c = \pi_b (1-e^{-Z})$$

$$\pi_a + \pi_b + \pi_c = 1$$

These give solutions:

$$\pi_a = 1-2e^{-Z} + e^{-2Z} = (f_a + f_c)/(f_a + f_b + 3f_c + 2S) \quad (2.16)$$

$$\pi_b = e^{-Z} = (f_b + f_c + S)/(f_a + f_b + 3f_c + 2S) \quad (2.17)$$

$$\pi_c = e^{-Z} (1-e^{-Z}) = (f_c + S)/(f_a + f_b + 3f_c + 2S) \quad (2.18)$$

The probability of detection of the event (A, B, C) as measured by the observed success rate for Circuit II, $P''(ABC)$, is the probability of a successful exit from state C; i.e. the probability of being in state C times the probability of a successful transition, P_{cb} . Hence:

$$P''(ABC) = \pi_c \cdot P_{cb} = \pi_c \cdot e^{-Z}$$

$$P''(ABC) = e^{-Z} (1-e^{-Z}) e^{-Z}$$

This result is in agreement with equation (2.1) because overlapping events have been accounted for in Circuit II, and for the particular event we are dealing with here the total transition rate M is equal to the input pulse rate N.

Substituting for π_c and P_{cb} in terms of the measured parameters we have in this instance:

$$P''(ABC) = \pi_c \cdot P_{cb} = \frac{f_c + S}{(f_a + f_b + 3f_c + 2S)} \cdot \frac{S}{f_c + S} = \frac{S}{(f_a + f_b + 3f_c + 2S)} = \frac{S}{N}$$

Hence Circuit II will give the probability of single coincidences as defined by the original criteria as the ratio S/N i.e. the observed success rate for Circuit II is the actual occurrence rate for the single coincidence events being measured:

$$P(ABC) = S/N$$

2.4 A SIMPLIFIED MARKOV ANALYSIS FOR RANDOM EVENT DETECTION LOGIC CIRCUITS

In the previous section the matrix equations for Circuit I and Circuit II were solved for a specific set of event probabilities. It will now be shown that a solution in terms of the transition probability notation P_{aa} , P_{ab} , etc., leads to a more direct method for computation of the probability of a compound event in terms of the measured parameters for these and any other circuits.

2.4.1 CIRCUIT I: SIMPLIFIED MARKOV ANALYSIS

In the case of Circuit I the matrix equation is:

$$(\pi_a \ \pi_b \ \pi_c) = (\pi_a \ \pi_b \ \pi_c) \begin{bmatrix} P_{aa} & P_{ab} & 0 \\ P_{ba} & 0 & P_{bc} \\ 1 & 0 & 0 \end{bmatrix}$$

This gives solutions:

$$\pi_a = 1/(1 + P_{ab}(1 + P_{bc})) \quad (2.19)$$

$$\pi_b = P_{ab}/(1 + P_{ab}(1 + P_{bc})) \quad (2.20)$$

$$\pi_c = P_{ab} \cdot P_{bc}/(1 + P_{ab}(1 + P_{bc})) \quad (2.21)$$

The probability $P(ABC)$ of the event we are interested in is the probability of successful consecutive transitions from state A to B to C and to A again. From Figure 2.4 this means that:

$$P(ABC) = P_{ab} \cdot P_{bc} \cdot P_{ca}(S)$$

It is apparent as before with Circuit I, that this result is obtained only indirectly as the probability of being in state C times the probability of a successful transition from state C, since multiplying both sides of equation 2.21 by $P_{ca}(S)$ gives:

$$\pi_c \cdot P_{ca}(S) = P_{ab} \cdot P_{bc} \cdot P_{ca}(S)/(1 + P_{ab}(1 + P_{bc}))$$

from which as previously:

$$P_{ab} \cdot P_{bc} \cdot P_{ca}(S) = \pi_c \cdot P_{ca}(S)/\pi_a \quad (2.22)$$

A direct evaluation of this event probability can be obtained by substituting the measured parameter equivalents for P_{ab} , P_{bc} and $P_{ca}(S)$ from Table 2.1 into the left hand side of (2.22)

$$P_{ab} \cdot P_{bc} \cdot P_{ca}(S) = \frac{f_b + f_c + S}{f_a + f_b + f_c + S} \cdot \frac{f_c + S}{f_b + f_c + S} \cdot \frac{S}{f_c + S} \quad (2.23)$$

The numerator of each term cancels with the denominator of the one following giving the result obtained previously, i.e.

$$P(ABC) = P_{ab} \cdot P_{bc} \cdot P_{ca}(S) = \frac{S}{f_a + f_b + f_c + S}$$

2.4.1.1 CIRCUIT I AS A CANONICAL MODEL FOR MULTIPLE EVENT DETECTION

It is now clear from the form of (2.23) that the three-state model of Circuit I can be extended to any number of states, provided that a failure to meet the criterion for the sub-event represented by any given state always forces a return to state A. Thus Circuit I is the basis for a canonical form of logic circuit designed to detect any multi-component compound random event, and to provide a reliable method of measuring the probability, which would be given by:

$$P(ABC \dots N) = P_{ab} \cdot P_{bc} \cdot P_{cd} \dots P_{na}(S) = \frac{S}{f_a + f_b + f_c + \dots + f_n + S} \quad (2.24)$$

In moving from the specific to the general as has been done, the nature of the compound random event to be detected has been left unspecified. The only assumption implicit in the above argument is that the compound event consists of sub-events which are consecutive in time and which index the circuit from one state to the next when detected.

2.4.2 CIRCUIT II: SIMPLIFIED MARKOV ANALYSIS

Circuit II was designed to deal with a specific type of event and as such does not lend itself to more general application. The matrix equation for it in terms of transition probability notation is:

$$(\pi_a \ \pi_b \ \pi_c) = (\pi_a \ \pi_b \ \pi_c) \begin{bmatrix} P_{aa} & P_{ab} & 0 \\ 0 & P_{bb} & P_{bc} \\ P_{ca} & P_{cb} & 0 \end{bmatrix}$$

This leads to solutions:

$$\pi_a = P_{ca} \cdot P_{bc}/D$$

$$\pi_b = (1-P_{aa})/D = P_{ab}/D$$

$$\pi_c = (1-P_{aa}) \cdot P_{bc}/D = P_{ab} \cdot P_{bc}/D \quad (2.25)$$

where $D = (P_{ca} \cdot P_{bc} + (1 - P_{aa})(1 + P_{bc}))$

Multiplying both sides of (2.25) by P_{cb} and rearranging we get:

$$P(ABC) = P_{ab} \cdot P_{bc} \cdot P_{cb} = \pi_c \cdot P_{cb} \cdot D$$

In the specific solution for the single coincidence, insertion of the exponential values for P_{ca} , P_{bc} , etc., from Table 2.2 for the factor D , shows that it evaluates to unity. Substitution of the measured parameter equivalents for π_c , P_{cb} and D from equation (2.18) and Table 2.2 gives:

$$\begin{aligned} P(ABC) = \pi_c \cdot P_{cb} \cdot D &= \frac{f_c + S}{(f_a + f_b + 3f_c + 2S)} \cdot \frac{S}{(f_c + S)} \cdot \frac{f_c(f_a + f_b + 3f_c + 2S)}{(f_a + f_c)(f_b + f_c + S)} \\ &= \frac{S}{M} \cdot \frac{f_c \cdot M}{(f_a + f_c)(f_b + f_c + S)} \end{aligned}$$

Thus $D = f_c \cdot M / (f_a + f_c)(f_b + f_c + S)$, but in the case of the single coincidence where $M = N$, we know that the value of D is unity, implying that $(f_a + f_c)(f_b + f_c + S) / f_c = N$. The final expression for event occurrence probability in terms of the measured parameters is then:

$$P(ABC) = S \cdot f_c / (f_a + f_c)(f_b + f_c + S) \quad (2.26)$$

This result is obtained more directly however by noting as with Circuit I, that $P(ABC)$ is the probability of consecutive transitions from A to B to C, and (in the case of Circuit II) a successful exit from C back to B:

$$P(ABC) = P_{ab} \cdot P_{bc} \cdot P_{cb}$$

From Table 2.2 the measured parameter equivalents give:

$$P(ABC) = \frac{f_c}{(f_a + f_c)} \cdot \frac{f_c + S}{(f_b + f_c + S)} \cdot \frac{S}{(f_c + S)} = \frac{S \cdot f_c}{(f_a + f_c)(f_b + f_c + S)}$$

2.4.3 SUMMARY OF SIMPLIFIED MARKOV ANALYSIS FOR RANDOM EVENT DETECTION

It is apparent therefore from the discussion above that a complete formal Markov analysis of a logic circuit is not necessary in order to arrive at the correct formula for event occurrence probability in terms of the measured parameters. The more direct alternative procedure can be summarized as follows:

- * Assign state transition probabilities in terms of the measured parameters according to the method outlined for compiling Tables 2.1 and 2.2.**
- * Determine which set of consecutive transitions constitutes a successful event.**
- * Express the product of these transition probabilities in terms of the theoretical equivalents to verify that the result is in accord with the event specified.**
- * Express them again in terms of the measured parameter equivalents to generate a formula which can be used for the experimental measurements.**

This streamlined procedure is based on a rigorous Markov analysis tailored to provide probability expressions when the event under investigation is specified by a unique set of consecutive transitions.

The key point with this procedure is that implementation does not require the tedious solution of a matrix equation. If the transition rate diagrams for a circuit are generated from first principles, in terms of parameters which can be measured, (such as f_a , f_b , f_c and S), as was done

in Sections 2.3.1 and 2.3.2, then they can be used in turn to compile a table of transition probabilities equivalent to Tables 2.1 and 2.2. Multiplication of the consecutive transition probabilities required to specify the event then produces the formula in terms of the measured parameters necessary to compute the probability and/or occurrence rate of the event from experimental data.

2.5 TRANSITION RATES OBTAINED BY CALCULATION FOR FIXED-NUMBER PULSE STRINGS

It is always desirable to be able to check experimental results against theoretically predicted values. It is shown in this section that the state transition rates, as determined by measurements of f_a , f_b , f_c and S for Circuits I and II, can be derived theoretically for the fixed-number multiple pulse strings. The theory is developed using the single coincidence case once again as an example to illustrate the derivation. It is then extended to cover the whole class of fixed-number pulse strings, i.e. for $n = 3$ and up.

At this point it is convenient to introduce a parameter which is useful both for the development of the method for calculating transition rates, and for checking experimental results against theory. This parameter is the ratio of failures to successes for each state and these ratios are essentially independent of one another. For example a fault in the circuitry pertaining to one state may cause an error down the line in the number of times a subsequent state is entered. However the ratio of failures to successes in meeting the criteria for exit from the later state would not be affected.

In the single coincidence example we see that for both Circuit I and Circuit II the ratios (failure)/(success) in exiting from state A or state C are the same and are given from Tables 2.1 and 2.2 by:

$$\frac{\text{failure}}{\text{success}} = \frac{1 - e^{-\lambda}}{e^{-\lambda}}$$

For a given value of z this ratio gives a number which should be obtainable from an equivalent ratio of the measured parameters. Any discrepancy is a measure of the extent to which the assumed exponential distribution of inter-arrival times is not realized in practice, for whatever reason.

Circuit I Failure/Success Ratios

Reference to Table 1 shows that for Circuit I the ratios (failure)/(success) are given by:

$$\text{State A: } \frac{P_{ca}}{P_{ab}} = \frac{1 - e^{-z}}{e^{-z}} = \frac{f_a}{f_b + f_c + S}$$

$$\text{State B: } \frac{P_{ba}}{P_{bc}} = \frac{e^{-z}}{1 - e^{-z}} = \frac{f_b}{f_c + S}$$

$$\text{State C: } \frac{P_{ca}(F)}{P_{ca}(S)} = \frac{1 - e^{-z}}{e^{-z}} = \frac{f_c}{S}$$

Circuit II Failure/Success Ratios

The equivalent data for Circuit II are:

$$\text{State A: } \frac{P_{aa}}{P_{ab}} = \frac{1 - e^{-z}}{e^{-z}} = \frac{f_a}{f_c}$$

$$\text{State B: } \frac{P_{bb}}{P_{bc}} = \frac{e^{-z}}{1 - e^{-z}} = \frac{f_b}{f_c + S}$$

$$\text{State C: } \frac{P_{ca}}{P_{cb}} = \frac{1 - e^{-z}}{e^{-z}} = \frac{f_c}{S}$$

For the single coincidence we set:

$$\frac{1 - e^{-z}}{e^{-z}} = R$$

and solve for f_a , f_b and f_c in terms of R and S in each case:

Circuit I

$$f_a = S(1 + R)^2; \quad f_b = \frac{S(1 + R)}{R}; \quad f_c = SR \quad (2.27)$$

Circuit II

$$f_a = SR^2; \quad f_b = \frac{S(1 + R)}{R}; \quad f_c = SR \quad (2.28)$$

At this point we can evaluate the ratio R for a given value of z and hence evaluate f_a , f_b and f_c provided that we know S .

For both Circuit I and Circuit II the parameter S is the observed success rate and signifies the detection of a complete single coincidence event. In both cases the apparent probability of the event is obtained by dividing the observed success rate by the total transition rate M . For Circuit II it was shown that the apparent probability $P''(ABC)$ is also the actual probability $P(ABC)$, because in that case M is equal to the input pulse rate N .

$$P''(ABC) = S/M = S/N = P(ABC)$$

Hence for Circuit II:

$$S/N = e^{-2z}(1 - e^{-z})$$

so that

$$S = N \cdot e^{-2z}(1 - e^{-z})$$

We can thus compute the observed success rate S for given values of N and z , which in turn allows f_a , f_b and f_c to be evaluated from the expressions derived above.

Computation of the observed success rate S for Circuit I however is not so simple. The apparent probability $P'(ABC)$ is given by:

$$P'(ABC) = S/M$$

so that

$$S = M \cdot P'(ABC)$$

We do not have a way to connect M with N and z from any previous derivation, however it can be evaluated indirectly from a ratio M/N . Rewriting the above equation to introduce this ratio gives:

$$S = N \cdot (M/N) \cdot P'(ABC)$$

We can evaluate $P'(ABC)$ from equation (2.15) which gives:

$$P'(ABC) = \frac{e^{-2z}(1 - e^{-z})}{1 + 2e^{-z} - e^{-2z}}$$

Hence:

$$S = N \cdot (M/N) \cdot e^{-2z}(1 - e^{-z}) / (1 + 2e^{-z} - e^{-2z}) \quad (2.29)$$

We know from equations (2.5) and (2.6) that:

$$N = (f_a + f_b + 3f_c + 2S) \quad ; \text{ only true for the single coincidence case.}$$

$$M = (f_a + 2f_b + 3f_c + 3S) \quad ; \text{ always true for Circuit I.}$$

The "S" in these two expressions is understood to be the measured (observed) value for Circuit I. As pointed out earlier they are different for Circuit I and Circuit II, however the expression for N will evaluate to the correct result using data from either one. Since we also have f_a , f_b , and f_c in terms of S and R (2.27), we can evaluate the ratio M/N as:

$$M/N = \frac{S \left\{ (1 + R)^2 + (1 + R)/R + 3R + 2 \right\}}{S \left\{ (1 + R)^2 + 2(1 + R)/R + 3R + 3 \right\}}$$

Since S cancels out, this enables the ratio M/N to be evaluated directly, since R is known. This in turn leads to a value for S from equation (2.29) and hence to values for f_a , f_b and f_c for Circuit I from (2.27).

2.5.1 TRANSITION RATE PREDICTIONS FOR STRINGS WITH THREE OR MORE PULSES

The probability for a fixed-number pulse string with n pulses is (from Chapter 1, equation (1.4) given by:

$$P(n\text{-pulses}) = e^{-Z} \cdot (1 - e^{-Z})^{(n-1)} \cdot e^{-Z}$$

This is a triple event which can be measured by Circuits I and II. The only difference from the single coincidence, for which $n = 2$, is that the probabilities for exit from state B are now different.

For Circuit I we now have:

$$\begin{aligned} P_{bc} &= (1 - e^{-Z})^{(n-1)} &= & (f_c + S)/(f_b + f_c + S) \\ P_{ba} &= 1 - (1 - e^{-Z})^{(n-1)} &= & f_b/(f_b + f_c + S) \end{aligned}$$

For Circuit II we now have:

$$\begin{aligned}
 P_{bc} &= (1-e^{-z})^{(n-1)} &= & (f_c + S)/(f_b + f_c + S) \\
 P_{bb} &= 1-(1-e^{-z})^{(n-1)} &= & f_b/(f_b + f_c + S)
 \end{aligned}$$

The (Failure)/(Success) ratios for each state for Circuits I and II now become:

Circuit I (Failure)/(Success)

$$\begin{aligned}
 \text{State A: } P_{aa}/P_{ab} &= (1-e^{-z})/e^{-z} &= & f_a/(f_b + f_c + S) = R_a \\
 \text{State B: } P_{ba}/P_{bc} &= (1-(1-e^{-z})^{(n-1)})/(1-e^{-z})^{(n-1)} &= & f_b/(f_c + S) = R_b \quad (2.30) \\
 \text{State C: } P_{ca}(F)/P_{ca}(S) &= (1-e^{-z})/e^{-z} &= & f_c/S = R_c
 \end{aligned}$$

Circuit II (Failure)/(Success)

$$\begin{aligned}
 \text{State A: } P_{aa}/P_{ab} &= (1-e^{-z})/e^{-z} &= & f_a/f_c = R_a \\
 \text{State B: } P_{bb}/P_{bc} &= (1-(1-e^{-z})^{(n-1)})/(1-e^{-z})^{(n-1)} &= & f_b/(f_c + S) = R_b \quad (2.31) \\
 \text{State C: } P_{ca}/P_{cb} &= (1-e^{-z})/e^{-z} &= & f_c/S = R_c
 \end{aligned}$$

Since these ratios are no longer all the same they are now denoted as R_a , R_b and R_c (even though R_a is still the same as R_c). We proceed as previously to solve for f_a , f_b and f_c in terms of S and R_a , R_b and R_c for each case:

Circuit I

$$f_a = S \cdot R_a(R_c + 1)(R_b + 1); \quad f_b = S \cdot R_b(R_c + 1); \quad f_c = S \cdot R_c$$

Circuit II

$$f_a = S \cdot R_a \cdot R_c; \quad f_b = S \cdot R_b(R_c + 1); \quad f_c = S \cdot R_c$$

Thus far, the extension from the previous argument has been straightforward, but at this point a complication arises, we can no longer write the input pulse rate, N , as:

$$N = (f_a + f_b + 3f_c + 2S)$$

This was derived for the single coincidence where only two pulses were involved and it was quite clear that a failure in state B meant that only one pulse had been recorded. Now we have n pulses for a success in state B, and some number k of pulses where k can be anywhere in the range $1 < k < (n-1)$, for possible failures. A failure in state C is still unambiguous, it means a success in state B, with n pulses, and a single pulse in state C, spoiling a successful complete event and giving a total of $(n + 1)$ pulses. A successful event S means n pulses, while a failure in state A is still one pulse.

We must therefore re-write the relation as:

$$\text{Input pulse rate } N = (f_a + kf_b + (n + 1)f_c + nS)$$

The transition rates, M , for both circuits are still the same in terms of f_a , f_b , f_c and S . These were derived without any consideration of the nature of the process driving the state transitions. It matters not whether the input is a random pulse train or lottery ticket numbers. These rates are given by:

Circuit I

$$\text{Total transition rate } M = (f_a + 2f_b + 3f_c + 3S)$$

Circuit II

$$M = (f_a + f_b + 3f_c + 2S)$$

2.5.2 CALCULATION OF "FAILURE" PULSE RATE IN STATE B

There is a method by which we can find the probable value of k , the number of "failure" pulses associated with f_b . If n pulses are a success, then clearly the failure is either 2, 3, 4 ... $(n-1)$ pulses with consecutive inter-arrival times $< \tau$, or just a single "space" with no pulses after the "marker" pulse which sets the start of the state B interval. In other words events classed as failures are simply lower order fixed-number pulse

strings which were not successful because their number was less than n . We can make a table of the number of pulses in each and the probability of their occurrence as follows, on the premise that each failure consists of a number of consecutive inter-arrival times $< \tau$, terminated by a space $> \tau$:

<u>"Failure" Pulses</u>	<u>Occurrence Probability</u>
1	e^{-z}
2	$(1-e^{-z}) \cdot e^{-z}$
3	$(1-e^{-z})^2 \cdot e^{-z}$
$\vdots$	$\vdots$
$n-1$	$(1-e^{-z})^{(n-2)} \cdot e^{-z}$

From this it is not difficult to see that the probable value for k , the number of "failure" pulses, will be given by a sum of weighted probabilities as:

$$k = \frac{1 \cdot e^{-z} + 2 \cdot (1-e^{-z}) \cdot e^{-z} + 3 \cdot (1-e^{-z})^2 \cdot e^{-z} + \dots + (n-1)(1-e^{-z})^{(n-2)} \cdot e^{-z}}{e^{-z} + (1-e^{-z}) \cdot e^{-z} + (1-e^{-z})^2 \cdot e^{-z} + \dots + (1-e^{-z})^{(n-2)} \cdot e^{-z}}$$

The numerator is the familiar arithmetic-geometric progression as discussed in Chapter 1 in connection with the original derivation of fixed-number pulse string probabilities by *Tenney*⁵⁸. The denominator is the even more familiar simple geometric progression. Accordingly we can write the expression in terms of summations as:

$$k = \frac{\sum_{m=1}^{m=n-1} m(1-e^{-z})^{(m-1)} \cdot e^{-z}}{\sum_{m=1}^{m=n-1} (1-e^{-z})^{(m-1)} \cdot e^{-z}}$$

This is considerably simplified if we let $r = (1-e^{-z})$. Then we have:

$$k = \frac{\sum_{m=1}^{m=n-1} mr^{(m-1)}}{\sum_{m=1}^{m=n-1} r^{(m-1)}}$$

For the numerator the sum to m terms is given by:

$$\sum_1^m mr^{(m-1)} = \frac{1-r^m}{(1-r)^2} - \frac{mr^m}{(1-r)}$$

Hence when $m = (n-1)$:

$$\sum_{m=1}^{m=n-1} mr^{(m-1)} = \frac{1-r^{(n-1)}}{(1-r)^2} - \frac{(n-1)r^{(n-1)}}{(1-r)}$$

Similarly for the denominator:

$$\sum_{m=1}^{m=n-1} r^{(m-1)} = \frac{1-r^{(n-1)}}{1-r}$$

thus:

$$k = \frac{\left(\frac{1-r^{(n-1)}}{(1-r)^2} - \frac{(n-1)r^{(n-1)}}{(1-r)} \right)}{\left(\frac{1-r^{(n-1)}}{1-r} \right)} = \frac{1}{1-r} - \frac{(n-1)r^{(n-1)}}{1-r^{(n-1)}} \quad (2.33)$$

This is an expression which gives hard numbers for k when the appropriate values of z and n are inserted.

2.5.3 DERIVATION OF FINAL EQUATIONS FOR TRANSITION RATES

To summarize the progress thus far; we are now in a position to calculate f_a , f_b and f_c provided we can find S . This can be done as previously by taking the ratio M/N where M and N are expressed in terms of the known

ratios R_a , R_b and R_c . Until now, evaluation of N was not possible because of the state B "failure" pulses uncertainty which we wrote in as " k ". Now however we can put a firm value on k and obtain the ratio M/N for both circuits.

2.5.3.1 CIRCUIT I: TRANSITION RATES

For Circuit I the ratio M/N is given by:

$$M/N = (f_a + 2f_b + 3f_c + 3S)/(f_a + kf_b + (n+1)f_c + nS)$$

Substituting for f_a , f_b and f_c from equation (51) the ratio becomes:

$$M/N = \frac{S \left\{ R_a (R_c + 1)(R_b + 1) + 2R_b (R_c + 1) + 3R_c + 3 \right\}}{S \left\{ R_a (R_c + 1)(R_b + 1) + kR_b (R_c + 1) + (n+1)R_c + n \right\}}$$

Since S cancels out we can evaluate M/N directly for given values of n and z , since we can now evaluate k from equation (2.33).

A value for the observed success rate, S , can now be found for Circuit I as previously, starting from the relation:

$$P'(ABC) = \pi_c \cdot e^{-z} = S/M$$

However π_c must now be re-evaluated from equation (2.21), since P_{bc} has changed from $(1-e^{-z})$ to $(1-e^{-z})^{(n-1)}$. When this is done we have:

$$P'(ABC) = e^{-2z}(1-e^{-z})^{(n-1)} \left(1 + e^{-z}(1 + (1-e^{-z})^{(n-1)}) \right) = S/M$$

However:

$$S = M \cdot P'(ABC) = N \cdot (M/N) \cdot P'(ABC)$$

and since (M/N) and $P'(ABC)$ are now known in terms of n , N and z , S can be evaluated. This in turn allows f_a , f_b and f_c to be evaluated from (2.31), also in terms of n , N and z . Thus all the transition rates can be predicted for given values of n , N and z .

2.5.3.2 CIRCUIT II: TRANSITION RATES

We now follow an identical procedure in order to calculate the transition rates f_a , f_b , f_c and S for Circuit II. The probable number of pulses k involved in a failure at state B is exactly the same as for Circuit I, because the criteria are identical. The only difference between the two circuits is the destination state in the event of a failure. The expression for N is the same and that for M is as originally derived in equation (2.11). Hence we have:

$$M/N = (f_a + f_b + 3f_c + 2S)/(f_a + kf_b + (n+1)f_c + nS)$$

Substituting for f_a , f_b and f_c from (2.32) gives:

$$M/N = \frac{S \left\{ R_a \cdot R_c + R_b(R_c + 1) + 3R_c + 2 \right\}}{S \left\{ R_a \cdot R_c + kR_b(R_c + 1) + (n+1)R_c + n \right\}}$$

again, since S cancels out, M/N can be evaluated directly for given values N and z . The observed success rate is then given by:

$$P''(ABC) = \pi_c \cdot e^{-z} = S/M$$

π_c , when evaluated using equation (2.25), and noting as before that P_{bc} is now $(1-e^{-z})^{(n-1)}$ for n pulses in a string, gives:

$$e^{-z} \cdot \pi_c = e^{-2z} (1-e^{-z})^{n-1} / ((1-e^{-z})^{n-1} + e^{-z}) = P'(ABC)$$

$$S = M \cdot P''(ABC) = N \cdot (M/N) \cdot P''(ABC)$$

We thus have an expression for S which can be directly evaluated, leading to evaluation of f_a , f_b and f_c from equation (2.32). All the transition rates for Circuit II can therefore be predicted for given values of n , N and z .

2.5.4 NUMERICAL EVALUATION OF TRANSITION RATES

A FORTRAN computer program was written to evaluate the expressions for R_a , R_b , R_c , M/N and hence f_a , f_b , f_c and S . They were computed for the values of z , n and the input pulse rate N used in the experimental measurements. The predicted data so obtained are included in Appendix 1.B and 2.B.

CHAPTER 3 MEASUREMENT OF MULTIPLE PULSE STRING OCCURRENCE RATES

3.0 REVIEW OF SUB-EVENT PROBABILITIES TO BE MEASURED

Two classes of multiple pulse strings were identified in Section 1.1. One has fixed-numbers of pulses with consecutive inter-arrival times $< \tau$ preceded and followed by spaces $> \tau$. The other is based on fixed time intervals which are integer multiples of τ and which contain pulses having inter-arrival times each $< \tau$, again preceded and followed by spaces $> \tau$.

The occurrence probabilities of these strings were derived previously and are given again here for reference. For a string with a fixed-number n of pulses the occurrence probability is given by (Tenney⁵⁸):

$$P(n \text{ pulse string}) = e^{-Z} \cdot (1 - e^{-Z})^{(n-1)} \cdot e^{-Z}$$

Both types of string involve a triple compound event (ABC) and as such are amenable to detection by Circuits I and II. From the discussion in the previous section we know that it is not necessary to evaluate P_{aa} , P_{ab} etc., once the sub-events A, B and C have been defined in terms of their probabilities $P(A)$, $P(B)$ and $P(C)$, then:

$$P(A) = P_{ab}$$

$$P(B) = P_{bc}$$

$$P(C) = P_{cb} \text{ (or } P_{ca}(S) \text{ for Circuit I)}$$

For a fixed-number pulse string with n pulses:

$$P(A) = e^{-Z}; P(B) = (1 - e^{-Z})^{n-1}; P(C) = e^{-Z}$$

Provided that the logic circuits are designed so that they conform either to the Circuit I model of Figure 2.1(a), or the Circuit II model of Figure 2.1(b), then the actual occurrence rates of these strings can be measured and computed from equations (2.24) or (2.26) respectively.

3.1 LOGIC CIRCUIT IMPLEMENTATION

The common element for the logic circuits which were devised to measure single coincidences and the two classes of multiple pulse strings is the state sequence controller. This unit has four states A, I, B and C. States A, B and C correspond directly with sub-events A, B and C, while state I is an idle state into which the circuit moves following a successful transition from state A (i.e., detection of the first space $> \tau$). Arrival of the next pulse (the "marker" pulse), advances the sequence controller from I to B, but the next transition depends on the criteria to be met in each case.

Reference to Chapter 2 Figure 2.1 shows that for Circuit I the state sequence controller can be a simple counter with four decoded outputs, since with this model either there is an advance to the next state, or a reset to the initial state. The Circuit II model however requires that transitions be made to the idle state for new attempts from both state B and state C. If a simple counter is used, then these transitions have to be accomplished by some sort of sleight-of-hand; e.g. a reset to state A followed by an additional advance pulse to clock the counter to the idle state. Another alternative is to use a shift register with parallel inputs, thereby allowing a direct transition from one state to any other state by changing the inputs. Both of these techniques involve concealed additional logic transitions which must take place "on the fly" whenever one of these non-sequential state transitions takes place. This does not present a difficulty when the input to the circuit is a periodic pulse train, but a random pulse train can cause logical "collisions" which makes these techniques unreliable in practice.

3.1.2 STATE SEQUENCE CONTROL LOGIC

The scheme finally adopted for the state sequence controller is shown in Figure 3.1.

Figure 3.1: State sequence control logic used in Circuits I and II.

Figure 3.2: Circuit I, including the state sequence control logic of Figure 3.1, as implemented in a single 24 pin EPLD (Erasable-Programmable-Logic-Device) integrated circuit.

Figure 3.3: Circuit II, including the state sequence control logic of Figure 6, as implemented in a single 24 pin EPLD (Erasable-Programmable-Logic-Device) integrated circuit.

Figure 3.4: Block diagram of entire circuitry required for measurements of single coincidences and both types of multiple pulse strings.

It consists essentially of four R-S flip-flops which have true outputs corresponding to the four states A, I, B and C. Their RESET inputs are each connected to 3-input NOR gates so that applying a set pulse to any one of the flip-flops causes the other three to be reset. This scheme allows a transition from any state to any other state directly and without any concealed additional steps being required. An additional 4-input OR gate was incorporated which sets state A if at any time all four states are in a reset condition, a circumstance which might arise for example on power-up. This circuit was used as the state sequence controller for both Circuit I and Circuit II as a matter of convenience, even though a simple counter would have sufficed for Circuit I.

All three types of measurements required a time interval generator which generates a suitable logic pulse after a preset time τ and is reset by each incoming pulse. The value of τ was set at 100 μ s to ensure that the incoming pulse widths (approximately 1 μ s) were negligible by comparison. The interval generator was implemented using a 2 MHz clock and frequency divider with a monostable circuit generating the pulse marking an elapsed time of τ .

3.1.3 PARAMETER INPUTS AND LOGIC ALGORITHMS FOR CIRCUITS I AND II

The algorithm for driving the sequence controller consists of a series of AND conditions, implemented with suitable logic gates. The parameters involved are as follows:

τ	pulse generated after an elapsed time τ
A, I, B and C	logic levels generated by the sequence controller
P	incoming pulse from the random source
$n\tau$	pulse generated after a time $n\tau$
nP	logic level from pulse counter n^{th} output.

Tables 3.1 and 3.2 show the AND conditions as " $X \bullet Y$ " for success and failure in meeting individual state criteria and the destination state in each case. Table 3.1 entries are for Circuit I and Table 3.2 entries are for Circuit II.

Consider the entries in Table 3.1 for the single coincidence measurement. If the circuit is in State A, and a pulse arrives before the time interval end pulse τ , then the necessary initial space $> \tau$ was not registered and we have a failure, the condition for which is:

"State A logic level true-AND-an input pulse".

The resulting output from the appropriate AND gate will therefore reset the time interval generator and a new attempt will be made.

A success means the τ pulse arrives before another input pulse, the condition for which is:

"State A logic level true-AND-the τ pulse".

The output from this AND gate advances the circuit to State I (the "idle" state) where it remains inactive until the next pulse, (the "marker" pulse) arrives. The circuit is then advanced to State B and the τ interval generator is reset, the condition being:

"State I logic level true-AND-an input pulse".

A success in State B is the arrival of a second pulse before the τ pulse:

"State B logic level true-AND-an input pulse"

causing an advance to State C. No second pulse in the elapsed time τ is a failure, (ie no coincidence), the condition for which is:

"State B logic level true-AND-a τ pulse"

which causes a reset to State A.

Reference to Figure 2.1 will clarify the various AND conditions listed in Tables 3.1 and 3.2. The actual logic schematics used to implement the algorithms are shown in Figures 3.2 and 3.3.

Table 3.1: Circuit I Sate Transition Criteria

EVENT	INITIAL STATE	SUCCESS	STATE	FAILURE	STATE
Single Coincidence	A	A $\bullet$ τ	I	A $\bullet$ P	A
	I	I $\bullet$ P	B	N/A	N/A
	B	B $\bullet$ P	C	B $\bullet$ τ	A
	C	C τ	A	C $\bullet$ P	A
Fixed Interval String	A	A $\bullet$ τ	I	A $\bullet$ P	A
	I	I $\bullet$ P	B	N/A	N/A
	B	B $\bullet$ n τ	C	B $\bullet$ τ	A
	C	C $\bullet$ τ	A	C $\bullet$ P	A
Fixed Number Pulse String	A	A $\bullet$ τ	I	A $\bullet$ P	A
	I	I $\bullet$ P	B	N/A	N/A
	B	B $\bullet$ nP	C	B $\bullet$ τ	A
	C	C $\bullet$ τ	A	C $\bullet$ P	A

Table 3.2: Circuit II State Transition Criteria

EVENT	INITIAL STATE	SUCCESS	STATE	FAILURE	STATE
Single Coincidence	A	A $\odot$ τ	I	AP	A
	I	I $\odot$ P	B	N/A	N/A
	B	B $\odot$ P	C	B $\odot$ τ	I
	C	C $\odot$ τ	I	C $\odot$ P	A
Fixed Interval String	A	A $\odot$ τ	I	A $\odot$ P	A
	I	I $\odot$ P	B	N/A	N/A
	B	B $\odot$ n τ	C	B $\odot$ τ	I
	C	C $\odot$ τ	I	C $\odot$ P	A
Fixed Number Pulse String	A	A $\odot$ τ	I	A $\odot$ P	A
	I	I $\odot$ P	B	N/A	N/A
	B	B $\odot$ nP	C	B $\odot$ τ	I
	C	C $\odot$ τ	I	C $\odot$ P	A

3.1.4 CIRCUITS I AND II IN INTEGRATED CIRCUIT FORM

These tables show that the various AND conditions for the three measurements are identical with one exception; the condition for a successful transition from state B to C. All the conditions involve simple 2-input logic gates and only one of the inputs for the state B exit condition is different for each measurement. The significance of this is that the logic elements of Circuits I and II can be hardwired with only one connection being changed for the three different measurements. This is a situation which is ideal for the use of programmable-array-logic (PAL). Figures 3.2 and 3.3 show the portions of Circuits I and II which were implemented in this way, each one fitting into a single 24 pin DIP integrated circuit.

The PAL system used for these circuits is produced by the ALTERA CORP. of Santa Clara, California. It is based on an IBM-PC and includes a PROM burner, blank ICs and software containing a library of popular TTL functions which can be implemented in whole or in part in a blank chip. Two particularly appealing features of this system are (a) the use of high-speed complementary-metal-oxide-semiconductor (HCMOS) technology, giving both high speed and low power consumption, and (b) the capability for erasing a circuit from a chip using ultra-violet light, allowing it to be re-used.

Figure 3.4 is a block diagram of the entire circuitry required for all three measurements. The τ -interval generator is reset on each pulse arrival. When the fixed-interval multiple pulse string measurements are being made the $n\tau$ pulse signal is applied to the B-EXIT terminal. The $n\tau$ interval generator is held reset in the idle state and starts when the circuit enters state B, on arrival of the first pulse ("marker pulse") of the string. When fixed-number pulse strings are being measured, the nP signal is applied to the B-EXIT terminal. The pulse counter is held reset while the circuit is in state A. Single coincidence measurements are made with the

input pulse train P being applied to the B-EXIT terminal. In all cases the clock is inhibited while the circuit is in the idle state.

3.2 EXPERIMENTAL DETAILS

The Poisson random pulse train was generated by a scintillation detector receiving gamma radiation from a low level radioisotope source. The radioisotope used was ^{137}Cs which has a single energy level of 661 keV. A pulse height threshold level comparator was set to exclude counts due to radiation energies less than 600 keV. The count rate was adjusted from 3,000 c/s to 20,000 c/s by changing the source-detector distance.

A Data General Corp. NOVA-4C minicomputer was available for data collection and it was used to acquire the f_a , f_b , f_c and S readings under software control. The duration of a measurement was set by a keyboard entry specifying the number of successfully completed events (ABC). This ensured a predetermined statistical precision in the measurements. The software was written in assembly language for maximum speed of response to hardware interrupts. It provided printouts of key parameters in each case.

The photographs of Figures 3.5 and 3.6 show the laboratory set-up for making the measurements, and the prototype "scrabble-board" on which the circuitry of Figure 3.4 was arranged.

3.2.1 PULSE WIDTHS

All of the measurements were made with τ set at 100 μs . The scintillation detector pulses were passed directly into the base of a transistor from the photomultiplier tube anode. The high voltage was set at a level such that pulse amplitudes from the ^{137}Cs source were sufficient to turn the transistor on to saturation. The output from this transistor was then applied to a CMOS Schmitt trigger logic gate which ensured well-defined pulses for counting. The transistor bias and high voltage supply were

adjusted to produce final logic pulses $1\text{ }\mu\text{s}$ wide. Ideally these pulses would be infinitely narrow in order that their distribution be unaffected. In practice the decay time of the sodium iodide scintillator is the limiting factor, so that a pulse width of $1\text{ }\mu\text{s}$ is a realistic lower limit.

Figure 3.5: Experimental Set-up: The sodium Iodide Scintillation Detector and ^{137}Cs radioactive source are seen at left. The prototype board containing the experimental circuitry is interfaced to the NOVA-4C minicomputer which acquires the data. A display presents both graphic and alphanumeric data.

Figure 3.6: The custom EPLD chips containing "Circuit I" and "Circuit II" are shown on the prototype board together with the other necessary peripheral logic.

The effect of a finite pulse width is to add "dead-time" to the statistical process, which depresses the measured count rate below what it would otherwise be. As the count rate increases this effect becomes more pronounced. The nominal values of count rate used to calculate predicted data for comparison with measured data, are all adjusted to include this effect.

3.2.2 DEAD-TIME MEASUREMENTS

In order to maintain an absolute check on the various time dependent experimental parameters, an auxiliary circuit was used to generate and monitor a dead-time function.

The τ interval generator was used to generate a second 100 μ s time interval beginning on each random pulse arrival. This interval was not retriggered or extended by more pulse arrivals and could only be reactivated after timing out. It thus simulated a real-world ADC having a fixed dead-time of 100 μ s. The count rate for these 100 μ s wide pulses was recorded by the data acquisition system along with the transition rate data for Circuit I only.

The well established formula (amply documented in the literature, see e.g. *Müller*²⁶) for calculation of a non-extended dead-time is given by:

$$N_r = N_t / (1 + N_t \cdot T_d) \quad (3.1)$$

where:

N_r = recorded count rate; c/s

N_t = true count rate; c/s

T_d = dead-time in seconds (100 μ s in this case)

In this instance N_r was the count recorded for the 100 μ s wide pulses generated as described above, while N_t was the input count rate from the scintillation detector.

The N_t , N_r and T_d values obtained experimentally would only conform to equation 3.1 if the following criteria were met:

- * N_r and N_t accurately measured
- * Time interval generator accurately set

- * Distribution of inter-arrival times closely follows an exponential law.

Agreement would therefore provide additional independent evidence that these parameters were accurately set during the pulse-string measurements.

3.2.3 SIMULTANEOUS MEASUREMENT OF FIXED-NUMBER AND FIXED-INTERVAL STRINGS

A relatively minor modification to the Circuit II input connections, with corresponding data acquisition software changes, allowed on-line sorting of multiple pulse strings into both the fixed-number and fixed-interval categories.

The principle was to set up Circuit II to measure fixed-interval strings as described above, but with the $N\tau$ pulse generator disconnected from the "B-EXIT" terminal, (see Figure 3.4). This prevents the circuit from ever reaching state C, and ensures that it remains in State B until an inter-arrival time $>\tau$ occurs, marking the end of the multiple pulse string and inducing a "failure" in state B. The number of fixed- τ intervals and the number of pulses which elapse until this point are recorded by the acquisition system.

The software keeps a tally of the numbers of up to 50 orders of each type of string in a 50 x 50 memory array, incrementing the contents of the appropriate location according to the type of multiple pulse strings acquired. For example if the τ -interval counter has recorded 3, and the pulse counter has recorded 15, then the content of the (3, 15) location, keeping score of 3τ fixed-interval strings and 15-pulse fixed-number strings, is incremented.

At the end of the experiment the software uses the contents of the array to compute for each fixed-interval string order, the percentage contribution of the fixed-number strings involved.

This method of measurement provided a direct printout of the relative probabilities of n -pulse fixed-number strings qualifying as $m\tau$ fixed-interval strings, as well as occurrence rates of all orders of each kind.

CHAPTER 4 DATA ANALYSIS AND INTERPRETATION

4.0 OVERVIEW

Both Circuit I and Circuit II were used to make measurements of the transition rates and occurrence probabilities of fixed-number pulse strings from $n = 2$ to $n = 9$, where n is the number of pulses per string, or "string number". The single coincidence, used as an example in the derivation of much of the theory in previous chapters, corresponds to a fixed-number pulse string with $n = 2$. The data for it were acquired along with the data for the other n values. No distinction is made hereafter between it and the higher order string numbers.

The fixed-interval string data were acquired only with Circuit II, since the fixed-number pulse string data adequately demonstrated the equivalence of the two circuits and the experimental verification of the theoretical treatment of the two models.

In the case of the fixed-number string measurements using Circuit I, and the fixed-interval string and simultaneous measurements using Circuit II, eight different input count rates were used. All data were measured with the τ interval generator set at $100 \mu\text{s}$. The count rates and the corresponding values, (based on the relation $Z = N\tau$), were as follows:

<u>Input Count Rate c/s</u>	<u>Z</u>	($\tau = 100 \mu\text{s}$)
3,000	0.30	
5,000	0.50	
7,000	0.70	
10,000	1.00	
12,500	1.25	
15,000	1.50	
17,500	1.75	
20,000	2.00	

The measurements on fixed-number strings with Circuit II omitted the 5,000, 12,500 and 17,500 input count rates.

4.1 CORRECTION FOR FINITE PULSE WIDTHS

The input count rates for all the measurements were adjusted, using a frequency counter, to the values chosen for the experiment as given above. However these recorded values were depressed due to the effect of the finite pulse width as discussed earlier. Equation (3.1) gives the relation between true and recorded count rates, and when rearranged gives the true count rate in terms of the recorded one as:

$$N_t = \frac{N_r}{1 - N_r \cdot T_d} \quad (4.1)$$

The substitution $Z = N\tau$ has been used in all of the many derivations for the sake of convenience. The value of τ is fixed at 100 μ s so that a value for Z of 0.3 means a measured input count rate of 3000 c/s. In fact the actual count rate is slightly higher after the correction is made, which is equivalent to making the same correction to Z . Substituting Z_t/τ for N_t , and $Z_r/\tau = N_r$, in equation (3.2) gives

$$Z_t/\tau = \frac{Z_r/\tau}{1 - Z_r \cdot (T_d/\tau)}$$

or

$$Z_t = \frac{Z_r}{1 - Z_r(T_d/\tau)}$$

but T_d in this case represents the 1 μ s pulse width while $\tau = 100 \mu$ s, hence

$$Z_t = \frac{Z_r}{1 - Z_r/100} \quad (4.2)$$

2

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1010a
(ANSI and ISO TEST CHART No. 2)

In subsequent sections of this chapter comparisons are presented between experimental and predicted data for various values of Z . In all cases the equations used in generating the predicted data have been evaluated with the nominal Z values corrected according to equation (3.3).

4.2 FIXED-NUMBER PULSE STRING DATA FROM CIRCUIT I MEASUREMENTS

The parameters measured were as follows:

- * Dead-time from the simulated ADC circuit as percentage of the counting time.
- * The four transition rates:
 f_a
 f_b
 f_c
 S
as defined in the previous discussions.
- * The failure/success ratios
 R_a
 R_b
 R_c
as defined in Chapter 2.6.

The computer interface was equipped with four 16-bit counters and two interrupts. The simulated ADC counts were connected to one interrupt and the "S" (successful event) transition rate logic output was connected to the other one. The contents of the f_a , f_b and f_c counters were acquired at each "S" interrupt. The total transition rate "M" was computed on-line according to equation (2.6): $M = f_a + 2f_b + 3f_c + 3S$. The failure/success ratios were also computed on-line from the relations derived in Section 2.6.

Appendix 1.A contains samples of the data for the Circuit I measurements as generated by the computer making the measurements. Each measurement was made three times. Table 4.1 shows the number of events entered for each measurement which sets the statistical precision of the acquired data. The transition rates printed out are all normalized to counts/sec.

Table 4.1: Number of Events per Measurement
String Number

<u>Z</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>
0.3	5000	5000	1500	500	200			
0.5	5000	5000	5000	3000	1000	500		
0.7	5000	5000	5000	3000	2500	1500	1000	500
1.0	5000	5000	5000	5000	5000	5000	3000	2000
1.25					5000			
1.50					5000			
1.75					5000			
2.00					5000			

4.2.2 PREDICTED TRANSITION RATE DATA

Appendix 1.B contains a listing of the transition rates and failure/success ratios calculated using the equations derived in Chapter 2.6. A correction was made to the value of Z used in the various exponential functions involved in these equations to allow for the effect of the one microsecond dead-time discussed previously.

4.2.3 COMPARISON OF EXPERIMENTAL AND PREDICTED DATA

Figures 4.1 to 4.5 show plots of the five individual transition rates, f_a , f_b , f_c , S and M, against string number. Each figure shows eight curves, these are the transition rate in question vs string number at each of the eight Z

Figure 4.1: Transition rate f_a plotted against string number for the eight input count rates (eight Z values). Experimental data are from the Circuit I measurements listed in Appendix 1.A with data points denoted by crosses. Predicted data points from Appendix 1.B are denoted by circled dots.

Figure 4.2: Transition rate f_b plotted against string number for the eight input count rates (eight Z values). Experimental data are from the Circuit I measurements listed in Appendix 1.A with data points denoted by crosses. Predicted data points from Appendix 1.B are denoted by circled dots.

Figure 4.3: Transition rate f_c plotted against string number for the eight input count rates (eight Z values). Experimental data are from the Circuit I measurements listed in Appendix 1.A with data points denoted by crosses. Predicted data points from Appendix 1.B are denoted by circled dots.

Figure 4.4: Transition rate S plotted against string number for the eight input count rates (eight Z values). Experimental data are from the Circuit I measurements listed in Appendix 1.A with data points denoted by crosses. Predicted data points from Appendix 1.B are denoted by circled dots.

Figure 4.5: Transition rate M plotted against string number for the eight input count rates (eight Z values). Experimental data are from the Circuit I measurements listed in Appendix 1.A with data points denoted by crosses. Predicted data points from Appendix 1.B are denoted by circled dots.

values listed earlier. The experimental data points, (taken from the numbers in Appendix 1.A; the mean of the three measurements being shown in each case) are denoted by crosses and are connected by straight lines. The calculated values are taken from the listing in Appendix 1.B and are denoted by circles on the plots, (not connected by lines).

The plots for transition rates f_b and S in Figures 4.2 and 4.9 respectively, show curves which cross other curves. For example the curves for z values of 0.3 and 0.5 in the f_b rate data in Figure 4.2 do not follow the nested pattern of the rest of the family. The reason is related to the fact that each string rate reaches a maximum at a different z value. In the case where the string consists of only two pulses ($n = 2$), the transition rate f_b is the rate for single pulses, i.e. the situation where a pulse arrives preceded by a space $> \tau$, but a second one does not arrive within a time τ following the first one. The occurrence rate for single pulses is $Ne^{-2N\tau}$. This rate has a maximum (found by equating the first derivative to zero) at $N\tau = z = 0.5$. Reference to Figure 4.2 shows that for the string with two pulses, (the data set with points on the vertical axis), the transition rates for $z = 0.3$ and $z = 0.7$ are both lower than that for $z = 0.5$. The f_b transition rate calculated for the string with six pulses reaches its maximum almost exactly at $z = 0.7$, (from data in a more detailed listing than the one included in Appendix 1.B), so that the rates for $z = 0.3$ and $z = 0.5$ are both lower and lead to the cross-over points evident in the plots. A parallel argument accounts for the crossed curves in the S transition rate plots of Figure 4.4.

It is clear from these plots that the agreement between measured and predicted transition rates is quite remarkable. This close agreement leads to the following conclusions:

- * The theoretical derivations giving the occurrence rates for the fixed-number pulse strings are soundly based.
- * The model theory developed for the analysis of Circuit I is correct.

- * If the distribution of inter-arrival times departs from an exponential function, the deviation is not measureable by these experiments.
- * The transition rates f_c and S for the $n = 9$ fixed-number string involve exponentials raised to the 8th power. Hence any departure from an exponential distribution would be considerably amplified at these high-order string numbers. In fact the plots of these transition rates (Figs. 4.3 and 4.4) show no measureable discrepancy between the experimental and the predicted data.
- * Experimental parameters, e.g. input pulse rate, time interval generators etc., were accurately determined. Errors in these would have caused discrepancies between experimental and predicted data.

There has been debate in the literature concerning the real statistical distribution of radioactive decay. *Frigerio*⁸² conducted experiments which led him to reject a Poisson distribution in favour of one proposed by *Brockwell and Moyal*⁸³. Later this was discounted by *Jordan and McBeth*⁸⁴; their experiments showed that when count rate data were properly corrected for instrumental dead-time effects, they conformed with Poisson statistics.

The results of the multiple pulse string measurements documented in this study lend further incidental support for a Poisson distribution governing the radioactive decay process.

4.3 SIMULATED ADC DEAD-TIME MEASUREMENTS

These measurements were described in Chapter 3.2.2, where the formula was given which relates the recorded count rate N_r , to the true count rate N_t and dead-time T_d :

$$N_r = \frac{N_t}{1 + N_t \cdot T_d}$$

In this instance the T_d was chosen to be equal to τ , the 100 microsec time interval used as the "space" in defining multiple pulse strings. Hence in this case:

$$N_r = \frac{N_t}{1 + N_t \tau} = \frac{N_t}{1 + Z}$$

The dead-time per second is the number of simulated ADC counts/second N_r , multiplied by the time for each one which is τ . Hence:

$$\text{Dead time/second} = N_r \cdot \tau = \frac{N_t \tau}{1 + Z} = \frac{Z}{1 + Z}$$

Dead-time is normally expressed as a percentage, so that:

$$\% \text{ Dead-time} = 100 N_r \tau = \frac{100Z}{1 + Z}$$

The quantity $100 N_r$ was computed on-line during data acquisition by the minicomputer and produced the data in Appendix 1.A under the column "% DT". Table 4.2 shows the mean of these data for each input count rate against predicted values computed from $100Z/(1 + Z)$ with the nominal Z values corrected for the one microsec dead-time due to the detector as described earlier.

Table 4.2: Measured VS Predicted Dead-Time

<u>Z</u>	% Dead-Time	
	<u>Measured</u>	<u>Predicted</u>
0.3	23.2	23.3
0.5	33.5	33.6
0.7	41.5	41.4
1.0	50.3	50.2
1.25	56.0	55.8
1.50	60.6	60.2
1.75	64.4	63.9
2.00	67.4	66.9

The agreement between measured and predicted data is entirely satisfactory and provides additional evidence in support of the conclusions drawn on the basis of the fixed-number pulse measurements.

4.4 FIXED-NUMBER PULSE STRING DATA FROM CIRCUIT II MEASUREMENTS

Samples of the data generated by the data acquisition system for these measurements are presented in Appendix 2.A. The predicted values, computed using the equations developed in Chapter 2.5.3.2, are in Appendix 2.B.

Again the close agreement between the measured and predicted data leads to the same conclusions as for the Circuit I data. Plots corresponding to those in Figures 4.1 to 4.5 are shown for the Circuit II data in Figures 4.6 to 4.10. Only five Z values were used for the Circuit II measurements as was explained earlier.

Figure 4.6: Transition rate f_a plotted against string number for the five input count rates (five Z values). Experimental data are from the Circuit II measurements listed in Appendix 2.A with data points denoted by crosses. Predicted data points from Appendix 2.B are denoted by circled dots.

Figure 4.7: Transition rate f_b plotted against string number for the five input count rates (five Z values). Experimental data are from the Circuit II measurements listed in Appendix 2.A with data points denoted by crosses. Predicted data points from Appendix 2.B are denoted by circled dots.

Figure 4.8: Transition rate f_c plotted against string number for the five input count rates (five Z values). Experimental data are from the Circuit II measurements listed in Appendix 2.A with data points denoted by crosses. Predicted data points from Appendix 2.B are denoted by circled dots.

Figure 4.9: Transition rate S plotted against string number for the five input count rates (five Z values). Experimental data are from the Circuit II measurements listed in Appendix 2.A with data points denoted by crosses. Predicted data points from Appendix 2.B are denoted by circled dots.

Figure 4.10: Transition rate M plotted against string number for the five input count rates (five Z values). Experimental data are from the Circuit II measurements listed in Appendix 2.A with data points denoted by crosses. Predicted data points from Appendix 2.B are denoted by circled dots.

The new data values marked "DENOM" in Appendix 2-A are the values computed for the factor:

$$(f_a + f_c) \cdot (f_b + f_c + S) / f_c$$

the reciprocal of which appears in equation (2.26). the probability of the compound event, $P(ABC)$, is given by that equation as:

$$P(ABC) = S \cdot f_c / (f_a + f_c) \cdot (f_b + f_c + S)$$

where S is the observed success rate. Since $P(ABC) = \text{Success/Attempts}$, then the (reciprocal) factor $(f_a + f_c) \cdot (f_b + f_c + S) / f_c$ must represent "Attempts". The measured values listed for this factor come out to be effectively the input pulse rate N in all cases, confirming that for Circuit II $P(ABC) = S/N$.

4.5 FIXED-INTERVAL STRING DATA FROM CIRCUIT II MEASUREMENTS

Samples of the data generated by the data acquisition system are presented in Appendix 3. Calculation of the transition rates for the fixed-number pulse strings was relatively straightforward because the "Failure/Success" ratios, R_a , R_b and R_c , were easily derived as the analytical expression given by equations (2.30) and (2.31) in terms of n the number of pulses in the string. The value of k involved in the calculation of the transition rate f_b was also given by a relatively simple expression (equation 2.33).

Any attempt to calculate the transition rates for the fixed-interval strings would involve multiple numerical convolution operations of the type described in Chapter 1.3.6 for each one of the transition probabilities from state B, and for computation of the value of k .

There seemed little point in pursuing this sledge-hammer approach since (a) the result would undoubtedly be subject to considerable cumulative error, and (b) the validity of the transition rate measurement theory had already been verified by the fixed-number pulse string experimental data.

However one cross check does exist and this concerns the data for the f_b transition rates for the lowest order string $TAU = 1$. A failure in state B involves the same criteria for this case as for the fixed-number pulse string $n = 2$. In both cases it is that no pulse is received within a single τ interval following the marker pulse. This implies that the marker pulse has a space $> \tau$ on either side, which makes it a "single" pulse, the probability for which is e^{-2z} . The corresponding single pulse rate is therefore $N \cdot e^{-2z}$. Table 4.3 is a comparison of the predicted single pulse rates (from f_b at $n = 2$, Appendix 2B) for the eight Z values, with the measured f_b values for fixed single-interval strings and fixed-number strings, (from the Circuit II data in both cases). The agreement between predicted and measured data is seen to be close.

Table 4.3 Single Pulse Count Rates for f_b Transition Rates, (Circuit II Data)

Z	Predicted, f_b @ $n = 2$	Experimental	
		f_b Fixed-Number	f_b Fixed-Interval
0.30	1643	1660	1656
0.50	1830	----	1830
0.70	1709	1721	1733
1.00	1326	1350	1331
1.25	994	----	1015
1.50	714	727	730
1.75	497	----	513
2.00	338	335	341

Tables 4.4 and 4.5 show predicted and measured data for the fixed-interval string occurrence rates, and the number of pulses per string for the eight input count rates. The predicted data were obtained from the computer program referred to in Chapter 1.3.6, which performed repeated numerical convolution of the truncated exponential probability density function to produce the required probability values. The numerical computation involved here is very much less than would have been required for calculation of the transition rates.

TABLE 4.4

Predicted Fixed Interval String Occurrence Rates (per Sec)
from Computer Convolution of Truncated Probability Density Function

Z	1-Tau	2_Tau	3_Tau	4_Tau	5_Tau	6_Tau	7_Tau	8_Tau	9_Tau
0.30	494.41	72.65	7.92	.76	.07	.01	.00	.00	.00
0.50	919.11	220.80	41.38	7.17	1.22	.21	.04	.01	.00
0.70	1203.42	394.18	104.67	26.37	6.59	1.65	.41	.10	.03
1.00	1336.33	594.86	224.10	81.88	29.82	10.86	3.95	1.44	.52
1.25	1253.42	664.37	306.23	138.31	62.36	28.12	12.68	5.71	2.56
1.50	1080.33	651.54	349.25	184.64	97.53	51.51	27.21	14.35	7.52
1.75	877.55	583.47	351.05	209.21	124.62	74.23	44.21	26.30	15.50
2.00	682.02	488.53	321.45	210.11	137.30	89.72	58.62	38.27	24.83

Corresponding Experimental Measurements

(Circuit II Data)

Z	1-Tau	2_Tau	3_Tau	4_Tau	5_Tau	6_Tau	7_Tau	8_Tau	9_Tau
0.30	487.1	79.0	8.7	0.9	-	-	-	-	-
0.50	927.9	231.1	44.1	7.7	-	-	-	-	-
0.70	1201.5	416.8	113.7	29.2	6.9	-	-	-	-
1.00	1348.0	620.2	234.6	86.1	28.3	10.1	4.3	-	-
1.25	1254.2	696.3	329.1	142.9	59.0	28.1	14.9	7.1	3.0
1.50	1070.7	689.9	370.3	180.7	91.8	55.3	32.0	16.9	8.9
1.75	878.0	599.0	368.5	199.6	122.7	83.6	51.3	30.6	18.6
2.00	679.4	514.3	330.5	200.9	144.4	104.6	68.0	44.6	31.3

TABLE 4.5: Fixed Interval Strings, Pulses/String

Predicted Pulses/String from Computer Convolution
of Truncated Probability Density Function

Z	1_Tau	2_Tau	3_Tau	4_Tau	5_Tau	6_Tau	7_Tau	8_Tau	9_Tau
0.30	2.15	3.39	4.75	6.20	7.72	9.24	10.77	12.29	13.81
0.50	2.25	3.65	5.19	6.83	8.50	10.17	11.84	13.52	15.18
0.70	2.36	3.90	5.61	7.40	9.22	11.03	12.84	14.66	16.45
1.00	2.51	4.28	6.22	8.23	10.25	12.27	14.29	16.31	18.28
1.25	2.64	4.59	6.73	8.92	11.11	13.30	15.49	17.68	19.81
1.50	2.77	4.91	7.24	9.60	11.97	14.34	16.71	19.06	21.35
1.75	2.90	5.24	7.75	10.30	12.84	15.39	17.94	20.47	22.91
2.00	3.03	5.57	8.27	11.00	13.74	16.47	19.20	21.92	24.57

Corresponding Experimental Measurements

(Circuit II Data)

Z	1_Tau	2_Tau	3_Tau	4_Tau	5_Tau	6_Tau	7_Tau	8_Tau	9_Tau
0.30	2.14	3.34	4.71	6.11	-	-	-	-	-
0.50	2.24	3.60	5.14	6.69	-	-	-	-	-
0.70	2.34	3.84	5.54	7.24	8.80	-	-	-	-
1.00	2.47	4.20	6.10	7.91	9.83	12.15	14.21	-	-
1.25	2.60	4.50	6.56	8.50	10.78	13.30	15.23	17.04	19.17
1.50	2.74	4.82	6.98	9.14	11.80	14.17	16.20	18.30	20.70
1.75	2.85	5.13	7.40	9.84	12.74	15.04	17.25	19.69	22.48
2.00	2.97	5.40	7.91	10.69	13.62	15.97	18.47	21.28	23.98

Figure 4.11 shows the predicted and measured rates for the various fixed-interval string orders (τ , 2τ , 3τ to 9τ) as plots for eight different z values. The agreement is good at low order string numbers, but deteriorates at the higher orders. This may have to do with the cumulative numerical error inherent in the repeated convolution of a digitized function. The higher the string number, the greater the number of repeated passes required to generate the density function appropriate for that order. The same explanation applies to the pulses-per-string predicted vs measured data.

It is also worth noting that the "DENOM" columns in the raw data print out in Appendix 3, (the factor computed from $(f_a + f_c) \cdot (f_b + f_c + S) / f_c$) give values which are essentially equal to the input pulse rate N , as was the case with the fixed-number pulse string Circuit II data.

4.6 SIMULTANEOUS FIXED-NUMBER AND FIXED-INTERVAL STRING MEASUREMENT DATA

The details of these measurements were described in Section 3.2.3. Appendix 4B is a listing of the predicted data generated by the program which was written to compute fixed-interval string occurrence rates by repeated convolution of the truncated density function. The listing shows the percentage contribution of each of the fixed-number pulse strings (orders 1 to 17) to fixed-interval strings (orders 1 to 9). This follows the approach of determining the relative probability of an n -pulse fixed-number string qualifying as an $m\tau$ fixed-interval string, which was used to verify the analytical basis for deriving fixed-interval string occurrence rates.

The predicted data are listed for the eight z values which have been used throughout the experimental work. A given column of data shows the distribution (as percentages) of a given order of fixed-number pulse strings among the first nine orders of fixed-interval strings. The program made as many numerical convolution passes as were necessary to reach a

Figure 4.11: Fixed-interval string occurrence rates plotted against string number for eight different input count rates (Z values). Experimental data denoted by * with connecting line segments; predicted data denoted by circled dots.

point where additional passes produced a change of less than 1% in the predicted occurrence rate of the $9\text{-}\tau$ fixed-interval string.

For example 27 convolutions were required to achieve the necessary degree of accuracy for the data listed for $Z = 1.25$ (input count rate 12,500 c/s). The column marked "9" of this data set shows that fixed-number pulse strings of order $n = 9$, (i.e. 9 consecutive pulses with inter-arrival times $< \tau$) qualified as fixed-interval strings of order $\tau, 2\tau, 3\tau \dots 9\tau$ with percentage probabilities of 0.09, 6.51, 35.17, 39.94, 12.26, 1.04 and 0.02 respectively.

Appendix 4A shows samples of the raw data in a similar format printed out by the data acquisition system. In this case the data covered fixed-number strings up to order 40 (four lines for each order of fixed-interval string identified by the "tau" column).

A comparison of the predicted and measured data show only fair agreement in absolute terms. However it is evident that the correlation between the locations of maximum values in corresponding rows and corresponding columns of the "matrices" formed by the data sets is exact. One row of 17 numbers in the predicted data corresponds with the first row of 10 numbers and seven from the second row in the measured data for each order of fixed-interval string.

The predicted data are subject to cumulative errors due to the repeated numerical convolution of a digitized function as was mentioned previously. In addition the relative probabilities were obtained by dividing each absolute probability by the factor $(1-e^{-z})^{(n-1)}$ where n is the string order number. The precision of this computation obviously deteriorates as the value of n increases toward 20.

4.7 CONCLUSIONS

The purpose of this part of the study was to investigate the phenomenon of coincidences in Poisson-distributed pulse trains which cause errors in pulse-height analysis leading to distortion of the acquired spectra.

A classification structure has been defined, based on the work of *Tenney*⁵⁸, which covers all types of coincidences. It has been shown that a multiple pulse string (a sequence of pulses with inter-arrival times $< \tau$), can be analysed on the basis of the number of pulses it contains, or on the basis of the number of fixed-intervals, τ , which it covers. The theory developed in Chapter 1 for the occurrence rates of the fixed-interval strings has been shown to agree with that due to Tenney for the fixed-number strings.

Techniques for experimental measurements of the occurrence rates of both types of multiple pulse string have been investigated. It has been shown in Chapter 2 that compound random event rates such as these can be measured with relatively simple circuits (the "Circuit I" model for example), provided that a proper analysis is made. A practical and theoretically rigorous procedure has been described for ensuring that correct event rates are recorded, based on Markov process theory. This procedure involves the measurement of circuit transition rates and their incorporation into formulae which are easily derived and do not require either a detailed knowledge of Markov chain theory, or the solution of matrix equations.

The experimental data obtained agree closely with theoretical calculations based on the assumption that pulse inter-arrival times have an exponential probability density function. This agreement supports the generally accepted hypothesis that emissions resulting from radioactive decay are Poisson-distributed.

PART 2 PULSE PILEUP

OVERVIEW

There has been and continues to be a large amount of effort devoted to solving the problems introduced by pulse pileup and dead time in radiation spectra. Appendix 5 contains a summary review of previous work, as reported in the literature, through to 1989, (see in particular references 9, 10, 13, 14, 17, 19, 27, 29, 29-32, 35, 36, 39, 57, 58, 71).

This work can be classified into two broad categories as follows:

- (A) Situations which involve a changing total count rate during measurement times.
- (B) Situations where the total count rate is constant during measurement times.

In the case of (A) the solutions involve instantaneous corrections, at suitably short intervals, based on the instantaneous count rate average over the short intervals. The techniques for accomplishing this also fall into two classes:

- (A-1) Augmentation of the contents of each channel addressed by the ADC by an amount related to the instantaneous count rate;
- (A-2) Random injection of fixed amplitude pulses at an average rate related to the instantaneous count rate.

In the case of (B) the solutions are simpler and fall again into two identifiable classes:

- (B-1) Corrections based on overall count rate losses as determined from pulse injection methods or by an analysis of system BUSY signal duty cycles;

(B-2) Post acquisition correction of entire spectra to correct for distortion (including count rate errors) due to pileup.

The techniques involving pulse injection are enhanced by the use of well designed hardware pileup rejectors.

None of the techniques except (B-2) take into account the fact that spectral peaks (depending on their energies with respect to other peaks) will gain counts as well as lose them, and that this omission will cause an error which increases with spectral energy level. The peaks themselves are also subject to distortion which results in peak definition errors apart from simple magnitude errors. To date it appears that only *Wiclopolski and Gardner*^{35,36}, and *Johns and Yaffe*⁷¹ have attempted to make the (B-2) correction technique a practical proposition.

This study begins in Chapter 5 with a theoretical analysis of pulse pileup, and the development of Monte Carlo simulation models which emulate the action of modern pulse height analysis systems in assigning an energy level to a composite pulse formed by the partial superimposition of two normal pulses. It is shown that a discrete correction algorithm, developed on the basis of the theoretical analysis, can be used reversibly to induce or correct pileup distortion on the spectra generated by Monte Carlo simulations.

Chapter 6 extends the theory to include 2nd order pileup (the coincidence of three pulses) with demonstrations of the phenomenon, and the correction of the resultant distortion, by means of more sophisticated Monte Carlo simulations. Some additional theory concerning the probability of multiple coincidences of pulses within and between spectral channels is also developed.

Chapter 7 is concerned with verification of the theoretical analysis and the results of the Monte Carlo simulations by means of experiments involving the acquisition of spectra with a laboratory pulse height

analysis system. The main experiment involved the design and use of a hardware flat-spectrum generator which at low count rates produces equal count rates in all channels, but which is subject to 1st order pileup at higher count rates. The results provided quantitative data which were consistent with the theory and simulations. The final experiment involved the correction of a real radiation spectrum acquired at a high count rate for 1st and 2nd order pileup. In order to demonstrate the accuracy of the correction it is necessary to acquire a reference spectrum at a vanishingly low count rate which is identical in shape to the high count rate one except for the effects of pileup. It was only possible to accomplish this with a spectrum having a single peak, nevertheless the result did confirm the equivalence of the simulation models and the processes involved in acquiring real spectra.

CHAPTER 5 A THEORETICAL ANALYSIS OF PILEUP, DEVELOPMENT OF A DISCRETE CORRECTION ALGORITHM AND MONTE CARLO MODELS

5.0 INTRODUCTION

This chapter is concerned with 1st order pileup, that is the effects produced by a single coincidence involving two pulses. Such a coincidence produces a spectrum of partial sum pulses extending up to the algebraic sum of the two participating pulses. This coincidence spectrum is in essence a probability density function, but by analogy with filter theory could also be viewed as the impulse response due to coincidence. The shape of this function is analyzed and the way in which it contributes to pileup when a whole range of pulse amplitudes are involved is considered. An analytical expression is developed for the distortion produced in a flat spectrum, (one with identical count rates in all channels), with some simplifying assumptions. A discrete correction algorithm is described which can be used to remove the distortion due to 1st order pileup, and its efficacy is demonstrated with the application of it to pileup generated by Monte Carlo simulation techniques. These techniques and the results obtained with them lead to specific criteria for

the definition of 1st and 2nd order pileup and to the introduction of a new concept for the definition of pulse resolving time.

5.1 THE COINCIDENCE SPECTRUM OF TWO PULSES, (1st ORDER PILEUP)

When two pulses due to radiation detector events of specific energies sum at random, the peak amplitude as interpreted by an ADC will vary from that of the larger of the two, up to the amplitude which represents their algebraic sum. As a result a spectrum of pulse amplitudes will be recorded in this region as well as those due to the radiation events themselves, if a sufficiently large number of events occur in the recording period.

The shape of this coincidence spectrum, which is strictly speaking a probability density function, depends on the shapes of the participating pulses. If the pulses are square, then the coincidence spectrum or impulse response is a single line located at the amplitude corresponding to the sum of the amplitudes of the two pulses. The next simplest possible shape is the flat continuum which would be generated if the pulse shapes were triangular as shown in Fig. 5.1.1. The envelopes representing the composite pulse for various separations of two triangular pulses are shown in Fig. 5.1.1(a), with the spectrum due to summing with uniform random separations being shown in Fig. 5.1.1(b). It is not difficult to see that since the pulses consist essentially of linear segments, the maximum amplitude of the sum becomes a linear function of the pulse separation once that has decreased to less than half the pulse width. This in turn leads to a uniform or flat probability density function, for random separations having a uniform density function.

Most ADC's incorporate peak detection schemes such that the sample-and-hold circuitry is activated following the first peak of an incoming waveform. The pulse-height analysis cycle then proceeds with the remainder of the waveform being locked out, even if a higher peak exists. An ADC operating on this principle will thus record the true amplitude of

the first pulse of a pulse-pair unless coalescence has reached the point where its peak is augmented by the leading edge of the second one, or is no longer distinguishable from the peak of the composite sum pulse. In the case of triangular pulses this coalescence involves a transition from the peak amplitude of the first pulse to that of the second (assuming the first is lower than the second), followed by a continuous distribution of sum amplitudes beginning at the amplitude of the larger and extending up to the algebraic sum of the two, as shown in Fig. 5.1.1.

When pulse shapes are other than composites of linear segments then the situation is less simple. A widely used pulse shape is the so called "semi-Gaussian". This term covers a variety of shapes which approximate to equations of the form (see e.g. *Datlowe*³⁹):

$$\text{Pulse Amplitude} = t^n e^{-nt} \quad (5.1)$$

where t is in suitable units, e.g. microsecs. The larger the value of the exponent n , (which will be termed the "shape index") the more nearly the waveform approaches a true Gaussian shape. The coincidence spectrum for two such pulses of different amplitudes does not always begin at the amplitude of the larger one. There tends to be a "forbidden region" in which there are no partial sum amplitudes. This region is largest for pulses of equal amplitude and is due to the fact that the peak of the second pulse is augmented by summation with the falling tail of the first one before the peak of the first one has disappeared, i.e. coalesced with that of the second pulse. This sequence is shown in Fig. 5.1.2.

Fig. 5.1.3 shows a selection of coincidence spectra for different pulse shapes. These were generated using a Monte Carlo simulation in which a random variable having the exponential density function appropriate for a Poisson process was derived from a standard random number generator and used as the separation τ in a two-pulse equation of the form:

Figure 5.1.1: When two triangular pulses have a separation t which is less than τ (half the pulse width), the sum amplitudes produced as t decreases to zero generate a uniform coincidence spectrum.

Figure 5.1.2: The ADC will process the first maximum which it sees. As two pulses begin to "pileup" the recorded amplitude jumps from that of the first to that of the composite sum when the separation is less than a critical value which is a function of pulse shape. The middle plot shows the critical separation case.

Figure 5.1.3: Coincidence spectra (impulse responses) for different pulse shapes. For asymmetric shapes there is a "forbidden region" where no sum amplitudes exist. The true Gaussian shape (symmetrical) has no forbidden region.

$$y = a t^n e^{-nt} + b (t-\tau)^n e^{-n(t-\tau)} \quad (5.2)$$

The first maximum starting from $t=0$ was located for each value of τ generated, and was assigned a channel number as the next highest integer. A fixed pulse amplitude ratio a/b was selectable, with the coefficients a and b being interchanged for each new trial to provide an even probability that the order would be "large" followed by "small" or vice versa. A more detailed discussion of the criteria used to define pileup in these Monte Carlo simulations is given in Section 5.5.

The spectra of Fig. 5.1.3 closely resemble those obtained by other workers *Soucek*⁹, *DeLotto and Dotti*¹³, *Waibel*^{17,19}, *Datlowe*³⁰, and are typical of the coincidence spectra generated by conventional pulse shapes widely used in commercial spectroscopy amplifiers. These coincidence spectra are pileup probability density functions, but as indicated earlier can also be considered as impulse response functions in the context of modern signal processing theory and terminology.

5.1.2 THE THEORETICAL COINCIDENCE PROBABILITY DENSITY FUNCTION FOR SEMI-GAUSSIAN PULSES

Equation 5.2 gives the sum amplitude y of two semi-Gaussian pulses in terms of a random separation variable τ , and a time coordinate t . For a given value of the separation time τ , between the two pulses, there is one value of t measured from the start of the first pulse, which corresponds to the composite maximum of the coalescing pulse pair. Fig. 5.1.2 indicates that true partial sums only begin for values of τ less than about 0.8 of a rise time for pulses having a shape index (the value of the exponent n in equation 5.2) of 5, and that the locations of the maximum occur for values of t which are never less than unity. It is a property of the pulse shape given by:

$$y = t^n e^{-nt}$$

that the peak amplitude is reached at $t = 1$. Thus perfect superimposition of the two pulses occurs at $t = 1$ and $\tau = 0$. Since the maximum separation τ for a partial sum is of the order of 0.8, then the valid ranges of interest are (approximately) $(0 \leq \tau < 0.8)$ and $(1 \leq t < 1 + \tau)$.

The standard expression for finding the probability density function $f(y)$ of a random variable y , which is itself a function of another random variable x , having a probability density function $f(x)$, is given by the Jacobian:

$$f(y) = \frac{f(x)}{\left| dy / dx \right|}$$

subject to the restriction that y is a monotonically increasing (or decreasing) function of x .

In this case the problem is to find the probability density function of the composite maximum of the pulse pair given by (5.2) as a function of τ .

$$f(y) = \frac{f(\tau)}{\left| dy / d\tau \right|}$$

The inter-arrival times represented by τ have the probability density function (Chapter 1 section 3.2)

$$f(\tau) = \lambda e^{-\lambda\tau} \quad (\lambda = \text{event rate})$$

and from (5.2)

$$dy / d\tau = nb(t-\tau)^{n-1} \left(t - (\tau + 1) \right) e^{-n(t-\tau)}$$

and hence

$$f(y) = \frac{\lambda e^{-\lambda\tau}}{\left| nb(t-\tau)^{n-1} \left(t - (\tau + 1) \right) e^{-n(t-\tau)} \right|} \quad (5.3)$$

This function has a singularity when $t = \tau$, which as indicated earlier is not within the valid range to produce the composite maximum. It also

has one for $t = 1$, $\tau = 0$; which is the perfect superimposition case and is valid.

A significant point to note is that the density function as given by (5.3) is count rate dependent in as much as it is a function of the rate parameter λ .

In order to produce a plot of the required coincidence probability density function from (5.3) it is necessary to take a suitable range for τ with appropriately small increments, and then evaluate (5.2) at each τ value to find the composite maximum amplitude and the corresponding value of t . The value for $f(y)$ is then found by inserting these values of t and τ into (5.3).

A suitable program was devised to do this and plots are shown in Fig. 5.1.4. They are for a shape index (the value of n in equation 5.2) of 5 and show the effect of count rate on the shape of the coincidence impulse response. There clearly is an effect, however it would appear to be minimal over a wide range of count rates.

5.2 ANALYSING THE SPECTRAL DISTORTION DUE TO PULSE PILEUP

In this section the quantitative effect of 1st order pulse pileup (only two pulses participating) is considered. An extension of the theory to higher orders is discussed in Chapter 6.

As a starting point three simplifying assumptions will be made about the nature of the coincidence spectrum discussed in section 5.1. The first two assumptions will require some modification to conform with the real-world situation, and the theory involved is developed in Chapter 6. The three assumptions are as follows:

Figure 5.1.4: Theoretical coincidence spectra for a pulse shape index of 5 at different count rates. The probability shifts toward perfect superimposition of the two pulses as count rate increases. The spectra were plotted from equation (5.3). The lower plot shows the same thing with the peaks truncated and the plateau portions magnified.

- * The coincidence spectrum (which represents a probability density function) exists as a continuous function between the amplitude of the larger of the two pulses involved in the random summing, and the amplitude corresponding to their algebraic sum only, (as is the case for triangular pulses).
- * When the area under this curve is divided by twice the product of the count rates of the two pulses concerned, the result is τ , the effective resolving time of the system. In other words the coincidence rate between pulses having rates N_1 and N_2 is assumed to be $2\tau N_1 N_2$, an assumption which strictly speaking is only true if the inter-arrival times have a uniform density function.
- * For a given data acquisition system the shape of the coincidence spectrum is unaffected by the pulse amplitudes involved in the summation, apart from horizontal and/or vertical axis scale changes.

The pileup process encompasses all channels in a spectrum and produces a pileup spectrum which extends to twice the "active spectrum", a term that will be used to define the spectrum up to the channel number which represents the maximum energy recorded by the system. A simplified picture of the process is shown in Fig. 5.2.1. Pulses in channels A, B and C sum randomly with pulses in channel M to produce a distribution of counts in channels above M, according to suitably scaled representations of the probability density function. The distributions of each set of coincidence summations ($A+M$), ($B+M$), ($C+M$) and ($M+M$) are shown in Fig. 5.2.1(a). Consider now the total contribution to channel ($M+J$) from their summations. It will receive contributions from coincidences of those pairs whose channel numbers have a sum greater than or equal to ($M+J$). Fig. 5.2.1(b) shows a more generalized representation where the pulses in channel I sum with those in channel M to produce a probability density function spectrum from M to ($M+I$). Channel M can be thought of as a "mirror" channel in which all channels i , ($i = 1, M$) are reflected. All channels i will make a contribution to ($M+J$) for $i \geq J$.

Figure 5.2.1 (a): The coincidence spectrum above channel M is a summation of the individual spectra generated by coincidences (M^*A) , (M^*B) , (M^*C) and (M^*M) . (b) the $(M+J)^{\text{th}}$ channel receives contributions from all coincidences (M^*I) for which $I \geq J$.

The actual amount of the contribution will be a slice of the probability density function shown shaded in Fig. 5.2.1(b). The total area under the function will be given by

$$\int_M^{M+I} f(PU) = 1$$

where $f(PU)$ is the coincidence spectrum/probability density function. The slice applicable to channel $(M + J)$ is given by

$$PROB..J = \int_{J-1}^J f(PU) \quad (5.4)$$

Clearly the total contribution to channel $(M + J)$ from coincidences in all channels up to and including M can be found by a double summation.

For coincidences between a channel I and the mirror channel M , the function $f(PU)$, must be digitized into I slices, so that the width of a slice corresponding to the contribution to a given channel e.g. $(M + J)$ is $(1/I)$. This makes the contribution $Prob..J$ a function of I as well as J . Any attempt to compute $Prob..J$ in a discrete summation algorithm using an experimentally acquired coincidence spectrum would be very time consuming, since it would have to be redigitized for each increment of I .

A more realistic approach to that problem is to find an analytic function which can be used to represent the coincidence probability density function. As such it must have a total area in the region in which it exists $(M \text{ to } M + I)$ of unity. Ideally such a function would have I as a parameter and could be integrated analytically to produce a probability distribution function, $F(PU)$.

If the integration were to be carried out between the limits $(J-1)$ to J , then the resulting cumulative distribution function would have I and J as variables and would be such that inserting appropriate values of I and J

produces the probability slice Prob..J directly. It would thus be entirely suitable for use in a discrete algorithm.

5.3 A DISCRETE SUMMATION ALGORITHM TO SIMULATE PULSE PILEUP

Clearly any algorithm of this nature will be implemented by a computer and the most commonly accepted language for scientific modelling of this kind is still FORTRAN. Accordingly a basic familiarity with some FORTRAN terminology will be assumed in what follows.

Since pulse pileup is an asymmetric operation, causing increasing accumulations from low to high channels, it is logical to start the summation at the lowest channel which receives a contribution, i.e. channel #2. With that reasoning the ranges for I, M and J can be started at channel #1.

The algorithm for pileup contributions (losses are considered in a later section) can then be written as a triple nested DO loop which gives the contribution to channel (M + J) (in a highly simplified form) as follows:

```
DO  M = 1, MAX..CHAN                                (5.5)
  DO  I = 1, M
    DO  J = 1, I

      PROB..J = FUNCTION (F(PU) I, J)
      GAIN..M..J = COUNT (I) * COUNT (M) * 2 * TAU * PROB..J
      CHAN (M + J) = CHAN (M + J) + GAIN..M..J

    END DO
  END DO
END DO
```

The first execution of the multiple loop results in channel #2 receiving the entire contribution of coincidences between pulses in channel #1, since

the probability density function will be digitized into a single slice, i.e. it will reside entirely in channel #2.

A FORTRAN program was written which generated model gamma radiation spectra with such parameters as the number, amplitudes and full widths at half maximum (FWHM) of peaks, and the amplitude and exponent of an exponential background, being specified by keyboard entry. Provision was also made for generating a simple uniform flat spectrum. Other entry parameters included the resolving time τ and the total count rate. The pileup algorithm described above was incorporated so that the model spectra could be distorted and/or corrected.

5.3.1 CORRECTION FOR 1st ORDER PULSE PILEUP USING THE ALGORITHM IN REVERSE

The previous section was concerned with gains to a specific channel as a result of pulse pileup. For every pulse gained however there are two pulses lost from a point lower down in the spectrum, (as indicated earlier the discussion here will be limited to 1st order pileup). As a result all channels both gain and lose pulses with the single exception of the lowest channel, which will be defined as channel #1. This channel cannot gain any pulses by summation since there are none below it. (The possibility that there are pulses below the ADC threshold which can combine to produce pulses in channel #1 will be considered in Chapter 7.2.2.) Assuming for the moment that this is true, then channel #1, contains only pulses which were not involved in coincidences.

*Wytenbach*²⁴ first pointed out that pulses in a sequence with overall count rate N , which are not involved in coincidences, must be preceded and followed by resolving times τ , and that their rate will consequently represent a fraction $e^{-2N\tau}$ of the total rate N . This assumes that N is known, an assumption which will be made at present for the purposes of the discussion which follows.

The same argument can be applied to all channels as far as losses are concerned, so that the entire spectrum can be corrected for losses by multiplying the contents of each channel by the factor $e^{2N\tau}$. The only channel for which this produces the correct count rate however is channel #1. All others have gains due to pileup which must be corrected. Channel #2 contains only gains from channel #1, for which we have the correct rate, hence it is possible to compute the gain for channel #2 as:

$$\text{channel \#2 gain} = C_1^2 \tau$$

where C_1 = true count rate in channel #1. This gain however was augmented by the factor $e^{2N\tau}$ in correcting for the losses, so the correction to channel #2 becomes:

$$\text{channel \#2 correction} = -C_1^2 \tau (1 + e^{2n\tau}) \quad (5.6)$$

The procedure is now clear, by starting with channel #1, for which the true count rate is known, it is possible to correct channel #2. Channel #3 can then be corrected since it only receives pileup contribution from channels #1 and 2. Proceeding in this fashion it is then possible to correct the entire spectrum, since the pileup process is cumulative and proceeds from low to high channels.

Specifically the algorithm for accomplishing the correction of spectral distortion due to pileup is essentially a minor modification of the one described in the previous section for inducing pileup. The concept is that channels below and including M the mirror channel, having been corrected, are used in coincidences with M to calculate the correction to the channels above M , where again M initially takes the value 1. The DO loop for the correction algorithm is then:

```

DO  M = 1, MAX..CHAN
  DO  I = 1, M
    DO  J = 1, I

      PROB..J = FUNCTION (F(PU) I, J)
      CORRECTION..M..J = COUNT (I) * COUNT (M) * 2 * TAU * PROB..J
      CHAN (M + J) = CHAN (M + J) - CORRECTION..M..J

    END DO
  END DO
END DO

```

It is a prerequisite that the distortion and correction algorithms which have been described must be complementary, i.e. the distortion induced by one must be accurately corrected by application of the other if the technique is ever to be viable. Application of the two algorithms to artificial spectra generated by the program described earlier verified that this was the case. More rigorous checks of the validity of the method are described in subsequent sections.

The essential differences between the pileup correction technique developed by *Wielopolski and Gardner*^{32,35} and used by *Johns and Yaffe*⁷¹, and the correction algorithm described above are as follows:

- * The present algorithm is designed to operate directly on a distorted spectrum and to produce a correction in a single pass. This is based on the assumption that the lowest channel in the spectrum suffers losses but receives no pileup gains.
- * The Wielopolski and Gardner technique involves the iterative distortion of a spectrum. This spectrum must be provided as the best initial estimate of the true undistorted spectrum. The estimate is then updated at each iteration until the distorted result matches the

observed spectrum, at which time the updated estimate is deemed to be the sought true undistorted spectrum.

5.4 THE EFFECT OF PILEUP ON A FLAT SPECTRUM

A good insight was obtained into the distorting effects of pulse pileup by reducing the problem to the minimum level of complexity. This consisted of (a) assuming that the probability density function for coincidences was uniform, i.e. that the coincidence spectrum was flat with pileup contributions to all affected channels being identical, and (b) assuming that the spectrum in which coincidences occurred was also flat, i.e. a constant count rate in all channels.

Application of the distorting algorithm under those conditions produced the result shown in Fig. 5.4.1 which distorts the flat spectrum into a ramp and adds a tail which might be described as pseudo-exponential and stretches as would be expected from the upper end of the recorded flat spectrum (the active spectrum) to twice the equivalent maximum energy.

5.4.1. A THEORETICAL ANALYSIS OF SPECTRAL PILEUP

The simplicity of the ramp function resulting from the pileup of a flat spectrum was somewhat surprising and it was deemed worthwhile to attempt a more rigorous theoretical analysis, rather than to rely on a largely empirical discrete computer model. The necessary function and double integral were formulated on the basis of the concept shown in Fig. 5.4.2. In this representation channels shrink to zero width as the discrete algorithm changes to a continuous function version. The spectrum is specified by its amplitude at any point (equivalent to count rate) and channels become points on the horizontal or energy axis. The point marked m is analogous to the mirror channel and the point marked r is analogous to the channel I . The point marked N is analogous to channel $(M + J)$. We want to arrive at an expression which will give the contribution at point N of coincidences between points r and m . We

Figure 5.4.1: Result of applying the discrete pileup algorithm to a flat spectrum. The probability density function used for this numerical computation was uniform with no forbidden region.

Figure 5.4.2: For a continuous function derivation of the pileup at point N , due to coincidences between points r and m , the function giving the spectrum amplitude at any point r , $S(r)$, is required, together with a function $f(PU(r, N-m))$, which gives the pileup density functions as an analytical expression.

assume as before that such coincidences will produce a probability density function extending from m up to $(r + m)$ and that point N will receive a contribution (in this case multiplied by the infinitesimal dr) of that function. It is clear that point N can only receive contributions provided that:

- (i) $m \geq N/2$
- (ii) $r \geq N-m$

We also need analytic functions for:

- (a) The spectrum amplitude as a function of any point r along it in the form:

$$\text{Spectrum amplitude} = S(r)$$

- (b) The coincidence probability density function amplitude at any point $(N-m)$ along it, with r being a parameter, in the form:

$$\text{Pileup density function} = f(PU(r, N-m))$$

Given these functions, the pileup contribution ΔN to point N in Fig. 5.4.2. can then be expressed as the double integral:

$$\Delta N = \int_{m=N/2}^N \int_{r=(N-m)}^m 2\tau \cdot S(m) \cdot S(r) \cdot \left\{ f(PU(r, N-m)) dr \right\} \cdot dm \quad (5.8)$$

This expression is analogous to the discrete version in that the contribution is twice the product of two amplitudes (count rates), the resolving time τ and an infinitesimal increment of the relevant coincidence probability density function at the appropriate point.

This would be a daunting integral indeed if the functions $S(r)$ and $f(PU(r, N-m))$ were even moderately complicated. However it becomes

entirely manageable if the spectrum and density function are both flat, then $S(r)$ is a constant for all r and the value of $f(\text{PU}(r, N-m))$ is simply $1/r$ for all points along it, since it is uniform over a segment r .

Thus the expression (5.8) for a flat spectrum of constant amplitude C with a uniform coincidence probability density function, reduces to

$$\begin{aligned}\Delta N &= \int_{m=N/2}^N \int_{r=(N-m)}^m 2\tau \cdot C^2 \cdot (1/r) dr \cdot dm \\ &= 2C^2\tau \int_{m=N/2}^N (\ln(m) - \ln(N-m)) dm\end{aligned}$$

Integration by parts leads to the expression:

$$\Delta N = 2C^2\tau \left[m(\ln(m) - 1) + (N-m)(\ln(N-m) - 1) \right] \Bigg|_{m=N/2}^{m=N} \quad (5.9)$$

From which:

$$\Delta N = 2C^2\tau N \ln(2)$$

This expression thus confirms that for a uniform coincidence probability density function, the pileup contribution ΔN to each channel N in the spectrum is directly proportional to the channel number, thereby causing a flat spectrum to be distorted to a linear ramp. It also gives the slope of this ramp in terms of the count rate C in each channel and the resolving time τ of the incoming pulses.

The expression becomes more general if we define $\Delta = \Delta N/N$, as the slope of the ramp in units such as (c/s/channel) per channel, and observe that the factor $2C^2\tau$ is a simplified expression for the cross-channel coincidence rate, (in a flat spectrum they are all identical). With these two modifications we have:

$$\text{Ramp Slope } \Delta = (\text{cross-channel coincidence rate}) \ln(2) \quad (5.10)$$

With the limits given in (5.9) the receiving channel N is also assumed to be one in which incoming pulses are counted. In order to find out what the pileup contribution is to channels above the active part of the spectrum, it is only necessary to limit the range of m the mirror channel, to a channel which is less than N, say M. With these limits the expression (5.9) becomes

$$\Delta N = 2C^2 \tau \left\{ M \left(\ln(M) - \ln(N-M) + N \right) \left(\ln(N-M) - \ln(N/2) \right) \right\} \quad (5.11)$$

This expression is only valid for $N > M$ ie. the situation where channel N is above the active part of the spectrum and receives pileup contributions only. A plot of (5.10) for the active portion of the spectrum where $N < M$ together with (5.11) for $N > M$, is shown in Fig. 5.4.3, which is clearly the same as that produced by the discrete algorithm discussed previously and shown in Fig. 5.4.1.

The theoretical expression for pileup contributions (5.8), is a general one which allows (in theory at least) 1st order spectral pileup to be calculated for any spectrum if analytic functions for the spectrum shape and coincidence probability density function are available, and if the resulting integral is soluble. The relatively simple solution for the case of a flat spectrum and uniform coincidence density function confirms the linear ramp result produced by the discrete algorithm, and in addition provides a quantitative result for the slope of the ramp in terms of the count rate and resolving time τ . The exercise thus provides some credibility for the veracity of the discrete algorithm and hence some confidence that the results for more complicated spectra are likely to be reliable.

5.4.2 A FLEXIBLE MATCHING FUNCTION TO REPRESENT $f(\text{PU}, I, J)$

The next step was to investigate the more realistic case of a non-uniform coincidence probability density function of the type shown in Fig. 5.1.3. A

Figure 5.4.3: A comparison of the theoretically derived pileup in a flat spectrum from equations (5.10) and (5.11), with the results of applying the discrete algorithm shown in Fig. 5.4.1.

relatively simple empirical function to match any of these can be synthesized from a pedestal and a reciprocal function of the form:

$$f(PU) = c + \frac{1}{k(1 - i/I) + 1} \quad (5.12)$$

the form of which is illustrated in Fig. 5.4.4. The constants C and k can be adjusted to produce almost any variant likely to be required, while the variable i is in a form which allows adaptation to the discrete algorithm where digitizing of the function into slices each of width $(1/I)$ will be required.

The range of this function for the purposes of simulating the coincidence probability density function is from $i = 0$ to $i = I$. The area over this range is given by

$$\begin{aligned} \text{Area} &= \int_{i=0}^{i=I} \left\{ C + \frac{1}{k(1 - i/I) + 1} \right\} di \\ &= \left[Ci - (I/k) \ln(k(I-i) + I) \right]_{i=0}^{i=I} \\ &= CI + (I/k) \ln(k + 1) \\ &= I \times \text{NORM} \end{aligned}$$

where $\text{NORM} = C + (1/k) \ln(k + 1)$

This is a constant factor which is used to normalize the function (5.12) to unity as required for a probability density function. By changing the limits on the integral to cover the width of a single channel j from $i = (j - 1)$ to $i = j$, the slice of the probability density function appropriate to channel j is given by:

Figure 5.4.4: The pileup probability density function can be represented by a pedestal of height c , and a reciprocal function with parameters k and I for shape and for digitizing into I discrete channels respectively.

$$\begin{aligned}
 \text{PROB..J} &= \frac{1}{I \cdot \text{NORM}} \left[C_i - (1/k) \ln \left(k(I-1) + 1 \right) \right]_{i=j-1}^{i=j} \\
 \text{PROB..J} &= \frac{1}{\text{NORM}} \left\{ \frac{C}{I} - (1/k) \ln \left(\frac{k(I-j) + 1}{k(I-j-1) + 1} \right) \right\} \quad (5.13)
 \end{aligned}$$

This expression is used to compute the value of the term denoted as "PROB..J = FUNCTION (F(PU) I, J)" in the DO LOOP of equation (5.5), with the value of I being incremented up to M by one DO LOOP and the value of J from 1 to I within it by the second one.

The function (5.13) was incorporated into the discrete algorithm with the values of k and c being selected by keyboard entry. When it was used to distort a flat spectrum, the result was still a linear ramp. The slope however was reduced by a very small amount and the shape of the tail in the channels above the active portion of the spectrum was altered slightly. Since the algorithm had been verified by a theoretical derivation in the case of the simple uniform coincidence spectrum, no attempt was made to incorporate the function (5.12) into the continuous function solution of (5.8) and solve it directly. Similarly the real-life truncation of the coincidence spectrum above the mirror channel, (the "forbidden region" discussed earlier) was incorporated into the discrete algorithm without difficulty and the results showed again that the linear ramp pileup characteristic was modified only slightly in that the slope was marginally different. The theory involved in implementing the truncation is covered in Chapter 6.

This exercise indicated that modifications to the shape and range of the coincidence spectrum of two pulses did not alter the basic pileup spectrum to a significant extent, since any major changes would have been readily apparent on something as simple as a linear ramp. Further experiments involving pileup induced in artificially produced flat spectra and their correction are detailed in a later discussion.

5.5 PILEUP SIMULATION USING MONTE CARLO TECHNIQUES

Reference was made in Section 5.1 to the use of computer-generated random variables to produce the necessary Poisson distribution of pulse inter-arrival times τ in the two-pulse equation (5.2).

$$y = at^n e^{-t} + b(t-\tau)^n e^{-(t-\tau)}$$

For each value of τ , a program loop generates the composite pulse amplitude represented by (5.2) for small increments of t until a maximum is reached, which is the value the ADC would assign in a multichannel analysis system.

The detailed algorithm is a little more subtle in that a single "event" in this context involves the "processing" of more than one pulse. Reference to Fig. 5.4.5 will illustrate the logic of what follows.

For 1st order pileup, three inter-arrival times τ_1 , τ_2 and τ_3 are picked by the random number generator having an exponential distribution. These intervals separate four pulses, A, B, C and D, with their amplitudes being determined according to whether the experiment is designed to produce an impulse response from the coincidence of two pulses of fixed amplitudes, or some other spectrum with a specified range of active channels. The fate of pulse B only is then determined as being either (a) "single", with no contribution from adjacent pulses; or (b) involved in pileup with the following pulse C only; or (c) involved in pileup with the previous pulse A and or C and D together. Only coincidence between B and C with no contribution from A or D is considered as legitimate 1st order pileup. The first step is to locate the first maximum of the composite pulse formed from the addition of pulses B and C, which are separated by τ_2 . This maximum is located at a time from the start of τ_2 which is defined as TMAX. Four criteria are then applied in the order given below to determine the classification of pulse B.

Figure 5.4.5: The criteria for determining whether pulse B is single, or involved in 1st order pileup with pulse C, or rejected because pulses A and/or D are too close.

- * If τ_1 is short enough that the tail of pulse A contributes more than one channel of amplitude to the maximum of pulse B, then the event is classified as "tail pileup" and no channel is incremented.
- * ELSE
If the contribution of the leading edge of pulse C at TMAX is less than one channel, the event is classified as SINGLE and the contents of the channel corresponding to pulse B are incremented.
- * ELSE
If the contribution of the leading edge of pulse D at TMAX is less than one channel, the event is classified as DOUBLE and the channel corresponding to the composite sum is incremented.
- * ELSE
Pulse D contribution at TMAX is greater than or equal to one channel and the event is pileup of order higher than one, in which case no channel is incremented.

When an impulse response experiment is selected, then the amplitudes (channels) of A, B, C and D are fixed by keyboard entry.

The simulations generated for both impulse responses (coincidence probability density functions) and other spectra are thus realistic and consistent with the real-world behaviour of both pulses and ADCs. "SINGLE" and "PILEUP" counters keep track of single and legitimate 1st order pileup events.

5.5.1 RE-DEFINING THE PULSE RESOLVING TIME τ

The criteria for pileup detailed in the previous section were established according to the behaviour of modern real-world ADCs as discussed in section 5.1. The processing times for these devices now tend to be less than the decay times of pulses typically found in modern gamma and X-

ray analysis systems. As such they do not contribute to the dead time encountered in these systems, and are ready to analyze another pulse as soon as the baseline has returned to a predetermined level close to zero, a time which is a function of the pulse decay time, and its amplitude.

These considerations suggest that it would be more realistic to redefine the pulse resolving time as a two-part parameter consisting of an effective or equivalent rise time τ_R and fall time τ_F . The concept proposed by Wytttenbach²⁴ that a single pulse must be preceded and followed by a time equal to the resolving time, led to the probability of a single pulse event being given by

$$P(\text{single}) = e^{-2N\tau}$$

and the probability for 1st order pileup being given by

$$P(\text{1st order PU}) = e^{-2N\tau} (1 - e^{-N\tau})$$

(See Chapter 1, section 1.2.2)

With the new definition these expressions would be modified to read:

$$P(\text{single}) = e^{-N(\tau_F + \tau_R)}$$

$$P(\text{1st order PU}) = e^{-N(\tau_F + \tau_R)} (1 - e^{-N\tau_R})$$

The occurrence rates for single and 1st order pileup events, S and $PU_{..1}$, are then given by

$$S = N e^{-N(\tau_F + \tau_R)} \quad (5.14)$$

$$PU_{..1} = N e^{-N(\tau_F + \tau_R)} (1 - e^{-N\tau_R}) \quad (5.15)$$

These modifications are consistent with the premise that the space which must precede a single pulse is equal to the fall time of the preceding

pulse, while the time which must follow the start of a pulse need be no more than its rise time for a correct analysis to be made by a modern nuclear pulse-height ADC.

It should be noted however that the fall time of the preceding pulse will in practice depend both on its amplitude, and on the predetermined threshold above the base line to which the level must fall before the ADC is ready to process another pulse. The latter is frequently a function of the lower level discriminator (LLD) setting. For the purposes of these simulation experiments it was decided to define τ_F effectively by the first of the four criteria above, in which a one-channel contribution from the preceding pulse is cause for rejection.

Equations (5.14) and (5.15) can be used to compute τ_F and τ_R if the single and 1st order pileup rates and true overall count rate N are all known. From (5.14):

$$(\tau_F + \tau_R) = \left\{ \ln(N/S) \right\} / N \quad (5.16)$$

from (5.14) and (5.15)

$$(PU..1) = (S) (1 - e^{-N\tau_R})$$

$$\tau_R = (1/N) \ln \left(\frac{S}{S - PU..1} \right) \quad (5.17)$$

The Monte Carlo simulations provided an ideal method of testing this hypothesis and produced results for τ_F and τ_R in impulse response runs which were entirely consistent with the actual rise and fall times of the pulses as specified by keyboard entry of the rise times and shape indices. The program computed the values of τ_F and τ_R from the data it generated and Table 5.1 lists the ratio of pulse amplitudes and the resolving times recorded. The pulse shapes were specified with a shape index of 5 and a peak time of 2 microseconds. The mean amplitude of the

two pulses involved was arranged to be in channel 125 of the spectrum generated and totals of 10^4 or more events were accumulated for each simulation. The data in Table 5.1 show a remarkable independence of τ_F and τ_R from pulse amplitude ratio.

Table 5.1

Values of τ_F , τ_R and Total Resolving Time τ_T computed from 1st order Impulse Responses at different count rates and with different pulse amplitude configurations (mean pulse amplitudes centred at channel 125).

Count Rate (c/s)	Pulse Amplitude Configuration		τ_F	τ_R	τ_F/τ_R	τ_T
150,000	125	125	4.109	1.830	2.245	5.939
	100	150	4.106	1.834	2.239	5.940
	70	180	4.093	1.843	2.221	5.936
	30	220	4.101	1.811	2.264	5.911
250,000	125	125	4.043	1.910	2.116	5.954
	100	150	4.028	1.930	2.087	5.958
	70	180	3.986	1.970	2.023	5.956
	30	220	3.988	1.938	2.058	5.926
500,000	125	125	3.725	2.221	1.677	5.947
	100	150	4.028	1.930	2.087	5.958
	70	180	3.986	1.970	2.023	5.956
	30	220	3.988	1.938	2.058	5.926

5.6 APPLICATION OF CORRECTION ALGORITHM TO MONTE CARLO PILEUP SIMULATIONS

The program with the algorithm designed to correct or distort spectra has provision for keyboard entry of the constants k and c for computation of one-channel slices of the probability distribution $F(\text{PU})$ according to (5.13). Experiments in the distort mode were conducted on a spectrum with all the counts in channel 125, i.e. a spectrum with a single line in it. The algorithm computes the pileup due to self coincidences in this channel according to the $F(\text{PU})$ constants and the values of τ_F and τ_R as entered and produces a corresponding impulse response. Fig. 5.6.1 shows the impulse response due to pulses in channel 125 generated by the Monte Carlo simulation, together with that produced by the discrete algorithm used as described above.

The values of τ_F and τ_R used in the distortion algorithm were those computed by the Monte Carlo simulation, with the $f(\text{PU})$ function constants k and c being selected empirically for the best fit to the Monte Carlo produced pileup impulse response. The goodness of fit is considered remarkably close for the simple function that was chosen to match the real $f(\text{PU})$. A further verification of the accuracy of the algorithm was that the areas of the Monte Carlo and algorithm-generated pileup distributions (as computed by integrating the channel counts) were identical.

5.6.1 GENERATION OF A MONTE CARLO FLAT SPECTRUM WITH PILEUP

It was felt that a good test of the correction algorithm would undoubtedly be the removal of pileup from a flat spectrum if such a spectrum could be generated in a realistic way. A fairly simple modification of the Monte Carlo simulation program which produced two-pulse coincidence spectra enabled a flat spectrum, with the input count rate and the number of active channels specified, to be generated. The modification was to make the amplitudes of the pulses A, B, C and D ,

Figure 5.6.1: Comparison of the Monte Carlo 1st order impulse response for pulses of equal amplitude, with that produced by the application of the discrete algorithm to a single line spectrum using the matching probability density function with optimum k and c values.

random variables by calling the random number generator having a uniform distribution in the interval $0 \leq x \leq 1$ and multiplying the values obtained by the number of active channels desired.

The result was a linear ramp with a pseudo-exponential tail similar in all respects to those shown in Fig. 5.4.3. When the correction algorithm was applied, using the values of the constants τ_F , τ_R , c and k determined from the coincidence spectra, a flat spectrum without a pileup tail was restored. The piled-up spectrum as generated by the Monte Carlo simulation and the corrected version are shown in Fig. 5.6.2 together with the distortion produced on a flat spectrum by the correction algorithm in reverse.

The obvious success of this test provided reasonably convincing evidence on a number of points:

- * The Monte Carlo simulation confirmed the theory that a flat spectrum subject to 1st order pileup will produce a linear ramp in the active portion.
- * The determination of the unknown resolving time components τ_F , τ_R from the impulse response coincidence spectra was correct.
- * The theory underlying the correction algorithm was essentially correct.

In addition it demonstrated that the concept of a flat (albeit artificial) spectrum as a model for the development of correction algorithms has considerable potential. Once again there is an obvious analogy with filter theory, where the characteristics of an undesirable filter can be determined by its response to a step function and in some cases an inverse filter can be designed to remove the undesirable effects. In this case the flat spectrum is the step function and pulse pileup is the undesirable distortion the effects of which must be removed.

Figure 5.6.2 (a): Piled-up flat spectrum produced by a Monte Carlo simulation. **(b)** That produced by the discrete algorithm (applied to a set of channels with identical numbers) using the same τ_R and τ_F values. **(c)** The Monte Carlo flat spectrum restored by application of the discrete correction algorithm.

The effects of a small mismatch of the constants used in the correction algorithm are immediately apparent with the flat spectrum. An example is shown in Fig. 5.6.3, where the constants used were deliberately chosen to cause over-correction. The result is strikingly similar to under-correction in another domain of spectral processing, the analogous one being pulse height analysis, where the pulse shapes themselves are optimized by pole-zero cancellation to reduce undershoot. These effects would be much less obvious, and hence more difficult to remove, if the spectrum was a conventional one as normally acquired from radiation detectors.

5.6.2. GENERATION OF MONTE CARLO SPECTRA WITH GAUSSIAN PEAKS

The Monte Carlo simulation program was further modified so that Gaussian-shaped spectral peaks could be generated with and without pileup, with the number, locations, relative amplitudes and resolutions of the peaks (Full Widths at Half Maximum, FWHM) being specified by keyboard entry. Pulse heights are generated by calling a random variable with a normal distribution, mean of zero and standard deviation of unity. The numbers obtained are multiplied by an appropriate factor to produce the desired FWHM and then added to a specified peak centre channel. If for example three peaks are specified with relative amplitudes $a:b:c$, then the peak to which the event is assigned is determined by sampling the uniformly distributed random variable $gener$ over the interval $(0 \leq x \leq 1)$. This interval is divided into subinterval widths in the required amplitude ratio $a:b:c$, with the peak being chosen according to which interval contains the random number sampled.

The program always produces two spectra simultaneously, one with and one without pileup in order that the efficacy of the pileup correction can be evaluated. The criteria for single events, legitimate 1st order pileup and others are identical with those described in Section 5.5.

Figure 5.6.3: The piled-up flat spectrum of Figure 5.6.2 over-corrected by using different τ_R and τ_F values. The effects of such mis-matches are easier to identify and interpret on a simple spectrum such as this than would be the case with one having gaussian peaks.

5.6.3 CORRECTION OF PILEUP IN A SPECTRUM WITH GAUSSIAN PEAKS

Figures 5.6.4 and 5.6.5 show the Monte Carlo simulations of a spectrum containing three peaks. The energies were deliberately chosen so that the pileup tail from the large amplitude peak at left, would cause the maximum distortion of the two smaller peaks to the right of it. Figure 5.6.4 shows the Monte Carlo spectrum distorted by pileup, together with the result of applying the correction algorithm to remove the distortion, and of applying it to the original reference spectrum to match the distortion. They are normalized to have the same areas. This method of presentation shows at what point along a spectrum the losses due to coincidences become equal to gains from coincidences in preceding channels. In this case that point is at about channel 85 in Figure 5.6.4. It is seen that the two peaks are severely distorted by the pileup, and that they themselves produce a pileup tail with a maximum at about channel 200.

Figure 5.6.5 shows the portion of Figure 5.6.4 from channel 80 to 220 on an expanded scale, with the original reference spectrum added. The data point symbols are un-connected circles, zero values are omitted. The fit of the correction to the reference is so close that the circles appear to be connected, which is why the reference spectrum was not included in Figure 5.6.4. The values of τ_F and τ_R used in the correction/distortion operations were those generated by the Monte Carlo simulation. The performance of the algorithm in removing the pileup distortion is very satisfactory and again demonstrates that the theory which has been developed to implement it is soundly based.

The correction to the large peak (not shown) is covered almost entirely by the linear multiplication which is carried out across the entire spectrum to correct for losses, since there are virtually no gains involved in this peak.

Figure 5.6.4: A Monte Carlo simulation (unconnected dots) showing the effect on two small peaks of pileup from a large one. Superimposed on it are the results of applying the algorithm (a) to correct the distortion, and (b) to match it.

Figure 5.6.5: An expanded portion of the spectra in the previous figure with the original reference spectrum added. (Data points are unconnected circles, zero values are omitted.) (a) Monte Carlo spectrum (b) matching distortion produced by algorithm (c) correction by algorithm removes pileup and closely matches reference spectrum.

5.6.4 ERRORS AND NOISE SOURCES DUE TO THE CORRECTION ALGORITHM

The various noise (and hence error) sources can be broken down as follows:

- * Statistical uncertainties in original (undistorted) channel counts. These are Poisson-distributed with a standard deviation which is the square root of the accumulated count (not count rate).
- * Amplification of these uncertainties by the loss factor $e^{N(\tau_F + \tau_R)}$. This operation compensates for losses from all channels due to pileup, but artificially amplifies the Poisson distributed uncertainties by this factor. It also amplifies any differences between the true pileup profile of a spectrum and that generated by the algorithm operating in reverse on a reference spectrum. The magnitude of this effect can be seen in the region covered by channels 85-100 in Figure 5.6.5. The difference between true pileup and the distortion generated by the algorithm operating in reverse is the source of the amplified negative error (shown shaded) which appears in the correction. Ultimately this is due to a slight mismatch between the real impulse response function and the one used as a match for it. Obviously a balance has to be struck between the time spent on "fine-tuning" this match, and the accuracy required from the correction.
- * Statistical uncertainties involved in computing gains from products of channel counts which already include amplified re-cycled errors from previous corrections. As the correction process proceeds, the channel being corrected loses an ever larger number of discrete contributions for coincidences which are deemed to have occurred between channels below it. This multiple re-cycling process however must surely produce sufficient randomness that the added noise is essentially independent of other noise sources. The RMS

value of this noise however must increase in a non-linear fashion with increasing channel number, since each additional correction involves a contribution from one more of the previously corrected channels than did the correction which preceded it.

From the above analysis it seems likely that the worst-case spectrum would be one with all channels having an equal count rate, i.e. a flat spectrum, since each channel contains a linearly increasing fraction of pileup gains, from coincidences between a linearly increasing number of channels. Correction of the tail above the active portion would thus introduce the maximum possible cumulative noise. The result of correcting the flat spectrum shown in Figure 5.6.2 indicates that the noise level is nevertheless low enough to make this correction technique a viable one. The correction of the Gaussian peaks shown in Figure 5.6.5 also provides fairly strong evidence in support of that conclusion.

CHAPTER 6 EXTENSION OF THE THEORY AND CORRECTION ALGORITHM TO INCLUDE 2ND ORDER PILEUP

6.0 INTRODUCTION

2nd order pileup is the result of coincidences between three pulses and produces an impulse response spectrum in the same way that 1st order pileup does. In this chapter the criteria for classifying 2nd order pileup in the Monte Carlo simulations are extended, with the programs being modified so that they can generate either 1st or 2nd order pileup or both simultaneously. The nature of the 2nd order impulse response is discussed in detail and the theory involved in determining the "forbidden region" and the coincidence rates within and between different spectral channels is developed in more detail than in the previous chapter.

The modification required to the Correction/Distortion algorithm to enable it to deal with either 1st or 2nd order pileup, or both, is described.

The relative distortion due to 1st and 2nd order pileup is examined and the Z values (the products $N(\tau_F + \tau_R)$) at which they are each maximized is derived from theoretical considerations.

6.1 2ND ORDER PULSE PILEUP FROM MONTE CARLO SIMULATIONS

It has been shown by *Datlowe*³⁰ that the probability density functions for pulse pileup involving more than two pulses, i.e. of order higher than one, can be obtained by repeated convolution of the 1st order density function with itself. The Monte Carlo simulation provided a convenient way to produce 2nd order pileup, however a further modification of the criteria discussed in Section 5.5 was necessary. The modification was designed to allow either 1st or 2nd order pileup, or both, as specified by keyboard entry, with counters keeping track of SINGLE, DOUBLE and TRIPLE events. The same basic approach was retained of determining the fate of pulse B and incrementing channels corresponding to a composite

maximum only if a legitimate single, double, or triple event involving B occurs.

An additional inter-arrival time τ_4 is required which is the separation between pulse D and the start of an additional pulse E. For each event a composite pulse is formed from the sum $B + C + D$. The first maximum encountered is stored together with its location time TMAX, measured from the start of τ_2 , i.e. from the start of pulse B as previously. The extended criteria are then applied in the order given below.

- * If the contribution of the tail of pulse A at the maximum of pulse B is greater than one channel of amplitude the event is classified as tail pileup and no channel is incremented.
- * ELSE
If the contribution of pulse C at TMAX is less than one channel, the event is classified as a SINGLE pulse and the contents of channel B are incremented.
- * ELSE
If the contribution of pulse D at TMAX is less than one channel, the event is classified as a DOUBLE pulse, and the channel corresponding to the composite sum maximum is incremented.
- * ELSE
If the contribution of pulse E at TMAX is less than one channel the event is classified as a TRIPLE pulse, and the channel corresponding to the composite sum maximum is incremented.

Additional logical conditions are included to provide the necessary capability for restricting the pileup additions according to the pileup order(s) specified by keyboard entry.

Fig. 6.1.1 shows the impulse response obtained for three pulses having fixed amplitudes corresponding to channel 125. The "forbidden region" is now greater than with 1st order pileup with the maximum of the continuum at channel 375, three times the fixed amplitude as would be expected.

6.2 ANALYSIS OF THE FORBIDDEN REGION AND 2ND ORDER IMPULSE RESPONSE

The shape of the 2nd order impulse response for semi-Gaussian pulses with a shape index of 5 as seen in Fig. 6.1.1 is entirely consistent with what would be expected from a convolution of the 1st order response with itself. For example if the simple uniform response of the triangular pulse coincidence were to be convolved with itself the result would be as shown in 6.2.1(a) which is a triangle (totally unrelated to the triangular shapes of the pulses causing the response). If however the pulses were asymmetric and the uniform response had a forbidden region where the value of the function was everywhere zero, then the convolution would again be an isoscoles triangle, but with a forbidden region covering twice the number of channels that the 1st order response had. This is shown in Fig. 6.2.1(b).

This raises the question of the parameters which govern the length of the forbidden region when asymmetric pulses are involved. For example what happens with participating pairs of pulses with different rather than equal amplitudes? Monte Carlo impulse response simulations were run separately for 1st and 2nd order pileup with the participating pulses having a variety of different amplitudes, but with the mean of the two, (and for 2nd order pileup, three) pulses being arranged to be the same in all cases at channel 125. As long as the largest pulse fell in a channel within the forbidden region, the impulse responses were identical. This led to the following fairly significant conclusions:

Figure 6.1.1: 2nd Order Impulse Response with 3 pulses of equal amplitude in channel 125. (cf Figure 5.6.1)

Figure 6.2.1 (a): Convolution (with itself) of a uniform 1st order impulse response with no forbidden region produces a triangular second order impulse response. (b) Same as (a) but with a forbidden region in the original 1st order response. (c) With unequal pulse amplitudes, the responses begin at the channel corresponding to the mean amplitude.

- * For the purposes of convolution the 1st order impulse response extends from the channel which is the mean of the amplitudes of the two participating pulses to the channel which is their sum.
- * The forbidden region thus extends from the mean of the amplitudes of the two participating pulses to an upper limit which is a function of the pulse shapes. As the amplitude of one of the pulses is increased from the mean, the forbidden region decreases, reaching zero when the pulse amplitude reaches this upper limit.
- * If the two pulses concerned are in channels I and M, and if the non-zero portion of the impulse response (the "permitted region") is a fraction α of the total response, which includes the forbidden region, then the number of channels covered by the non-zero portion will be given by:

$$\text{Range} \left| f(PU) \right| = \alpha \left((I + M) - (I + M) / 2 \right) = \alpha (I + M) / 2 \quad (6.1)$$

The forbidden region will thus cover the interval from channel M up to channel $(1 - \alpha/2) (I + M)$. If this is less than or equal to zero, then there is no forbidden region. This is shown in Figure 6.2.1(b).

- * When the matching function discussed in section 5.4.2 is incorporated as $f(PU)$, then the digitizing parameter, now denoted by L rather than I to avoid confusion, is calculated as

$$L = \alpha (I + M) / 2 \quad (6.2)$$

to allow for the forbidden region.

- * An exactly analogous argument is applicable to the 2nd order response forbidden region (which will be the same fraction $(1 - \alpha)$ of the total) where three pulses in channels I, J and M are involved, with M being the largest. The start of the response will be at

channel $(I + J + M) / 3$ and it will extend to their sum at channel $(I + J + M)$. Denoting the self convolution of $f(PU)$ as $f^*(PU)$, the 2nd order equivalent of equation (6.1) becomes:

$$\text{Range} \left[f^*(PU) \right] = a \left((I + J + M) - (I + J + M) / 3 \right) = 2a(I + J + M) / 3 \quad (6.3)$$

This is shown in Figure 6.2.1(c).

Fig. 6.2.2 shows the result of a numerical self-convolution of the 1st order impulse response obtained from a Monte Carlo simulation using a standard discrete convolution algorithm. On the same diagram is shown the 2nd order impulse response obtained by a numerical convolution of the optimum matching function used for the corrections of 1st order pileup discussed in Chapter 5. The agreement between these and the Monte Carlo simulation of Fig. 6.1.1 is far from perfect but good enough to indicate that the conclusions reached above represent a satisfactory working hypothesis.

6.3 COINCIDENCES WITHIN AND BETWEEN CHANNELS

6.3.1 SINGLE COINCIDENCES

One of the simplifying assumptions made in the discussion of coincidence spectra in Chapter 5.2 was that the coincidence rate for two pulse trains with count rates N_1 and N_2 would be given by:

$$\text{Coincidence Rate} = 2 N_1 N_2 \tau$$

where τ is the effective pulse width. This expression is based on an assumed uniform distribution of pulse inter-arrival times, rather than the exponential distribution which applies in this case.

Figure 6.2.2 (a): Self-convolution of the 1st order impulse response produced by a Monte Carlo simulation (b) self-convolution of the matching analytical function $f(\text{PU})$. A standard numerical convolution algorithm was used on the responses of Figure 5.6.1 to produce the 2nd order responses shown here.

Consider two channels A and B in a spectrum with individual count rates N_A and N_B , with a total spectrum count rate of N . The probability of a coincidence between pulses from these two channels is a compound event which includes the requirement for a pulse pair to be preceded and followed by intervals τ_F and τ_R , resulting in what can be termed a resolution factor

$$p(res) = e^{-N(\tau_F + \tau_R)}$$

The remainder of the event can be broken down into sub-events as follows:

- (a) Probability of a pulse A (or B) as opposed to B (or A) beginning an inter-arrival time.
- (b) Probability that the inter-arrival time is $< \tau_R$.
- (c) Probability that a pulse B (or A) as opposed to A (or B) ends the inter-arrival time.

The probabilities (a) and (c) above, can be determined on a relative frequency basis as:

$$p(A) = \frac{N_A}{N} \quad p(B) = \frac{N_B}{N}$$

the probability (b) above is given by

$$p(t < \tau_R) = (1 - e^{-N\tau_R})$$

The probability of a coincidence between A and B is thus given by:

$$p(\text{coinc } A * B) = p(res) \left\{ p(A) \cdot p(t < \tau_R) \cdot p(B) + p(B) \cdot p(t < \tau_R) \cdot p(A) \right\}$$

from which:

$$p(\text{coinc } A*B) = \frac{2 N_A N_B}{N^2} p(\text{res}) (1 - e^{-N \tau_R})$$

Now

$$\text{Coinc. Rate } A*B = N \cdot p(\text{coinc } A*B)$$

$$\text{Coinc. Rate } A*B = \frac{2 N_A N_B}{N} p(\text{res}) (1 - e^{-N \tau_R}) \quad (6.4)$$

6.3.2 MULTIPLE COINCIDENCES

For a single Poisson process with count rate N , the probability of n^{th} order pileup involving $(n + 1)$ pulses is given by:

$$p(PU^n) = p(\text{res}) (1 - e^{-N \tau_R})^n \quad (6.5)$$

(see Section 1.2.2, eq. (1.4)).

Suppose that we have k channels involved in cross-coincidences, i.e. pileup of order $(k-1)$. By an extension of the argument used in the previous section the probability of a multiple coincidence involving k different pulses $A, B, C, \dots$ separated by intervals $< \tau_R$ is given by

$$p(\text{coinc } A, B, C \dots K) = p(\text{res}) \cdot p(A, t_1 < \tau, B, t_2 < \tau, \dots, t_{k-1} < \tau, K) k!$$

The factor $k!$ represents the number of possible permutations of the order of the pulses in the sequence.

If the count rates in the k channels are $N_1, N_2, \dots, N_k$, then the probability of a pulse from any channel with count rate N_k , will be N_k/N , on a relative

frequency basis. Thus the total probability of a multiple coincidence is given by:

$$p(\text{coinc } A, B, C, \dots, K) = p(\text{res}) k! \frac{N_1 N_2 \dots N_k}{(N^k)} \left(1 - e^{-N/R}\right)^{k-1}$$

The coincidence rate is then given by

$$\text{Coinc. Rate } A*B*C \dots *K = p(\text{res}) k! \frac{N_1 N_2 \dots N_k}{(N^{k-1})} \left(1 - e^{-N/R}\right)^{k-1} \quad (6.6)$$

where the notation "A*B" is used here and in Section 6.4 to denote "Coincidences between pulses A and B".

6.4 MODIFICATION OF THE CORRECTION ALGORITHM TO INCLUDE 2ND ORDER PILEUP

The principle used in dealing with single coincidences involved the mirror channel M being fixed, while coincidences between it and all channels I below were computed. For each pair selected corrections were made to channels (M + J) in the range M to (M + I). This involved three nested DO LOOPS.

Extension of this principle to include multiple coincidences involves, as one might expect, the addition of more DO LOOPS in the nest. The principle for 2nd order pileup is to compute coincidence rates of pulses in channels J, I and M and for each triple combination, a fourth DO LOOP, with K running from K = M to (M + I + J), corrects the channels M to (M + I + J) for 2nd order pileup, using the 2nd order coincidence density function.

This arrangement preserves the concept that only channels that have themselves been corrected are used to compute the corrections to be

made to those above the mirror channel M. It also allows a selection to be made as to which order of pileup to include in the correction. If only 1st order correction is required, execution of the fourth loop is skipped. If only 2nd order correction is required, the loop involving J is executed to compute the triple coincidence rates, but the corrections to channels (M + J) for 1st order pileup are skipped.

The coincidence rates for channels I, J, M can be found from (6.6) with $k = 3$

$$\text{Coinc. Rate } I^*J^*M = 6 \frac{N_I N_J N_M}{(N^2)} e^{-N(\tau_F + \tau_R)} \left(1 - e^{-N\tau_R}\right)^2 \quad (6.7)$$

We must then consider the modification required to (6.7) when self coincidences occur for equalities as follows:

<u>Coincidence</u>	<u>Equality</u>
M*M*M	I = J = M
I*I*M	J = I
J*M*M	I = M

The DO LOOPS are so arranged that if $J = M$, then $I = M$, so that only the three coincidence possibilities listed above need be considered. The first one is covered by (6.5) with the expression for a single Poisson process. The other two require that (6.6) be modified for the following reason: the general expression (6.6) is for k different channels being involved in the coincidence. If r of the k channels are one and the same channel, then the factor $k!$ is replaced by the factor $k C r$, for combinations of r objects in a set of k . In this case $k = 3$ and $r = 2$ for both I^*I^*M and J^*M^*M coincidences, and we have coincidence rates for these two cases given by

$$\text{Coinc. Rate } I * I * M = 3 \frac{N_i^2 N_m}{(N^2)} e^{-N(\tau_F + \tau_R)} \left(1 - e^{-N\tau_R}\right)^2 \quad (6.8)$$

$$\text{Coinc. Rate } J * M * M = 3 \frac{N_j N_m^2}{(N^2)} e^{-N(\tau_F + \tau_R)} \left(1 - e^{-N\tau_R}\right)^2 \quad (6.9)$$

As was stated earlier, the second order impulse responses shown in Fig. 6.2.2 are the results of numerical convolutions of the 1st order impulse responses obtained from a Monte Carlo simulation and from the matching function described in Section 5.4.2. This function was also convolved by solving the standard convolution integral which resulted in a somewhat complicated expression. Another integration would have been necessary to produce a cumulative probability distribution from the convolution and this was not possible except by standard numerical techniques. The 2nd order impulse response from the Monte Carlo simulation was reasonably close to an empirical matching function consisting of two simple line segments and shown superimposed on it in Fig. 6.4.1. The negative slope of the second segment was made selectable by keyboard entry in terms of the fraction of the peak height at which the segment terminated.

This matching function was easily integrated to provide an analytical expression for the single channel probability slice covering the range indicated in equation (6.3) and the result was incorporated into the modified algorithm which follows (simplified for clarity), where it is denoted as PROB..K.

Figure 6.4.1: The 2nd order impulse response of Figure 6.1.1 with a piecewise segment empirical matching function used to match it in the discrete correction algorithm used for 2nd order pileup. (cf Figure 6.2.2)

DO M = 1, MAX..CHAN (6.10)

DO I = 1, M

COMPUTE COINC. RATE I*M

DO J = 1, I

COMPUTE COINC. RATE I*J*M

PROB..J = FUNCTION (F (PU I, J)) (1st order impuse response)

1ST ORDER CORRECTION = (COINC. RATE I*M) PROB..J

CHAN (M + J) = CHAN (M + J) - 1ST ORDER CORRECTION

DO K = 1, (I + J)

PROB..K = FUNCTION (F (PU I, J, K)) (2nd order impulse response)

2ND ORDER CORRECTION = (COINC. RATE I*J*M) PROB..K

CHAN (M + K) = CHAN (M + K) - 2ND ORDER CORRECTION

END DO

END DO

END DO

END DO

6.5 FRACTION OF PULSES MOVED OR LOST DUE TO NTH ORDER PILEUP

Pileup of order n involves the coincidence of $(n + 1)$ pulses, i.e. a fixed-number pulse string having $(n + 1)$ pulses with separations less than the system resolving time. Following Tenney, the occurrence rate of such a string was derived in Section 1.2.2 equation (1.4) for n pulses as:

$$\text{occurrence rate} = N e^{-2N\tau} (1 - e^{-N\tau})^{(n-1)}$$

As a result of the two-part resolving time concept involving τ_F and τ_R , and bearing in mind that n th order pileup involves $(n + 1)$ pulses, the above equation now becomes

$$PU..n = N e^{-N(\tau_F + \tau_R)} (1 - e^{-N\tau_R})^n \quad (6.11)$$

where $PU..n$ is the occurrence rate for coincidences involving $(n + 1)$ pulses which are effectively processed by an ADC as a single event.

A slight change in notation will be introduced now to make (6.11) more compact. For a given pulse shape let the ratio $\tau_F/\tau_R = \alpha$, so that $(\tau_F + \tau_R) = \tau_R(1 + \alpha)$. It is also convenient to write the product $N \cdot \tau_R = Z_R$. With these changes equation (6.11) then becomes:

$$PU..n = N e^{-(1+\alpha)Z_R} (1 - e^{-Z_R})^n \quad (6.12)$$

The probability of an n^{th} order pileup event is then:

$$p(n^{\text{th}} \text{ order event}) = e^{-(1+\alpha)Z_R} (1 - e^{-Z_R})^n \quad (6.13)$$

In order to find the value of Z_R which produces the highest probability, the derivative of (6.13) with respect to Z_R is equated to zero. The result is then solved for Z_R to give

$$Z_R = \ln \left(\frac{n + 1 + \alpha}{1 + \alpha} \right) \quad (6.14)$$

One can therefore determine for a given order n of pileup and a given value of τ_R , what count rate will maximize the fraction of the total number of pulses which are involved in coincidences of that particular order.

For example the pulses used in the Monte Carlo simulations had a shape index of 5. This resulted in a ratio $\alpha = \tau_F/\tau_R$ of approximately 2.3. When this value is inserted into (6.14) for 1st and 2nd order pileup ($n = 1$ and 2) we find

$$Z_R \text{ for max } \underline{1^{\text{st}}} \text{ order} = 0.264$$

$$Z_R \text{ for max } \underline{2^{\text{nd}}} \text{ order} = 0.474$$

The value of τ_R was typically 1.8 μ secs, so that the count rates which maximized 1st and 2nd order pileup were respectively:

147,000 c/s

263,000 c/s.

There are two aspects to the distortion caused by the pileup process, (a) that caused by pulses being removed from channels where they should be, and (b) that caused by pulses being added to channels where they should not be. For a given order n of pileup, $(n + 1)$ pulses are removed from channels where they should have been recorded and just one pulse is added to a channel higher in the spectrum, or possibly beyond the active region altogether.

In the case of 1st order pileup, half of the pulses removed are redistributed over a region covering twice the active spectrum. For 2nd order pileup, one third of the pulses removed are distributed over a region covering three times the active spectrum. The potential interference due to 2nd order pileup is thus two thirds of that due to 1st order by reason of the larger region covered, with an additional factor of two thirds due to the smaller number of pulses produced per coincidence. The interference is thus approximately $(2/3)^2 = 4/9$ that of 1st order pileup. Extension of the argument to higher orders suggests that the expression $(2/(n + 1))^2$ is the ratio of distortion produced by pileup of order n to that produced by 1st order pileup. This is a largely qualitative approach obviously and the actual level of distortion in a particular case will depend on the complexities of the spectrum involved.

Based on this reasoning it becomes clear that the higher the order of pileup, the less one need be concerned by distortion due to additions. The major distortion for the higher orders is therefore a linear reduction in the count rate across the entire spectrum, which is corrected for by applying the factor $e^{N(\tau_F + \tau_R)}$ as was explained earlier. 1st order pileup, with one pulse being added for every two removed, is thus the

predominant cause of peak distortion in most gamma and X-ray spectrometry applications, where the count rates would not normally involve Z_R values much beyond 0.5.

6.6 VALUES OF $\tau_F + \tau_R$ FROM 2ND ORDER PILEUP IMPULSE RESPONSES

The Monte Carlo simulation program was modified to compute values of $\tau_F + \tau_R$ from the 2nd order pileup only, when this mode is selected. Equation (5.14) still applies, however equation (6.11) must now be used with $n = 2$ for 2nd order pileup giving:

$$PU..2 = N e^{-N(\tau_F + \tau_R)} (1 - e^{-N\tau_R})^2 \quad (6.15)$$

from which

$$PU..2 = S (1 - e^{-N\tau_R})^2$$

$$\tau_R = (1/N) \ln \left(\frac{1}{1 - (PU..2/S)^{1/2}} \right) \quad (6.16)$$

Monte Carlo 2nd order impulse response simulations were run with the same parameters (count rates and distribution of pulse amplitudes) as was done for the 1st order impulse responses, the data for which are listed in Table 5.1. The addition of a third pulse was the only difference, with the mean amplitude again being centred at channel 125. Table 6.1 shows the 1st and 2nd order results together for comparison.

The following salient points arise from the data:

- * The total resolving time ($\tau_F + \tau_R$), denoted by τ_T is essentially constant and independent of count rate or pileup order.
- * The τ_F , τ_R values are essentially independent of pulse amplitude distribution for both orders.

- * The τ_F/τ_R ratios are consistently higher for the 2nd order than for the 1st order except at the highest count rate.
- * The ratio τ_F/τ_R decreases by about 15% with count rate for the 2nd order, there is less dependence on count rate of this ratio for the 1st order data.

Strictly speaking the value of τ_R should be independent of count rate and pileup order. In practice the effective value for a triple-pulse coincidence will be less than that for a double-pulse coincidence and the reason is as follows; the requirement for a single composite maximum from three pulses would not be met with two successive separations (inter-arrival times) of the maximum allowable value (τ_R) that holds for the single separation to produce a composite maximum with only two pulses. The constraint on the separations is tighter and as a result the effective value computed from the triple-event rate is less. The total resolving time τ_T , is computed from the single pulse rate, which as expected is a fixed fraction of the incoming rate and independent of it. τ_F is computed as $(\tau_T - \tau_R)$, so that its value merely reflects the differences in the value of τ_R as computed from the double or triple event rates.

Table 6.1

Comparison of τ_F and τ_R Values for Order 1 and Order 2

Count Rate (c/s)	Pulse Amplitude Configuration			τ_F	τ_R	τ_F/τ_R	τ_T	Order
150,000	125		125	4.109	1.830	2.245	5.939	1
	125	125	125	4.278	1.661	2.576	5.939	2
	100		150	4.106	1.834	2.239	5.940	1
	100	125	150	4.278	1.663	2.572	5.941	2
	70		180	4.093	1.843	2.221	5.936	1
	112	113	150	4.277	1.663	2.572	5.940	2
	30		220	4.101	1.811	2.264	5.911	1
	30	50	295	4.297	1.640	2.620	5.937	2
	125		125	4.043	1.910	2.116	5.954	1
	125	125	125	4.204	1.743	2.412	5.947	2
250,000	100		150	4.028	1.930	2.087	5.958	1
	100	125	150	4.212	1.743	2.417	5.954	2
	70		180	3.986	1.970	2.023	5.956	1
	112	113	150	4.204	1.747	2.406	5.951	2
	30		220	3.998	1.938	2.058	5.926	1
	30	50	295	4.234	1.704	2.485	5.938	2

Table 6.1 (continued from previous page)
Comparison of τ_F and τ_R Values for Order 1 and Order 2

Count Rate (c/s)	Pulse Amplitude Configuration			τ_F	τ_R	τ_F/τ_R	τ_T	Order
500,000	125		125	3.725	2.221	1.677	5.947	1
	125	125	125	4.000	1.971	2.029	5.971	2
	100		150	4.028	1.930	2.087	5.958	1
	100	125	150	4.003	1.973	2.029	5.976	2
	70		180	3.986	1.970	2.023	5.956	1
	112	113	150	4.031	1.973	2.043	6.005	2
	30		220	3.988	1.938	2.058	5.926	1
	30	50	295	4.013	1.960	2.047	5.973	2

6.7 EFFECT OF 2ND-ORDER PILEUP ON A FLAT SPECTRUM

Initial attempts to correct a flat spectrum with 2nd order pileup from a Monte Carlo simulation were notably unsuccessful. The pileup tail was matched with reasonable accuracy by the correction algorithm (modified for 2nd order pileup) operating in reverse, but the ramp portion showed a significant departure from the Monte Carlo simulation. The comparison is shown in Figure 6.7.1. The count rate for the simulation was 250,000 c/s giving a value for Z_R (the $N\tau_R$ product) of 0.4.

It appeared that the pileup gain in the last channel (33) of the active flat spectrum exceeded that of the first channel in the tail (34) by a factor of almost two. There was no obvious reason for this discontinuity since pileup gains should be essentially continuous functions, regardless of abrupt changes in channel count rates of incoming pulses.

Figure 6.7.1: *2nd order pileup in a flat spectrum generated by a Monte Carlo simulation. The matching distortion produced by applying the correction algorithm in reverse, does not track the simulation in the active portion of the spectrum.*

The simulation program was modified to generate four separate spectra simultaneously: a composite pileup spectrum (1st or 2nd order or both); a second spectrum with only single pulses which survived pileup, a third spectrum with only pulses classed as "Doubles", and a fourth spectrum with only pulses classified as "Triples". The last two are effectively pileup-gain spectra for 1st and 2nd order pileup respectively.

A re-run of the simulation of Figure 6.7.1 produced these four spectra, the composite one being for 2nd order pileup only, they are shown superimposed in Figure 6.7.2. The continuous line spectrum is the composite 2nd order pileup one, with the other three being shown respectively by asterisks, (singles paralleling the ramp portion); circles, (1st order pileup gains); and " + " symbols, (2nd order pileup gains, merging with the composite plot).

From this the source of the discrepancy is readily apparent, the single pulses which survived pileup are not uniformly distributed with amplitude.

6.7.2 NON-UNIFORM DISTRIBUTION OF SINGLE PULSES SURVIVING COINCIDENCES

The reason for this somewhat surprising result is not difficult to see when coincidences between pulses of unequal amplitude are considered. Figure 6.7.3 shows what happens when pairs of triangular pulses of unequal amplitude are at the critical separation of one-half pulse width. A separation any greater than this precludes a single composite maximum from the two pulses. As long as the amplitude of the second pulse is less than that of the first, then the first one will be a single. For all greater amplitudes the first pulse is absorbed into a coincidence with the second one. It is clear therefore that in an encounter with that particular time separation, the larger pulse will always survive as a single with a probability equal to the number of channels below it divided by the total number of channels. This reasoning leads to the conclusion that there

Figure 6.7.2: A re-run of the simulation of Figure 6.7.1 with accumulation of single pulses (a), 1st order gains (b), and 2nd order gains (c), in separate spectra. The single pulses are clearly not uniformly distributed.

Figure 6.7.3: Pulse-pairs of different amplitude ratios with the critical separation of one-half a pulse width. In each case the smaller pulse would be ignored by an ADC and the larger one would survive.

should be a linear ramp in the distribution of surviving single pulses. The largest pulse therefore has a probability of unity of surviving, but only if the separation is exactly τ_R .

Additional Monte Carlo simulations over a range of count rates showed that the slopes of the ramps on the singles spectra were count rate dependent, or rather dependent on the value of Z_R . The reason for this dependency can be understood by considering one of the criteria adopted for a pulse to be a single; this was that the leading edge of the second pulse should contribute less than some predetermined fixed amplitude to the composite maximum. As the ratio of the amplitude of the second pulse to the first increases, so does the minimum allowable separation (actually τ_R) which is required to meet that condition.

Suppose that for two equal amplitude pulses the time is τ_1 and that for two pulses of unequal amplitude the time is τ_2 . Their respective survival probabilities as singles are therefore $e^{-N\tau_1}$ and $e^{-N\tau_2}$. The ratio of their survival probabilities is therefore $e^{-N\tau_1}/e^{-N\tau_2} = e^{-N(\tau_1-\tau_2)} = e^{N(\tau_2-\tau_1)}$. This ratio increases with increasing difference between τ_2 and τ_1 , i.e. with increasing pulse amplitude ratio, which supports the earlier reasoning. The main point however is that it is clearly count rate dependent, with the slope of the non-uniformity in the singles distribution increasing with increasing count rate.

The unequal pulse amplitude ratio argument has a corollary. If the order of the two pulses is reversed, then the minimum allowable separation to meet the original condition is now much less than τ_1 , i.e. a small pulse can follow a large one more closely without adding more than the chosen threshold amplitude level to the composite maximum.

This is why the value of τ_R is essentially independent of the amplitudes of participating pulses, provided that their mean amplitudes are constant.

It is interesting to note that initially in this work with Monte Carlo simulations, the criterion adopted for a single pulse was simply that the separation from the following one be $\geq \tau_R$. The problem of the non-uniform distribution of the surviving singles did not then arise, because it would only have occurred for separations that were exactly τ_R , and this has a vanishingly low probability of occurrence. The criterion was changed to the requirement that the contribution to the composite maximum be less than a chosen threshold level because it seemed that this would be a more realistic emulation of the way real-world ADC's process pulses.

The data obtained from the Monte Carlo simulations concerning the non-uniform singles distribution is given below as a tabulation of the slope of the ramp (assumed linear) as the percentage increase in c/s per channel with increasing count rate. The slopes were obtained from regression analyses of the 33 data points in each singles spectrum. The correlation coefficients are also given.

Count Rate (Kc/s)	Singles Ramp Slope (%/Chan)	Correlation Coeff
150	0.125	0.60
250	0.300	0.80
500	0.686	0.75
1000	1.30	0.71

The relationship is broadly linear and may well be strictly linear, the correlation coefficient values indicate that the values obtained for the slopes are probably subject to considerable error.

6.7.3 CORRECTION OF A FLAT SPECTRUM WITH 2ND ORDER PILEUP

A modification was incorporated into the algorithm based on these data which effectively corrected for the problem of the non-uniform

MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS
STANDARD REFERENCE MATERIAL 1910a
(ANSI and ISO TEST CHART No. 2)

distribution of singles. The modification involves a secondary correction to the active portion of the spectrum only, following completion of the primary correction. It operates in such a way that the count-rate in the centre channel of the active spectrum is left unaltered, while the counts in those below and above are augmented and reduced respectively. The implementation is by means of a DO LOOP of the form:

```
DO  I = 1, ACT..CHAN
      J = ACT..CHAN/2 - I + 1
      SNGL..CORRN = J*AMPL..FACT
      IF (SPECTRUM (I).GT.O)
#    SPECTRUM (I) = SPECTRUM (I)*(SNGL..CORRN + 1)
END DO
```

The "AMPL..FACT" is an empirical function of the count rate based on the data in the tabulation above. The correction is applied with a linear weighting across the active portion of the spectrum. The maximum positive value occurs at $I = 1$, and the maximum negative value at $I = \text{ACT..CHAN}$. The correction is zero at $I = \text{ACT..CHAN}/2$ (or the nearest integer).

Figure 6.7.4 shows the result of incorporating this correction. The distortion produced by operating the algorithm in reverse on a flat spectrum (identical numbers in all channels) matches that produced by the Monte Carlo simulations over both the ramp portion, and the tail above the active spectrum. The correction produces a flat spectrum, within the limits of the statistical noise in the original. There is a residual error in the tail and the sources are (a) the statistical noise for which the algorithm is not responsible; (b) whatever mis-match there may be between the real 2nd order impulse response and the piece-wise triangular function used to simulate it; and (c) the linear factor (almost three in this case) by which the entire spectrum is multiplied to compensate for losses during the correction process. In relative terms the residual error remaining in the tail is the same percentage of the

Figure 6.7.4: A modification to the algorithm to correct for the non-uniform distribution of single pulses produces a good match of the distortion (a) and a satisfactory correction (b) to remove it and restore a flat spectrum.

amplitude of the corrected flat spectrum, as the difference between true and simulated distortion on the tail pileup is of the distorted spectrum amplitude.

This test indicates that the theory which has been developed both to simulate and to correct for 2nd order pileup is basically sound. With the level of computing power used, which was a Data General Corp MV4000 system, the 2nd order correction would require a prodigious amount of CPU time if many hundreds of channels were involved.

The experience of the last few years however indicates that computing power increases virtually exponentially with time, so that the CPU time presently required will be drastically reduced before very long.

CHAPTER 7 CORRECTION OF SPECTRA ACQUIRED WITH A MULTICHANNEL ANALYSIS SYSTEM

7.0 INTRODUCTION

The experiments described in this chapter were conducted with a computer-based multichannel analysis system designed by the author which consists of the following components.

- * A Data General Corp. NOVA-4 Mini-computer.
- * A Nuclear Data Inc. High-speed successive-approximation pulse height ADC Model ND 582. This unit has a pulse processing time of approximately $3.0\mu\text{s}$, which is less than the peak-to-base-line times of any pulse shapes used in these experiments.
- * A cylindrical sodium iodide scintillation detector, 76 mm diameter by 76 mm thickness, connected to a D.C.-coupled preamplifier and pulse shaping main amplifier designed to minimize the negative-going base-line excursions known as "pulse undershoot".

The object was to test the validity of the simulations and theoretical work described in Chapters 5 and 6. Two main experiments were carried out and the results are discussed in detail. The first one involved the design and evaluation of a hardware flat spectrum generator which produced only 1st order pileup, to see if the real-world result matched the simple ramp pileup produced by the theory and simulation. The second experiment was the acquisition and correction of a high count rate radiation spectrum with the scintillation detector and a radioactive source.

7.1 FLAT SPECTRUM GENERATOR, CONCEPT AND DESIGN

The requirements for a real-world flat spectrum generator which would match the properties of the Monte Carlo simulation were as follows:

- * Pileup must be limited to 1st order.
- * Pulse inter-arrival times must have an exponential distribution.
- * Pulse amplitudes must be confined to a fixed range.
- * Pulse amplitudes must have a uniform probability density function.
- * There should be (as nearly as possible) a uniform probability that the amplitudes of any two consecutive pulses are anywhere in the entire range of amplitudes.

The simplest design concept which met these requirements is shown in the block diagram of Figure 7.1.1. The scintillation detector is used in conjunction with a radioactive source to generate pulse inter-arrival times at a rate determined by the source-detector geometry. A simple Schmitt-trigger and mono-vibrator produced logic-level pulses of a fixed (1.5 μ s) duration from the charge signals produced by the detector. This scheme ensured that the logic pulses had the appropriate distribution.

The logic pulses were then fed to a divide-by-two circuit which steered alternate pulses to two separate and identical sampling circuits. Each one consisted of an 8-bit D/A converter, with a settling time of less than 200 ns, driven by an 8-bit counter. The counters were fed from separate (and asynchronous) clocks with rates of approximately 30 MHz. The entire range of each D/A converter was thus covered, saw-tooth fashion, in a time of approximately 8 μ s. The outputs of the two D/A's were held disabled (i.e. at zero or baseline level) until they were sampled by the logic pulses from the scintillation detector. A second and third mono-

FLAT SPECTRUM GENERATOR WITH PILEUP

Figure 7.1.1: Block diagram of Hardware Flat-Spectrum Generator. A scintillation counter provides logic pulses which are steered alternately to a pair of D/A converters. 1st order pileup occurs between the channels, with random amplitudes in the 256 increment range of the D/A converters.

vibrator in each circuit, with durations of approximately 20 and 67 μs , were triggered on receipt of a logic pulse which in turn enabled the corresponding D/A output (and disabled its clock), producing a rectangular pulse of duration 20 μs and amplitude somewhere along the saw-tooth, according to the state of the counter at the instant of sampling.

The result was two sequences of inter-leaving pulses of essentially uniformly random amplitudes within the 8-bit (256 channel) range of the two D/A converters. These were summed and shaped in a subsequent analogue amplifier stage to produce the pulse shape shown in the oscillogram of Fig. 7.2.1(a). This was taken using a digitizing oscilloscope with a 10 MHz sampling rate. It has a rise time of approximately 22 μs and a total time of about 60 μs . The shape index is very close to the value of 5 used in the simulations. The purpose of the third 67 μs period monovibrator was to disable the sampling circuitry until the pulse in progress in a given channel had decayed down to the baseline, thereby ensuring that pileup did not occur within a channel. The only pileup which could occur was therefore between pulses from different channels which were themselves guaranteed to be single (undistorted) pulses. This scheme thus precluded any pileup of order higher than one. However it effectively converts all higher orders of pileup down to order one. Consider the situation that occurs when a multiple pulse string arrives containing n pulses with separations less than τ_R . The first one will be fed to one of the channels and the second one to the other channel. A 1st order coincidence will be generated and the remainder of the pulses in the string will be ignored, so that what should have been an n -fold coincidence is effectively converted to a single coincidence.

(a)

(b)

Figure 7.2.1 (a): The pulse shape generated by the Hardware Flat-Spectrum Generator recorded by a digitizing oscilloscope. **(b)** A one millisecond sequence of the output pulse train at approximately 15,000 c/s.

7.2 EXPERIMENTAL DATA FROM THE HARDWARE FLAT SPECTRUM GENERATOR

Impulse responses were obtained by setting all the eight bits of the D/A's to a logic "one" after disconnecting the counters, thereby producing pulses of fixed full-scale amplitude. A typical impulse response is shown in Fig. 7.2.2(a). Values of τ_F and τ_R computed from this agreed closely with the observed pulse shape parameters. Suitable values for the parameters k and c in the analytical matching function for $f(\text{PU})$ were found which produced the plot of Figure 7.2.2(b). This particular impulse response was recorded at an input count rate of 30,000 c/s, as determined by a measurement of the logic pulse rate from the 1.5 μ s period monovibrator following the Schmitt-trigger circuit.

The circuitry was then connected to produce a flat spectrum and data were acquired at true input count rates ranging from 5,000 c/s to 30,000 c/s, examples are shown in Figure 7.2.3. It is apparent that the shape of the piled-up spectra are remarkably similar to those produced by the Monte Carlo simulations. The relatively high noise levels evident on the ramp portions in each case are due to in part to quantizing errors and aliasing between the D/A converters which produce the pulse amplitudes and the ADC which re-digitizes them into a range of approximately 128 channels. The true statistical noise level is that on the pileup tails in each case.

Figure 7.2.1(b) is an oscilloscope record capturing a single millisecond of the output pulse train. The height for single pulses is about 3.7 divisions. A hardware pulse-pileup detector based on pulse width would have accepted at least two near-perfect coincidences in this sequence.

Because of the conversion of all orders of pileup to order one, a modification of the theory is required in order to use the impulse response data to compute the values of τ_F and τ_R . From the multiple pulse string theory outlined in Chapter 1, the entire pulse train can be

Figure 7.2.2 (a): Typical Impulse Response from coincidences between pulses of fixed amplitude in the Hardware Flat-Spectrum Generator. **(b)** The matching $f(\text{PU})$ function generated by the correction algorithm with k and c values selected for an optimum fit.

Figure 7.2.3: The pileup produced by the Hardware Flat-Spectrum Generator at count rates of (a) 5000 c/s; (b) 15,000 c/s; and (c) 30,000 c/s.

classified in terms of single pulses and an infinite number of orders of pileup:

$$\text{Total Count Rate} = N e^{-2N\tau} + \sum_{n=1}^{\infty} N e^{-2N\tau} (1 - e^{-N\tau})^n$$

where pulses in a string are treated as a multiple coincidence resulting in one pulse. With the τ_F, τ_R concept this becomes:

$$\text{Total Count Rate} = N e^{-N(\tau_F + \tau_R)} + \sum_{n=1}^{\infty} N e^{-N(\tau_F + \tau_R)} (1 - e^{-N\tau_R})^n$$

Summing the geometric progression to infinity gives:

$$\sum_{n=1}^{\infty} N e^{-N(\tau_F + \tau_R)} (1 - e^{-N\tau_R})^n = \frac{a}{1 - r}$$

where a is the first term and r is the common ratio

$$a = N e^{-N(\tau_F + \tau_R)} (1 - e^{-N\tau_R}) ; r = (1 - e^{-N\tau_R})$$

Hence

$$\sum_{n=1}^{\infty} \text{Pileup} = N e^{-N\tau_F} (1 - e^{-N\tau_R})$$

The total coincidence rate for pileup with the flat spectrum generator will therefore be given by:

$$\text{Total Coincidence Rate} = N e^{-N\tau_F} (1 - e^{-N\tau_R})$$

If there are M channels in the active spectrum, then the count rate in each one is N/M . The rate for cross-channel coincidences will be $2(N/M)^2 e^{-N\tau_F} (1 - e^{-N\tau_R})$ and that for self coincidences will be one half of this. Each

channel is involved in $M-1$ cross-coincidences for each self coincidence, so that for a 128 channel spectrum the rate is effectively that for cross-coincidences.

From equation (5.10) the slope of the regression line Δ is equal to the channel coincidence rates times $\ln(2)$

$$\Delta = 2 (N/128)^2 e^{-N\tau_F} (1 - e^{-N\tau_R}) \ln(2)$$

$$\Delta = 2 (N/128)^2 (e^{-N\tau_F} - e^{-N(\tau_F + \tau_R)}) \ln(2)$$

from which:

$$\tau_F = \frac{1}{N} \ln \left(\frac{2 (N/128)^2 \ln(2)}{\Delta + 2 (N/128)^2 e^{-N(\tau_F + \tau_R)} \ln(2)} \right) \quad (7.1)$$

The pulses in the lowest channel will be single, i.e. not subject to pileup, hence the count rate in the lowest channel which will be denoted as MIN, will be a fraction of the true count rate in that channel ($N/128$) given by

$$MIN = (N/128) e^{-N(\tau_F + \tau_R)}$$

from which:

$$e^{-N(\tau_F + \tau_R)} = \left(\frac{128 \cdot MIN}{N} \right) = e^{-N\tau_T} \quad (7.2)$$

$$(\tau_F + \tau_R) = (1/N) \ln \left(\frac{N}{128 \cdot MIN} \right) = \tau_T \quad (7.3)$$

We now have all we need to evaluate τ_F and τ_R from the slopes of the ramp segments and thereby check if the results are close enough to reality to confirm that the theory behind equation (5.10) is valid in the

real-world of hardware, as well as in the abstract one of Monte Carlo simulations.

The slopes were determined by regression line analyses of the ramp segments in the spectral data. These analyses also provided the MIN values required in equation (7.2). The values of $e^{-N(\tau_F + \tau_R)}$ were then computed along with the values of τ_T for each of the spectra acquired. They were then used to evaluate τ_F by substitution in (7.1) of the $e^{-N\tau_T}$ values and the values of the slope Δ . The values of τ_R were then found as $\tau_T - \tau_F$.

Table 7.1 shows these data, from which it is seen that the values of τ_R are indeed acceptably close to that observed in the oscilloscope record shown in Figure 7.2.1(a). The values of τ_F show considerably more variation but nevertheless the results reflect the fact that an effective τ_T of $67\mu\text{s}$ was imposed on all the pulses by the monovibrator which locked out any further triggers for that time after receiving a valid trigger.

The recorded count rates shown in Table 7.1 were all corrected for the $1.5\mu\text{s}$ dead time imposed by the mono-vibrator before being used as values for N , the true input count rate.

Figure 7.2.4 shows the result of applying the correction algorithm to the piled-up flat spectra for count rates of 15,000 and 30,000 c/s. It also shows the result of applying the algorithm in reverse to produce a piled-up spectrum from a set of identical channels each containing the true input count rate divided by 128. The correction has essentially removed the pileup tail and restored a flat plateau in each case. The noise level has of course been subject to amplification by the computed loss factor $e^{2N(\tau_F + \tau_R)}$ which is applied to all channels before computing pileup gains.

A correction for the non-uniform distribution of surviving single pulses was found to be necessary, the magnitude was consistent with the

Figure 7.2.4: The correction of spectra produced by the Hardware Flat-Spectrum together with the matching distortion produced by the correction algorithm operating in reverse. The Count Rates are (a) 30,000 c/s (b) 15,000 c/s.

Table 7.1

Values of τ_F and τ_R (Microsecs) Computed from the Flat Spectrum Pileup Ramp Slopes

<u>Count Rate</u> <u>(c/s)</u>	<u>Min (c/s)</u>	<u>Δ (c/s)</u>	<u>τ_F</u>	<u>τ_R</u>
10,000	38.00	0.097	55.6	22.8
15,000	41.30	0.162	51.4	21.7
20,000	41.14	0.222	47.8	21.1
25,000	39.50	0.274	45.3	20.8
30,000	37.30	0.319	42.5	20.14

predictions of the Monte Carlo simulations as discussed in the previous chapter.

The conclusion from these experiments is that real-world ADC's and pulses are indeed accurately modelled and represented by the theoretical derivations and Monte Carlo simulation criteria which were developed in arriving at a viable spectral correction algorithm.

7.2.2 COINCIDENCES BELOW THE LOWER LEVEL DISCRIMINATOR (LLD)

The possibility of coincidences below the lowest channel, as set by the LLD, appearing in the lowest channel, and thereby adding to the contents of what is claimed to be the only channel with single pulses, was mentioned in Chapter 5.3.1. There is unfortunately no question that this can happen. To take an extreme case consider the situation in the flat spectrum with pileup shown in Figure 5.6.2. If the LLD were set at a point halfway up the ramp segment, then the lowest channel would be the one

above the LLD threshold and it would indeed receive the same pileup gains that it did with the LLD set close to zero.

It is usually possible to record the count rate of pulses which are below the LLD threshold via an output on the ADC. If the contents of the (empty) channels below the LLD threshold are replaced by the mean count rate as recorded from the ADC, (or by a separate Single Channel Analyser adjusted to cover the range of pulse amplitudes below the LLD threshold), then the correction can still be made with minimal error. The mean count rate below the LLD is divided by the number of channels in this region before each channel is initialized with this value.

The DO LOOP for the mirror channel M in the algorithm is then set to the first channel above the LLD threshold less one. The pre-initialized channel contents are then used to correct channels M and above. For the first few channels there is an error which quickly diminishes as the correction involves more and more of the channels with real data.

Figure 7.2.5 shows the result of correcting a flat spectrum with the same count rate and τ_F , τ_R parameters as the one of Figure 5.6.2, but with the lowest 40 channels of data deleted. The 40 channels were replaced with the average value of the deleted data and the correction algorithm was applied with the correction beginning at channel #41, using coincidence rates computed from the data inserted into the first 40 channels. The error in the correction is so small as to be indiscernable on the plot of Figure 7.2.5. The elevated count below channel 41 is of course the result of inserting data where there were null values previously and is not a part of the spectrum being corrected. The proper values for all channels above channel #40 (the fictitious LLD setting) was 1200 c/s. The correction gave a value of 1189.7 in channel #41, an error of -10 c/s. The error decreased pseudo-exponentially to -6 c/s at channel #125, the last channel in the active spectrum. The error in the correction to the pileup tail above channel #125 is just visible in the figure and goes from -5 c/s at channel

Figure 7.2.5: Error introduced with LLD threshold set at channel 40. The correction must start at channel 41 which contains pileup gains. The error is negligible if channels 1 to 40 are initialized with the mean channel count rate in that region.

#126 to a positive maximum of 1.6 c/s at channel #167 decreasing thereafter to zero.

It would appear from this test that it would be feasible to use the technique of initializing channels below the LLD threshold with mean values obtained experimentally from the Single Channel Analyser reject output in the ADC if that is accessible. Under normal circumstances it would be unlikely that the LLD would be set to a level high enough that the error due to coincidences below it would be significant enough to require a correction.

7.3 ACQUISITION AND CORRECTION OF A REAL RADIATION SPECTRUM

It was indicated in the overview preceding Chapter 5, that evidence of a satisfactory correction on a real radiation spectrum acquired at high count rate requires a low count rate reference spectrum.

It is well known that changing the source-detector geometry in order to vary the count rate inevitably produces changes in the spectrum shape due to back-scatter and other geometry-dependent effects. The ideal way to accomplish the required result would have been to activate a sample of manganese, or other element with a half life of a few hours, and acquire spectra at intervals as the count rate dropped from a level high enough to produce significant pileup, to one suitable for a reference spectrum undistorted by pileup. The necessary facilities were not however available for such an experiment.

An alternative approach was to design a modification to the pulse shaping circuitry that would enable the pulses to be stretched in time while still maintaining a constant shape index. This would allow a fixed source-detector geometry to be selected with pileup occurring with the long pulse rise-times, and virtually none with the short rise-times.

A modification was designed to enable the pulses to be essentially scaled up in time by a factor of ten at the push of a switch. This would thus change the $N\tau$ product by a factor of ten, which would be sufficient to produce significantly distorted, and then undistorted spectra without changing the source-detector geometry. Experiments indicated however, that the longer non-optimum shaping times introduced so much additional noise, that the system resolution (FWHM) was seriously degraded. The result was that in solving one problem, (the geometry-dependent effects on the spectral shape), another had now been created; the degradation of spectral shape due to noise.

The only remaining choice with the resources available was to change the source-detector geometry and use a source with a single energy peak which would only be subject to geometry dependent effects on the low energy side. This was done with the radioisotope ^{137}Cs as the source.

7.3.1 EXPERIMENTAL MEASUREMENTS OF SYSTEM VALUES τ_F AND τ_R

The optimum shaping incorporated into the main amplifier in the system used gave a pulse rise time of approximately $2.5\mu\text{s}$, with a shape index very close to 5. Figure 7.3.1 shows a comparison of the actual pulse shape with that given by equation (5.1) scaled to the same rise time and amplitude. The digital-oscilloscope record was obtained by attaching the preamplifier to a photo-multiplier tube (P.M.T.) mounted in a chamber containing two LED strings, each driven by independent pulse generators. These were standard laboratory units with periodic rather than random outputs. If both were run simultaneously (but asynchronously), then a 1st order impulse response was obtained. By suitable adjustment of the driver circuits the "scintillations" generated by the LED strings could be equalized in amplitude and duration to those that the P.M.T. normally receives from a real NaI scintillation crystal. By using only one generator, periodic pulses are generated at the main amplifier output, suitable for viewing on an oscilloscope.

Figure 7.3.1 (a): Pulse shape from laboratory MCA system recorded with a digitizing oscilloscope. **(b)** Semi-gaussian pulse with a shape index of 5 scaled to same time and amplitude frames.

Figure 7.3.2 shows the 1st order impulse response obtained with this arrangement. The total input count rate from the two generators was 20,000 c/s, they were adjusted to have approximately equal pulse repetition rates of 10 kHz each. Superimposed on the plot is the response obtained by the discrete correction algorithm operating in reverse on a single channel with 20,000 counts in it, and with matching $f(PU)$ function constants selected for an optimum fit.

Computation of the values τ_T , τ_F and τ_R from the data recorded is different from that used in previous calculations because in this case the inter-arrival times have a uniform, as opposed to an exponential distribution, thanks to the use of two standard periodic pulse generators. The coincidence rate for pulses which are processed as "DOUBLES" in this case will be given by:

$$\text{Coincidence Rate} = \sum PU = 2N_1 N_2 \tau_R$$

where N_1 , N_2 are the pulse repetition rates of the two pulse generators, and $\sum PU$ is the area under the impulse response of Fig. 7.3.2, with the vertical axis marked in c/s. Hence τ_R is evaluated as:

$$\tau_R = \frac{\sum PU}{2N_1 N_2} \quad (7.4)$$

The value of τ_F can be found indirectly from the number of pulses which were "SINGLES", i.e. processed, but not involved in coincidences. This is easily obtainable from the spectrometric data which were recorded. The entire sample space of N_1 and N_2 pulses is accounted for in three categories:

- Pulses recorded in the impulse response (coincidences)
- Singles
- Pulses which were not processed.

Figure 7.3.2: Impulse Response from laboratory MCA system with optimum matching $f(PU)$ function (i.e. with constants selected for an optimum fit).

The last category consists of pulses which were affected by the tail of the preceding one. The probability of this event in a sample time of one second is τ_F and the rate of occurrence for such non-processed pulses will be $2N_1 N_2 \tau_F$. This rate is found as $N_1 + N_2 - S - \sum PU$ where S is the singles count rate, thus:

$$\tau_F = \frac{N_1 + N_2 - S - \sum PU}{2N_1 N_2} \quad (7.5)$$

The addition of (7.4) and (7.5) gives:

$$\tau_F + \tau_R = \frac{N_1 + N_2 - S}{2N_1 N_2} = \tau_T$$

which is consistent with the idea that the total count rate of all pulses which are not singles should be $2N_1 N_2 \tau_T$.

The value obtained for τ_R was $2.8\mu s$, in close agreement with the digital oscilloscope record in Figure 7.3.1. The value of τ_F from (7.5) was $7.6\mu s$, suggesting a measurement error. Equation (7.5) involves a small difference of relatively large numbers and with the data being used it turned out that a 1% error in just one of the counting rates changed the value of τ_F by $1.0\mu s$. Equation (7.4) is not subject to this sort of devastating error source. Accordingly it was decided to use the oscilloscope record as the basis for setting τ_T at $8.0\mu s$, giving a value for τ_F of $5.2\mu s$.

One might well ask why it is necessary to conduct the experiment in the first place rather than use values for both τ_F and τ_R taken from the oscilloscope record. The reason is that the value of τ_R particularly is a function of the sample-and-hold circuitry in the ADC being used, and will depend on how much hysteresis there is (if any) in the peak detection circuitry. It is therefore better to let the ADC speak for itself via this

experimental procedure and announce what it has determined the values to be.

7.3.2 APPLICATION OF T_F T_R DATA TO CORRECTION OF A ^{137}Cs SPECTRUM

Figure 7.3.3 shows three superimposed spectra. One is recorded at a total count rate of 97,000 c/s, a second one is recorded with the source moved away from the detector at a count rate of 6800 c/s. The third one is the correction for both 1st and 2nd order pileup applied to the high count rate spectrum. They have been normalized to a common peak height rather than to a common area because the back-scatter peak in the reference spectrum would distort the comparison on an areal basis, although in fact the areas are probably equal to within ten percent or so. The back-scatter peak is due to the fact that when the source-detector distance is relatively large, the ratio of radiation which is scattered into the detector from surrounding objects to that which is received directly, is very much greater than it is when the source is virtually in contact with the face of the detector, as it was in this case for the high count rate measurement. This is exactly the sort of geometry-dependent effect on the spectral shape which made comparisons of low and high count rate spectra impossible to interpret.

It can be seen that the correction is effective in reducing the pileup tail almost to the level observed in the low count rate spectrum. The cross-over point where gains equal losses seems to be at about channel 55 (approximately 340 keV). The difference between the tail remaining after correction and the tail of the low count rate spectrum is almost certainly due to orders of pileup greater than 2.

Figure 7.3.3: Correction of a ^{137}Cs spectrum recorded at 97,000 c/s for 1st and 2nd order pileup. (a) reference spectrum recorded at 6,800 c/s (shown dotted), (b) spectrum recorded at 97,000 c/s, (c) spectrum corrected for 1st and 2nd order pileup.

CHAPTER 8 SUMMARY, CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

8.0 SUMMARY OF CONTRIBUTIONS ARISING FROM THIS THESIS.

The work in Part One of the thesis defines a dual classification for coincidences in Poisson-distributed pulse trains. It covers the situation where coincidences contain a fixed-number of pulses, which has been analysed by *Tenney*⁵⁸, and introduces an alternative concept involving coincidences occurring within fixed time intervals which are integer multiples of the coincidence time τ . The theory leading to expressions for the occurrence rates of these fixed-interval multiple-pulse strings is developed in Chapter 1, and has been shown to be consistent with the fixed-number pulse string theory developed by *Tenney*⁵⁸.

The concepts developed in Chapter 2 have demonstrated that Markov process theory can be applied in a simple, direct and very practical way to solve real-world problems involved in the design of instrumental techniques and apparatus to measure random events connected with radiation, and in the analysis of such measurements.

These techniques were applied in the measurement of multiple-pulse string occurrence rates. The close agreement between the observed and predicted data support the assumption that events associated with radioactive decay are Poisson-distributed. It also provides validation for the model theory developed in Chapter 2.

The work in Part Two of this thesis is concerned directly with the effect of coincidences on pulse-height spectra. An analysis of the coincidence probability density function for two and three pulse coincidences shows that a "forbidden region" exists for asymmetric pulses and that it will extend down to a point in the spectrum which is the mean of the amplitudes of the participating pulses, thus defining the origin of the coincidence probability density function.

A theoretical analysis of the processes involved in pulse pileup has led to the development of a discrete correction algorithm, based on the fact that the lowest channel in the spectrum loses pulses due to pulse pileup but does not gain any. Once the entire spectrum has been corrected for losses due to pileup, then correction can proceed for gains, beginning with the lowest channel-plus-one. This concept allows correction of a distorted spectrum without any prior knowledge or estimate concerning the un-distorted spectrum. It does however require that the coincidence probability density function be known for the shapes of the pulses involved in coincidences.

A relatively simple empirical function has been synthesised to simulate the coincidence probability density functions (referred to as "Impulse Responses" in this thesis) of a wide range of pulse shapes. This empirical function has been integrated to produce an expression for the probability "slice" for a single channel in terms of parameters which are ideally suited for use in the discrete correction algorithm.

A continuous-function equivalent of the discrete correction algorithm has been formulated as a double integral. Solution of this for a flat spectrum, (the simplest non-trivial spectrum which can be devised), showed that pileup distortion should produce a linear ramp in the active portion, followed by a pileup tail conforming to a specific analytic function. It also showed that the slope of the linear ramp could be predicted from a knowledge of the effective rise and fall times of the pulse shapes involved.

Monte Carlo simulations confirmed the prediction of the continuous-function solution for pileup in a flat spectrum. They also allowed extensive testing of the correction algorithm under conditions which cannot be realised with laboratory hardware. The software developed during this phase of the study allows impulse responses and flat spectra to be generated, as well as conventional spectra with predetermined

Gaussian peaks. In all cases the output includes the undistorted spectrum together with separate spectra of 1st and 2nd order pileup gains. All are produced simultaneously as the simulation proceeds. Use of this capability revealed the phenomenon that the distribution of pulses which survive pileup is weighted in favour of larger pulses. The effect is small but readily discernible.

A hardware flat spectrum generator has been described and the 1st order pileup spectra obtained with it confirm the predictions of the theory and the Monte Carlo simulations. In particular the data confirm the theory that the pulse rise and fall times can be determined from the slope of the pileup ramp which is produced. They also demonstrate that the correction algorithm is applicable to spectra with real-world multichannel analysers.

8.1 CONCLUSIONS

- * The theoretical analysis of the pulse pileup process has led to the development of a sophisticated correction algorithm which can be used in principle to correct a spectrum for as many orders of pileup as are desired, provided that matching impulse response functions can be generated.
- * It seems possible at least that the correction of spectra after-the-fact with the algorithm which has been developed here could replace the need for hardware pulse pileup rejectors in many cases. In any event such devices can never detect the near-perfect superimposition of two or more pulses where the composite pulse width and shape is virtually indential with that of a single one.
- * The concept of a flat spectrum as the simplest model for demonstrating pileup phenomena and associated theory has proven to be a powerful tool. The extension of it to an actual hardware device demonstrated a most convincing link between the theory

and simulations and the behaviour of real-world multi-channel analysers.

- * The dominance of 1st order pileup in causing actual spectral distortion, as opposed to the linear attenuation caused by a uniform distribution of losses, is such that for most purposes correction for it alone will be sufficient. The versatile reciprocal function developed to match the impulse response of semi-Gaussian pulse shapes makes possible very accurate correction for 1st order pileup.
- * The concept of splitting the pulse resolving time into the two components τ_F and τ_R is a very appropriate and useful one when dealing with modern ADC's, because they are now fast enough that these two components are the limiting factors in their ability to analyse pulses at high count rates and the effects of each must be considered separately.
- * The concept of the probability density function of pulses in coincidence being an impulse response, places the whole pulse pileup process in the context of modern signal processing theory and associated terminology which is used and understood by a wide and rapidly growing segment of the scientific and engineering community.

8.2 RECOMMENDATIONS FOR FUTURE WORK

The fast pace of science and engineering on many different fronts will undoubtedly have an on-going impact in the field of radiation detector pulse processing. Already the ability to digitize individual pulses at intervals of a few tens of nanoseconds suggests that a radical departure from conventional multichannel analysers will not be long in coming. On another front, the inexorable advances in computing speed and power

will make algorithms which are presently too time-consuming for routine use, a viable option in the future.

It has been reiterated a number of times throughout this study that there can never be a hardware solution to the problem of perfectly correlated pileup, i.e. the near perfect superimposition of two or more pulses. This also applies to the ultimate hardware solution, the high-speed sampling of individual pulse mentioned above.

Future work on the correction of spectral distortion due to pulse pileup might usefully focus on the following aspects.

- * Further refinement of the correction/distortion algorithm that has been developed with a view to implementation in an application-specific integrated circuit. For example the pileup gains to each channel, computed for a given coincidence, could be added or subtracted essentially in one multiparallel operation if the chip contained the necessary arrays and co-processing capability. This would increase the speed by orders of magnitude and would be analogous to ASIC's that are designed for such algorithms as the Fast Fourier Transform.
- * Refinement and further development of hardware simulators to generate pileup effects under controlled conditions. For example the flat-spectrum generator described in this study could be adapted to produce pileup of any required order, or combination of orders. A possible alternative approach to the one used here might be to use the high-speed digital sampling technique in reverse to generate real pulses with specific separations, shapes and amplitudes. This would amount to a "Hardware Monte Carlo" simulation and would provide a rigorous testing technique for laboratory multichannel analysers in applications from neutron activation analysis to high energy physics experiments.

APPENDIX 1A: Fixed-Number Pulse String Data

Sample Data from Circuit I Measurements generated by the Data Acquisition System

Z = 1.75 (Input Count Rate = 17500 c/s)

f_a	f_b	f_c	S	M	R_a	R_b	R_c	P(ABC)
---	---	---	-	-	---	---	---	-----
Pulses/String = 2								
11585	403	1626	323	18239	4.925	0.207	5.037	0.0232
11576	391	1638	323	18239	4.923	0.199	5.072	0.0232
11600	399	1623	329	18253	4.935	0.204	4.937	0.0236
Pulses/String = 3								
10687	677	1260	255	16587	4.874	0.447	4.935	0.0198
10744	671	1260	254	16629	4.916	0.443	4.962	0.0196
10738	678	1263	252	16636	4.898	0.448	5.020	0.0195
Pulses/String = 4								
10187	870	985	203	15492	4.949	0.732	4.842	0.0166
10151	875	990	198	15463	4.922	0.737	4.999	0.0162
10141	868	992	203	15464	4.914	0.727	4.878	0.0167
Pulses/String = 5								
9715	1027	781	159	14590	4.939	1.092	4.904	0.0136
9690	1027	786	160	14582	4.911	1.086	4.917	0.0137
9736	1028	783	158	14612	4.947	1.094	4.967	0.0134
Pulses/String = 6								
9391	1137	626	130	13933	4.961	1.503	4.822	0.0115
9394	1144	622	128	13931	4.961	1.526	4.883	0.0113
9387	1148	622	127	13928	4.950	1.534	4.908	0.0112
Pulses/String = 7								
9034	1239	507	103	13342	4.886	2.032	4.919	0.0095
9073	1233	509	102	13372	4.920	2.016	4.980	0.0094
9065	1234	505	104	13358	4.921	2.027	4.875	0.0095
Pulses/String = 8								
8836	1314	407	84	12937	4.900	2.679	4.876	0.0078
8845	1308	409	83	12937	4.914	2.656	4.930	0.0078
8843	1311	406	84	12934	4.911	2.675	4.861	0.0078
Pulses/String = 9								
8721	1361	333	68	12648	4.947	3.391	4.878	0.0065
8717	1355	338	68	12645	4.950	3.341	4.980	0.0065
8697	1359	337	69	12631	4.930	3.354	4.903	0.0065

APPENDIX 1_B

Fixed_Number Pulse Strings: CIRCUIT_I Predicted Data

N	fa	fb	fc	S	H	Ra	Rb	Rc	P(ABC)	
2	627.5	1322.9	120.7	343.8	4666.7	.3511	2.8484	.3511	.1424	Z = 0.30
3	621.0	1649.4	31.0	88.4	4278.2	.3511	13.8099	.3511	.0370	
4	619.3	1733.2	8.0	22.9	4178.6	.3511	55.9940	.3511	.0096	
5	618.9	1754.9	2.1	5.9	4152.8	.3511	218.3338	.3511	.0025	
6	618.8	1760.5	.5	1.5	4146.1	.3511	843.0769	.3511	.0006	
7	618.8	1762.0	.1	.4	4144.3	.3511	*****	.3511	.0002	
8	618.8	1762.3	.0	.1	4143.9	.3511	*****	.3511	.0000	
9	618.8	1762.4	.0	.0	4143.7	.3511	*****	.3511	.0000	
2	1481.2	1372.6	354.0	542.2	6914.8	.6529	1.5317	.6529	.1446	Z = 0.50
3	1440.9	1862.7	136.0	208.3	6199.3	.6529	5.4095	.6529	.0571	
4	1425.6	2049.0	53.2	81.4	5927.2	.6529	15.2270	.6529	.0226	
5	1419.6	2121.5	20.9	32.0	5821.4	.6529	40.0818	.6529	.0089	
6	1417.3	2150.0	8.2	12.6	5779.8	.6529	103.0068	.6529	.0035	
7	1416.3	2161.2	3.3	5.0	5763.4	.6529	262.3142	.6529	.0014	
8	1416.0	2165.6	1.3	2.0	5757.0	.6529	665.6328	.6529	.0005	
9	1415.8	2167.4	.5	.8	5754.4	.6529	*****	.6529	.0002	
2	2572.6	1241.8	643.1	628.2	8870.0	1.0237	.9768	1.0237	.1235	Z = 0.70
3	2458.4	1786.9	310.9	303.7	7875.8	1.0237	2.9079	1.0237	.0625	
4	2404.4	2044.7	153.8	150.2	7405.8	1.0237	6.7252	1.0237	.0316	
5	2378.0	2170.8	76.9	75.2	7175.8	1.0237	14.2715	1.0237	.0160	
6	2364.8	2233.5	38.7	37.8	7061.4	1.0237	29.1893	1.0237	.0081	
7	2358.2	2265.0	19.5	19.1	7004.0	1.0237	58.6792	1.0237	.0041	
8	2354.9	2280.8	9.9	9.6	6975.0	1.0237	116.9759	1.0237	.0021	
9	2353.2	2288.8	5.0	4.9	6960.4	1.0237	232.2189	1.0237	.0010	
2	4611.5	961.9	1067.8	611.6	11573.5	1.7459	.5728	1.7459	.0843	Z = 1.00
3	4318.3	1473.5	635.8	364.1	10265.0	1.7459	1.4736	1.7459	.0536	
4	4150.5	1766.3	388.5	222.5	9516.2	1.7459	2.8905	1.7459	.0341	
5	4050.4	1940.8	241.1	138.1	9069.6	1.7459	5.1188	1.7459	.0217	
6	3989.3	2047.5	151.0	86.5	8796.7	1.7459	8.6236	1.7459	.0138	
7	3951.4	2113.7	95.1	54.5	8627.4	1.7459	14.1357	1.7459	.0088	
8	3927.6	2155.2	60.1	34.4	8521.4	1.7459	22.8051	1.7459	.0056	
9	3912.7	2181.2	38.1	21.8	8454.7	1.7459	36.4401	1.7459	.0035	
2	6658.7	737.5	1348.2	529.6	13767.1	2.5460	.3928	2.5460	.0571	Z = 1.25
3	6180.1	1176.0	898.4	352.9	12286.1	2.5460	.9398	2.5460	.0410	
4	5876.8	1453.9	613.4	240.9	11347.6	2.5460	1.7017	2.5460	.0294	
5	5676.8	1637.1	425.4	167.1	10728.6	2.5460	2.7629	2.5460	.0211	
6	5541.3	1761.2	298.2	117.1	10309.6	2.5460	4.2408	2.5460	.0152	
7	5448.0	1846.7	210.5	82.7	10020.9	2.5460	6.2993	2.5460	.0109	
8	5382.9	1906.3	149.3	58.6	9819.5	2.5460	9.1662	2.5460	.0078	
9	5337.2	1948.2	106.3	41.8	9677.8	2.5460	13.1592	2.5460	.0056	

Fixed_Number Pulse Strings: CIRCUIT_I Predicted Data

N	fa	fb	fc	S	M	Ra	Rb	Rc	P(ABC)	
2	8995.3	547.2	1533.9	427.8	15975.0	3.5852	.2789	3.5852	.0372	Z = 1.50
3	8329.3	902.8	1110.6	309.8	14396.1	3.5852	.6356	3.5852	.0291	
4	7873.5	1146.3	820.9	229.0	13315.4	3.5852	1.0919	3.5852	.0227	
5	7550.4	1318.8	615.5	171.7	12549.5	3.5852	1.6753	3.5852	.0178	
6	7315.7	1444.1	466.3	130.1	11993.0	3.5852	2.4215	3.5852	.0139	
7	7142.1	1536.8	356.0	99.3	11581.5	3.5852	3.3759	3.5852	.0109	
8	7011.9	1606.3	273.3	76.2	11273.0	3.5852	4.5964	3.5852	.0085	
9	6913.5	1658.9	210.7	58.8	11039.5	3.5852	6.2773	3.5852	.0066	
2	11581.2	395.1	1622.2	328.6	18223.7	4.9368	.2026	4.9368	.0236	Z = 1.75
3	10752.4	671.9	1252.4	253.7	16614.5	4.9368	.4462	4.9368	.0196	
4	10148.4	873.6	982.9	199.1	15441.8	4.9368	.7391	4.9368	.0163	
5	9695.6	1024.9	780.9	158.2	14562.6	4.9368	1.0914	4.9368	.0136	
6	9348.7	1140.7	626.1	126.8	13889.0	4.9368	1.5150	4.9368	.0113	
7	9078.6	1230.9	505.6	102.4	13364.6	4.9368	2.0244	4.9368	.0094	
8	8865.6	1302.1	410.6	83.2	12951.0	4.9368	2.6370	4.9368	.0078	
9	8695.9	1358.7	334.9	67.8	12621.6	4.9368	3.3738	4.9368	.0065	
2	14364.8	278.7	1623.8	242.5	20521.2	6.6969	.1493	6.6969	.0147	Z = 2.00
3	13417.0	486.8	1319.6	197.1	18940.7	6.6969	.3209	6.6969	.0128	
4	12688.6	646.7	1085.9	162.1	17726.0	6.6969	.5182	6.6969	.0111	
5	12116.2	772.4	902.2	134.7	16771.6	6.6969	.7449	6.6969	.0097	
6	11658.6	872.8	755.3	112.8	16008.5	6.6969	1.0054	6.6969	.0084	
7	11287.7	954.2	636.3	95.0	15390.1	6.6969	1.3049	6.6969	.0073	
8	10983.7	1021.0	538.7	80.4	14883.1	6.6969	1.6491	6.6969	.0064	
9	10732.2	1076.2	458.0	68.4	14463.7	6.6969	2.0446	6.6969	.0055	

APPENDIX_2A: Fixed-Number Pulse String Data

Sample Data from Circuit II Measurements generated by the Data Acquisition System

Z = 1.50 (Input Count Rate = 15000 c/s)

f_a	f_b	f_c	S	Denom	N	R_a	R_b	R_c	P(ABC)
---	---	---	-	-----	-	---	---	---	-----
Pulses/String = 2									
7224	718	2008	544	15033	15055	3.597	0.281	3.692	0.0363
7157	723	2003	573	15086	15035	3.572	0.281	3.497	0.0381
7151	741	2003	568	15139	15037	3.570	0.288	3.525	0.0379
Pulses/String = 3									
5593	1287	1572	440	15037	12477	3.557	0.640	3.573	0.0293
5520	1304	1561	443	15006	12393	3.536	0.650	3.521	0.0295
5580	1284	1563	441	15026	12432	3.571	0.641	3.547	0.0293
Pulses/String = 4									
4325	1749	1218	348	15084	10424	3.550	1.117	3.502	0.0231
4378	1730	1226	343	15079	10474	3.570	1.102	3.570	0.0229
4330	1727	1222	348	14984	10418	3.544	1.100	3.508	0.0232
Pulses/String = 5									
3374	2065	967	263	14791	8868	3.488	1.679	3.676	0.0175
3375	2082	951	271	15032	8852	3.549	1.704	3.505	0.0181
3396	2067	961	262	14913	8871	3.532	1.689	3.667	0.0175
Pulses/String = 6									
2642	2347	746	209	14989	7644	3.540	2.458	3.582	0.0139
2636	2340	744	209	14958	7625	3.543	2.455	3.558	0.0139
2626	2352	743	208	14976	7623	3.534	2.474	3.575	0.0138
Pulses/String = 7									
2056	2556	580	164	15005	6679	3.547	3.437	3.530	0.0109
2071	2545	582	164	14998	6691	3.556	3.411	3.551	0.0109
2054	2560	578	166	15054	6679	3.557	3.441	3.471	0.0111
Pulses/String = 8									
1587	2723	452	127	14897	5920	3.512	4.705	3.566	0.0084
1594	2724	448	130	15054	5920	3.560	4.720	3.457	0.0086
1594	2720	453	128	14906	5930	3.515	4.681	3.549	0.0085
Pulses/String = 9									
1239	2839	353	102	14866	5339	3.514	6.249	3.464	0.0067
1253	2847	356	97	14913	5362	3.519	6.279	3.654	0.0065
1245	2854	351	98	15026	5347	3.550	6.364	3.591	0.0065

APPENDIX 2_B

Fixed_Number Pulse Strings: CIRCUIT_II Predicted Data

N	fa	fb	fc	S	M	Ra	Rb	Rc	P(ABC)	
2	52.6	1643.5	149.9	427.1	3000.0	.3511	2.8484	.3511	.1424	Z = 0.30
3	13.7	2070.5	39.0	111.0	2423.0	.3511	13.8099	.3511	.0370	
4	3.6	2181.5	10.1	28.8	2273.1	.3511	55.9940	.3511	.0096	
5	.9	2210.3	2.6	7.5	2234.1	.3511	218.3338	.3511	.0025	
6	.2	2217.8	.7	1.9	2224.0	.3511	843.0769	.3511	.0006	
7	.1	2219.8	.2	.5	2221.4	.3511	*****	.3511	.0002	
8	.0	2220.3	.0	.1	2220.7	.3511	*****	.3511	.0000	
9	.0	2220.4	.0	.0	2220.5	.3511	*****	.3511	.0000	
2	308.1	1830.2	472.0	722.9	5000.0	.6529	1.5317	.6529	.1446	Z = 0.50
3	121.7	2553.1	186.4	285.5	3805.1	.6529	5.4095	.6529	.0571	
4	48.1	2838.6	73.6	112.8	3333.2	.6529	15.2270	.6529	.0226	
5	19.0	2951.4	29.1	44.5	3146.8	.6529	40.0818	.6529	.0089	
6	7.5	2996.0	11.5	17.6	3073.1	.6529	103.0068	.6529	.0035	
7	3.0	3013.6	4.5	7.0	3044.0	.6529	262.3142	.6529	.0014	
8	1.2	3020.5	1.8	2.7	3032.5	.6529	665.6328	.6529	.0005	
9	.5	3023.3	.7	1.1	3028.0	.6529	*****	.6529	.0002	
2	906.1	1709.2	885.1	864.6	7000.0	1.0237	.9768	1.0237	.1235	Z = 0.70
3	458.4	2573.9	447.8	437.4	5250.2	1.0237	2.9079	1.0237	.0625	
4	231.9	3011.2	226.5	221.3	4365.1	1.0237	6.7252	1.0237	.0316	
5	117.3	3232.5	114.6	111.9	3917.4	1.0237	14.2715	1.0237	.0160	
6	59.3	3344.4	58.0	56.6	3690.9	1.0237	29.1893	1.0237	.0081	
7	30.0	3401.0	29.3	28.6	3576.3	1.0237	58.6792	1.0237	.0041	
8	15.2	3429.7	14.8	14.5	3518.3	1.0237	116.9759	1.0237	.0021	
9	7.7	3444.2	7.5	7.3	3489.0	1.0237	232.2189	1.0237	.0010	
2	2570.4	1326.3	1472.3	843.3	10000.0	1.7459	.5728	1.7459	.0843	Z = 1.00
3	1634.3	2169.6	936.1	536.2	7684.5	1.7459	1.4736	1.7459	.0536	
4	1039.1	2705.7	595.2	340.9	6212.2	1.7459	2.8905	1.7459	.0341	
5	660.7	3046.6	378.4	216.8	5276.1	1.7459	5.1188	1.7459	.0217	
6	420.1	3263.4	240.6	137.8	4680.9	1.7459	8.6236	1.7459	.0138	
7	267.1	3401.2	153.0	87.6	4302.5	1.7459	14.1357	1.7459	.0088	
8	169.8	3488.8	97.3	55.7	4061.9	1.7459	22.8051	1.7459	.0056	
9	108.0	3544.5	61.8	35.4	3908.9	1.7459	36.4401	1.7459	.0035	
2	4626.7	994.1	1817.2	713.8	12500.0	2.5460	.3928	2.5460	.0571	Z = 1.25
3	3321.9	1707.9	1304.8	512.5	9969.0	2.5460	.9398	2.5460	.0410	
4	2385.1	2220.3	936.8	368.0	8151.8	2.5460	1.7017	2.5460	.0294	
5	1712.5	2588.3	672.6	264.2	6847.0	2.5460	2.7629	2.5460	.0211	
6	1229.6	2852.5	482.9	189.7	5910.2	2.5460	4.2408	2.5460	.0152	
7	882.8	3042.2	346.7	136.2	5237.6	2.5460	6.2993	2.5460	.0109	
8	633.9	3178.3	249.0	97.8	4754.7	2.5460	9.1662	2.5460	.0078	
9	455.1	3276.1	178.8	70.2	4407.9	2.5460	13.1592	2.5460	.0056	

Fixed_Number Pulse Strings: CIRCUIT_II Predicted Data

N	fa	fb	fc	S	H	Ra	Rb	Rc	P(ABC)	
2	7170.7	713.5	2000.1	557.9	15000.0	3.5852	.2789	3.5852	.0372	Z = 1.50
3	5606.8	1271.3	1563.9	436.2	12442.1	3.5852	.6356	3.5852	.0291	
4	4384.0	1707.5	1222.8	341.1	10442.0	3.5852	1.0919	3.5852	.0227	
5	3427.9	2048.6	956.1	266.7	8878.2	3.5852	1.6753	3.5852	.0178	
6	2680.3	2315.3	747.6	208.5	7655.4	3.5852	2.4215	3.5852	.0139	
7	2095.8	2523.8	584.6	163.0	6699.3	3.5852	3.3759	3.5852	.0109	
8	1638.7	2686.8	457.1	127.5	5951.7	3.5852	4.5964	3.5852	.0085	
9	1281.3	2814.3	357.4	99.7	5367.1	3.5852	6.1573	3.5852	.0066	
2	10062.8	496.5	2038.3	412.9	17500.0	4.9368	.2026	4.9368	.0236	Z = 1.75
3	8367.8	909.4	1695.0	343.3	15048.8	4.9368	.4462	4.9368	.0196	
4	6958.3	1252.7	1409.5	285.5	13010.5	4.9368	.7391	4.9368	.0163	
5	5786.2	1538.2	1172.1	237.4	11315.5	4.9368	1.0914	4.9368	.0136	
6	4811.6	1775.7	974.6	197.4	9906.0	4.9368	1.5150	4.9368	.0113	
7	4001.1	1973.1	810.5	164.2	8734.0	4.9368	2.0244	4.9368	.0094	
8	3327.2	2137.2	674.0	136.5	7759.3	4.9368	2.6370	4.9368	.0078	
9	2766.7	2273.8	560.4	113.5	6948.8	4.9368	3.3738	4.9368	.0065	
2	13173.6	337.6	1967.1	293.7	20000.0	6.6969	.1493	6.6969	.0147	Z = 2.00
3	11462.0	631.3	1711.5	255.6	17739.1	6.6969	.3209	6.6969	.0128	
4	9972.9	886.9	1489.2	222.4	15772.0	6.6969	.5182	6.6969	.0111	
5	8677.2	1109.3	1295.7	193.5	14060.5	6.6969	.7449	6.6969	.0097	
6	7549.8	1302.8	1127.4	168.3	12571.3	6.6969	1.0054	6.6969	.0084	
7	6568.9	1471.1	980.5	146.5	11275.6	6.6969	1.3049	6.6969	.0073	
8	5715.5	1617.6	853.4	127.4	10148.3	6.6969	1.6491	6.6969	.0064	
9	4972.9	1745.0	742.6	110.9	9167.4	6.6969	2.0446	6.6969	.0055	

APPENDIX 3: Fixed-Interval Pulse String Data

Sample Data from Circuit II Measurements generated by the Data Acquisition System

Z = 1.75 (Input Count Rate = 17500 c/s)

P/Ev	f_a	f_b	f_c	S	DENOM	N	R_a	R_b	R_c	P(ABC)
---	---	---	---	-	-----	-	---	---	---	-----
Intervals/String = 1										
2.84	7605	501	1606	880	17128	14683	4.736	0.201	1.825	0.0502
2.86	7693	519	1597	884	17451	14772	4.816	0.209	1.806	0.0505
2.85	7760	520	1587	870	17532	14782	4.88	0.212	1.825	0.0497
Intervals/String = 2										
5.09	4817	1419	978	604	17776	10377	4.924	0.897	1.620	0.0345
5.07	4988	1386	988	593	17941	10524	5.048	0.877	1.667	0.0338
5.15	4950	1364	998	600	17652	10508	4.960	0.854	1.664	0.0342
Intervals/String = 3										
7.42	2754	2047	568	373	17476	7251	4.849	2.177	1.524	0.0213
7.39	2856	2025	578	367	17644	7350	4.940	2.143	1.576	0.0209
7.40	2777	2044	569	366	17517	7261	4.879	2.186	1.555	0.0209
Intervals/String = 4										
9.84	1550	2642	318	196	17484	5358	4.875	4.790	1.622	0.0112
9.84	1558	2454	314	200	17677	5355	4.956	4.770	4.571	0.0114
9.85	1512	2461	316	203	17253	5325	4.791	4.747	1.577	0.0115
Intervals/String = 5										
12.76	977	2647	200	123	17458	4472	4.877	8.186	1.629	0.0070
12.71	1019	2639	204	124	17766	4520	4.987	8.033	1.647	0.0071
12.75	1010	2639	207	121	17429	4513	4.874	8.036	1.713	0.0069
Intervals/String = 6										
15.00	658	2754	137	85	17241	3992	4.794	12.410	1.618	0.0048
15.08	675	2746	138	83	17486	4000	4.894	12.449	1.669	0.0047
15.04	708	2736	142	84	17740	4036	4.990	12.145	1.698	0.0047
Intervals/String = 7										
17.25	417	2842	83	50	17913	3609	5.021	21.383	1.667	0.0028
17.20	425	2833	83	53	18193	3613	5.128	20.788	1.551	0.0030
17.30	420	2832	85	51	17637	3609	4.942	20.871	1.676	0.0029
Intervals/String = 8										
19.61	244	2891	48	31	18120	3340	5.102	36.716	1.549	0.0017
19.74	236	2892	49	29	17377	3332	4.851	37.166	1.665	0.0016
19.71	229	2892	47	32	17311	3327	4.826	36.570	1.498	0.0018
Intervals/String = 9										
22.45	157	2916	30	20	18502	3202	5.329	58.966	1.537	0.0011
22.57	154	2926	31	18	17747	3209	4.965	59.637	1.717	0.0010
22.42	150	2929	31	18	17524	3207	4.885	59.975	1.681	0.0010

APPENDIX_4A: Relative Probability Data

Simultaneous Measurements with Circuit II as generated by the Data Acquisition System

$Z = 1.5$ (Input Count Rate = 15000 c/s)

Tau	% Contribution of Fixed Number Strings, n = 2, to n = 40									
1	0.0	99.2	74.6	42.9	18.9	6.7	1.9	0.6	0.1	0.0
	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-
2	0.0	0.7	25.4	53.5	63.9	54.4	36.0	20.2	9.3	4.1
	1.5	0.6	0.1	0.1	0.0	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-
3	0.0	0.0	0.1	3.6	16.9	35.2	49.8	51.0	43.7	31.0
	19.5	11.2	5.6	2.1	0.9	0.5	0.1	0.1	0.0	-
	-	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-
4	0.0	0.0	0.0	0.0	0.4	3.6	11.7	25.3	37.9	44.4
	44.6	35.6	27.2	18.4	10.2	5.9	2.8	1.2	0.6	0.2
	0.2	0.1	0.0	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-
5	0.0	0.0	0.0	0.0	0.0	0.0	0.6	2.9	8.4	18.2
	28.1	37.7	40.2	38.6	33.0	23.8	17.4	10.6	5.7	2.3
	1.4	1.3	0.3	0.0	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-	-
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.6	2.2
	5.7	13.0	21.9	29.5	35.1	37.3	33.5	28.5	21.5	16.5
	11.2	4.4	3.4	1.9	0.5	0.75	0.0	-	-	-
	-	-	-	-	-	-	-	-	-	-
7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1
	0.6	1.9	4.5	9.9	17.2	23.7	31.0	34.2	35.6	30.7
	25.3	22.4	14.2	10.4	5.9	3.6	1.3	1.0	0.8	0.0
	-	-	-	-	-	-	-	-	-	-
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.1	0.5	1.4	3.3	7.4	12.5	19.4	25.1	31.3
	32.0	30.4	27.4	23.3	18.5	16.3	11.5	8.0	3.4	3.6
	1.9	0.9	0.0	-	-	-	-	-	-	-
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.1	0.3	1.1	2.5	5.5	9.4	14.3
	19.5	25.4	29.4	30.4	30.5	25.2	19.9	14.2	9.7	12.2
	3.8	4.0	1.5	0.0	-	-	-	-	-	-

APPENDIX 4_B: Relative Probability Data Generated by Numerical Convolution.

% Contributions of Fixed Number Strings (n = 2 to 20), to Fixed Interval Strings (M = 1_Tau to 9_Tau)

Z= .30 Number of Convolutions = 19

M	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_10	n_11	n_12	n_13	n_14	n_15	n_16	n_17
1	.00	99.85	55.33	21.17	6.20	1.47	.30	.05	.01	.00	.00	.00	.00	.00	.00	.00	.00
2	.00	.00	44.37	65.69	31.42	28.11	12.00	4.25	1.30	.35	.08	.02	.00	.00	.00	.00	.00
3	.00	.00	.00	12.69	39.11	53.47	47.00	30.78	16.23	7.22	2.80	.96	.30	.09	.02	.01	.00
4	.00	.00	.00	.00	2.67	15.76	34.89	45.97	43.11	31.57	19.09	9.89	4.50	1.83	.68	.23	.07
5	.00	.00	.00	.00	.00	.44	4.85	16.67	31.57	40.72	39.81	31.47	20.97	12.12	6.21	2.87	1.21
6	.00	.00	.00	.00	.00	.00	.06	1.22	6.32	16.75	28.90	36.76	37.01	30.92	22.15	13.92	7.82
7	.00	.00	.00	.00	.00	.00	.00	.01	.26	2.00	7.26	16.46	26.70	33.63	34.60	30.15	22.84
8	.00	.00	.00	.00	.00	.00	.00	.00	.00	.05	.55	2.68	7.84	16.00	24.85	31.06	32.50
9	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.01	.13	.86	3.22	8.18	15.49	23.26

Z= .50 Number of Convolutions = 21

M	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_10	n_11	n_12	n_13	n_14	n_15	n_16	n_17
1	.00	99.75	58.50	24.10	7.65	1.98	.43	.08	.01	.00	.00	.00	.00	.00	.00	.00	.00
2	.00	.00	41.00	64.48	54.23	32.05	14.85	5.73	1.91	.56	.15	.04	.01	.00	.00	.00	.00
3	.00	.00	.00	10.68	35.09	51.59	49.01	34.80	19.94	9.66	4.08	1.53	.52	.16	.05	.01	.00
4	.00	.00	.00	.00	2.03	12.83	30.61	43.60	44.33	35.27	23.21	13.09	6.50	2.89	1.17	.43	.15
5	.00	.00	.00	.00	.00	.30	3.57	13.24	27.13	37.95	40.32	34.67	25.16	15.85	8.86	4.47	2.05
6	.00	.00	.00	.00	.00	.00	.04	.81	4.54	13.01	24.36	33.67	36.88	33.56	26.20	17.96	11.01
7	.00	.00	.00	.00	.00	.00	.00	.00	.16	1.30	5.09	12.52	22.09	30.27	33.92	32.22	26.62
8	.00	.00	.00	.00	.00	.00	.00	.00	.00	.03	.32	1.69	5.38	11.94	20.18	27.48	31.34
9	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.07	.49	1.99	5.50	11.34	18.55

Z= .70 Number of Convolutions = 23

M	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_10	n_11	n_12	n_13	n_14	n_15	n_16	n_17
1	.00	99.65	61.60	27.22	9.34	2.62	.62	.13	.02	.00	.00	.00	.00	.00	.00	.00	.00
2	.00	.00	37.70	62.84	56.61	36.05	18.07	7.56	2.73	.87	.25	.07	.02	.00	.00	.00	.00
3	.00	.00	.00	8.90	31.14	49.10	50.25	38.57	23.94	12.58	5.77	2.36	.87	.30	.09	.03	.01
4	.00	.00	.00	.00	1.52	10.29	26.37	40.51	44.52	38.37	27.38	16.78	9.05	4.38	1.92	.78	.29
5	.00	.00	.00	.00	.00	.20	2.58	10.29	22.75	34.42	39.61	36.95	29.11	19.93	12.12	6.64	3.33
6	.00	.00	.00	.00	.00	.00	.02	.53	3.17	9.82	19.91	29.81	35.42	35.00	29.68	22.13	14.76
7	.00	.00	.00	.00	.00	.00	.00	.00	.09	.81	3.46	9.20	17.59	26.16	31.83	32.86	29.51
8	.00	.00	.00	.00	.00	.00	.00	.00	.00	.01	.18	1.03	3.55	8.55	15.67	23.18	28.73
9	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.03	.27	1.18	3.53	7.91	14.05

Z= 1.00 Number of Convolutions = 25

M	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_10	n_11	n_12	n_13	n_14	n_15	n_16	n_17
1	.00	99.50	66.08	32.21	12.34	3.89	1.04	.24	.05	.01	.00	.00	.00	.00	.00	.00	.00
2	.00	.00	32.93	59.65	59.24	41.93	23.49	11.02	4.48	1.62	.53	.16	.04	.01	.00	.00	.00
3	.00	.00	.00	6.65	25.48	44.42	50.57	43.36	30.16	17.81	9.19	4.23	1.77	.68	.24	.08	.02
4	.00	.00	.00	.00	.97	7.18	20.40	34.91	42.88	41.42	33.20	22.88	13.90	7.58	3.76	1.72	.73
5	.00	.00	.00	.00	.00	.11	1.53	6.76	16.69	28.25	36.45	38.19	33.84	26.10	17.89	11.07	6.25
6	.00	.00	.00	.00	.00	.00	.01	.26	1.77	6.11	13.86	23.29	31.10	34.57	33.03	27.75	20.88
7	.00	.00	.00	.00	.00	.00	.00	.00	.04	.38	1.82	5.43	11.65	19.46	26.65	30.99	31.39
8	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.07	.46	1.77	4.78	9.86	16.42	22.93
9	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.01	.10	.50	1.67	4.20	8.41

Z= 1.25 Number of Convolutions = 27

M	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_10	n_11	n_12	n_13	n_14	n_15	n_16	n_17
1	.00	99.38	69.59	36.58	15.26	5.27	1.55	.40	.09	.02	.00	.00	.00	.00	.00	.00	.00
2	.00	.00	29.16	56.43	60.48	46.50	28.42	14.59	6.51	2.58	.92	.30	.09	.03	.01	.00	.00
3	.00	.00	.00	5.13	21.14	39.91	49.41	46.22	35.17	22.75	12.89	6.52	3.00	1.26	.49	.18	.06
4	.00	.00	.00	.00	.65	5.18	15.97	29.77	39.94	42.20	37.06	28.03	18.70	11.21	6.12	3.08	1.44
5	.00	.00	.00	.00	.00	.06	.95	4.59	12.36	22.86	32.30	37.09	36.07	30.55	23.02	15.66	9.74
6	.00	.00	.00	.00	.00	.00	.01	.14	1.04	3.92	9.72	17.89	26.20	31.97	33.55	30.99	25.65
7	.00	.00	.00	.00	.00	.00	.00	.00	.02	.19	1.01	3.29	7.74	14.19	21.34	27.26	30.35
8	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.03	.22	.93	2.74	6.21	11.36	17.44
9	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.04	.22	.83	2.28	5.02

Z= 1.50 Number of Convolutions = 29

M	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_10	n_11	n_12	n_13	n_14	n_15	n_16	n_17
1	.00	99.25	72.88	41.06	18.56	6.98	2.25	.63	.16	.04	.01	.00	.00	.00	.00	.00	.00
2	.00	.00	25.63	52.81	60.81	50.54	33.54	18.75	9.12	3.95	1.55	.55	.18	.06	.02	.00	.00
3	.00	.00	.00	3.90	17.24	35.10	47.06	47.81	39.59	27.93	17.28	9.56	4.81	2.22	.95	.38	.14
4	.00	.00	.00	.00	.42	3.65	12.17	24.60	35.89	41.30	39.56	32.67	23.83	15.63	9.35	5.15	2.63
5	.00	.00	.00	.00	.00	.04	.58	3.02	9.82	17.75	27.33	34.25	36.38	33.69	27.77	20.69	14.10
6	.00	.00	.00	.00	.00	.00	.00	.07	.58	2.40	6.50	13.05	20.86	27.81	31.92	32.26	29.23
7	.00	.00	.00	.00	.00	.00	.00	.00	.01	.09	.53	1.89	4.85	9.72	15.97	22.32	27.20
8	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.01	.10	.46	1.48	3.65	7.31	12.27
9	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.02	.09	.38	1.15	2.77

Z= 1.75 Number of Convolutions = 31

M	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_10	n_11	n_12	n_13	n_14	n_15	n_16	n_17
1	.00	99.13	75.91	45.58	22.20	9.05	3.17	.97	.27	.07	.01	.00	.00	.00	.00	.00	.00
2	.00	.00	22.35	48.90	60.26	53.88	38.62	23.38	12.36	5.82	2.48	.97	.35	.12	.04	.01	.00
3	.00	.00	.00	2.93	13.83	30.24	43.72	48.03	43.09	32.99	22.18	13.36	7.31	3.68	1.72	.75	.31
4	.00	.00	.00	.00	.27	2.51	9.02	19.72	31.14	39.85	40.41	36.28	28.79	20.57	13.40	8.05	4.49
5	.00	.00	.00	.00	.00	.02	.34	1.91	6.06	13.23	22.10	30.09	34.76	35.04	31.45	25.53	18.97
6	.00	.00	.00	.00	.00	.00	.00	.04	.32	1.41	4.14	9.04	15.71	22.79	28.48	31.36	30.96
7	.00	.00	.00	.00	.00	.00	.00	.00	.00	.04	.27	1.03	2.87	6.26	11.20	17.04	22.63
8	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.01	.04	.21	.75	2.01	4.37	8.00
9	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.01	.04	.16	.54	1.41

Z= 2.00 Number of Convolutions = 34

M	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_10	n_11	n_12	n_13	n_14	n_15	n_16	n_17
1	.00	99.00	78.67	50.07	26.14	11.50	4.36	1.45	.43	.12	.03	.01	.00	.00	.00	.00	.00
2	.00	.00	19.35	44.81	58.86	56.38	43.44	28.36	16.19	8.26	3.82	1.62	.64	.23	.08	.03	.01
3	.00	.00	.00	2.16	10.90	25.54	39.66	46.90	45.39	37.54	27.30	17.81	10.57	5.77	2.92	1.38	.62
4	.00	.00	.00	.00	.17	1.69	6.52	15.33	26.11	35.18	39.56	38.43	33.04	25.58	18.08	11.78	7.14
5	.00	.00	.00	.00	.00	.01	.19	1.18	4.02	9.47	17.10	25.20	31.52	34.42	33.50	29.51	23.80
6	.00	.00	.00	.00	.00	.00	.00	.02	.17	.80	2.52	5.95	11.21	17.62	23.87	28.51	30.55
7	.00	.00	.00	.00	.00	.00	.00	.00	.00	.02	.13	.53	1.61	3.79	7.36	12.15	17.51
8	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.02	.09	.35	1.03	2.44	4.85
9	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.01	.07	.24	.67

APPENDIX 5

SUMMARY REVIEW OF PREVIOUS WORK

This summary is intended to highlight the more significant developments in solving the pulse pileup problem, (as it applies to the acquisition of nuclear radiation spectra), which have been reported in the literature. This literature is voluminous and it is not considered useful to examine every single account of work relating to pulse pileup which is on record. The references are listed in chronological order for historical continuity, but are not always discussed in that order if common factors emerge in papers which are far apart in time. The initial portion reviews the very early history of the art, including Geiger Muller (G.M.) Counters and the approaches taken to deal with lost counts, in order to provide an historical perspective for the techniques subsequently developed to deal with pulse pileup.

The discipline of radiation spectroscopy as applied to nuclear radiation had its beginnings toward the end of the second world war. The first multichannel analysers (MCAs) were used to acquire spectra from gas proportional counters, which were the only energy dispersive detectors available at that time. One of the earliest accounts of the development of MCAs was that of *Freundlich et al.*,⁶ then at the National Research Council of Canada, who provided a brief but interesting vignette on the origins of MCAs in the introduction to their paper. It seems that the idea of an instrument to sort pulse amplitudes and hence radiation energies in real time was originated by O.R. Frisch, the celebrated Austrian physicist, when he was at the Cavendish laboratory at Cambridge. It was Frisch, in collaboration with Lise Meitner, who first confirmed the spontaneous fission of the uranium atom in the 1930's, at a time when the implications for a bomb were just beginning to be realized at the political level on both sides. Frisch's suggestion was immediately taken up in England and Canada and resulted in five versions of a rudimentary MCA being built. One of these was the one built by Hanna, Ward and Wescott at the

National Research Council in Montreal (the forerunner of AECL) which predated the one described by Freundlich et al. It was known as the "1945 Kicksorter" and was described in an unpublished NRC report in 1946. It was not until 1949 that this description was published in a formal paper by *Westcott and Hanna*⁷. Both instruments were daunting in their physical size and complexity but represented imaginative innovations in that complete spectra (albeit only 20 or 30 channels) could be acquired in real time without recourse to the pseudo mass spectrometer approach of the magnetic β particle device.

Nuclear event counters had been in use since the 1930's and the problem of chance coincidences causing errors had already been addressed as had the dead-time problem. The latter was a concern in G.M. Counter measurements where the counter is paralyzed for a fixed time following the detection of an event. *Schiff*¹ in a 1936 paper published the relevant theory and a nomogram for computing the counts lost by a single counter having a known resolving time when exposed to a source which decayed significantly during the measurement time. He also discussed the problem of coincidences in multiple counter systems, which is analogous to the pulse pileup problem but in a different context.

*Alaoglu and Smith*² developed the theory relating to the derandomising effect of scaling circuits used to reduce the count rate of nuclear events being counted by mechanical registers in a 1938 paper. In the same year *Eckart and Shonka*³ investigated the probability of accidental coincidences in multiple Geiger Counter systems. Two years later *Feller*⁴, of renewal theory fame, gave the solution to a problem involving the probability of the time separation between two events being greater than some fixed arbitrary constant, given n events in a time interval of length D . This problem arose in the context of lost counts in nuclear event counters. *Elmore*⁸, some ten years later, again tackled the problem of the effect of scaling circuits previously described by Alaoglu and Smith, but demonstrated the use of Laplace transforms for finding probability distribution moments.

Another interesting early development was the construction of an "interval selector" by *Roberts*⁵ in 1940. This device allowed a tally to be maintained of all counts where the inter-arrival times were less than a predetermined time. The purpose of this was to check on counting systems which were suspect because of uncertain resolving times. Again this represents an early attempt to deal with what was essentially a pulse pileup problem as applied to simple counting circuits.

As was indicated at the outset the foregoing discussion has been included to provide some background relating to the early history of radiation measurement instrumentation techniques and theory. From this point on the focus will be on the work relating to the spectral distortion and erroneous count rates (caused by pulse pileup) dating from the early sixties.

*Soucek*⁹ in a 1963 paper, generated pileup density functions experimentally by using a random pulse generator which allowed pulses of constant amplitude to pile up. He showed that the results were in accordance with theory which he presented and that errors could be predicted. *Gold*¹⁰ assumed that a distorted spectrum could be expressed as the convolution of an ideal spectrum and a detector response function which manifested itself as a base-line distortion. He used numerical techniques to carry out the convolution for various assumed detector responses and presented predicted distortions of selected ideal spectra.

*DeLotto and Dotti*¹³, in a 1966 paper, considered an alternative approach to the numerical convolution of the detector response function, which was to evaluate the products of the characteristic functions of the respective probability distributions. However the integrals involved were such that they had to be evaluated using cumulants which involved the same order of numerical computation as Gold had used. They then considered specific pulse shapes and derived pileup distributions based on several commonly used ones. They showed that experimental data

agreed well with their predicted distributions and used the results to optimize the resolving time to be employed in a hardware pileup rejector. In a parallel paper published within months of this one, the authors were joined by Mariotti, (*DeLotto et al.*¹⁴) in yet another approach to the pileup problem where only the pulse tail contributions were considered. At the time these papers were submitted (1965), the technique of pole-zero cancellation had only just been published¹¹, so that pulse tail undershoot was still a major source of amplitude error. They sampled the base-line level on a random basis by injecting rectangular pulses into the chain and using the spectral tail above its channel position as the probability density of the base-line amplitude over the long term. They then used it to perform a pseudo-numerical deconvolution of the distorted spectrum to recover the true one. The results showed a dramatic improvement but "ideal" spectra (i.e. ones with vanishingly low count rates) were not presented for comparison.

*Harms*¹⁵ was the first to describe a technique for compensating spectra for lost counts under conditions of changing count rates due to the decay of short lived nuclides during the measurement time. His technique involved the storage of extra counts in channels as they were addressed by the system ADC when discriminator logic outputs indicated that incoming pulses were not registered. This amounted to pseudo-instantaneous corrections being made on a continuous basis, the amount of the compensation in any given short interval being proportional to the total count rate at that time.

*Anders*¹⁶ deduced that "pulses lost" due to coincidences, i.e. pileup, from each channel are proportional to the channel count rates across the entire spectrum. He reasoned that this would include a channel where pulses from a pulse generator of known input rate were being accumulated and that therefore the measured loss in this channel would allow a correction factor to be computed for the entire spectrum. What he did not consider apparently was that for every two pulses lost as a result of a single coincidence, one is gained in some other channel which may still be in the

portion of the spectrum being measured. Nevertheless the idea of using a pulser having a known input rate to monitor the data acquisition as a control parameter was an important advance.

*Bolotin et al.*¹⁸ took this idea one step further and proposed the use of a pulse generator where the repetition rate was controlled by the total system count rate. This technique allowed corrections to be made for dead-time and pileup when the counting rate was changing with time, as is the case when short half life radioisotopes are being analyzed. *Wiernik*^{22,23} pointed out that the limitation on pulse repetition rate, imposed by the necessity to avoid the statistical errors due to the pulser being non-random, could be avoided by using a randomly triggered pulse generator. He carried out experiments using random and periodic pulse generators and the results showed that a random pulse produced an acceptable correction, even when its rate was comparable to the total count rate from the system detector.

Two papers in 1969 and 1970 by *Waibel*^{17,19} presented theoretical and experimental work on the determination of pileup probability density functions for unipolar and bipolar RC pulse shapes and for single and double delay line pulse shapes. The theoretical and measured density functions showed remarkable similarity in all cases.

*Soucek et al.*²⁰ took a theoretical approach and derived a random process which took account of pulse shape, and amplitude distribution of the pulses, (i.e. spectral shape). He applied the discrete theory appropriate to shot noise analysis to produce this model, and then used numerical processing to evaluate the integrals he obtained to produce plots of the base-line amplitude probability density functions for various combinations of pulse shapes and spectral shapes. He then discussed these in the context of optimizing hardware pileup rejection circuits.

*Sabbah and Klein*²¹ in 1971 described a hardware pileup rejector based on the idea of summing a gaussian shaped pulse with its second

derivative. By adding these in appropriate proportions a double peak resulted, with a time from initial rise to first return to base-line which was independent of amplitude but very sensitive to the partial superimposition of even a small pulse on the main one.

*Wyittenbach*²⁴ also in 1971 was possibly the first to realize that the criterion for a pulse not to be involved in a coincidence was that the adjacent inter-arrival times must each be greater than the resolving time τ of the system electronics. This led him to the correct conclusion that such pulses represented a fraction $e^{-2N\tau}$ of the incoming total. He devised a formula with which his name was subsequently associated in the literature, to correct the observed count rates for losses due to chance summing of pulses. He too however did not consider the possible gains in a spectral peak in a high energy part of the spectrum due to coincidences in lower energy channels.

At this point mention should be made of the major contribution to the study of dead-time in quantitative spectrometry made by *Müller*²⁶. The reference given is but one of a large number of his publications, but encapsulates the relevant theory relating to dead-time and indirectly to some aspects of pulse pileup.

*Cohen*²⁷ in 1974 devised a formula for correction of peak count rates for pileup losses which took into account multiple coincidences, however it did not ensure that these were preceded and followed by spaces greater than the system resolving time. He also assumed that all spectral peaks would be subject only to losses due to pileup, although he did refer to gains on the high sides of peaks and above the upper level discriminator. *Fedotov and Volkov*²⁸ presented the theory of spectral deconvolution described by Gold, but modified to allow for dead-time as well as pileup. They obtained good agreement between calculated and experimentally obtained spectral distortion.

*Datlowe*³⁰ investigated the effect of pulse pileup on the X-ray spectra acquired from solar flares in the OSO-7 satellite experiment. He assumed a "square wave approximation" for pulse summation, i.e. that coincidences resulted in pulses that were the total sums of the energies of the contributing pulses. He also made assumptions about the true spectral shapes involved. The key point however was to show that the probability density function for say 3-pulse coincidences could be obtained by the convolution of that for 2-pulse coincidences with itself. Then the density function for say 4-pulse coincidences would involve another convolution with the 2-pulse or "single coincidence" density function. Because his X-ray spectra had constant and predictable pulse height distributions, he was able to pre-compute and store the various orders of pileup density functions, thereby reducing the computation time required for spectral correction.

He expanded on this theme in a later paper³⁹ in which he made the distinction between "leading edge" and "base-line" pileup. He derived pileup density functions based on pulse shapes for leading edge pileup and also base-line functions for incorporation into an overall model. It should be pointed out however that the saving in computational time which he achieved by having pre-computed density functions, only applied when the pulse height distribution (shape) of the spectra being acquired was known.

*Blatt*³¹ investigated the probabilities of multiple coincidences but the formulae he derives do not agree with the analysis by *Tenney*⁵⁸.

Wielopolski and Gardner^{32,35,36} in a series of papers in 1976 and 1977 investigated leading edge pileup for single coincidences (two pulses) only. They computed the probabilities of partial sums of pulses which occurred within the system resolving time falling between channel boundaries k and $(k + 1)$. This essentially amounts to computing the density function and digitizing it into an arbitrary number of channels. They used a specific pulse shape (parabolic) as a starting point and

generated distorted spectra with this model, and showed that it agreed well with experimental data. They accomplished the reverse procedure, correction of a distorted spectrum for which the true one was unknown, by a least squares iteration procedure. This involved an initial estimate of the true spectrum, application of the model to generate pileup, comparison of the result with the measured piled up spectrum, generation of a better estimate of the true spectrum, and so on until no further significant improvement of the estimate resulted.

In the medical field the high count rates involved in gamma-camera work were known to cause severe pileup problems with associated errors. *Sorenson*³³ realized that a simple way to evaluate this would be to monitor a spectral window above the highest photopeak of interest where sum pulses would accumulate. He found that the method was inexpensive and reasonably accurate in correcting observed count rates.

*Debertin and Schotzig*³⁷ reported errors in the pulser method for pileup loss correction due to distortion of spectral peaks introducing uncertainties into the background subtraction.

*Roscoe and Furr*³⁸ devised a semi-empirical solution to correct spectral peaks for dead-time and pileup under conditions of time-dependent count rates from short half-life nuclides. It involved the determination and tabulation of a time-dependent correction factor which was used in a computer memory look-up table. They showed that satisfactory results could be obtained by measuring the count rates from a constant intensity source in the presence of a rapidly decaying one and demonstrated that the estimate of the constant intensity source did not vary with a changing total count rate.

*Dyer et al.*⁴² (1979) attempted to model pulse pileup for square and triangular pulse shapes. They were apparently unaware of the previous work discussed above and their derivation of the number of single coincidences which will occur in a Poisson pulse train seems to be in error.

They were concerned with X-ray spectra having a specific distribution as was *Datlowe*³⁰.

*Fleming*⁴⁴ modified the correction formula developed by *Wyttenbach* to include the case of a time-dependent count rate due to a source with a half life comparable to the measurement time. This allows correction for pileup losses for analyses involving a specific nuclide once an initial experiment has been done to determine a constant in the formula.

*Whittlestone*⁴⁵, also apparently unaware of previous work, developed the basic theory to correct for single coincidences based on the square pulse approximation and used it as a basis for pileup loss correction of neutron spectra being acquired with organic scintillators.

An intriguing approach to the whole problem of rejection of piled-up pulses and the effect on recorded spectra is provided by *Skopyuk and Shevchenko*⁴⁶. They postulate a player playing against nature. Each has two options, the player to record or not to record an incoming pulse, and nature to provide either a single pulse, or two or more in a coincidence. This leads to a two-by-two game matrix which they show will have an optimum solution if the player chooses a suitable strategy appropriate to the probability of coincidences occurring. This can be realized by calculating the probability for coincidences at a given count rate and using this in conjunction with a random number generator to accept or reject pulse heights measured by an ADC. There will be an optimum acceptance/rejection ratio for a particular count rate based on Poisson statistics that will minimize spectral distortion due to chance coincidence.

*Volkov et al.*⁴⁷ pursued the approach of describing pileup distortion as a convolution of the base-line distribution with the ideal spectrum, but with an added factor. They computed the probability of pulse maxima as a joint probability of the first derivative of the base-line distribution being zero, and the second derivative being negative. They expressed the characteristic function of this distribution as an integral which included

the pulse shape and showed that this allowed multiple coincidences to be handled. They calculated the spectral distortion of a specific spectral shape that would result, based on their model with some drastically simplifying assumptions. It is not clear that their theoretical results would be practical for numerical processing, or that they would lead to any better results than the other more direct approaches to leading edge pileup correction discussed above.

*Lewellen and Murano*⁴⁸ in 1981 were still finding that pulse pileup in gamma-camera work for medical diagnoses was a limiting factor on accuracy. They conducted tests which showed that an estimate of "misplaced events" (sum pulse energies) was the most reliable way to correct for lost counts.

*Greenburger*⁴⁹ described a sophisticated hardware pileup rejector based on adaptive pattern recognition techniques for detecting pileup in the presence of significant noise. This was designed to cope with problems encountered in low energy X-ray spectroscopy. A similar approach was followed by *Melbert*⁵² who developed a generalized filter model as a computer program which allowed him to generate hardware filter designs to optimize signal-to-noise ratios for given input parameters.

A vast amount of work has been done on improvements to the signal conditioning electronics in spectrometry systems, particularly since the mid-sixties. The improvements have contributed significantly to reducing spectral distortion due to base-line pileup. No attempt is made here to review this work which is a separate subject in itself, however it is appropriate to note the contributions of *Fairstein and Hahn*¹², *Radeka*²⁵, and *Goulding and Landis*⁵⁷. The publications listed in each case are but one of many by these authors.

*Desilva and Chatt*⁵⁰ described a modified version of the pileup correction formula due to *Cohen* which allowed a time-dependent count rate to be corrected. It involved the initial evaluation of a number of system

constants at different constant count rates. These constants could then be used in a formula to handle a changing count rate.

*Koskelo*⁵³ observed that pileup errors in neutron activation analytical work could be related empirically to count rate. These included not only lost counts but spectral peak shape changes. He developed simple factors to account for these and showed that corrections based on them produced results which were comparable with much more sophisticated algorithms.

*Gal and Bibok*⁵⁵ described a clever variation of the pulser method for real time count rate correction. They used a counter to divide the total count rate from the detector down to a lower rate and to produce a logic level pulse train in which all the pulse widths were made equal to the system resolving time. This pulse train was then ANDed with the system BUSY signal to reject pulses in the train when a coincidence occurred between the BUSY signal and the pulse train. The count rate from the resulting pulse train was then used as a live time clock to correct the observed spectral count rates for both dead-time and pileup losses. They presented data to show that the scheme allowed corrections that were accurate to within 1% for count rates up to 50 kHz.

*Dorfel et al.*⁵⁶ provided yet another variant on the pulser method by the development of a pseudo-random pulser based on shift register "add and shift" algorithms. They showed that it produced an acceptable inter-arrival time distribution for the purpose of pileup and dead-time correction.

*Tenney*⁵⁸ in a 1983 paper went further than previous workers in arriving at a general expression for multiple coincidences which takes account of the fact that there must be preceding and following inter-arrival times which are greater than τ , the system resolving time. He showed that any pulse train can be expressed as an infinite geometric series consisting of single pulses, and coincidences of two or more pulses in strings, where the

successive inter-arrival times are less than τ . He went on to consider pulse pileup based on the rectangular pulse or "square wave" approximation and arrived at the same general formula as *Datlowe*³⁰ but did not refer to his work.

*Bell et al.*⁵⁹ described a hardware pileup rejector based on a digital technique. Instead of generating inspection times by monostable circuits, they used a 50 MHz clock and counters which allowed the time to be preset in steps of 20 ns. It did not however ensure that two successive inter-arrival times were greater than the system resolving time.

*Kennedy*⁶⁰ developed a variant of the *De Silva and Chatt*⁵⁰ method for correcting peak count rates, where these change over the measurement time due to short-lived isotopes. He noted that for a given radioisotope and a given data acquisition system, the dead-time/pulse-pileup losses could be expressed as a polynomial in terms of the true instantaneous count rate and a series of coefficients. He reasoned that for a number of isotopes these would sum in a linear fashion and that once the various coefficients for each isotope had been determined, (in separate once-and-for-all experiments with a system having fixed settings), then the appropriate corrections could be applied, since the isotopic content of the samples in his case were known. His experimental data showed satisfactory results at count rates which caused up to 50% dead-time.

*Macwani*⁶¹ noted that corrections for pileup and dead-time are considerably simplified for time-dependent count rates if the ADC has a fixed dead-time. Accordingly he modified a conventional *Wilkinson*⁷³ ramp ADC so that it had a fixed dead-time of 50 μ secs. The technique was inexpensive but not as effective as the other more elaborate ones. It also had the serious drawback of greatly increasing the system dead-time.

In the late 70's *Westphal*^{40,43} introduced the concept of "Loss Free Counting" (LFC). This technique involves a real-time estimate of the true instantaneous total count rate from a detector, as given by the system

event discriminator. Whenever the ADC is busy processing a pulse into a given channel, additional counts are added to this channel based on the BUSY interval multiplied by the true instantaneous count rate. This scheme in principle allows on-line correction for time-dependent count rates.

In a subsequent refinement of this technique *Westphal*⁶² described a "virtual pulse generator" scheme for the correction of time-dependent count rates for dead-time and pileup. This involved extending the system BUSY signal by a single pulse rise time, thereby ensuring that a new pulse would be free of pileup, and using this extended BUSY signal to gate a high frequency clock. The gated and ungated signals were fed to two counters one of which would overflow (possibly more than once) by an amount proportional to the extended BUSY signal duty cycle. For rapidly changing input count rates the counter capacities were chosen such that they only took a millisecond or so to produce the ratio as an integer plus a fraction. The integer was used as a weighting factor to augment the contents of the next channel addressed by the ADC, while the fraction was retained and added to the next ratio measurement.

Westphal^{34.51.54.65.74} also developed hardware solutions to minimize base-line pileup. One of these was a time variant base-line restorer and another was a discrete reset for the capacitor which stores the detector charge in the preamplifier.

In yet another development he introduced the concept of a "pre-loaded filter", in which the resistor in an RC low-pass filter is shorted out during a pulse rise time to track that portion of the pulse, but then a switch is opened to allow the averaging property of the filter to be applied until the next pulse arrives. This technique is essentially another variant of time-dependent filtering.

Another area in which pulse pileup can cause errors is in coincidence counting, whereby the activity of an isotope can be measured on the basis

of separately detected disintegrations being coincident. The preset coincidence time window is a factor in the coincidence count rate and is thus a parameter in the measurement. Pulse pileup can distort the bipolar pulse shape and hence the position of the zero crossing time reference. *Furrer et al.*⁶³ conducted a theoretical and experimental investigation to quantify the errors that can be anticipated under given conditions and showed that the effect is real and predictable.

*Chalupka and Tagesen*⁶⁷ described a pileup rejector which ensured that there were inter-arrival times $> \tau$ preceding and following an accepted pulse. They used a digital inspection time generator consisting of a 100 MHz oscillator and two high speed counters, one for the interval preceding and one for that following the pulse being inspected. A comparison of spectra acquired at 10^5 counts per second (c/s) with and without the inspection circuit showed excellent results against a reference spectrum taken at less than 1000 c/s.

*Petersen et al.*⁶⁸ developed a software routine to eliminate pileup and other distortions from X-ray spectra. The second derivative of the smoothed spectrum is used to identify peaks as valleys between two maxima. Those that do not correlate with a look-up table of predetermined peaks are identified as being either the result of pileup, or as escape peaks (due to annihilation of 511 keV photons) and are subtracted from the original measured spectrum. Examples were presented which indicated that the scheme was effective. It is however limited to situations where a considerable number of measurement parameters must be determined a priori.

The emergence of "flash" ADCs has resulted in their application to pulse height analysis. A paper in 1985 by *Hilsenrath et al.*⁶⁶ describes a self contained single chip pulse integrator-processor system. It is designed to acquire 8-bit pulse height data at a 5 MHz rate. Threshold discriminators detect pulse leading and trailing edges and execute decisions as to whether a pulse is present, absent or distorted due to pileup. The output

can be either an integration of the digital samples over a predetermined period to produce a number proportional to pulse area, or a peak reading determined internally by comparisons of successive sample magnitudes. The various threshold limits, integration times, etc., can be software controlled. Preliminary data were presented which indicated performance comparable to that of a dedicated MCA.

A similar approach was tried on an experimental basis by *Chrien and Sutter*⁶⁹. They captured successive 320 nanosecond segments of an incoming pulse train from a detector preamplifier in a transient waveform recorder. These segments or "frames" were divided into three regions corresponding to pre-pulse, pulse and post-pulse intervals. Algorithms were used which either rejected frames where the pre- and post-pulse intervals exceeded predetermined levels, or attempted to estimate the true pulse height by subtracting the base-line elevation due to the pileup. Their experiments showed that the effect of deliberately introduced pulse undershoot, (base-line pileup), could be removed by this individual pulse processing capability.

A paper by *Drndarevic et al.*⁷⁷ also investigated the digital sampling approach and exploited the possibilities for individual pulse correction which it offers. In another paper⁷⁵ they describe a system designed to extract a weak high-energy signal from pileup caused by high count rate low-energy signals. An improved pulse-width-sensitive pileup-rejector is described, which in conjunction with improved base-line inspection and shortening of the detector signal allows count rates up to 400,000 c/s without significant distortion due to pileup. A third paper⁷⁶ by these authors reports on the use of a pulse amplifier with time-invariant trapezoidal shaping, together with a shape-sensitive pileup rejector.

Hardware pileup rejectors are only as good as the efficacy of the pulse-shape discrimination which they contain. *Bialkowski et al.*⁷⁸ developed a zero-crossing discriminator which incorporates D.C. negative-feedback to stabilize the trigger point against variations in temperature and count

rate. This greatly improved the performance of their pileup-rejection circuit at very high count rates (400,000 c/s).

*Johns and Yaffe*⁷¹ in a substantial 1987 paper combined the single and multiple coincidence formulae developed by *Tenney*⁵⁸, with the spectral correction techniques of *Wielopolski and Gardner*^{32,35,36}. They developed theory to allow the use of either periodic or random pulsers as reference signals in order to compute true spectral count rates, and thereby reconstruct true spectra from measured ones distorted by pulse pileup.

*Proctor*⁷² used a pulser injection technique which involved applying the divided true count rate (obtained from the rate output of the main amplifier) to two tail pulse generators. Both provided constant amplitude pulses, one for energy calibration of the system, and the other for estimating counts lost due to dead-time and pileup. The controlling logic introduced delays to avoid self-defeating pileup between the triggering pulses and the generated ones. It also subtracted one pulse from the division scalars for each one generated, thereby not adding to the rate being used to generate pulses in the first instance, which would amount to positive feedback and cause an unstable injection rate. The ratio of the known to the observed pulser count rates provided a true correction factor for all spectral peaks for counts lost due to pileup and dead-time.

Proctor pointed out that the pulser injection technique has a significant advantage over the time correction techniques, whereby logic "BUSY" signals are combined to extend the live time. When true coincidences occur involving the pulser pulses which do not extend the pulse width, then pulses are removed from the pulser channel(s) and are accounted for. They are not accounted for in the other time correction schemes.

REFERENCES

- 1) Statistical Analysis of Counter Data
L.I. Schiff
Physical Review, Vol. 50
1936
- 2) Statistical Theory of a Scaling Circuit
L. Alaoglu, N.M. Smith, Jr.
Physical Review Vol. 53
1938
- 3) Accidental Coincidences in Counter Circuits
C. Eckart, F.R. Shonka
Physical Review, Vol. 53
1938
- 4) On the Time Distribution of So-Called
Random Events
W. Feller
Physical Review, Vol. 57
1940
- 5) The Interval Selector: A Device for
Measuring Time Distribution Pulses
A. Roberts
Rev. Sci. Inst., Vol. 12
1941
- 6) A Pulse Analyser for Nuclear Research
H.F. Freundlich, E.P. Hincks, W.J. Dzeroff
Rev. Sci. Inst., Vol. 18
No. 2 1947
- 7) A Pulse Amplitude Analyzer for Nuclear
Research Using Pretreated Pulses
C.H. Westcott, G.C. Hanna
Rev. Sci. Inst., Vol. 20
No. 3 1949
- 8) Statistics of Counting
W.C. Elmore
Nucleonics Jan. 1950
- 9) Pulse-Height Analysis Accuracy at High
Rates
B. Soucek
Proc. Int. Symp. on
Nuclear Electronics,
Paris, p. 437-41 1963
- 10) Calculation of Spectral Distortion due
to Pile-up Effect
R. Gold
The Review of Scientific
Instruments, Vol. 36, No.
6, p. 784-96 1964
- 11) Elimination of Undesirable Undershoot
in the Operation and Testing of
Nuclear Pulse Amplifiers
C.H. Nowlin, J.L. Blankenship
Rev. Sci. Inst. (36)
1830 1965
- 12) Nuclear Pulse Amplifiers - Fundamentals
and Design Practice
E. Fairstein, J. Hahn
Nucleonics 23 Nos. 7,
9, 11 1965; Nucl. 24
Nos. 1, 3 1966
- 13) Pile-up Effects in Time Measurements
I. DeLotto, D. Dottie
Nucl. Inst. & Meth. (39)
p. 281-92 1966

- 14) **A Method to Correct the Amplitude Distortions due to Pile-Up Effects**
I. DeLotto, D. Dotti, D. Mariotti
Nucl. Inst. & Meth. (40)
p. 169-172 1966
- 15) **Automatic Dead-Time Correction for Multichannel Pulse-Height Analyzers at Variable Counting Rates**
J. Harms
Nucl. Inst. & Meth. (53)
p. 192-196 1967
- 16) **Experiences with the Ge(Li) Detector for High Resolution Gamma-ray Spectrometry & A Practical Approach to the Pulse Pile-up Problem**
O.U. Anders
Nucl. Inst. & Meth. (68)
p. 205-8 1969
- 17) **Zur Berechnung Von Pile-Up-Spektren**
E. Waibel
Nucl. Inst. & Meth. (74)
p. 236-244 1969
- 18) **Simple Technique for Precise Determination of Counting Losses in Nuclear Pulse Processing Systems**
H.H. Bolotin, M.G. Strauss, D.A. McClure
Nucl. Inst. & Meth. 83 (1)
p. 1-12 1970
- 19) **Bestimmung Der Faltungsfunktion Zur Berechnung Von Spektren Aufgestoekter Impulse**
E. Waibel
Nucl. Inst. & Meth. (86)
p. 29-34 1970
- 20) **Amplitude Distributions of the Poisson Process Transformed by Systems with Memory, with Special Reference to Shot Noise and Pile-up**
B. Soucek, M. Cimerman, B. Breyers
IEEE Trans. Nucl. Sci.
NS-17 1970, p. 375-82
- 21) **Pile Up Rejection by Comparison of the Shaped Pulse with its Second Derivative**
B. Sabbah, I. Klein
Nucl. Inst. & Meth. (95)
p. 163-171 1971
- 22) **Comparison of Several Methods Proposed for Correction of Dead-Time Losses in the Gamma-Ray Spectrometry of Very Short Lived Nuclides**
M. Wiernik
Nucl. Inst. & Meth. 95 (1)
p. 13-18 1971
- 23) **Normal and Random Pulse Generators for the Correction of Dead-Time Losses in Nuclear Spectrometry**
M. Wiernik
Nucl. Inst. & Meth. (96)
p. 325-29 1971
- 24) **Coincidence Losses in Activation Analysis**
A. Wyttenbach
Jour. Radioan. Chem.
Vol. 8, p. 335-43 1971

- 25) **Trapezoidal Filtering of Signals from Large Germanium Detectors at High Rates**
V. Radeka
IEEE Trans. Nucl. Sci. NS-19, No. 1 1972
- 26) **Deadtime Problems**
Jorg W. Müller
Nucl. Inst. & Meth. (112) p. 47-57 1973
- 27) **Live Time & Pile-up Correction for Multichannel Analyser Spectra**
E.M.J. Cohen
Nucl. Inst. & Meth. (121) p. 25-32 1974
- 28) **Analysis of the Pulse-Height Spectra Distortion Caused by the Pile-Up Effect**
S.N. Fedetov, N.G. Volkov
Nucl. Inst. & Meth. 122 p. 463-65 1974
- 29) **A Simple Accurate Model of Correcting, XRF Pulse-Height Spectra for Pulse Pileup**
L. Wielopolski, R.P. Gardner
Trans. Am. Nucl. Soc. Supplement, Vol. 21 No. 3 p. 39-41 1975
- 30) **Pulse Pile-Up in Hard X-Ray Detector Systems**
D.W. Datlowe
Space Science Instrumentation, Vol. 1, p. 389-406 1975
- 31) **Characteristics of Random-Sum Peaks in Nuclear Radiation Spectra**
S.L. Blatt
Nucl. Inst. & Meth. (128) p. 277-281 1975
- 32) **Prediction of the Pulse Height Spectral Distortion Caused by the Peak Pile-up Effect**
L. Wielopolski, R.P. Gardner
Nucl. Inst. & Meth. (133) p.303-9 1976
- 33) **Methods of Correcting Anger Camera Deadtime Losses**
J.A. Sorenson
J. Nucl. Med. V. 17 (2) p. 137-141 1976
- 34) **A High Rate Gamma Spectroscopy System for Activation Analysis of Short-Lived Isomeric Transitions**
G.P. Westphal
Nucl. Inst. & Meth. (136) p. 271-283 1976
- 35) **A Generalized Method for Correcting Pulse-Height Spectra for the Peak Pile-Up Effect due to Double Sum Pulses Part I. Predicting spectral distortion for arbitrary pulse shapes**
R.P. Gardner, L.Wielopolski
Nucl. Inst. & Meth. (140) p. 289-296 1977

- 36) **A Generalized Method for Correcting Pulse-Height Spectra for the Peak Pile-Up Effect due to Double Sum Pulses Part II. The inverse calculation for obtaining true from observed spectra**
L. Wielopolski, R.P. Gardner
Nucl. Inst. & Meth. (140)
p. 297-303 1977
- 37) **Limitations of the Pulser Method for Pile-up Corrections in Ge(Li) Spectrometry**
K. Debertin & U. Schotzig
Nucl. Inst. & Meth. (140)
p. 337-40 1977
- 38) **Time-dependent Deadtime and Pile-up Corrections for Gamma Ray Spectroscopy**
B.A. Roscoe, A.K. Furr
Nucl. Inst. & Meth. (140)
(2) p. 401-4 1977
- 39) **The Role of the Waveform in Pulse Pile-Up**
D.W. Datlowe
Nucl. Inst. & Meth. (145)
p. 365-378 1977
- 40) **Loss-Free Counting - A Concept for Real-Time Compensation of Dead-Time and Pile-Up Losses in Nuclear Pulse Spectroscopy**
G.P. Westphal
Nucl. Inst. & Meth. (146)
p. 605-606 1977
- 41) **Digital Separator of Pile-Ups of Time Intervals**
S.G. Basiladze, N. Kuang Min
Instruments & Expt'l. Techniques, Vol. 21, No. 6, Pt. 1, p. 1518-1521 1978
- 42) **Pulse Pileup Effects on Plasma Electron Temperature Measurements by Soft X-Ray Energy Analysis**
G.R. Dyer, G.H. Nelson, G. Kelley
Nucl. Inst. & Meth. (161)
p. 365-370 1979
- 43) **On the Performance of Loss-free Counting, A Method for Real-time Compensation of Deadtime and Pile-up Losses in Nuclear Pulse Spectroscopy**
G.P. Westphal
Nucl. Inst. & Meth. (163)
p. 182-96 1979
- 44) **A Direct Measurement of Random Summing Losses in Gamma Counting**
R.F. Fleming
Trans. Am. Nucl. Soc.
Vol. 33, p. 234-5 1979
- 45) **The Effect of Pulse Pile-up on Discrimination Between Neutrons and Gamma Rays**
S. Whittlestone
Nucl. Inst. & Meth. (173)
(2) p. 347-50 1980

- 46) **Using Gate Methods to Eliminate Overlapping Pulses in Gamma Spectrometry**
M.I. Skopyuk, V.A. Shevchenko
Trans. from Pribory i Tekhnika, Eksperimenta, No. 2 p. 63-65 1980
- 47) **Method for Analyzing Distortions of Amplitude Spectra due to Pulse Overlapping in Spectrometer Channel**
N.G. Volkov, V.V. Sin'kov, S.N. Fedotov
Prib. Tekh. Eksp. No. 1, p. 105-107, Jan/Feb 1980
- 48) **A Comparison of Count Rate Parameters in Gamma Cameras**
T.K. Lewellen, R. Murano
J. Nucl. Med. (22) p. 161-168 1981
- 49) **Optimizing Pulse Pileup Detection for Soft X-Ray Spectroscopy**
A.J. Greenberger
Nucl. Inst. & Meth. (188) p. 125-132 1981
- 50) **Development of a Method to Correct for Coincidence Losses in High Count Rate Gamma-Ray Spectrometry**
K.M. Desilva, A. Chatt
Can. Nucl. Soc. 2nd Annual Conf. 1981
- 51) **High Rate Gamma-Spectroscopy and Related Problems**
G.P. Westphal
Jour. Radioan. Chem. Vol. 61, No. 1-2, p. 111-119 1981
- 52) **On the Design of Filters for Pulse-Height and Time Analysis**
J.G. Melbert
IEEE Trans. Nuc. Sci. Vol. NS 29, No. 1 1982
- 53) **A Solution to the Pulse Pileup and Peak Shape Distortion Problem**
M.J. Koskelo
J. Radioanalytical Chem. Vol. 70, No. 1-2, p. 473-481 1982
- 54) **Real-Time Correction of Counting Losses in Nuclear Pulse Spectroscopy**
G.P. Westphal
Jour. Radioan. Chem. Vol. 70, No. 1-2, P. 387-410 1982
- 55) **Simple Dead-Time and Pile-Up Correction Technique Using a Gated Periodic Pulse Train**
J. Gal, G. Bibok
Nucl. Inst. & Meth. (206) p. 465-469 1983
- 56) **A Pseudorandom Pulser Technique for the Correction of Deadtime & Pile-up Losses in X-ray Spectrometry**
G. Dorfel, W. Kluge, M. Kubsch
Nucl. Inst. & Meth. (214) p. 435-40 1983

- 57) **Design Philosophy for High-Resolution Rate and Throughput Spectroscopy Systems**
F.S.Goulding, D.A. Landis, N.W. Madden
IEEE Trans. Nucl. Sci. NS-30, No. 1 1983
- 58) **Idealized Pulse Pileup Effects on Energy Spectra**
F.H. Tenney
Nucl. Inst. & Meth. (219) p. 165-172 1984
- 59) **A Digital Pulse-Pair Detecting Circuit**
Z.W. Bell, J.W. McConnell, E.D. Carroll
Nucl. Inst. & Meth. (211) p. 551-553 1983
- 60) **A Dead-Time Correction Method for Short-Lived Radioisotopes Using Measured Peak Areas**
G. Kennedy
Nucl. Inst. & Meth. (224) p. 207-210 1984
- 61) **An Empirical Correction for Deadtime & Pulse Pile-up Losses in Gamma-Ray Spectrometry**
M. Macwani
Nucl. Inst. & Meth. (225) p. 122-26 1984
- 62) **Application of the Virtual Pulse Generator Method for Real Time Correction of Counting Losses in High Rate Gamma Spectroscopy**
R. Pepelnik, G. Westphal, B. Anders
Nucl. Inst. & Meth. (226) (2-3), p. 411-17 1984
- 63) **Timing Problems due to Pile-Up in Coincidence Counting**
M. Furrer, J.J. Gostely, P. Lerch
Nucl. Inst. & Meth. (226) p. 455-462 1984
- 64) **Determination of Counting Losses and Pile-Up Rejection in Ionizing Radiation Spectrometry**
J. Kierzek, J.L. Parus
J. Radioanalytical Nucl. Articles, 81/1 p. 131-142 1984
- 65) **Instrumental Correction of Counting Losses in Nuclear Pulse Spectroscopy**
G.P. Westphal
Nucl. Inst. & Meth. Vol. B10-11 PT p. 1047-50, Vol. 2, 1985
- 66) **A Single Chip Pulse Processor for Nuclear Spectroscopy**
F. Hilsenrath, H.D. Voss, J.C. Bakke
IEEE Trans. Nuc. Sci. Vol. NS-32, No. 1, 1985
- 67) **A Digital High Precision Pile-up Inspection Unit**
A. Chalupka, S. Tagesen
Nucl. Inst. & Meth. A245(1), p. 159-61 1986

- 68) **XSPEK - A New Program for the Evaluation of Energy Dispersive X-Ray Fluorescence Spectra**
W. Petersen, P. Ketelsen, A. Knochel
Nucl. Inst. & Meth. (A245) p. 535-541 1986
- 69) **Noise and Pileup Suppression by Digital Signal Processing**
R.E. Chrien, R.J. Sutter
Nucl. Inst. & Meth. (A249) p. 421-425 1986
- 70) **Superimposed Renewal Processes: A New Method of Superimposing a Poisson Distribution with Periodic Pulses**
G. Faraci, A.R. Pennisi
Phys. Rev. A (USA) 33(1), p. 583-8 1986
- 71) **Correction of Pulse-Height Spectra for Peak Pileup Effects using Periodic and Random Pulse Generators**
P.C. Johns, M.J. Yaffe
Nucl. Inst. & Meth. (A255) p. 559-581 1987
- 72) **An Implementation of the Pulse Injection Technique for Dead-Time and Pileup Corrections in Gamma Spectroscopy Suitable for Automated Systems**
A.E. Proctor
Nucl. Inst. & Meth. (A256) p. 355-360 1987
- 73) **A Stable Ninety-nine Channel Pulse Amplitude Analyser for Slow Counting**
D.H. Wilkinson
Proc. Cambr. Phil. Soc. 46, p. 508 1950
- 74) **Quantitative Gamma Spectrometry at High Counting Rates**
G.P. Westphal
Jour. Radioan. Chem. Vol. 114, No. 2, P. 257-64 1987
- 75) **A Signal Processor for High Counting Rate Gamma Ray Spectroscopy with NaI(Tl) Detectors**
V.R. Drndarevic, P. Ryge, T. Gozani
IEEE Trans. Nucl. Sci. Vol. 35, No. 1, p. 222-25 1988
- 76) **Amplifier with Time-Invariant Trapezoidal Shaping and Shape-Sensitive Pileup Rejector for High-Rate Spectroscopy**
V.R. Drndarevic, P. Ryge, T. Gozani
IEEE Trans. Nucl. Sci. Vol. 36, No. 1, p. 1326-9 1989
- 77) **Digital Signal Processing for High Rate Gamma-Ray Spectroscopy**
V.R. Drndarevic, P. Ryge, T. Gozani
Nucl. Inst. & Methods Phys. Res. A., Vol. A277, p. 532-6, 1989
- 78) **A Pulse Shape Discriminator with High Precision of Neutron and Gamma Ray Selection at High Counting Rate**
J. Bialkowski, M. Moszynski, D. Wolski
Nucl. Inst. & Methods Phys. Res. A., Vol. A275, No. 2, p. 322-8

79)	Probabalistic Models in Engineering Sciences Vol. II (Chapters 2 and 3)
Larson and Shubert	Wiley & Sons
1979
80)	Introduction to Stochastic Processes (Chapters 1 and 2)
Hoel, Port and Stone	Houghton Mifflin Co.
Boston 1972
81)	First Course in Stochastic Processes
2nd Ed. (Chapters 1, 2 and 3)
Karlin and Taylor	Academic Press
82)	Poisson and Non-Poisson Behaviour of Radioactive Systems
N.A. Frigerio	Nucl. Inst. & Meth. (114)
p. 175-7 1974
83)	A Stochastic Population Process & its Application to Bubble Chamber Measurements
P.J. Brockwell and J.E. Moyal	J. Appl. Prob.
p. 3280-84 1966
84)	An Experimental Test of Muller Statistics Counting Statistics with a Non-extending Deadtime
D.R. Jordan & G.W. McBeth	Nucl. Inst. & Meth. (155)
p. 557-62 1978
85)	On Probability Problems in the Theory of Counters
W. Feller	Courant Anniversary Vol
Studies & Essays
Interscience 1948
86)	Renewal Theory
D.R. Cox	Methuen Monographs
on Applied Probability
and Statistics 1962
87)	The Statistical Analysis of Series of Events
D.R. Cox and P.A.W. Lewis	Methuen Monographs
on Applied Probability
and Statistics 1966
88)	Interval Densities for Extended Dead Times
J.W. Muller	Rapport BIPM-112
1971
89)	Probability, Random Variables and Stochastic Processes
A. Papoulis	McGraw Hill
New York, 1965

END

1 | 1 | 0 4 | 9 1

FIN