The Data Acquisition Circuit for a Fourth Generation γ-ray Computed Tomography Scanner

by

Kishalay Kundu

Thesis submitted to the Faculty of the Graduate Studies
of the University of Manitoba in partial fulfillment
of the requirements for the degree of
Master of Science
2003

Department of Electrical and Computer Engineering and
Department of Medical Physics,
University of Manitoba,
Winnipeg, Canada

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION PAGE

THE DATA ACQUISITION CIRCUIT FOR A FOURTH GENERATION γ-RAY COMPUTED TOMOGRAPHY SCANNER

BY

KISHALAY KUNDU

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University

of Manitoba in partial fulfillment of the requirements of the degree

of

Master of Science

This thesis is dedicated to my mother who taught me that in life it is better to be good than it is to be better

Abstract

Brachytherapy is a branch of radiotherapy which treats cancer by placing radioactive sources close to or inside tumors in controlled doses. One of the very important steps in brachytherapy is Treatment Planning. Treatment planning is required to pre-compute the exact locations and dwell times of the radioisotopes inside the body.

One of the primary inputs to the treatment planning procedure is an estimation of the location of applicators (through which the isotopes are delivered) inside the body. Present applications use two orthogonal radiographs (an anterior-posterior and a lateral) to estimate the applicator positions. Radiographs provide 2-D projections of the 3-D anatomy. This makes it difficult to estimate the locations of important 3-D structures and is therefore an error prone method.

One of the solutions to this problem is to acquire 3-D CT (computed tomography) images of the anatomy with the applicators. However, due to numerous complications. it is impractical to move the patient, with the surgically inserted applicators, to and from the CT scanning room and the Brachytherapy room. The solution to this is having a dedicated CT scanner for brachytherapy treatment planning purposes. This thesis describes the design and implementation of the Data Acquisition System of an in-house CT scanner which can be used as a treatment planning aid for brachytherapy. The CT scanner described in this thesis was a result of joint group effort between the department of the Electrical and Computer Engineering and the department of Medical Physics at Cancer Care Manitoba.

The Data Acquisition System comprises of several different modules, the primary of which are the analog radiation detector module which samples radiation beam attenuations which cross the body, the digital hardware control module which controls the operations of the analog module, and the embedded micro-controller based software module which load parameters into the hardware control module.

This thesis describes the design considerations and implementation details of each of the aforementioned modules and explains the interactions between them.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my academic and thesis advisor, Dr. Gregory Bridges for his help, support and constant involvement in guiding me towards my goals. I would like to thank him for lending me a helping hand and a patient ear whenever I needed one. Without his support over this period of time, this thesis would not have come to a fruition.

I would also like to thank the staff at Department of Electrical and Computer Engineering at the University of Manitoba for their support and help whenever I have needed it. I would also like to thank Dr. Satyapal Rathee and Dr. Daniel Rickey for their help in the research work involved in the formation of this thesis. Thanks also to Dr. Anita Berndt for her role in the project involved in this thesis work.

This thesis would not have been a success without the contribution of my friends and colleagues at the Manitoba Cancer Treatment and Research Foundation (now Cancer Care Manitoba), who provided me with moral support when the goings got tough and lots of laughs when times were easy. Thanks in particular to Steven Millter and Boyd McCurdy for sticking with me through tougher times. I am a better person for having known them.

Finally, I would like to thank my thesis examining committee members for taking the time to go through my thesis and giving their suggestions and corrections. Their valuable comment helped make this a more complete work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

Cancer is a leading cause of death in Canada. In 2002, an estimated 136,900 Canadians were diagnosed with cancer and about 66,200 died of cancer [13]. About 50 out of every 100 Canadian males and 48 out of every 100 females who contract cancer die. Age-standardized incidence of cancer has risen from 291.8 per 100,000 in 1973 to an estimated 347.3 per 100,000 in 2002 for females and from 355.8 per 100,000 in 1973 to an estimated 442.0 per 100,000 in 2002 for males [13]. Furthermore, in Manitoba, about 2,700 males and about 2,600 females were diagnosed with some form of cancer in 2002. Research in cancer treatment is of utmost significance in the fight against cancer.

The methods of cancer treatment can be broadly categorized as *surgery*-based, *chemotherapy*-based and *radiotherapy*-based. There are several different treatment procedures in each of these broad categories. The work described in this thesis is related to the field of radiotherapy.

Radiotherapy can be divided into *teletherapy* and *brachytherapy*. Teletherapy is a method of cancer treatment where cancer cells inside the body are irradiated by radiation beams generated by an external instrument. In brachytherapy (*brachy*- near, *therapy* - treatment), radioactive sources are placed in close proximity to the tumor.

Brachytherapy is further divided into high-dose-rate (HDR) and low-dose-rate (LDR) types. In high-dose-rate brachytherapy, high-activity radio-isotopes are sent out into applicators placed within or near the tumor, where they reside for small amounts of times, irradiating the tumor. In low-dose-rate brachytherapy, short-half-life and low-activity radio-isotopes are surgically inserted in or near the tumor , where the source decays over a period of time, irradiating the tumor.

This thesis concerns the high-dose-rate brachytherapy method of cancer treatment. More specifically, it concentrates on computed tomography (CT) imaging as applied to treatment planning in high-dose-rate brachytherapy. This thesis describes the design and implementation of the Data Acquisition System (DAS) hardware for an in-house 4^{th} generation CT scanner built at the Medical Physics department at the Manitoba Cancer Treatment and Research Foundation (now Cancer Care Manitoba). This CT scanner is designed to be integrated with the Brachytherapy system at MCTRF.

Sections 1.1 and 1.2 give a brief introduction to the field of HDR brachytherapy and CT scanning as applied to the specific problem addressed in this thesis.

1.1 High Dose Rate Brachytherapy: A Brief Overview

In order to understand the various issues involved in the design of a CT-based brachytherapy , it is helpful to have some understanding of the field of brachytherapy itself. This section gives a brief overview of the specialized field of high-dose-rate brachytherapy. Chapter 2 provides a more detailed description of HDR brachytherapy.

Brachytherapy is a form of cancer treatment where one or more sealed radioactive sources are placed in close proximity to or inside the tumor. In high-dose-rate brachytherapy specialized applicators are first strategically placed (interstitial or oth-

erwise) within the tumor or close to it. A high-activity radioactive source is then sequentially put into these applicators. The source resides for predetermined dwell-times at various dwell locations inside these applicators and irradiate the cancer cells.

Treatment Planning is an important step in HDR brachytherapy. It optimizes the dwell-times of the radioactive source at various dwell locations within the applicators. These dwell-times are based on optimum dose distributions required in the treatment procedures. In treatment planning, the dose distribution inside the tumor is estimated. This impacts on how well the tumor cells are terminated and the healthy cells spared. In order to maximize the dose to the diseased cells and minimize the dose to healthy tissue, the patient anatomy and applicator placement must first be visualized.

X-ray techniques are primarily used for determination of the location of the applicators inside or close to the target volume. Conventionally, radio-opaque markers are placed inside the applicators and two orthogonal X-ray radiographs are obtained: 1) an anterior-posterior (AP) radiograph, and 2) a lateral radiograph. The location of the applicators and various landmarks are identified from these radiographs. This information is then entered into the Treatment Planning system. The system then computes the optimum dwell-times of the radioactive source at each dwell locations within the applicators.

The method of calculating the dose to adjacent anatomical structures is limited by the inability to clearly delineate organ boundaries from the radiographs [22]. This is caused by the projection of overlapping anatomical structures (especially soft tissues) on to a single plane in traditional X-ray radiography. One way of enhancing the visualization of critical organs is to inject these organs with X-ray contrast agents. However, a complete 3-D geometric delineation of these structures is still impossible.

The 2-D nature of conventional radiographical techniques is a significant draw-

back in delineating soft tissues. In cases such as brachytherapy procedures of uterine and cervical cancers rectal applicators are often used. These applicators are much smaller in diameter than the rectum and any lateral displacement of these applicators may contribute to a significant skew in the dose estimation[9]. Radiographic imaging is very limited in their use for treatment planning in these cases.

The various limitations of radiographic imaging techniques in brachytherapy can be alleviated by the use of Computed Tomography (CT) images of the patient and the treatment applicators[4, 7, 9, 15, 22]. In tomographic[1] images the critical structures (including soft tissues) are clearly visible. Geometrical locations and 3-D shapes of critical structures and applicators are readily obtainable. This is especially important for treatment planning procedures involving radio-sensitive organs like the urinary bladder and the rectum.

CT images also speed up the treatment planning procedures since less time is spent analyzing the overlapped sections and critical structures as in the case of radiographs.

1.2 Computed Tomography: A Brief Overview

This section provides a brief overview of Computed Tomography (CT) which is one of the most popular medical imaging techniques. Chapter 2 provides a more rigorous description of the workings of a CT scanner.

Computed tomography is a well-established diagnostic imaging technique used in radiology. In CT scanning, cross-sectional images (tomograms) are obtained of the patient anatomy. The patient is irradiated with a narrow beam of X-rays. Collimated radio-detectors placed at the opposite end detect the intensity of the X-rays which are transmitted through the patient. Acquiring adjacent slices of the patient enables

[1]Thin slices of objects are known as *tomograms*

the complete 3-D reconstruction of the region of interest.

The first steps towards the eventual development of computed tomography were made possible by two events:

- The discovery of X-rays by *Röntgen*, and
- The mathematical work by *Radon* in 1917 on back-projection of 2 and 3-D data.

Radon's work was in the field of gravitational theory. *Bracewell* re-invented the back-projection technique in 1956 while working in radio-astronomy. The first works on computed tomography were done by *Olendorf* (1961) and *Cormack* (1963).

Practical considerations regarding the computational power needed for data analysis in CT scanning, or the lack thereof, prevented any further progress towards CT scanning. In 1968 *Kuhl* and *Edwards* built the first successful CT scanner for nuclear imaging. However, they did not extend their work to diagnostic radiology. The first commercial CT scanner for diagnostic purposes was developed in 1971 by *Dr.Godfrey Hounsfield*, a research scientist at EMI. It was originally built to scan the human brain.

CT scanning technology has advanced since the first EMI head scanner. They are more precise, accurate and less time consuming than before. Computed tomography is now a standard diagnostic imaging modality in most hospital radiology departments and are widely used for high-contrast-resolution visualization of anatomical structures for diagnostic purposes.

There are however potential drawbacks to the process of using CT scanning techniques in brachytherapy treatment planning. Using conventional CT scanners requires transporting the patient (with the inserted applicators) from the CT scanning room to the brachytherapy room. This can cause significant discomfort to the patient. Additionally, the applicators can move inside the patient. This may not only cause

injury to the patient but also invalidate all data collected. The moving process also adds significant amount of time to the planning procedure.

A dedicated CT scanner present at the brachytherapy room would alleviate most of these aforementioned problems. Unfortunately, the use of a conventional diagnostic CT scanner for the exclusive brachytherapy purposes is cost-prohibitive.

The CT scanner described in this thesis uses the brachytherapy ^{192}Ir isotope instead of an X-ray tube as its radio-source.

1.3 Organization of Thesis

This thesis describes the design and implementation of the data acquisition system (DAS) for a γ-ray CT scanner designed for brachytherapy treatment planning. Chapter 2 provides background information about HDR brachytherapy and CT scanning theory.

The CT scanner described in this thesis was a result of a joint group effort between the Department of Medical Physics and the Department of Electrical and Computer Engineering at the University of Manitoba. The work for this scanner took place at Cancer Care Manitoba. The members of the group involved Dr. Anita Berndt, Dr. Satyapal Rathee and Dr. Daniel Rickey from the department of Medical Physics and the author who is from the department of Electrical and Computer Engineering. The rest of the section, in addition to providing an overview of the rest of the thesis delineates the contributions of each of these individual members to different phases of the project.

Chapter 3 addresses the design considerations for the DAS. Issues and their solutions are outlined. A brief description of the different hardware and software modules is provided.

Chapter 4 discusses the design and implementation of the analog detector as-

sembly boards for the 4^{th} generation scanner. The detector boards are based on a prototype design described and characterized by Berndt et al. [2]. This prototype design was done primarily by Dr. Anita Berndt, Dr. Daniel Rickey and Dr. Satyapal Rathee at the Medical Physics department of Cancer Care Manitoba. They designed and implemented the prototype and tested it out for noise characteristics. This design was then ported on to Printed Circuit Boards (PCB's) to fit the requirements of the final CT scanner by the author. The boards were then populated and individually tested by the author.

Chapter 5 describes the design and implementation of the digital control-hardware circuitry which synchronizes the data acquisition operations of the CT scanner. The design, implementation and testing of the hardware module described in Chapter 5 was done in collaboration with Dr. Satyapal Rathee.

Chapter 6 describes the micro-controller based software written for the CT scanning system. The design and implementation of the software module described in Chapter 6 was done in joint collaboration with Dr. Daniel Rickey of Cancer Care Manitoba. Chapter 7 concludes with a look at the accomplishments and future considerations. Appendix A provides the source code used by the software module described in Chapter 6.

Section 3.3 was describes in greater details the contributions of every member of the group to the design and implementation of the Data Acquisition System.

Chapter 2

BACKGROUND

Sections 1.1 and 1.2 provided brief overviews about the fields of HDR brachytherapy and Computed Tomography. This chapter discusses these in greater details. Issues regarding the underlying theory behind CT scanning are also explained. A chronological view of the different generations of CT scanners and their data acquisition methodologies is also provided. We conclude with a look at some of the popular image reconstruction algorithms used in CT today. It should be noted that the reconstruction algorithm described in this Chapter was not used as a part of the thesis work and is included for the sake of completeness.

2.1 High Dose Rate (HDR) Brachytherapy

Brachytherapy is a method of cancer treatment in which sealed radioactive sources are used to deliver radiation at a short distance by interstitial, intracavitary, or surface application. In HDR brachytherapy, a high radiation dose can be delivered locally to the tumor. The high dose-rate isotope usually has a rapid fall-off so that the surrounding healthy tissue is spared from any damage. Technical developments such as the introduction of artificial isotopes, afterloading devices (which reduce personnel exposure) and automatic devices (which use remote control to deliver controlled radiation) have helped advance the field of brachytherapy.

In the past, brachytherapy was carried out mostly with radium or radon sources. However, the use of artificially produced radionuclides such as ^{137}Cs, ^{192}Ir, ^{198}Au and ^{125}I have increasingly become common[11, 14, 19, 23]. The *Nucletron HDR microSelector Afterloader* brachytherapy instrument used during the course of this thesis project uses a ^{192}Ir source. The following discussion concentrates on the characteristics of ^{192}Ir as a radioactive source.

The choice of the radioactive source is an intimate part of brachytherapy. It depends on issues such as its half-life, photon energy, half-value layer rating and exposure rate constant.

The half-life of a source is the time required by it to decay to half its original strength. An effective radioactive source has a half-life which is comparable to the average treatment time. ^{192}Ir which has a half-life of 74.2 days is ideal for the non-permanent transplants used in High Dose Rate brachytherapy.

The photon energy of a radioactive source is the average energy of radiated photons. The ^{192}Ir has a complicated γ-ray spectrum with an average energy of 0.38 MeV (Mega Electron Volts). Because of the lower energy, ^{192}Ir sources require less shielding for personnel protection. However, its wide polychromatic spectrum ($0.36 - 1.06$ MeV's) is a disadvantage when using it as an imaging source. Ideal imaging sources should have a very narrow energy spectrum.

The half-value layer of a source is the thickness of lead required to halve the radiation dose of the source. ^{192}Ir has relatively low half-value layer of $2.5mm$.

Calibration of the radioactive sources is a major concern in brachytherapy. There are two essential ingredients in the calibration process:

- specification of source strength, and
- exposure rate calibration

Source strengths are specified in terms of:

1. source activity (expressed in terms of Ci or curies)
2. exposure rate at a specified distance (expressed at a standard distance of 1m)
3. apparent activity. This is the activity of a "bare" source which has the same exposure rate at 1 meter as the sealed nuclide under test.

Various techniques exist for the calibration of the exposure rate of the different sources used in brachytherapy. The National Institute of Standards and Technology (NIST) has established calibration standards for some of the the more popular brachytherapy sources, like ^{226}Ra, ^{60}Co, ^{137}Cs and ^{192}Ir. One of the calibration techniques in use is the "open air measurements" technique, in which a relatively large ion chamber is placed at a large distance from the source and the exposure rate is measured. The large distance is maintained so as to minimize any scatter radiation.

One of the main components in brachytherapy is treatment planning. Treatment planning determines the strength and spectral distribution of sources. This is done to optimize the dose distribution and provides a complete dose distribution in the irradiated volume. Numerous systems have been devised over the past fifty years to implement dosimetric planning in brachytherapy. Of these, the most extensive and widely used are the Paterson-Parker and the Quimby systems[10, 16]. Extensive tables and elaborate rules of source distribution have been devised to facilitate the process of manual treatment planning. The advent of digital computers has led to significant developments in treatment planning schemes as they allow the therapist to deviate from the established systems and help treatment planning cater to the anatomical peculiarities of every individual patient.

2.1.1 Computer Dosimetry

Dosimetry is the process of pre-computing the radiation doses that a patient is to be exposed during a radiotherapy session. In brachytherapy, dosimetry determines

the dwell locations and the dwell times of a radioisotope inside an applicator, based on the radiation dose characteristics of the isotope[1].

Older manual dosimetric schemes are based on idealized implants conforming to certain distribution rules[1]. In actual practice such ideal distributions are seldom realized. With powerful computers, it is possible to pre-plan not only implants but a complete isodose distribution, corresponding to the final source distribution. The rapid turnaround time with the modern computer systems allows the therapist to modify the implant, if necessary, on the basis of three-dimensional distribution.

Computation of a brachytherapy dose distribution consists of repeated calculation of dose at certain given points for each of the implant sources. The total dose is determined by summing the individual source contributions to that particular point. Point dose rates are computed for each of a grid of points arranged in a cubic lattice so that isodose curves may be generated in any arbitrary plane. These isodose patterns can be then magnified and superimposed on an implant radiograph for viewing the distribution in relation to the patient's anatomy.

One of the very important steps in this process is the visualization of these implants source in 3-D plane[5, 17]. These inputs act as a very important parameter for the computers to calculate the dose distribution in a given anatomy.

The conventional method of visualization of the implant sources is the orthogonal radiograph imaging method. In this method, two radiographs are taken at right angle, with central axes of the X-ray beam meeting approximately in the middle of the implant. Usually, an AP and a lateral radiograph, exposed isocentrically, provide such a geometry. The coordinate system is conventionally established with the X-axis from the right to the left of the patient, the Y-axis from inferior to superior, and the Z-axis from posterior to anterior. The AP film represents a magnified view of the implant image projected onto the X-Y plane while the lateral film presents the image

projected onto the Z-Y plane. The origin of the coordinate system is chosen to be point identified to be the same on both films such as one end of a source.

After all the sources are identified, the tip and end of each linear source image sampled with a digitizer, sampler each end sequentially from one film to the other. This method is an inherently fault-prone method because of the interpolation of 3-D data from 2-D radiographs[9]. An imaging modality involving CT scanning would intuitively seem to provide a better access to more accurate visualization of the sources[7, 9, 15].

2.2 Computed Tomography

Computed tomography is the most commonly used radiological imaging modality of precise, accurate and quick visualization of human anatomy in 2-D sections. Classical X-ray techniques produce a photographic recording of the 2-D projection of a 3-D object, projected by the radiation cone onto the image plane. As a result, overlapping anatomical structures projected on to a single plane make it difficult to visualize soft tissues. Computed tomography avoids this superposition effect by processing information on a particular 2-D slice.

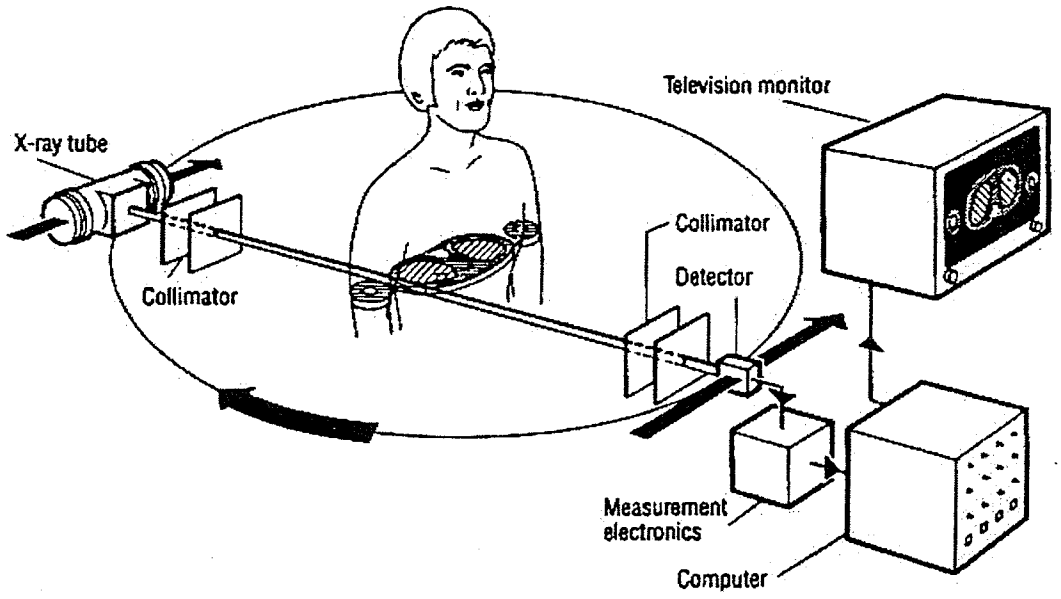

FIG. 2.1. Basic representation of a computed tomograph in the simplest design form. The diagram shows the basic units: the X-ray tube, collimators, detector, control hardware and host computer[18].

The important components of a CT scanner are shown in the form of a block diagram in Figure 2.1. A pencil beam of X-ray photons is provided by an X-ray tube. As the beam passes through the patient, it gets attenuated. A radiation detector measures the intensity of the beam as it emerges from the patient, thereby providing a measure of attenuation by the patient along the beam path. A large number of such attenuation measurements are obtained in order to reconstruct the final image of the 2-D slice.

The simplest form of the measurement system uses a collimated X-ray beam about the thickness of a pencil passing through the slice for measuring the attenuation[12]. This is very similar to the system implemented by Hounsfield in the very first EMI head scanner. The attenuation is determined by moving the measurement system perpendicular to this beam in the slice plane across the entire object. During this transverse motion the radiation intensity at the detector is recorded at fixed

spatial intervals. This set of measurements is called a projection. A large number of such projections, at different angles, are required to completely reconstruct the image of the object. After one projection is measured, the measurement system is rotated through a small angle (usually 1°) about the system axis, perpendicular to the tomogram plane and another linear scanning process is initiated. This goes on until the measurement system has traversed a total angle of rotation of at least 180°[12].

As the projection data continues to be measured for each fixed orientation of the X-ray tube, the measured data is digitized in a suitable form and transferred to a computer. Using the digitized data, the computer then calculates a 2-D attenuation distribution of the object. The digitized values are first normalized. The objects with high attenuation are assigned high numerical values, and those with low attenuation are assigned low numerical values[12].

Since the computer can only calculate a finite number of attenuation values, the image slice must be thought of as consisting of small discrete volume elements (voxels) with a square base. The attenuation value is assumed to be constant inside each voxel. Gray levels are assigned to the different attenuation values, and then displayed on a monitor as a 2-D image.

Ideally, the linear attenuation coefficient calculations by a CT scanner should be accurate both in the absolute as well as in the relative sense. In diagnostic radiology, it is much more important to use the relative attenuation characteristics. The CT computer computes a relative linear attenuation coefficient for each voxel. It calculates integer attenuation values relative to the attenuation of water for comparison purposes. This number is known as the *CT_Number* and can be mathematically termed as[12]:

$$CT\_Number = K \cdot \frac{\mu_p - \mu_w}{\mu_w} \tag{2.1}$$

where,

K is the magnification constant,

μ_p is the linear attenuation coefficient of the tissue in voxels, and

μ_w is the linear attenuation coefficient of water.

The normalized attenuation constants for the densest body tissue (compact bone) and the least dense body tissue (air) are assigned the value of $+1$ and -1 respectively. The magnifying constant K is $1,000$, resulting in the CT number for dense bone being $+1000$ and that for air being -1000. The CT number for water, irrespective of the magnification constant, is always 0[12].

When viewing on the console, the whole range of the obtained CT numbers cannot be displayed simultaneously because monitors can display up to a maximum of 256 gray levels. Windowing and level controls are introduced to take care of this difficulty. Not all tomograms have values ranging from $+1000$ to -1000 in their images. Therefore, only a small "window" of CT numbers (corresponding to the usual CT numbers present in particular part of the anatomy), centered at some CT number levels, is chosen, and displayed in gray levels on the monitor.

2.2.1 CT Scanner Generations

This section describes the various imaging modalities which have existed and exist in CT scanner in a chronological fashion. Even though the basic idea behind CT imaging and reconstruction has remained unchanged over the years, significant progress has been made at faster data acquisition and image reconstruction techniques. These progresses mark the different generations of CT scanners.

First Generation The original EMI brain scanner implemented by Hounsfield in 1971 belongs to the 1^{st} generation of CT scanners. It consists of an X-ray tube (or any alternate radiation source) and one NaI (sodium iodide) scintillation detector crystal per slice (Figure 2.2)[12]. Each scanning cycle consists of a synchronized

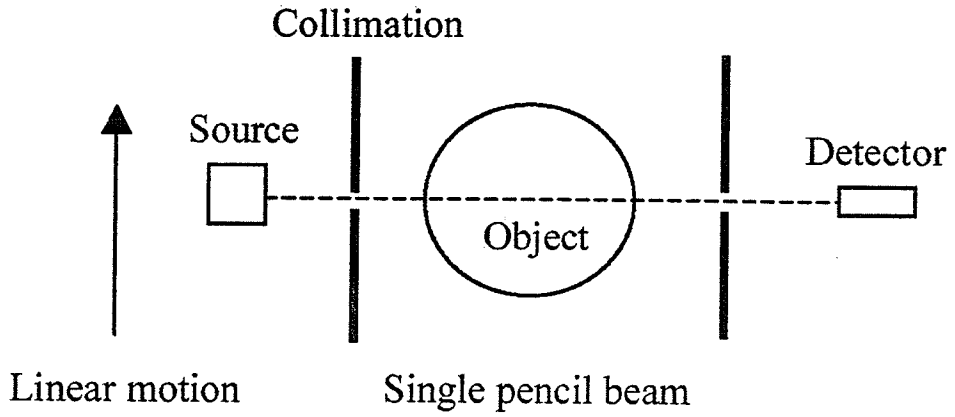

FIG. 2.2. Scanning geometry: 1^{st} generation CT scanner schematics. Highly collimated single-source-and-detector combination. Translational and rotational motions necessary

linear translation of the X-ray source and detector, sampling about 200 data points per translation. The collection of these data points forms a parallel projection. The linear translation is followed by an angular displacement of about 1^o, followed by a linear translation cycle. These projections are acquired at equally spaced angular displacements spanning 180^o. Practically, an overscan of about 45^o may be used to reduce the effects of any artifacts such as patient movement.

One of the advantages of the 1^{st} generation CT scanners is that the detector translation past the patient makes it possible to calibrate the detector during a data acquisition sequence[12]. Pre-patient X-ray source and post-patient detector collimators virtually eliminate the effects of scatter radiation. A total control on the step sizes in both linear and angular sampling ensures good control of the spatial resolution of the desired image.

Unfortunately, the repetitive linear scanning motions result in long scan times (about 4.5 minutes for 180 projections). Therefore, patient motion may degrade the image by introducing artifacts. First generation scanners are a thing of the past in

diagnostic radiology.

Second Generation The major disadvantage of the 1^{st} generation scanner geometry is the prohibitively long scan times. There are three possible ways of reducing the scan times. They are:

- The X-ray output could be increased, thereby performing a faster linear translation while still maintaining a reasonable number of photons per reading. The X-ray source in most applications however always run on full capacity.
- An increase in the beam dimensions can lead to increase in the number of photons and thus leading to faster linear translation. This however leads to decrease in spatial resolution. Also this leads to an increase in the thickness of the tomograms, which is unfavorable for certain clinical investigations.
- The number of detectors can be increased.

The last option enables us to simultaneously acquire data at different angles. This is the basis of the 2^{nd} generation scanners. Figure 2.3 shows the geometry of a 2^{nd} generation scanner. In second generation scanners, the single detector is replaced by a large number of detectors. These detectors are equally spaced along an arc centered at the X-ray tube[12].

Each linear translation of the X-ray tube-detector set-up measures projection at a number of angles determined by the number of detectors in the system. A rotation of the X-ray tube-detector set-up is still required to completely measure the projection over 180^{o}. However, the number of rotations and linear translations is significantly reduced.

The number of detectors are usually about 30. The number of angular rotations are reduce from 180 to 6. The scan time is thus reduced to about $5 - 20$ seconds.

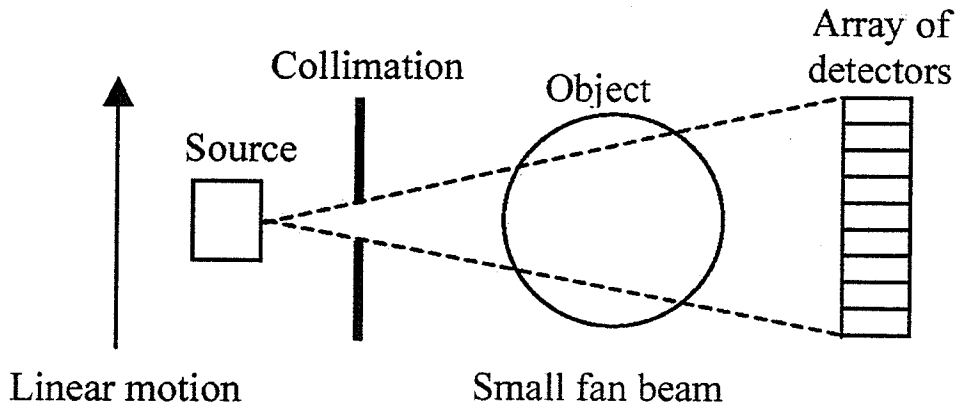

FIG. 2.3. Scanning geometry: 2^{nd} generation CT scanner. Source collimated for fan-beam. Array of detectors separately collimated. Translational and rotational motions necessary.

Second generation scanners commonly used a combination of NaI scintillation crystals and photo-multiplier tubes as the detector assembly.

The pencil beam of the first generation scanners is replaced by a fan beam in the second generation to allow simultaneous irradiation of all the detectors in a linear motion. This causes irradiation of a larger section of the patient. This results in significantly larger scattered radiation. Therefore, the second generation scanners require a substantially improved collimator system to reduce the effects of scattered radiation. While the number of rotations are decreased in the second generation, there is still the requirement of both translation as well as rotation. This geometry is not suitable for ultra-fast scan times.

Third Generation The linear motion still in use in the 2^{nd} generation scanner geometry can be eliminated if the number of detectors can be increased such that the fan formed by the detectors completely encompasses the patient being scanned. The main advantage of this process is that the translation motion is no longer required. The source and detector gantry rotates synchronously about the patient[12]. The

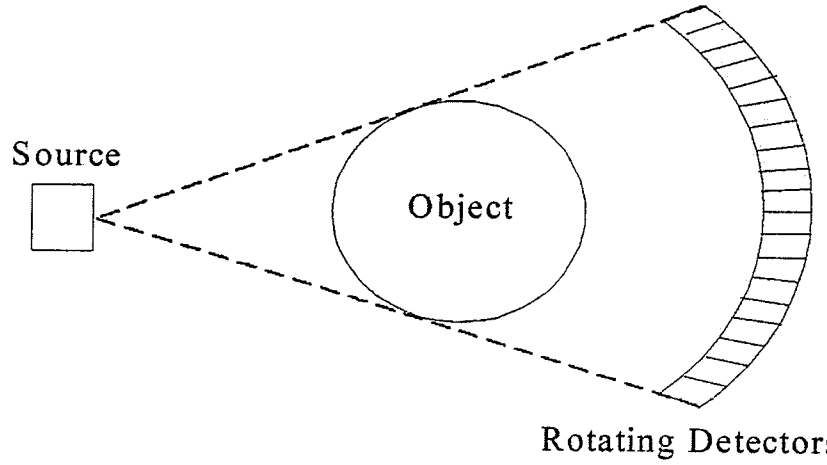

FIG. 2.4. Scanning geometry: 3^{rd} generation CT scanner. Array of detectors collimated. Only rotational motion necessary.

scanning gantry does not stop between readings. The rotary movement of the X-ray tube is continuous. The reconstruction programs are designed to compensate for this movement. Scan times in third generation scanners are between 1 and 10 seconds. Most commercial diagnostic CT scanners available today are 3^{rd} generation scanners. Figure 2.4 shows the schematics of a 3^{rd} generation CT scanner.

As the number of radial sampling positions are the same as the number of detectors, a large number of detectors ($\geq$ 300) are required to achieve a reasonable spatial resolution. NaI scintillation crystals and photomultiplier tubes are not suitable for this geometry due to their large sizes. The detector of choice is an array of high-pressure xenon gas chambers.

There are several advantages of the 3^{rd} generation CT scanners. The long septa (divider) between the xenon gas detector cells provide very good collimation and are good for scatter rejection. Also, the radiation completely encompasses the patient. This causes better X-ray utilization than the 2^{nd} generation scanners.

A major disadvantage of the 3^{rd} generation scanners is that the lack of translation makes it impossible to obtain dynamic detector calibration. In other words, detector

calibration cannot be achieved on the fly as in the previous generations. Detector stability therefore becomes an important issue. Errors introduced by a faulty detector show up in scans as ring artifacts.

Another issue in 3^{rd} generation scanners is that the number of ray samples collected per projection equals the number of detectors in the array. Therefore the resolution is achieved at the level of detector widths. It is however desirable to have twice that many readings to sample at a desirable rate. This is achieved by scanning a complete 360^o and offsetting the detector array in such a way that the samples taken 180^o apart are interleaved.

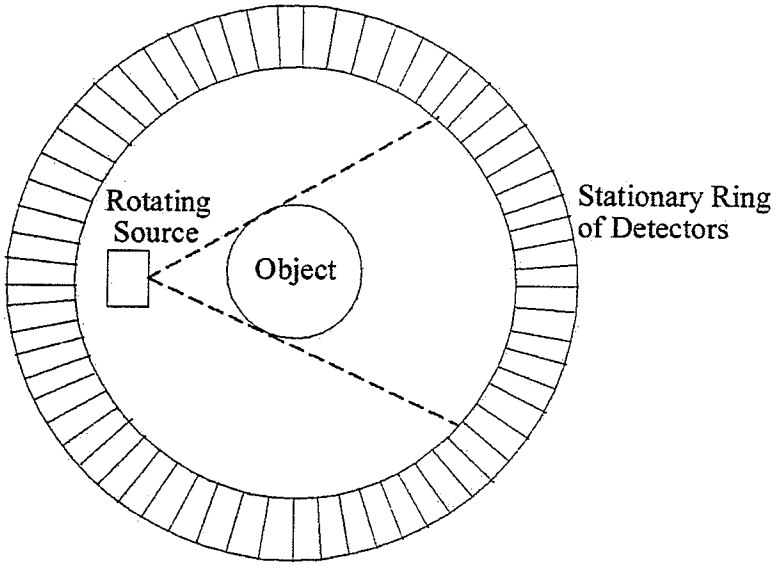

FIG. 2.5. Scanning geometry: 4^{th} generation CT scanner. Rotating source and stationary detectors. Detector ring collimated.

Fourth Generation The CT scanner described in this thesis has 4^{th} generation scanning geometry. In a 4^{th} generation scanner, the patient is completely surrounded by a circular ring of detectors which remain stationary during data acquisition[3]. This is shown in Figure 2.5. Only the X-ray tube and collimators rotate.

The scan times in this geometry, in principle, do not differ from the 3^{rd} generation scanners. However, it does not suffer from the circular artifact and undersampling problems experienced by 3^{rd} generation scanners.

In 3^{rd} generation scanners the radial sampling is determined by the number of detectors. However there is no limitation on the angular sampling. Readings can be achieved in as many angles as desired. This situation is reversed in 4^{th} generation scanning geometries.

The detectors in the ring fix the number of angular sampling positions. The radial sampling becomes unlimited as readings of a detector can be taken at arbitrary positions of the X-ray tube. The X-ray tube, moving continuously around the patient, provides unlimited radial sampling.

A drawback of the 4^{th} generation scanning geometry is that it cannot achieve dynamic detector calibration[3]. Also the non-stationary relationship between the X-ray tube and the detectors make collimation an extremely difficult task. The problem of detector cross-talk is much more pronounced in 4^{th} generation scanners than in any other generation. The mechanical arrangement of the detectors and the X-ray tube so that they do not get in the way of each other is also a cumbersome task.

Fifth Generation The massive size of the scanning gantry in CT scanners makes it unlikely that scan times by conventional radiography can be reduced to sub-second levels. The mechanical nature of the motions makes it very difficult to achieve millisecond scans.

Fifth generation scanners no longer use moving parts and are therefore able to achieve millisecond scan times[3]. Figure 2.6 shows the schematics of the 5^{th} generation scanning geometry. The design principle is the same as 4^{th} generation scanners. However, instead of a moving X-ray source, there is a giant X-ray tube

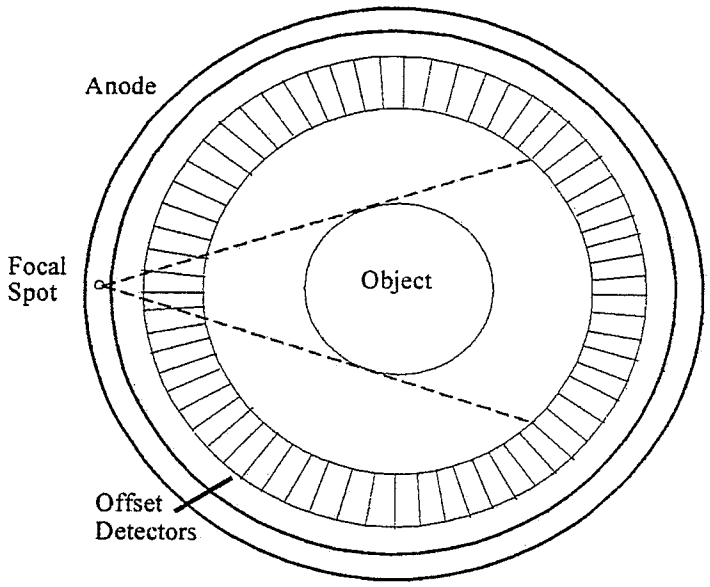

FIG. 2.6. Scanning geometry: 5^{th} generation CT scanner. Source consists of moving electronic beam inside anode. Detectors are stationary and collimated. No mechanical motions.

which completely encompasses the patient. An electronic deflecting system inside the X-ray tube moves the electron beam around the anode circle in the X-ray tube. This simulates a moving radiation source[3].

Fifth generation scanners largely exist in places of research at the present[1].

2.3 Image Reconstruction Techniques

In order to form an image from the projection data, the data has to be reconstructed. There are a number of demands placed on the reconstruction algorithm. It needs to be computationally fast and relatively tolerant of noise generated by the data acquisition electronics and inherent photon statistics. The reconstruction algorithms can be broadly categorized under four headings:

1. Iteration-based *algebraic reconstruction techniques* (ART's),

[1]The *Mayo Research Foundation* does use a 5^{th} generation scanner for clinical purposes

2. Reconstruction in the spatial-frequency[2] domain,

3. Hankel transform method[8], and

4. Filtered backprojection in the spatial domain

Iterative reconstruction techniques such as ART[6] are based on hit and trial estimation of the slice pixel values. A sequence of steps is performed that provides better and better estimates of the unknown image until it converges on the correct answer. One begins by assuming almost any values for the image. Each ray sum is compared with that of the image and the error is the difference. The error is then smeared uniformly across the ray path. This process is repeated until every the ray sum for the image conforms to all the obtained ray sums.

The ART technique was used in the first EMI scanner but not after that. Its major difficulties are computational complexity and low accuracy. In presence of noisy data, there is a possibility of the technique never converging. The ART technique however does not require a complete set of data for reconstruction since each ray sum is used to just improve the accuracy of the image.

The *Fourier transform*-based spatial-frequency domain algorithm is based on Fourier central slice theorem. This theorem states that the Fourier transform of a 2-D function along an angle in the 2-D frequency space, is the same as the 1-D Fourier transform of the projection of the function taken at the same angle as the slice.

A 2-D spatial Fourier transform can thus be obtained by taking the 1-D Fourier transform of the projections and arranging them in order of the angles at which the projection has been originally acquired[12, 21]. An inverse 2-D Fourier transform of this distribution results in the desired 2-D tomogram image. The discrete nature of the projection data however poses a practical limitation to this approach. The points

[2]The fourier transform domain for time is *frequency*. The fourier transform domain for space is therefore termed *spatial-frequency*

acquired in the spatial-frequency domain are unevenly distributed (this is caused by the circular scanning geometries). Therefore the number of samples per unit area decrease with an increasing distance from the origin. This poses the need for compensation routines to make up for the uneven resolution.

The *Hankel transform technique* also employs the polar-coordinate based Fourier techniques [8, 12]. It employs the Fourier transform for the whole image set and generates the Fourier values in the polar coordinates. Usually FFT is employed[8]. The appropriate Hankel transform (eqn.2.2) for each F_n term is evaluated. The Fourier series is resolved and the values are then interpolated to the rectilinear grid. The advantage of this transform that using this a subset of the Fourier transforms can be employed. This facilitates the trade-off between considering too few Fourier coefficients for poor resolution and too many Fourier coefficients for computational complexity and high frequency noise.

$$f_n(\rho) = \frac{1}{2\pi} \cdot \int_0^\infty r F_n(r) \cdot J_n(r\rho) \cdot dr \tag{2.2}$$

where,

F_n is the n_{th} Fourier coefficient, and

J_n is the n_{th}-order Bessel function of the first kind.

The filtered backprojection technique is one of the most popular image reconstruction techniques used today and is discussed in greater details in Section 2.3.1. Section 2.3.1 provides the backprojection technique for a parallel beam geometry and section 2.3.2 discusses the rebinning algorithm which converts the radial projection data into a parallel beam format.

2.3.1 Filtered Backprojection Reconstruction Technique

The filtered backprojection technique is one of the standard reconstruction techniques used in CT scanners[20]. The algorithm mentioned in this section presumes the existence of parallel beam scanning geometry. The description in this section is based largely on the treatment of this subject in [12]. In order to use this technique for fan beam geometries such as the 3^{rd} and 4^{th} generation CT scanners, a rebinning technique (Section 2.3.2) has to be applied first in order to translate the fan beam geometry to a parallel beam geometry.

Figure 2.7 shows the coordinate systems involved in the backprojection algorithm. Let (x, y) represent the coordinates inside the object and $f(x, y)$ be the density in cartesian coordinates of the object under consideration. Let eqn.2.3 represent the projection at a distance t from the origin. t can be represented as $(x \cdot \cos\theta + y \cdot \sin\theta)$, where x and y belong to the object coordinate system and θ is the angle which the normal to the projection makes with the object coordinates. The projections $P_\theta(t)$ are thus be defined as:

$$P_\theta(t) = \int_{(\theta,t)line} f(x, y)ds \tag{2.3}$$

Eqn.2.3 means that if we integrate along s, the summations of $f(x, y)$ along the path form the projections. It should be noted at this point that we consider (t, s) system as the projection coordinate system. The projections can also be denoted in the projection coordinate system as:

$$P_\theta(t) = \int_{-\infty}^{\infty} f(t, s)ds \tag{2.4}$$

We can prove that the Fourier transform of the object in (x, y) coordinate system is interchangeable with the Fourier transform of the projection data in the (t, s) coordinate systems. In the (x, y) system, the Fourier transform of the object is

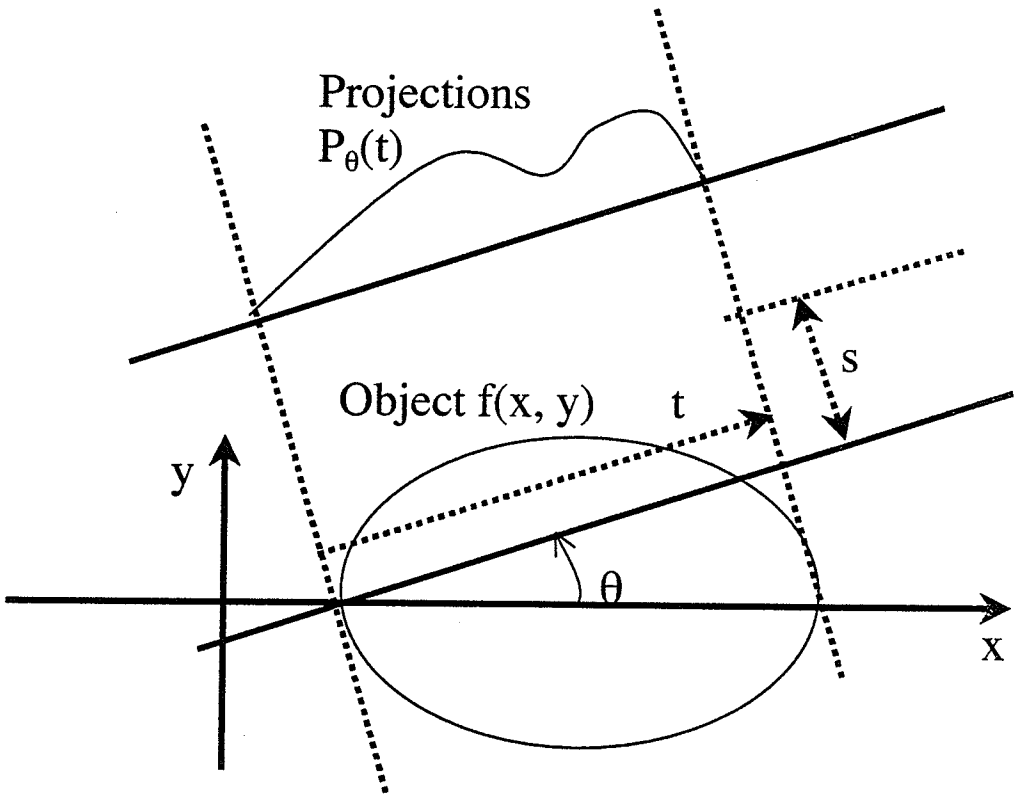

FIG. 2.7. Formation of back-projections and the different coordinate systems they follow

formulated as:

$$F(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\cdot e^{-j2\pi(ux+vy)}dxdy \tag{2.5}$$

where, (u,v) is the coordinate system in the Fourier domain.

The Fourier transform of the projection data (from eqn.2.4) is formulated as:

$$S_\theta(\omega) = \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(t,s)\cdot e^{-j2\pi\omega t}ds\right]dt \tag{2.6}$$

In eqn.2.6, the Fourier domain for the projection data is taken to be the polar

(ω, θ) coordinate system. Using the transformations $(t = x\cos\theta + x\sin\theta)$ and $(s = -x\sin\theta + y\cos\theta)$, eqn.2.6 can be reformulated as:

$$S_\theta(\omega) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi\omega t}dxdy \tag{2.7}$$

The right hand side of eqn.2.7 represents the two-dimensional Fourier transform of the density $f(x,y)$ and the left hand side is the $1D$ Fourier transform of the projections. Therefore, taking the $1D$ Fourier transform of the projections of an object at an angle θ is equivalent to obtaining the two dimensional Fourier transform of the density $f(x,y)$ along the line t inclined at an angle θ. This is the crux of the Central Slice Theorem. If we take these projections at many angles, then we can get this $2D$ Fourier transform of the projections at many such lines inclined at various angles. If the number of angles is large enough, we get many lines of the 2D Fourier transforms of the object. If we now find the inverse Fourier transform of all these lines, we can obtain the object's densities $f(x,y)$ for all the (x,y) in the object's cross section. That is,

$$f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} S_\theta(\omega)e^{j2\pi(ux+vy)}dudv \tag{2.8}$$

where $u = \omega\cos\theta$, $v = \omega\sin\theta$ and $S_\theta(\omega) = F(\omega\cos\theta, \omega\sin\theta) = F(u,v)$.

Converting the cartesian (u,v) coordinate system into the polar (ω,θ) coordinate system, eqn,2.8 can be re-written as:

$$f(x,y) = \int_0^{2\pi}\int_{-\infty}^{\infty} F(\omega,\theta)e^{j2\pi\omega(x\cos\theta+y\sin\theta)}\omega d\omega d\theta \tag{2.9}$$

$$= \int_0^{\pi}\int_{-\infty}^{\infty} F(\omega,\theta)e^{j2\pi\omega(x\cos\theta+y\sin\theta)}\omega d\omega d\theta + \tag{2.10}$$

$$\int_{\pi}^{2\pi}\int_{-\infty}^{\infty} F(\omega,\theta+\pi)e^{j2\pi\omega(x\cos(\theta+\pi)+y\sin(\theta+\pi))}\omega d\omega d\theta \tag{2.11}$$

$$= \int_0^{\pi}\int_{-\infty}^{\infty} \left[F(\omega,\theta) \mid \omega \mid e^{j2\pi\omega(x\cos\theta+y\sin\theta)}d\omega\right] d\theta \tag{2.12}$$

In eqn.2.12 the terms inside the square brackets (the operation indicated by the inner integral) represents the filtering operation and evaluate the filtered projections, and the operation being performed by the outer integral evaluate the back-projections, which basically represents a smearing of the filtered projections back on to the object and then finding the mean over all angles.

The filtered back projection algorithm can therefore be thought of as a three step process:

1. Find the Fourier transform in $1D$ of the projections.

2. Find the filtered projections. This essentially means multiplying the results of step 1 with the filtering function in the frequency domain and finding its inverse Fourier transform.

3. Finding the back projections. This step is the smearing of the filtered projections back on to the object.

2.3.2 The Rebinning Algorithm

High-speed data gathering methodologies employed in the 3^{rd} and 4^{th} generations usually require fan-beam geometries. Therefore, the projections acquired are no long parallel beam geometries. Algorithms such the filtered backprojection reconstruction algorithm described in Section 2.3.1 are no longer sufficient for image reconstruction.

One of the ways of solving this problem is to implement a rebinning algorithm which converts the fan-beam data into a parallel beam format. Figure 2.8 shows correspondence between the fan beam and the parallel beam geometries. In Figure 2.8, ϕ_F is the angle of the central ray of the beam, and x_F is the curvilinear distance along the detector ray. The source defines the center of the array. The relationship between the fan beam $[x_F, \phi_F]$ coordinate system and the parallel beam $[x_r, \phi]$ coordinated

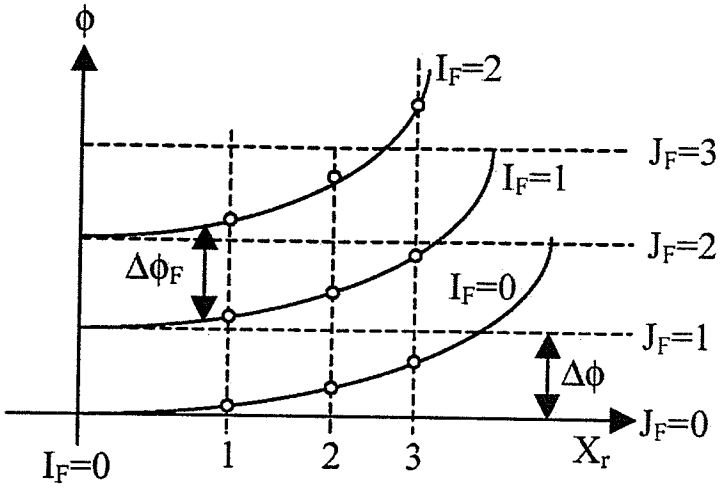

FIG. 2.8. Correspondence between actual data coordinates and the intermediate coordinates. The hollow points are original data and their projections on x-axis are the intermediate points.

system is given by:

$$x_r = R \cdot \sin(x_F/2R) \tag{2.13}$$

$$\phi = \phi_F + x_F/2R \tag{2.14}$$

where, $2R$ is the source-to-detector distance.

Fan beam projection data is represented by the array $\tau_F[i_F, j_F]$, where the detector position $x_F(i_F)$ and the central ray angle $\phi_F(j_F)$ is given by:

$$x_F(i_F) = i_F \Delta x_F \tag{2.15}$$

$$\phi_F(j_F) = j_F \Delta \phi_F \tag{2.16}$$

The parallel beam data-set is recorded in the $\tau[i, j]$ coordinate system. where i

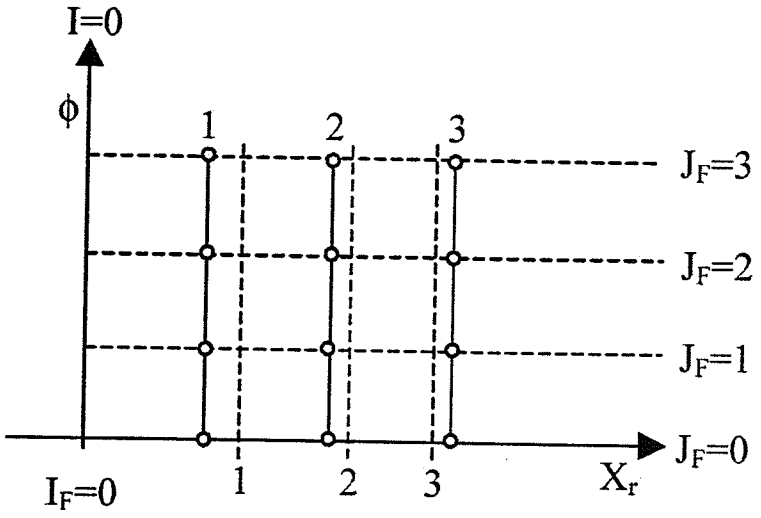

FIG. 2.9. Correspondence between intermediate coordinate points and homogeneous rectilinear grid points. The hollow points are regular rectilinear grid points.

and j specify the detector position $x_r(i)$ and the projection angle $\phi(j)$ according to:

$$x_r(i) = i\Delta x_r \tag{2.17}$$

$$\phi(j) = j\Delta\phi \tag{2.18}$$

Recording data into the parallel beam geometry is done by two stages of interpolation. The first stage consists of generation of an intermediate array $\tau'[i_F, j]$, which is obtained by holding i_F constant and performing a 1-D interpolation over j_F. Figure 2.9 shows the intermediate coordinate system. $\tau'[i_F, j]$ corresponds to the parallel beam data obtained at uniform angular intervals but with non-uniform spaced detectors. This is denoted by:

$$\tau'[i_F, j] = \tilde{\tau_F}(i_F, j_F^*) \tag{2.19}$$

where $\tilde{\tau_F}$ denotes the result of the interpolation of $\tau'[i_F, j_F]$ to non-integer valued j_F^*

given by:

$$j_F^* = j\Delta\phi/\Delta\phi_F - i\Delta x_F/(2R\Delta\phi_F) \tag{2.20}$$

The second interpolation is to correct for non-uniform spacing by holding j constant. Figure 2.9 shows the implementation of the second interpolation:

$$\tau(i,j) = \tilde{\tau}'(i_F^*, j) \tag{2.21}$$

where, the interpolated value $\tilde{\tau_F}'$ of τ' is evaluated at:

$$i_F^* = (2R/\Delta x_F)\arcsin(i\Delta x_F/R) \tag{2.22}$$

Chapter 3

PROBLEM OVERVIEW

This chapter discusses the issues involved in the design and implementation of the 4^{th} generation CT scanner that was implemented in this thesis project. The three chief design modules of the data acquisition system (DAS) are introduced. These are:

- The analog detector assembly,
- The digital control hardware, and
- The software control system.

Chapters 4, 5 and 6 provide a more detailed description of these individual modules.

As discussed in Section 1.1, the use of orthogonal radiographs for visualization of patient anatomy and treatment applicators is inherently error-prone. A CT scanner which uses the brachytherapy isotope is implemented. This helps obtain 3-D images of the patient anatomy.

In many respects, the *Nucletron Afterloader* at Cancer Care Manitoba provides a good basis for implementing a 4^{th} generation CT scanner. The $7.5Ci$ ^{192}Ir source used by the HDR brachytherapy machine has several qualities suitable for use as an imaging source for a CT scanner. It is small enough ($0.35cm$) to achieve a reasonable spatial resolution in the resulting CT image. The high activity from the $7.5Ci$ source

helps in keeping scan times short. The high energy γ-rays emitted by the source makes it possible to image patients with large girths without excessively increasing the noise (or conversely, the scan times) in the reconstructed CT images. Moreover, the source moves under the HDR unit's computer control and no extra means of mechanical movement of the source is required. Because the source moves through a thin $2mm$-diameter-wide catheter, the size of the gantry can be quite small. These properties make it very suitable for use in the project.

The *Nucletron* HDR brachytherapy machine was not manufactured with imaging issues (such as image quality and scan-times) in mind. Therefore, there are some limitations posed by the machine. These had an impact on the design of the CT scanner.

There are certain drawbacks to the use of the HDR machine. The orientation of the source is not ideal for imaging. The source must travel lengthwise $(3 - 5mm)$ in the imaging plane. The scan times (approximately $2min$) also end up being longer than conventional X-ray CT scanners.

Figure 3.1 provides a cross-sectional view of the CT scanner. It shows the gantry (which houses the detectors, the source ring and the collimators among other things), the patient support system and the necessary mechanical support systems for the gantry. The usable circle inside the collimator ring (the bore diameter) is $50cm$. The external circle of the collimator ring (the detector ring diameter) is $77.6cm$.

The γ-ray detectors are mounted to a lead collimator consisting of two lead discs. These discs have an internal diameter of $53cm$ and an external diameter of $77.6cm$. The discs are separated by $0.8cm$. Adjustable collimator jaws, located on the inner diameter of the scanner, reduce the amount of spurious radiation to the patient.

The scanner size is largely determined by the maximum diameter of the source ring which is $68.8cm$. The minimum diameter is constrained by the size of the patient

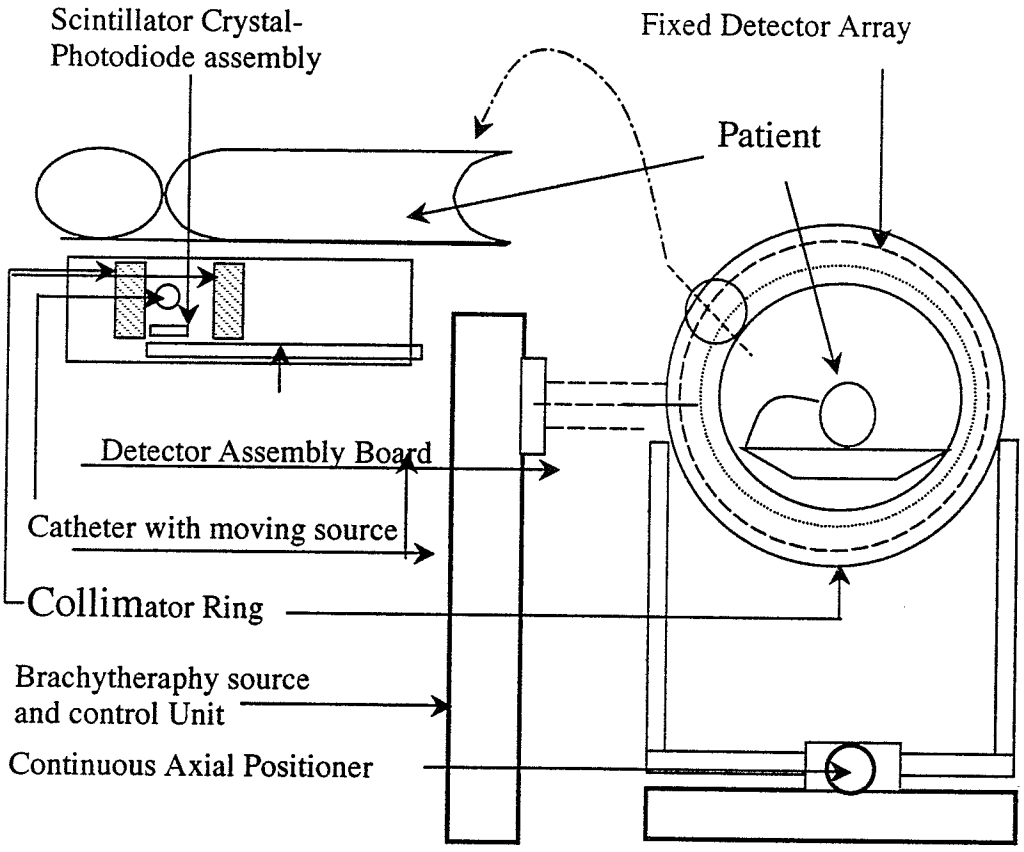

FIG. 3.1. Schematics for the scanner gantry showing a cross-sectional view of the collimator ring, the patient support system and the mechanical support system.

table as well as that of the patient. Ideally, the bore diameter should be larger than $50cm$. But, this was prevented by the minimum acceptable resolution, and the constraint on the source ring diameter. Also, the need to collimate the slice thickness affected the size of the bore diameter. Increasing the bore diameter without increasing the source ring diameter results in larger slice thickness.

Catheters are located between the lead discs and reside just in front of the scintillator crystals which act as the detectors. These catheters bear the γ-ray source during imaging.

3.1 System Modules

Figure 3.2 shows the schematics of the major electronic components in the CT scanner. The scanner electronics comprise of the following sections:

1. The HDR brachytherapy instrument, which provides and controls the γ-ray source.

2. The 96 eight-channel γ-ray detector assemblies, which detect the γ-rays striking the channel and convert them into corresponding electrical signals.

3. The 18 γ-ray source position sensors which keep track of the γ-ray source at the beginning position of each catheter.

4. The 96 : 1 analog multiplexer, which multiplexes the analog signals from the 96 detector assemblies to a single channel.

5. The analog-to-digital (A/D) converter, which digitizes the analog signal for storage and processing.

6. The digital computer, which stores the digitized data and performs rebinning and filtered backprojection to generate the image.

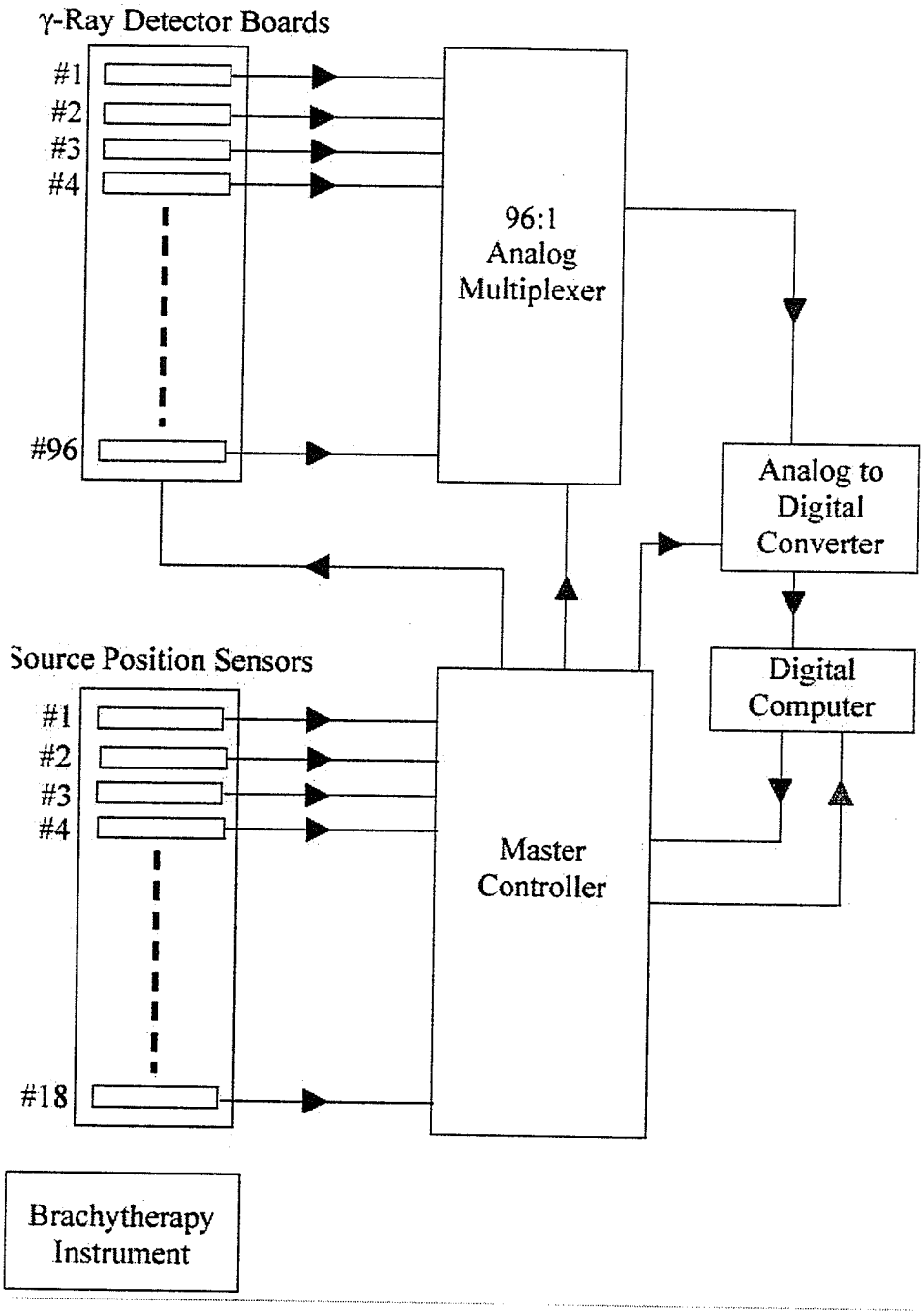

FIG. 3.2. Schematics for the CT scanner electronics. Note that there is no electrical connectivity between the brachytherapy machine and the CT scanner.

7. The master controller, which synchronizes the operations of the γ-ray detector assemblies, the 96 : 1 analog multiplexer and the A/D converter with the motion of the γ-ray source.

3.1.1 The HDR Brachytherapy Machine

An understanding of the *Nucletron Afterloader* HDR machine is important in order to make design decisions for the scanner. The machine itself cannot be modified in any way. Moreover, there can be no electronic connection between the HDR instrument and the rest of the CT scanner.

The *Afterloader* provides the means to move the radioactive γ-ray source at the end of a steel cable. It uses an ^{192}Ir source with a strength between 5 and 10 Curies. It uses a $0.35cm$ long source with a diameter of around $0.06cm$, and is mounted at the end of a $200cm$ long metal cable.

The γ-ray source is automatically positioned anywhere within any one of the 18 catheters connected to the machine and moves at relatively high speed between dwell locations. Each catheter can be a maximum of $150cm$ long.

Due to restrictions imposed by the HDR machine, the source can dwell at a maximum of 48 positions inside each catheter. Additionally, each catheter can be accessed only once during a "treatment" program. The minimum source increment required for desirable spatial resolution in the CT image is $0.25cm$. This means that only $12cm$ of each catheter can be effectively used. This provides us with a ring diameter of $68.8cm$.

The dwell times must be in increments of 0.1 seconds. The source initially moves to the position closest to the instrument and then periodically steps to each dwell position in the catheter.

3.1.2 The γ-Ray Detector Assembly

Each γ-ray detector assembly consists of eight $CdWO_4$ (cadmium tungstate) scintillation crystals, a 16-element photodiode array (model $S5668$ Hamamatsu photodiode array), and associated electronics.

Cadmium tungstate has several desirable properties which make it desirable for our application. It is non-hygroscopic. Therefore, no air-tight enclosures are required. The crystals only need to be light-tight. The light output per incident γ-ray is only 30% of that of other detectors like cesium iodide (CsI). The afterglow[1] factor is about 10 times lower than CsI. This is very important to reduce scatter and scan-times when being used for CT imaging.

The light emission spectrum for $CdWO_4$ crystal peaks at around $500nm$. This is a reasonable match for the spectral response of the photodiode array. In addition, $CdWO_4$ crystals are relatively inexpensive.

Two elements of each photodiode array are used for each detector channel. The crystal-photodiode assembly produces a current proportional to the amount of γ-radiation incident on the crystal.

Each of the eight-channel detector arrays is $2.52cm$ wide. A total of 96 boards therefore form a detector ring which is $77.6cm$ in diameter. Chapter 4 gives a more detailed description of the detector assemblies.

3.1.3 The Source Position Sensor

The 18 independent γ-ray source position sensors are positioned at the beginning of each catheter in the ring. They generate interrupts for the DAS of the CT scanners and trigger the beginning of a data acquisition cycle.

[1]The afterglow of a radioactive source is the time required for the source to decay to 10% of its initial strength

The dwell-times and positions of the source in the catheters are reproducible for the *Nucletron* HDR brachytherapy machine. Therefore, if the time and location of the source is measured once it enters the catheter, the other dwell-times and locations for the catheter can be calculated. Therefore only 18 source position sensors (one for each catheter) is required for estimating all 864 (18×48) positions of the source inside the catheters.

3.1.4 The 96 : 1 Analog Multiplexer Unit

The 96 : 1 analog multiplexer unit multiplexes the analog output of the 96 detector assemblies into a single channel. Each detector assembly houses an 8 : 1 multiplexer on-board. The output of the 768 analog channels are therefore multiplexed to a single A/D board on the digital computer. The multiplexing of the analog channels into a single channel considerably simplifies the design of the master controller and also helps in reducing the cost of the CT scanner. Analog multiplexers are considerably cheaper than A/D converters.

3.1.5 The Analog-to-Digital (A/D) Converter

A *National Instruments* A/D converter is used to convert the analog signal acquired from the detector channels into digital data to be stored in the host computer. The A/D board can be clocked by external pulses at a maximum frequency of $20kHz$. Data is digitized to a precision of 16 bits.

3.1.6 The Host Computer

The host computer (*Apple MacIntosh G3*) in conjunction with the master controller, synchronizes the data acquisition sequence. The host computer is used to enter all the data acquisition parameters. These are then input to the different reg-

isters of the master controller during different stages of the data acquisition. An embedded micro-controller (*I386EX Tern Engine*) works in direct interface with the host computer. It is software-driven. Chapter 6 discusses the software implemented on the micro-controller.

The host computer stores the digitized data and reconstructs the tomogram images from the acquired data.

3.1.7 The Master Controller

The master controller consists of the dedicated control hardware used to synchronize the data retrieval from proper detector channels during the data acquisition cycle. It interfaces with the host computer at one end and the analog data acquisition hardware on the other. Chapter 5 gives a more detailed description of the design and implementation of the master controller.

3.2 Data Acquisition Sequence

The data acquisition sequence of the scanner is determined largely by the limitations and capabilities of the HDR brachytherapy unit. Thus, the acquisition sequence depends on the spatial and temporal accuracy of the HDR instrument.

To begin the acquisition of an image, the HDR instrument is programmed such that the radioactive source moves through each of the catheters surrounding the patient. The source is positioned for a programmed dwell-time at all allowed locations within each catheter. This ensures that the source is positioned at equally spaced locations in the ring around the patient.

At each source position, a number of detectors, positioned on the opposite side of the source, detect the radiation passing through the patient. To generate an artifact-free image, it is important to know the source positions and their dwell-times.

When the radioactive source arrives at the beginning of the catheter, a source position sensor sends the trigger to the master controller. The master controller then initiates a data acquisition sequence for that catheter.

The control circuitry for the Data Acquisition System is a software-controlled interrupt-driven system. Figure 3.3 provides a flow-chart representation of the data acquisition sequence. Before an image is acquired, the host computer transfers the control data to the master controller. This is used to set all the requisite parameters. These include the sequence of the dwell locations and their respective dwell times, including the detector sequences that are to be read for each dwell location.

Once the acquisition parameters are in place, the controller begins waiting for the first sensor interrupt to arrive. When the source is fed to a particular catheter, an interrupt is generated and the particular catheter number is recorded. For proper image reconstruction, the order in which the source enters various catheters must agree with the order expected by the computer.

The active-catheter number is compared with the expected-catheter. A positive-check indicates that the right catheter is in use. The rest of the data acquisition cycle can then progress. The data acquisition cycle is described in greater details in Chapter 5.

A negative check on the encoded source catheter indicates that the wrong catheter is being used. In such a case, the data is discarded. Ideally, any such error detection would trigger the abortion of the data acquisition sequence. However, the A/D converter used in the DAS requires the requested amount of data be transferred before an error message can be generated. Thus, the data acquisition sequence must be carried out and then the subsequent data is to be discarded.

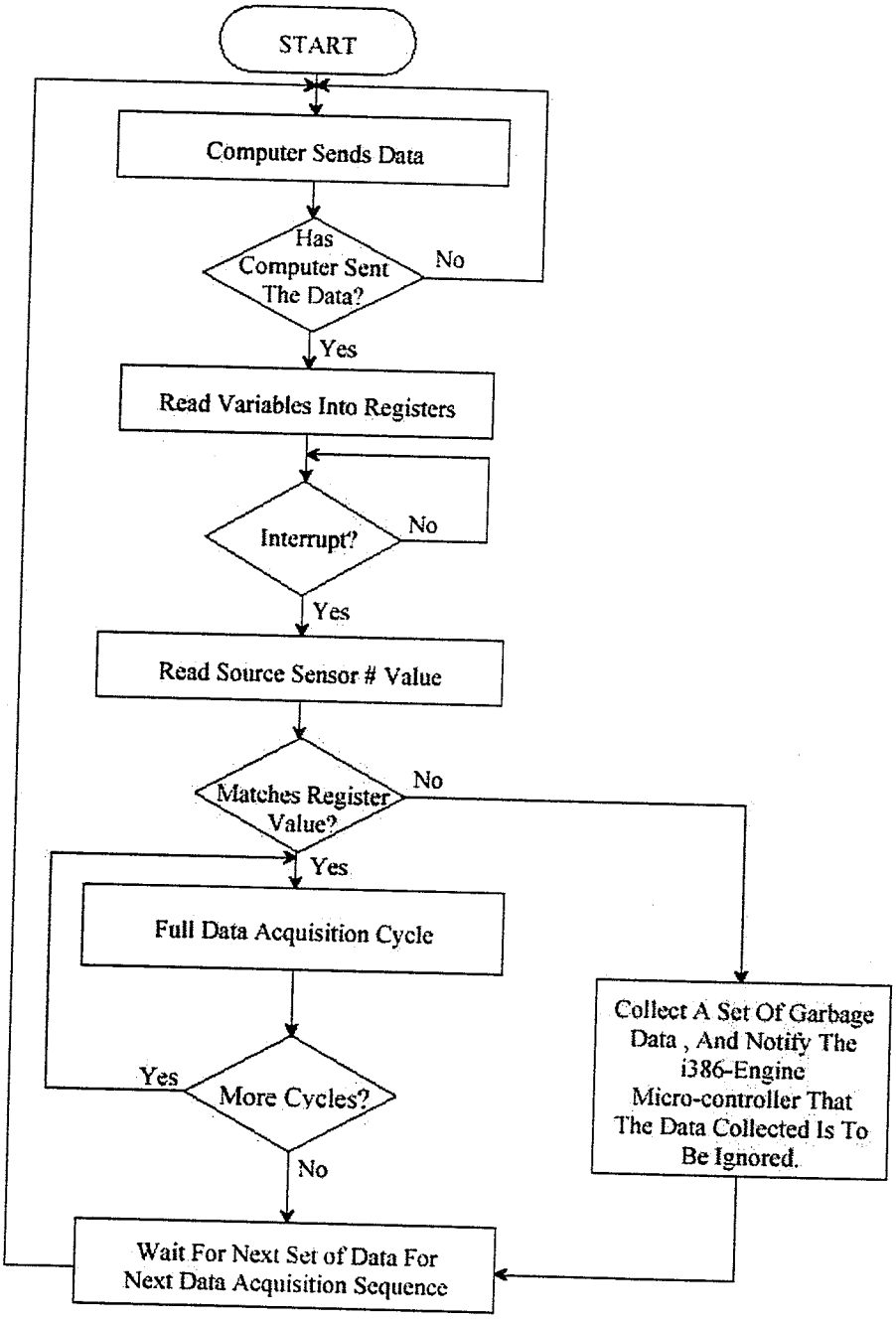

FIG. 3.3. Flow chart representation of the Data Acquisition Sequence.

3.3 Contributions

This thesis was based on work done at Cancer Care Manitoba and was part of joint group project. Numerous sections described in this thesis was a result of contributions from different members of this group. This section enumerates these contributions.

Chapter 4 describes the workings of the analog detector assembly module. The design and implementation of the initial prototype was performed by Dr. Anita Berndt, Dr. Daniel Rickey and Dr. Satyapal Rathee. The eight channel assembly was primarily designed by Dr. Rathee. The prototype was implemented and tested for noise characteristics by Dr. Berndt.

This prototype was ported over onto Printed Circuit Boards by the author. There were 96 such boards which were fabricated. These PCB's were then populated jointly by the author and Dr. Berndt. These 96 boards were then individually tested for both light and dark current noise characteristics by the author. The noise characteristics from the PCB assemblies were validated with those from the prototype design for authentication purposes.

Chapter 5 describes the digital hardware control module. The overall module and interface (Figures 3.2 and 5.1) was designed by Dr. Rathee. Dr. Rathee and the author jointly designed the individual modules with the exception of the Source Sensor Circuit which was designed by Dr. Rathee. These modules were then implemented by the author. Dr. Rathee and the author then jointly tested the modules for correctness. The corresponding signals generated by the modules (Figures 5.8, 5.9 and 5.10) were then recorded for correctness by the author.

Chapter 6 describes the software control module. Dr. Rickey and the author jointly worked on the design and implementation of the software code for the micro-controller modules which is the liason between the host computer and the rest of the

DAS hardware. The software code was individually tested with a single assembly detector board for validation. This testing was also jointly done by Dr. Rickey and the author. Appendix A presents the code for the software.

The Port Controller Unit described in Chapter 6 was jointly designed and tested by the author and Dr. Rathee.

Chapter 4

THE EIGHT CHANNEL ANALOG DETECTOR ASSEMBLY

As noted in Chapter 2, the CT reconstruction is based on measured transmission data of γ-rays at various angles around the object. The first step towards retrieval of meaningful transmission data is the accurate generation of electronic signals proportional to the intensity of X-rays incident on the scintillation detector crystals.

The scintillation crystals produce optical signals proportional to the incident γ-rays. The visible light is then measured by photodiodes. The eight-channel analog detector assembly, described in this chapter, detects the optical signal incident on the photodiode arrays[2]. It generates an electrical voltage proportional to the intensity of the light rays and processes the signal to minimize the effects of any electronic noise and passes the signal onto the A/D converter on the host computer to be digitized and stored.

4.1 Detector Circuit Design

Figure 4.1 shows a single channel in the analog detector assembly board. The detector circuit is divided into four distinct parts. The photodiode converts the optical signal generated by the cadmium tungstate ($CdWO_4$) scintillation crystals

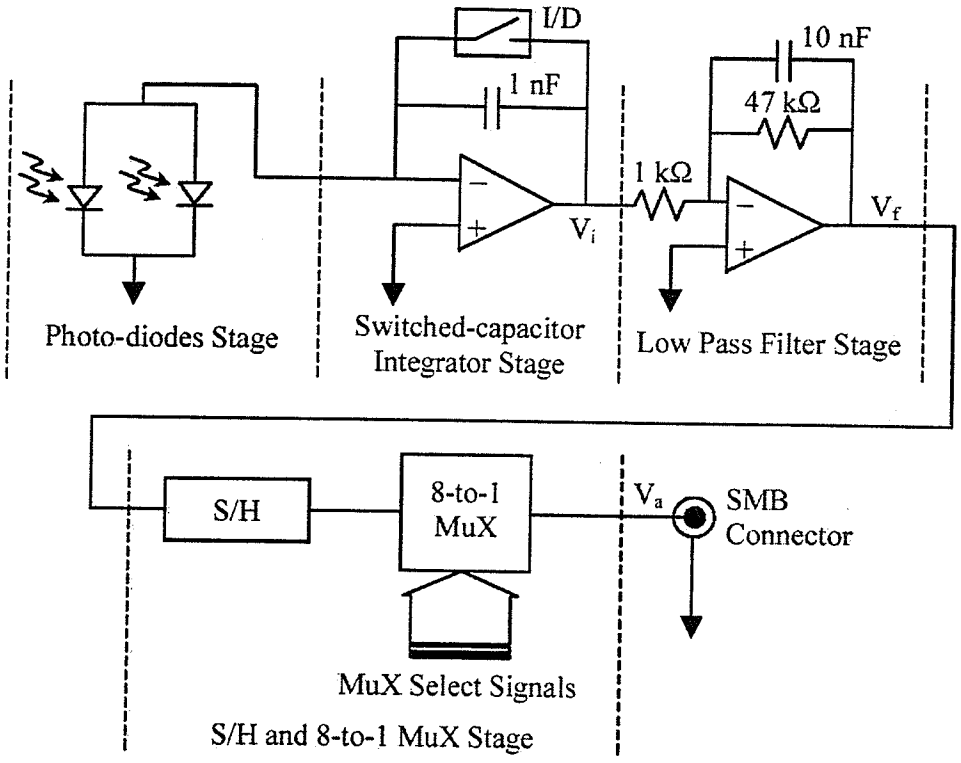

FIG. 4.1. Circuit diagram of one channel on an 8-channel analog detector assembly board. Each channel comprises of four distinct stages: a photodiode pair, a switched-capacitor-integrator stage, a low-pass-amplifier stage, and a S/H-mux stage

into electrical current.

The current is then integrated over a period of time by a switched capacitor op-amp integrator and converted to voltage. The low-pass filter amplifier stage removes high-frequency noise components in the signal and amplifies the signal before it is retrieved by the A/D converter on the host computer. The sample-and-hold and the multiplexer stages coordinate the retrieval of the analog signal by the A/D converter and are controlled by signals from the hardware control unit (Chapter 5).

4.1.1 Photodiode Characteristics

The photodiode array used in this design is a *S5668 Hamamatsu* 16-element diode array. Each photodiode is $0.1775cm$ wide and $0.2cm$ long with a $0.04cm$ gap between consecutive diodes. Each scintillation crystal covers exactly two photodiodes. Therefore, each of the eight analog channels uses two photodiodes to derive the photocurrent to be supplied to the signal processing unit of the assembly (Figure 4.1).

4.1.2 Op-amp Integrator Characteristics

The second stage of the detector assembly is the switched op-amp integrator circuit which converts the electrical current to voltage (Figure 4.1). This stage is required before the signal can be filtered and sampled. Figures 4.2 and 4.4 show two current-to-voltage converter op-amp circuits.

Figure 4.2 shows an op-amp integrator circuit similar to the one implemented in the analog detector assembly. The inverting input terminal of the op-amp is supplied by a current source, which simulates the photodiode in Figure 4.1. Because of the high open loop gain of the op-amps, there is almost zero potential difference between the inverting and the non-inverting terminals of the op-amp. When a DC-current I is passed through the integrator circuit with the switch open for a time period of ΔT, the resulting output voltage V_i can be represented as:

$$V_i = -\frac{1}{C}\int_0^{\Delta T} idt \sim -\frac{I \cdot \Delta T}{C_i} \tag{4.1}$$

Considering a single spectral component of the time-dependent (AC component) current flowing through a photodiode can mathematically represented as:

$$I(t) = I_o(\omega) \cdot \cos(\omega t) \tag{4.2}$$

For an integration time of ΔT seconds (when the switch is open), the output voltage can be formulated as:

$$\begin{aligned} V_i &= -\frac{1}{C_i} \cdot \int_0^{\Delta T} I \cdot dt \\ &= -\frac{1}{C_i} \cdot \int_0^{\Delta T} I_o(\omega) \cos(\omega T) \cdot dt \\ &= \frac{1}{C_i} \cdot \frac{\sin(\omega \Delta T)}{\omega} \end{aligned} \tag{4.3}$$

The transfer function of the switched capacitor integrator circuit is given by eqn.4.4. It relates the voltage output of the integrator to the current input amplitude.

$$\begin{aligned} H(\omega) &= \left| \frac{V(\omega)}{I(\omega)} \right| \\ &= \left(\frac{\sin(\omega \Delta T)}{\omega \Delta T} \right) \cdot \left(\frac{\Delta T}{C_i} \right) \\ &= \frac{\Delta T}{C_i} \cdot sinc(\omega \Delta T) \end{aligned} \tag{4.4}$$

The impulse response of the integrator can be represented as the inverse Fourier transform of the transfer function in eqn.4.4:

$$h(t) = F^{-1}[H(\omega)] = \begin{cases} \frac{1}{2C_i} & -\Delta T \leq t \leq \Delta T \\ 0 & \text{otherwise} \end{cases} \tag{4.5}$$

Sources of electrical noise generated by the detector assembly include [2]:

1. current noise in the photodiode and op-amps,
2. input noise voltage of the op-amps,
3. cycle-to-cycle variation of charge injector into the integrator capacitor from the digital switch

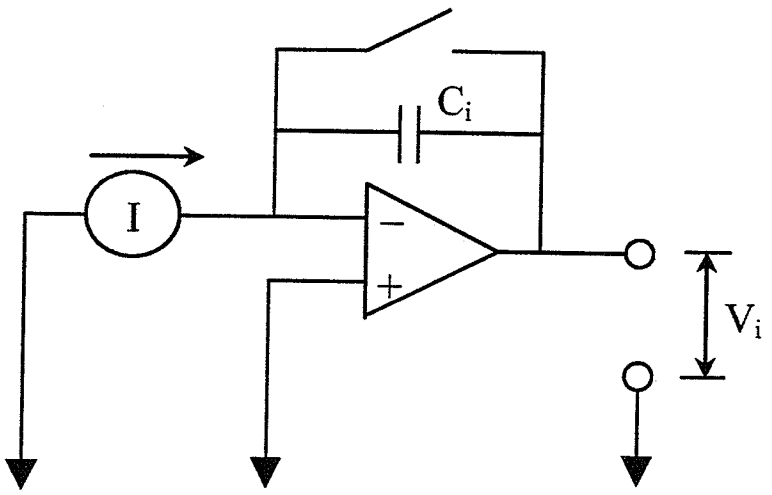

FIG. 4.2. Schematics for a switched capacitor op-amp integrator based current-to-voltage converter

4. cycle-to-cycle variation of the retained charge after integration capacitor discharge, and

5. shot noise due to leakage current in the switch open state.

For minimizing the effects of the cycle-to-cycle variation of the injected and retained charges on the capacitor C_i, the output voltage is sampled twice - once before the integration commences (when the capacitor is in a discharged state) and a second time after the integration is complete. The difference between the two voltage is stored as a valid reading. Since both these charges are placed on the C_i at the beginning of the integration period, this scheme virtually eliminates injection and retention current noise.

The gains for the the switched capacitor integrator design are $\Delta T/C_i$. The gain for the implemented converter stage for a ΔT of $4.0ms$ and C_i of $1nF$ is about 4×10^6.

4.1.3 Switched Capacitor Integrator Noise Characteristics

There are several different sources of noise in the aforementioned switched-capacitor integrator circuit. The dark current produced by the photodiode when no light is incident on it is a source of noise. The op-amp is also a source of noise. The digital switch in the circuit is also a source of leakage current noise. The theoretical estimation mathematics is also documented in Dr. Anita Berndt's doctoral thesis.

Assume that the cumulative noise *current* spectral density is given in the frequency domain as σ_{ni}^2. The noise spectral density of the output *voltage* σ_{nv}^2 is represented as:

$$V_n(\omega) = H^2(\omega) \cdot \sigma_{ni}^2 \tag{4.6}$$

,or

$$\sigma_{nv}^2 = \frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} \sigma_{ni}^2 \cdot H^2(\omega) \cdot d\omega \tag{4.7}$$

Eqn.4.7 can be reformulated by inserting the expression of $H(\omega)$ from eqn.4.4. The noise variance can then be formulated as:

$$\sigma_{nv}^2 = \frac{1}{2\pi} \cdot \int_{-\infty}^{\infty} \sigma_{ni}^2 \cdot \frac{\Delta T^2}{C_i^2} \cdot sinc^2(\omega \Delta T) \cdot d\omega \tag{4.8}$$

The output noise function for the switched capacitor can be obtained by solving eqn.4.8, and the result obtained is:

$$\sigma_{nv} = \sqrt{\frac{\Delta T}{2}} \cdot \frac{\sigma_n i}{C_i} \tag{4.9}$$

In order to estimate the noise voltage of the detector assembly board, we use the following noise characterizations to calculate the final noise [2]. The noise spectral densities of the different components are:

- Dark current - $2.9 \times 10^{-15}\ A/\sqrt{Hz}$.
- Op-amp input noise - $10.0 \times 10^{-15}\ A/\sqrt{Hz}$.
- Digital switch - $1.8 \times 10^{-15}\ A/\sqrt{Hz}$.

The cumulative input current noise spectral density can be calculated by the square root of the sum of the squares of the components and comes out to be $1.06 \times 10^{-14}\ A/\sqrt{Hz}$. The output voltage noise spectral density by eqn.4.9 can be calculated to be approximately $4.69 \times 10^{-7}\ V/\sqrt{Hz}$.

Using Section 4.1.4, we can see that the bandpass is $339Hz$ and the gain is about 47.0. Multiplying the voltage noise spectral density, the bandwidth of the low pass filter and the filter gain, we get the figure of $747\mu V$ for the total noise voltage at the end of the analog detector boards.

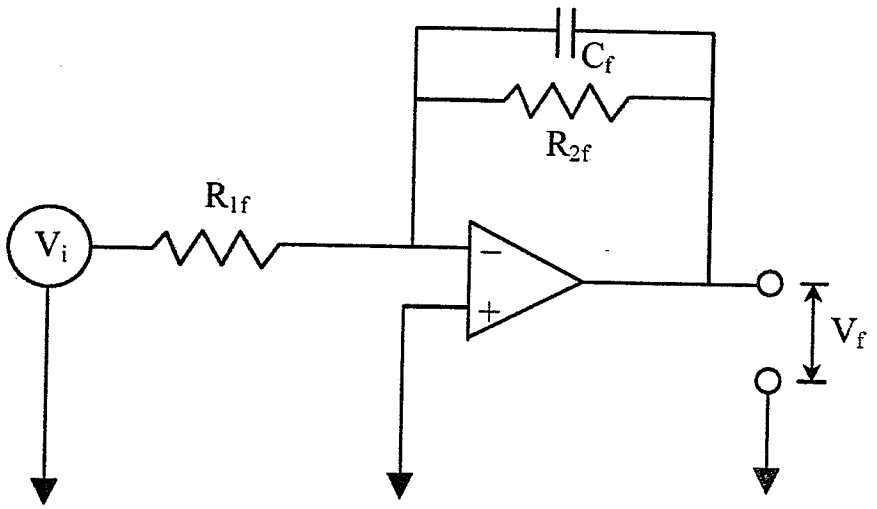

FIG. 4.3. Schematics for a low-pass-filter used to suppress high frequency noise in the analog detector boards

4.1.4 Low Pass Filter Characteristics

The integrated signal is passed through a low-pass filter stage which not only reduces the effect of high frequency noise but also amplifies the voltage by a factor of 47 (Figure 4.3). The $-3dB$ cut-off frequency for the low-pass filter can be given by:

$$f_c = \frac{1}{2\pi R_{2f} C_f} \tag{4.10}$$

With R_{2f} being $47k\Omega$ and C_f being $10nF$, the cut-off frequency comes out to be about $339Hz$. The gain for the low pass filter amplifier stage (as in given in eqn.4.10) is 47.

$$G = \frac{R_{2f}}{R_{1f}} \tag{4.11}$$

The transfer function of the filter stage is given by:

$$f(t) = F^{-1}\left[\frac{G}{1 + j\omega/\omega_c}\right] = G\omega_c \exp\left(-\omega_c t\right) \tag{4.12}$$

where ω_c is $2\pi f_c$ and F^{-1} is the inverse Fourier transform operation.

4.1.5 The S/H and 8:1 Multiplexer Stage

Each detector assembly consists of eight individual detector channels. These channels are multiplexed into a single analog channel. There are 96 such eight-detector assembly boards. These are fed to the 96 : 1 multiplexer unit (Chapter 5) to be eventually connected to the single A/D converter. The 8 : 1 multiplexer chip used in this case is the $CD14051$.

All detector elements (a maximum of 768 have to be measured at the same instant in time during a measurement cycle. This is required to keep the electronic gain identical for all the channels. Therefore, the amplified voltage signal is held at

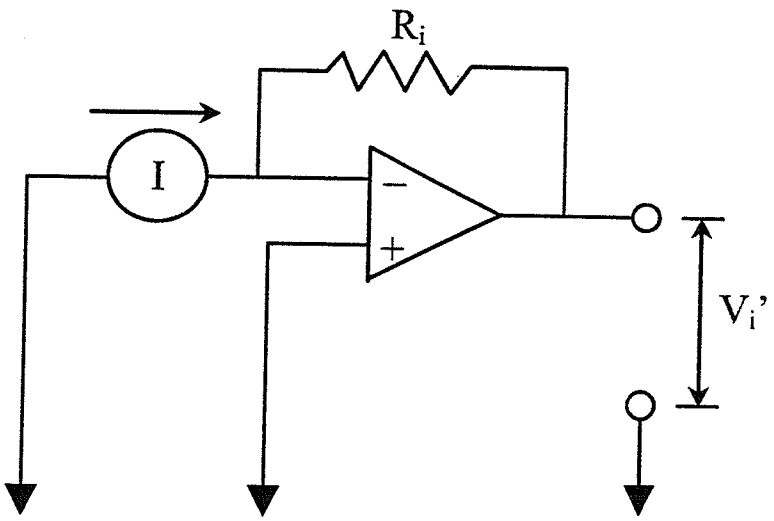

FIG. 4.4. Schematics for a resistor based op-amp current-to-voltage converter

the same time for all detector channels. This is achieved by the S/H circuit. Two 4-channel S/H chips (Analog Devices $SMP-04$ are used for this purpose.

The width of the holding pulse is determined by the number of detectors to be sampled during the pulse. During this time period, multiplexer control signals are used to cycle the analog input of the A/D converter. These signals control the on-board 8 : 1 multiplexers and the rest of the analog multiplexer unit described in Chapter 5

4.2 Transconductance Amplifier Overview

The output voltage of the circuit in Figure 4.4 is represented as:

$$V_i' = I \cdot R_i \tag{4.13}$$

For an ideal noise-free constant current source, both the circuits can be used interchangeably. However specific limitations on the noise requirements of the signal generated makes the switched capacitor op-amp integrator circuit a better choice.

The gain of the transconductance amplifier is R_i. For comparable gains between the two designs, it was experimentally determined that the switched-capacitor based integrator outperformed the resistor based op-amp amplifier circuit. The resistor based op-amp has the disadvantage of generating substantial thermal noise for comparable gains.

4.3 Detector Assembly Board Fabrication

The circuit described in Section 4.1 was first implemented as a prototype wire-wrap circuit board. The average dark noise was experimentally noted to be approximately $110\mu V$ at a sampling frequency of $100Hz$[2]. There was a difference of approximately a factor of 7.0 between the estimated noise voltage and the experimentally obtained one. This can possibly be attributed to inaccurate assumptions about op-amp noise current characteristics during the estimation phase.

The PCB-design package ORCAD^{TM} was used to transport the original design on to PCB. The eight-channel photodiode assembly is $2.52cm$ wide, and this constitutes the width of the boards. Since each of these detector assemblies need to be flush with each other, these constraints were kept in mind when designing the board. Furthermore, since these boards sit on top of the gantry, these boards were designed so that all populated components are placed on only one side. Ninety-six of these boards were then fabricated on printed-circuit-boards (PCB's).

Each board is $2.52cm$ wide and approximately $25cm$ long. Figure 4.5 shows both sides of the PCB board designed. All individual components were hand-soldered on to each of the 96 boards and were tested for noise characteristics for both dark as well as light currents.

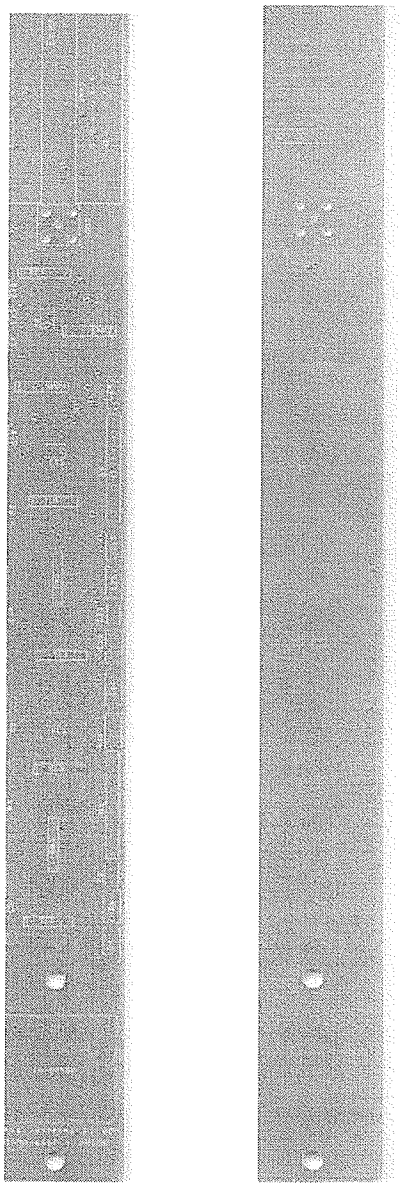

FIG. 4.5. The two sides of the printed circuit board hosting the eight channel analog detector circuit. Components were hand-soldered on to the top of the PCB

4.4 Detector Assembly Characterization

Once fabricated and populated, each of these individual detector assembly boards have to be tested to observe if they confirm the noise figures gained from the prototype experiment results. Assembly boards which do not conform to the noise figures have to be discarded.

In order to characterize the detector assemblies, a dedicated clock board was fabricated. One board was tested at a time, i.e., only one assembly was connected to the clock board. To reduce RF interference, the assembly was mounted on an aluminium bar that was connected to ground. The output of the detector assembly was connected directly to the A/D converter board. The 96 : 1 analog multiplexer unit was not used during the test.

The source of light used in the experiments was a 150-mCandela green LED. This mimics the spectral output of the scintillator crystals. Earlier experiments conducted had sought to get the validity between the intensities of the LED and the radioisotope source[2]. The light was connected in series with a $4.7k\Omega$ resistor and a $10V$ DC power supply.

4.5 Detector Output Results

The two sets of experimentations conducted are enumerated in the following two sections. Section 4.5.3 summarizes the result of the tests.

4.5.1 Dark Current Characteristics

The results for dark current measurements are enumerated below:

1. Figure 4.9 shows the distribution of dark current noise among each of the eight channels on the 96 boards checked.

Channel #	1	2	3	4	5	6	7	8
Voltage (μV)	112.13	112.21	112.61	114.51	113.99	113.03	113.32	113.46

Table 4.1. Average noise measurements for each of the eight channels in dark-current conditions

Channel #	1	2	3	4	5	6	7	8
Voltage -(μV)	128.48	127.61	127.38	129.26	124.86	125.61	127.91	126.39

Table 4.2. Average noise measurements for each of the eight channels in light-current conditions

2. Table 4.1 enumerates the average noise measurements for channels 1 through 8. The noise measurements statistics and the average and standard deviation for each channel is plotted in Figures 4.7 and 4.8 respectively.

3. The cumulative average noise was calculated to be $113.16\mu V$, which is similar to the $110\mu V$ reported by Berndt et al,[2].

4. The standard deviation of the noise is $9.16\mu V$.

5. The average dark current readings for each individual channels is shown in Figure 4.6.

6. The average dark current readings come out to be $258.53mV$ with a standard deviation of $42.40mV$.

4.5.2 Light Current Characteristics

The results for the light current measurements are enumerated below:

1. Figure 4.10 shows the distribution of light current noise figures among each of the eight channels on the 96 boards checked.

2. For light readings, the average noise figures for channels 1 through 8 are provided in Table 4.2.

3. Figures 4.11 and 4.12 shows the cumulative noise statistics and the average and standard deviation for each of the individual channels respectively.

4. The average cumulative noise figure obtained was $127.19\mu V$.

5. The standard deviation on the noise is approximately $9.75\mu V$.

6. The average light readings for each of the individual channels is shown in Figure 4.13.

7. The average readings come out to be $-211.07mV$ with a standard deviation of $55.58mV$.

4.5.3 Results

As is obvious from the noise characteristics from the previous two sections, the dark noise current figures closely compared with the prototype but not with the theoretically derived number. The figures from the light noise current characterization are slightly higher than those of the dark noise current.

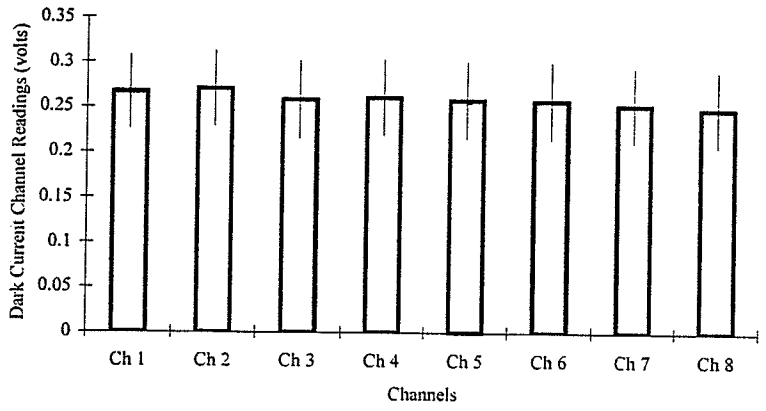

FIG. 4.6. Average signal readings for the dark-current experiments.

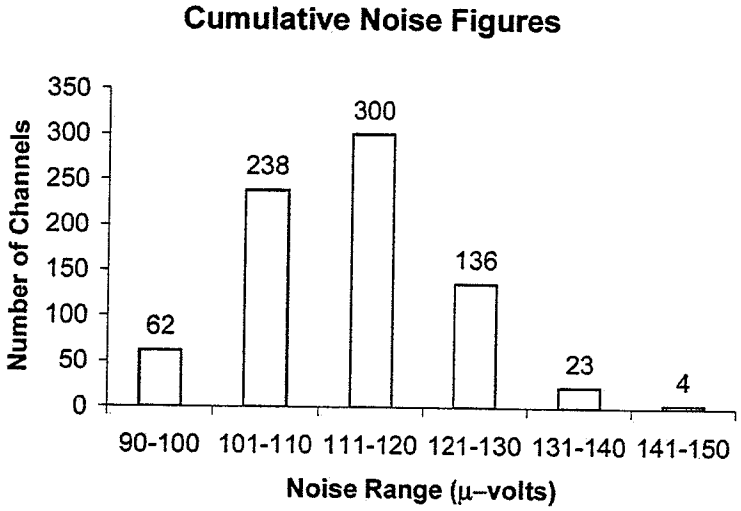

FIG. 4.7. Noise statistics for dark-current experiments - cumulative noise statistics

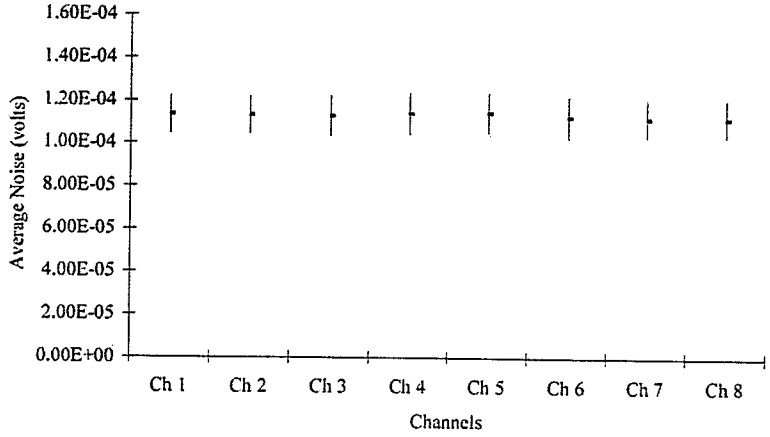

FIG. 4.8. Noise statistics for dark-current experiments - average noise.

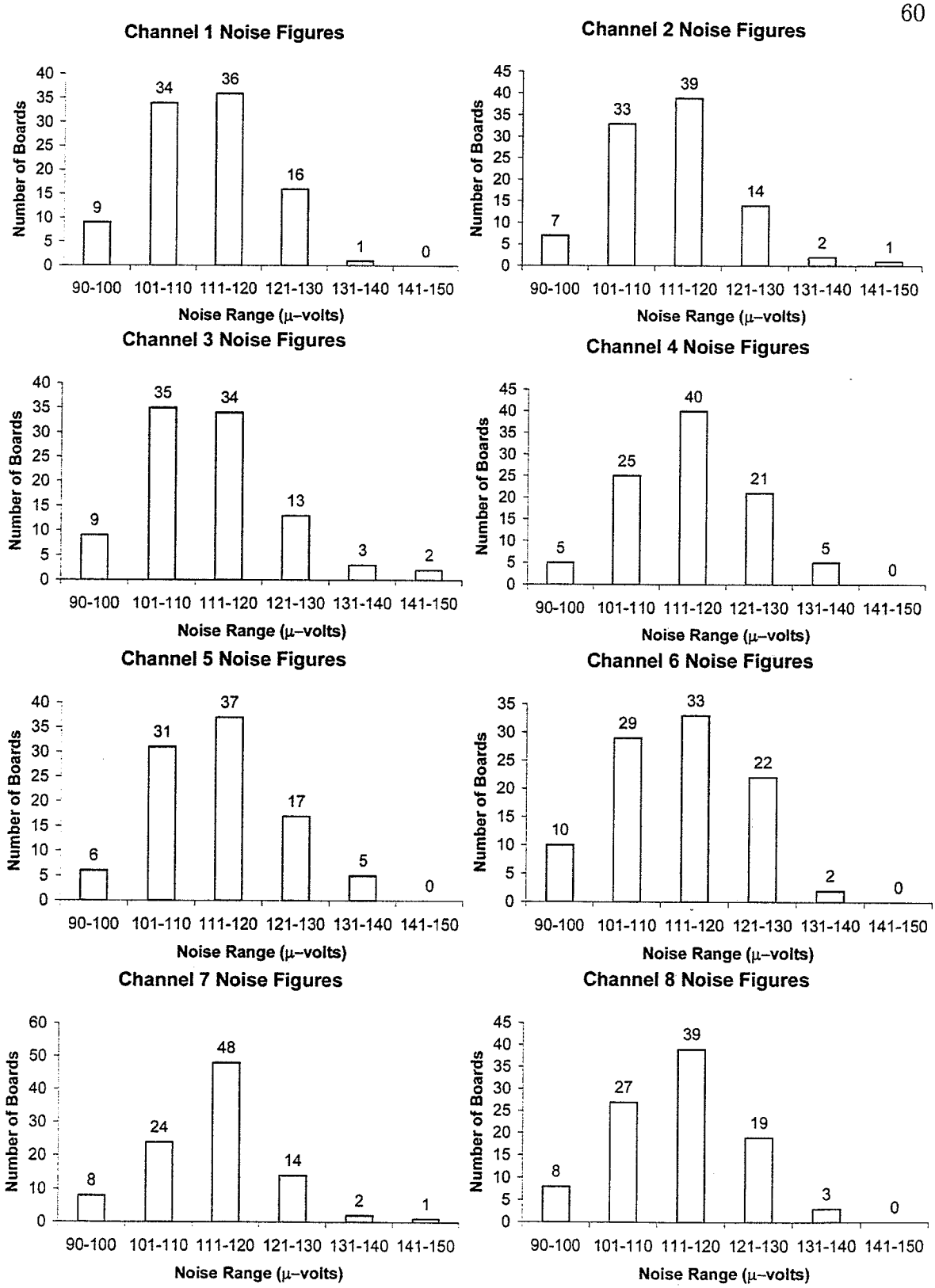

FIG. 4.9. Noise statistics for the eight channels on the analog detector assembly board for dark current measurements

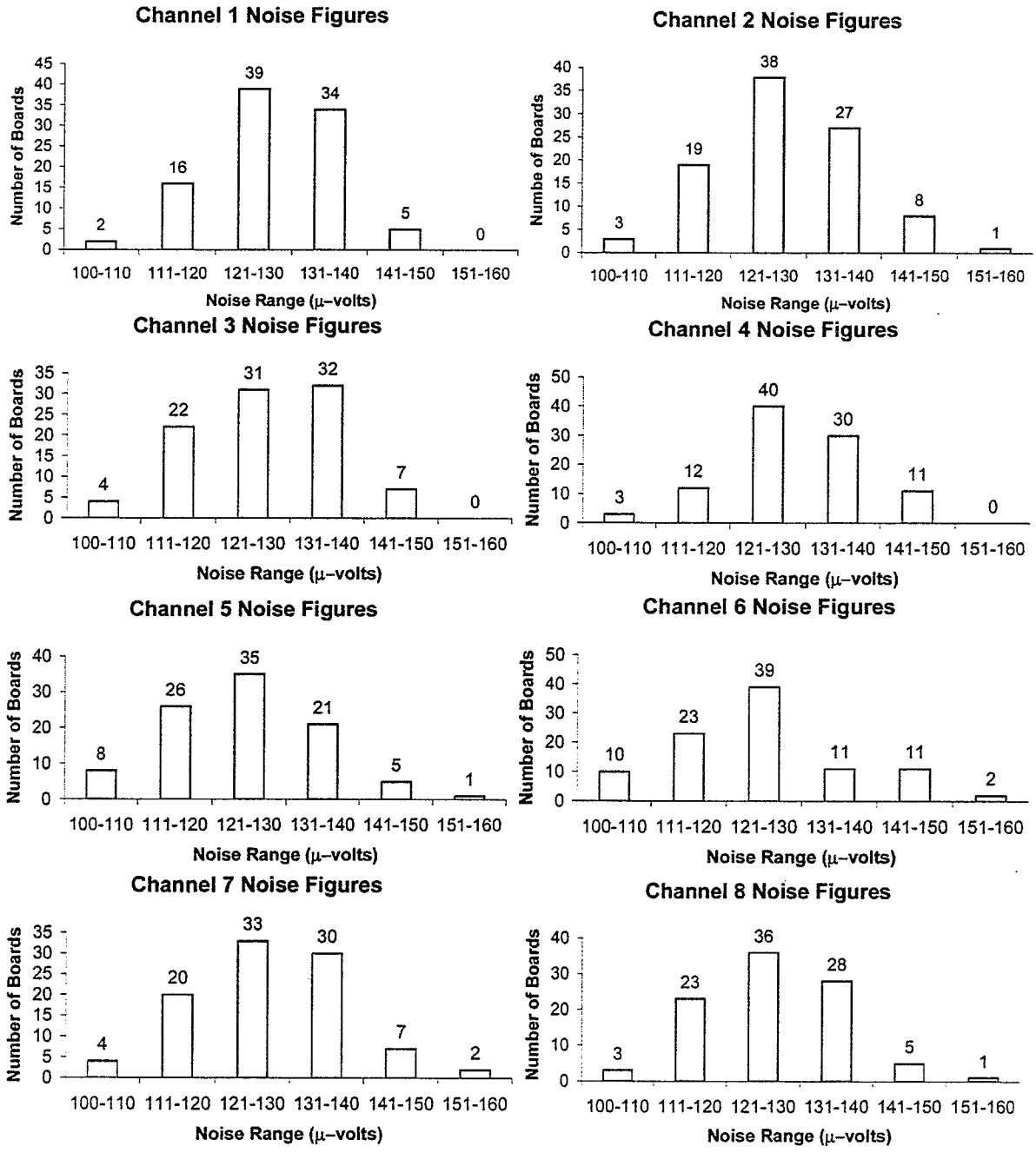

FIG. 4.10. Noise statistics for the eight channels on the analog detector assembly board for light current measurements

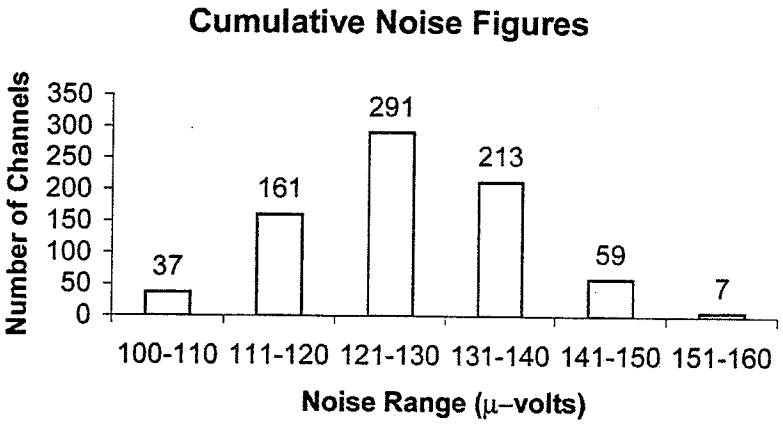

FIG. 4.11. Noise statistics for light-current experiments - cumulative noise statistics

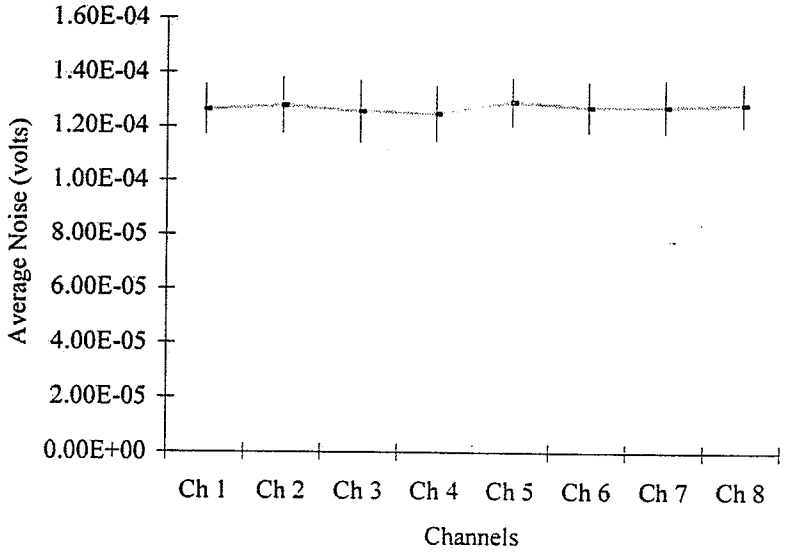

FIG. 4.12. Noise statistics for light-current experiments - average noise.

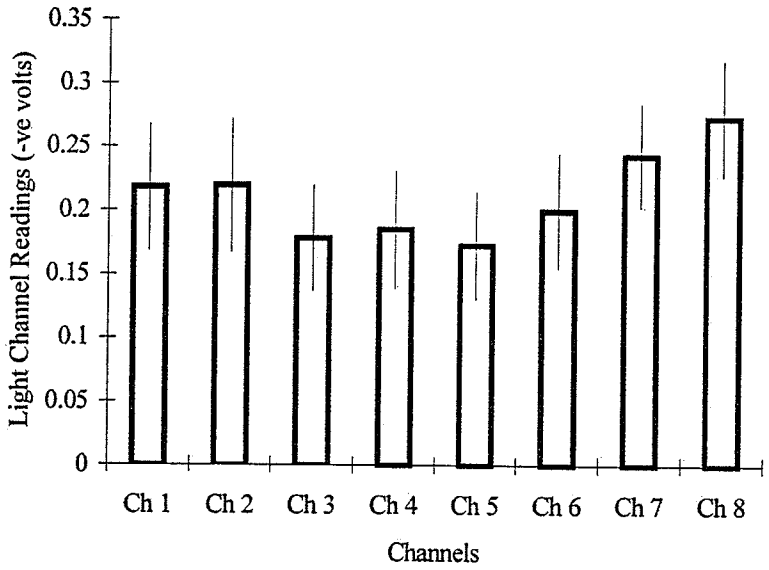

FIG. 4.13. Average signal readings for the light-current experiments.

Chapter 5

DESIGN AND IMPLEMENTATION OF THE HARDWARE CONTROL CIRCUIT

This chapter describes the design and implementation of the various modules in the hardware control circuit which functions as the master controller for the data acquisition system of the 4^{th} generation CT scanner described in this thesis. The modules described in this Chapter is a result of joint work between the author and Dr. Satyapal Rathee. Section 3.3 enumerates the contributions in greater details. Figure 5.1 shows a block diagram of the various modules implemented in the hardware control circuit as well as the signals between each module.

Chapter 3 provides an overview of the system and a description of the HDR brachytherapy machine which is used as a controller for the imaging source. We take another look at a complete data acquisition sequence before we begin our description of the control hardware.

At the beginning of a data acquisition sequence, the HDR brachytherapy unit puts the source in the first catheter at the position nearest to itself (the 48^{th} position). The source then moves from the 48^{th} through the 1^{st} position in the catheter according to pre-programmed dwell times at each dwell location. This cycle is repeated in each of the remaining 17 catheters.

Each catheter has a source sensor, which when it detects a source, signals the

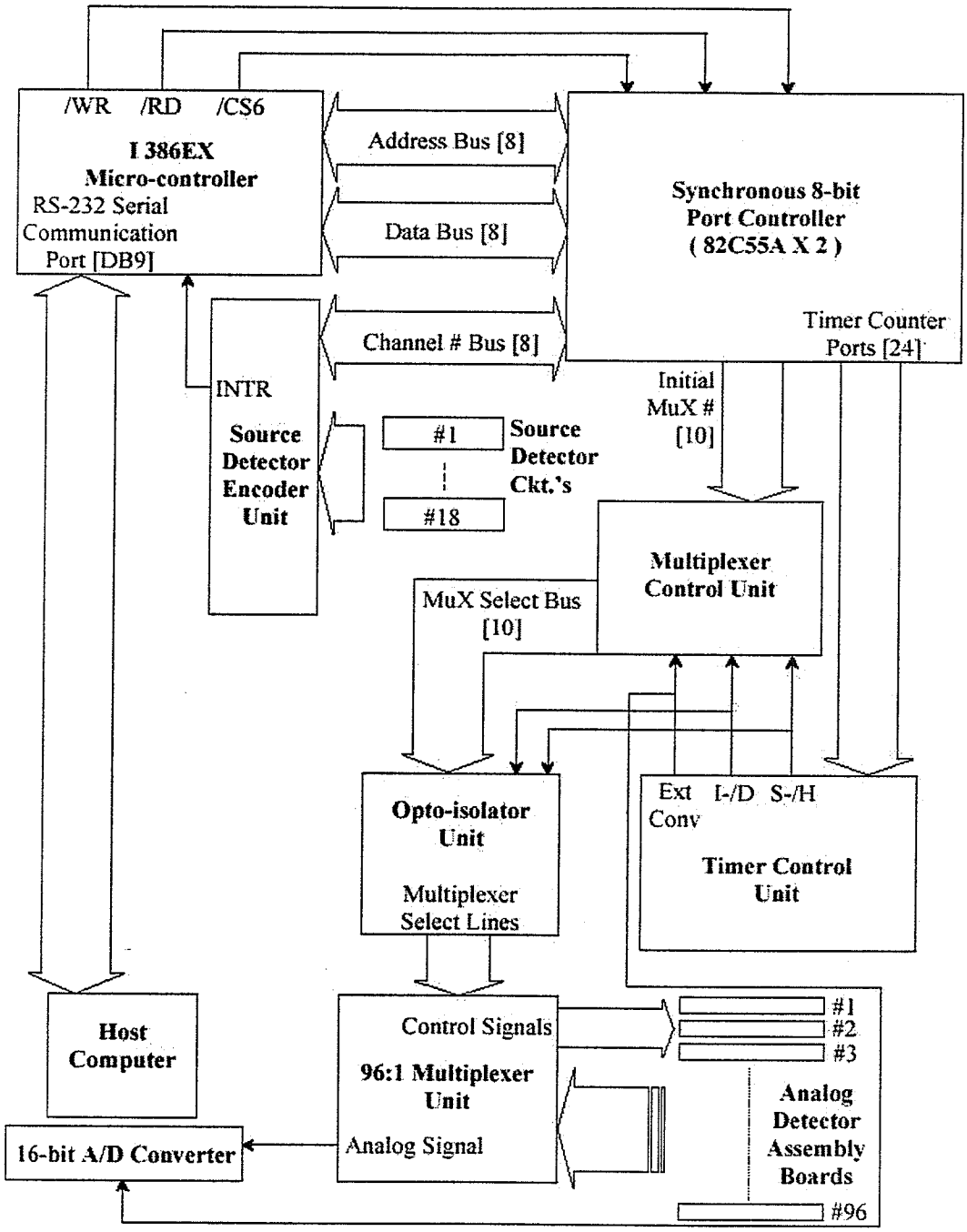

FIG. 5.1. Block diagram showing the different modules of the Data Acquisition System.

arrival of the source at that particular catheter. The DAS then measures a set of pre-programmed detector signals located on the other side of the patient in an arc large enough to encompass the patient completely. After the first set of detector readings, the source changes it dwell location.

We now discuss the various modules in the hardware control circuit. Section 5.2 takes a more detailed look at the timing signals in the system.

5.1 The Source Sensor Circuits and the Source Sensor Encoder Unit

The function of the source sensor circuits is to detect the arrival of the radioisotope source in its assigned catheter. Figure 5.2 shows the schematic diagram for one of the 18 source sensor circuits in use.

A scintillator crystal-photodiode assembly is used to convert the γ-radiation into an electric current. It is then converted to voltage by a trans-conductance amplifier and further amplified (Figure 5.2)[1]. The RC time constant in the second stage of the amplifier is set such that the response time remains low enough on one hand and the ringing of the output signal is suppressed on the other.

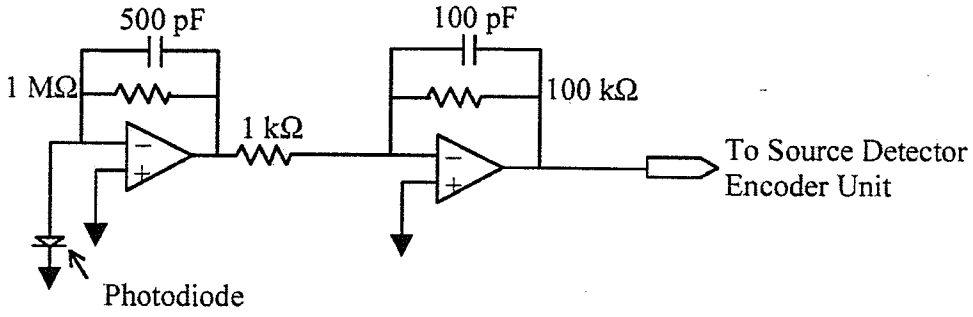

FIG. 5.2. Schematics for the source sensor circuit which detects source in the catheter during a data acquisition sequence.

[1]Work done after this thesis have shown problems with the source sensor circuit when used in the overall system

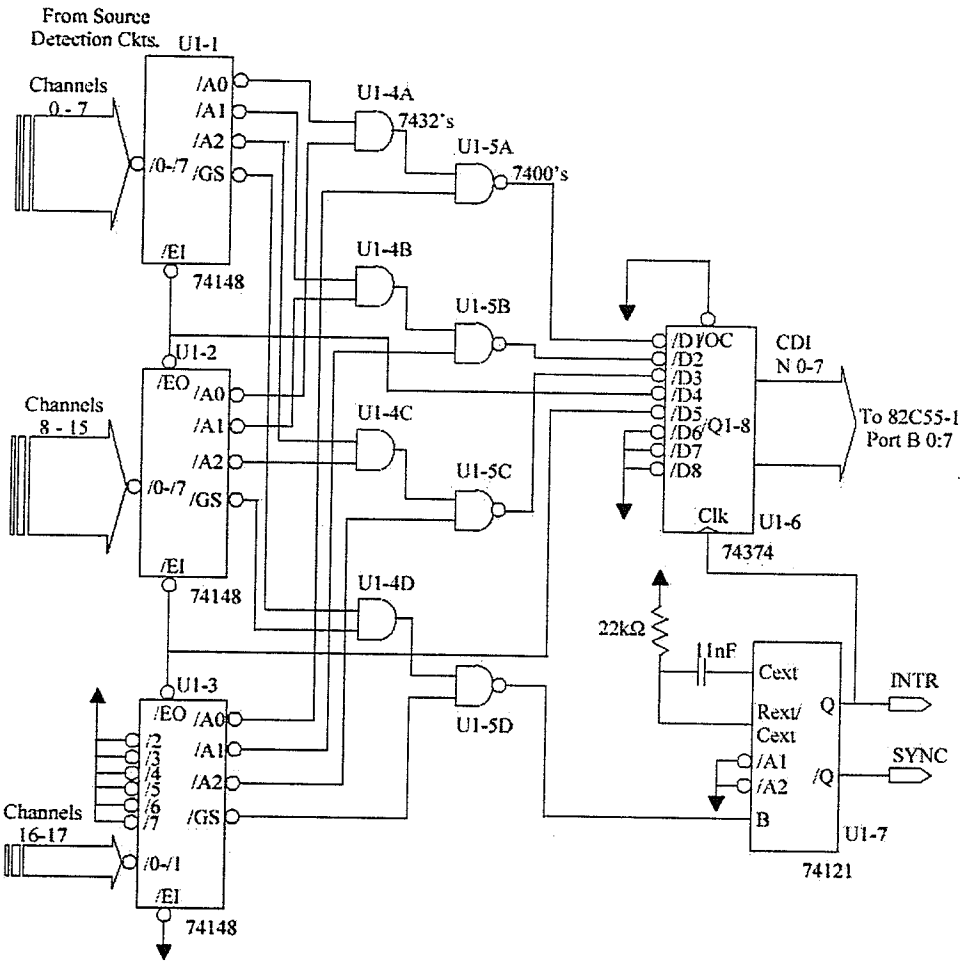

FIG. 5.3. The sensor encoder circuit which encodes the signal from the source sensor circuits and generates the number for the particular sensor circuit the signal came from.

The source sensor encoder unit is used to determine the channel number occupied by the radioisotope. Figure 5.3 shows the schematics of the encoder unit. There is no direct way of knowing whether the radioisotope source has been launched into the desired channel. Therefore, the 18 outputs from the source sensor circuits are fed to an 18 : 5 encoder. This is implemented by using two 8 : 3 $74LS148$ encoders, as shown in Figure 5.3. The encoded channel number is held at the input port of the port controller unit (part of the embedded micro-controller unit which is described in

Chapter 6). This is done by using an 8-bit $74LS374$ latch. The micro-controller then verifies whether the proper channel number is in use or not.

The source detector encoder unit also generates a digital interrupt signal when it detects radiation at one of its inputs. This signals the start of the acquisition cycle. The `SourceDetect` interrupt generated by the encoder unit uses a $74LS121$ monostable multivibrator. This generates a $170\mu s$ wide pulse. This ensures that there is enough time for the micro-controller (Chapter 6) to load the set of detector parameters onto the counters in the timer control unit. Section 5.2 gives a more detailed explanation of this.

5.2 The Timer Control Unit

The timer control unit is responsible for generation of the synchronous control signals used by the other modules. As discussed in Chapter 4 the data acquisition cycle for a particular source position consists of sampling the required detectors twice. For each data acquisition cycle, each detector is first sampled at the start of the integration cycle. The signal is then integrated over a certain pre-determined period of time and then sampled again. The difference between the second and the first samplings constitutes the data collected for reconstruction. This, as discussed in Chapter 4 is one of the steps taken towards reducing the effects of noise. This constitutes a single signal acquisition sequence.

The timer control unit generates three control signals for the analog detector boards, namely:

- The integrate/discharge `I/D` signal,
- The sample-and-hold `S/H` signal, and
- The gated clock `ExtConv` signal.

The software controlled I/D signal is used to discharge the integrating capacitors on the analog boards at the end of a signal acquisition sequence. For a given source position, the time duration of the signal acquisition sequence is determined by the pre-programmed dwell time of the source at that particular position. This may or may not be the same for every dwell location.

The S/H signal is used to hold the initial and the final integrated output of each channel till the signal is read and stored by the host computer. There are two S/H signals during a signal acquisition sequence - one for the initial and a second one for the final integrated output. By using a universal S/H signal to trigger all the channels, it is ensured that all channel integrate over exactly identical periods of time. The width of the S/H pulses is determined by the number of channels to be sampled. There is a fixed gap of $4.0ms$ between the initial and the final pulses of S/H signals. This time delay is a function of the RC constant of the low-pass-filter/amplifier stage of the analog unit.

The ExtConv signal is a gated clock signal which causes the A/D conversion sequence in the host computer. The number of ExtConv pulses during a given hold period equals the number of detector channels to be sampled.

Since the maximum speed of the A/D converter is $20kHZ$, the basic clock speed of the timer control unit is also $20kHz$. Figure 5.5 shows a $1MHz$ crystal oscillator unit which is hard-wired to generate a $20kHz$ signal.

Figure 5.4 shows the wirings of $4n$-bit counters used in the timer control unit using 4-bit $74LS163$ counters. These counters are responsible for generating the detector numbers to be read during a specific S/H holding cycle. The minimum time of acquisition allowed by the HDR brachytherapy machine is $0.1s$. The time-of-acquisition counter (Figure 5.5) can be implemented as a 12-bit counter triggered by the $20kHz$ clock.

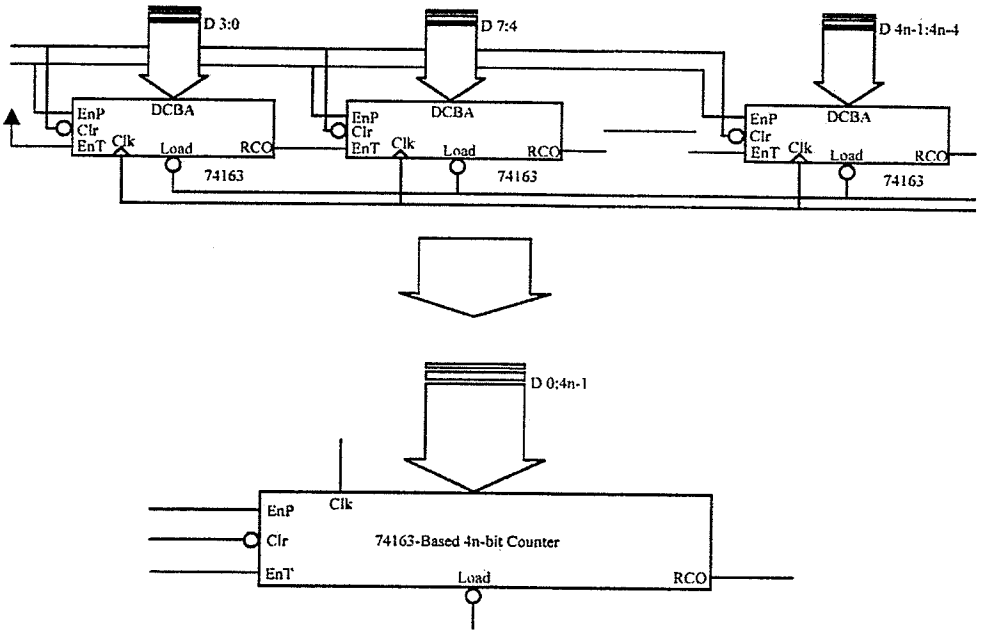

FIG. 5.4. Wiring diagram for converting n 4-bit counters into a $4n$-bit counter

There are, as mentioned earlier, a total of 768 detectors in the collimator ring. The 12-bit number-of-detector counter is used to accomplish a detector counter. The fixed time gap of $4.0ms$ is accomplished by the time-delay counter. These counters are shown in Figure 5.5

5.2.1 Description of Timing Control Signals

Figure 5.6 shows the timing requirements of the important signals in the circuit. The hardware implemented to achieve this is shown in Figure 5.7. The `I/D` and the `S/H` signals are implemented by $74LS74$ D flip-flops controlled by the terminal counts of their respective counters. A description of the important signals and a typical signal acquisition sequence is provided below is provided below.

1. A `PowerON` reset signal is generated when the hardware control unit is switched ON. The `PowerON` reset signal sets the `S/H` signal to sample (not hold), the `I/D`

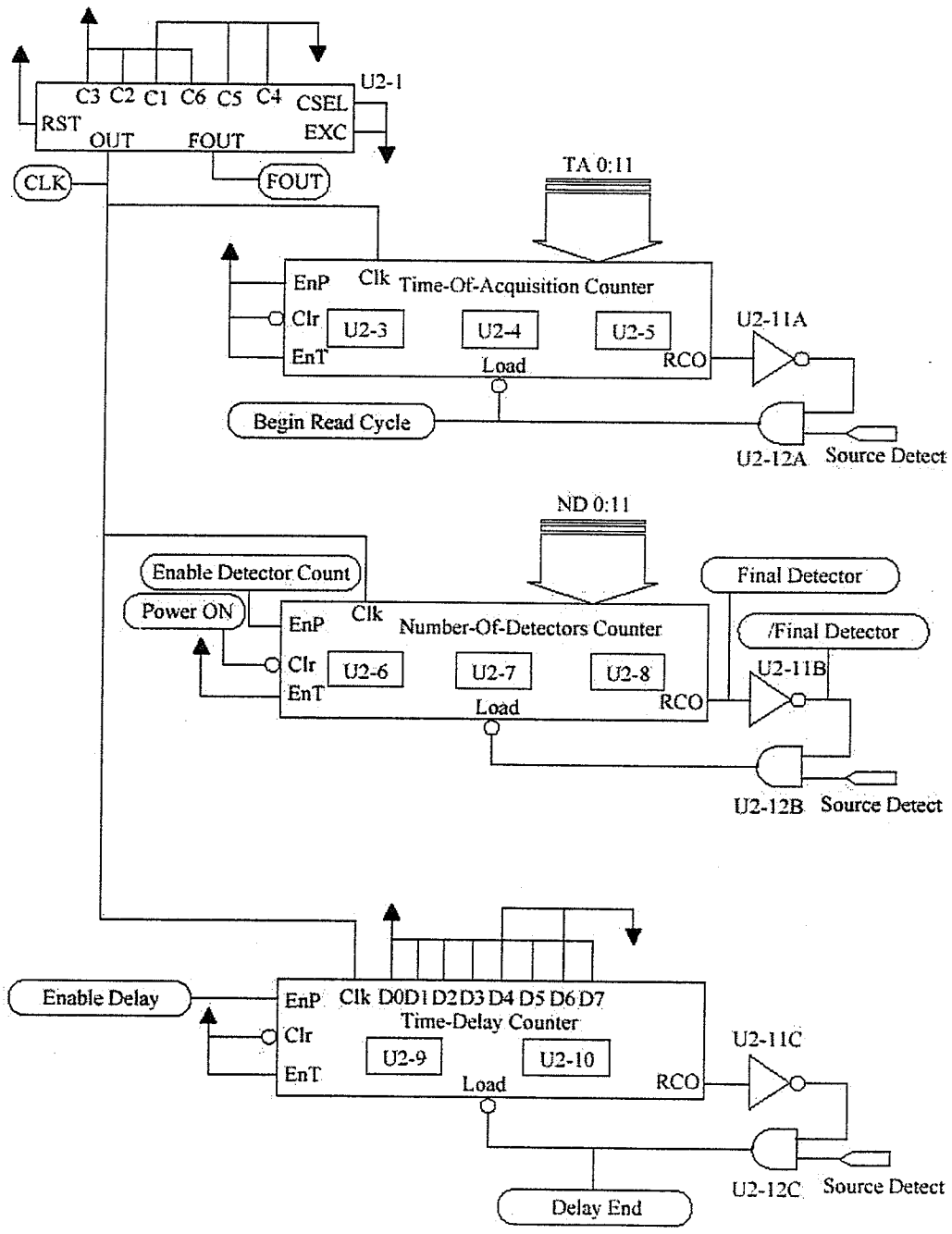

FIG. 5.5. Block diagram for the Timer Control Unit. Shown are the counters which generate the number-of-detectors-to-be-read, the time-of-acquisition and the time-delay-between-two-S/H

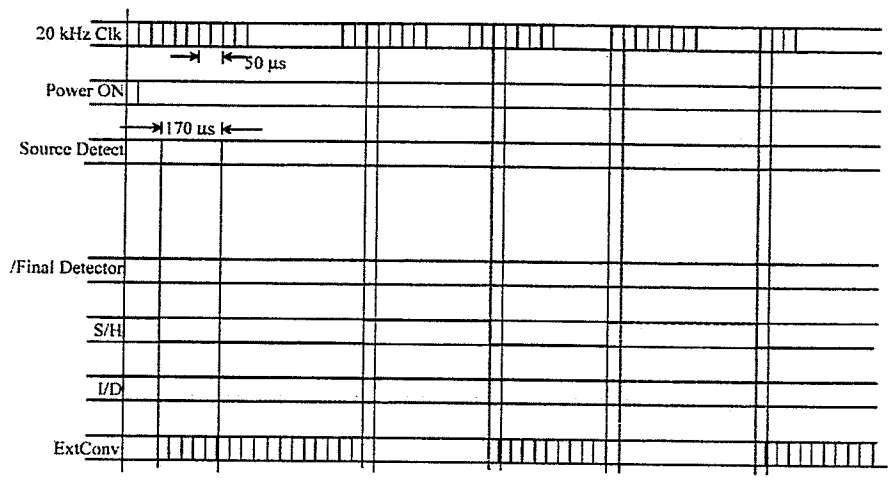

FIG. 5.6. The synchronous timing control signals generated by the timer control unit.

to discharge (not integrate) and disables the `ExtConv` signal.

2. The `SourceDetect` interrupt signifies that the radioisotope has entered the catheter and is at the first position in the catheter. The interrupt service routine on the host computer pre-loads the dwell times on the time-of-acquisition counter and the number of detectors to be loaded on to the number-of-detectors counter. It then initiates the `S/H` signal.

3. The `S/H` signal enables the number-of-detectors counter and the `ExtConv` pulses to be sent to the host computer. When all the detectors have been read, a `FinalDetector` signal disables the hold pulsed on the `S/H`. It also puts `I/D` to discharge and enables the time-delay counter. Since the initialization of various registers occurs after the `SourceDetect` interrupt is serviced by the software, the data read during the very first holding pulse of `S/H` is irrelevant and thus discarded.

4. After a delay of $4.0ms$, a `DelayEnd` signal initiates another hold pulse on `S/H`

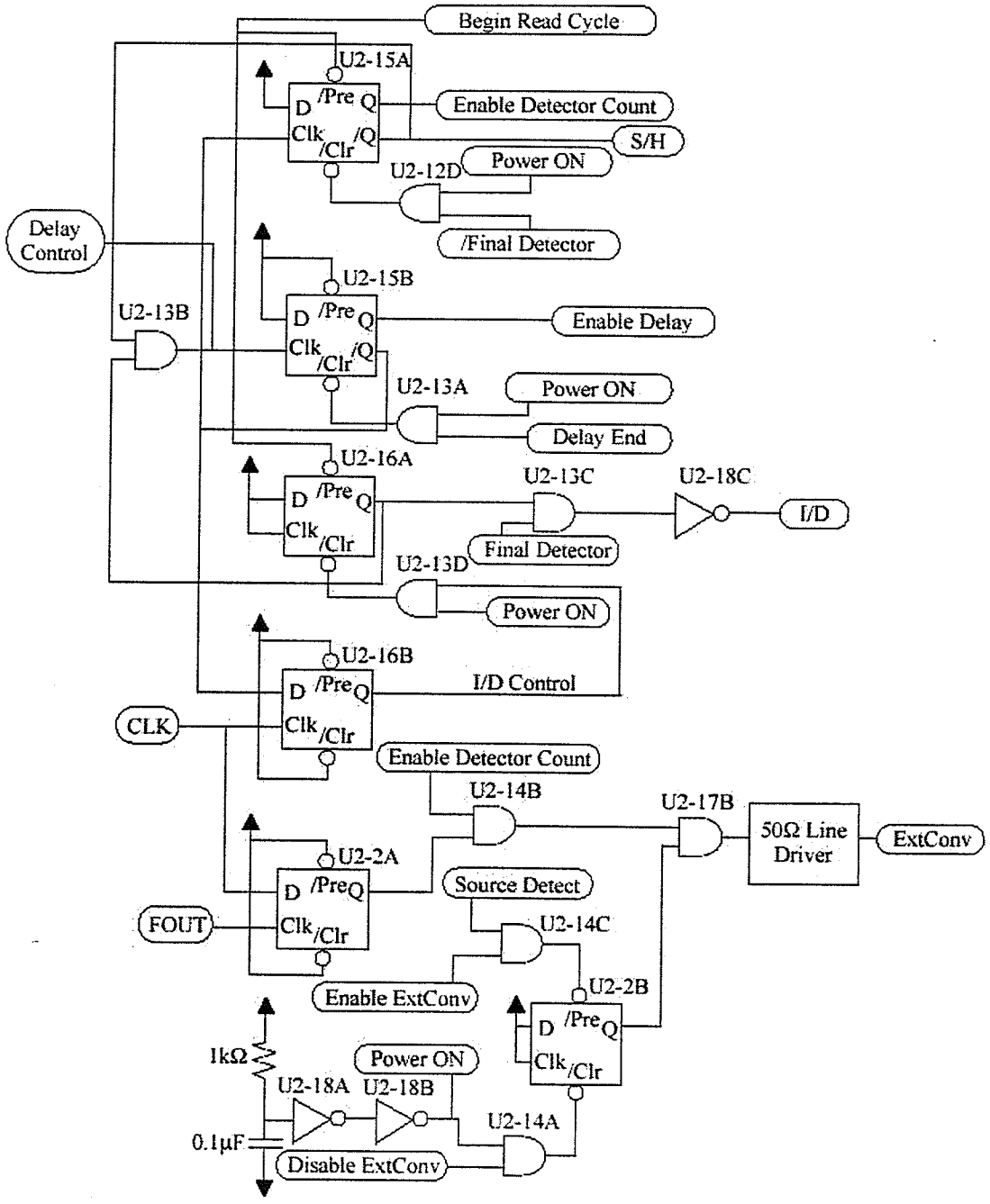

FIG. 5.7. Block diagram for the Timer Control Unit. Shown are the generators of important control signals in the circuit.

and asserts the EnableDetectorCount. A set of initial detector readings is taken during this period. During this time, the capacitors keep on charging.

5. At the terminal count of the time-of-acquisition counter, another hold pulse is generated by the S/H and the same detectors which were used for the initial reading are read again for their after-integration readings. The ExtConv pulses are enabled during this.

6. A FinalDetector signal is asserted after all the detectors are read. This ends one signal acquisition sequence. The S/H is put on sample again, the I/D put to discharge and the time-delay counter is enabled with the EnableDelay signal.

7. The embedded micro-controller then loads the next set of counter initializing data on to the Timer Control Unit. These include the number of the first detector to be read, the number of detectors to be read and the time delay.

8. Steps 4 through 7 are repeated for all dwell locations in a catheter.

9. Steps 2 through 8 are repeated for all the catheters used.

Figures 5.8 and 5.9 show signals generated by the fabricated timer control unit. The S/H and I/D signals are demonstrated for 8 ExtConv signals. These signals are generated while sampling a single analog detector assembly board. Each board has eight analog channels. The Analog Multiplexer Unit was thus not tested during this phase of experimentation.

Figure 5.10 was obtained to show and confirm that the I/D signal was indeed generated every two sets of ExtConv signals which are separated by $4.0ms$. This provides the time delay for the integration and a method to sample the discharged and integrated signals for the same set of detectors.

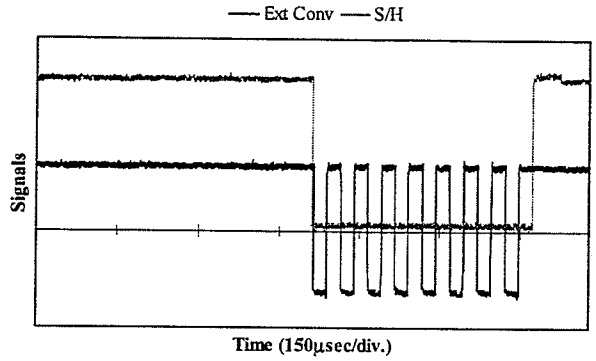

FIG. 5.8. The S/H signal with respect to eight ExtConv signals.

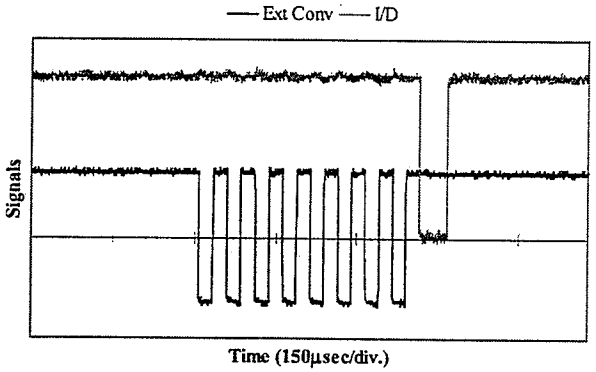

FIG. 5.9. The I/D signal with respect to eight ExtConv signals.

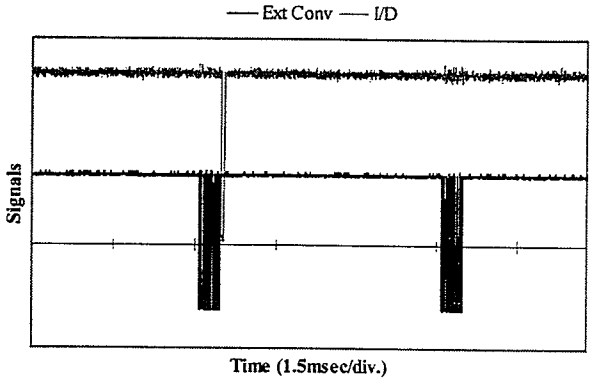

FIG. 5.10. The I/D signal is generated only once every two sets of ExtConv signals. This is different from the S/H signal which is generated every set of ExtConv signals.

5.3 The Analog Multiplexer Unit

The Analog Multiplexer unit is used to multiplex the 768 detector channels into a single channel which is fed to the analog input of the A/D converter housed in the host computer. 8 : 1 analog multiplexers (Motorola 14151) were used to accomplish the task. Figure 5.11 shows the schematics of the Analog Multiplexer unit.

The entire unit is implemented in four levels. The first level consists of the 768 : 96 stage which is implemented on-board of the analog detector assemblies (Chapter 4). The second, third and the fourth levels consist of the 96 : 12, the 12 : 2 and the 2 : 1 stages respectively (Figure 5.11). A 10-bit control signal is required to select a particular detector.

5.4 The Multiplexer Control Unit

The Multiplexer Control unit is used to generate the 10-bit control signal for the analog multiplexer unit (Section 5.3). When the detectors are being read during the holding pulse of `S/H`, the multiplexer control word (10-bit word) is incremented by each `ExtConv` pulse. The initial control word is the detector number of the first channel to be read.

The multiplexer control word is provided by a 10-bit synchronous counter shown in Figure 5.12. The counter is implemented using 4-bit $74LS163$ counters. The initial detector number is provided by the output port of the Port Controller Unit of the micro-controller (Chapter 6) either by the `SourceDetect` interrupt or the `BeginReadCycle` (Figure 5.5). It is on to the output lines of the counter by the `FinalDetector` signal (Figure 5.5).

The counter are wired to wrap around after a count of 767 so that after reading up-to channel number 767, the next channel read is 0. A 13-input NAND gate is used to generate a RESET signal on a count of 767.

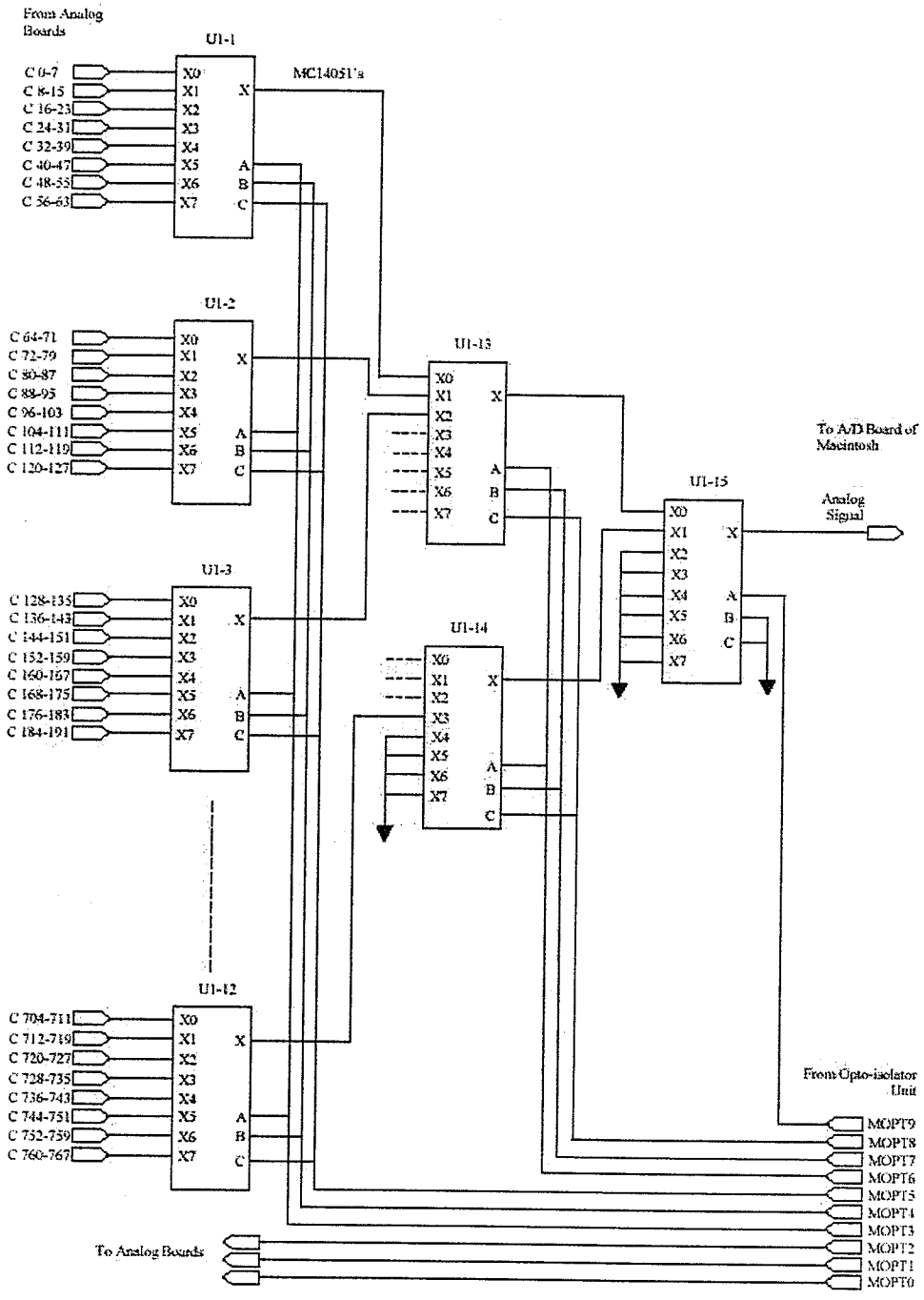

FIG. 5.11. The 96:1 Analog multiplexer unit which takes an input from every analog detector board and channels them through a single line to the A/D converter

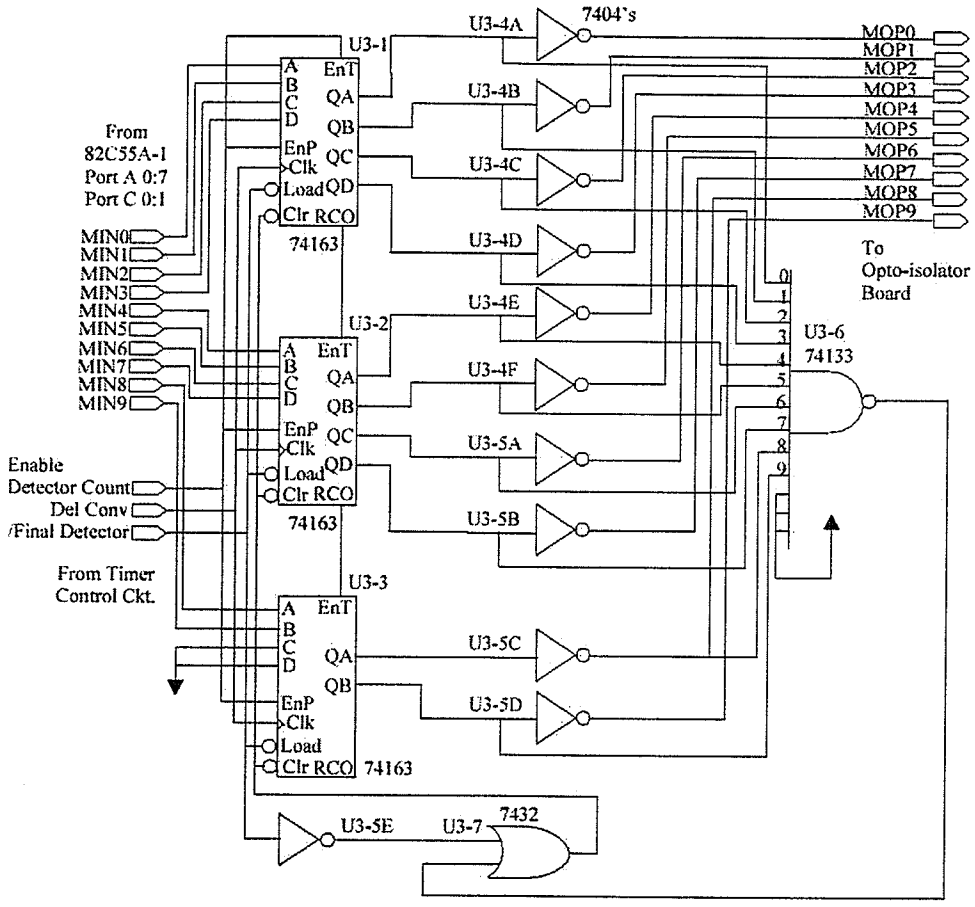

FIG. 5.12. The multiplexer control unit which generates the signal to identify which of the 768 channels is to be sampled and converted to digital form.

5.5 The Opto-isolator unit

The Opto-isolator unit is used to electrically isolate the digital portion of the hardware from the analog part. This is especially important in the eight channel detector assembly boards and the Analog Multiplexer Unit.

Switching signals generated by the digital control circuit is a major source of noise. Switching causes changes in the currents drawn in wires. These changes are prone to generate noise. As a result, these signals are unsuitable for use in the analog part of the hardware. These signals are therefore electrically isolated first before they

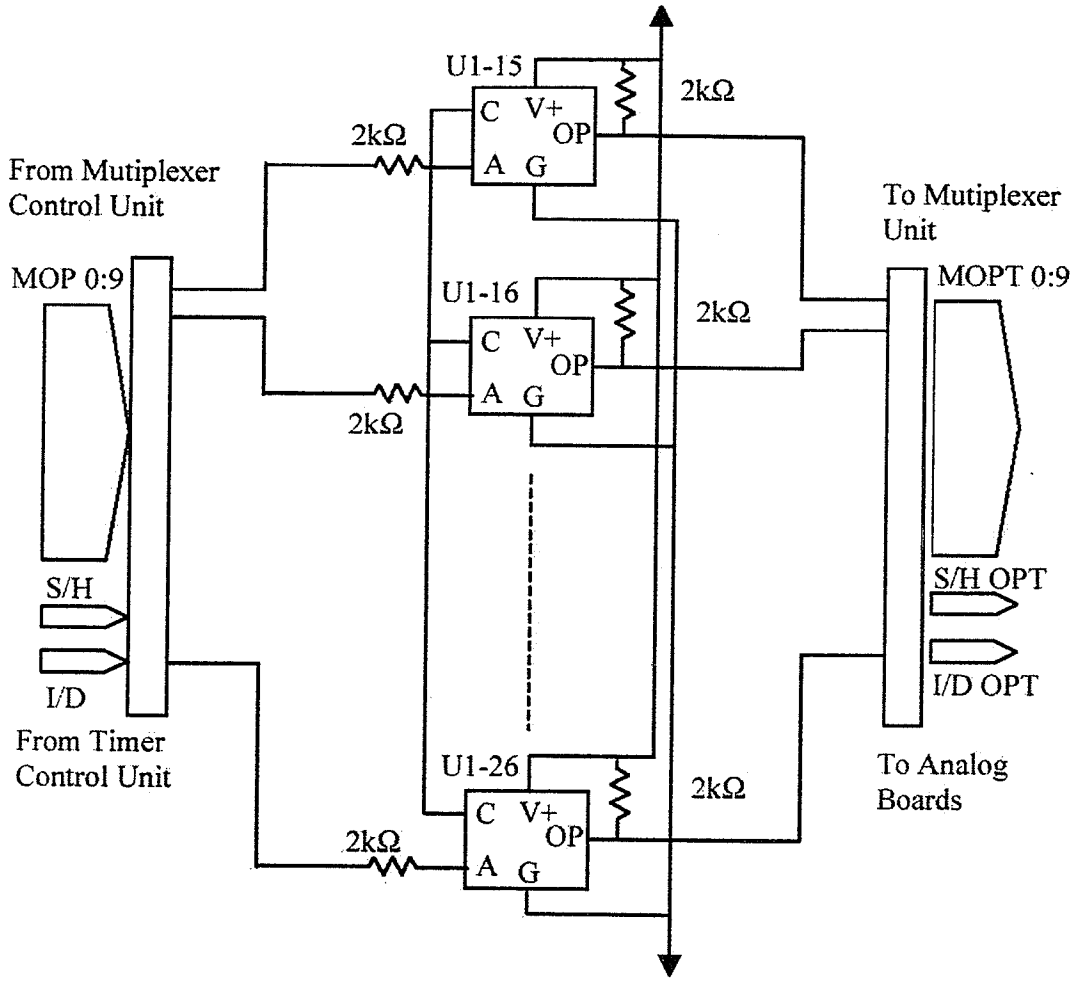

FIG. 5.13. The opto-isolator unit which electrically isolates signals between the analog and the digital halves of the hardware.

are provided to the analog ends.

Figure 5.13 shows the schematics of the opto-isolator circuit. Each opto-isolator unit has a light-emitting diode at the input and a photo-transistor at the output. This arrangement causes a complete electrical isolation between the input and the output. The opto-isolator unit is used to isolate the digital signals `MOP0:9`, the `S/H` and the `I/D` signals before they reach the analog circuit.

Chapter 6

THE SOFTWARE CONTROL MODULE

The analog and digital hardware modules discussed in chapters 4 and 5 respectively are software driven units. They react to controls generated by the software stage of the scanner. In this chapter the different units of the software control module are discussed. The software code implemented on the dedicated micro-controller unit is given in Appendix A. This work was accomplished jointly by Dr. Rickey and the author. Section 3.3 enumerates these contributions.

The designs of the analog and digital modules discussed so far point out to the emphasis laid on the timing requirements of the circuits. This is due to the fact the Data Acquisition System (DAS) of the CT scanner is a hard-real-time system. This means that the timing requirements for the signals generated in the hardware part of the DAS are very restrictive. The synchronicity of the control signals is essential for acquiring meaningful data. Any latency in any of the signals is inadmissible because it leads to spurious data.

The software control unit is designed to adhere to this time-restrictive nature of the DAS. Ordinarily, the host computer already in use would be used to generate the necessary control signals. However, the *Macintosh G3* computer running *MacOS* is unsuitable for this purpose because of the multi-tasking time-sharing nature of

MacOS. This is fairly true for all "off-the-shelf" Operating Systems available[1]. A dedicated micro-controller is therefore necessary to provide precisely timed control signals for the DAS hardware.

Section 6.1 describes the specifications of the *Tern i386-Engine* micro-controller unit used in the project. Section 6.2 describes the Port Controller Unit which serves as the interface between the micro-controller unit and the hardware control unit.

6.1 The *Tern I386-Engine* Embedded Micro-Controller Unit

The *Tern i386-Engine* micro-controller consists of a 32-bit microprocessor core based on a $33MHz$ Intel386EX chip. It is designed for embedded systems that require PC-compatibility, compactness and low power consumption.

The *i386-Engine* has several features to make it suitable for use as the micro-controller master unit which generates the necessary control signals for the DAS for our CT scanner. It features $512kB$ for battery-backed RAM and $128kB$ of ROM. This provides enough memory space to host the software control executable and the run-time parameters.

It also provides 16 external data lines and 26 address lines. These were used to load the different counter-parameters onto the registers of digital hardware module. It also features three 8-bit I/O ports which were used to transmit different control signals to the digital hardware. There are two PC-compatible $RS232$ asynchronous serial ports which are used for communication with the host computer. One port is used to download the parameters from the host computer. The second port is used to download the executable code onto the micro-controller. The second-port feature is of great convenience during test stage, because the executable code was kept on

[1] A possible solution would be to use DOS but that would make the computer unsuitable for other applications

the RAM module. The stable code is designed to be executed from a ROM chip.

The *i386-Engine* also provides a dedicated C/C++ Development Kit which provides a user-friendly code generating and debugging environment for a micro-controller programmer.

6.2 The Port Controller Unit

The port controller unit is used to direct pertinent data to the different counters on the hardware control unit (Chapter 5). The J1 header on the Tern Engine accommodates a 16-bit data line and an 8-bit address line. The address lines are used to encode the data into the different ports from where they are loaded onto the different counters:

1. The 12-bit dwell-time counter,
2. The 12-bit number-of-detectors counter,
3. The 10-bit start-detector counter, and
4. The 8-bit input encoded-detector-number.

The input mode of the 82C55A's used in the port controller unit are not inherently latched like the output mode. Therefore the input encoder on the source sensor encoder unit is latched before being sent to the Port Controller Unit. Figure 6.2 shows the schematics of the Port Controller Unit.

6.3 The Software Control Program

The code of the software control program implemented in this project is event-driven. The flow of control of the software code is given below:

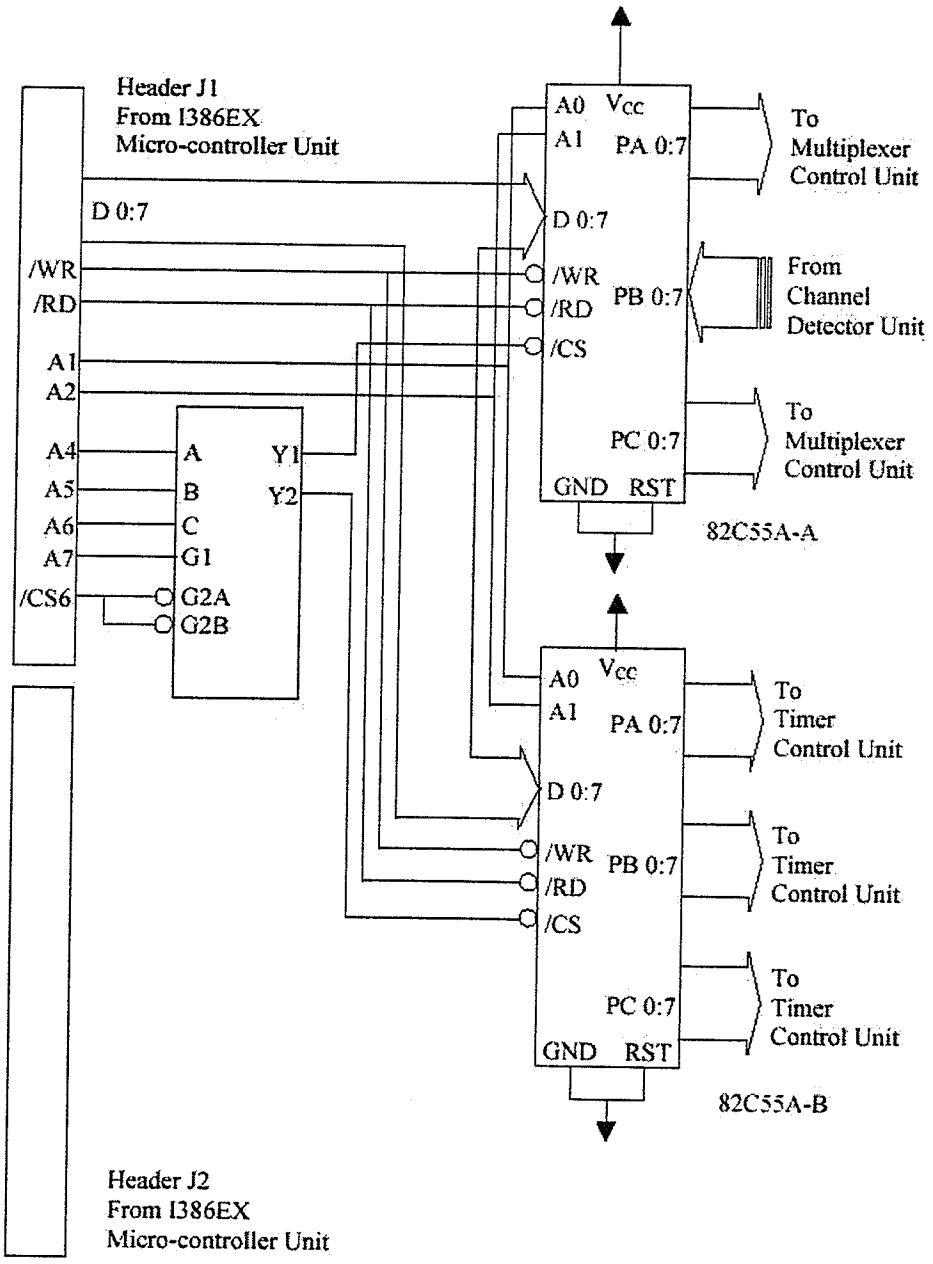

FIG. 6.1. The port controller unit showing the interfaces between the microcontroller unit and the hardware control unit.

1. All the internal registers of the *i386Engine* are initialized. This is provided as a function by the software toolkit provided with it.

2. All the hardware control parameters, the two $82C55$ ports and all the interrupt vectors are reset.

3. The acquisition clock on the A/D converter is disabled.

4. The source and the trigger interrupts are enabled.

5. The $RS232$ line parameters are initialized.

6. The counters and registers on the hardware control circuit are set to default parameters.

7. The program then sits in a `wait` state. Whenever an event triggers an interrupt, the interrupt is processed depending on the kind of interrupt it is. There are four possible events:

   (a) Parameter data may be coming in from the host computer.

   (b) A source may have been identified in one of the catheters.

   (c) A source may have just finished its dwell time at a dwell location and is moving on to the next location.

   (d) An error has occured and needs to be processed.

There are numerous errors which the software code handles internally. Appendix A gives a more detailed description of the error types. The next three sections explain the first three points of the aforementioned events.

6.3.1 Handling Data from the Host Computer

When an interrupt has been identified as `data-received-from-host-computer` type, the following checks and steps are taken. If the data passes all the checks, the information is regarded valid, and the data acquisition can begin. If the data fails any of the tests, the error is reported and the program quits.

1. The code looks for the word 'Eber'[2] in the first 4 characters of the data.
2. The code looks for the word 'CTac' in the next 4 characters to check if the data indeed corresponds to the CT acquisition data. Track is kept for the timing out of the $RS232$ connection.
3. The rest of the parameter data is loaded into the memory, always keeping track of whether the size of the data corresponds to what it should be (see Appendix A).
4. The counter values are loaded onto the ports of the Hardware Control Circuit. Interrupts are enabled. The system is now ready for CT data acquisition.

The counter values are set in the following manner:

1. The catheter number is retrieved.
2. The source position index in the catheter is retrieved.
3. The clocking frequency of the A/D clock is set.
4. The `number-of-detectors` counter is set according to the appropriate data.
5. The `starting-detector` counter is set.
6. The data is sent to the proper ports on the hardware control circuit.

[2]'Eber' is the name of the analog detector ckt. boards and serves as part of the header data

6.3.2 Handling Source Interrupts

When one of the source sensor circuits (Chapter 5) detects the presence of a source in the catheter, it sends a `source-interrupt` to the software unit. The sensor number is also sent encoded in the signal. When the code receives this signal, it performs the following steps:

1. It checks if an acquisition is already in progress. If it is, that means that there is more than one catheters with the source in it. The error is reported, and the program quits. Otherwise, the acquisition flag is set and it proceeds to next step.

2. The catheter number is checked with the number of the catheter where it is supposed to be. If there is an inconsistency, the error is reported, and the program quits.

3. If the previous checks are good, then the appropriate counter-value data is sent over the ports and the acquisition sequence can begin.

In order to enable interrupts, all pending interrupts are cleared before the interrupts are handled. In order to disable interrupts, the source and trigger interrupts are disabled and all pending interrupts are cleared.

6.3.3 Handling Trigger Interrupts

Within a single catheter, every time the source changes its dwell location, a trigger is generated. This interrupt handler works similar to the source-interrupt handler. However, these are two separate handlers, because the signals originate from different sources.

1. The number of source positions are acquired.

2. The counter values are transferred over to the hardware module.

3. If there are no more source positions left in the current catheter, the A/D clock is stopped.

4. The acquisition flag is reset, meaning that the next data acquisition sequence can commence.

Chapter 7

CONCLUSION AND FUTURE CONSIDERATIONS

7.1 Conclusion

Designing the hardware circuit which synchronizes the acquisition of attenuation data from a CT scan is a non-trivial task. Comprehensive knowledge is required in the fields of medical physics, analog and digital electronics design and software. The work done for this thesis demonstrates the intricate connections between different modules which perform their own specialized task in order to serve a common purpose.

The CT scanner built at Cancer Care Manitoba was divided into separate phases. The first task was to model the characteristics of ^{192}Ir source which acts the γ-ray source for the CT scanner. A prototype analog detector board was then designed and its signal characteristics were examined. This described by Berndt et al.[2].

The prototype was then ported into printed circuit boards to be used in the actual CT scanner and implemented (Chapter 4). There were a total of 96 boards which were hand-soldered and tested for noise characteristics. These characteristics closely resembled the noise figures reported by Berndt et al.[2].

The digital hardware control circuit used to generate the synchronising signals for the analog boards was designed and implemented as part of this thesis. The

hardware was designed to accomodate different parameters which need to be loaded during a data acquisition cycle.

The software interface which enables the scanning personnel to load and test the CT scanners was also designed.

7.2 Future Considerations

At the time of conclusion of this thesis work, there are numerous tasks which need to be accomplished in order for the CT scanner to be a functional unit.

- While the analog boards were tested individually, there needs to be further tests to determine the effectiveness of the boards when they are assimilated into the CT scanner frame. Such tests will not only evaluate the electrical noise characteristics of these boards but also help evaluate the collimation system of the scanner.

- The filtered backprojection algorithm (Chapter 2) which reconstructs an image from the acquired data still remains to be implemented.

- The whole system needs to be tested for multiple slices.

While there still remains work to be done, substantial progress has been made towards the realizationof a cost-effective CT scanner which can be used conjuctively with the HDR *Nucletron Afterloader* machine to accomplish better treatment planning for brachytherapy.

Appendix A

SOURCE CODE FOR SOFTWARE MODULE

```
// Default standard I/O header
#include <stdio.h>
// Default dynamic allocation header
#include <malloc.h>
// Memory allocation routines header
#include <alloc.h>
// Default math library
#include <math.h>
// Default MSDOS allocation header
#include <dos.h>
// I386EX initialization header
#include "ie.h"
// Serial Port# 0 initialization header
#include "ser0.h"
// Serial Port# 1 initialization header
#include "ser1.h"
// 80836EX definition header
#include "i80836ex.h"
// I386EX specific C low level commands header
#include "dos.h"

/********** Program constant definitions **********/
// boolean definitions
#define true 1
#define false 0

// crystal oscillator frequency
#define kBaseClockFrequency 20000.0
```

```
// port addresses
#define kCatheterPositionPort 0x8092
#define kOCW1 0xf021
#define kOCW2 0xf020

// event record types
#define kNoEvent 0
#define kRS232Event 1
#define kSourceInterrupt 2
#define kTriggerInterrupt 3
#define kError 4
#define kEventQueueSize 5

// error numbers
#define kNoErr 0
#define kCatheterNumErr 1
#define kOutOfMemErr 2
#define kWrongFileSize 3
#define kRS232TimedOut 4
#define kRS232EndOfBuffer 5
#define kRS232WrongMagicNum 6
#define kRS232WrongNumCatheters 7
#define kMultipleSourceInterruptsErr 8
#define kRS232Hopeless

// RS232 buffer sizes
#define MAXISIZE 2500
#define MAXOSIZE 2500

// port address definitions for 82C55's
#define kClockHighByte 0x80a0
#define kClockLowNibble 0x80a4
#define kNumDetectorsHighByte 0x80a2
#define kNumDetectorsLowNibble 0x80a4
#define kStartDetectorHighByte 0x8090
#define kStartDetectorLowNibble 0x8094
#define kClockStopAddress 0x80b0
#define kClockStartAddress 0x80c0

// catheter parameters
#define kMaxNumCatheters 18
```

```
#define kMaxNumSourcePositions 48
#define kBufferPadding 1
#define kDefaultDwellTime 0.1
#define kDefaultInitialDetector 0
#define kDefaultNumDetectors 7
#define kDefaultNumPositions 48

// RS232 constant definitions
#define kMagicNumber 0x72656245
#define kHDRCTacquisition 0x63615443
#define kRS232HeaderSize 3*4
#define kRS232Delay 15
#define kRS232OverHead 50
#define kRS232BaudRate 19200

/*********** Program structure definitions ***********/
typedef unsigned int boolean;

typedef struct EventRecord{
  // event that needs to be taken care of
  int currentEvent;
  int eventStart;
  int eventEnd;
  int  eventType[kEventQueueSize];
  int eventData[kEventQueueSize];
} EventRecord;

typedef struct EventRecord* EventRecordPtr;

typedef float DwellTimeType[kMaxNumSourcePositions + kBufferPadding];
typedef DwellTimeType* DwellTimeTypePtr;

typedef struct DetectorRange{
  int numPositionsUsed;
  int sourcePositionIndex;
  unsigned int initialDetector[kMaxNumSourcePositions + kBufferPadding];
  unsigned int numberDetectors[kMaxNumSourcePositions + kBufferPadding];
DwellTimeType dwellTime;
} DetectorRange;

typedef DetectorRange CatheterAcquisitionType[kMaxNumCatheters];
typedef CatherterAcquisitionType* CatheterAcquisitionPtrl;
```

```
typedef struct CTAcquisition{
  boolean acquisitionInProgress;
  unsigned int numCathetersUsed;
  unsigned int catheterNumber; // catheter with source in it
  int catheterIndex; // expected catheter
  unsigned int catheterOrder[kMaxNumCatheters];
  boolean firstTrigger;
  CatheterAcquisitionType catheterAcquisition;
} CTAcquisition;

typedef CTAcquisition* CTAcquisitionPtr;

// external variable from TERN libraries
extern COM ser1_com;
extern COM ser0_com;
typedef struct RS232Stuff{
  unsigned char ser1_in_buf[MAXISIZE];
  unsigned char ser1_out_buf[MAXOSIZE];
  unsigned char ser0_in_buf[MAXISIZE];
  usigned char ser0_out_buf[MAXOSIZE];
  COM *ser0;
  COM *ser1;
} RS232Stuff;

CTAcquisitionPtr gCTAcquisition;
EventRecord gTheEventRecord;

RS232Stuff gRS232;

// couple of internal check parameters
unsigned long gMemfree;
char gLedd;

// interrupt definitions
void interrupt far int0_isr(void); // source interrupt
void interrupt far int1_isr(void); // trigger interrupt

/********** Initialize RS232 Buffers **********/
void RS232Init( RS232Stuff* theRS232)
{
int isize, osize;
```

```
 usigned char baud1;

   // pointers to external structures
   theRS232->ser0 = &(ser0_com);
   theRS232->ser1 = &(ser1_com);
   isize = MAXISIZE;
   isize = MAXOSIZE;

   // receive DMA, transmit intr-driven noparity 8 bits, 1 stop bit
   baud1 = g; // set baud rate - default 19200
   s1_init(baud1, theRS232->ser1_in_buf, isize,
           theRS232->ser1_out_buf, osize, theRS232->ser1);
}

/********** Get Len bytes from serial 1 port **********/
int myGetSers1(COM* ComPort1, int len, unsigned char* theBytes)
{
int i = 0;
unsigned char ch;
int charsRead = 0;

 // loop through all the bytes
 while( i < len )
  {
    // make sure there are bytes to read int
    if(serhit(ComPort1))
      {
        ch = getser1(ComPort1);
        theBytes[i++] = ch;
        charsRead++;
      }
    else
      {
        break; // not enough bytes - bail out
      }
  }
  return(charsRead);
}

/********** Reset values of CT Acquisition Structures **********/
void ResetCTAcquisition(CTAcquisitionPtr theCTAcquisiton)
{
```

```
  int i,j;

   // set default values to 0 and number of catheters to 1
   theCTAcquisition->numCathetersUsed = 1;
   theCTAcquisition->catheterIndex = 0;
   theCTAcquisition->catheterNumber = 0;
   theCTAcquisition->acquisitionInProgress = false;
   theCTAcquisition->firstTrigger = true;

   for(i = 0; i < kMaxNumCatheters; i++)
     {
       theCTAcquisition->catheterAcquisition[i].sourcePositionIndex = 0;
       theCTAcquisition->catheterAcquisition[i].numPositionsUsed =
           kDefaultNumPositions;

       // set defaults: dwell times, initial detector# & # of catheters
       for(j = 0; j < kMaxNumSourcePositions + kBufferPadding; j++)
         {
           theCTAcquisition->catheterAcquisiton[i].dwellTime[j] =
              kDefaultDwellTime;
           theCTAcquisition->catheterAcquisition[i].initialDetector[j] =
              kDefaultInitialDetector;
           theCTAcquisition->catheterAcquisition[i].numberDetectors[j] =
              kDefaultNumDetectors;
         }
     }
}

/*********** Allocate memory and initialize variables ***********/
void InitVariables(void)
{
long bytesToAllocate;

bytesToAllocate = sizeof(CTAcquisition);
gCTAcquisition = (CTAcquisitionPtr)malloc((size_t)byteToAllocate);

if(gCTAcquisition == NULL)
  {
    gCTAcquisition = 0;
  }

  ResetCTAcquisition(gCTAcquisition);
```

```
   // initialize interrupt vectors
   outportb(0xf824, 0x0c);
   // data ctrl word for 82C55-1
   outportb(0x8096, 0x82);
   // data ctrl word for 82C55-2
   outportb(0x80a6, 0x80);
}

/*********** Add an event to the event queue ***********/
void AddEventToQueue(int theEventType, int theEventData)
{
  gTheEventRecord.eventEnd++;
  if(gTheEventRecordEvent.eventEnd >= kEventQueueSize)
    {
      gTheEventRecord.eventEnd = 0;
    }
  gTheEventRecord.eventType[gTheEventRecord.eventEnd] = theEventType;
  gTheEventRecord.eventData[gTheEventRecord.eventEnd] = theEventData;
}

/*********** Check for an event ***********/
boolean GetNextEvent(EventRecordPtr theEventRecord)
{
boolean eventPending;
boolean eventFound;
int i;

  // first need to check if there are any characters in RS232 buffer
  if(serhit1(gRS232.ser1))
    {
      i = theEventRecord->eventStart;
      eventFound = false;
      // check for event in queue
      while(i != theEventRecord->eventEnd)
        {
          eventFound = (theEventRecord->eventType[i] == kRS232Event);
          i++;
          if(i >= kEventQueueSize)
            {
              i = 0;
            }
        }
```

```
        if(theEventRecord->eventStart != kEventQueueSize)
          {
            theEventRecord->eventStart = 0;
          }
        // get current event
        theEventRecord->currentEvent = theEventRecord->
                eventType[theEventRecord->eventStart];
        eventPending = true;
      }
    else // no events to deal with
      {
        theEventRecord->currentEvent = kNoEvent;
        eventPending = false;
      }
     return(EventPending);
    }
 }

 /*********** Send an array of bytes down serial line ***********/
 void SendRS232Bytes(RS232Stuff* theRS232, char* buffer, int buffersize)
 {
 int i;
   for(i = 0; i < bufferSize; i++)
   {
     putser1(buffer[i], theRS232->ser1);
   }
 }

 /*********** Flush RS232 buffer: read it multiple times ***********/
 void FlushCom1Buffer(void)
 {
 unsigned char theChar;
   while(serhit1(gRS232.ser1)
     {
       theChar = getser1(gRS232.ser1);
     }
 }

 /*********** Stop main clock on clock board ***********/
 void StopAcquisitionClock(void)
 {
   outportb(kClockStopAddress, 0x00);
```

```
}

/*********** Start main clock on clock board ***********/
void StartAcquisitionClock(void)
{
  outportb(kClockStartAddress, 0x00);
}

/*********** Get RS232 data from host computer - MAC ***********/
void HandleDataFromMac(CTAcquisitionPtr theCTAcquisition)
{
typedef unsigned char* CharPtr;
int tempInt;
unsigned int numCatheters;
int i,j;
int byteToRead;
int bytesInBuffer;
unsigned long tempLong;
unsigned long acquisitionType;
unsigned long fileSize;
unsigned long timeOutCounter;
boolean timeOut;
float tempfloat[2];
float* myFloatPtr;
CharPtr myCharPtr;
unsigned long* a;
boolean dummy;

  bytesInBuffer = abs(gRS232.ser1->in_tail - gsRS232.ser1->in_head);

  // check if buffer has at least 4 characters
  if(bytesInBuffer < kRS232HeaderSize)
    {
      return(false);
    }

  // get first 4 bytes from serial buffer
  if(MyGetSers1(gRS232.ser1, 4, &(unsigned char)tempLong)
    {
      AddEventToQueue(kError, kRS232EndOfBuffer);
      return(false);
    }
```

```
// check for proper header tag
if(tempLong != kMagicNumber)
  {
    AddEventToQueue(kError, kRS232WrongMagicNum);
    FlushCom1Buffer();
    return(false);
  }

// get 4-byte acquisition type from header
if(MyGetSets1(gRS232.ser1, 4, &(unsigned char)acquisitionType)!=4)
  {
    AddEventToQueue(kError, kRS232EndOfBuffer);
    return(false);
  }
while(abs(gRS232.ser1->in_tail - gRS232.ser1.in_head) <
     (filesize - kRS232HeaderSize))
  {
    delay_ms(kRS232Delay);
    timeOutCounter++;
    // update RS232 structure
    dummy = serhit1(gRS232.ser1);
    if((timeOutCounter*kRS232Delay/1000) >
          (kRS232OverHead*filesize/kBaudRate))
      {
        timedOut = true
      }
  }

// check for RS232 reception time out
if(timedOut)
  {
    AddEventToQueue(kError, kRS232TimedOut);
    FlushCom1Buffer();
    return(false);
  }

// check if CT acquisition
if(acquisitionType == kHDRCTacquisition)
  {
    ResetCTAcquisition(theCTAcquisition);
    bytesToRead = sizeof(theCTAcquisition->numCathetersUsed);
```

```
// get number of catheters
if(MyGetSers1(gRS232.ser1, bytesToRead,
      &(unsigned char)numCatheters != bytesToRead)
  {
    AddEventToQueue(kError, kRS232EndOfBuffer);
    return(false);
  }
// sanity check for # of catheters
if(!((numCatheters > 0) && (numCatheters <= kMaxNumCatheters)))
  {
    AddEventToQueue(kError, kRS232WrongNumCatheters);
    FlushCom1Buffer();
    return(false);
  }
theCTAcquisition->numCathetersUsed = numCatheters;

// read in catheter order
bytesToRead = numCatheters *
     sizeof(theCTAcquisition->catheterOrder[0]);
for(i=0; i < numCatheters; i++)
  {
    if(MyGetSers1(gRS232.ser1, bytesToRead,
         &(unsigned char)theCTAcquisition->catheterOrder[0]))
      {
        AddEventToQueue(kError, kRS232EndOfBuffer);
        return(false);
      }
  }

// read in data for each catheter
for(i=0; i < numCatheters; i++)
  {
    // number of source positions per catheter
    bytesToRead = sizeof(
     theCTAcquisition->catheterAcquisition[i].numPositionsUsed);
    if(MyGetSers1(gRS232.ser1, bytesToRead, &(unsigned char)
    theCTAcquisition->catheterAcquisition[i].numPositionsUsed))
      {
        AddEventToQueue(kError, kRS232EndOfBuffer);
        return(false);
      }
    // initial detector# for each source position
```

```
for(j=0; j < kMaxSourcePositionsUsed; j++)
  {
    bytesToRead = sizeof(theCTAcquisition->
          catheterAcquisition[i].initialDetector[0]);
    if(MyGetSers1(gRS232.ser1, bytesToRead,
      &(unsigned char)theCTAcquisition->
        catheterAcquisition[i].initialDetector[j]))
      {
        AddEventToQueue(kError, kRS232EndOfBuffer);
        return(false);
      }
  }
// # of detectors for each source position
for(j=0; j < kMaxSourcePositionsUsed; j++)
  {
    bytesToRead = sizeof(theCTAcquisition->
      catheterAcquisition[i].numberDetectors[0]);
    if(MyGetSers1(gRS232.ser1, bytesToRead,
        &(unsigned char)theCTAcquisition->
           catheterAcquisition[i].numberDetectors[j]))
      {
        AddEventToQueue(kError, kRS232EndOfBuffer);
        return(false);
      }
  }
// dwell times for each source position
for(j=0; j < kMaxSourcePositionsUsed; j++)
  {
    bytesToRead = sizeof(theCTAcquisition->
      catheterAcquisition[i].dwellTime[0]);
    if(MyGetSers1(gRS232.ser1, bytesToRead,
      &(unsigned char)theCTAcquisition->
        catheterAcquisition[i].dwellTime[j])
      {
        AddEventToQueue(kError, kRS232EndOfBuffer);
        return(false);
      }
    else
      {
        theCTAcquisition->catheterAcquisition[i].
           dwellTime[j] = tempFloat[0];
      }
```

```
            }
          }
        return(true);
      }
    else
      {
        return(false);
      }
 }

 /*********** Send error number back to Mac ***********/
 void HandleError(EventRecordPtr theEventRecord)
 {
 int temp;

   temp = kError;
   SendRS232Bytes(&gRS232, &(char)temp, sizeof(temp));
   temp = theEventRecord->eventData[theEventRecord->eventStart];
   SendRS232Bytes(&gRS232, &(char)temp, sizeof(temp));
 }

 /*********** Interrupt service routines ***********/
 void ReenableIntr0(void)
 {
   outportb(kOCW2, 0x61); // reset intr0 to 0
   outportb(kOCW1, 0xfd); // reenable intr0
 }

 void interrupt far int0_isr(void)
 {
   outportb(kOCW1, inportb(kOCW1)|0x02);
   AddEventToQueue(kSourceInterrupt, kNoErr);
   gCTAcquisition->catheterNumer =
     (unsigned int)inportb(kCatheterPositionPort);
   outportb(kOCW2, 0x61);
 }

 void interrupt far int1_isr(void)
 {
   outportb(kOCW1, inportb(kOCW1)|0x20);
   AddEventToQueue(kTriggerInterrupt, kNoErr);
   outportb(kOCW2, 0x65);
```

```
    outportb(kOCW1, inportb(kOCW1) & 0xdf);
  }

  /*********** Write current values to counters ***********/
  void WriteCounterValues(CTAcquisitionPtr theCTAcquisition)
  {
  unsigned int clockValue;
  unsigned int numDetectorsValue;
  unsigned int startDetectorNum;
  unsigned int whichCatheter;
  unsigned int whichSourceIndex;
  unsigned int tempClockValue;

    // get current catheter and source position
    whichCatheter = (unsigned int)theCTAcquisition->catheterNumber;
    whichSourceIndex = (unsigned int)theCTAcquisition->
      catheterAcquisition[whichCatheter].sourceIndex;

    // set output clock rate
    clockValue = (unsigned int)(4095 - kBaseClockFrequency*
      theCTAcquisition->catheterAcquisition[whichCatheter].dwelltime[0];

    // set number of detectors used, starting detector#
    numDetectorsValue = (unsigned int)(4095 - theCTAcquisition->
      catheterAcquisition[whichCatheter].numberDetectors[bufferPadding];
    startDetectorNum = (unsigned int)(theCTAcquisition->
      catheterAcquisition[whichCatheter].initialDetector[bufferPadding];

    // write pertinent data to ports
    outportb(kClockHighByte, (unsigned char)(clockValue >> 4);
    tempClockValue = clockValue << 4;
    outportb(kNumDetectorsHighByte,
      (unsigned char)(numDetectorsValue >> 4);
    outportb(kNumDetectorsLowNibble, (unsigned char)
      ((tempClockValue & 0x00f0)+(numDetectorsValue >> 4)));
    outportb(kStartDetectorHighByte, (unsigned char)(startDetectorNum));
    outportb(kStartDetectorLowNibble,
      (unsigned char)(startDetectorNum & 0x0300)>>0);
  }

  /*********** Check for catheter# agreement ***********/
  void HandleSourceInterrupt(theCTAcquisitionPtr theCTAcquisition)
```

```
{
  // acquisition in progress
  if(theCTAcquisition->acquisitionInProgress)
    {
      AddEventToQueue(kError, kMultipleSourceInterruptsErr);
    }
  theCTAcquisition->acquisitionInProgress = true;
  if(theCTAcquisition->catheterNumber !=
     theCTAcquisition->catheterOrder[theCTAcquisition->catheterNumber])
    {
      AddEventToQueue(kError, kCatheterNumErr);
    }
  theCTAcquisition->catheterIndex++;
  if(theCTAcquisition->catheterIndex > kMaxNumCatheters)
    {
      theCTAcquisition->catheterIndex = kMaxNumCatheters;
    }
  WriteCounterValues(theCTAcquisition);
}

void HandleTriggerInterrupt(CTAcquisitionPtr theCTAcquisition)
{
int sourcePosition;
int desiredNumPositions;

  desiredNumPositions = theCTAcquisition->catheterAcquisition[
      theCTAcquisition->catheterNumber].numPositionsUsed;
  if(theCTAcquisition->catheterAcquisition[
     theCTAcquisition->catheterNumber].sourcePositionIndex)
    {
      theCTAcquisition->catheterAcquisition[theCTAcquisition->
        catheterNumber].sourcePositionIndex;
    }
 sourcePosition = theCTAcquisition->catheterAcquisition[
   theCTAcquisition->catheterNumber].sourcePositionIndex;
 WriteCounterValues(theCTAcquisition);
 if(theCTAcquisition->firstTrigger)
   {
     theCTAcquisition->firstTrigger = false;
   }
 else
   {
```

```
        theCTAcquisition->catheterAcquisition[theCTAcquisition->
          catheterNumber].sourcePositionIndex++;
      }
    if(sourcePosition >= desiredNumPositions)
      {
        StopAcquisitionClock();
        theCTAcquisition->acquisitionInProgress = false;
        theCTAcquisition->firstTrigger = true;
        theCTAcquisition->catheterAcquisition[theCTAcquisition->
          catheterNumber].sourcePositionIndex--;
        WriteCounterValues(theCTAcquisition);
        ReenableIntr0();
      }
  }

  /********** Initialize the interrupt handlers **********/
  void InitInterrupts(boolean enabled)
  {
    if(enabled)
      {
        // clear pending interrupts
        outportb(kOCW2, 0x61);
        outportb(kOCW2, 0x65);
        // enable interrupts
        init0_init(1, int0_isr);
        init1_init(1, int1_isr);
      }
    else
      {
        // disable interrupts
        int0_init(0, int0_isr);
        int1_init(0, int1_isr);
        // clear pending interrupts
        outportb(kOCW2, 0x61);
        outportb(kOCW2, 0x65);
      }
  }

  void main(void)
  {
    ie_init();
    InitVariables();
```

```
    StopAcquisitionClock();
    InitInterrupts(true);
    RS232Init(&gRS232);

    WriteCounterValues(gCTAcquisition);
    gMemfree = coreleft();
    led(1); // turn LED on

    // main event loop
    while(1){
      led(gLedd);
      gLedd = ~gLedd;

      // check for an received event
      if(GetNextEvent(&gTheEventRecord)){

        switch(gTheEventRecord.currentEvent){

          case kRS232Event:
            if(HandleDataFromMac(gCTAcquisition))
             {
               // write counter vals for initial n garbage readings
               WriteCounterValues(gCTAcquisition);
               // enable interrupts for acquisitions
               InitInterrupts(true);
             }
            break;
          case kSourceInterrupt:
            HandleSourceInterrupt(gCTAcquisition);
            break;
          case kTriggerInterrupt:
            HandleTriggerInterrupt(gCTAcquisition);
            break;
          case kError:
            HandleError(&gTheEventRecord);
            break;
        }
      }
    }
  }
}
```

REFERENCES

[1] L.L. Anderson. *Dosimetry for Interstitial Brachytherapy.* Publishing Science Group, 1975.

[2] A. Berndt, S. Rathee, D.W. Rickey, and J. Bews. An 8-channel detector for an ^{192}ir brachytherapy source-based computed tomography scanner. *IEEE Transaction on Nuclear Science*, 74:1261–1267, 2000.

[3] S.C. Bushong. *Computed Tomography.* McGraw-Hill/Appleton and Lange, 1^{st} edition, 2000.

[4] L. Cionini and R. Santoni. Treatment planning by computed tomography in curie-therapy of invasive cervical cancer. *Cervix*, 8:277–289, 1990.

[5] D.J. Demanes, R.R. Rodriguez, D.D. Bendre, and T.L. Ewing. High dose rate transperineal intersitial brachytherapy for cervical cancer: High pelvic control and low complication rates. *Internation Journal of Radiation Oncology and Biological Physics*, 45(1):105–112, 1999.

[6] R. Gordon, R. Bender, and G.T. Herman. Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography. *Journal of Theoretical Biology*, 29:471–481, 1970.

[7] P.C. Griffin, P.A. Amin, P. Hughes, A.M. Levine, W.W. Sewchang, and O.M. Salazar. Pelvic mass: Ct-guided interstitial catheter implantation with high-dose-rate remote afterloader. *Radiology*, 191:581–583, 1994.

[8] W.E. Higgins and D.C. Munson. A hankel transform approach to tomographic image reconstruction. *IEEE Transactions on Medical Imaging*, 7(1):59–72, 1988.

[9] K.S. Kapp, G.F. Stuecklschweiger, D.S. Kapp, and A.G. Hackl. Dosimetry of intracavitary placements for uterine and cervical carcinoma: results of orthogonal film, tld, and ct-assisted techniques. *Radiotherapy and Oncology*, 24:137–146, 1992.

[10] F.M. Khan. *The Physics of Radiation Therapy.* Lippincott Williams and Wilkins, 2^{nd} edition, 1994.

[11] L.L. Meisberger, R. Keller, and R.J. Shalek. The effective attenuation in water of the gamma rays of ^{198}au, ^{192}ir, ^{137}cs, ^{226}ra, and ^{60}co. *Radiology*, 90:953–957, 1968.

[12] T.H. Newton and D.G. Potts, editors. *Radiology of the skull and brain: Technical Aspects of Computed Tomography.* The C.V. Mosby Company, 1981.

[13] National Cancer Institute of Canada. Canadian cancer statistics, 2002.

[14] British Journal of Radio Supplements. Radionuclides in brachytherapy: Radium and after, 1990.

[15] F. Perera, F. Chisela, J. Engel, and V. Venkatesan. Method of localization and implantation of the lumpectomy site for high-dose-rate brachytherapy after conservative surgery for t1 and t2 breast cancer. *International Journal on Radiation Oncology and Biological Physics*, 31(4):959–965, 1995.

[16] B. Pierquin and G. Marinello. *A Practical Manual of Brachytherapy.* Medical Physics Publishing Corp, 1997.

[17] R.R. Rodriguez, D.J. Demanes, and G. Altieri. High dose rate brachytherapy in the treatment of prostate cancer. *Hematology/Oncology Clinics of North America*, 13(3), 1999.

[18] L.E. Romans. *Introduction to computed tomography.* Lippincott Williams and Wilkins, 1^{st} edition, 1995.

[19] M.C. Schell, C.C. Ling, Z.C. Gromadzki, and K.R. Working. Dose distribution of model 6702 ^{125}i seeds in water. *International Journal of Radiation Oncology and Biological Physics*, 13:795–799, 1987.

[20] H.J. Scudder. Introduction to computer aided tomography. *Proc. IEEE*, 66:628–637, 1978.

[21] L.A. Shepp and B.F. Logan. The fourier reconstruction of a head section. *IEEE Transactions on Nuclear Science*, 21:21–43, 1974.

[22] G.F. Stuecklschweiger, K.S. Arian-Schad, Poier E., J. Poschauko, and A.G. Hackl. Bladder and rectal dose of gynecologic high-dose-rate implants: comparison of orthogonal radiographic measurements with in vivo and ct-assisted measurements. *Radiology*, 181:889–894, 1991.

[23] K.A. Weaver, V. Smith, D. Huang, C. Barnett, M.C. Schell, and C.C. Ling. Dosimetry on the transverse axis of ^{125}i and ^{192}ir seed sources. *Medical Physics*, 16:636–643, 1989.