A THREE-DIMENSIONAL KINEMATIC ACQUISITION
AND INTERSEGMENTAL DYNAMIC ANALYSIS SYSTEM
FOR HUMAN MOTION

by

Erik Karl Antonsson

B.S. Cornell University
(1976)

S.M. Massachusetts Institute of Technology
(1978)

SUBMITTED TO THE DEPARTMENT OF MECHANICAL ENGINEERING
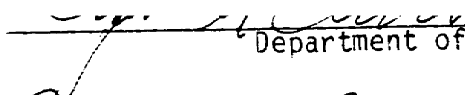IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 1982

Signature of Author _____
Department of Mechanical Engineering
May 19, 1982

Certified by _____
Professor Robert W. Mann
Thesis Supervisor

Accepted by _____
Chairman, Department Committee on Graduate Students

Archives

A Three-Dimensional Kinematic Acquisition
and Intersegmental Dynamic Analysis System
for Human Motion

by

Erik Karl Antonsson

Submitted to the Department of Mechanical Engineering on May
19, 1982, in partial fulfillment of the requirements for the
Degree of Doctor of Philosophy in Mechanical Engineering

ABSTRACT

A system for automatic accumulation and reduction of
three-dimensional, human kinematic data has been designed
and implemented which avoids the questionable and
non-physiologic gait kinematics and forces often produced by
contemporary techniques. The new system introduces
three-dimensional, linkage analysis methods to describe
completely the kinematics and estimate total moments and net
forces between links, including a momentary axis-of-rotation
solution approach to eliminate subjective location of "joint
centers." Kinematic data acquisition is accomplished by an
infra-red, opto-electronic device with thirty LED markers
sampled by two cameras. A calibration scheme produces
10-bit accuracy. The three-dimensional point reconstruction
is accomplished photogrammetrically. Sets of markers are
grouped in segment-oriented, rigid arrays each with an
imbedded body coordinate system (BCS). A least-squares,
best fit is used to resolve the rotation matrix for each
BCS. A piezo-electric, multi-component force platform
synchronously acquires a force reference in the lower
extremity, multi-segmented, kinematic chain. Kinematic data
are significantly improved over conventional gait analysis
techniques by a 315-Hz sampling rate, 1-mm and
20-milliradian resolution and accuracy and an improved
marker-array mounting system employing stable anatomical
mounting locations. The system also operates in real-time
at 100 Hz. The frequency domain response of the overall
system, as well as several classes of human motion, is well
characterized, providing for intelligent selection of
low-pass filter cutoff frequencies for noise elimination.

The first amplitude peak of noise is at more than twice the highest frequency contained in normal gait, and the residual noise contributes an error of less than 5 percent to the double differentiated kinematic data. The system represents an improvement over other gait and mobility analysis techniques in speed, accuracy, objectivity, and automaticity; further its noise and error properties are characterized and insure substantial improvement in the confidence of dynamic estimates.

Several studies of measured motion and estimated forces have been undertaken, both for machines (as demonstrations of accuracy) and humans. Total moments and net loads across the hip, knee, and ankle joints are presented for normal human gait.

Thesis Supervisor:
Prof. R. W. Mann
Title: Whitaker Professor of Biomedical Engineering

Thesis Committee Chairman:

Professor R. W. Mann
Department of Mechanical Engineering
Massachusetts Institute of Technology

Thesis Committee Members:

Prof. D. L. Bartel
Sibley School of Mechanical and Aerospace Engineering
Cornell University

Prof. D. E. Hardt
Department of Mechanical Engineering
Massachusetts Institute of Technology

Prof. T. A. McMahon
Division of Engineering and Applied Physics
Harvard University

Prof. W. P. Seering
Department of Mechanical Engineering
Massachusetts Institute of Technology

ACKNOWLEDGEMENTS

Neville, for introducing me to M.I.T. in ways that would have taken me years to find if left to myself, as well as introducing me to the Paradise and its crowd: Woodie, Gossard, Adam, Looie, ... And also for instilling in me a healthy distrust of computers.

My many office mates: Dave Hardt, Paul Estey, Jon Kaplan, Don Grimes, ...

Dave (Looie) Lewis, for convincing me many years ago that it really was possible to finish a Ph.D. (eventually).

Ann Rappaport for helping to convince me that not only was it possible to finish, but that I should actually do it.

Seth Lichter for helping to keep me in shape, and entertaining me through our daily workouts.

My parents not only (obviously) made this possible, and put me up to it all, but provided encouragement, consolation, and support (and eventually: needling, towards the end) that kept me going. I would like to recognize their special efforts and contributions to all of my schooling, and also for being two of the finest (albeit occasionally aggravating) people I know. How can I thank you?

Finally, and perhaps most importantly, to Barbara. I couldn't have done it without you.

The text was written, edited and formatted on the MIT Joint Computer Facility's VAX 11/782 running under VMS Version 3.0, using DEC Standard Runoff and printed on a QUME Sprint S9/45 with a Letter Gothic 12 point font.

All of the figures were drawn wholly or in part by a Hewlett-Packard 9872A 4 color pen plotter, driven by a National Instruments GPIB11-1 Unibus to IEEE-488 interface, mounted in a DEC PDP 11/60, controlled by a graphics software package written by the author. Slobodan Tepic built a set of special technical pens for the plotter to produce drafting quality lines.

Table of Contents

VOLUME I

VOLUME II

CHAPTER 1 FIGURES

CHAPTER 2 FIGURES

CHAPTER 3 FIGURES

CHAPTER 4 FIGURES

CHAPTER 5 FIGURES

CHAPTER 3 PHOTOGRAPHS

CHAPTER 4 PHOTOGRAPHS

CHAPTER 5 PHOTOGRAPHS

CHAPTER 1 TABLES

CHAPTER 3 TABLES

CHAPTER 4 TABLES

CHAPTER 5 TABLES

CHAPTER 1

INTRODUCTION

Walking (or more generally mobility) is a process most of us perform with little conscious effort, but this ease obscures the underlying complexity of normal motions. Gait alone involves the recruitment and control of hundreds of muscles in the torso and lower extremities to produce the actions of the multi-segmented skeleton. Scientists have been studying human motion for centuries, and a multitude of disciplines have been involved: mechanics, kinematics, dynamics, kinesiology, neurology, physiology, and anatomy. If muscle function is included, chemistry, cell biology, and thermodynamics also come into play. Ethical restrictions on use of humans as experimental subjects and deleterious influence on natural motion both prohibit invasive motion or force measurements, so an additional level of complexity is introduced. Virtually all mobility analysis techniques only measure external kinematics, foot-floor force interaction, and collect myo-electric signals (MES). Commercially available forceplatforms simplify the foot-floor interaction measurements, and MES, from skin-mounted electrodes, is only available from the superficial muscles to verify the temporal aspects of estimated muscle activity. For the

study reported here the most critical measurement problem is that of determining the kinematics of motion.

> "Formally, kinematics is that branch of mechanics which treats the phenomenon of motion without regard to the cause of the motion. In kinematics there is no reference to mass or force; the concern is only with relative positions and their changes."
> Bottema and Roth 1979 [15] page vii

Since motion analysis mandates the involvement of a wide range of disciplines, not surprisingly many individuals have investigated human motion: Aristotle, Leonardo DaVinci, E. and W. Weber (1836), M. G. Carlet (1872), Eadweard Muybridge (1877), Jules Etienne Marey (1887), Braune and Fischer (1889) [16], Schwartz and Heath (1932), Herbert Elftmann (1938) [56], Bresler and Frankel (1950) [17] limiting for now the list to the pioneers. One might assume that the years since the initial quantitative studies of the Weber brothers in 1836 would have produced substantial insight into human motion and the principles of its production. Certainly there has been no paucity of interest or publication. Only recently, however, has sufficiently sophisticated technology been applied, with the consequences that most of the earlier studies were either qualitative, or purely planar (and not spatial), or a thorough understanding of the measurement technique and its limitations were not appreciated. The continuing (and heated) debate amongst prominent human mobility researchers

concerning the frequency content of gait (possibly the most fundamental aspect of the phenomena from a measurement standpoint) stands as testimony to the limited knowledge, confusion, and formative state of the field.

Given knowledge of the inertial properties of the body segments of interest, three dimensional dynamic equations can be solved, starting with the input kinematics, and producing the net force and moment vectors that must have existed as a function of time to produce the observed motion. These forces and moments include no information concerning the joint motion actuators: The individual muscle action, tendons and ligaments, joint capsules, and tissue restraints are wholly aggregated into the moment vectors, which appear as hypothetical, pure moment generators about the joints. These dynamic results can be input, however, to the determination of principles of selection of muscle activation in order to ultimately estimate the total joint forces due to inertial and muscular effects. (Hardt [66,67,68], Patriarco [122], Crowninshield [37,38,39,40,41])

"To assess the total force acting through the head
of the femur at any time during the normal walking
cycle it is necessary to determine the tension
developed in the individual muscles which act on
the hip-joint. No technique for the direct
measurement of individual muscle tension is at
present available. A quasi quantitative
assessment could be arrived at indirectly through
the simultaneous correlation of torque at the
hip-joint and the electrical activity of the
muscles involved."
Sorbie and Zalter, in Kennedi, ed. [146] page 359

Several techniques have evolved to acquire and reduce
motion data; reduction of these data to the most
fundamental form produces estimates of aggregate muscle
force activity. These are of central interest to
understanding neurological control, muscle function, joint
mechanics, pathologies, and mobility rehabilitation. All
extant techniques have limitations, and these to-date have
precluded accurate estimates of joint loads and muscle
forces. In the 1978 publication of R. D. Crowninshield, et
al., [39], for example, the hip joint contact force vector
has a significant component directed laterally out of the
hip joint during some of the gait cycle. The lateral edge
of the acetabular socket only extends 5 to 10 degrees beyond
the vertical, a load vector close to the edge will place a
very small rim of cartilage under large loads, and a vector
directed beyond the edge will disarticulate the joint.
Experiments conducted in the Laboratory for Biomechanics and
Human Rehabilitation at M.I.T. have shown that the load can
be directed at most 5 degrees laterally in the majority of

normal joints. The medial-lateral component of hip joint contact force is critical to the contact pressure distribution, as demonstrated by the differences shown in pressure distribut on from a cadaverous hip subjected to 10 degree medial increments in load, Figure 1. Such comparisons underscore the necessity of producing estimates of joint load vector with a confidence bound smaller than 5 degrees.

PRESSURE CONTOUR MAP

PMAX=5.4 MPa
PAVG=1.8 MPa
AREA=775 mm 2
25° MED/10° ANT

PMAX=7.3 MPa
PAVG=2.3 MPa2
AREA=619 mm
15°MED/10°ANT

PMAX=11.4 MPa
PAVG=3.6 MPa2
AREA=410 mm
5° MED/10°ANT

LOAD=1350 N   SPECIMEN 126   TIME=30 sec
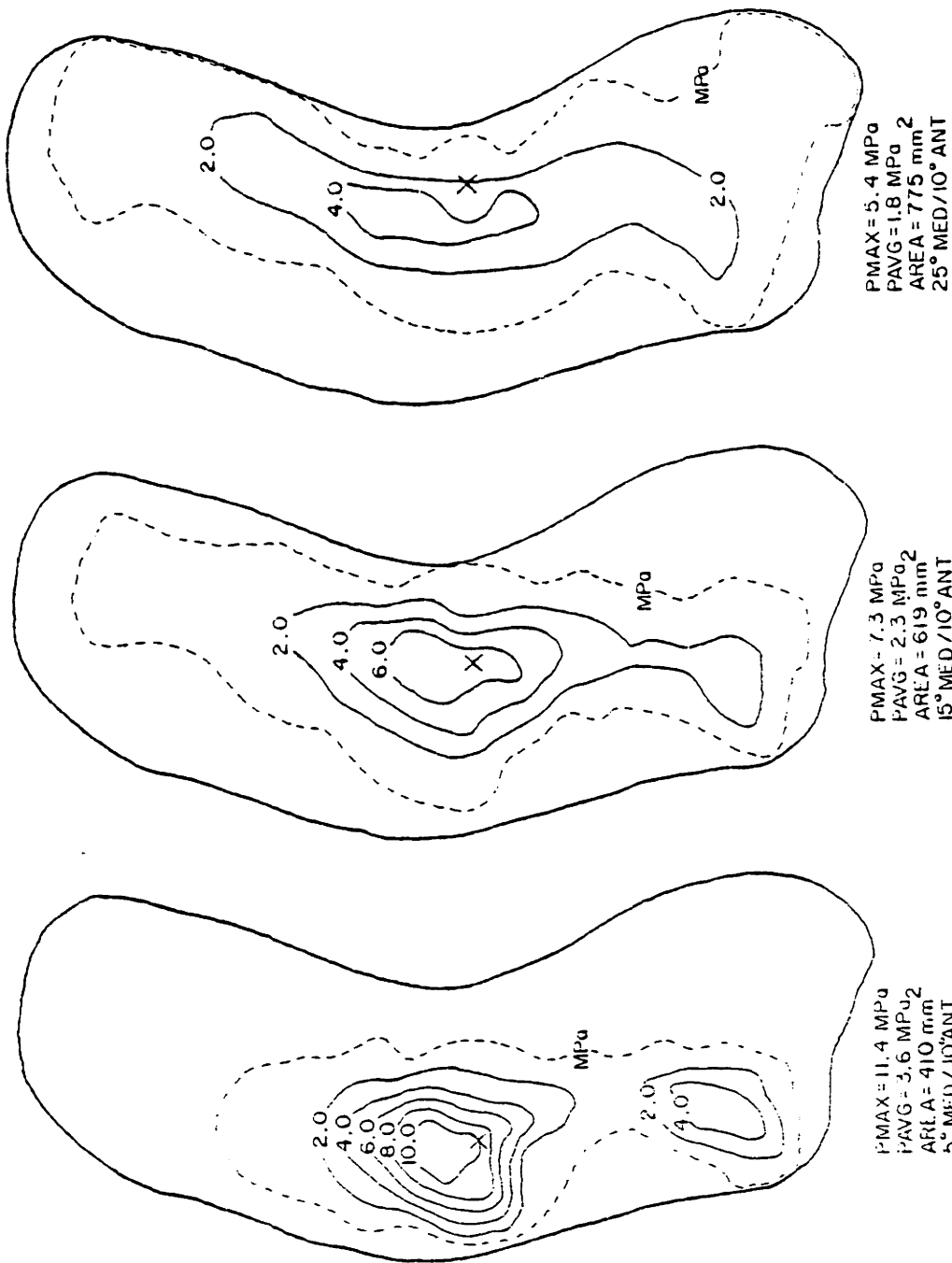
Figure 1. Hip pressure plots with load vectors varying laterally

As another illustration, Kettelkamp (1970 page 779 [87]), in an analysis of the complicated trajectory of the instant center of rotation of the human knee as the articulation undergoes rotation and sliding on complex cartilagenous surfaces notes that: "... the instant centers of knee motion ... were within a circle of 12.7 millimeter radius." Thus to track motions of the knee joint axes, the resolution of the kinematic acquisition and analysis system must be substantially smaller than a centimeter.

To underscore the need for full three-dimensional analysis, Eberhart, et al., (1968 page 460 [53]) notes that "The average (axial) rotation of the pelvis is 5 degrees, and the tibia and femur 9 degrees." "It is seen, then, that locomotion is a complex process involving displacement of the parts of the body in three directions and with 3 degrees of rotational freedom. Any discussion of locomotion involving less than these 6 degrees of freedom would of necessity be incomplete." (page 470) Despite this conjecture and probably because of all the spatial motions, axial rotations of segments are the hardest to measure and rotations about the long axes of segments are often ignored.

Any critique of the joint force estimation process easily identifies a multitude of areas of possible introduction of errors. A natural division of the whole process separates the kinematic acquisition and net motion

producing force and moment estimation from the redundant muscle optimization solution which is necessary to transform the torques estimated in the first estimation segment into muscle forces (including agonist-antagonist coactivation) and total joint loads. This thesis is concerned with the first part of the overall process; however, other investigators (Patriarco, et al., 1981 [122]) have determined that the influence of errors generated by input kinematics outweigh the influence of choice of muscle optimization scheme and have suggested that better kinematics and net load estimates will improve the capacity to identify individual force-time activation information.

In addition to a force plate, Crowninshield uses a photographic technique similar to Jules Etienne Marey's 1873 invention of "chrono-photography." A rotating disk with holes interrupts light projected onto a single frame of film, producing multiple images separated in time. The majority of contemporary mobility and gait analysis data transduction techniques are similar to this early effort and rely on systems that have benefitted very little from the technological advances of the last century. In fairness, Crowninshield's data are the final result of kinematic acquisition, net load estimation, and muscle force optimization; however, my conversations with him lead me to suspect that errors in kinematics are strongly influencing his final results. For instance, he claims a positional

resolution of 2.5 to 5.0 mm, using a

Only 1 mm of noise at 15 Hz (the hig

possibly be reconstructed with 30 Hz

accelerations of almost 1 gravity

smoothing to ameliorate this effect;

apply a smoothing filter, both the f

phenomena, as well as the noise,

Crowninshield does not know the nois

measurement system.*   That system use

hold three visible light LEDs each,

circuitry and a 9 volt "transistor

battery alone weighs 0.45 Newtons

motion of the LEDs with respect to the

The objective of my work is to de

must be measured and how well th

provide more physiologically valid

loading;   to design and build a c

objective, instant turn-around syste

estimates;   and to evaluate quantit

confidence on the results.

---

* Personal communication October 28,

NON-INVASIVE SPATIAL POSITION MEASU

To estimate the <u>net</u> joint loads (a for

moment vector) during human motion as a

good kinematic data are required. The

implies:

1) Full three-dimensional segment

orientation measurement.

"... there are only a few joints in th
that could be classified as stri
joints. When disease is involved,
planar joints will experience three-
deformity which requires a truly spati
accurately define the morphologi
involved."
Chao [36] page 386

2) High enough kinematic accuracy a

resolution to influence the estimates

criterion, say 5 percent.

3) Frequency of position sampling (for

system) above twice the highest f

phenomena of interest, <u>as well as</u> more

highest frequency of the significant mea

If each limb segment were a rigid link,

problems would end here; however, the

Each segment is comprised of muscle, l

cartilage, bone, and fluid. Producing tor

results in changes of the center of mas

respect to the skeleton, changes in the inertia tensor for the segment, and changes in the shape of the surface of the segment. Changes in the limb position and orientation with respect to another limb usually result in a different location and direction of the axis of rotation between them. Lastly, since the measurement subject is a living organism, special efforts must be taken to minimize the influence of the experiment on the subject's motion.

REVIEW OF CURRENT MOBILITY ANALYSIS TECHNIQUES

To put this thesis in perspective, a review of the measurement problem and the major techniques for accumulating and processing human motion data will be presented, along with the advantages and limitations of each, as well as an explanation of the design criteria for our system. For a more detailed description of each technique see:

Chao, E. Y.,"Experimental Methods for Biomechanical Measurements of Joint Kinematics," in: B. N. Feinberg, and D. G. Flemming, eds., CRC Handbook of Engineering in Medicine and Biology, Section B: Instruments and Measurements, Volume I, Florida; CRC Press, pages 335-411. [36]

Four major techniques are used for the accumulation of human motion kinematic data:

1. Electromechanical linkages (electro-goniometers)

2. Accelerometry

3. Roentgenography

4. Stereometry

ELECTRO-GONIOMETERS consist of an articulated exoskeleton with angular (and occasionally translational) transducers (usually potentiometers) at the joints. The exoskeletal linkage is attached to the skin of a subject, and the physiological joint angles are inferred from the measured angles of the goniometer. Goniometery is advantageous in that the instrumentation system is relatively easy to use, providing direct measurement of joint angle (often the object of motion measurement), and the mechanism is not subject to obscuration as are optical techniques. Difficulties arise with normal and pathological joints, which include translational as well as rotational components of motion, and with mis-alignment of the exoskeletal joint axes with the physiological axes. Motion of the soft tissue, to which the goniometer is attached, with respect to the skeleton introduces error. In general, goniometers are massive compared to optical targets or sources and the dynamics of the linkage become important. The mass and friction associated with an exoskeletal

structure can affect normal motion significantly. Lastly the goniometer only measures _relative_ intersegmental limb motion, and an external inertial reference is needed to produce absolute motion data and joint force estimates.

ACCELEROMETRY involves the measurement of a set of at least six accelerations of a segment of interest, and it has been suggested by Padgaonkar (1975) [119] that nine are necessary for stability. A variety of accelerometer types has been used, including strain-gauge and piezo-electric. Double integration of accelerations yields position histories to within two integration constants, avoiding the noise amplification suffered with differentiation. If the initial position and velocity are known, the solution can be specified. Ascertaining the initial positions is a relatively easy task; however, there is a more insidious problem associated with the integration. Local segment accelerations, not the influence of gravity, are required to produce correct kinematics; however, the orientation of the segment is required to remove the gravity vector, but the orientation is the desired result. Stated simply: The answer is required in order to calculate the answer.

ROENTGENOGRAPHY exposes subjects to X-Ray radiation, and, particularly in cinetographic applications, this exposure is excessive and unacceptable. The method does have advantages, however, in that the internal skeletal

kinematics can be measured directly instead of inferred from the outside motions. Beyond this roentgenography is substantially the same as visible light cine-techniques discussed under Stereometry.

STEREOMETRY is a common photogrammetric technique employing two cameras. Each camera collects azimuth and elevation data on several points. With knowledge of the camera locations and orientations, the azimuth and elevation data can be used to reconstruct analytically the vector from each lens front nodal point to the points of interest in the field of view. A simple trigonometric technique will yield the three-dimensional location of each point of interest with respect to the cameras. If a minimum of three points is in a previously specified rigid array, the position and orientation of a body-coordinate system fixed relative to the rigid point array can be found. Thus stereometry can provide full six-degree-of-freedom kinematic information on several bodies. If the camera data are collected continuously, or if they are sampled at a high enough rate, motion data can be measured.

The moving points of interest can either be reflectors or emitters, depending on the measuring hardware. For high-speed, motion-picture-film cameras, the points can be small light bulbs, light-emitting diodes (LEDs) or reflectors placed on the subject's skin. Film is exposed as

a subject walks through the viewing volume common to the several cameras and is subsequently developed. Then manual digitization proceeds frame-by-frame. For still cameras, a subject wears flashing lights or LEDs, and each flash is recorded during a single, long time-exposure and later developed and digitized. The major disadvantage of both systems is the manual digitization of film plane-position data. Beyond the time involved (in film processing, and digitization) the objectivity and reproducibility of the data so transduced are seriously called into question due to the manual human intervention. Since double differentiation of these data are required for any dynamic analysis of the kinematics, the frequency domain characteristics of the noise and data uncertainties are of paramount importance.

Television is an optical technique that has been partially automatized. Pattern recognition techniques must be used in conjunction with a scan of the entire frame of data to locate each target, because the target's image will appear in more than one pixel. All the image pixels must be used to locate the center of the target in the image plane; therefore, the volume of raw data produced is very large. Additionally, the targets used must be very large. Figure 2 from Winter (1974) [169] page 480 illustrates these problems.
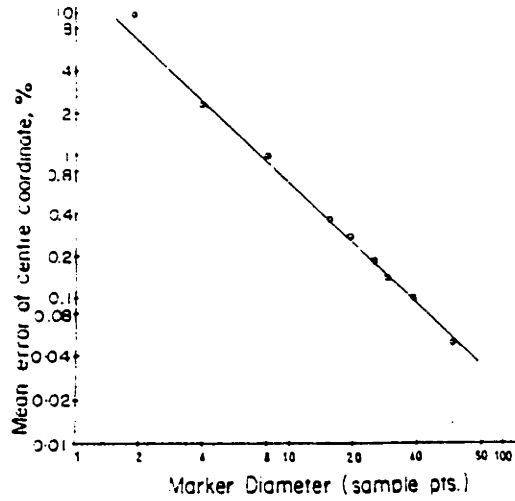
Fig. 3. Plot of error in coordinates of calculated centre vs
diameter of the marker. Centre is calculated by averaging
the co-ordinate (X and Y) of the sample points within the
marker. See Fig. 5 for format of sample points.
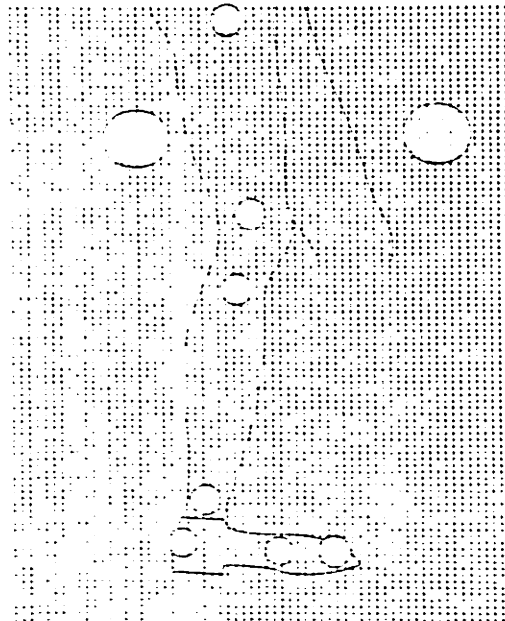
Fig. 5. Computer print-out of one converted T.V field. One
bit resolution is required to for black, 1 for bright. Contour
of leg is drawn in merely to indicate the position of the
markers. Background markers are larger and easily identi-
fied, and serve as an absolute spatial reference in the plane
of progression.

Figure 2. Figures 3 and 5 from Winter (1974) [169] page 480

These facts, in conjunction with the use of commercial video systems limited to 50-Hz frame rate and 525 lines by 416 pixels per line, result in relatively poor sampling frequency and positional resolution. Very bright lights are needed to produce enough contrast for the targets to be automatically discerned, although a method of using infra-red light has been introduced and should help alleviate the problems of contrast. Since the identical reflectors carry no unique identification, manual intervention to identify the trajectories of interwoven reflector paths is necessary.

Techniques using sound instead of light are also in use, but the measurement volume is quite small (0.7 meter cube), and acoustic reverberations and dispersions complicate the measurement process.

The final stereometric technique discussed is the one used in this investigation. It is an opto-electronic technique intrinsically oriented toward automatic data acquisition. The detection device is a lateral-photo-effect diode, a monolithic large linear-area device that produces differential currents at the diode edges proportional to the location of a light spot on the surface. The detector was developed in 1957 by Professor T. Wallmark at Chalmers University, Goteborg, Sweden. (Wallmark, 1957 [163], and Lindholm, 1974 [96])

$$-I_0 R \left(1 - \frac{x}{l}\right)$$

$$E + I_0 R \frac{y}{l}$$

R

R

E (bias)

E (bias)

R

R

$$E + I_0 R \left(1 - \frac{y}{l}\right)$$

$$-I_0 R \frac{x}{l}$$
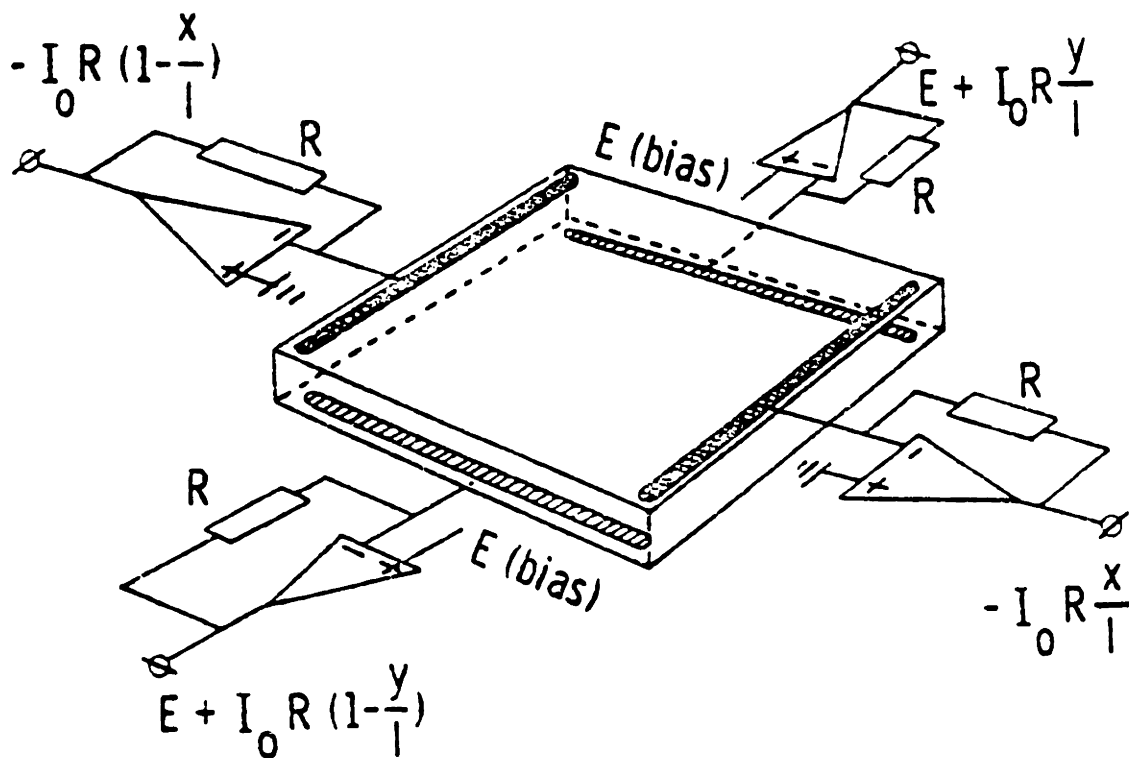
Figure 3. Lateral-photo-effect diode from Woltring and
Marsolais (1980) [178] page 47

In the device we used (a Selspot I manufactured by Selcom AB, Sweden) up to thirty light emitters are time-division multiplexed such that only one is on at a time, and valid data in two camera image planes can be collected while a single emitter is on. The emitters are 940 nm infra-red, light-emitting diodes (LEDs), and the lateral-photo-effect diode receivers have an optical filter to band pass light with a center wavelength of 940 nm. This provides a wide margin of background light interference rejection. An Analog-to-Digital (A-to-D) converter is synchronized to the multiplexing of the LEDs, and the digital data for the image plane location of each LED are produced. The frame rate, where one frame is a sweep through all thirty LEDs, is 315 Hz. The A-to-D converter resolves to 10 bits (1024). As received from the manufacturer, the system has two cameras each with an overall accuracy of about 1 percent. The accuracy is ten times worse than the resolution because of nonlinearities in the optics, and local variations in the detector diode.
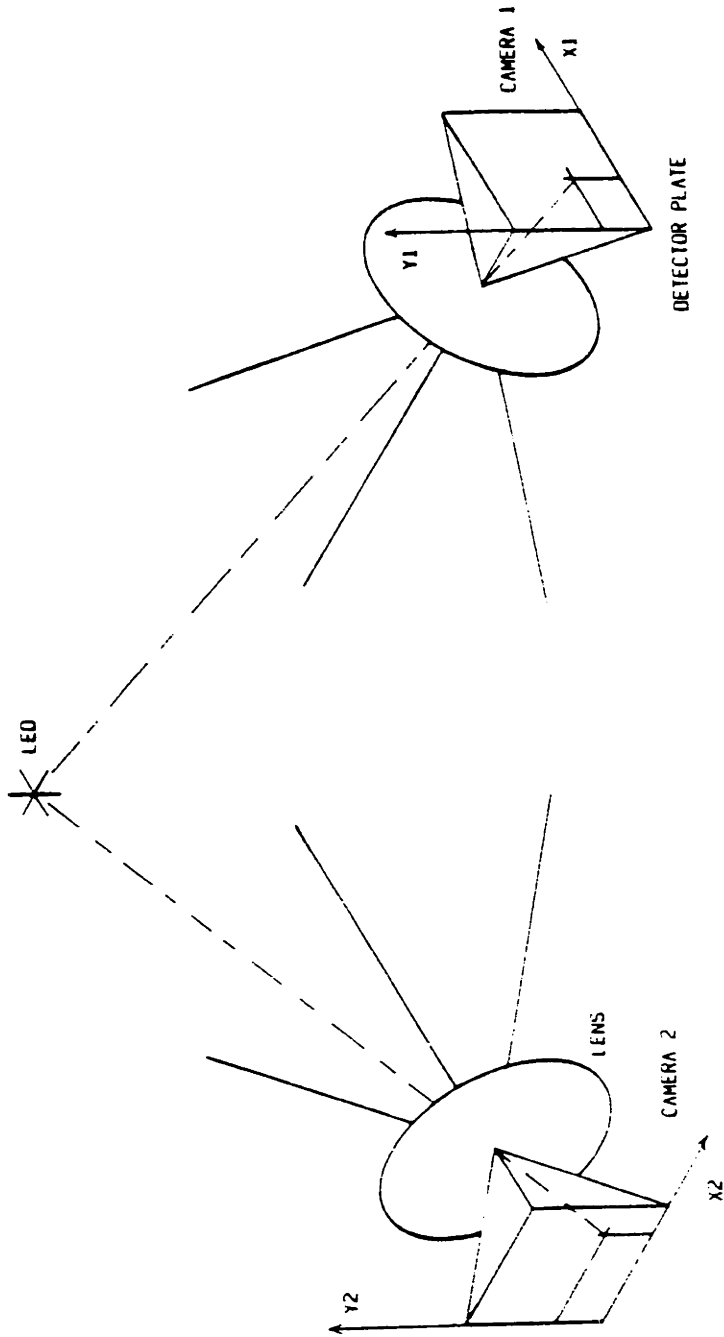
Figure 4. Detectors, lenses, LED, and reconstructed
light rays

Selspot represents a tremendous potential for high-speed, fully automatic, high-resolution (and with suitable calibration high-accuracy), spatial kinematic data on thirty points. Normal room illumination can be used in the measurement volume: The infra-red filters in the cameras, along with background rejection circuitry, eliminate the effects of ambient light. A range of emitters (LEDs) is available with the smallest and lightest useful for gait experiments, having a mass of only 200 milligrams. The system as delivered from the manufacturer is limited in the need for full-field calibration for each camera and the substantial development effort to design and fabricate a computer interface and write extensive data acquisition, reduction and management software.

1 Automaticity and instant or real-time data accumulation
2 Absolute frame-of-reference
3 3-D translation measurement capability
4 3-D rotation measurement capability
5 Reproducibility
6 Large (greater than 1 stride) viewing volume
7 Sampling speed
8 Number of targets
9 Resolution
10 Un-Calibrated Accuracy
11 Potential Calibrated Accuracy
12 Subject influence
13 Subject hazard
14 Soft tissue motion problems

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Goniometers	Y	N	Y	Y	Y	Y	>1k	?	.03%	1%		Large Mass Large Drag		Y
Still Film	N	Y	Y	Y	Y	Y	30	?	.03%*	?		Flashing lights Dark Room		Y
H.S. Film	N	Y	Y	Y	Y	Y	250	?	.03%*	?		Minimal		Y
T.V.	N	Y	Y	Y	Y	Y	60	?	1%	?		Large targets Bright lights		Y
Sonic	Y	Y	Y	N	Y	N	?	?	?	?		?		?
	(/w very few emitters)													
X-Rays	N	Y	Y	Y	Y	N	?	?	?	?			Radiation	N
Accelerometers	Y	N	Y	Y	Y	Y	>1k	?	?	?		Small mass		Y
Selspot	Y	Y	Y	Y	Y	Y	315	30	.1%	1%	.1%	Normal room light Small mass		Y

? denotes an unknown value
* indicates an approximate value for comparison

Table 1 Summary of mobility kinematic measurement techniques

Automaticity, high speed (real time), objectivity, and result data in an absolute frame of reference were all highly ranked design requirements considered during the selection of the measurement system. It was felt that satisfaction of those requirements would produce a facility capable of substantially surpassing those employing current techniques and would result in data capable of shedding new light on human mobility.

> "In view of the system requirements of the Mobility Laboratory's (now the Eric P. and Evelyn E. Newman Laboratory for Biomechanics and Human Rehabilitation [LBHR]) projects, it was felt that the Selspot System held the greatest potential for meeting them in the shortest time and at the lowest total cost. As a result, the Selspot System was acquired as the fundamental data-acquisition system."
> Conati (1977) [34] page 24

Analysis techniques producing estimates of force and moment vectors require a force reference at one point along the segment linkage chain. Virtually all investigators use a multi-component force-plate capable of measuring the complete force and moment vectors between the foot and the floor. This investigation is no exception, although in early applications of the system (Hardt 1978 [66,67,68]) before the forceplate was acquired, force analyses were performed based on the assumption of linear weight transfer from one foot to the other during double-leg stance.

The next chapter will describe the details of the measurement problem and those aspects of it that the "modern" techniques have not solved. The <u>measurement</u> solution implemented here is divided into three broad areas, each to be discussed separately: foot-floor, force-interaction measurement (first, because it is the simplest), position history measurement of kinematic markers, and position history measurement of skeletal links. Subsequently, the <u>analysis</u> (kinematic and dynamic) will be discussed.

CHAPTER 2

GAIT ANALYSIS TECHNIQUES LIMITATIONS

As discussed in the last chapter, accumulating data on the position histories of the limb segments of a moving human is a difficult and subtle process. If the human is overly constrained by the instrumentation, his gait is affected. Similarly, distractions such as noises or flashing lights will also affect the results. Since ethical clinical guidelines in this country do not allow the use of bone pins, a non-invasive technique is required. Bone pins also influence the subject's gait due to restriction of soft tissue motion. Non-invasive measurement requires that histories of motion of the skin surface be acquired and used to infer the motion of the underlying skeleton and center of mass. Since the muscle masses move relative to the bone during motion, the center of mass for a given segment will move with respect to the skeleton. None of the joints in the body are planar "pin" joints, necessitating full three-dimensional kinematic data to completely describe the motion. The precise inertial properties of each segment are difficult to determine and depend on segment position, muscle activity, and the geometric definition of the ends of each segment. Sampling noise accompanying the signal will be greatly accentuated once double differentiation has taken

place, necessitating filtering based on knowledge of signal-
and noise-amplitude spectra.

This chapter will discuss how these and other
limitations affect the results produced by a gait analysis
system using an opto-electronic, sterometric technique.

FREQUENCY CONTENT OF GAIT

Only a very few extant publications attempt to quantify
the frequency content of gait, and one only need attend any
conference on biomechanics to hear the current controversy
over valid cut-off frequencies to use with low-pass filters
applied to noisy gait data. A. Cappozzo (1975) [22] uses
the first five terms of a Fourier series to describe the
frequency content of gait. D. Winter (1974) [169] and
(1979) [166] applies a 5- or 6-Hz, low-pass filter.
K. Soudan (1979) [147] uses spline functions, and
R. Alexander (1980) [4] uses five terms of a Fourier series.
None of these investigators provide a thorough
substantiation for the selection of their noise rejection
technique.

> "The actual value of the sampling rate to be
> chosen is worthy of a separate discussion. It is,
> in fact, possible to find a value of it which is
> optimal from a practical point of view. This will
> be done when a deeper knowledge on the
> characteristics of the displacement functions and
> their measures, is available."
> Cappozzo, et al., (1975) [22] page 309

"... the problem of choicing (sic) the more
convenient number of data points can be faced.
Broadly speaking, this choice is the end result of
a compromise between accuracy of fit and cost of
data acquisition in terms of man/time necessary
for the reading of the film."

"... experience ... in some typical locomotor
patterns shows the more convenient number m equal
to 5 (or slightly more than 5) times n, in close
accordance with the more common proposal of the
applied measurement theory. In level walking at
normal speed, for instance, the highest value
found for n, among all the functions d(t), is 5 or
6; m must then be between 25-25 on a double step
period. Hence a 20-30 f.p.s film is generally
sufficient."
Cappozzo, et al., (1975) [22] page 312

Cappozzo's d(t) is a function containing the first five

Fourier series terms, and his justification for a low

sampling frequency is based on how well the data can be fit

with a series, not how closely the data match the real

motion, and on limiting the amount of time necessary to

manually reduce the film data.

The only literature citation that addresses measuring

the actual frequency content of gait that I could locate was

S. Simon, et al., (1982) "Peak Dynamic Force in Human Gait"

[143]. The frequency signatures reported by Simon could

only be reproduced by highly abnormal gait with sharp

posterior "kicks" at heel strike. The method of forceplate

data accumulation and reduction used here will be discussed

later; however, the following figures are included as a

comparison. Figure 5 is a typical record of vertical force

at the forceplate as a function of time for normal gait. Figure 6 is an expanded region at heel strike, and Figure 7 shows amplitude as a function of frequency for the same force record. The frequency domain response was obtained through the use of a digital Fast Fourier Transform (FFT) routine. All walking measured in this investigation was barefoot, no shoes were used. Figure 8 shows Simon, et al., published data.

Typical Vertical Forceplate Data

Time [Milli-Seconds], Sampled at 2.000 KHz.
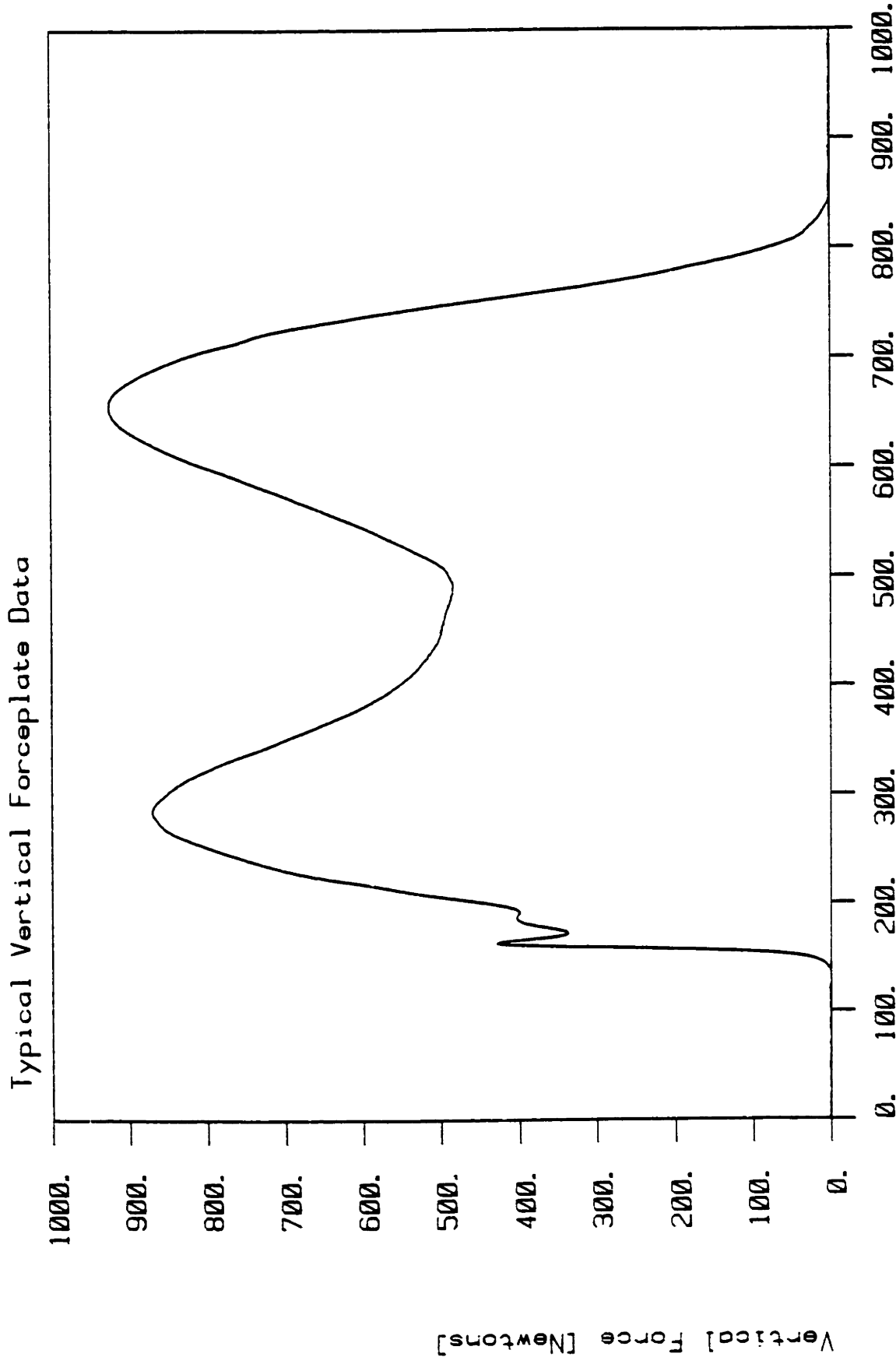
Figure 5. Normal gait vertical force record

Vertical Force [Newtons]
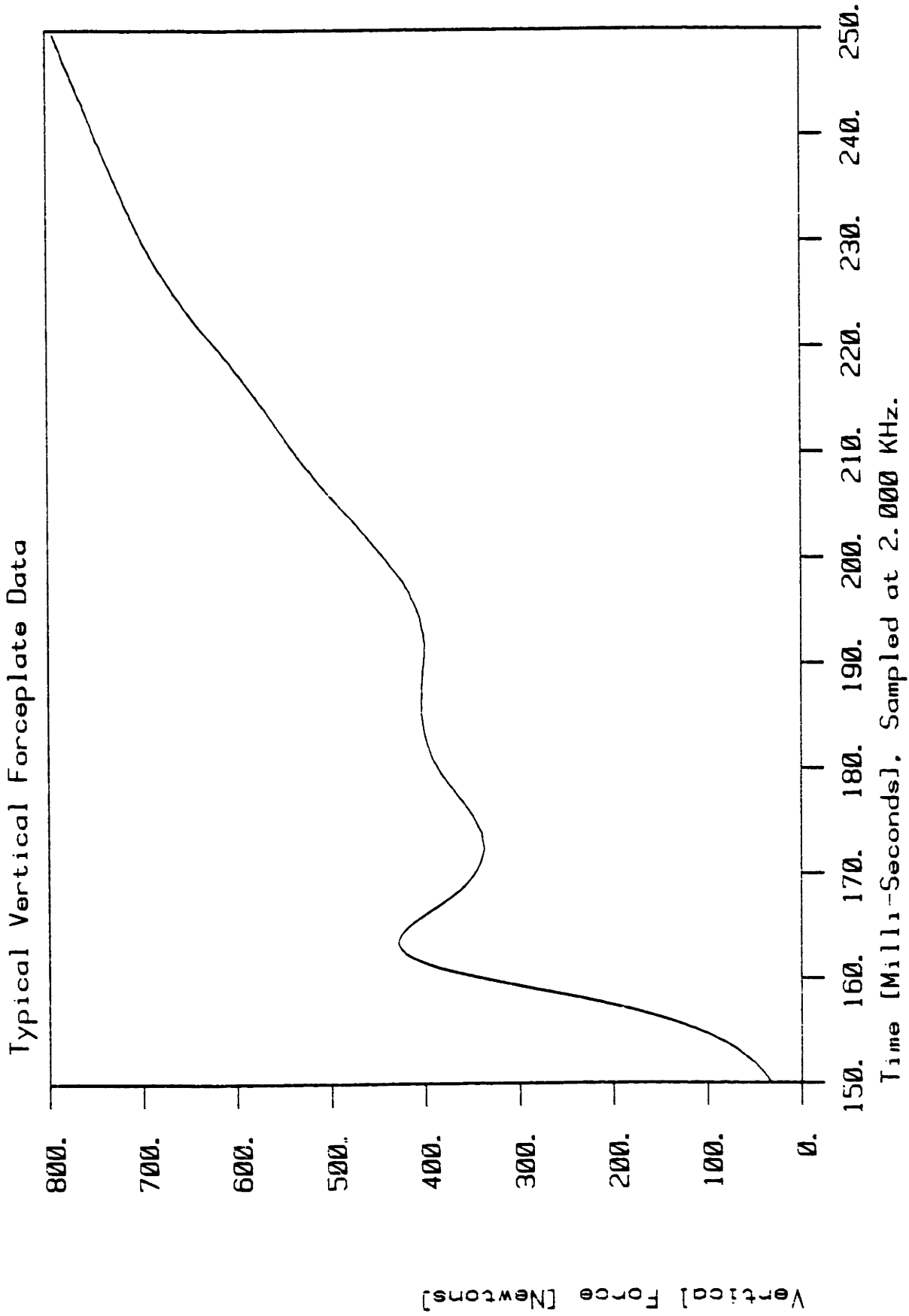
Figure 6. Expanded region about heel strike for normal gait
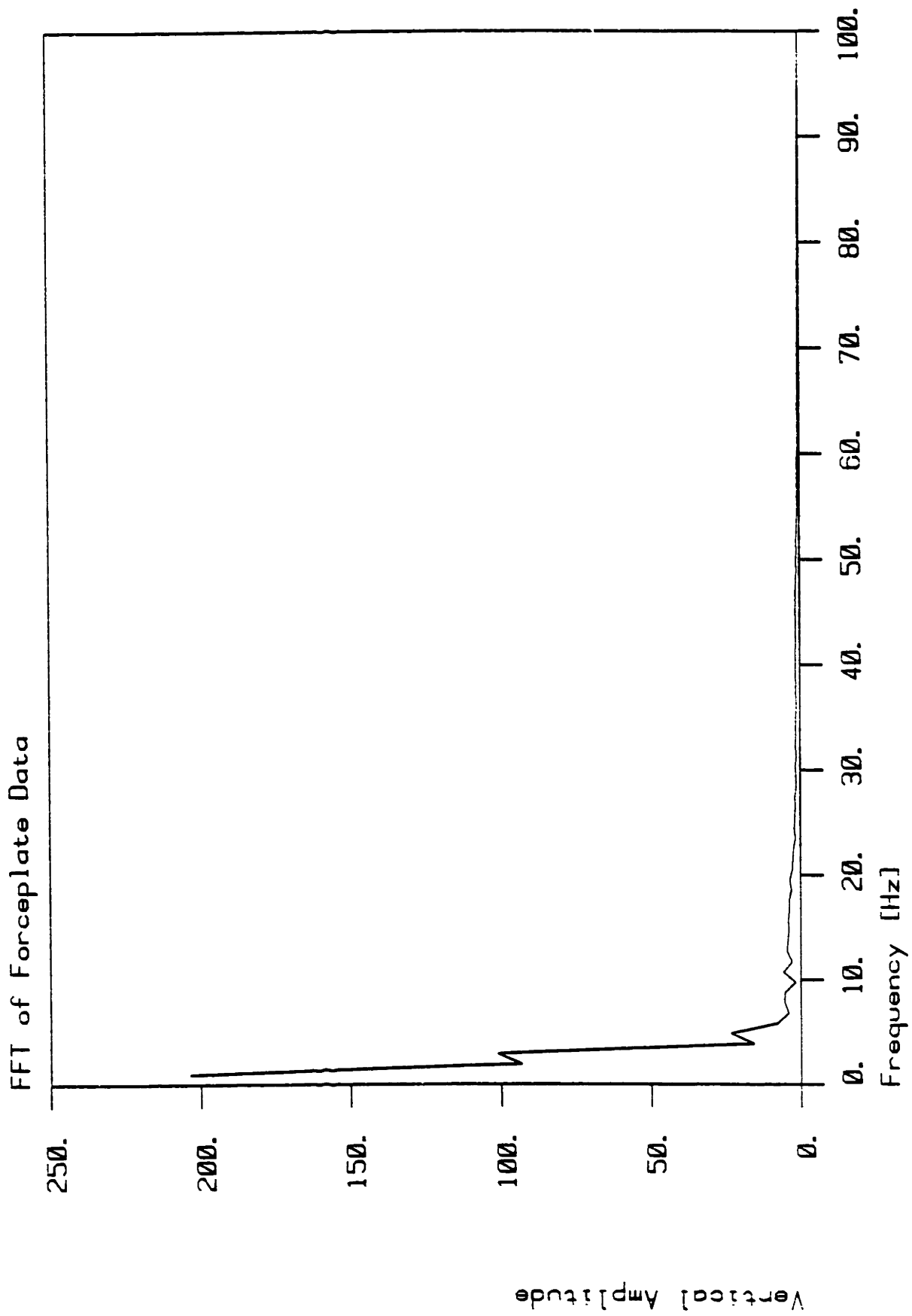
Figure 7. Frequency domain plot of normal gait vertical force

Fig. 5 Fourier analysis of vertical components of representative subjects walking with (A) and without shoes (B) The amplitude of the frequencies displayed is normalized to the maximum amplitude recorded at any frequency. Note is made that the majority of the signal exists below 10 Hz but definite higher frequency components of lower amplitude exist up to 75 Hz without shoes and up to approximately 50 Hz with shoes.

Fig. 1(B) Fourier analysis plot of dynamic response of 1300 Hz plate.

Fig. 4 Typical oscilloscope tracing of vertical force during the stance phase of a subject's normal gait cycle barefoot. Note the presence of a large spike occurring at heel strike which in this particular subject equals body weight and lasts 1.5 ms. This impulse consists of frequency components between 1.75 Hz. The remainder of the trace is similar to that already noted by previous studies consisting of frequencies from 1 to 10 Hz.

Figure 8. Simon et al. heel strike force data (1981) [143]   page 818

To reproduce the large amplitude spike at heel strike, several attempts at walking with a sharp posterior kick were recorded. Figure 9 shows the measured response from one of those trials. Again, all gait is un-shod. Figure 10 shows an expanded region around heel contact. Just as in Figure 6, it can be seen that there are many data points across the first peak in the data owing to the 2-kHz sampling rate.
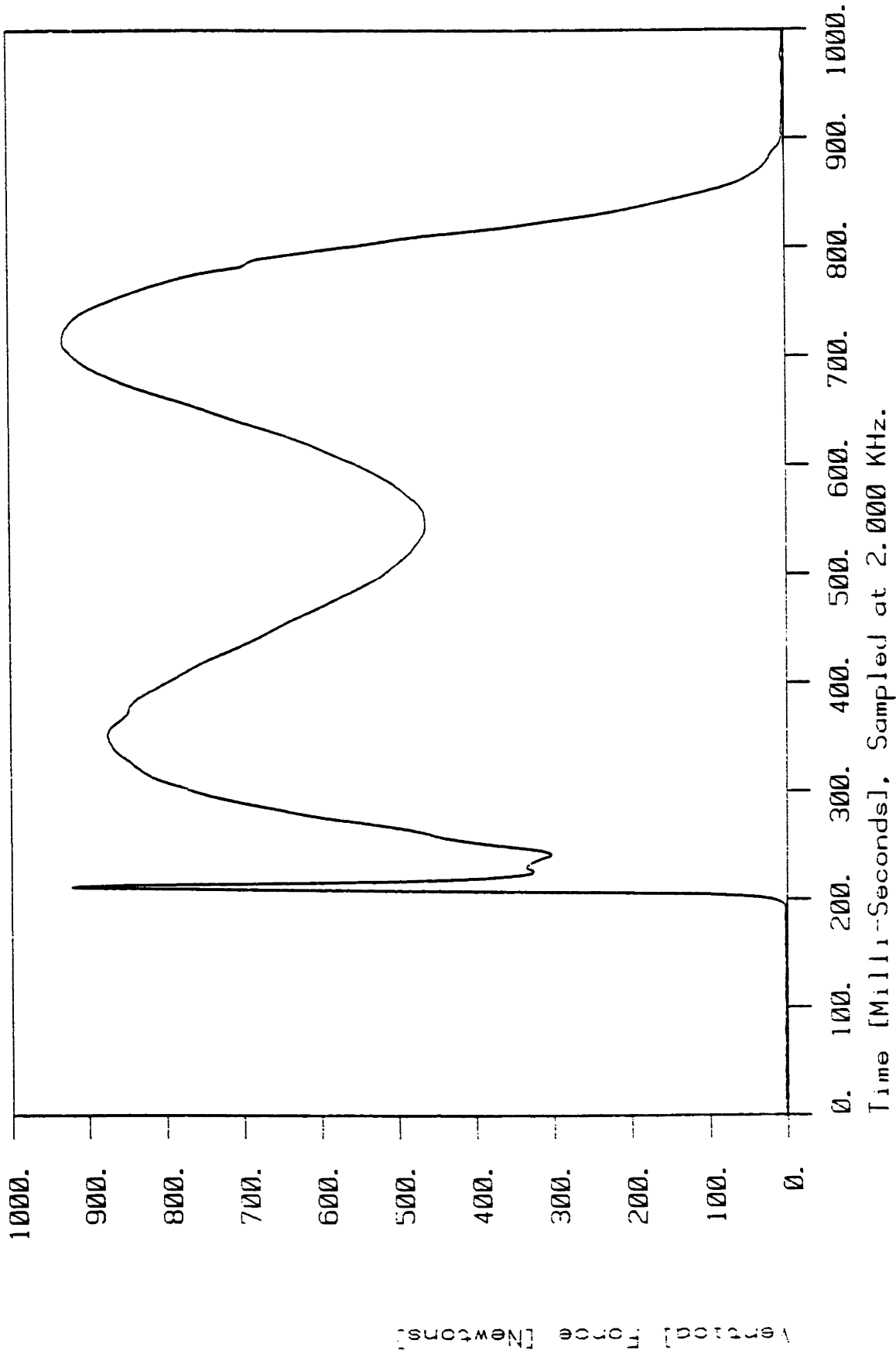
Figure 9. High frequency heel strike spike from my data
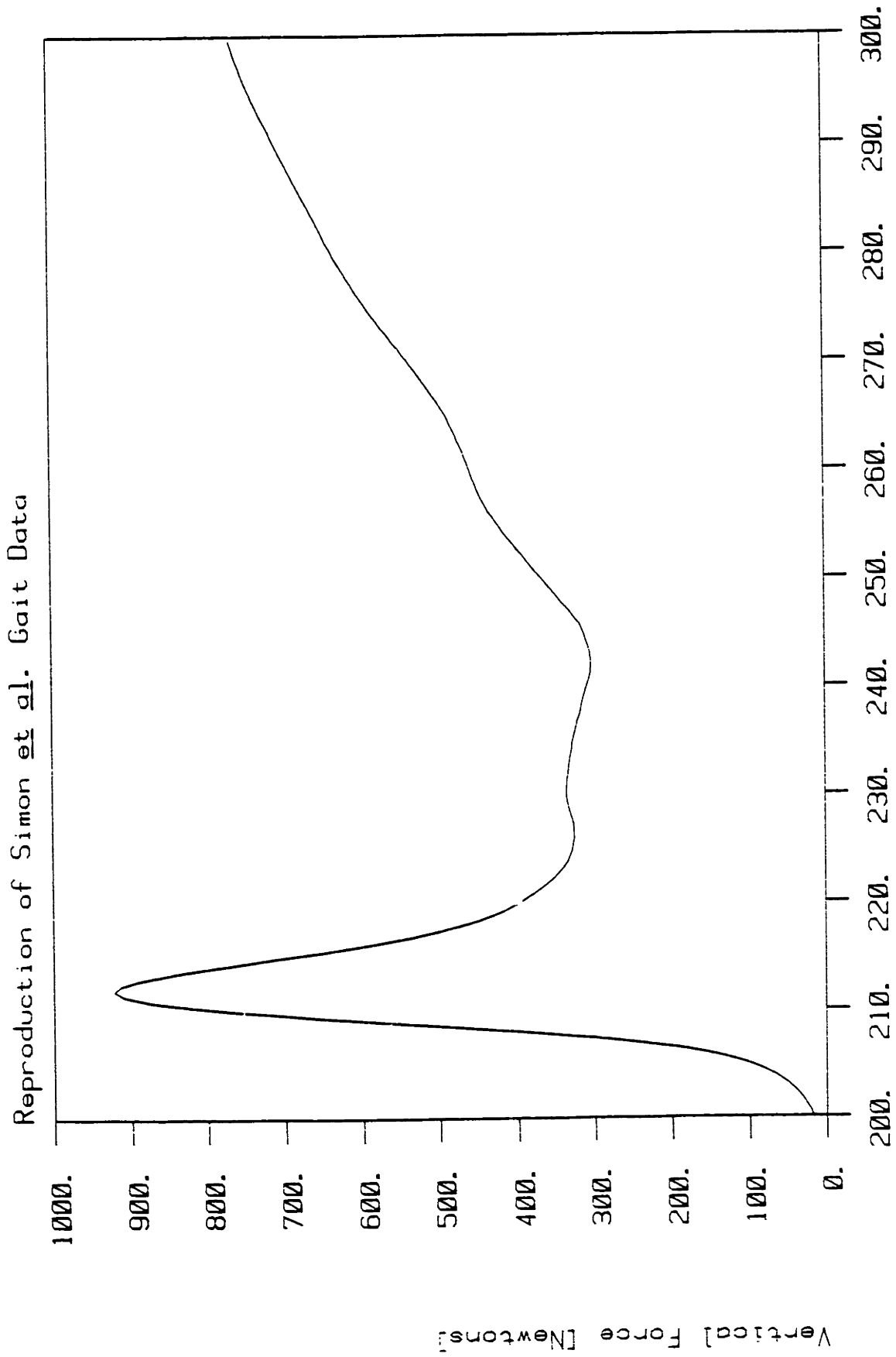
Figure 10. Expanded region about heel strike from figure 9
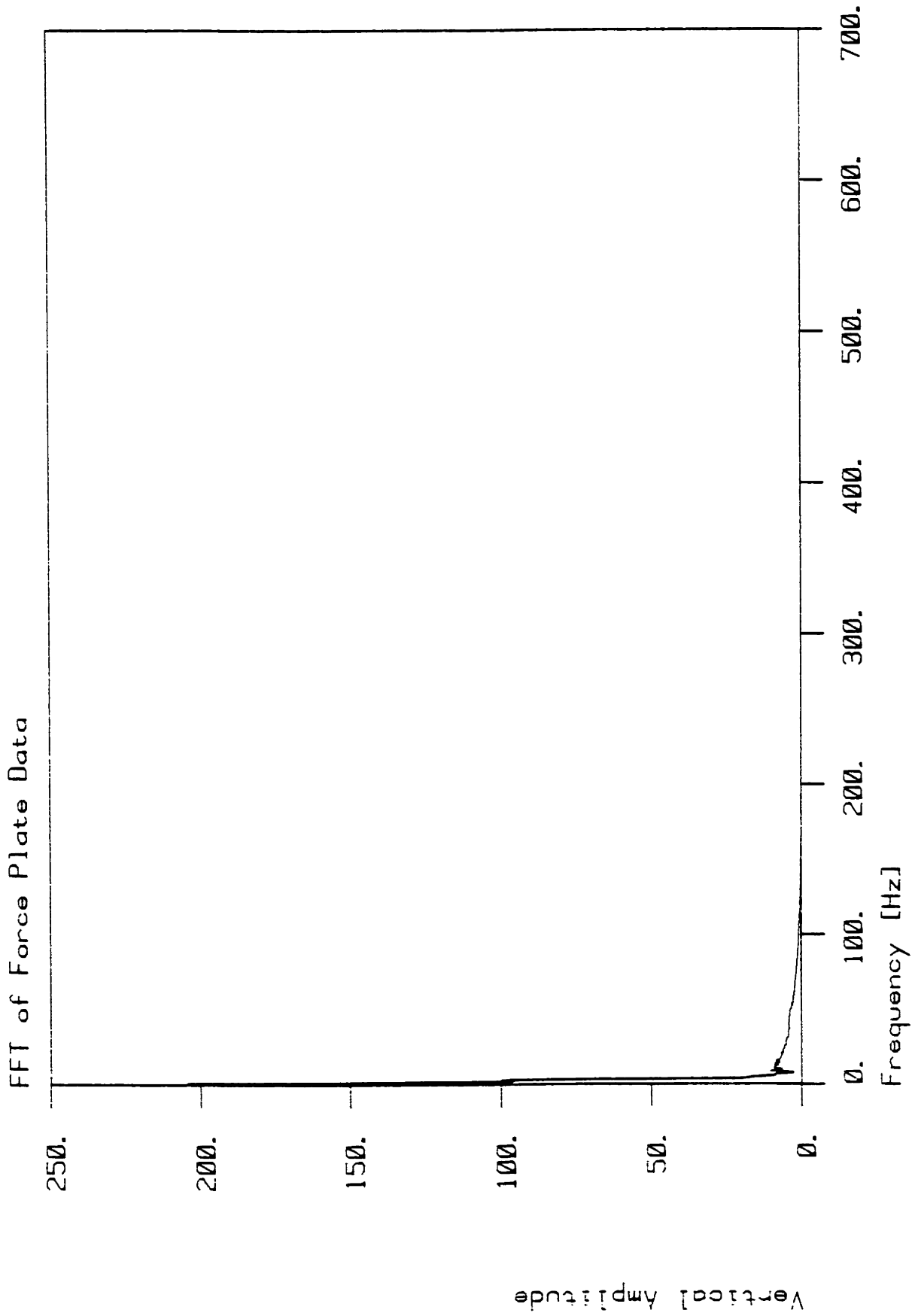
Figure 11. FFT of vertical force record on Simon et al. scales
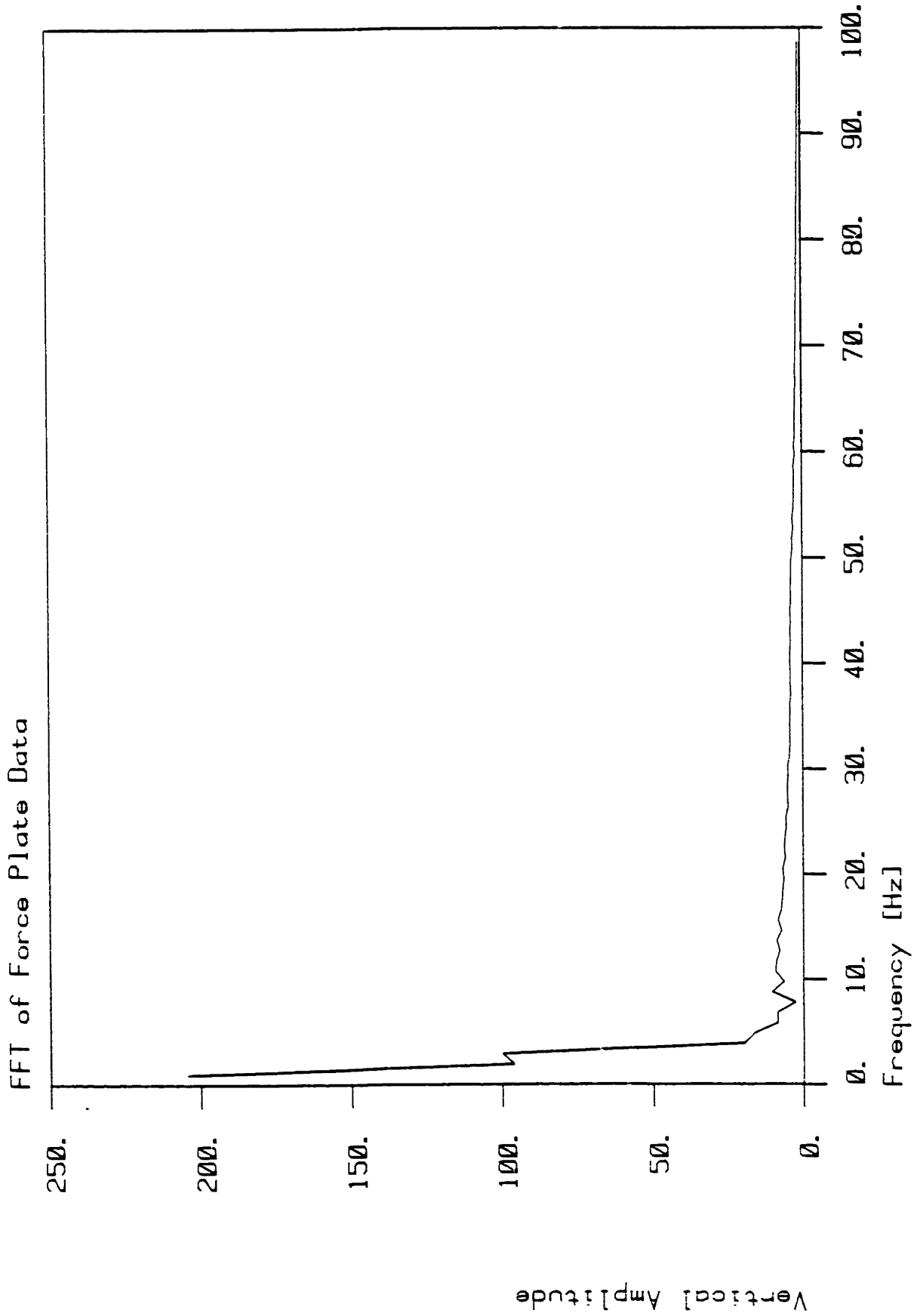
Figure 12. FFT of vertical force record

FFT of Forceplate Data
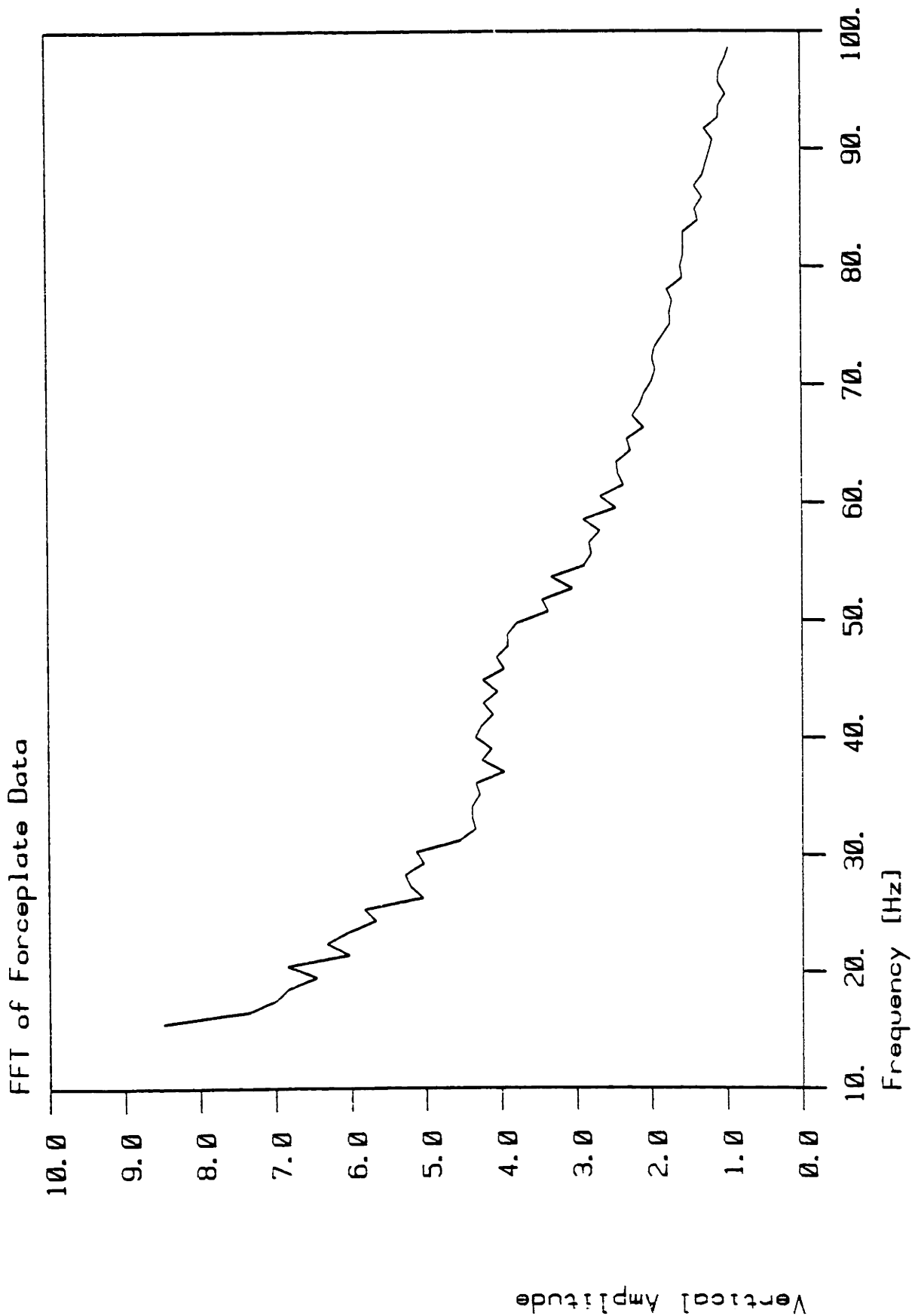
Figure 13. Expanded region of FFT above 15 Hz

Note that even with a reproduction of Simon's input vertical force data, the relatively large amplitudes at frequencies above 20 Hz were not reproduced.

FREQUENCY RESPONSE OF MOTION MEASUREMENT SYSTEMS

The first and most fundamental unknown relating to gait is the frequency content of the underlying phenomena, and knowledge of this content is vital to understanding any of the problems enumerated above. For instance when position measurement systems are used which include inherent sampling noise (particularly if a dynamic analysis is planned where the kinematic data require double differentiation), the noise and the underlying phenomena (the signal) must be thoroughly described in the frequency domain in order to separate the noise from the signal. The double differentiation of the position histories, necessary for dynamics, amplifies the noise by the square of its frequency and lends added importance to separation of signal and noise. For a discussion of the differentiation technique used in this investigation, see Antonsson (1976) [7] and also Chapter 3 of this thesis. If noise characterized by:

$$An*Sin(Wn)t$$

(where: An is an amplitude, and Wn is a frequency, and t is time) is added to a signal, after double differentiation the

noise becomes:

$$An*(Wn**2)*Sin(Wn)t$$

So eliminating frequencies higher than Wn or attenuating An becomes vital.

As a graphical illustration of the influence of noise, the following "iso-noise amplitude" plot is presented. The plot answers tne following question: If a sinusoidal signal with an amplitude of 100 units had superimposed noise at different frequencies, what amplitudes of noise (as a function of frequency) would produce equal noise and signal amplitudes (signal to noise ratio of 1) subsequent to double differentiation? A few points along the curve are interesting. A noise amplitude of only 10 percent of the signal at 3 Hz will produce a noise amplitude equal to the signal once double differentiated; only 1 percent is needed at 10 Hz; and 0.1 percent at 32 Hz. Obviously the function used to produce this curve is simply:

$$An=100/(Fn)^2$$

from

$$100(2\pi)^2 = An(Fn2\pi)^2$$

The only purpose in presenting it is to underscore the importance of understanding not only the frequency content of gait, but also the noise content of the measured signal.

To produce useful results, the additive noise in the signal
must have amplitudes well below this curve.
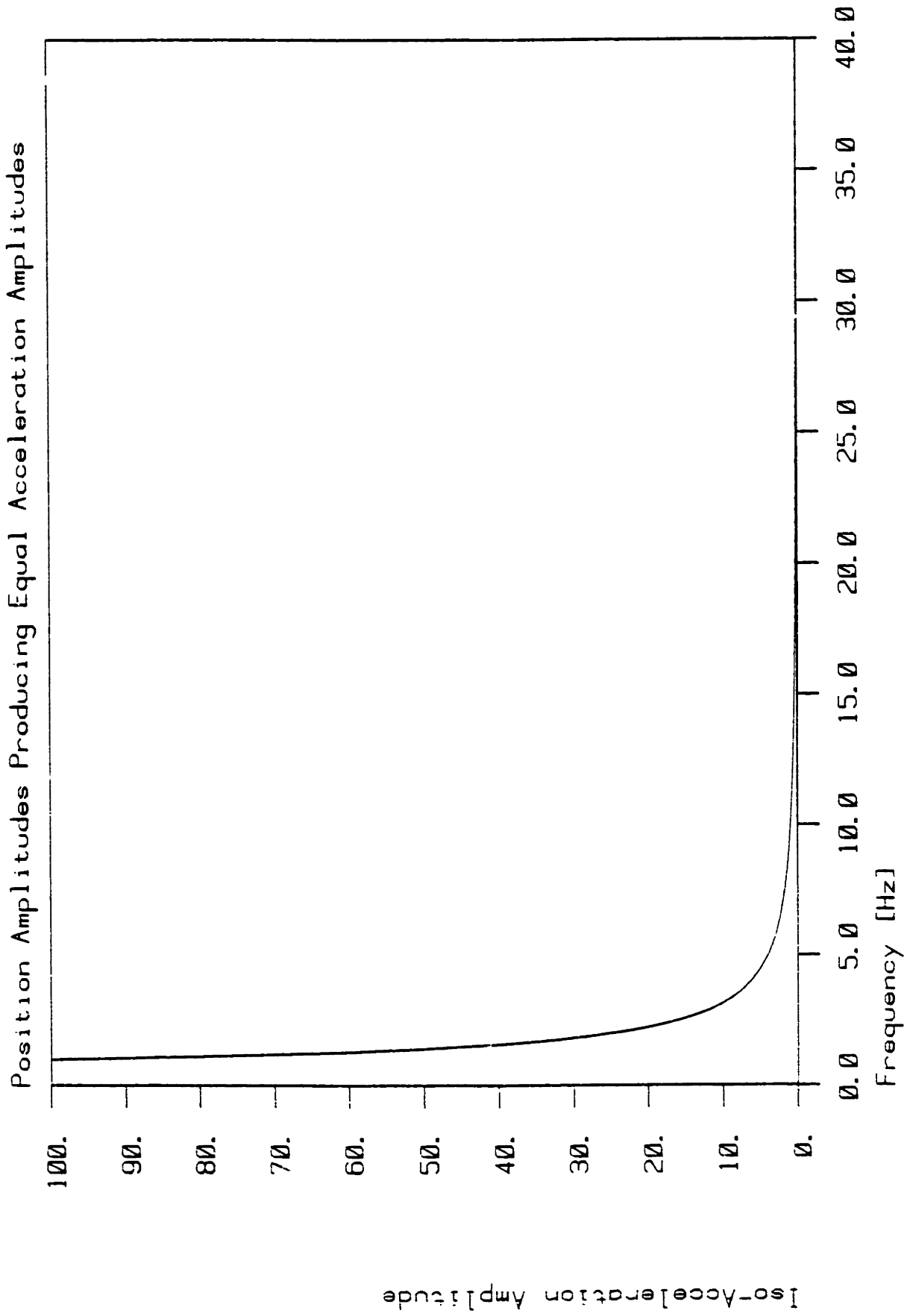
Figure 14. Iso-Acceleration amplitude plot

A final restriction for the frequency domain characteristics of a kinematic measurement system for gait is that the noise inherent in the sampling be above the frequency of the signal. It is of no use to be able to reject all noise between 15 and 150 Hz if the noise has significant components below 15 Hz, and since all measurement systems produce some noise in addition to the signal, the above aspects of the measurement system must be considered to produce kinematics and dynamic estimates with known accuracy.

STATIC AND DYNAMIC POSITIONAL ACCURACY AND RESOLUTION

If normal human walking were a quasi-static, gravity dominated activity, in the absence of measurement noise, a fairly coarse kinematic resolution and accuracy would produce quite accurate force estimates. For instance, a resolution of 5 mm spatially will yield limb orientations within +/-2 degrees for a 40 cm long segment measured with markers at the end points. For the quasi-static case, this magnitude of error will yield joint loads within 10 percent; however, should dynamics be present at all, the influence of noise and its amplitude and frequency becomes more important than static resolution. Gait contains dynamics, and though the static resolution is important, as shown above, the sampled noise content can outweigh the static forces in the

dynamic estimates. Rotational accuracy involves the same considerations. Fortunately photogrametrists have devoted much effort to the study of calibration of optical devices for three dimensional position measurement, and many of the same techniques can be applied to produce an optical system with an accuracy of 1 mm for a point within a viewing volume containing one stride.

SOFT TISSUE MOTION WITH RESPECT TO THE SKELETON
AND DETERMINATION OF JOINT AXES

Since all of the kinematics, upon which the dynamic estimates will be based, must be measured on the surface of the skin, the motion of the soft tissue with respect to the skeleton must be considered and measured. All markers mounted on the skin will exhibit some motion, and every effort to miminize it must be taken. This relative motion can be viewed as noise superimposed upon the motion of the skeleton, and the same techniques to reject noise as discussed above apply: Make the amplitude as small as possible in the frequency range of significant signal, and eliminate the higher frequency noise through filtering. The amplitudes can be made small by use of low mass markers closely coupled to prominent bony protuberances, such as the medial border of the tibia and the femoral epicondyles to provide the maximum skeletal trajectory fidelity. The skeletal elements are located with respect to the markers by

means of vectors to the segment endpoints determined by a kinematic examination of the data. This will be discussed fully in Chapter 4.

Further, since the markers will move with respect to the skeletal links and joints, markers placed on the skin at an assumed joint center or axis location will produce erroneous results. Fortunately, except for the frequencies of soft tissue motion in the range of gait phenomenon, these errors fall under the aegis of quasi-static force estimate errors (a positional error in the axis of rotation of a joint will only influence the static, gravity dominated joint forces and moments as shown above). The common method of placing a marker at a joint center, however, yields no information at all about the magnitude and frequency of soft tissue motion.

> "... segment lengths are measured directly on the subject as distances between the markers approximating the position of the articular centres ..."
> Cappozzo, et al., (1975) [22] page 313

> "... the markers borne by the subject indicate only approximately the position of the rotation axes of the joints and undergo displacements of the skin in relation to the bones during motion ..."
> Cappozzo, et al., (1975) [22] page 309

"Particular attention must be paid to the
inaccuracy of the segment lengths."
  "The actual distance between the projections on
the sagittal plane of two joint rotation centres
varies during motion for the following reasons:
  (a) the leg movement is not planar
  (b) the joint rotation centres move with respect
to the bones."
Cappozzo, et al., (1975) [22] page 313

In contrast, a method based on measuring complete link
kinematics using imbedded body-coordinate systems, to be
described later, produces enough information to locate the
joint axes. This reduces a possible source of error and can
be used to evaluate soft tissue motion.

RESULTING UNCERTAINTY IN KINEMATIC AND THUS DYNAMIC GAIT DATA

   The sources of error enumerated so far for a kinematic
measurement system for human motion to be used for dynamic
estimates are:

1. Sampling noise in the kinematics

2. Inadequate frequency bandwidth of the sampling system to
    capture signal and reject noise

3. Limited positional measurement accuracy and resolution

4. Inability to measure all three translations and
    rotations of each body segment.

5. Motion of measurement markers mounted on soft tissue,
    with respect to the skeletal links.

6. Inability to resolve joint axes correctly

All of these contribute to errors in the dynamic estimates, and some, like measurement noise, are magnified greatly in the calculation process.

CHAPTER 3

MEASUREMENT APPROACH

The measurement of the foot-floor interaction forces will be discussed first. Transduction is carried out by a Kistler 9281A, multi-component forceplatform (Kistler Instrumente AG, Winterthur, Switzerland). See Appendix A for specifications. This device is capable of measuring forces in three orthogonal directions; the position of the net center of force on the surface of the plate; and the net moment about an axis perpendicular to the surface. According to Kistler the fundamental natural frequency of the platform is above 1.0 kHz. This is achieved by the use of very stiff (1.5 Newtons/nano-meter [6.6 tons/micro-inch]) quartz crystal piezo-electric force transducers. Each corner of the plate is supported by a set of three toroidally shaped crystals stacked on top of each other, one for forces normal to the plate, and two for orthogonal shears. There are a total of twelve transducers in the platform assembly. Each stack of three rings is under a factory-adjusted pre-load to allow the platform to measure tensile as well as compressive loads, and to be able to maintain accuracy when the net center of force is outside the rectangle bounded by the transducer stacks.

The platform was mounted in the floor of the Mobility Facility in room 3-147 at M.I.T. with its upper surface flush with the surface of the surrounding floor. Figure 15 is a schematic of the mounting. Several vertical columns were installed between the bottom of the mounting beams in the Facility floor and the basement floor below to stiffen the platform's support. This was necessary to minimize the bending that the forceplate foundation suffered when the floor surrounding the platform was loaded. This problem was particularly acute for this force-measuring device because of the very high stiffness of the transduction elements.

280 mm

75 mm

12 mm

200 mm

Forceplate mounting frame
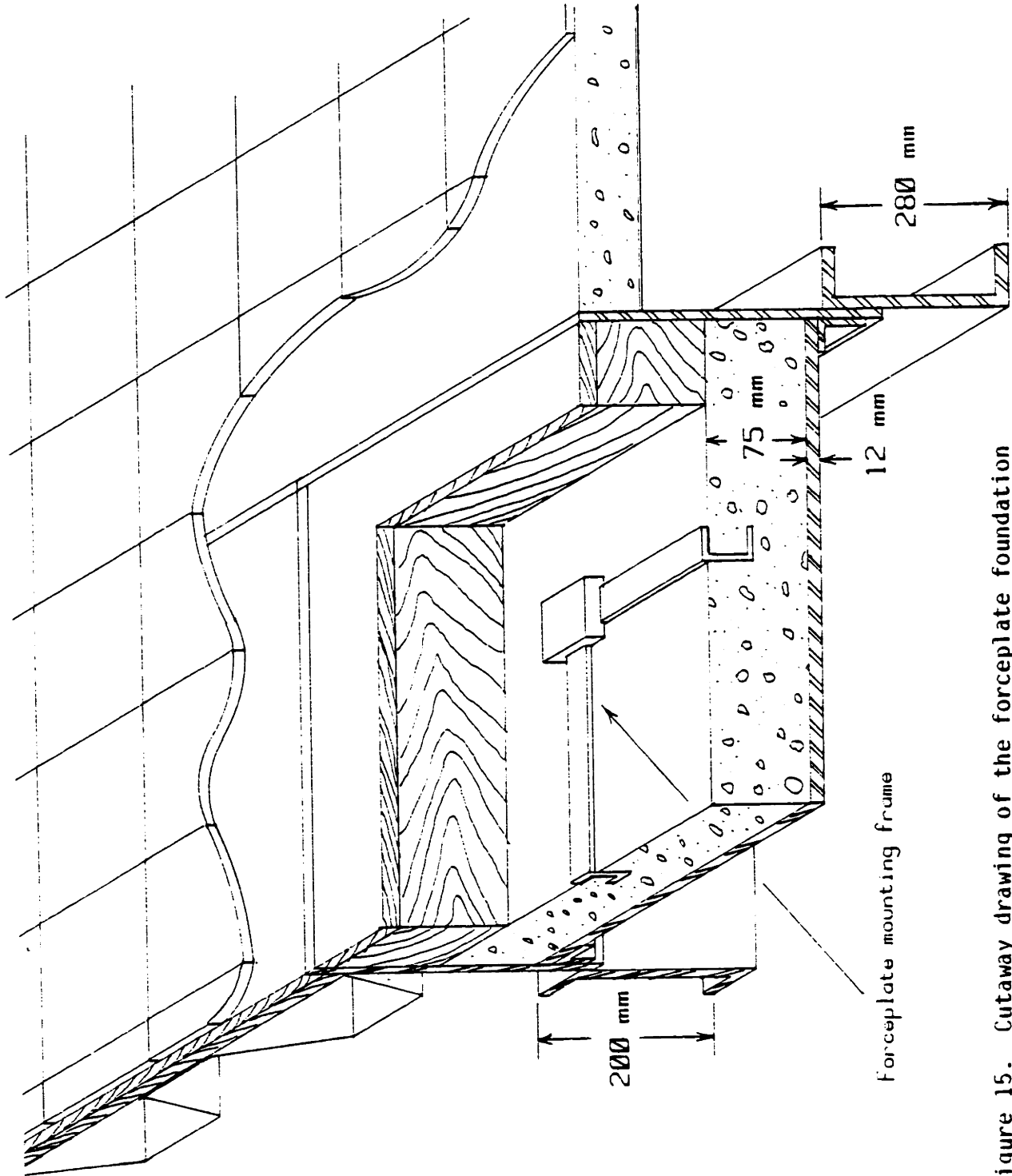
Figure 15. Cutaway drawing of the forceplate foundation

To measure the charges produced by the piezo-electric crystals, Kistler provides eight charge amplifiers. There are only eight because some of the charge signals are connected together at the inputs to the amplifiers. This reduces the total number of amplifiers necessary, but not the information available. Each charge amplifier produces voltages proportional to the load on its corresponding transducer. The possible output voltage range is +/-10.0 volts, and the input force range is selectable over a wide range (+/-1 Newton/volt to +/-5000 Newtons/volt) to allow different sensitivies. Appendix A includes the Kistler nomenclature and algorithms for determining the resultant forces acting on the plate from the eight analog voltages present at the charge amplifiers.

Since the entire motion measurement system is discrete-time-sampled, it was decided to convert the eight analog voltages to digital signals at the forceplate charge-amplifier box for several reasons. Most importantly analog signals are sensitive to electromagnetic noise, particularly when the signals must be transmitted over long distances. Using analog lines from the charge amplifiers to analog-to-digital (A/D) converters in the computer 50 feet away resulted in a signal-to-noise ratio of about 10. High fidelity is much more easily accomplished in a digital signal transmitted over that length.

To implement this A-to-D conversion at a remote (from the host data processor) location required the use of an Analog Devices (Norwood, Massachusetts) DAS 1128 Analog-to-Digital Converter Sub-system. See Appendix B for the manufacturer's specifications. The essential aspects of this device are that it can be configured to have eight double-ended (differential) input signals and produce 12-bit digital results compatable with Transistor - Transistor Logic (TTL) used throughout digital computers. The conversion rate can be adjusted, and for this application it was set to its slowest available value to obtain the maximum possible accuracy. This meant that a sweep through all eight channels converting each one to 12 bits (1 part in 4096) takes 0.32 milli-seconds. When computer intervention time is allowed between sweeps (frames), the maximum sampling rate available is 2.5 kHz. This could be increased by reducing the conversion time in the sub-system but at the expense of some accuracy and stability. Since the most often used mode for the forceplatform sampling system is in synchrony with the kinematic acquisition system (which runs at a maximum rate of 315 frames per second), and the fundamental natural frequency of the plate is about 1.0 kHz, a higher sweep rate was felt to be unnecessary.

The rate of sampling of the analog signals is determined by an external input to the A/D conversion sub-system. This framing and synchrony circuitry is external to the DAS 1128 and makes the A/D converter sweep through all eight channels (always starting with channel 0), once per pulse on the external frame start line. There are several external frame start lines available. One is wired to a user-controlled line, one is from a programmable clock, and one is from a signal denoting a frame of Selspot data. In summary, the analog-to-digital conversion system used with the forceplatform has three digital and eight analog input lines and seventeen digital output lines. Each strobe (pulse) on one of the input digital frame start lines produces a train of eight sequential 12-bit digital signals (and four more strapped to the most significant bit) demarked (in time) by a control line signalling the existence of each valid word of data on the digital lines. Since the conversion of this data occurs much faster than other associated sampling, a handshake has been also implemented to require the DAS 1128 to hold its digital data until the receiving computer is free to accept it. A "handshake" is simply a two-way communication where a device places a signal on a line indicating it has data available, and then waits until the computer responds with a signal indicating it has accepted the data and is ready for more.

Appendix B contains a more detailed description of the conversion system.

The forceplate data are accumulated by the host computer by a Direct-Memory-Access (DMA) interface. This device allows high-speed (approximately 100,000 words/second) acquisition of 16-bit digital signals directly into the host computer's main memory without the intervention of the processor for the transfer. The DMA used is an MDB (Orange, California) DR11-B.

Synchrony with the Selspot kinematic acquisition system was achieved by connecting one of the forceplate conversion frame start lines to a signal denoting the middle of a frame of Selspot data to be accumulated. The middle of a frame was chosen to avoid timing conflicts in the central processor. Occasionally the Selspot requires processor intervention (to swap buffers to be filled), and that always occurs at the beginning of a frame of Selspot data. Similarly, the forceplate occasionally requires a change of buffers, and that occurs at the beginning of a frame of forceplate data. Selecting the center of the Selspot frame avoids requiring the processor to deal with both swaps of buffers at the same time. The process and necessity of swapping buffers will be discussed later.

OPTO-ELECTRONIC POSITION MEASUREMENT SYSTEM

A Selspot I (Selcom AB, Goteborg, Sweden) was used to acquire kinematic data. It consists of a pair of cameras, each comprised of a Canon (Lake Success, NY) f 0.95 mm Television lens, a 40-mm-square lateral-photo-effect diode with its surface in the focal plane of the lens and a pair of amplifiers to convert the differential currents present at the edges of the diode plates to voltages. These voltages characterize the location of the centroid of a light spot on the surface of the detector plate (lateral-photo-effect diode). The detectors have optical filters placed over them to eliminate the influence of visible light, and retain sensitivity to infra-red, centered at 940 nanometers. For more information on lateral-photo-effect diodes, see Wallmark (1957) [163], Lucovsky (1960) [98], Broerman (1968) [18], Groden and Richards (1969) [64], Woltring (1974) [174], Woltring (1975) [176], Fieret, et al., (1977) [58], and Woltring, and Marsolais (1980) [178].

$-I_0 R(1-\frac{x}{l})$

R

E (bias)

$E + I_0 R\frac{y}{l}$

R

R

R

E (bias)

$-I_0 R\frac{x}{l}$

$E + I_0 R(1-\frac{y}{l})$
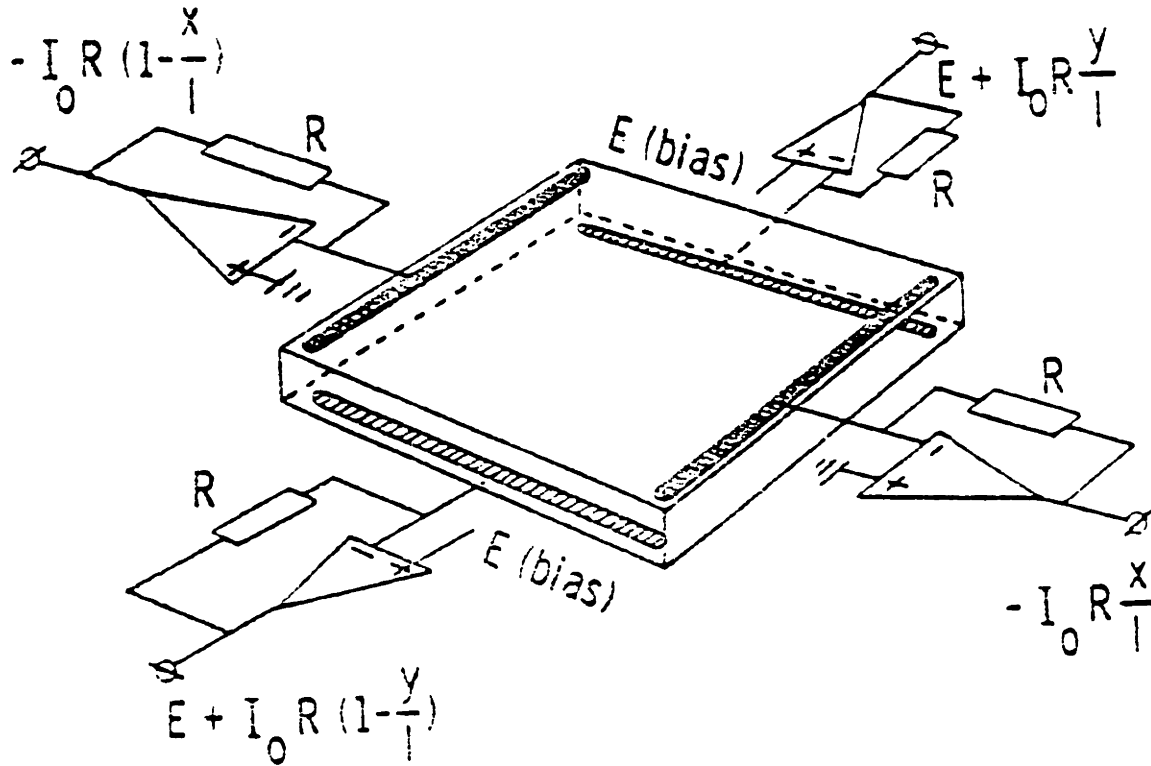
Figure 16. Drawing of a lateral-photo-effect diode from Woltring and Marsolais (1980) [178] page 47

WAVE LENGTH RANGES
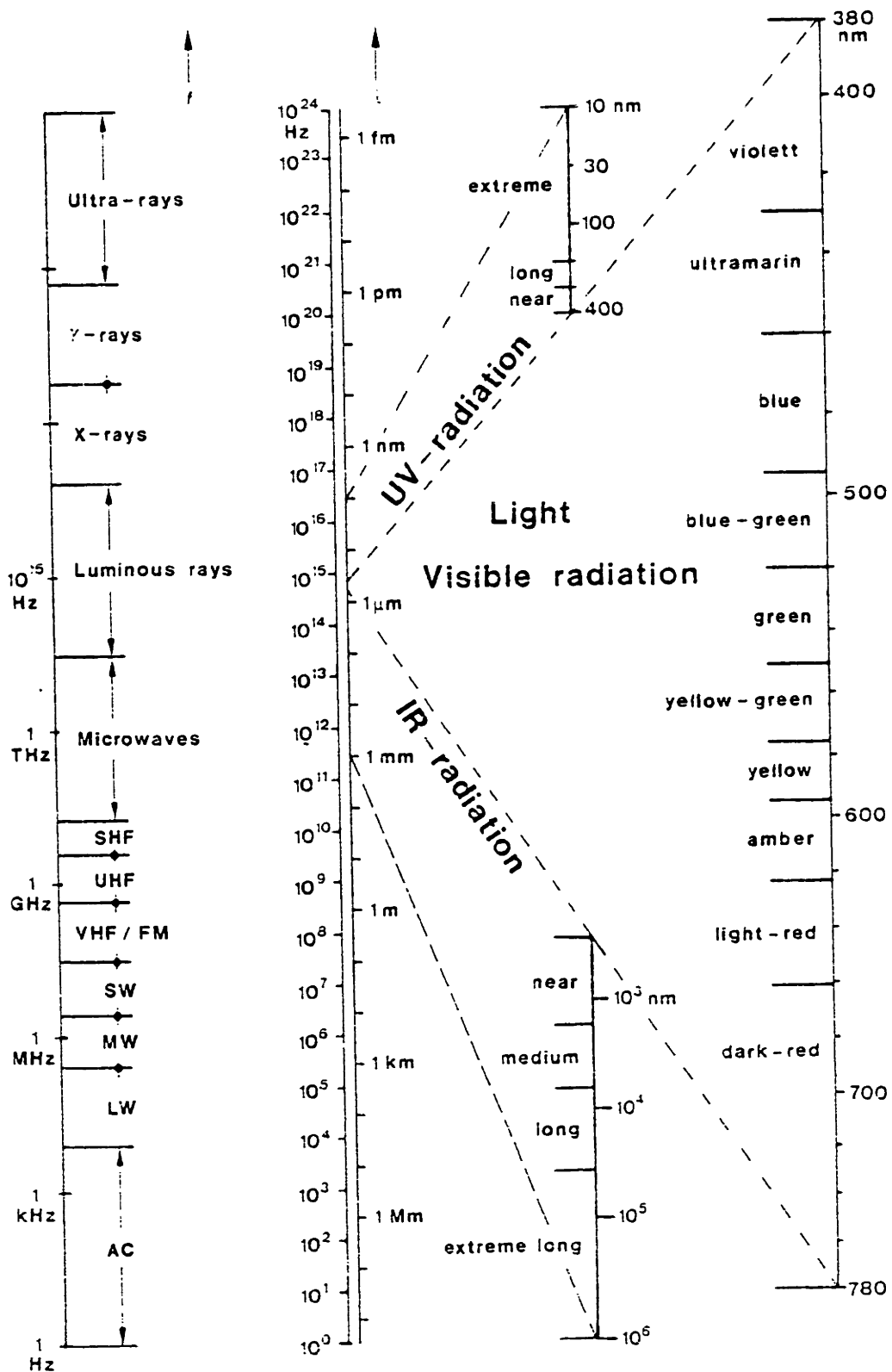
Figure 17. Wavelength Ranges, AEG-Telefunken (1981) [1]
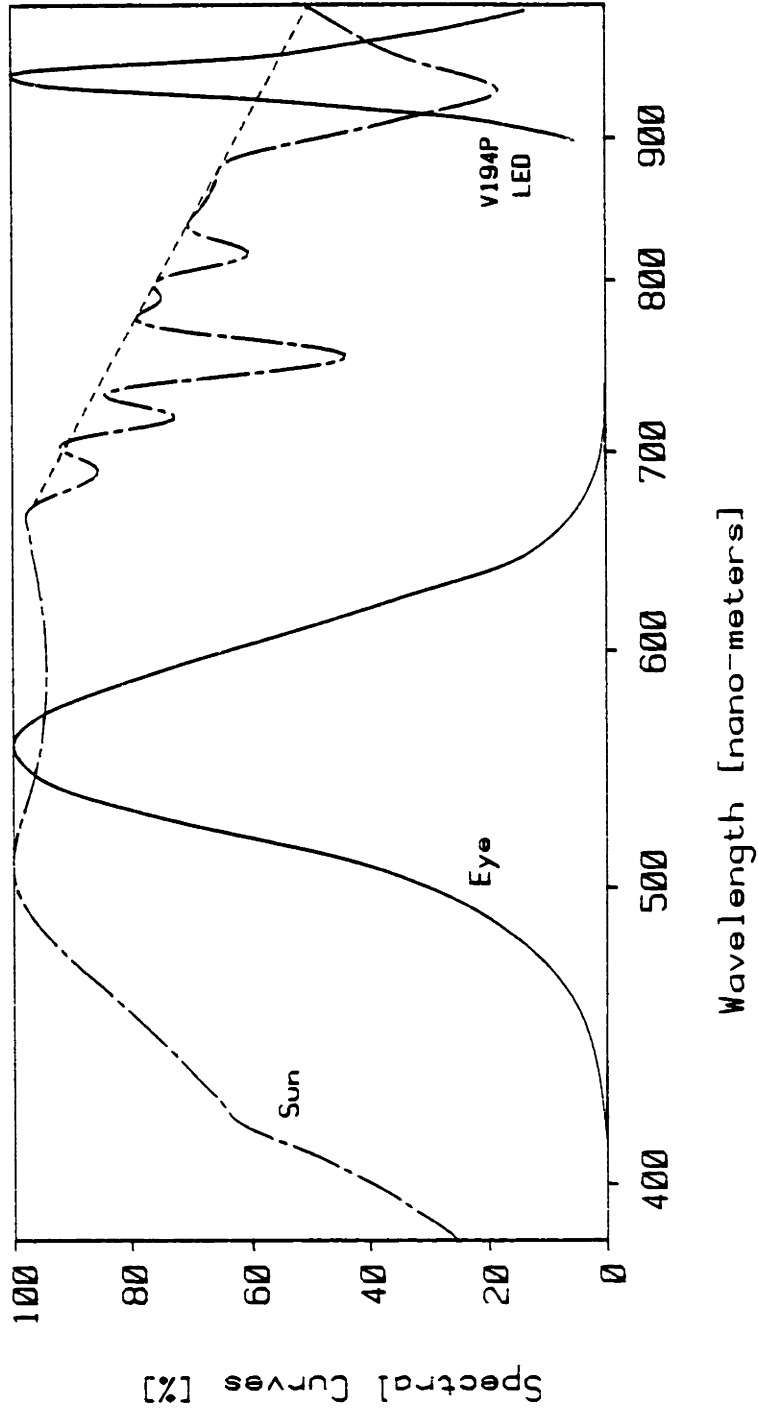
Figure 18. Spectral curves, AEG-Telefunken (1981) [1]

Particularly note the spectral sensitivity of the human
eye (the dashed curve labeled "EYE" in Figure 18). It
becomes insensitive at about 750 nano-meters, well below the
940 nm light used by the Selspot.

The Selspot system also includes an LED Control Unit
(LCU, see Figure 19) which multiplexes current (2 Amperes)
sequentially to each of thirty Infra-red Light Emitting
Diodes (LED). Normally the 2 amperes are supplied through
an umbilical by a transformer and rectifier operating from
line voltage; however, a rechargable battery can be used to
eliminate the trailing wires from the subject. The
umbilical also carries synchrony information from the master
multiplexer clock in the LCU back to the Main Unit. With
the battery in use, synchronization is provided optically by
a 55-micro-second light pulse from LED number 1, instead of
the usual 50. The LEDs used produce light with a spectrum
centered at 940 nanometers.

> "The influences of background light and diode
> leakage current are eliminated by measuring the
> change in output signals when an LED is turned on;
> thus, steady background illumination will not
> affect measurement precision."
> Woltring (1980) [178] page 47.

Each LED is allotted 100 microseconds, and the system
produces a blank pulse of 200 microseconds every frame
(where a frame is defined as one sweep through all thirty
LED channels). This is a frequency of 312 frames per

second. The actual device runs a little bit faster: at 315
Hz. See Appendix C for a timing diagram. The
200-micro-second blank pulse provides synchrony between the
master multiplexing clock in the LCU and the slave clock in
the main unit where the conversion of voltages arriving from
the cameras to digital words takes place. The multiplexed
A/D converter in the Selspot main unit discretizes to 10
bits of resolution (1 part in 1024). The specifications can
be found in Appendix C. The main unit produces 4 - 10-bit
words each time an LED is turned on -- X and Y image plane
location of that LED. With complete knowledge of the camera
locations and orientations, three-dimensional reconstruction
can be performed to determine the location of the LED. The
method used is widely used in photogrammetry and produces
highly accurate three-dimensional coordinates with a small
ammount of calculation and management. A good reference on
photogrammetry is the Manual of Photogrammetry (1980) [144].
Since the cameras produce image plane coordinates in real
time, the digital signals converted by the main unit can be
acquired by a computer and processed to yield 3-D
coordinates in real time. This provides a unique capability
to measure and analyze human motion. Further advantages of
the Selspot transduction system are that the room can be
illuminated normally. Since the markers produce infra-red
light out of the visible range, there are no flashing lights
to distract the subject. Each LED has a mass of 200

milligrams, and a typical plexiglas array with seven LEDs is 35 grams; thus the subject can very lightly encumbered. The LEDs used throughout this investigation were AEG-Telefunken (Somverville, New Jersey) number: V194P. See Appendix G for the manufacturer's specifications.
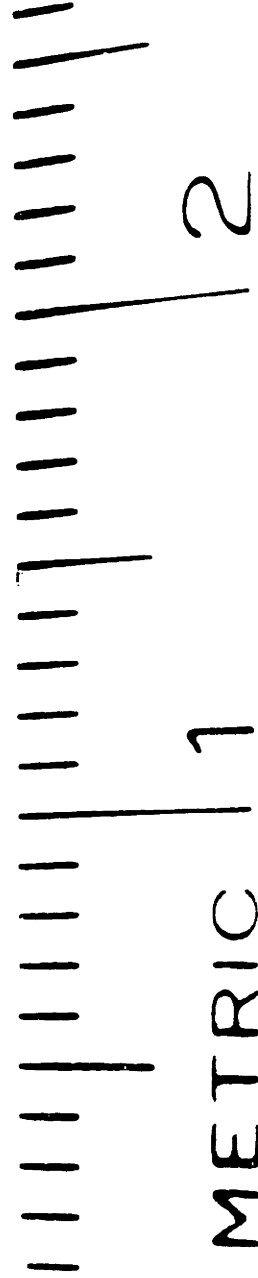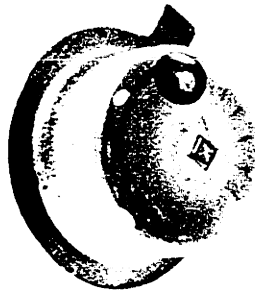
Photograph 1. Close-up of an LED with metric scale.
The smallest divisions on the scale are 1 mm.

Photograph 2. An instrumented ballerina, Patricia Miller
with James Canfield, both members of the Joffrey Ballet.
April 5, 1982

In order to realize the full potential of the Selspot
position transduction system, a computer must acquire and
process the data. The speed, large number of tracking
targets, and high resolution all result in a large volume of
data production: 37,500 16-bit words/second. An electronic
digital computer with a high-speed, 16-bit parallel
interface is the only way to acquire and manage all of these
data and produce useful three-dimensional results. The
computer used in this application is a Digital Equipment
Corporation (DEC) (Maynard, MA) Programmed Data Processor,
(PDP) 11/60, with 256 k bytes of dynamic random access
memory (RAM), a combination of disk drives with a total
storage capacity of 217 mega-bytes. The interface is a
Direct-Memory-Access (DMA) device very similar to the one
used for the forceplate. In this case it is a MDB11-B,
which is more primitive than the DR11-B, but includes some
area on the board for user-installed circuitry. Some
applications of the system do not always use all thirty
LEDs, nor need acquire data at the full 315 Hz. Since there
is no option in the Selspot I to change the frame frequency
or the number of LED channels, this discrimination must be
performed by the computer.* Originally this task was handled
by software, but now the user circuit space on the DMA card
is used to gate the data-available flag to allow only the
frames and channels desired by the user into the computer's
memory. The transfer of function from software to hardware

enables the simultaneous acquisition of other data (forceplate) because the central processor remains idle for the majority of time the Selspot data are being acquired. The specifications for the Selspot interface can be found in Appendix D. This interface provides the capability to place the digital words of Selspot data for the frames and LED channels desired into the memory of a host computer. The DMA nature of the interface insures that the data are accumulated at least as fast at the Selspot produces it. Since the processor is not burdened with the task of managing the accumulation of the data itself, the arrangement is ideally suited to real-time data reduction.

---

* Selective Electronics announced the Selspot II in July, 1981, however none are available as of this writing. This new device has 12 bits (4096 discrete levels) of resolution, improved infra-red compensated optics for increased accuracy, and programmable frequency, number of LEDs, and LED sequence, in addition to allowing multiple cameras, and up to 128 LEDs.

Photograph 3. The computers
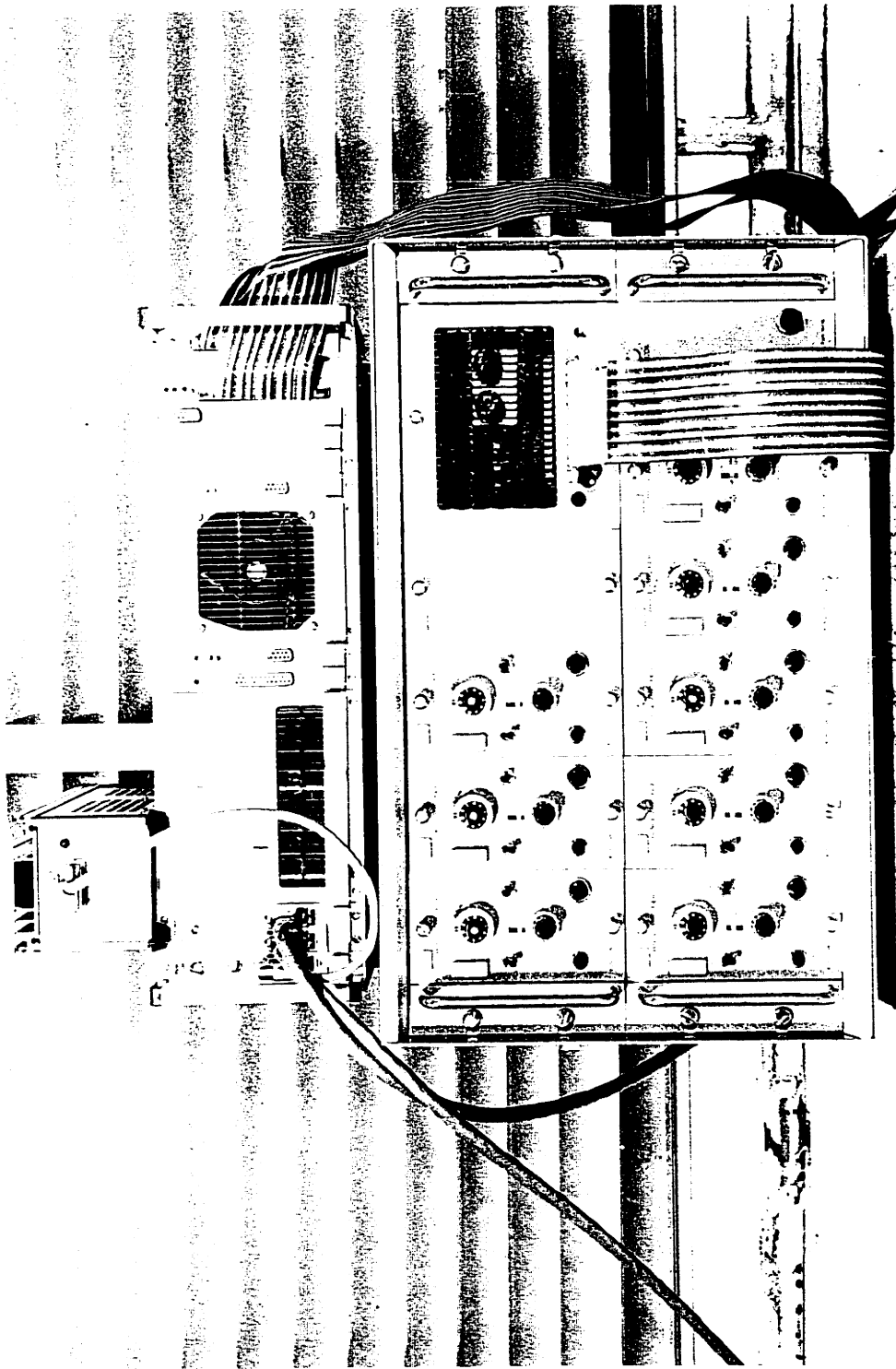
Photograph 4.   The Selspot and Kistler forceplate electronics
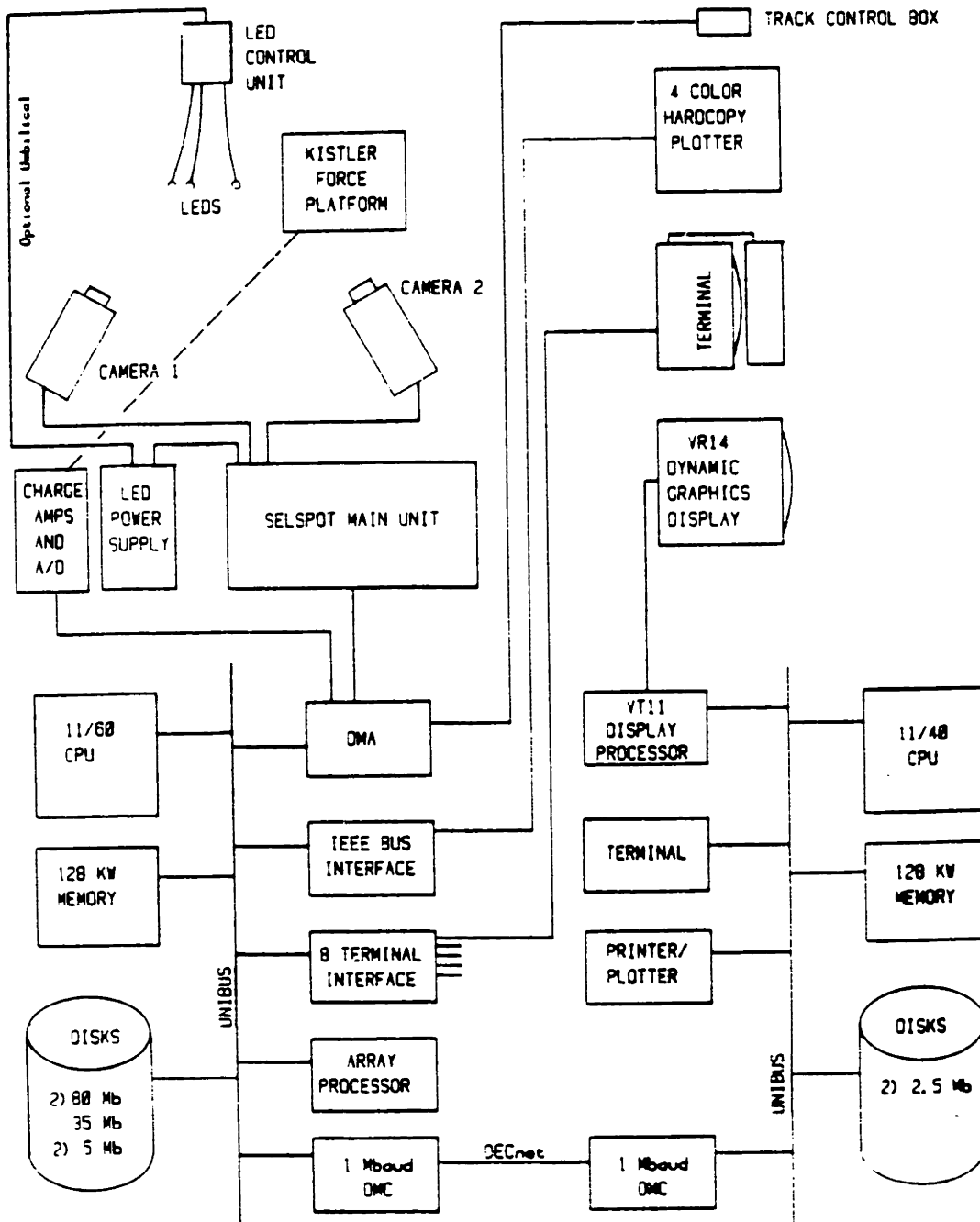
Figure 19. Block Diagram of the Selspot and forceplatform
systems with computers

CALIBRATION

Selcom AB claims that the Selspot I has an accuracy of 1 percent when delivered. The resolution of the system is 10 bits (1 part in 1024), or 0.1 percent. An effort was made to calibrate the positional accuracy of the system to match its resolution -- in other words, to obtain 0.1 percent positional accuracy over the entire field of view of each camera, indicating that a light ray from an LED, reconstructed from the image plane data and corrected by the results of the calibration, would pass within 1 mm of the real LED position at a range where the field of view is 1 meter across. The essential aspects of the calibration involved moving an LED to 12,000 different locations in the field of view of one camera in a plane parallel to the image plane, and determining the corrections necessary to apply to the camera image-plane-coordinate data to move the reconstructed ray to intersect the LED object plane at the LED position. These 12,000 pairs of correction coordinates were fit with a high-order surface function to reduce and smooth slightly the data. The corrections were made computationally efficient by a further reduction to a "look-up table" to drastically simplify (and speed up) the calculations necessary during a correction. The entire calibration process was repeated for both cameras.

Several aspects of the Selspot are important to note: The calibration is sensitive to both focus and lens aperature. Since there is no option to adjust the intensity of each LED, it was decided to perform a full-field-of-view calibration with the LEDs at the distance from the cameras most often encountered in experimental situations -- in this case 3 meters. In order to cover the whole field of each camera, an apparatus had to be constructed that would move an LED (or LEDs) over a square about 1.5 meters on a side. To achieve the 0.1 percent accuracy desired, the calibration must result in a correction that will produce measured LED positions within 1.5 mm of the correct location, for a camera 3 meters away. This requires that the physical LED positioning apparatus have a higher accuracy of LED location: at least 1 mm. The ability to measure LEDs to within 1 mm of their correct position at 3 meters range was a design goal applied throughout this investigation. A computer-controlled stepping motor driving a precision rotary table with a 2-meter bar mounted on it was used. All thirty LEDs were arrayed along the length of the bar at the following radii from the center.

Radius [inches]	Radial Increment [inches]	LED number	Position on side of beam at start
0.0		29	Top
1.0	1.0	28	Bottom
2.5	1.5	27	Top
4.0	1.5	26	Bottom
5.5	1.5	25	Top
7.0	1.5	24	Bottom
8.5	1.5	23	Top
10.0	1.5	22	Bottom
11.5	1.5	21	Top
13.0	1.5	20	Bottom
14.5	1.5	19	Top
16.0	1.5	18	Bottom
17.5	1.5	17	Top
19.0	1.5	16	Bottom
20.5	1.5	15	Top
22.0	1.5	14	Bottom
23.0	1.0	13	Top
24.0	1.0	12	Bottom
25.0	1.0	11	Top
26.0	1.0	10	Bottom
27.0	1.0	9	Top
28.0	1.0	8	Bottom
29.0	1.0	7	Top
30.0	1.0	6	Bottom
31.0	1.0	5	Top
32.0	1.0	4	Bottom
33.0	1.0	3	Top
34.0	1.0	2	Bottom
35.0	1.0	1	Top
35.0	0.0	30	Bottom

Table 2 LED radii used for Selspot camera calibration
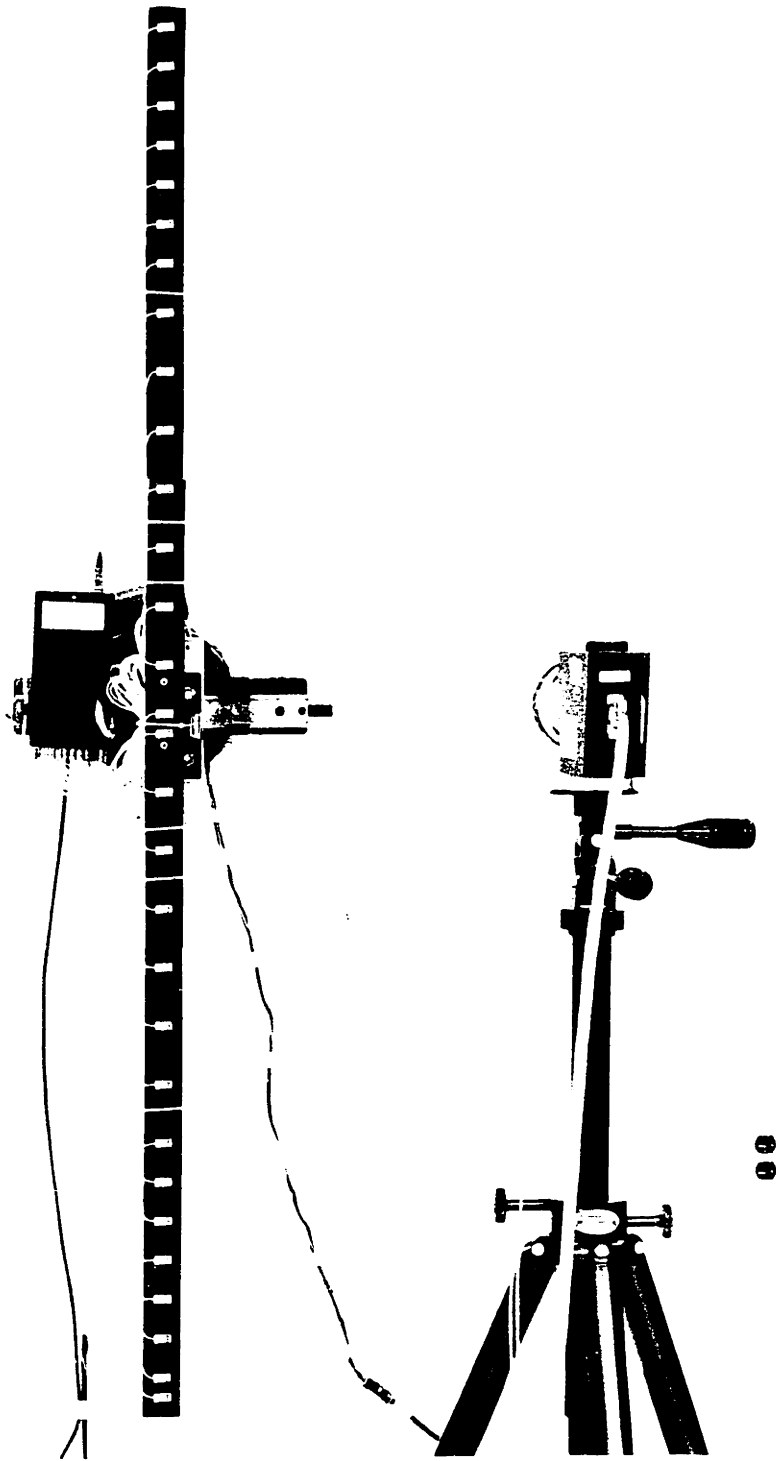
Photograph 5.   The calibration apparatus

Note that half the LEDs were on one side of the bar, and the other half on the other side. This was done to balance the bar to avoid the influence of gravity on the results. The LEDs are also more closely spaced at large radii than at small. This was done to provide a more uniform distribution of measured points over the field of view since a given angular increment will produce more circumferential motion at large radii. The bar rotated about its mid-point and was always the diameter of a 2-meter circle.

The stepper motor required 80 steps to index the rotary table by 0.9 degrees. The radial runout and rotational accuracy of the table were found (by direct measurement) to be precise enough to be able to locate the endpoint of the bar to well within +/- 0.5 mm.* The backlash in the gearing was removed with a string wrapped around the outer circumference of the table with a weight hanging from the free end. This applied a constant torque due to gravity on the table in the direction opposite to rotation. Since the bar and table were balanced, the anti-backlash torque was constant for all angles. Additionally the 2 meter aluminum, right-angle, cross-section bar used was found to deflect by less than 0.2 mm at the extremes under its own weight plus the weight of the LEDs.

To calibrate each camera, a mapping between the "real space," in the plane of rotation of the bar with the LEDs, and the "detector plate space," in the camera image-plane coordinates produced by the diodes and converters, was necessary. This mapping was produced by indexing the stepper table and bar through 400 - 0.9 degree steps (to complete a full circle) and storing the Selspot-measured location of each LED at each step. The camera's optical axis was aligned with the rotational axis of the rotary

---

* The stepper driven rotary table was found to have a total indicator runout of 0.006 inches, and an angular accuracy of better than 0.7 milli-radians.

table. At each step the bar was allowed 3 seconds to settle, to completely eliminate any dynamic effects; then 100 samples of each LED were taken for the camera being calibrated. The average of the 100 samples for each LED was stored in a data file, along with the standard deviation and the maximum error. Thus a matrix of detector plate coordinates for 12,000 points (30 LEDs times 400 steps) was produced. Because of the accuracy of the LED positioning apparatus, the actual position of each LED at each angular bar position was known.

Despite the intrinsically cartesian nature of the data produced by the detector plate, all of the calibration calculations were performed in the polar coordinate system of the rotary table. Two functions were fit to the location of each LED. The first characterized the deviations of the measured location radially from a circle, the second characterized the circumferential deviations. Each function consisted of fourteen terms including a constant, a linear term, and twelve terms of a Fourier series. The goal, once again, of the calibration was to obtain an accuracy of 0.1 percent, or stated another way, to produce corrected camera, focal-plane, detector-plate data accurate to 1 unit out of 1024. The application of these correction functions always brought each measured trajectory to within the 1-unit tolerance. Because 60 functions -- one radial and one circumferential for each LED -- each with fourteen terms for

each camera, represented an intractable amount of calculation, a new representation was desired. In fact, since one of the goals throughout has been high-speed data analysis, a "look-up table" was developed to allow the implementation of the calibration corrections in real time. With an unlimited amount of computer memory, the ideal table would contain a correction value to be applied for each of the 1024 x 1024 discretized points on the detector plate. Unfortunately, the computer installation available contained a small fraction of the 1 megabyte a look-up table of this size would require; further, four such tables would be required to perform both orthogonal corrections for both cameras. Instead, tables 51 by 51 were used to conform to the 64-k-byte restriction imposed by the 16-bit address-bus architecture of the PDP 11/60. To create these 51 x 51 lookup tables a single function was desired to fit all thirty of the 14th order functions describing one polar correction for one camera. Tenth-order spherical harmonics were chosen. These functions were able to fit the data with high fidelity: RMS errors of less than 0.9 units, and maximum errors of 3.0 units. This function allowed a relatively simple discretization into cartesian lookup tables. Figures 20 to 23 are contour plots of equal correction magnitude for radial and circumferential (polar) errors for both cameras. These "surfaces" were generated by determining (for instance) the radial error at each point on

the detector plate, and plotting that quantity as if it were a height above or below the detector plate plane. If the transduction system had been found to need no correction, there would be no contours at all; the "surface" representing the value of the correction at each point would be flat with a height of zero. Note that the manufacturer's claim of 1 percent un-calibrated accuracy is supported by "hills" and "valleys" of maximum height 10 units (out of 1024). The sign convention for positive radial error is out; positive circumferential error is clockwise. The "pathological" nature of the corrections is due to a summation of errors from the cartesian detector plate and the polar lens and is caused by manufacturing defects in both.
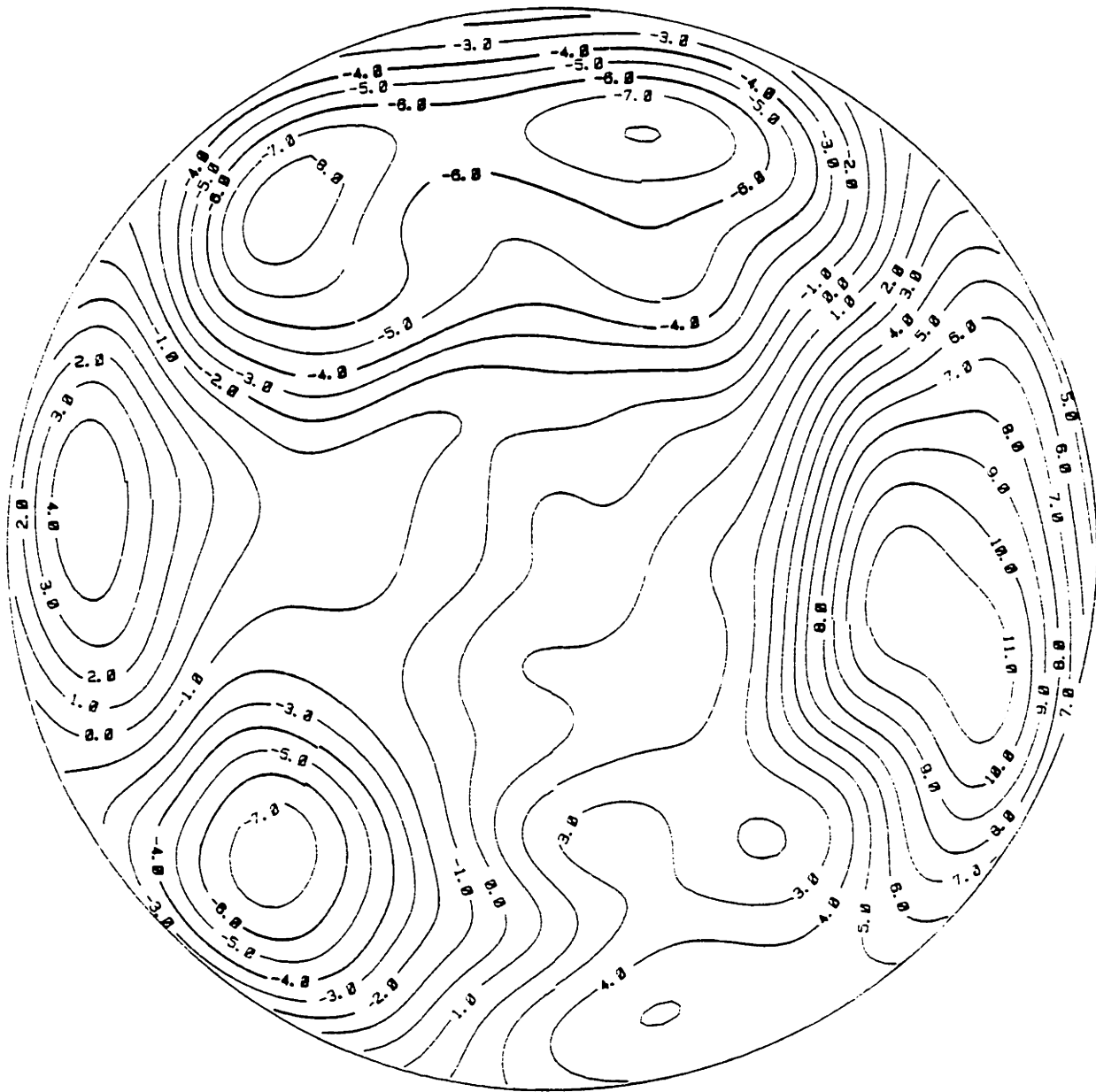
Figure 20. Plot of equal radial correction for
Camera 1 calibration [Selspot Units]
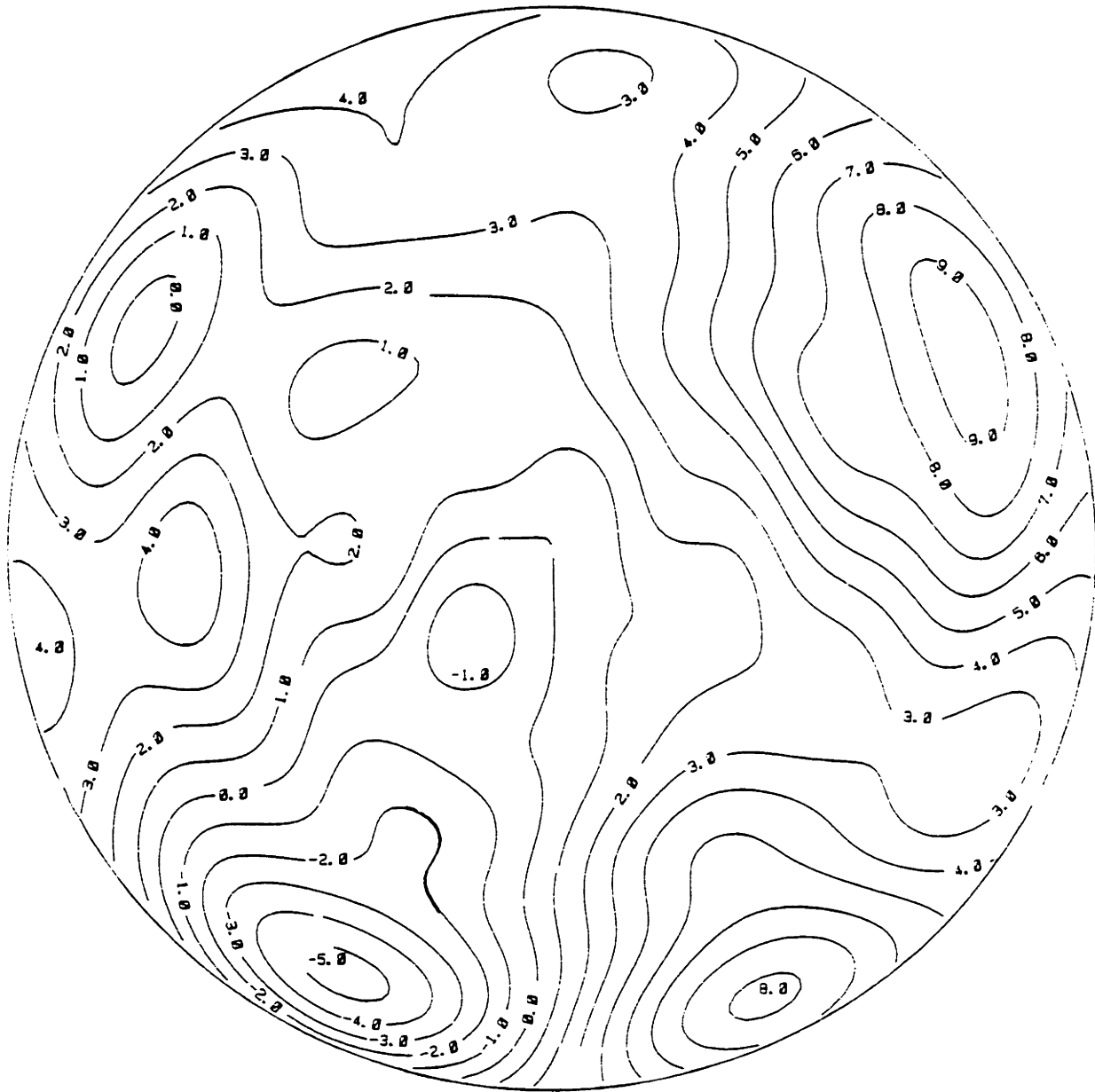
Figure 21. Plot of equal circumferential correction for
Camera 1 calibration [Selspot Units]
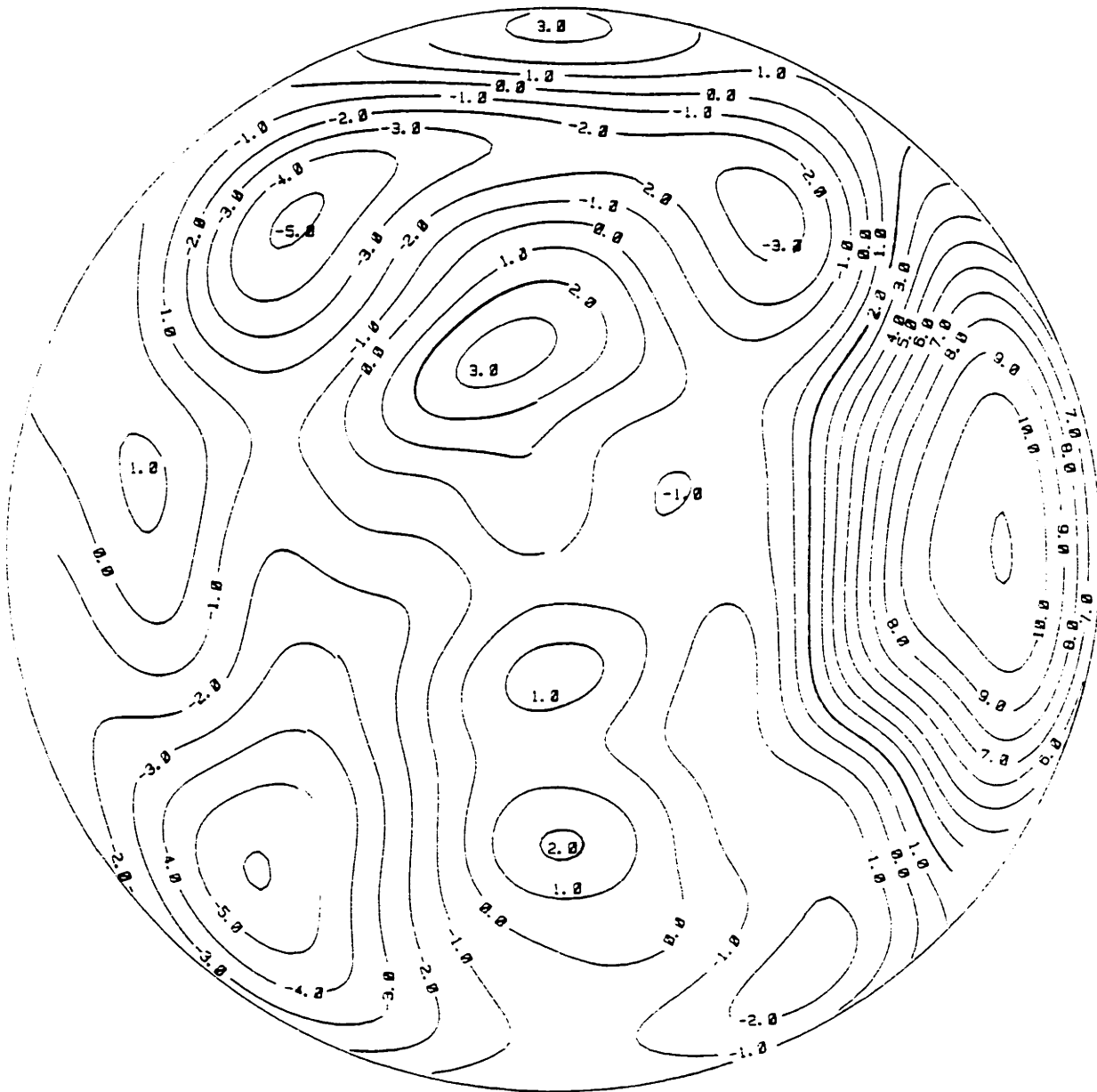
Figure 22. Plot of equal radial correction for
Camera 2 calibration [Selspot Units]
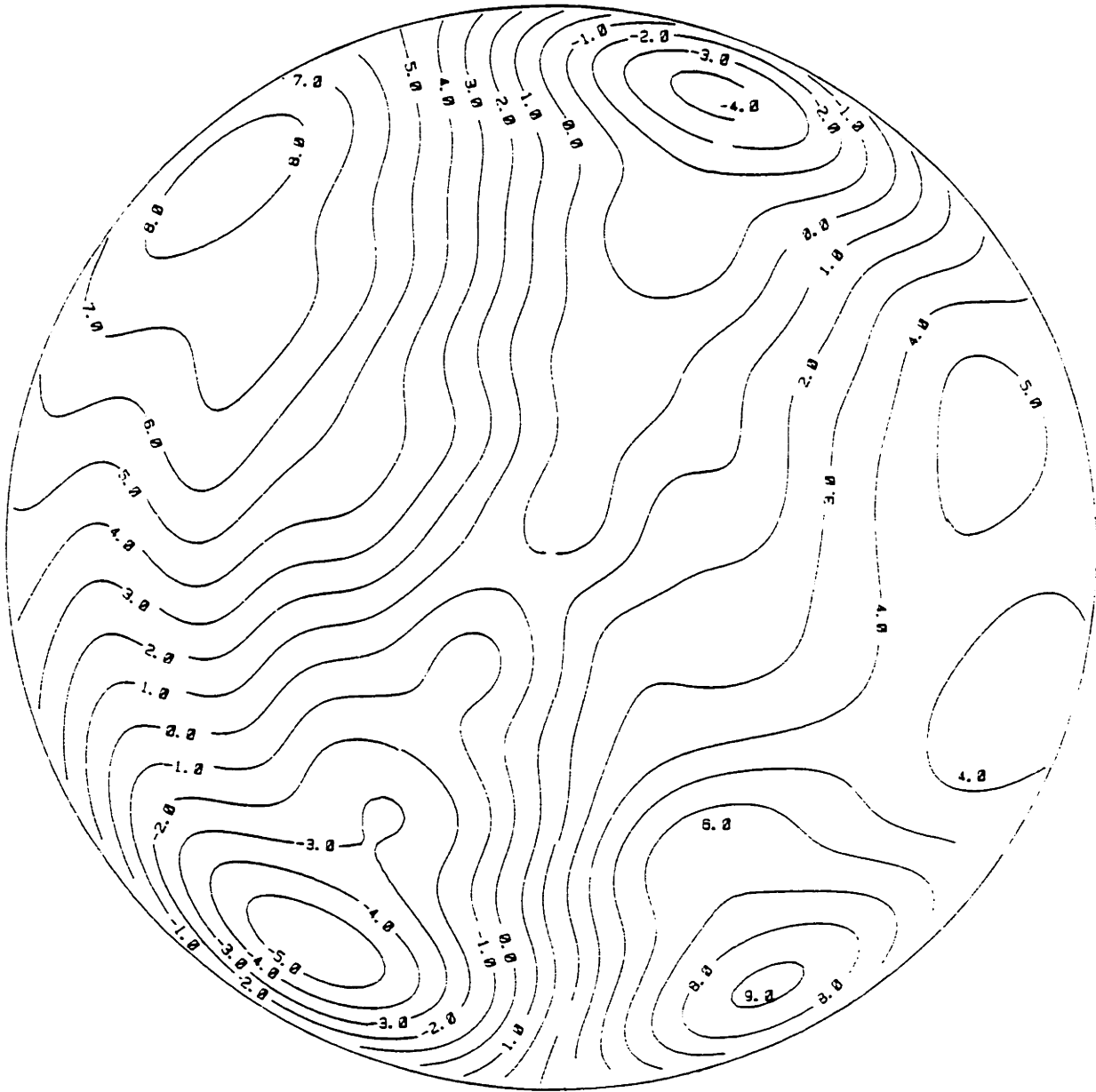
Figure 23. Plot of equal circumferential correction for
Camera 2 calibration [Selspot Units]

The 51 x 51 look-up tables were further justified, considering the goal of one part in 1024 accuracy; by no two adjacent table entries differing by more than 1 unit. Since data points to be calibrated that fall between entries in the table are linearly interpolated, this entire calibration scheme easily attained 0.1 percent accuracy.

A final correlation completed the calibration. As described thus far the look-up tables will make circles in real space appear as circles in detector plate space, but there is a scaling between the radius of the real circle in physical units and the radius of the detected circle in Selspot discretized units. The number relating these two is the focal distance of the lens. In order to determine the focal distance (the distance from the back nodal point of the lens to the focal plane), however, the location of the front nodal point must be known, so that the corresponding distance between the front nodal point and the plane of the LEDs can be found. An optical bench with a nodal slide is required to measure the node positions of a lens. This measurement would have been made even more difficult because the light used for detection is not visible (940 nm infra-red). Fortunately, Canon USA courteously provided the nodal locations of the f 0.95 50 mm T.V. lens used with the Selspot for 940 nm infra-red light. This last data permitted the calculation of the focal distance which completed the calibration (in this case it was found in

Selspot units for speed of calculation when using the correction tables).

Figure 24. Thick lens optical geometry
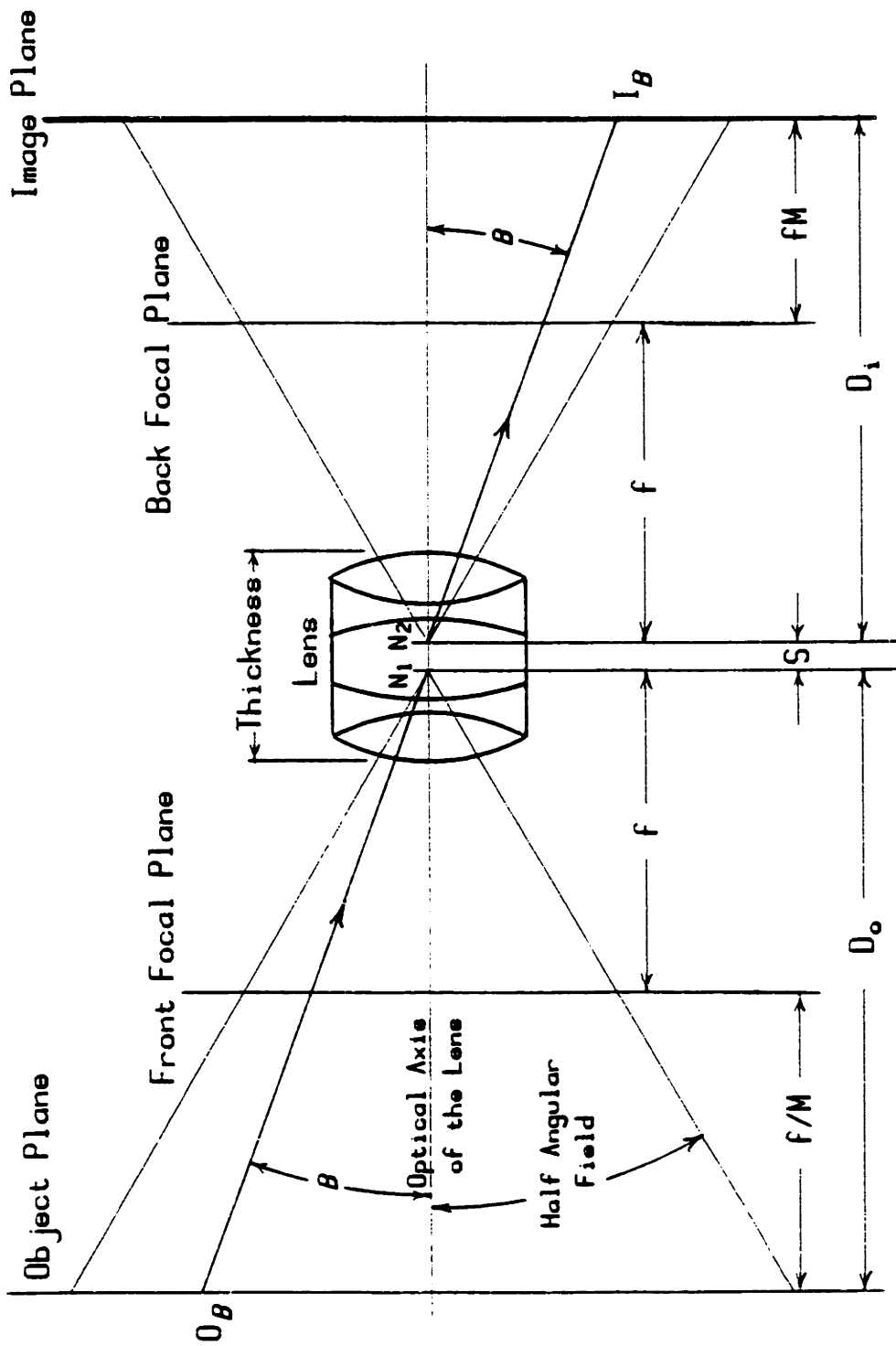
$D_o = f(1 + 1/M)$ [Object Distance]

$D_i = f(M + 1)$ [Image Distance]

N_1 = Front Nodal Point

N_2 = Back Nodal Point

f = Equivalent Focal Length

M = Magnification

S = Separation

Canon   TV-16   50 mm   f 0.95   Lens
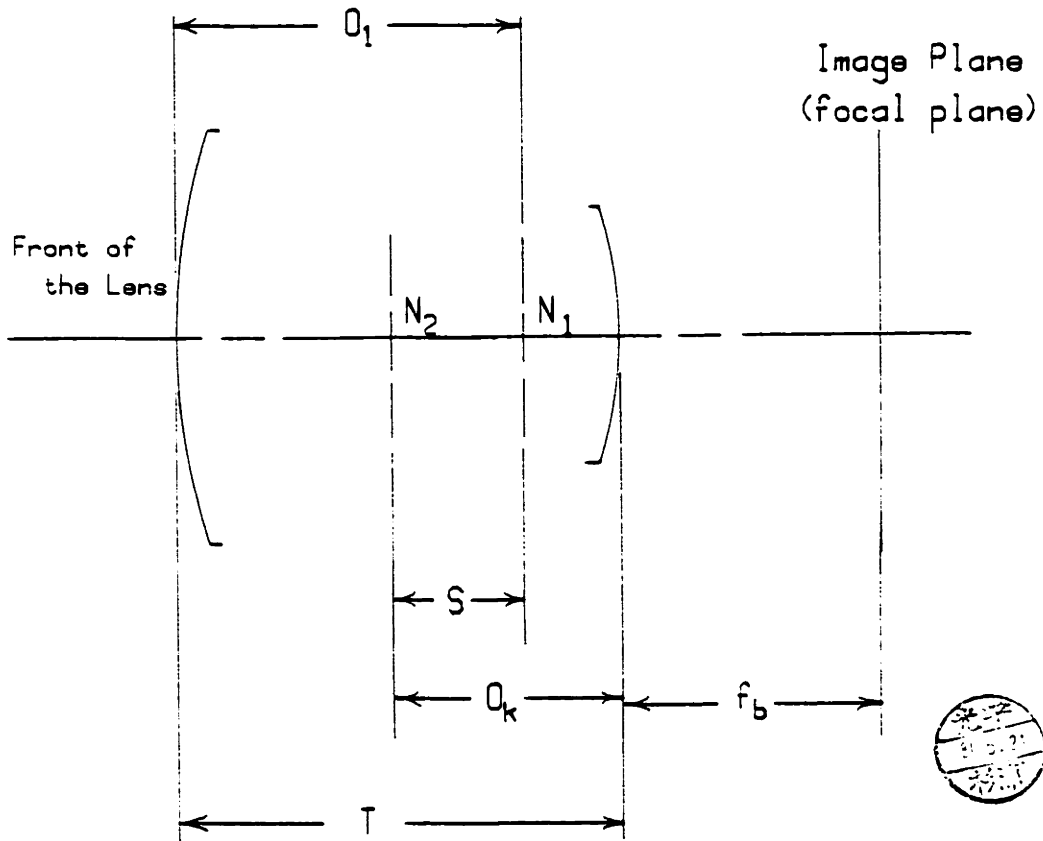
	587. 56 nm *	940. 0 nm †
Focal Length $(O_k + f_b)$	52. 40	52. 67 [mm]
T [Thickness]	55. 04	55. 04
f_b [Back Focal Distance]	20. 37	20. 74
O_1	41. 25	40. 77
O_k	32. 03	31. 93
S [Separation]	18. 24	17. 66
N_1 [Front Nodal Point]		
N_2 [Back Nodal Point]		

Figure 25.   Selspot lens optical dimensions

* Sodium "d-line"

† Infra-red

N. B. This is a "Crossed-Node" Lens. N_1 is behind N_2
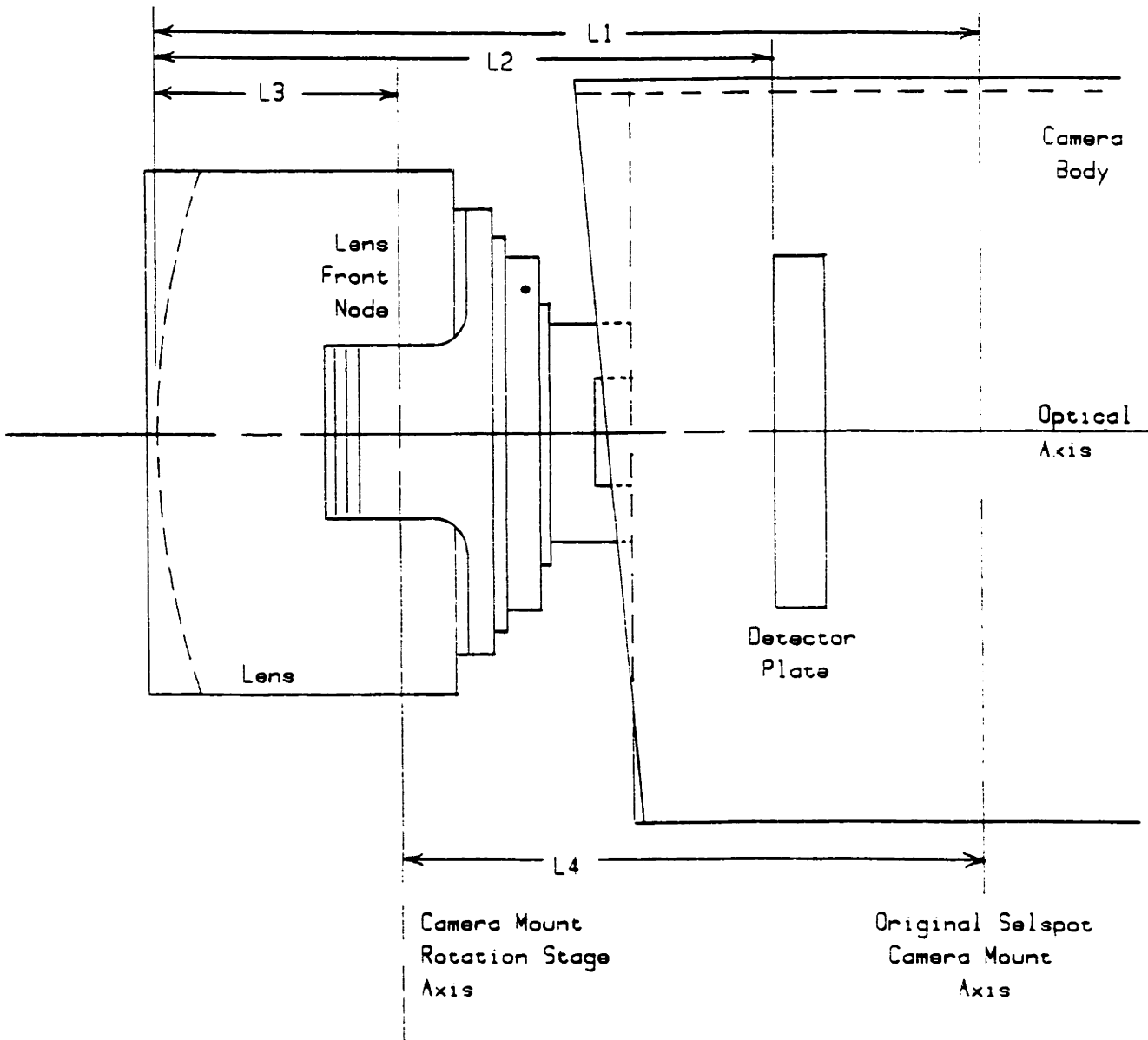
Figure 26. Selspot camera dimensions

Figure 27.   Calibration focal distance determination
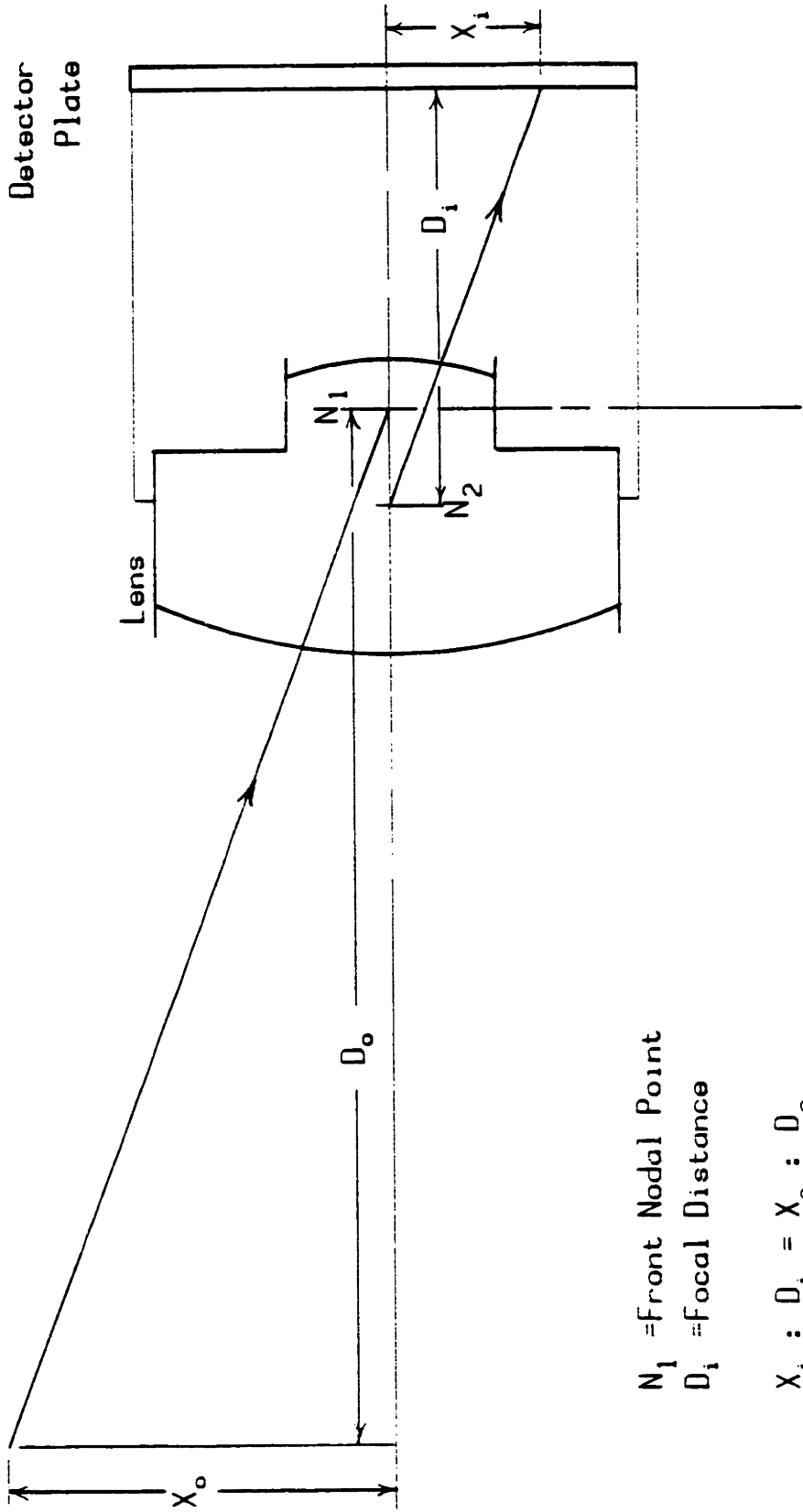
N_1 =Front Nodal Point
D_i =Focal Distance

$X_i : D_i = X_o : D_o$

CAMERA MOUNTING

As will be seen in the next section, in order to reconstruct photogrammetrically the three-dimensional position of a marker in the field of view of two cameras, the positions and orientations of both cameras must be known. To produce 1-mm accuracy at 3-meters range, the cameras must be positioned within 1 mm of the desired location and must be angularly positioned within 1.1 minutes of arc. To realize these stringent tolerances the cameras are mounted on an aluminum optical bench (Klinger Scientific Corp., Lake Success, New York). Each camera is mounted on a carriage that holds a micrometer adjustable translation stage and a fine screw adjustment rotation stage on top of that. All positions of the camera are referenced to the lens front node point. A mounting plate was fabricated to locate that node directly over the vertical rotational axis of the rotary stage. Photograph 6 is an overall view of the optical bench, and Photograph 7 is a closeup of one camera with its carriage, translation, and rotary stages.

Photograph 6. The optical bench with both cameras
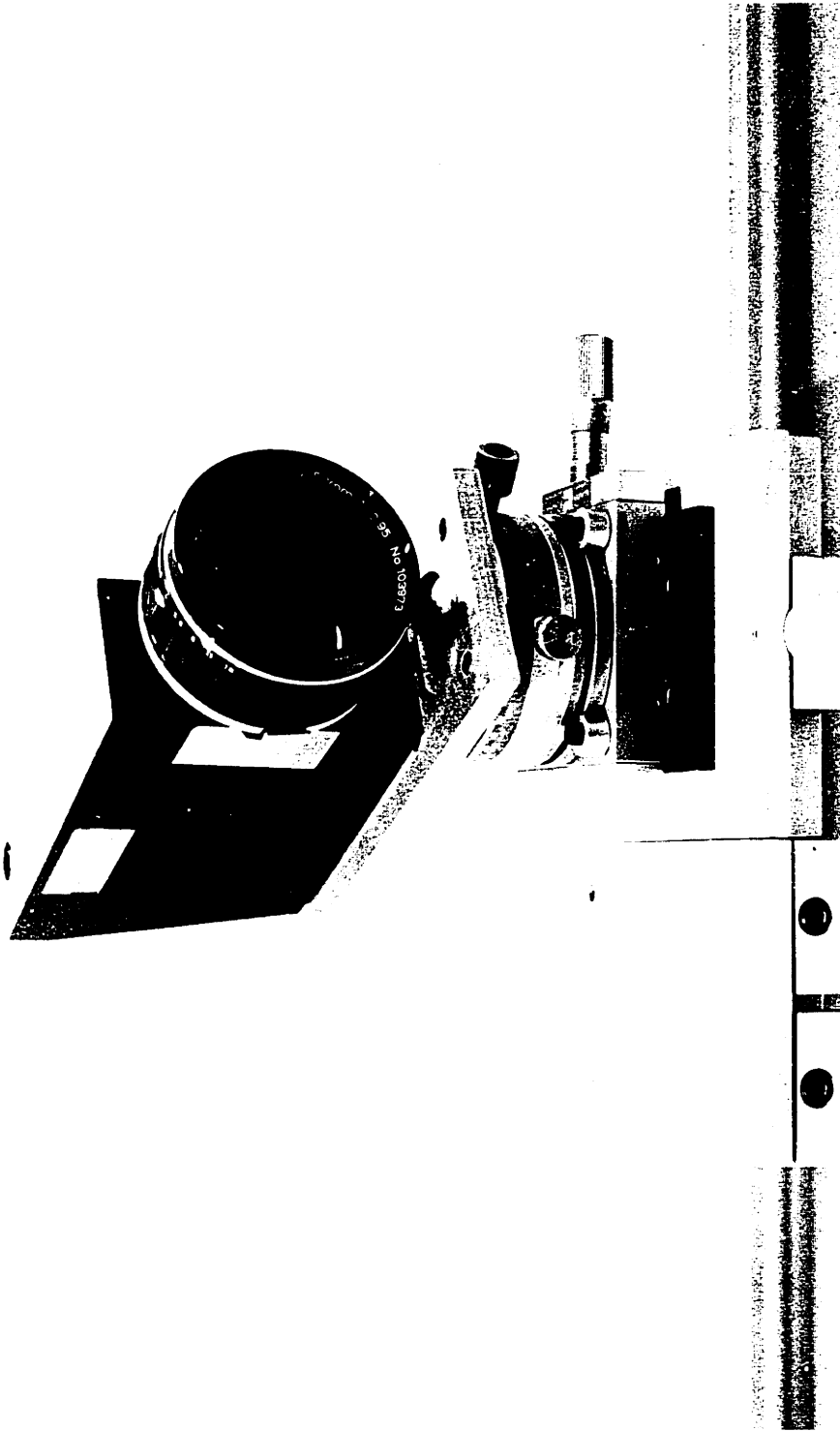
Photograph 7. Closeup of one camera with carriage and stages

The translational resolution of each stage is 0.001 mm, and the angular resolution is 1.0 minutes of arc. The aluminum beam comprising the central element of the optical bench was found to deflect less than 1 mm in the center under the combined load of two cameras and its own weight. The specifications for the bench and associated hardware can be found in Appendix E.

KINEMATIC PROCESSING TECHNIQUE

To fully utilize the kinematic potential of 1-mm precision and accuracy, and 315-Hz sampling rate on thirty markers, a method of trigonometric reconstruction of the three-dimensional location of each LED at each sample and of grouping sets of three or more LEDs into rigid arrays, (including defining an imbedded body-coordinate system with respect to that array) was developed (Conati (1977) [34]). This technique has been presented previously, and only a brief review will be included here.

Given the locations and orientations of two cameras (optics and detector plate) and knowledge of the optics (as described in the calibration section), a simple trigonometric reconstruction of the light ray to each camera from an LED in the common field of view can be performed. This involves using the detector plate image-location

coordinates, and the lens node point locations to determine the direction of the two rays (one into each camera).

The global coordinate system used throughout this investigation is depicted in Figure 28. The origin is at the front nodal point of the lens of camera number 1, and the positive X direction proceeds through the front nodal point of the lens of camera number 2. The positive global Y axis points vertically straight up, and Z is therefore straight out into the experimental space. The experimenter is provided with the option of translating this global coordinate system (GCS) to other locations during an experiment; however, the orientation of the GCS always remains parallel to the one just described.
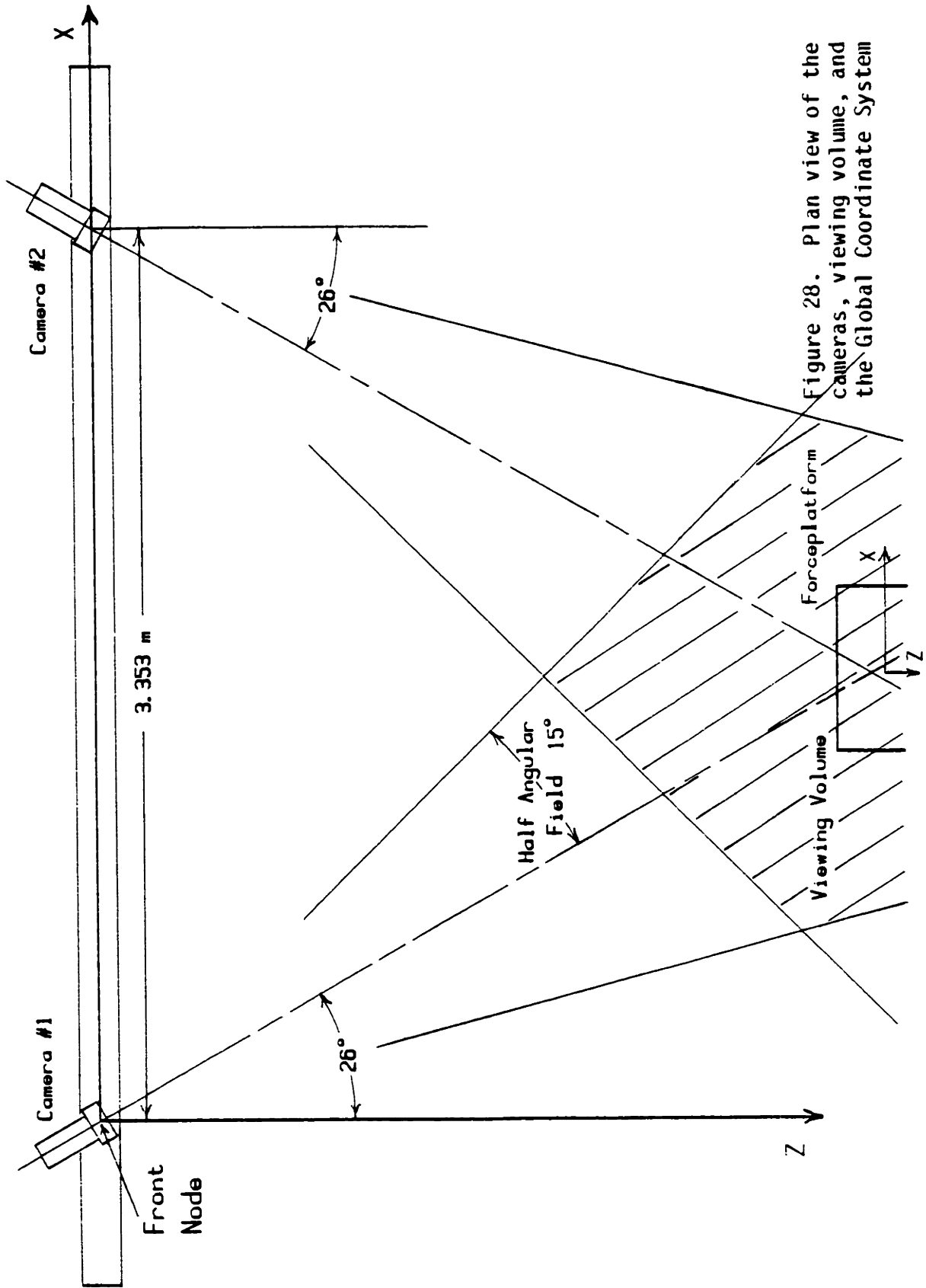
Figure 28. Plan view of the cameras, viewing volume, and the Global Coordinate System
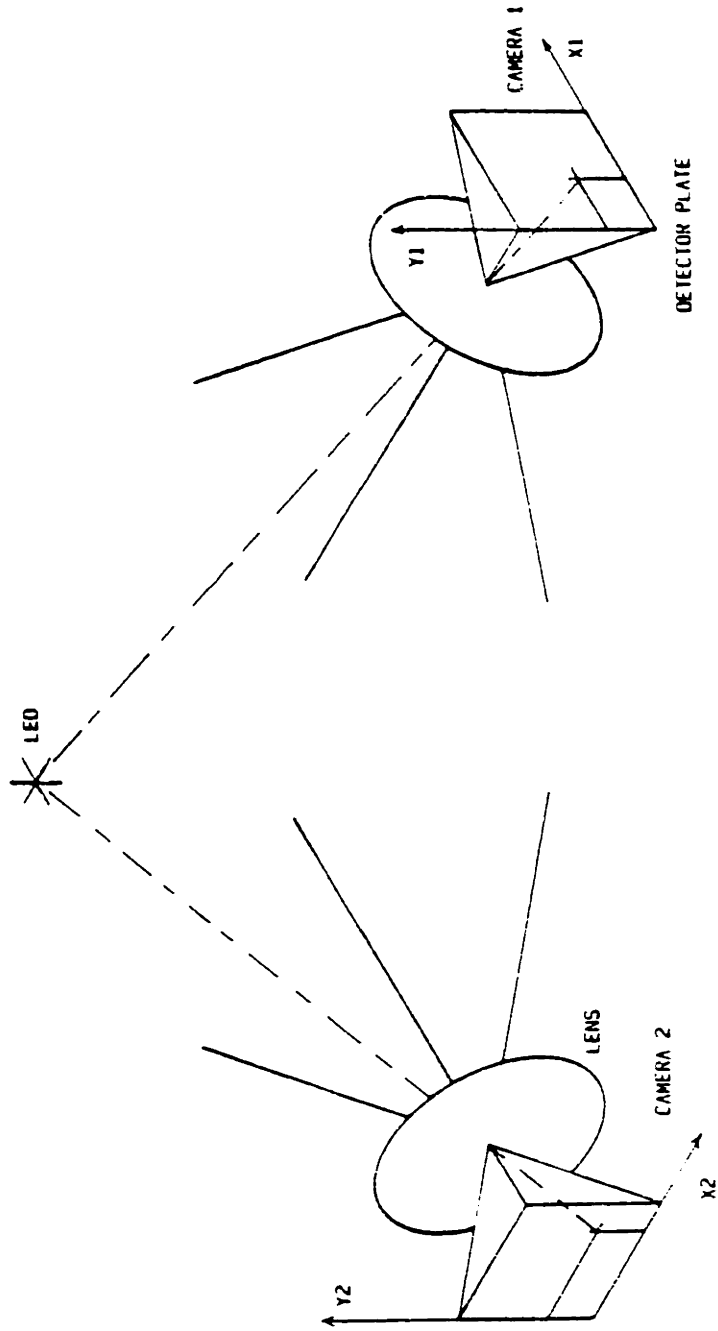
Figure 29. Lenses, detector plates, and 2 reconstructed rays

Figure 30. Plan view of the cameras, showing the 3-D reconstruction
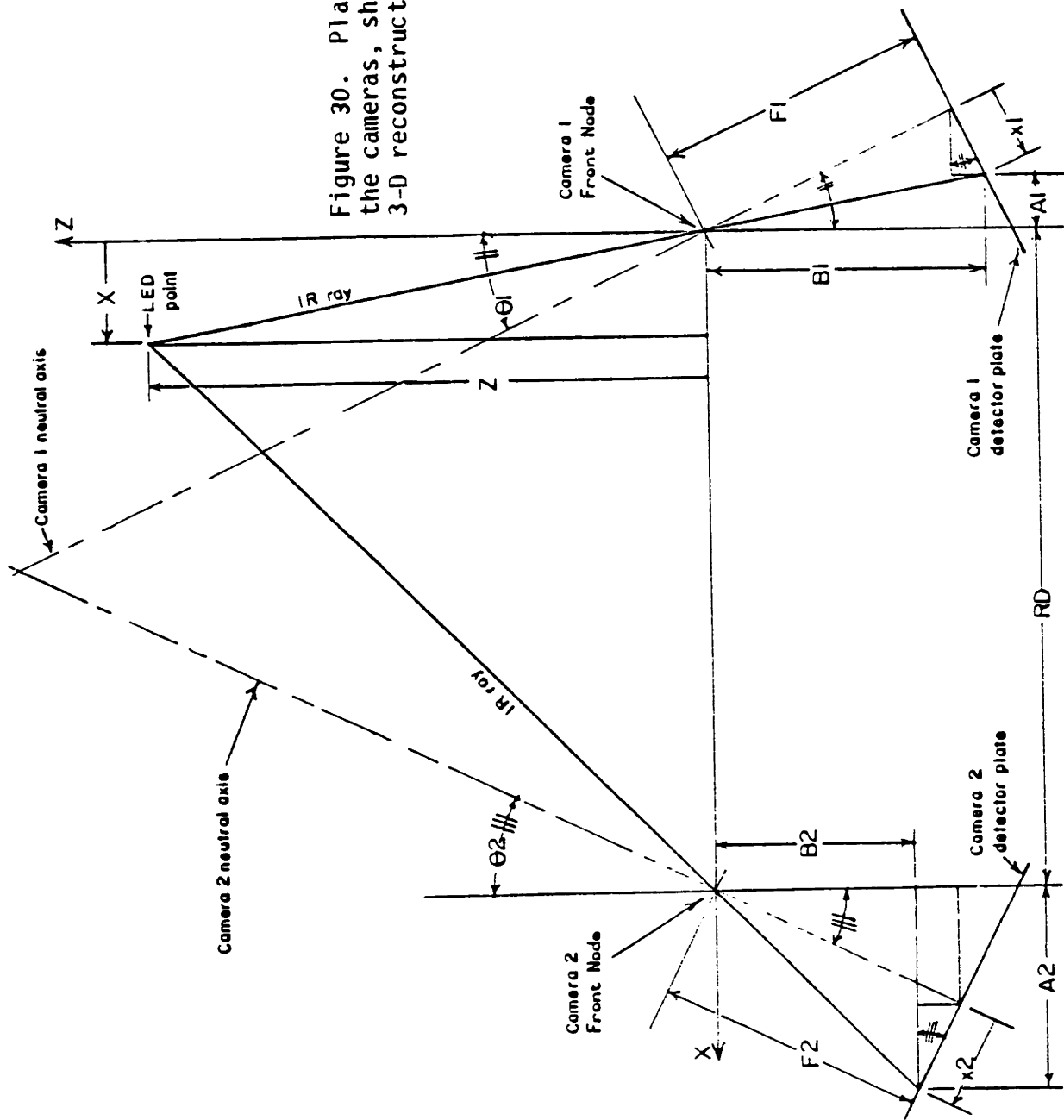
$$\frac{A1}{B1} = \frac{X}{Z}$$

$$\frac{A2}{B2} = \frac{(RD-X)}{Z}$$

$$X = \frac{RD\left(\frac{B2}{A2}\right)}{\left(\frac{B2}{A2}\right) - \left(\frac{B1}{A1}\right)}$$

$$Z = X\left(\frac{B1}{A1}\right)$$

or:

$$Z = \frac{(X-RD)}{\left(\frac{B2}{A2}\right)}$$

finally:

$$\frac{Y}{Z} = \frac{Y1}{B1} = \frac{Y2}{B2}$$

$$Y = Z\,\frac{\left(\frac{Y1}{B1}\right) + \left(\frac{Y2}{B2}\right)}{2}$$

where:

$$A1 = (X1)\cos(\theta1) - (F1)\sin(\theta1)$$

$$B1 = (X1)\sin(\theta1) - (F1)\cos(\theta1)$$

$$A2 = (X2)\cos(\theta2) - (F2)\sin(\theta2)$$

$$B2 = (X2)\sin(\theta2) - (F2)\cos(\theta2)$$

Xi, Yi = The light spot image plane coordinates from camera i

Fi = Camera i focal distance

RD = Distance between Camera 1 and Camera 2 front nodes

Figure 31.   3-D coordinate reconstruction equations

In the best of experimental circumstances, these reconstructed rays will intersect at the LED location. Unfortunately, due to small remaining uncalibrated error, noise, and infra-red light reflection, they frequently miss. The most common cause of skew reconstructed rays is reflection from the floor. Floor reflections cause problems because lateral-photo-effect diodes produce currents at their edges proportional to the location of the centroid of an image, and the reflected image will not, in general, be coincident with the major image. Data in these cases will be erroneous. Two different techniques are used to ameliorate the influence of reflections. The first deals with the reflections directly. To make a surface non-reflective to infra-red requires either a special coating (the right "color" to absorb the IR) or a surface with an average roughness height slightly less than the wavelength of light being used. Unfortunately, both of these are surfaces far too delicate to use as a floor covering for gait. A third possibility is to disperse the unwanted light. Photographs 8 and 9 are still photographs taken with a 35-mm single-lens-reflex camera using special high-speed, infra-red sensitive, black and white film (Kodak 2481 HIE135). Both were taken at night, with no visible light; only a set of LEDs in a closely spaced, rigid array was in the field of view. The still film camera was located near Selspot camera number 1. Photograph 8 was taken

without any floor covering, Photograph 9 was taken with a 3-mm thick, green, "indoor-outdoor" carpet covering the floor. In addition to the floor covering, an attempt was made to cover everything in the field of view during an experiment (including the subject) with a flat (non-glossy) fabric, or paint.

Photograph 8. Uncovered floor
infra-red refléctions
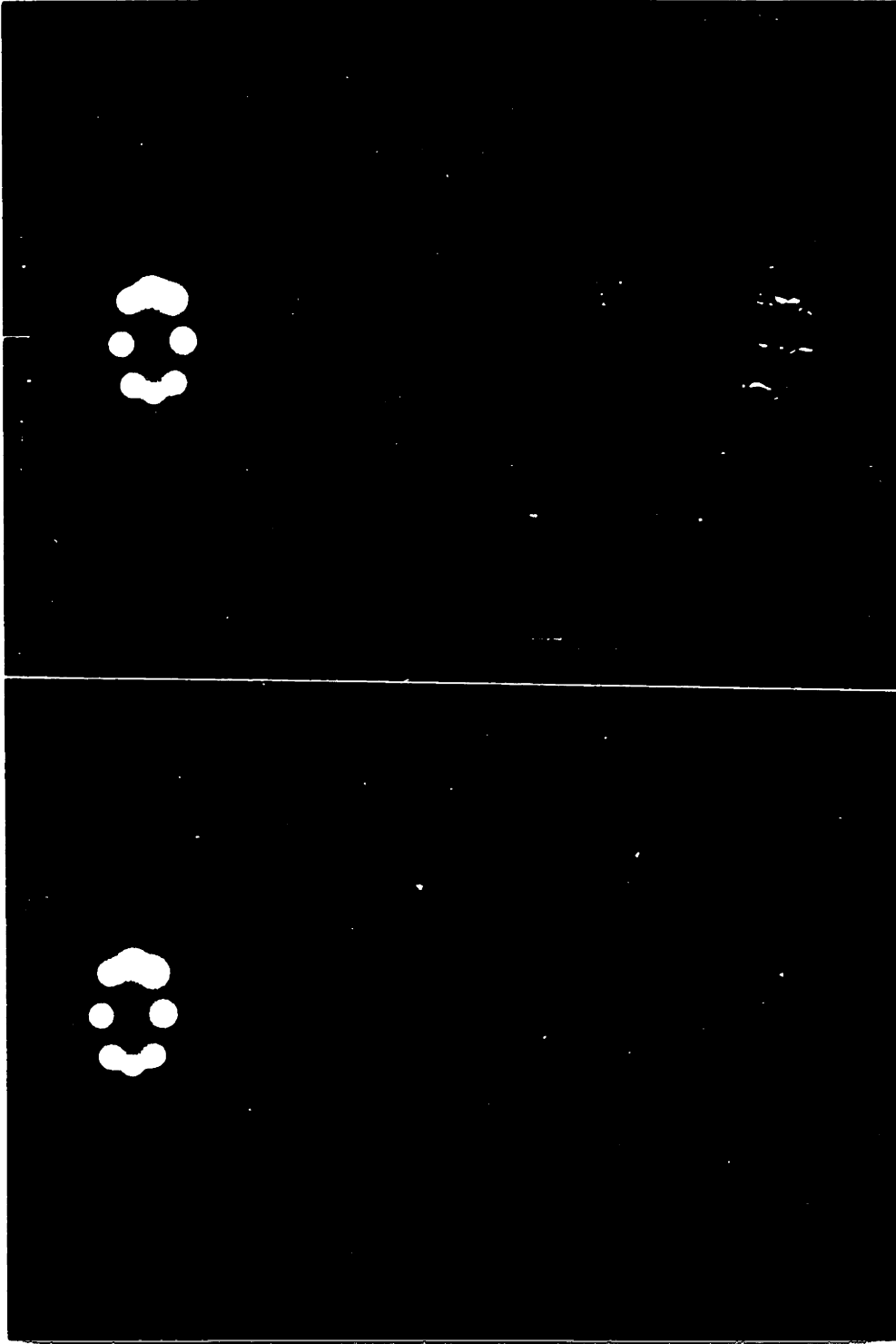
Photograph 9. Covered floor
infra-red reflections
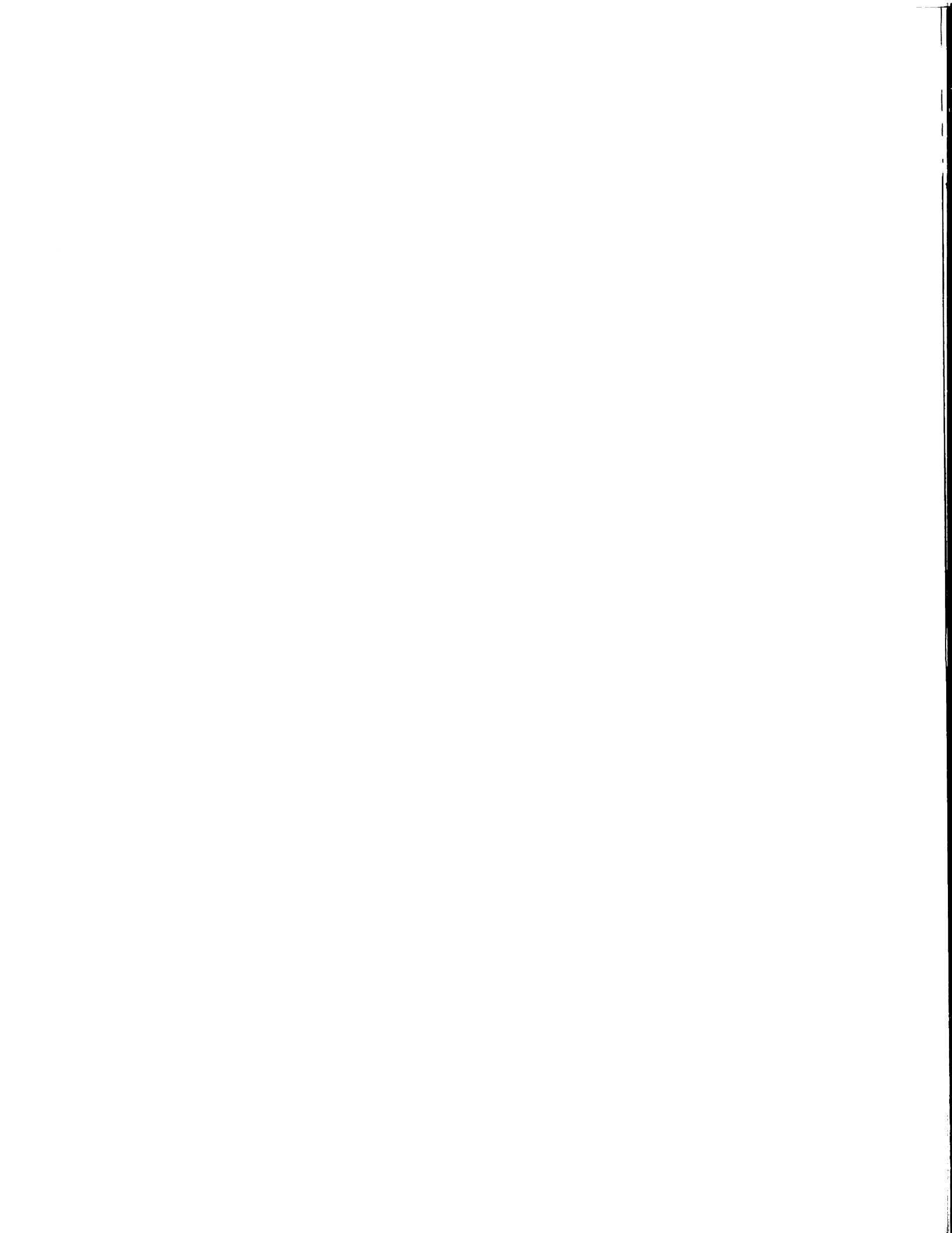
The second method of dealing with the remaining reflections is to examine the reconstructed light rays to determine the shortest distance between them. Because the direction of the reconstructed ray can be found for each camera individually, each ray can be analytically reconstructed. Figures 30 and 31 show a simplified technique for determining the three-dimensional position of an LED based on the average of the global Y coordinate to resolve the ray skewness. An elimination can be performed when the perpendicular distance between the skew rays is larger than a user selectable value. Dan Ottenheimer (1982) [118a] implemented such a technique which is used in the TRACK processing. It is slightly different from the procedure shown in Figure 31. Applying this routine will eliminate LEDs whose rays are "too skew" from the subsequent calculations.

A scaled parameter to be used as an upper limit on "skewness" was desired that would be independent of the distance an LED is from the cameras and independent of the camera positions and angles. To enable this the skewness is represented in Selspot discretization units by determining how far the measured light image would have to move on the detector plate to eliminate the skew error altogether. If the number of units is greater than the upper limit parameter selected by the experimenter, the data are eliminated.

As mentioned above, the LEDs are grouped in rigid arrays. This is done to calculate the position and orientation of an imbedded body-coordinate system fixed with respect to the LED array. The technique used involves a least squares, best fit of orientation based on the locations of three or more known points on a rigid body. The data produced are: the vector from the origin of the Global Coordinate System (GCS) to the origin of the BCS at this time step and the nine element, three by three rotation matrix representing the change in orientation from an initial reference. See Conati (1977) [34] for a detailed explanation of the Schut algorithm that performs this calculation. In addition to providing the benefit of full kinematic discription for each segment, the BCS technique enables a change of reference frame from the inertial-global frame to one of the body-coordinate systems. Thus relative kinematic data are available by simple matrix transpositions and multiplications. A great deal of the data presented later in this thesis is motion of one segment relative to another and is produced using these operations.

As a final test on the quality of the optical data, the spatial integrity of each array of LEDs is examined every frame. If the measured length between any two LEDs is longer or shorter than that originaliy specified in the vector definition of the array, one of the LEDs is eliminated from the calculations. The LED eliminated is the

one with the most inter-LED length errors between itself and every other LED in the segment. This continues until all remaining data passes the inter-LED length test. If the array integrity examination routine reduces a segment to fewer than three LEDs in any frame, that frame of data for that segment is not used in the subsequent calculations.

Derivatives of the kinematic data are calculated using a five-point Lagrangian technique for equally spaced abscissas. A thorough discussion of this method can be found in Antonsson (1976) [7].

The transduction and accumulation hardware with the software package to perform data reduction and storage is called TRACK, an acryonym for Telemetered Real-time ACquisition of Kinematics. A major portion of the effort associated with this thesis was devoted to designing and implementing the TRACK hardware and software package. The overall design goals of the system are listed in Table 3, and Table 4 is a flowchart of the acquisition and offline processing, included to provide a general overview of TRACK's operation. The TRACK III software package is discussed in some detail in Appendix H.

1. Complete automaticity

2. Full 3-D positions and rotations of imbedded body
    coordinate systems

3. 1-mm resolution and accuracy in position and 1 degree
    resolution and accuracy in rotations in a 1.5 meter cube

4. High sampling rate (315 Hz)

5. Large number of automatically monitored markers (30
    LEDs, 10 body segments maximum, fewer with more than 3
    LEDs per segment)

6. Synchronous forceplate data acquisition

7. High speed (real time) data reduction

8. Capable of expansion to a large viewing volume (moving
    cameras)

Table 3 TRACK system design goals

Adjust experimental parameters

Collect Selspot and Forceplate data
and write to disks in real time

Interrogate Selspot data for LEDs out
of view and translate to canonical form

Store RAW data under a user selectable file name

Correct Selspot data for camera nonlinearity errors

Low-pass filter the RAW data

Reduce Forceplate data to 3 Forces and 3 Moments

Low-pass filter the Forceplate data

Calculate the 3-D locations of LEDs

Low-pass filter the Selspot 3-D data

Store the individual LED 3-D data if desired

Eliminate Selspot bad data by
checking LED rigid array integrity

Calculate Body Coordinate system position and rotation

Calculate translational and rotational
velocities and accelerations

Low-pass filter the derivatives

Store the final result data if desired

Plot

Table 4   TRACK Version III Offline Flow Chart

VERIFICATION OF THE CALIBRATION IN THREE DIMENSIONS

A pendulum was used in a set of experiments (to be discussed later) having to do with locating axes of rotation; however, some of the kinematic results are presented here to demonstrate that the calibration described above met the 1-mm accuracy design goal.

The tests involved a 0.5-meter-long, 1 degree-of-freedom pendulum instrumented with six LEDs, hanging in the viewing volume of both Selspot cameras. The pendulum was offset from its equilibrium, allowed to oscillate, and its kinematics were accumulated by TRACK. The body-coordinate system associated with the LEDs was located 340 mm from the axis of rotation of the pendulum, measured independently. If the calibration achieved its goal, the TRACK measured path of the origin of the coordinate system mounted on the pendulum would not deviate from a circle of radius 340, by more than 1 mm. The following figures show data from the pendulum in three different locations in the viewing volume, and with axes of rotation in different directions. Figure 31a shows the same pendulum with 2 degrees of freedom. The rotation axis parallel to the body coordinate system X axis was held fixed for the experiments reported here.
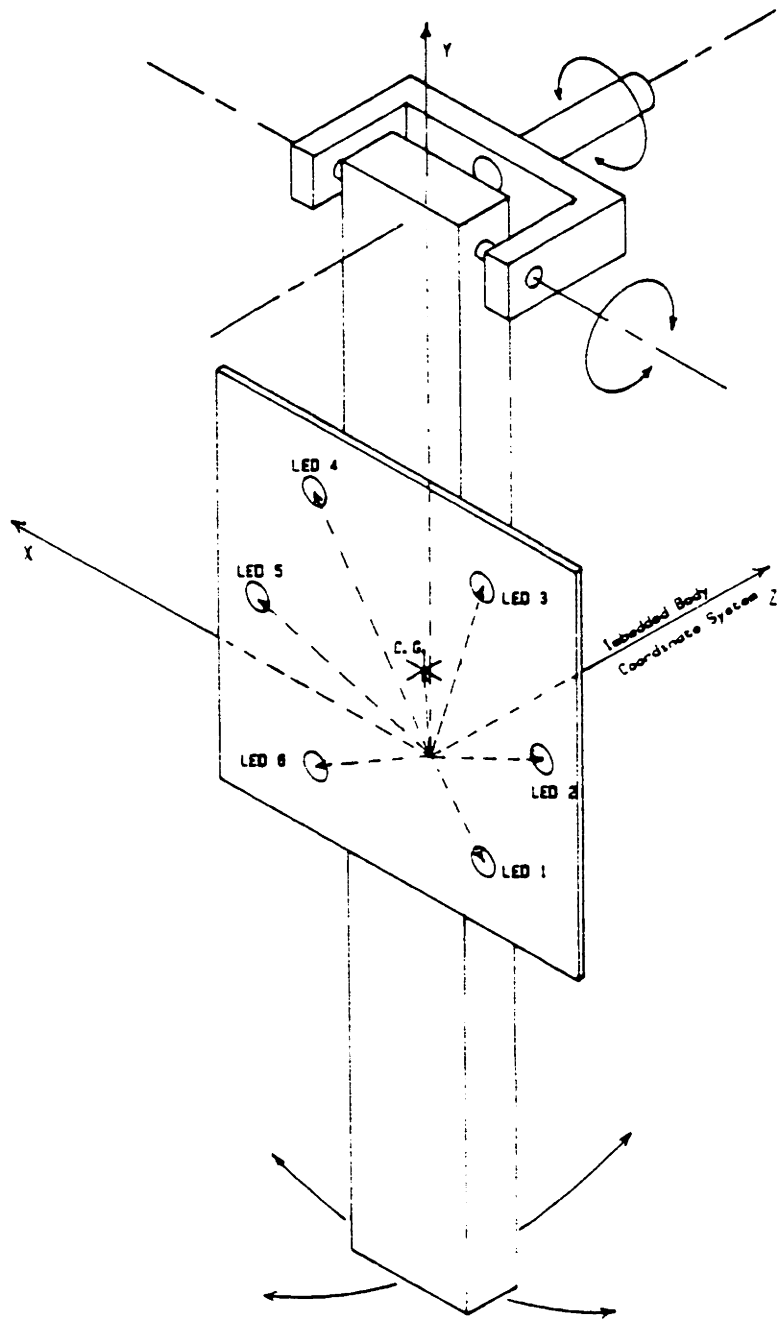
Figure 31a. The Pendulum

(0.0198, 0.7760, -0.0344)
R= 0.3390 [meters]
URV=-0.0057, -0.0109, 0.9999
Frames: 1 to 630
TK4: AP1503.DAT:0
Seg# 1 wrt Seg# 0.   ~X/Y Plane
AP1501w/10HzFltr,DerivsFltrd

Figure 32. Circular pendular data
viewed normal to the best fit plane.
Rotation axis roughly parallel to
the global Z axis

Circle

Data

10 mm

(0.0198, 0.7760, -0.0344)
R= 0.3390 [meters]
URV=-0.0057, -0.0109, 0.9999
Frames: 1 to 630
TK4: AP1503.DAT;0
Seg# 1 wrt Seg# 0.   ~Z/Y Plane
AP1501w/10HzFltr.DerivsFltrd

Axis

Plane

Sphere

10 mm
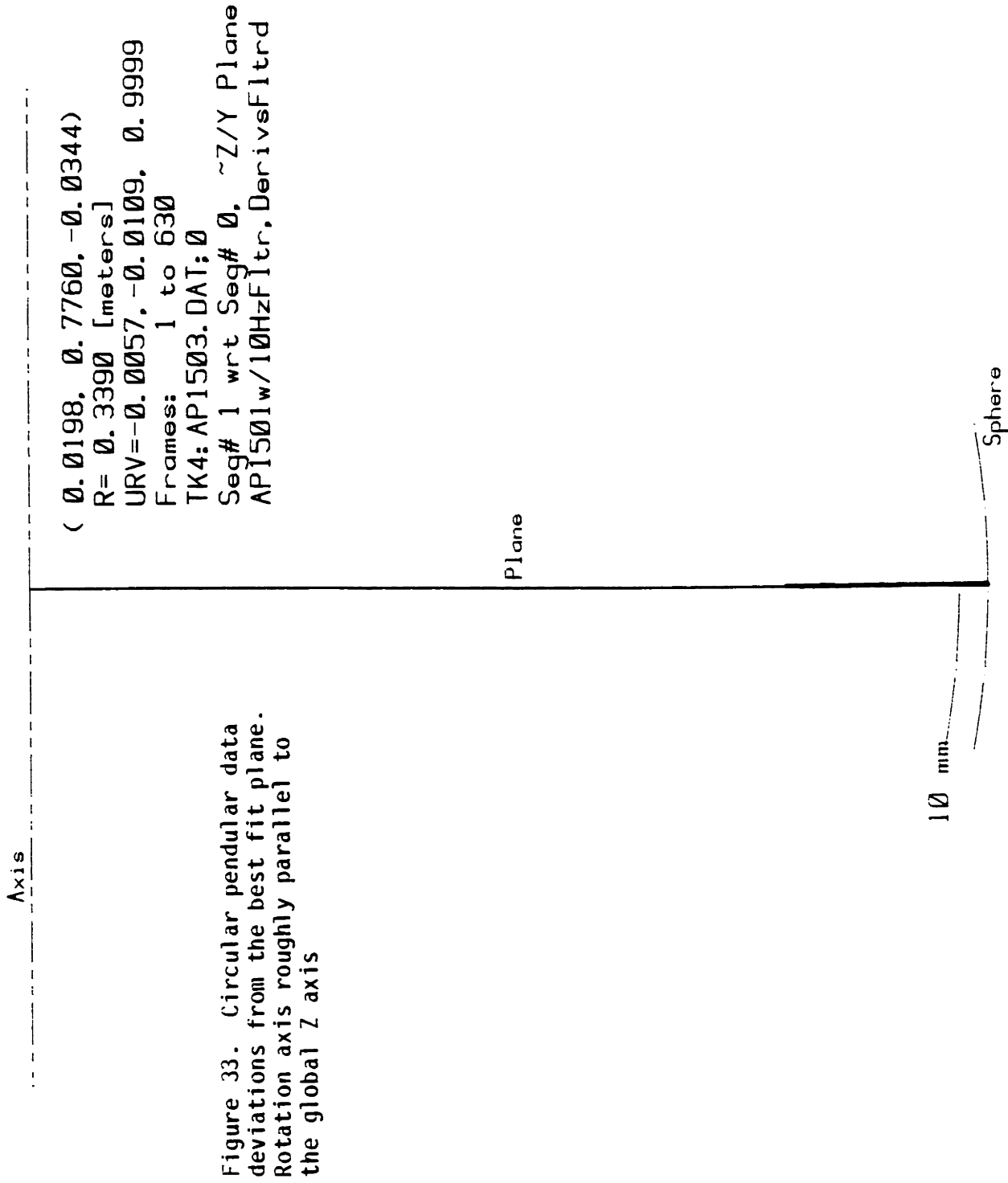
Figure 33. Circular pendular data
deviations from the best fit plane.
Rotation axis roughly parallel to
the global Z axis

(-0.1817, 0.7739, 0.0388)
R= 0.3382 [meters]
URV=-0.4129, 0.0101, -0.9107
Frames: 1 to 630
TK4: AP1506.DAT: 0
Seg# 1 wrt Seg# 0.   ~X/Y Plane
AP1505 w/10Hz Filter

Figure 34. Circular pendular data viewed normal to the best fit plane. The rotation axis is rotated about 30 degrees toward the +X axis

10 mm
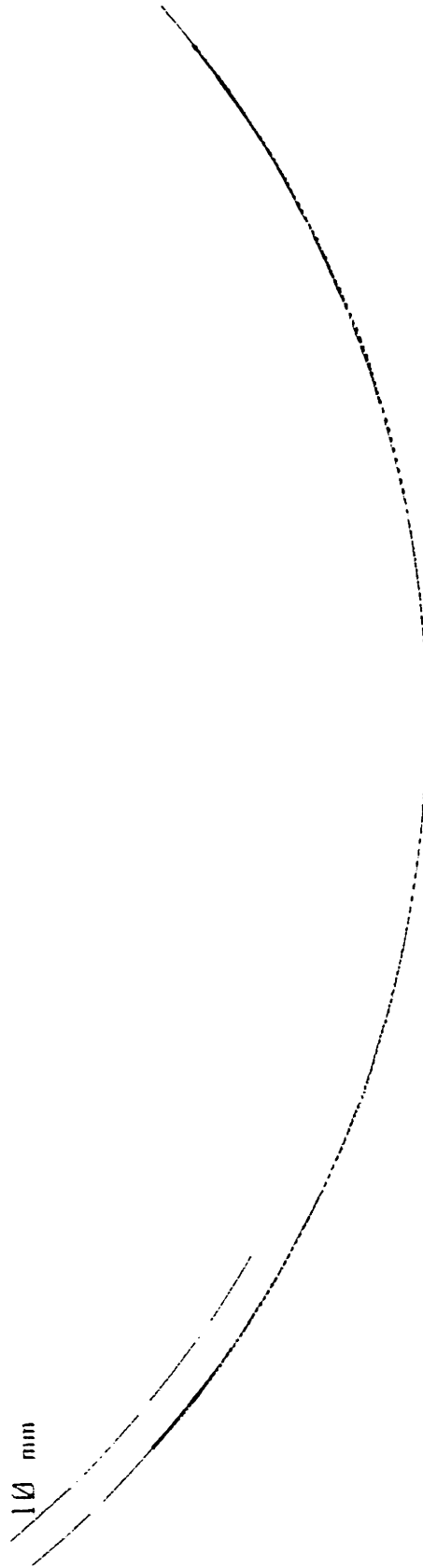
(-0.1817, 0.7739, 0.0388)
R= 0.3382 [meters]
URV=-0.4129, 0.0101, -0.9107
Frames: 1 to 630
TK4: AP1506.DAT;0
Seg# 1 wrt Seg# 0. ~Z/Y Plane
AP1505 w/10Hz Filter

10 mm

Figure 35. Circular pendular data
deviations from the best fit plane.
The rotation axis is rotated about
30 degrees toward the +X axis

(0.3679, 0.7825, 0.1845)
R= 0.3418 [meters]
URV= 0.5773, 0.0210, -0.8163
Frames: 1 to 630
TK4: AP1516.DAT: 0
Seg# 1 wrt Seg# 0.   ~X/Y Plane
AP1515w/10HzFltr (~out of view)

Figure 36. Circular pendular data
viewed normal to the best fit plane.
The rotation axis is rotated about
30 degrees toward the -X axis

10 mm

(0.3679, 0.7825, 0.1845)
R= 0.3418 [meters]
URV= 0.5773, 0.0210, -0.8163
Frames: 1 to 630
TK4: AP1516.DAT; 0
Seg# 1 wrt Seg# 0, ~Z/Y Plane
AP1515w/10HzFltr(~out of view)

10 mm

Figure 37. Circular pendular data deviations from the best fit plane. The rotation axis is rotated about 30 degrees toward the -X axis

The dot-dashed lines are best fit circle with radius (labeled "R= ") shown in each plot. The TRACK measured data points are the dots. The second short circular arc displaced from the one through the data has a 10-mm shorter radius, to provide some scale to the plots. In the view along the best fit plane (not normal to it), the double-dashed line represents an axis passing through the best fit circle, perpendicular to the best fit plane, and the solid line represents the best fit plane. The first trio of numbers in the legend are the global X, Y, Z of the center of the circle, R is the radius, and UVR stands for Unit Rotation Vector. This is a vector in the GCS perpendicular to the best fit plane of motion. Each of the three experiments depicted in Figures 32 through 37 has a different orientation, hence a different Unit Rotation Vector. Segment Number 1 is the pendulum BCS, and Segment Number 0 is ground. Note that the measured points do not deviate from circles by more than 1 mm and that the radii are also correct to within 1 mm. The second and third pair of plots show the pendulum swinging in a plane not close to parallel to one of the global coordinate axes, to demonstrate the applicability of the calibration to truely three-dimensional motion.

COORDINATION OF THE SELSPOT AND FORCEPLATE DATA

To enable the use of the forceplate data in conjunction with the kinematic data provided by TRACK, an alignment of the Selspot (SS) and forceplate (FP) global coordinate systems, as well as establishing timing synchrony, was necessary. The latter has been verified by determining the computer sampling timing with a high-speed storage oscilloscope. The former required a more cumbersome procedure.

The forceplate produces six pieces of information: three orthogonal forces, a moment about a vertical axis, and the two coordinates on the surface of the plate of the net center of applied force. It was these last two that were used to align the SS and FP GCSs. Essentially, a TRACK-instrumented weight was placed on the FP. The BCS of the LED array was coincident with the center of mass of the weight. The measured vector from the origin of the SS GCS to the BCS origin on the weight minus the FP center of force coordinates was used to locate the origin of the FP GCS with respect to the SS GCS. Several positions of the weight were used to verify the results and to determine that the global coordinate systems were parallel. Once the vector from the SS to the FP GCS was determined, it was used to translate the TRACK global coordinate system to be coincident with the forceplate global coordinate system. This measurement is

obviously critically related to the position of the cameras on the optical bench used for mounting as well as the position of the bench itself. The global coordinate system offset, as well as the camera positions and angles used throughout, is shown with the experimental parameters in Chapter 4. Figure 28 depicts the undisplaced location of the Selspot and forceplate global coordinate systems.

DYNAMIC RESPONSE: SIGNAL AND NOISE FREQUENCY CONTENT

At this stage in the discusson both transduction systems (forceplatform and Selspot) have been interfaced to the computer, and an extensive calibration of the position measurement system has been performed. The next determination to be made is the frequency content of the signals to be measured (position and force histories) and the frequency signature of the noise. I began with the following assumption: the foot is the most distal of the segments of interest (foot, shank, thigh, and pelvis), has the lowest mass, and is subject to the highest accelerations (heel strike during gait). Thus a signature of amplitude as a function of frequency for the foot will be an upper bound for the frequency content of gait as a whole.

In order to determine this upper bound, data from an
instrument with a very large signal-to-noise ratio are
mandatory. The piezo-electric, multi-component
forceplatform system (FP) represented such a source. The
platform is capable of measuring the foot-floor interaction
forces with a resolution of one part in 4096 (12 bits), and
the noise amplitude is less than +/- 1 bit, yielding a
signal-to-noise ratio of over 2000. Note also that the
largest accelerations occur (hence the highest frequencies
of motion) at heel strike, a phenomena capturable by the FP,
another reason for using the forceplate signature. Figure
38 is a typical vertical force history.

Typical Forceplate Data
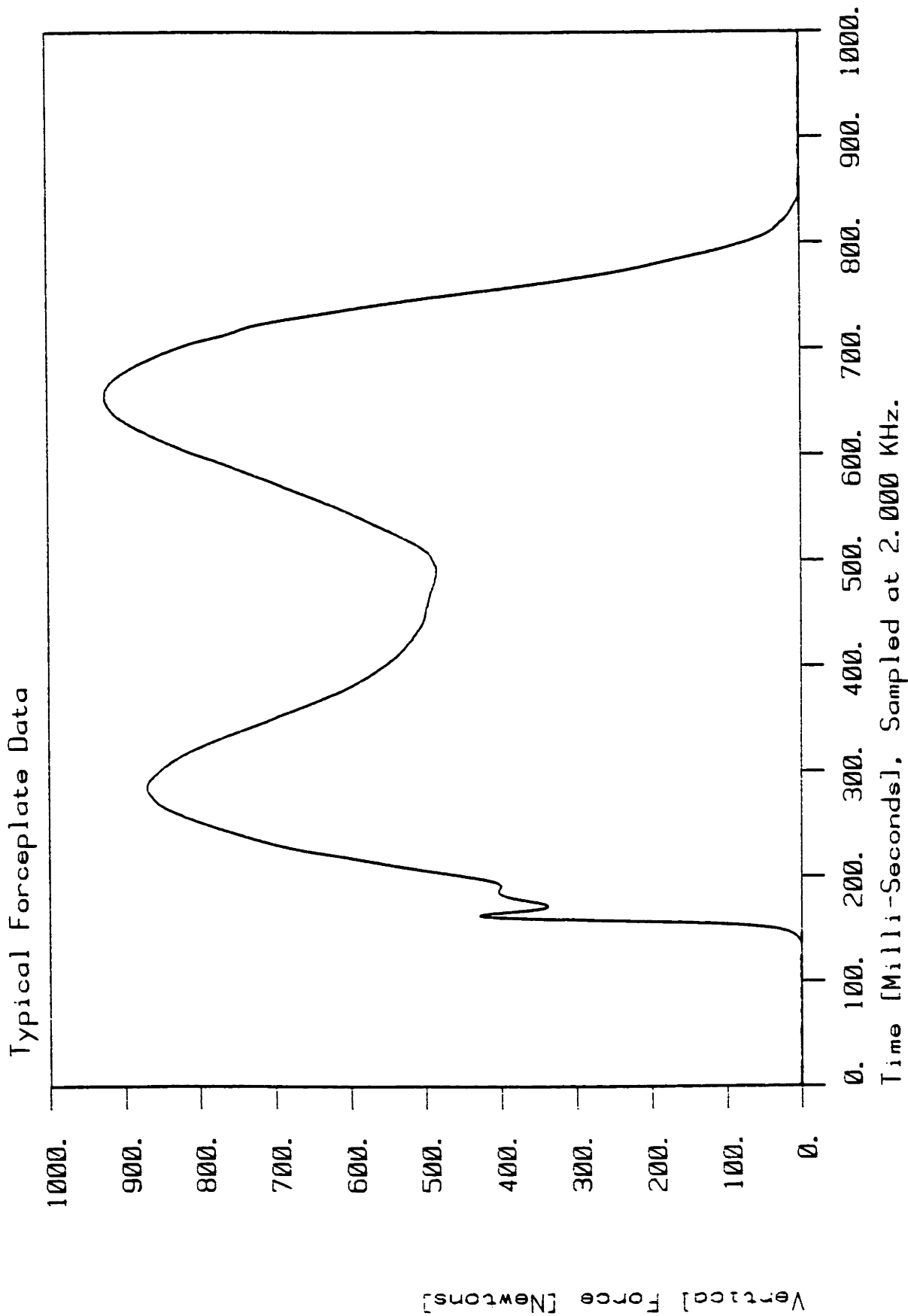
Figure 38. Typical vertical foot-floor force during gait

The vertical force was chosen as indicative of the frequency envelope because it has a larger signal to noise ratio than the other two orthogonal forces.

The following is the frequency content of that signal, produced by a digital Fast Fourier Transform (FFT) routine. The plotted results are amplitude as a function of frequency.   The data were sampled at 2.0 kHz.   The FFT produces amplitude as a function of frequency from 0.0 Hz to 1.0 kHz.   Only the lowest 100 Hz are plotted here because everything above that is essentially zero.   The function is monotonically decreasing from 100 to 1000 Hz.
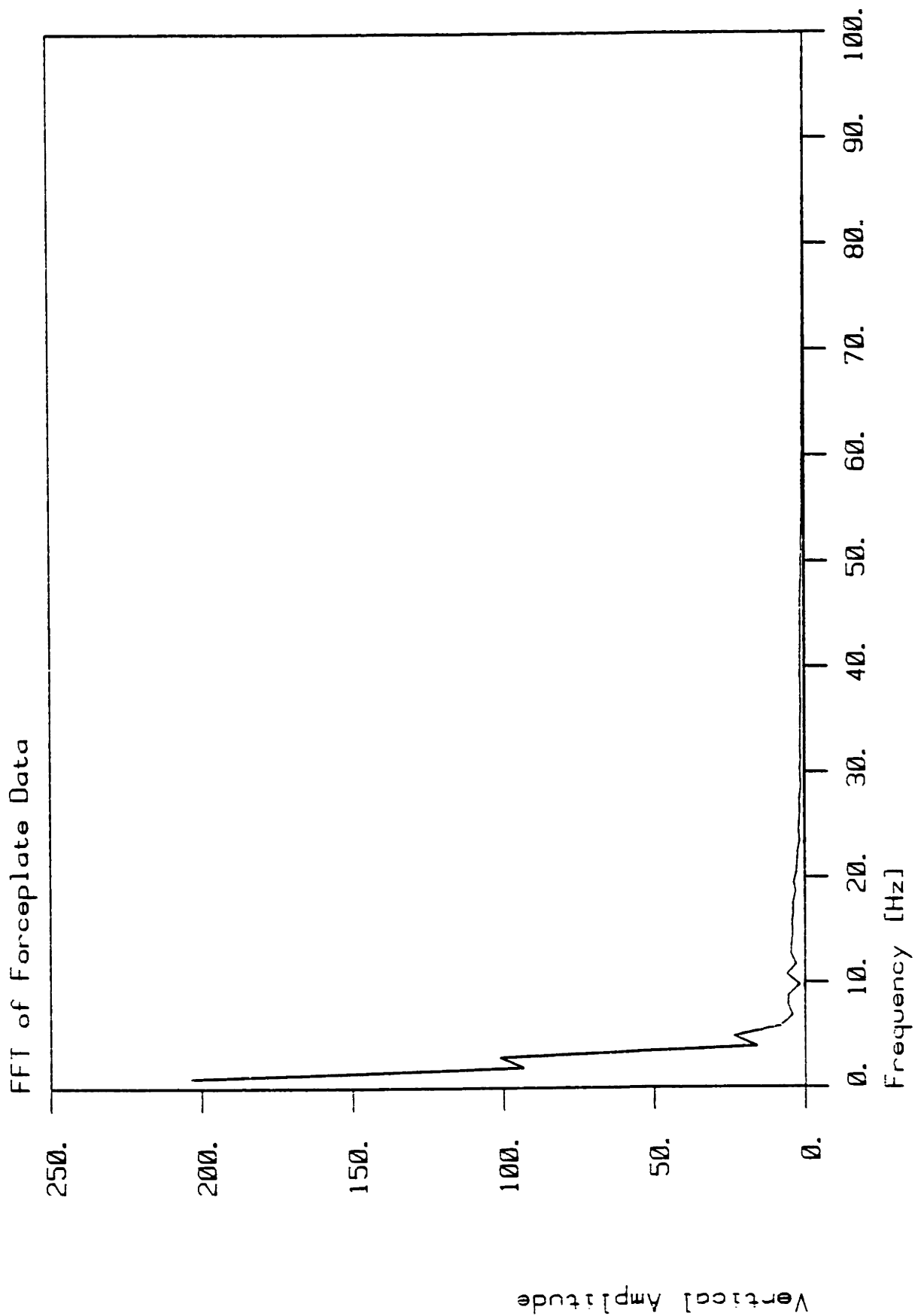
Figure 39. FFT of typical vertical foot-floor force

The forces during stance were measured for thirty trials of twelve subjects (two different steps were measured for most subjects; a few had three measured), and the upper envelope is presented here plotted in percent. Once again, the forceplate was sampled at 2.00 kHz throughout these experiments. Each individual force record was normalized to the maximum force during that step before the FFT was taken and aggregated after Figure 41 shows the integral of power in the composite vertical force envelope plotted in percent. Ninety-eight percent of the power is contained below 10 Hz, and 99 percent below 15 Hz.
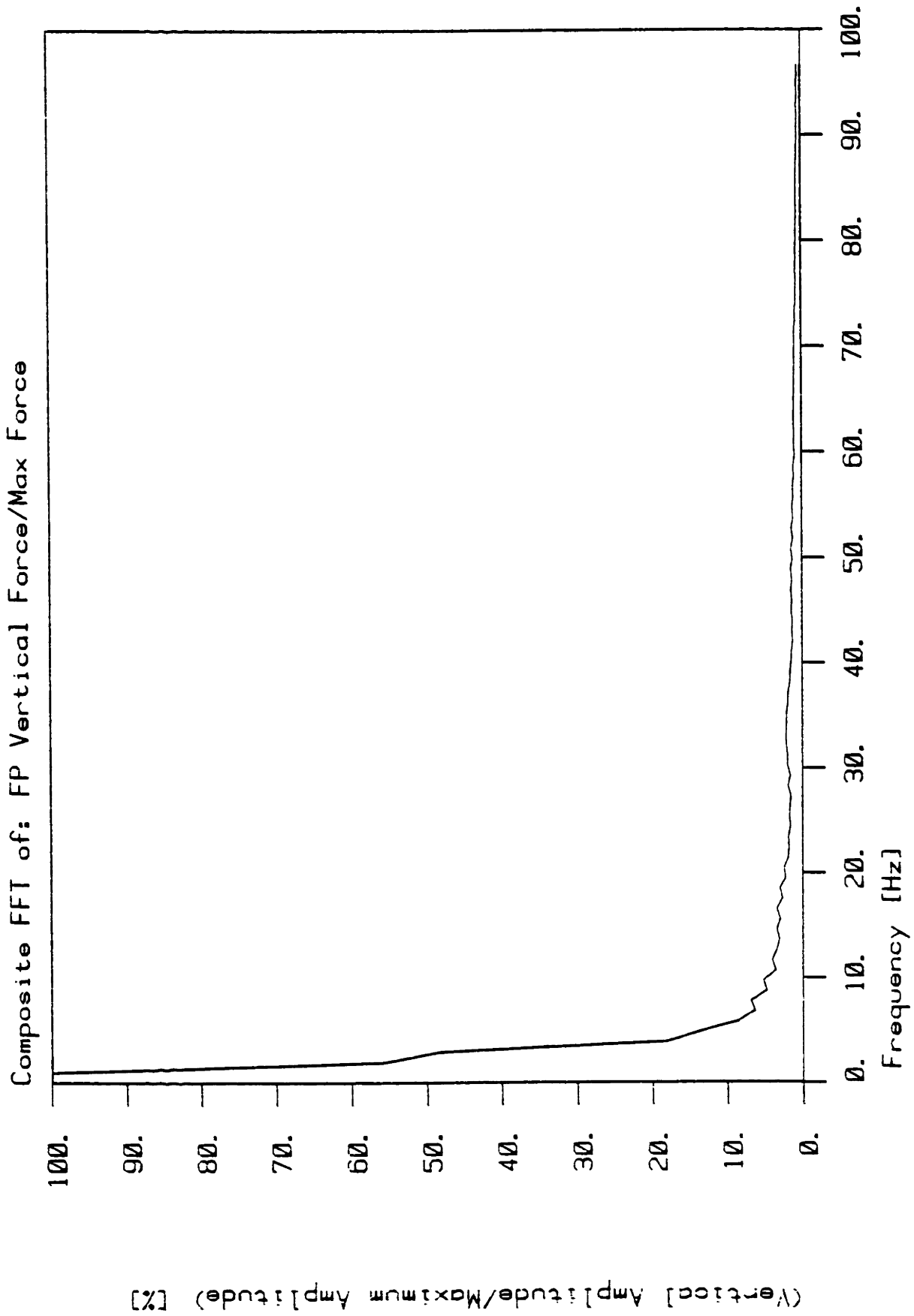
Figure 40. Extremum envelope of foot force signatures
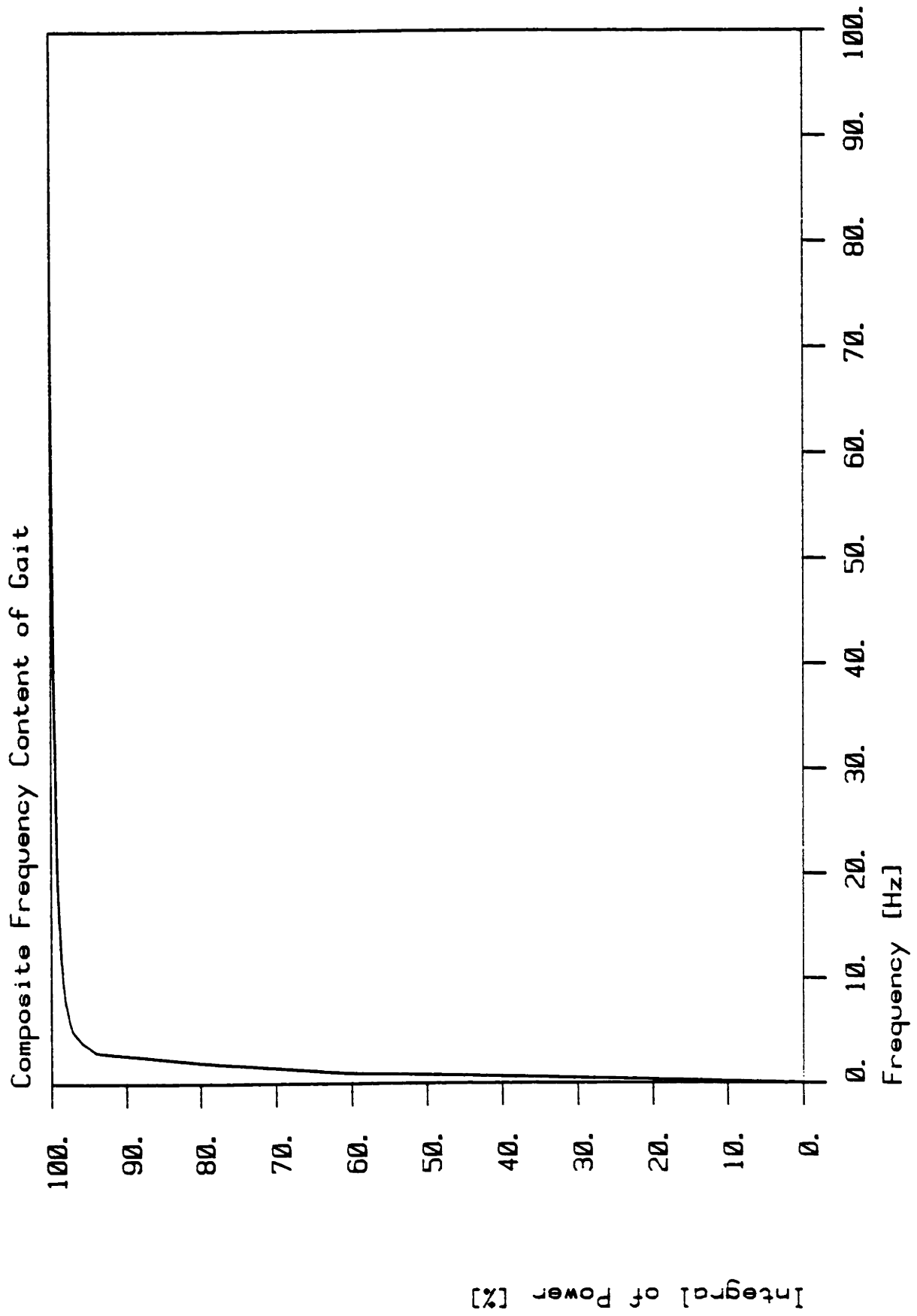
Figure 41. Power in the extremum envelope of foot force signatures

To be sure that these frequency domain signatures were not corrupted by resonances, the impulse response of the FP was measured. A 30-mm diameter, stainless steel ball was dropped 200 mm vertically onto the center of the forceplatform's surface. The analog output from the transducers was digitally sampled at 2.5 kHz. The manufacturer of the FP claims that the fundamental frequency of resonance for the plate is above 1.0 kHz; however, these plots suggest that 700 Hz is a closer estimate for our installation.
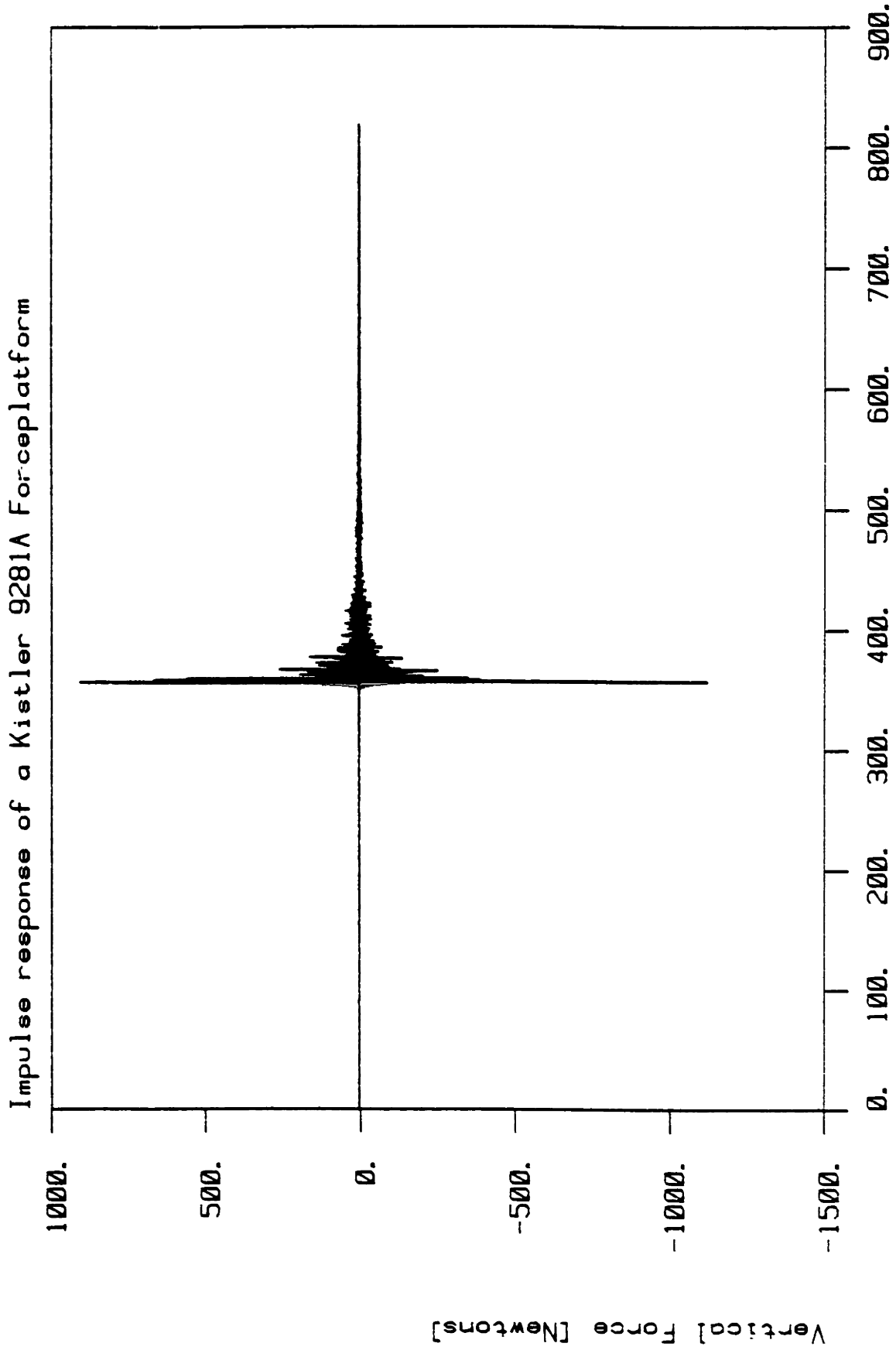
Figure 42. Forceplatform impulse time response
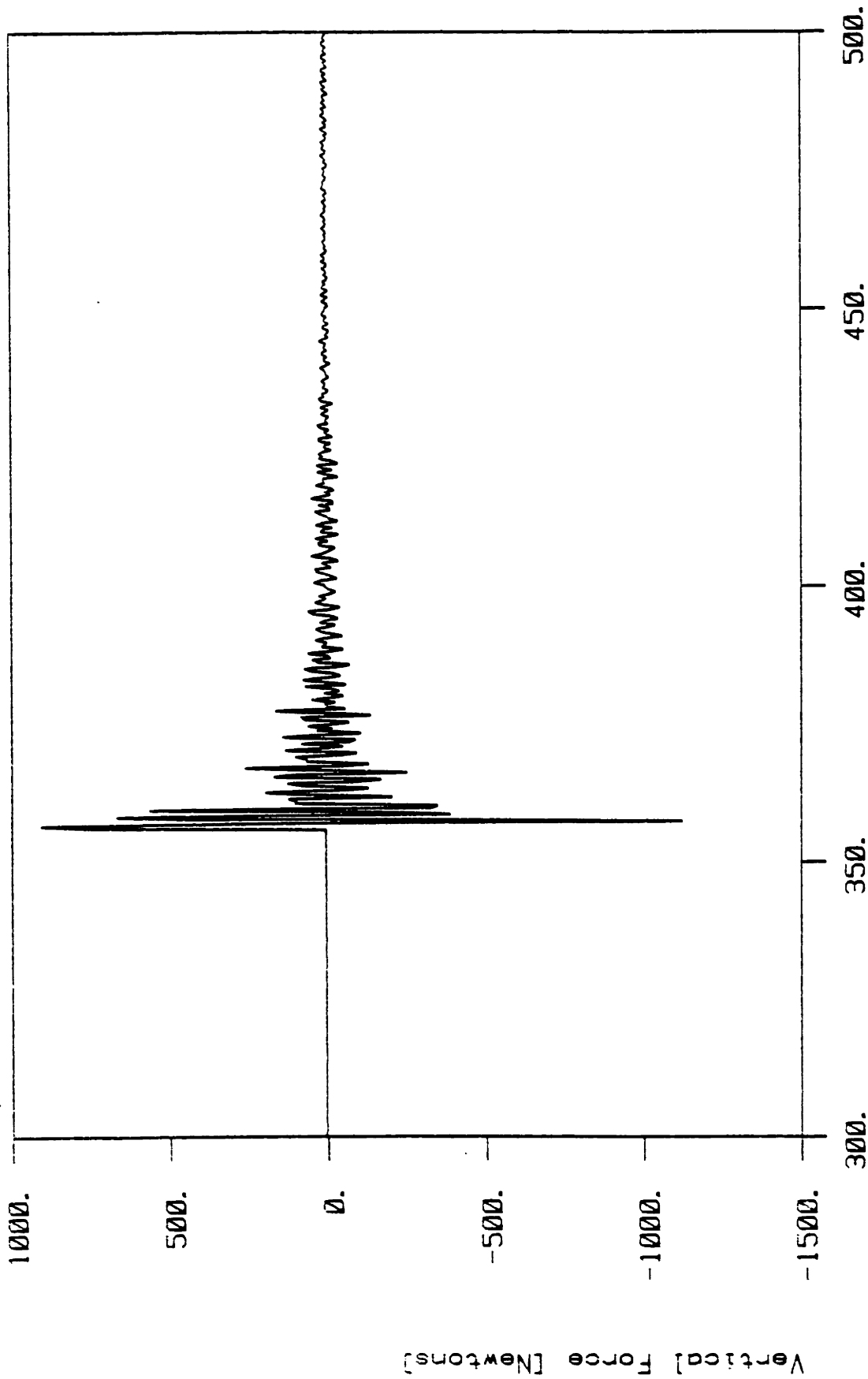
Figure 43. Expanded time scale forceplatform impulse time response
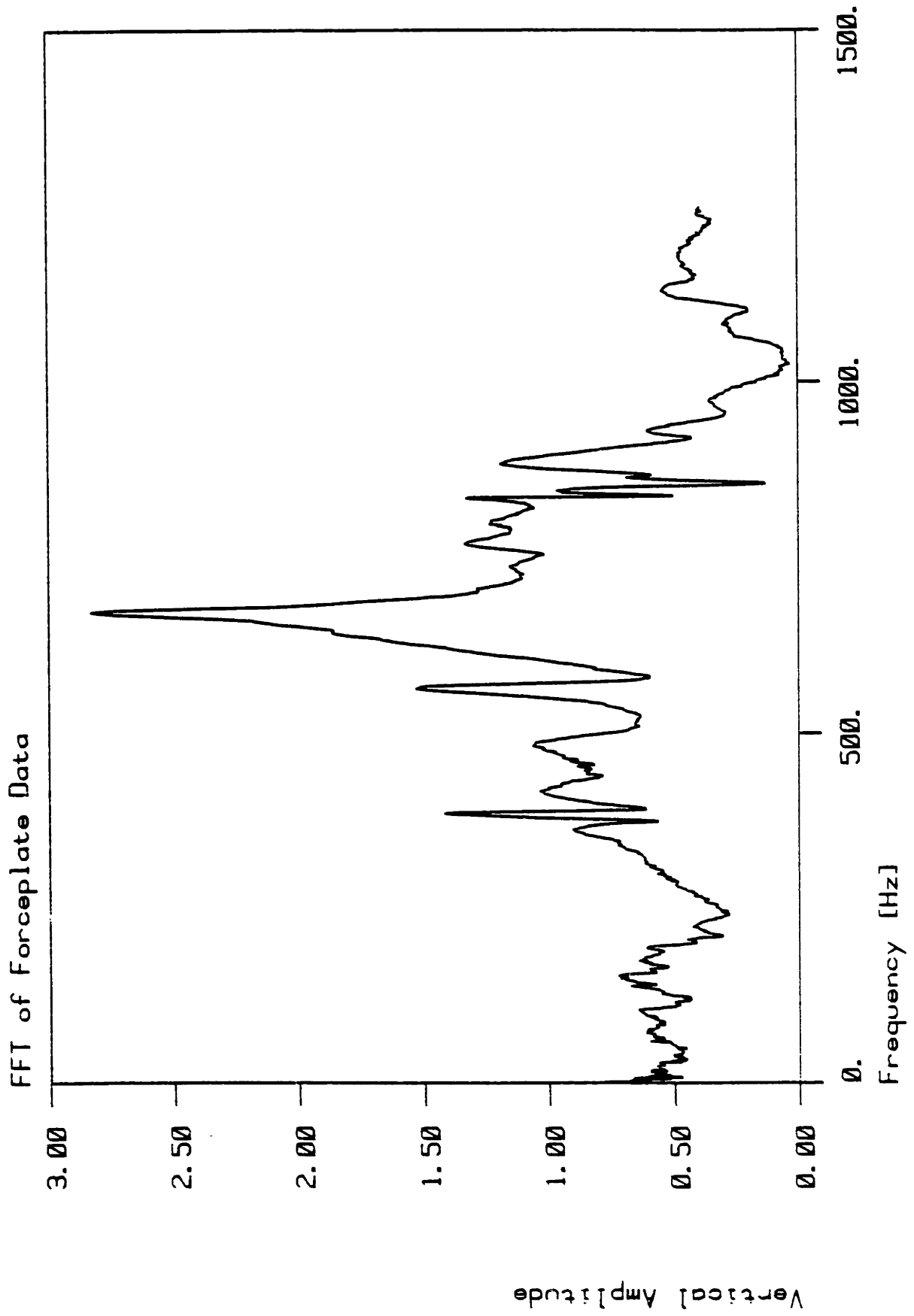
Figure 44.   Impulse frequency response

To be sure that the signals were sampled and analyzed correctly, a 30-Hz sine-wave from an analog voltage signal-generator was applied to the inputs of the analog to digital converter sub-system, instead of the forceplate charge amplifier output voltages. Data were acquired at 2.0 kHz. Figures 45 and 46 are plots of the data as accumulated. Then an FFT was applied to the 30-Hz sine-wave data set. Figures 47 and 48 show the frequency domain results. This simply verifies that a signal applied to the A-to-D converter is sampled correctly and that the FFT routine is producing correct results.
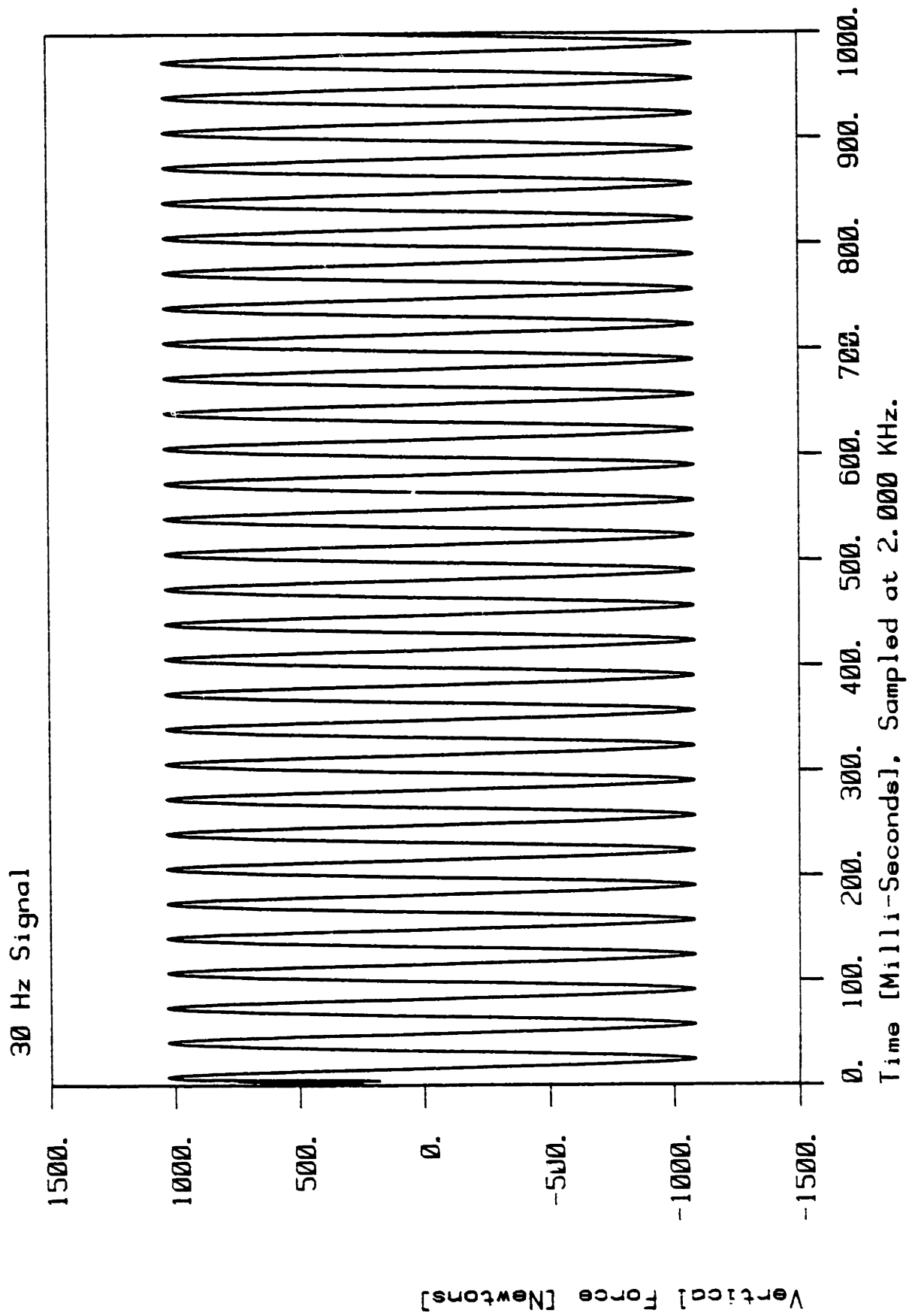
Figure 45.   30 Hz sine-wave from a signal generator

30 Hz Forceplatform Signal

Vertical Force [Newtons]
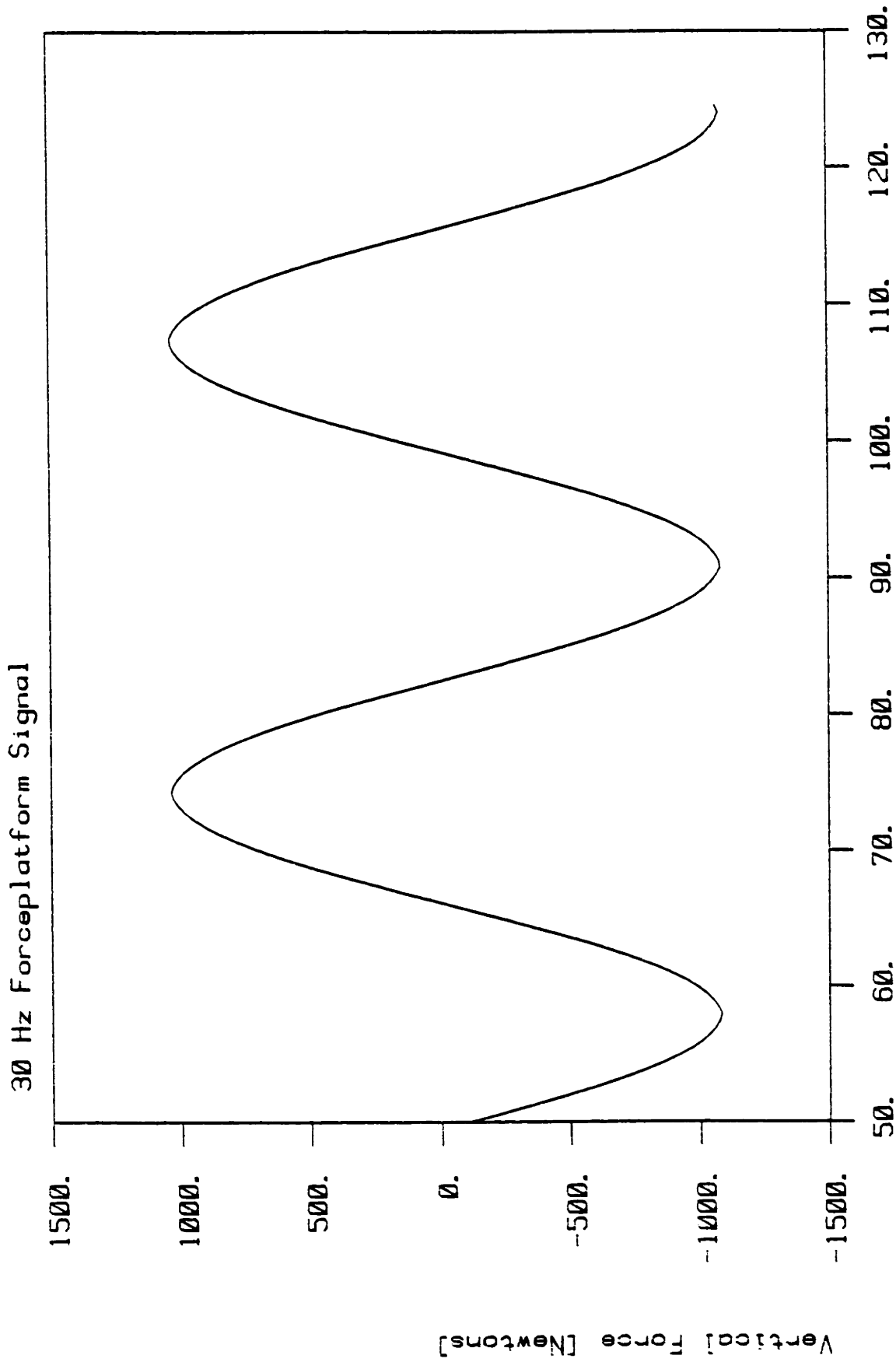
Time [Milli-Seconds]. Sampled at 2.000 KHz.

Figure 46. Expanded time scale 30 Hz sine-wave
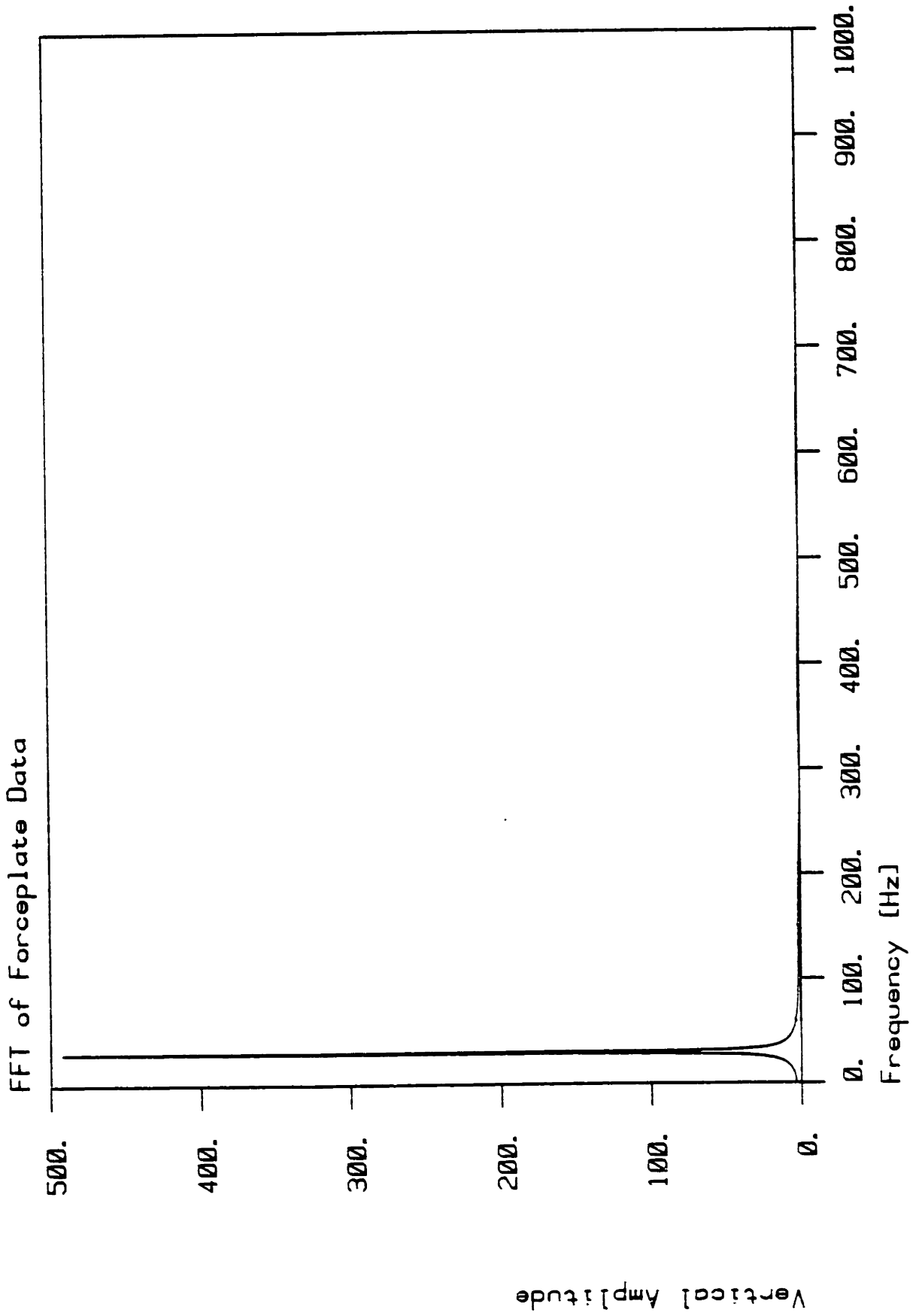
Figure 47. FFT of sampled sine-wave
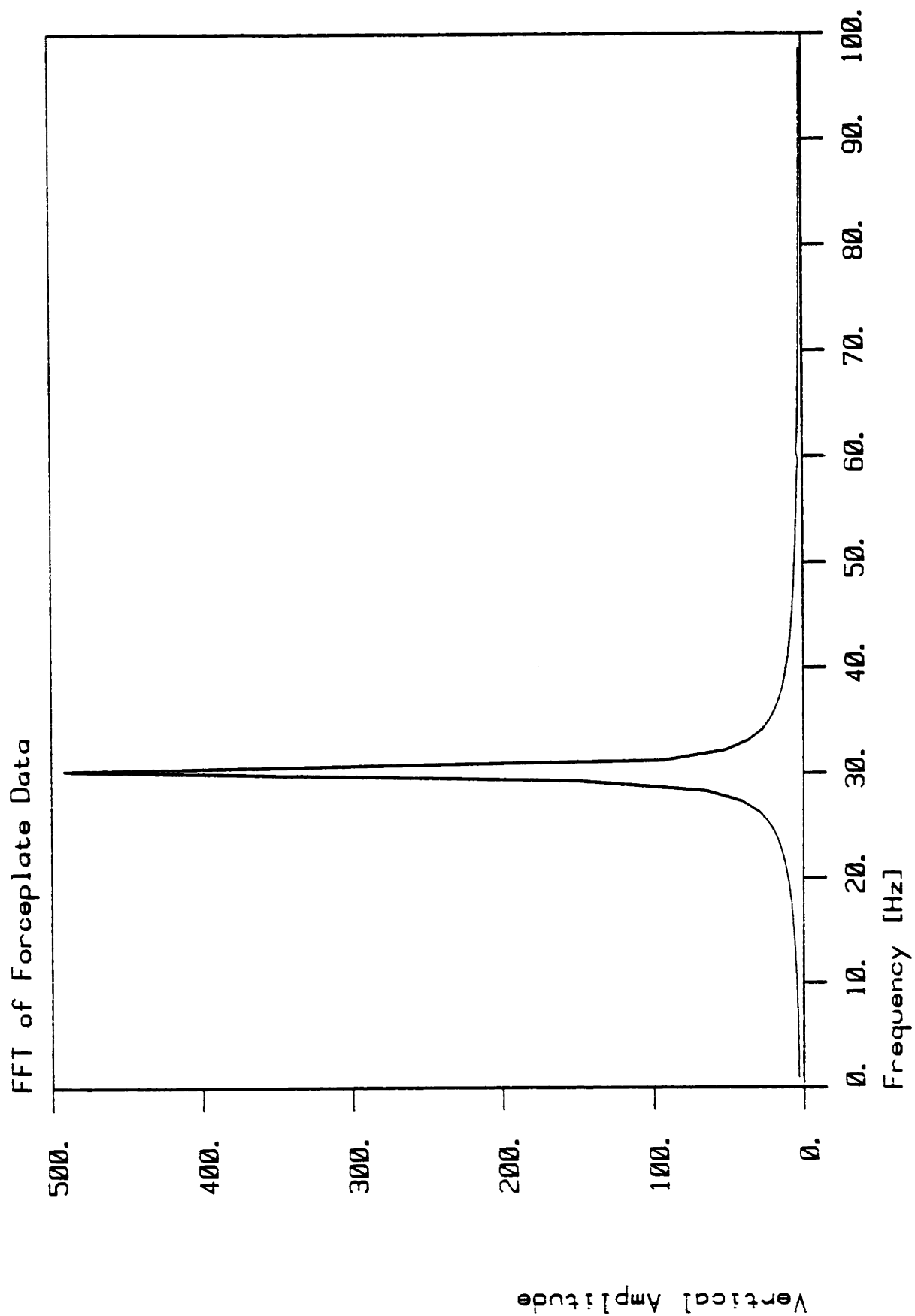
Figure 48. Expanded frequency scale FFT of sampled sine-wave

The FP produces force measures lower than 100 Hz with a signal-to-noise ratio of greater than 2000, thus producing an upper bound frequency envelope for foot-floor interaction forces that is accurate. The question remains, however, what is the highest significant frequency component of normal gait? Significant is the key word, and the accumulation of the above data permits us to find a quantitatively based answer. A relationship (contained in the "tail" of the composite forceplate frequency bound) now exists between the highest frequency included in the analysis and the required accuracy of the end result dynamic estimates.
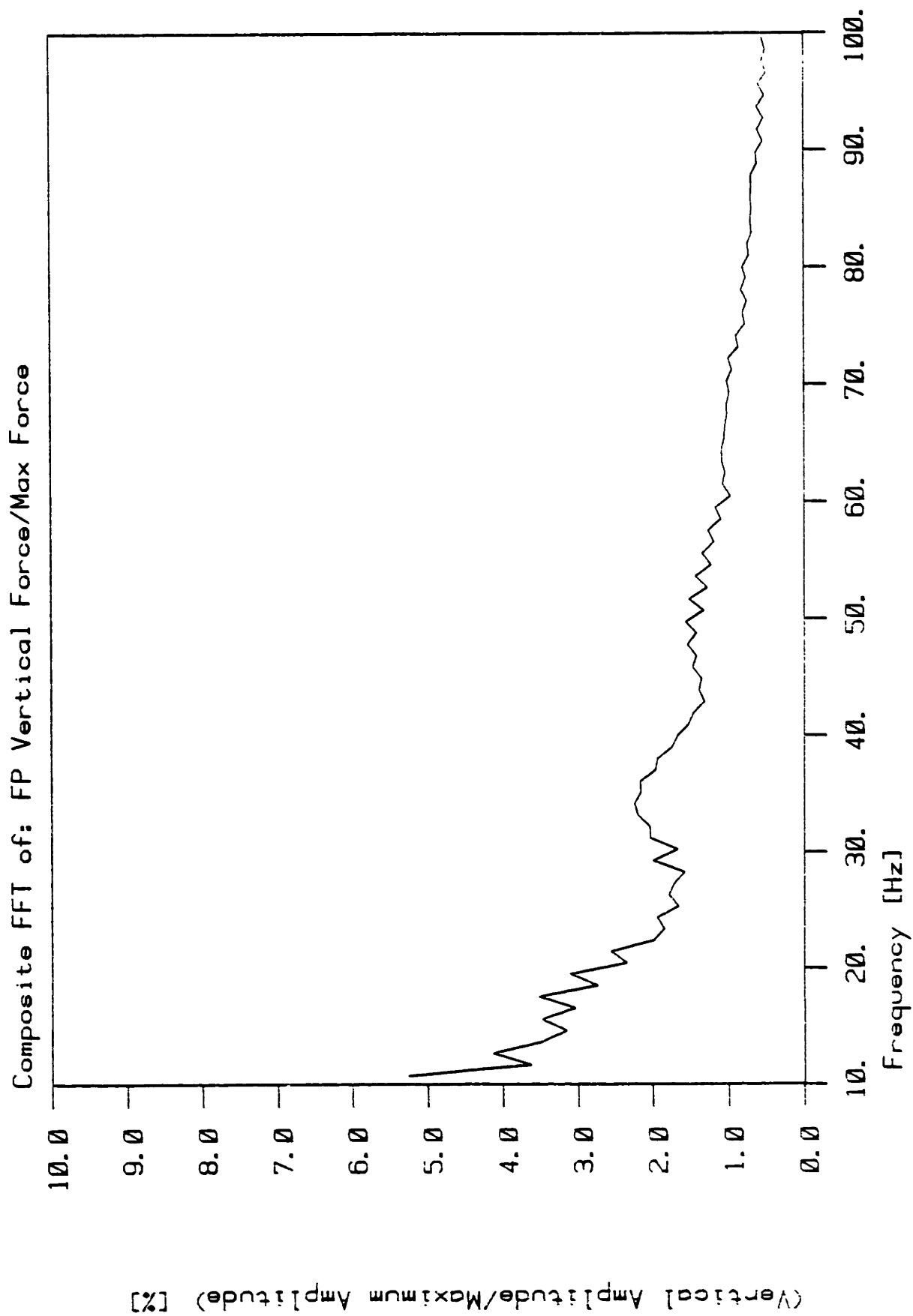
Figure 49. Expanded scale plot of forceplate amplitude as function of frequency for 10 to 100 Hz

If 5 percent accuracy is desired, at least everything below 10 Hz must be included; if 2 percent accuracy is desired, 20 Hz is needed. If forces accurate to 1 percent are needed, nothing below 50 Hz can be left out. This study includes all frequency components of gait up to 15 Hz in an effort to estimate net forces and moments at joints to within 5 percent.

To make these estimates, the <u>kinematic</u> measurement system used must be able to produce accurate measures up to the desired frequency. In this case positional fidelity must be preserved up to 15 Hz, which for a sampled system requires a <u>minimum</u> of 30 Hz sampling rate. At 30 Hz, however, the 15-Hz component would only have two data points per cycle and would be a crude estimate. To properly eliminate the effects of noise without "folding" in the frequency domain, the sampling rate must be above twice the highest significant noise frequency. A 300-Hz sampling rate will allow twenty points per cycle of the highest frequency component of the signal to be measured and allow proper noise rejection up to 150 Hz.

> "To be able to describe (a function) f(t) exactly, it is necessary to sample f(t) at a rate greater than twice its highest frequency."
> Stearns (1975) [150] page 37

An excellent treatment of discrete time sampled systems can be found in:

Stearns, S. D., Digital Signal Analysis. Rochelle Park, New Jersey: Hayden Book Company, Inc., 1975. [150]

Pages 37 through 43 particularly cover "folding" in the frequency domain caused by sampling a signal below twice its highest frequency.

To warrant confidence that the filtering selected above is producing the desired results, a measure of the noise content in the signal is as important as the signal measurement. In this case, the most significant noise in the displacement trajectories is sampling noise generated by the Selspot. To quantitatively determine the frequency domain characteristics of this noise, the following test was performed. A set of LEDs, similar to the sets associated with an imbedded body-coordinate system (BCS) of a single link during gait, was held stationary in the field of view of the cameras. The BCS approach is used throughout this thesis as the method of acquiring and describing the complete 6-degree-of-freedom kinematics of each body segment of interest. As mentioned in Chapter 2, this provides a unique capability of not only concisely and completely describing the kinematics of motion, but also of allowing subsequent analysis to locate joint axes and determine the magnitude of soft tissue motion.

To compare these frequency domain results with actual experimentally generated motion data, an array of six LEDs was used here. The stationary "position trajectory" of this segment was recorded. Figure 50 is the measured vertical, unfiltered displacement of the segment as a function of time.
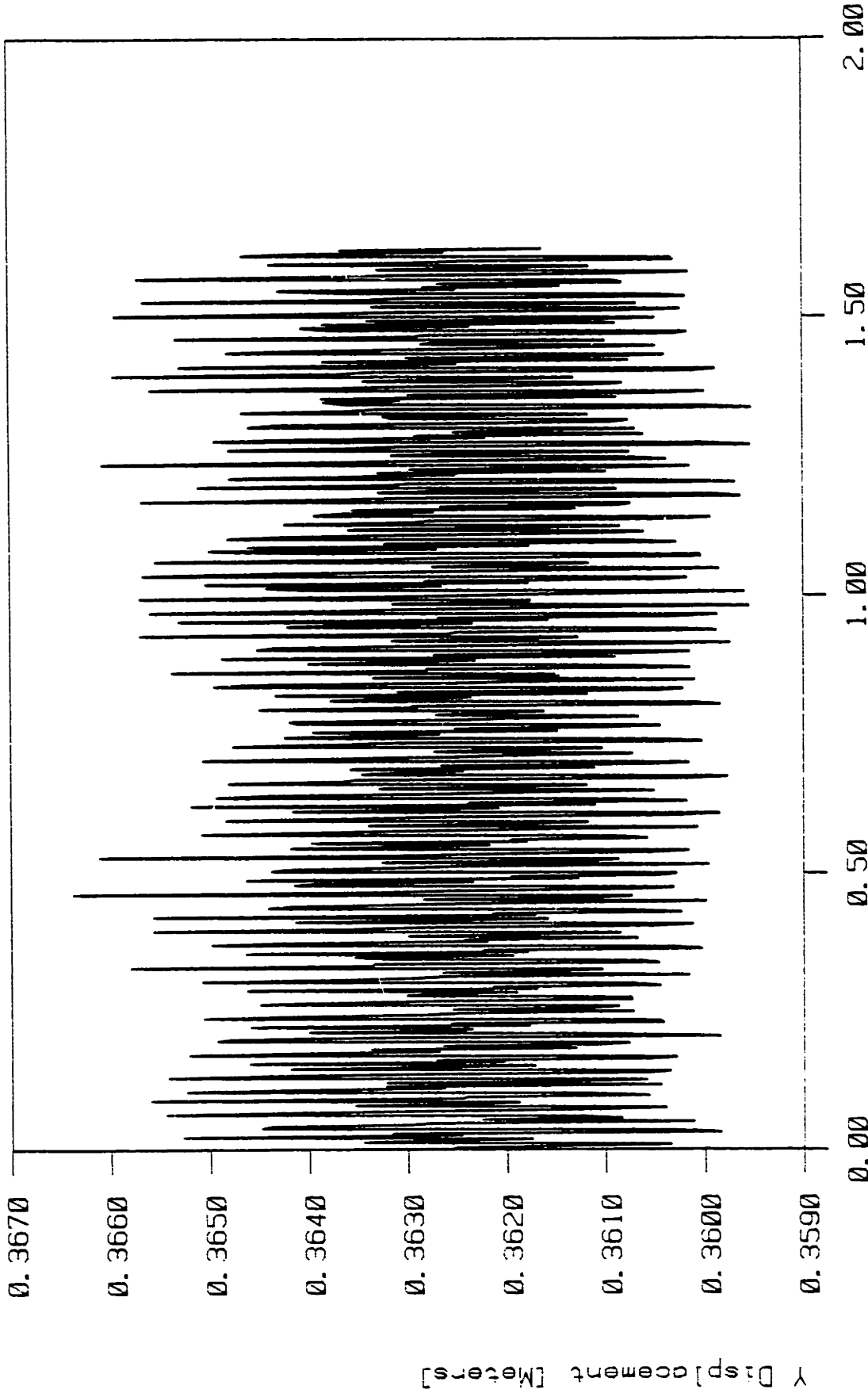
Figure 50. Unfiltered vertical displacement of a stationary segment

It is easily seen that sampling noise contaminates the
position history with a range of frequencies and with
maximum amplitudes of 3 mm. Since the segment was
physically held stationary, all of the non-stationary
components of this record are due to noise. The frequency
domain signature shown in Figure 51 is that noise.
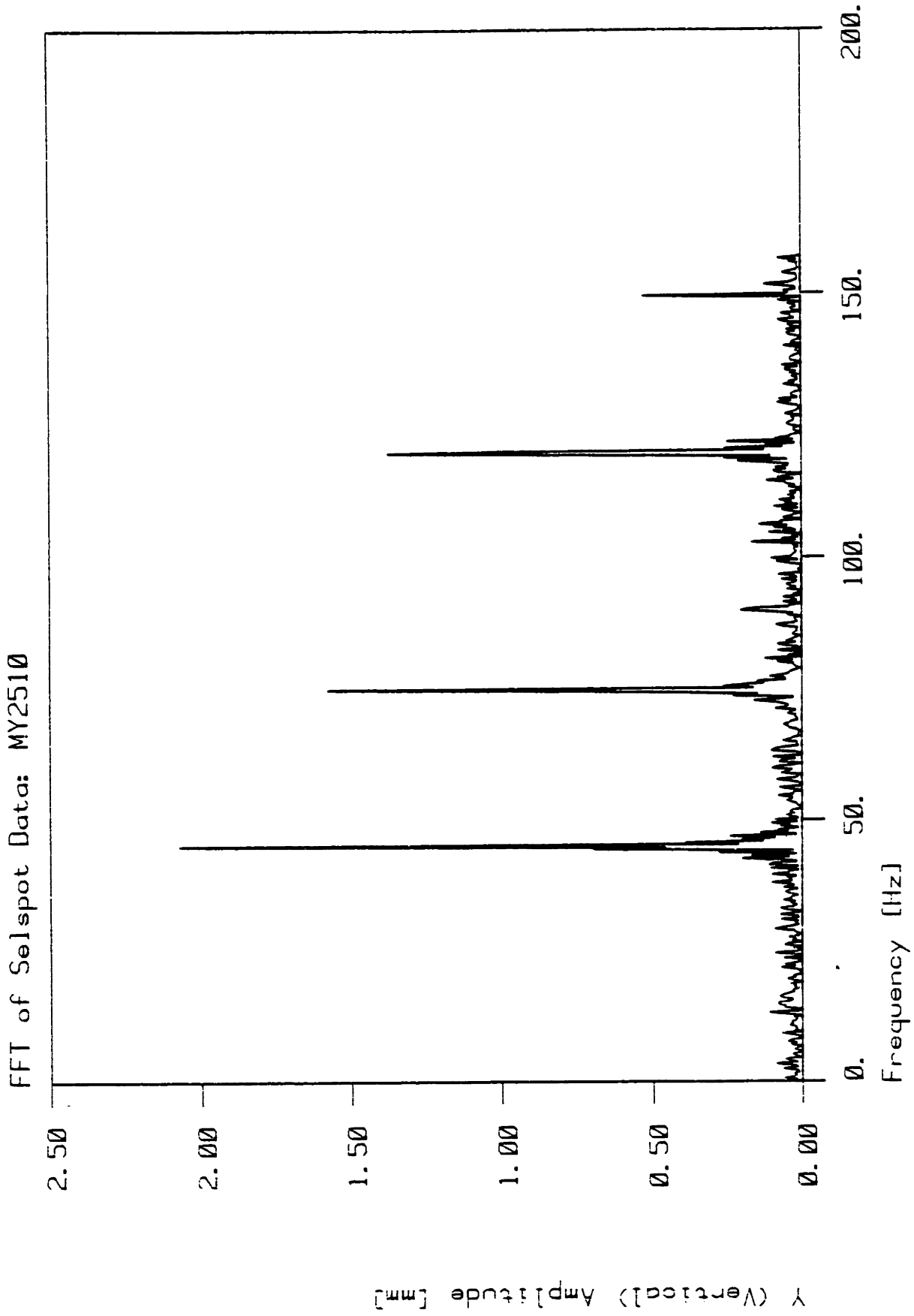
Figure 51.   FFT of the unfiltered vertical displacement

Precisely the same stationary LED array experiment was performed moments later with all of the overhead florescent lights <u>off</u>, and the results shown in Figures 52 and 53 were produced. As a comparison Figures 54 and 55 are from the same experiment repeated at midnight in total darkness.
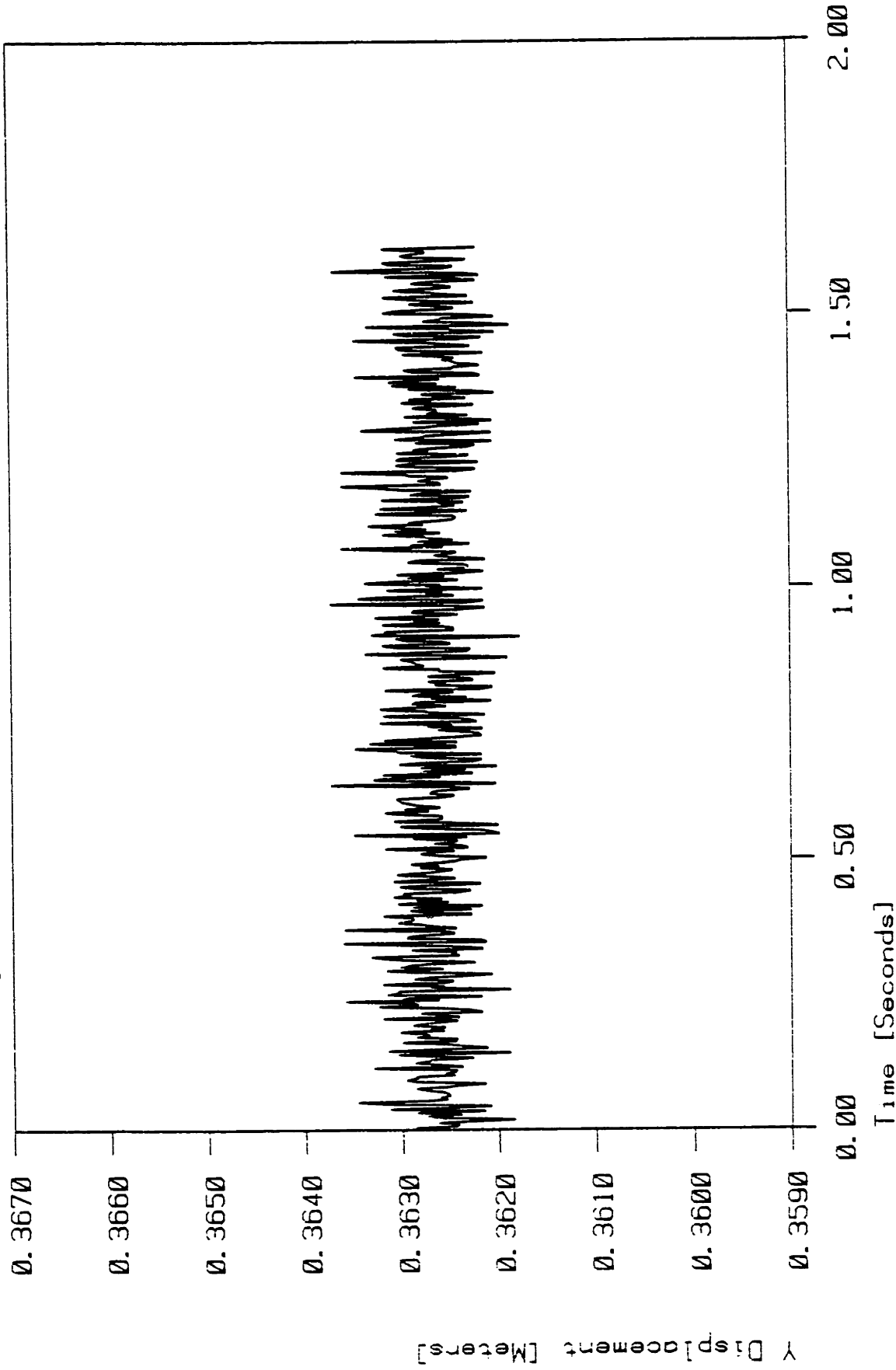
Figure 52. Unfiltered vertical displacement of a stationary LED array, florescent lights off
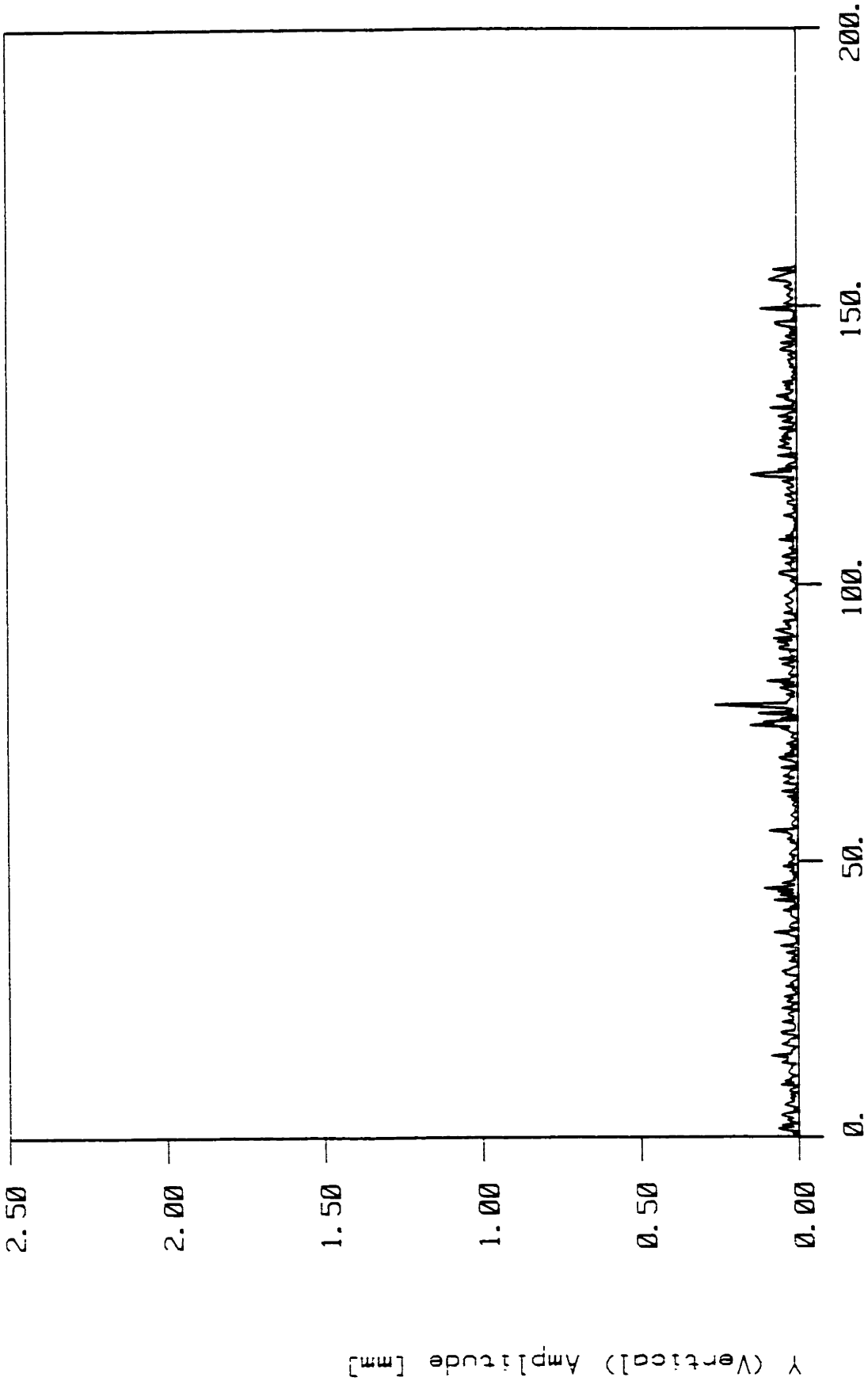
Figure 53. FFT of the unfiltered vertical displacement with the lights off
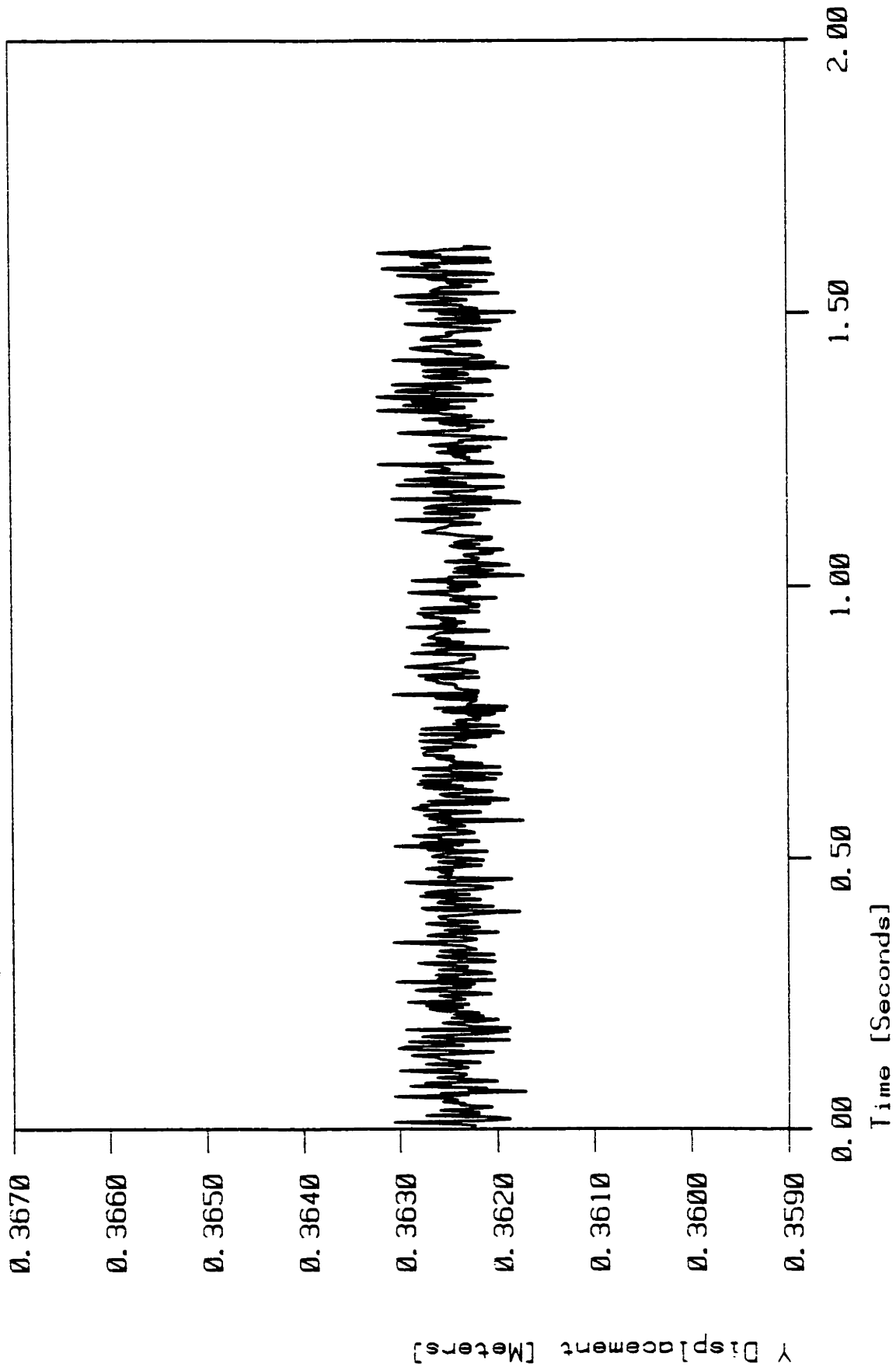
Figure 54. Unfiltered vertical displacement of a stationary LED array, data acquired in total darkness
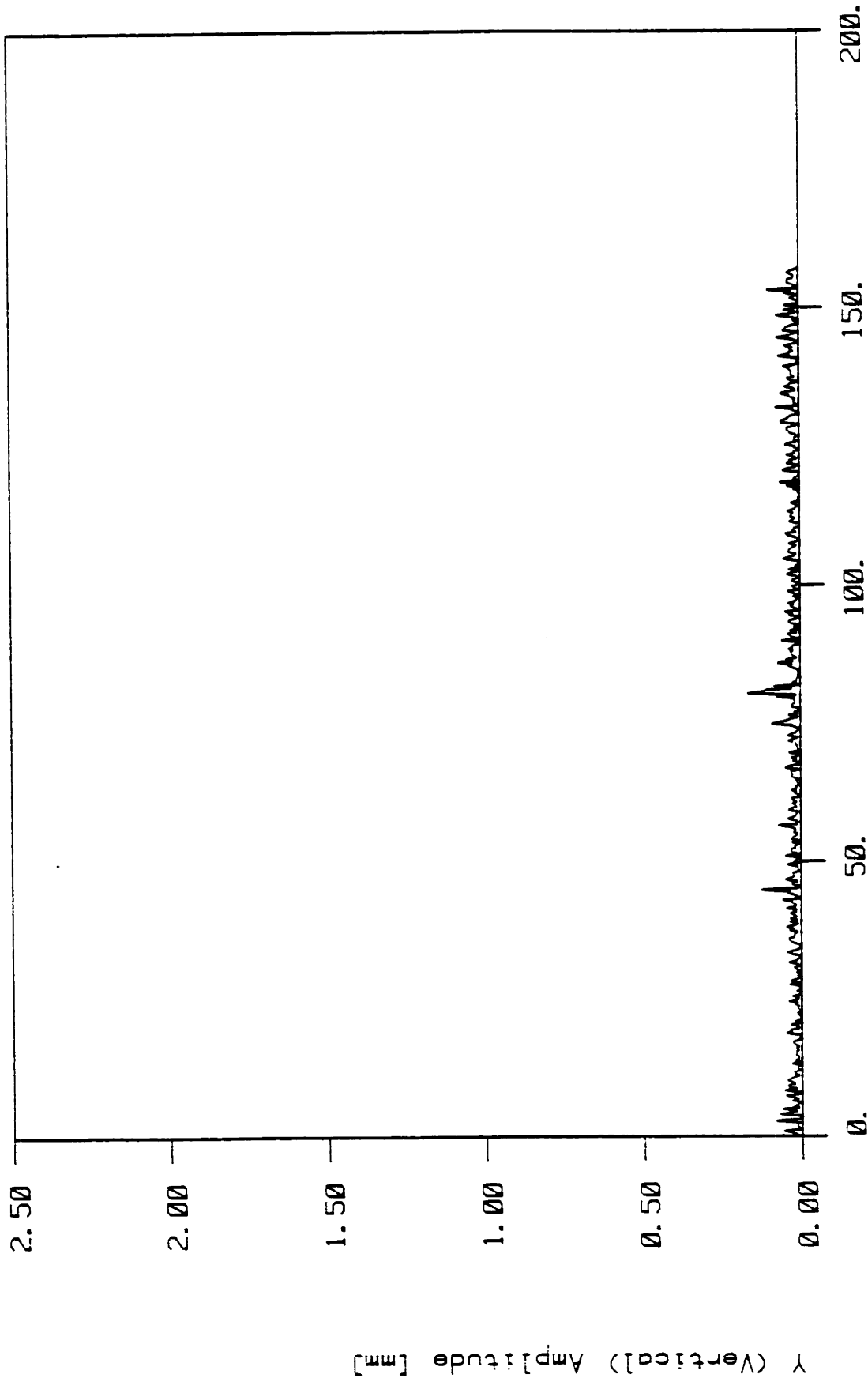
Figure 55. FFT of the unfiltered vertical displacement, data acquired in total darkness

The frequency domain noise signature of the Selspot system is highly repeatable. It has been observed in every position history acquired. The most remarkable aspect is that the first significant amplitude of noise occurs at approximately 40 Hz with the conventional overhead florescent lights on. Even with the lights off there are still small peaks in the frequency domain at 40, 75, 120, and 150 Hz, possibly due to florescent light leaking in from other rooms. Both of these experiments were conducted with bright, indirect sunlight entering the room. All experiments conducted as part of this investigation were performed with the room lights off and only sunlight illuminating the room. The 20-to-25-Hz frequency margin (between gait frequencies and the lowest significant noise frequencies) allows the application of a digital low-pass filter with a cutoff frequency between 15 and 40 to reject the majority of the noise and yet retain the significant frequencies in the signal. For all of the data filtering performed with this system, a 6th-order digital Butterworth filter applied both forward and backward in time was used. The power gain of this filter is shown in Figure 55a.

$$\left| H(jw) \right|^2 = \cfrac{1}{1 + 2\epsilon^2 \left(\cfrac{TAN\left(w\frac{T}{2}\right)}{TAN\left(w_c\frac{T}{2}\right)} \right)^{2N} + \epsilon^4 \left(\cfrac{TAN\left(w\frac{T}{2}\right)}{TAN\left(w_c\frac{T}{2}\right)} \right)^{4N}}$$

Where:

$\epsilon = 1.0$

w = the signal frequency

w_c= the filter cutoff frequency

N = the number of filter sections = 6

$T = 1/f$ = the time between samples = $1/315$ = 0.00317

Figure 55a.   6th order, 2 pass digital
Butterworth filter power gain equation

Figure 56. 15 Hz Filter Amplitude gain

6th-Order 2-Pass Butterworth Filter, 15 Hz Cutoff
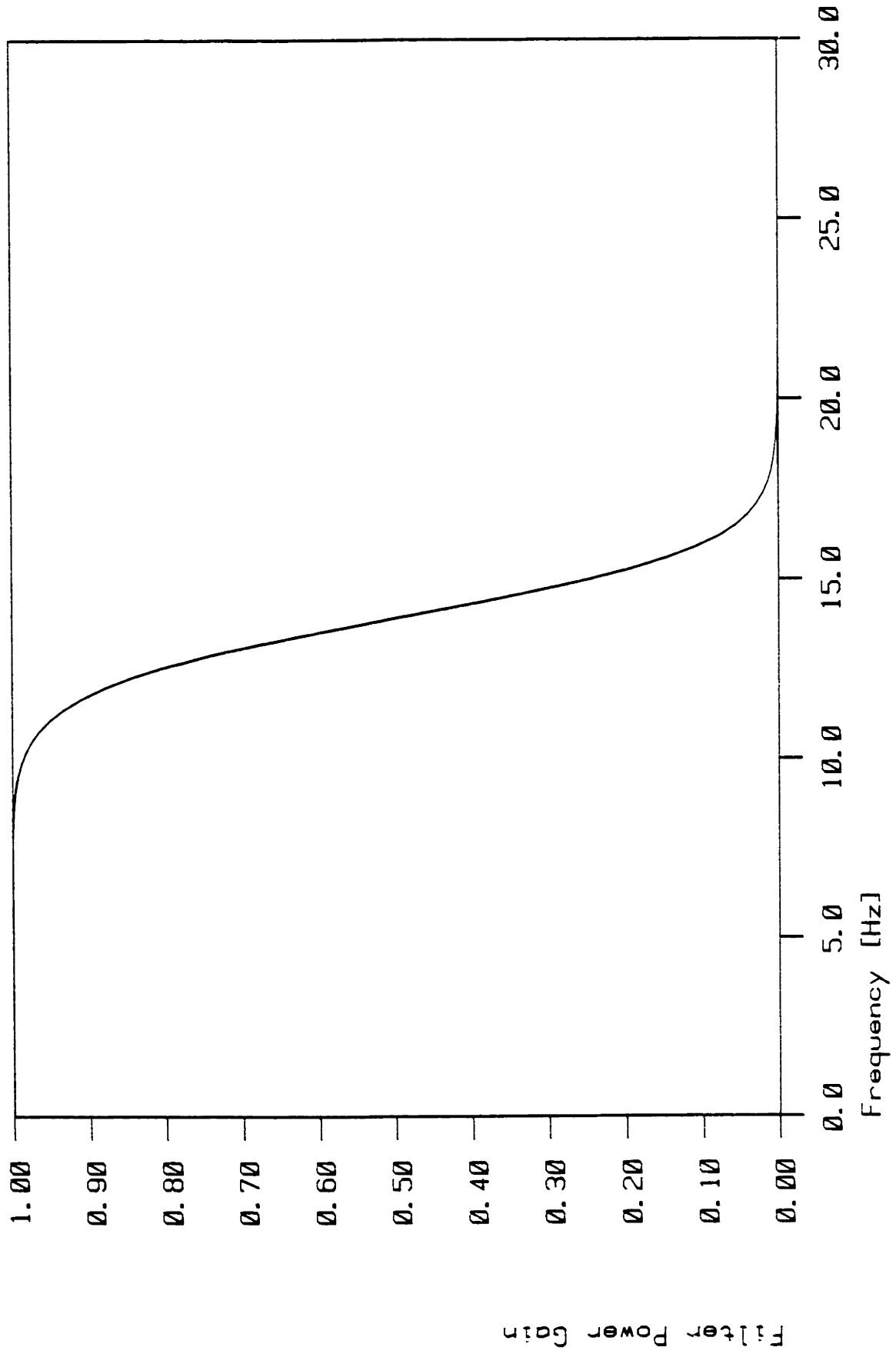
Figure 57. 15 Hz filter Power gain
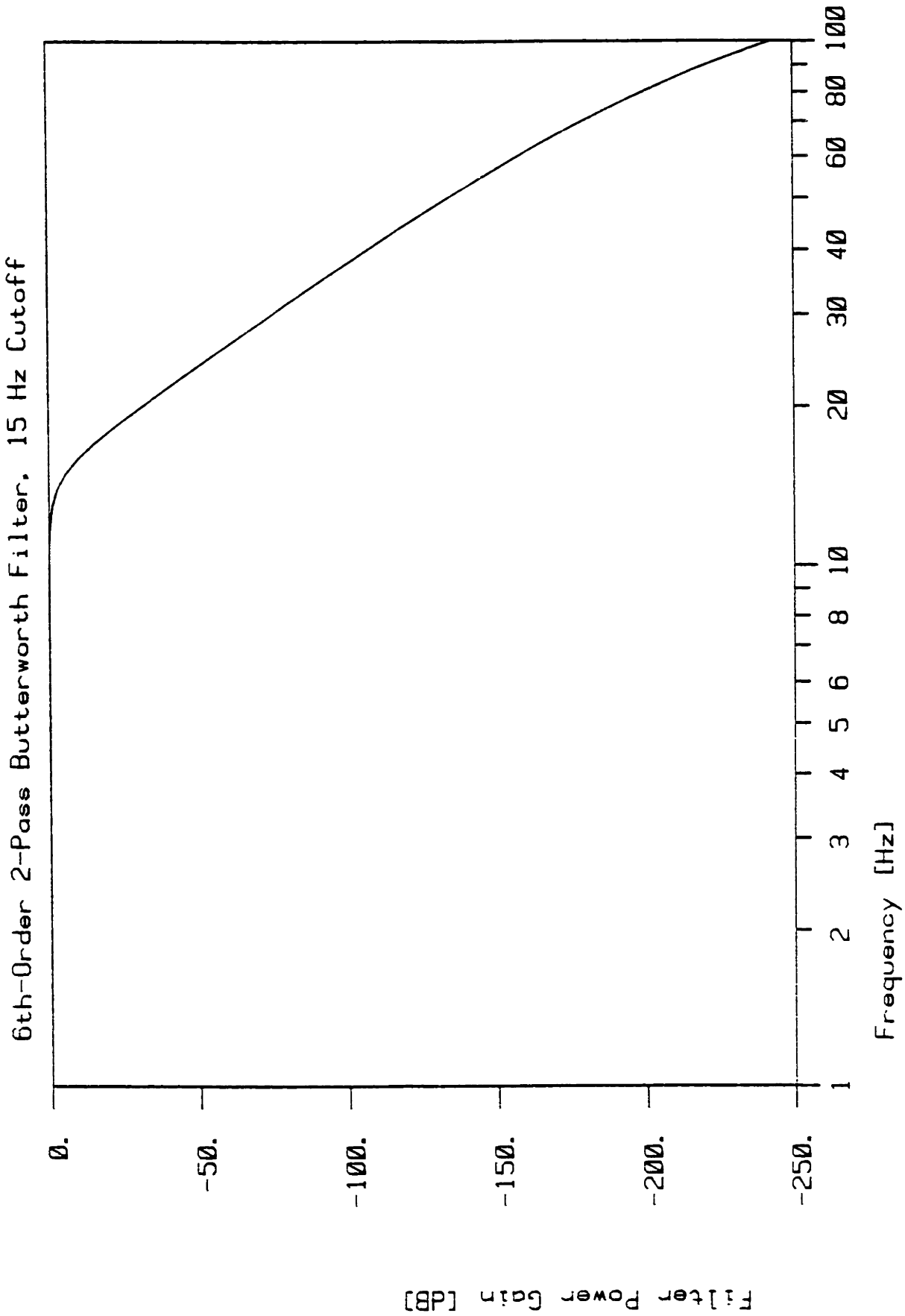
Figure 58.   15 Hz filter Power gain

The filter is applied forward and backward in time for all of the off-line analysis to eliminate any phase shift. This results in a filter with a frequency response very close to a 12th-order Butterworth. The only deleterious effect of applying the filter forward and backward is that the portion of the end of the data set that is shifted beyond the end of the sampled time is not recovered when the filter is applied backward, and the negative phase shift is encountered. This results is a short section at the end of each filtered time history with a slope of zero.

To complete the frequency domain analysis, a final question must be addressed: How much influence will the residual noise (that remaining at low frequencies subsequent to low-pass filtering) have on the final dynamic results? Or: Where does this residual noise lie on the iso-acceleration curve presented in Chapter 2? For an approximately 1-Hz phenomenon with an amplitude of 1.0 meter this residual noise will contribute less than 2 percent once the signal has been double-differentiated.

MY2502 Segment #1. w.r.t. Gnd. Stationry 15HzFilter No Lights

Y Displacement [Meters]

0.3670 0.3660 0.3650 0.3640 0.3630 0.3620 0.3610 0.3600 0.3590

Time [Seconds]

0.00   0.50   1.00   1.50   2.00
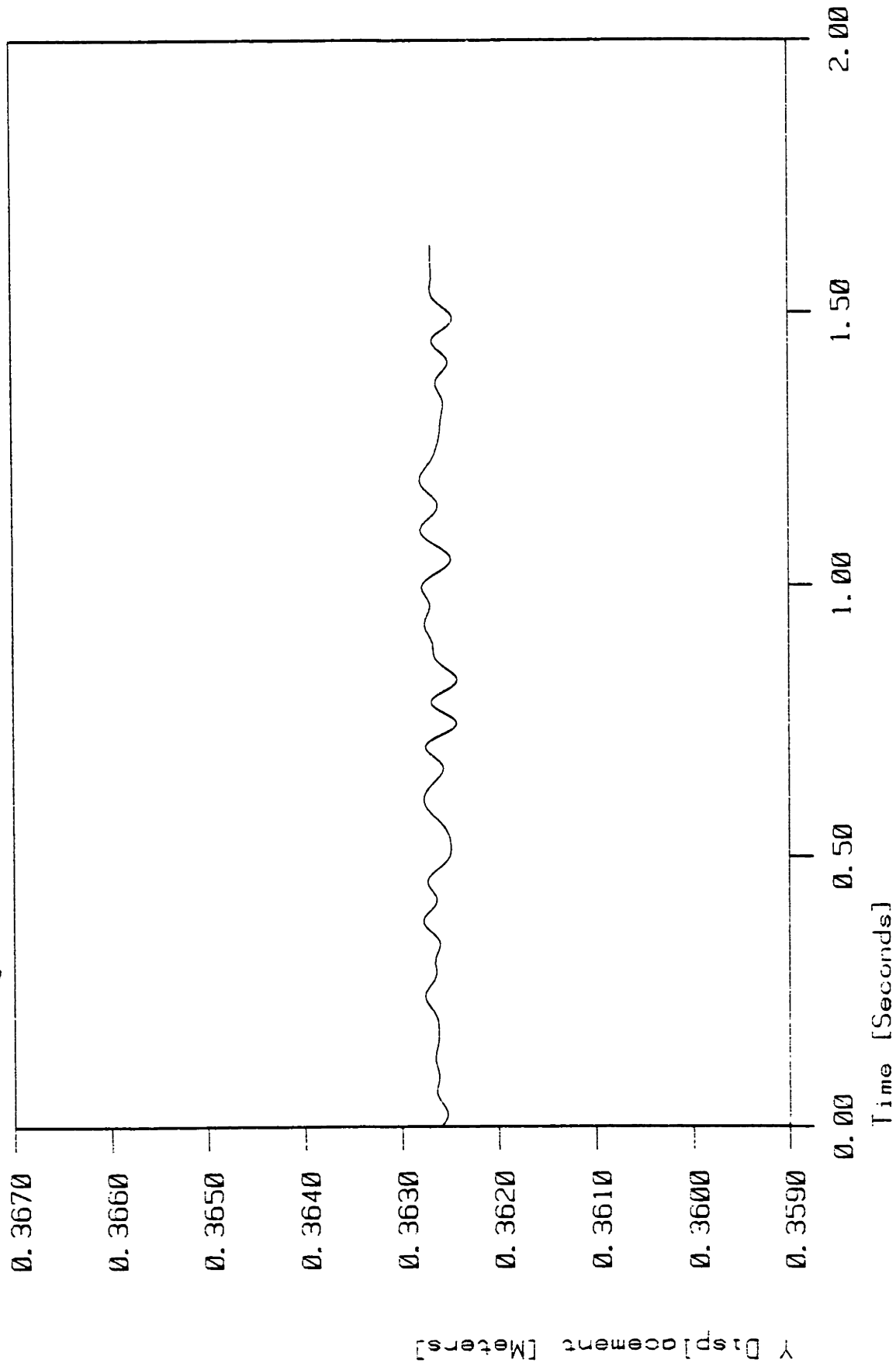
Figure 59.   Filtered vertical displacement (lights off)

FFT of Selspot Data: MY2502

Y (Vertical) Amplitude [mm]

2.50

2.00

1.50

1.00

0.50

0.00

0.   50.   100.   150.   200.
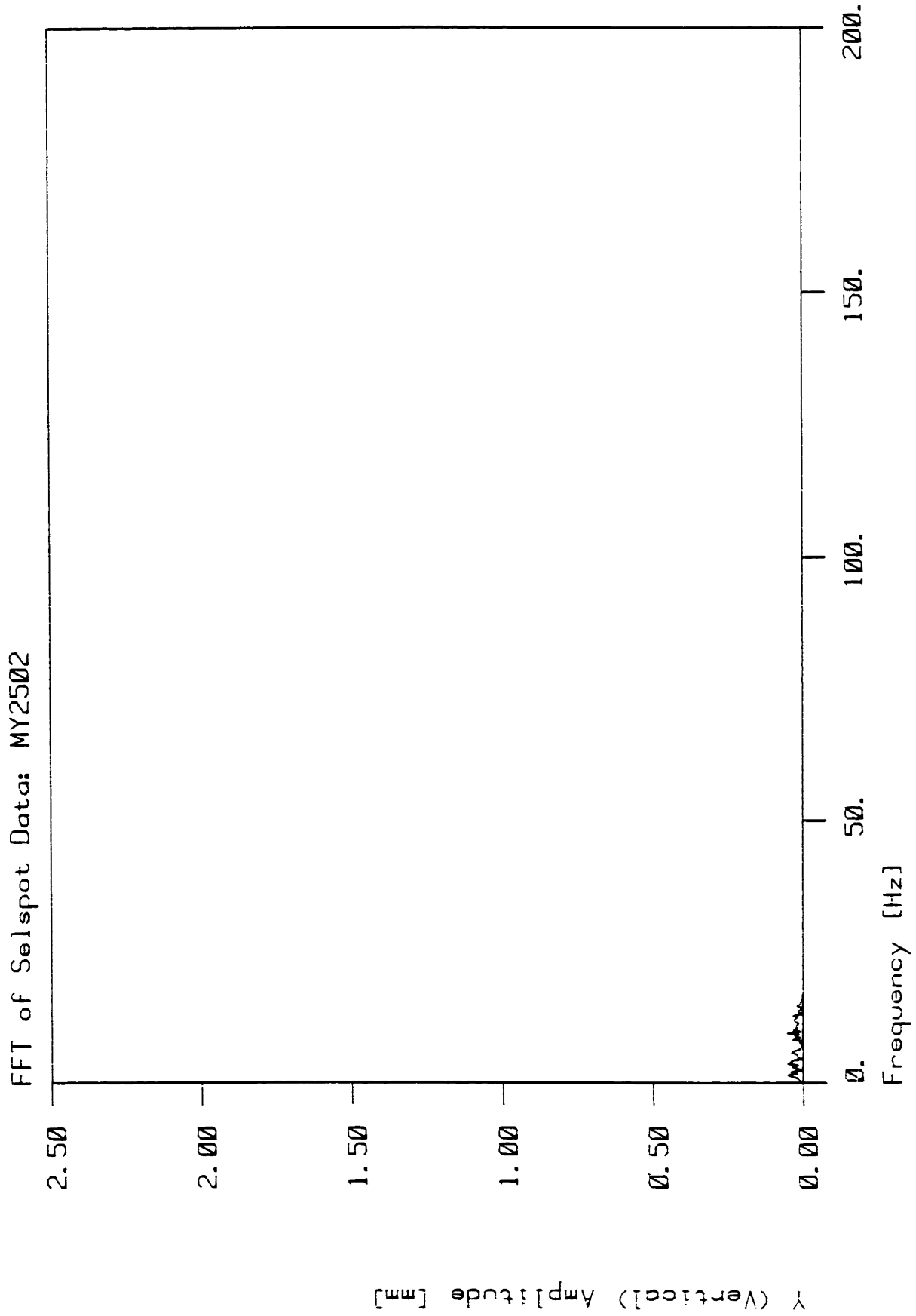
Frequency [Hz]

Figure 60.   Residual noise amplitudes in the frequency domain

Note that the amplitudes remaining are on the order of 0.01 mm, which accounts for their limited influence once double differentiation has been performed.

The following figures demonstrate the action of the low-pass filter on a motion trajectory.
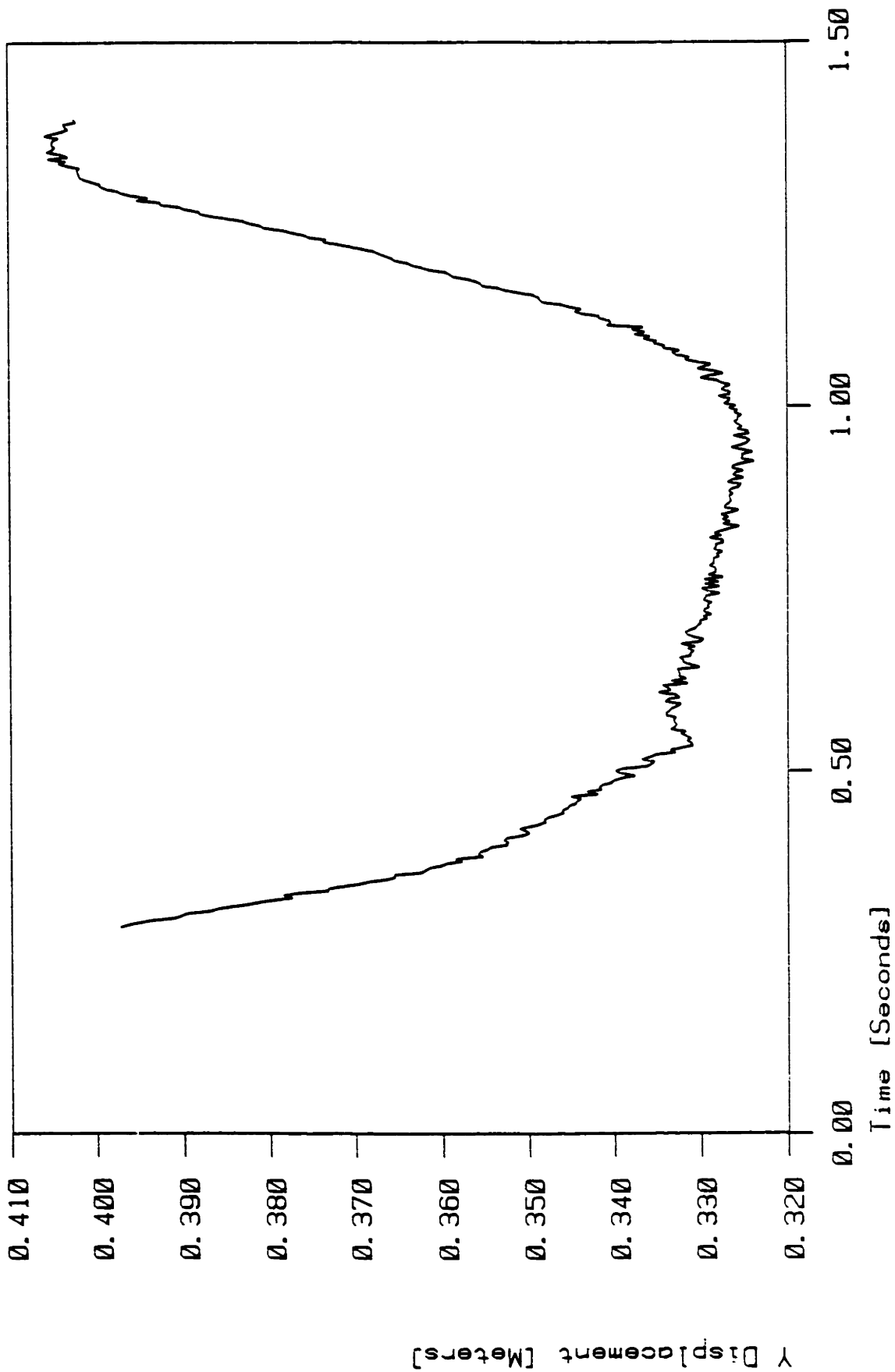
Figure 61.   Unfiltered lower leg trajectory in gait
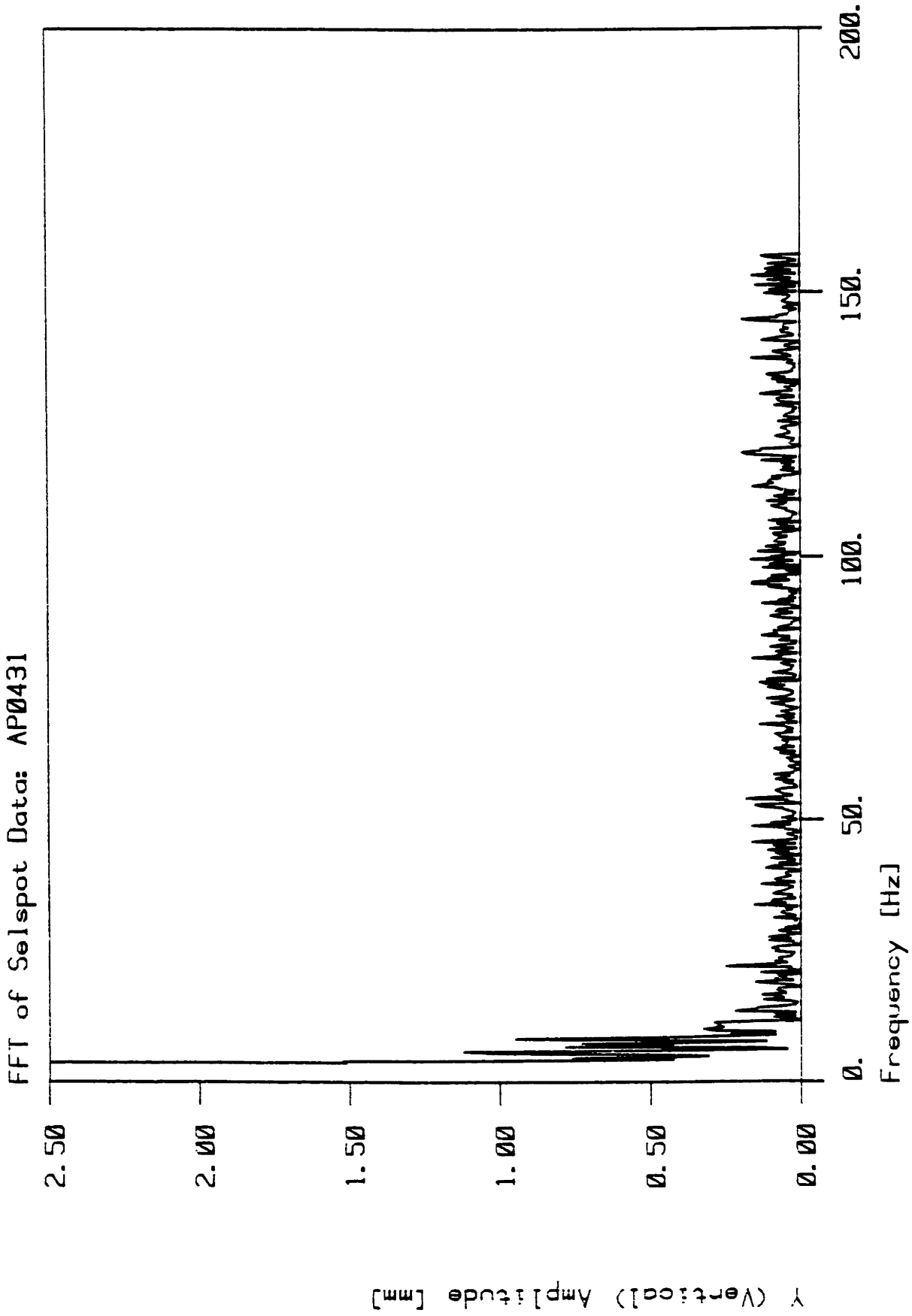
Figure 62. FFT of the unfiltered trajectory
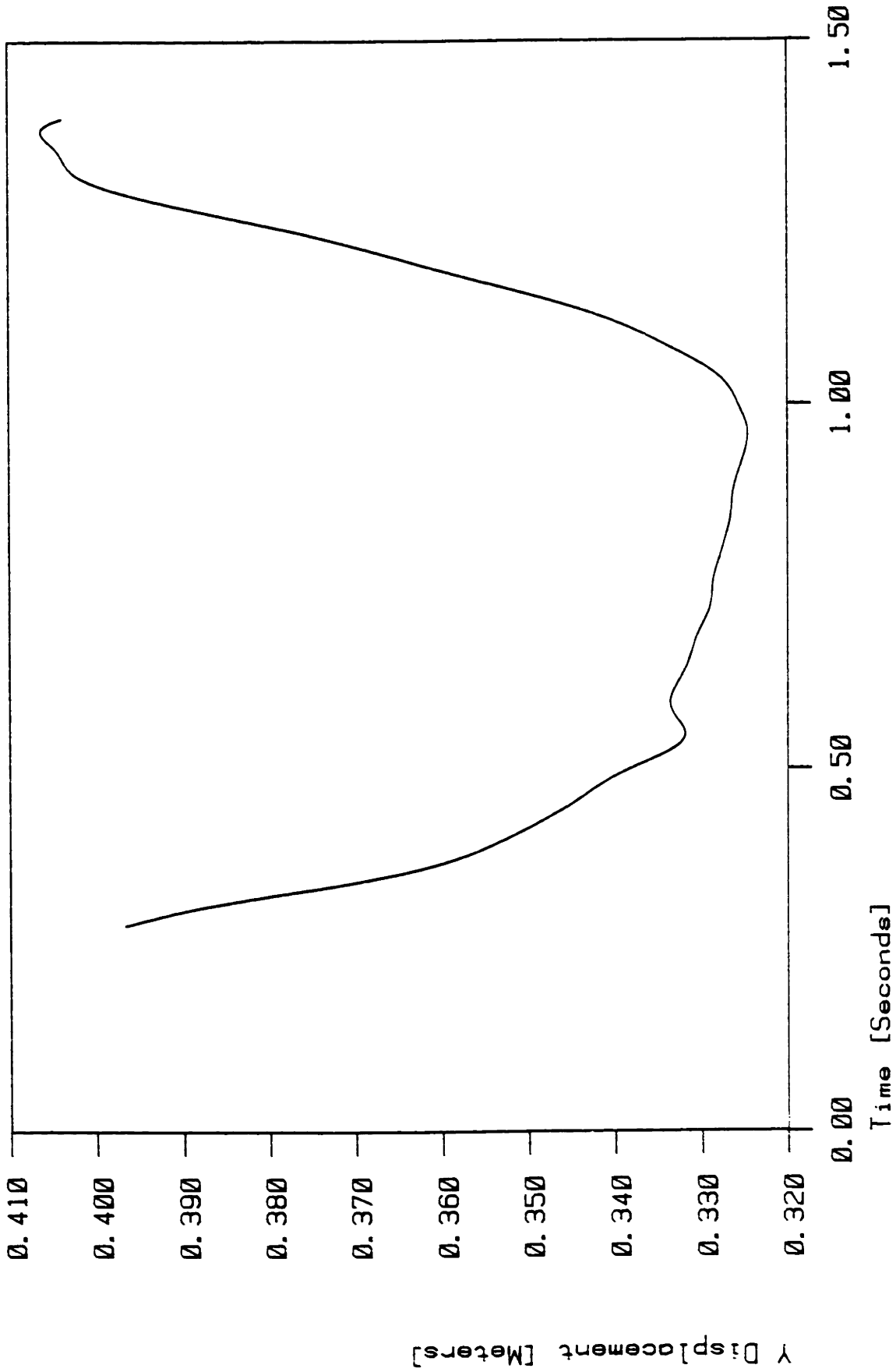
Figure 63. Filtered lower leg trajectory in gait
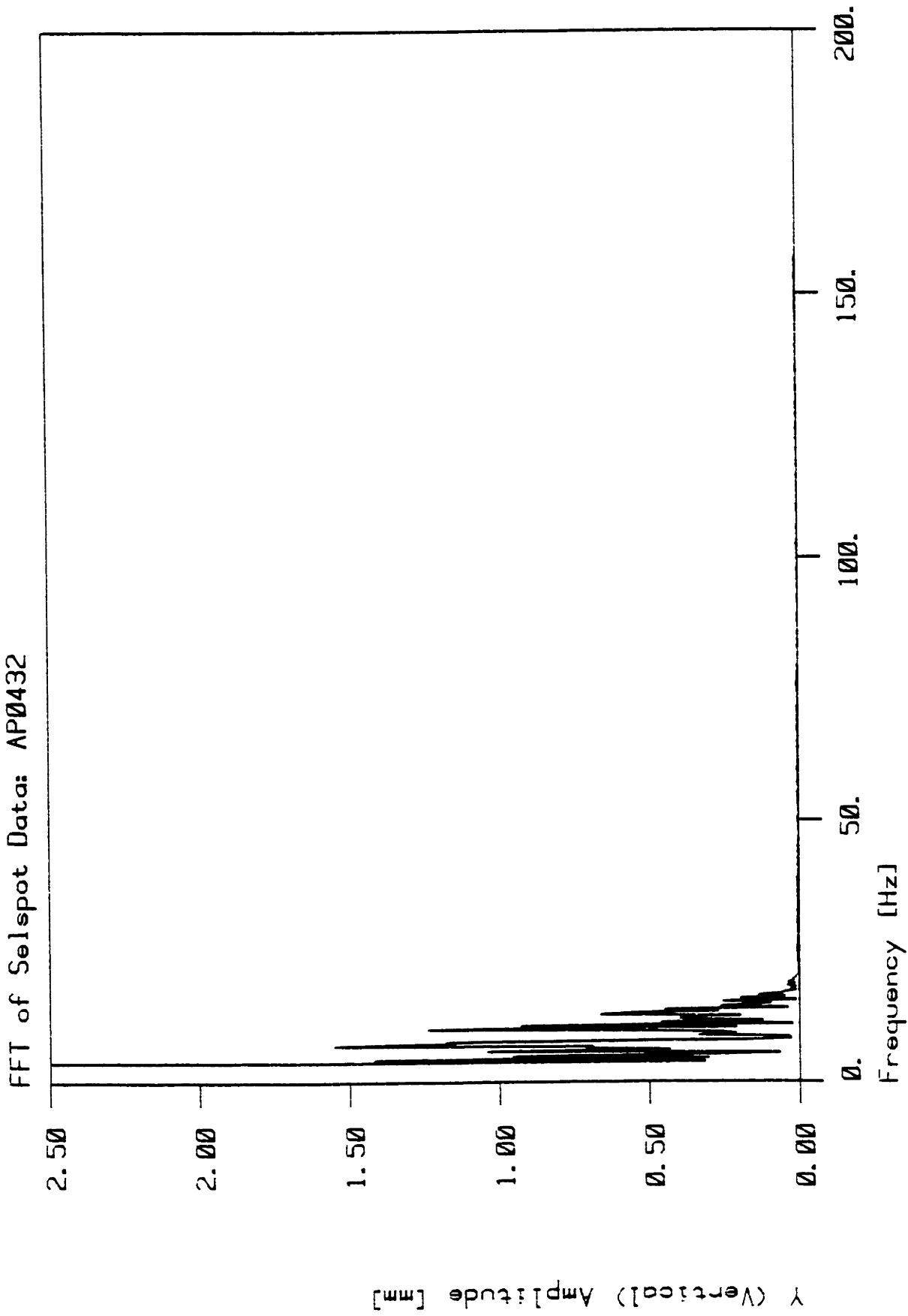
Figure 64. FFT of the filtered trajectory

A motion transduction, accumulation, and reduction
system with a spatial resolution of 1-mm, and absolute
accuracy of 1-mm, a 315 Hz sampling rate of thirty targets,
a well-known noise signature with no significant components
at the desired signal frequencies, all combine to produce a
system capable of producing kinematically, fully described
human motion trajectories with an error bound smaller than
0.1 percent and accelerations with an error bound smaller
than 2 percent.

CHAPTER 4

DATA ANALYSIS APPROACH

The use of imbedded, three-dimensional, body coordinate systems (BCS) is perhaps the most fundamental feature which separates this entire human motion acquisition and analysis system from more conventional techniques. Each instrumented segment has a fixed BCS defined at the start of an experiment, and all kinematic data derived from the measurement of the LEDs are referenced to those BCSs.

Figure 65 is a typical arrangement of LEDs (shown without the mounting apparatus for clarity). The position of each LED is described by a vector in the BCS. Several physiologic features, including the center of mass location and the locations of momentary axes of rotation (MAR), are also described by vectors. The MARs are axes of relative rotation described by the kinematics of one segment with respect to another. They will be more fully discussed in the next section. Once the accumulated LED data have been processed, the motion is represented by a vector from the origin of the (fixed) Global Coordinate System (GCS) to the origin of each BCS for each sampled time interval. The three by three rotation matrix characterizing the change in orientation from the original, as specified by the vectors from the BCS origin to each LED, is also included for each time step.
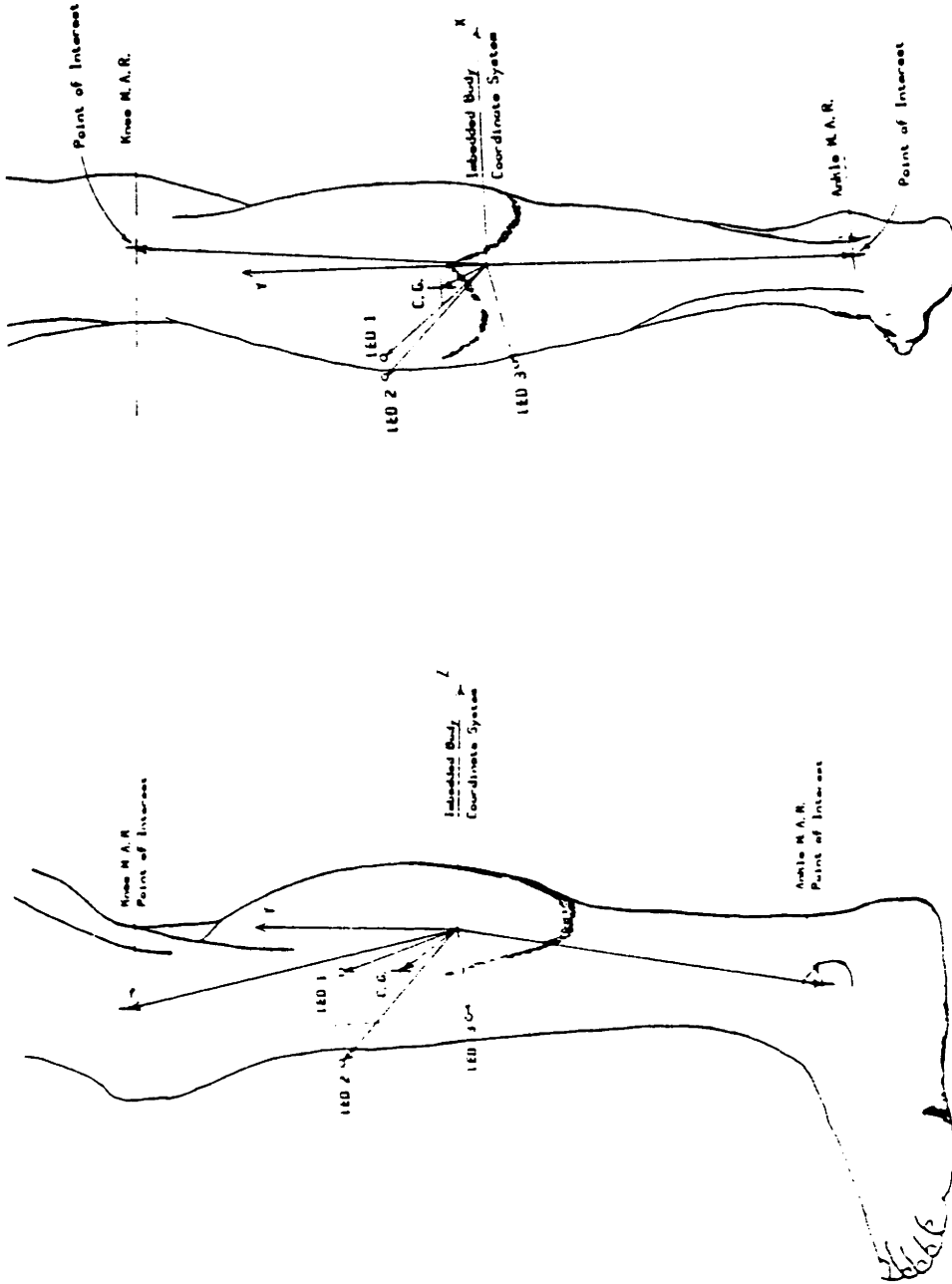
Figure 65. 2 views of the lower leg with BCS and LEDs

MOMENTARY AXES OF ROTATION

All of the articulations in the human lower extremity
are predominantly rotational; there are no prismatic
joints. Among the three under study, however, only the hip
can confidently be classed as a purely rotary joint. Many
investigators have determined that both the knee and the
ankle exhibit moving axes of rotation (Frankel, et al.,
(1971) [60], Soudan (1979) [147], Panjabi (1979) [120],
Panjabi (1981) [121], Wright, et al., (1964) [179], and
others). Since this investigation (as well as the human
lower extremity) is fully three-dimensional, axes of
rotation will be discussed. Centers of rotation, used by
many researchers, apply to planar relative motion. As
mentioned earlier, the BCS format of data accumulation and
reduction in TRACK yields data uniquely suited to the
kinematic analysis of linkage geometry and rotational axis
location.

The axis solution unfolded as a problem of substantial
complexity. To clarify the discussion, an introduction to
the kinematics of relative rotation will be presented. A
good starting point is the classical instant center method
of Reuleaux (1963) [130]).

Given a body in planar motion with respect to another, the motion at an instant can be characterized by velocity vectors at two points fixed in the first body. The intersection of lines perpendicular to these two velocities is the instant center of rotation. All rotation can be characterized by an axis; however, if the rotation is confined to a plane, the axis degenerates to the point of intersection of the plane and the axis.

An alternative but equivalent method utilizes the translational and angular velocity of a point fixed in the body. The instant center lies along the vector representing the cross product of the angular and the translational veclocity, at a distance from the point equal to the magnitude of the translational divided by the magnitude of the angular velocity.
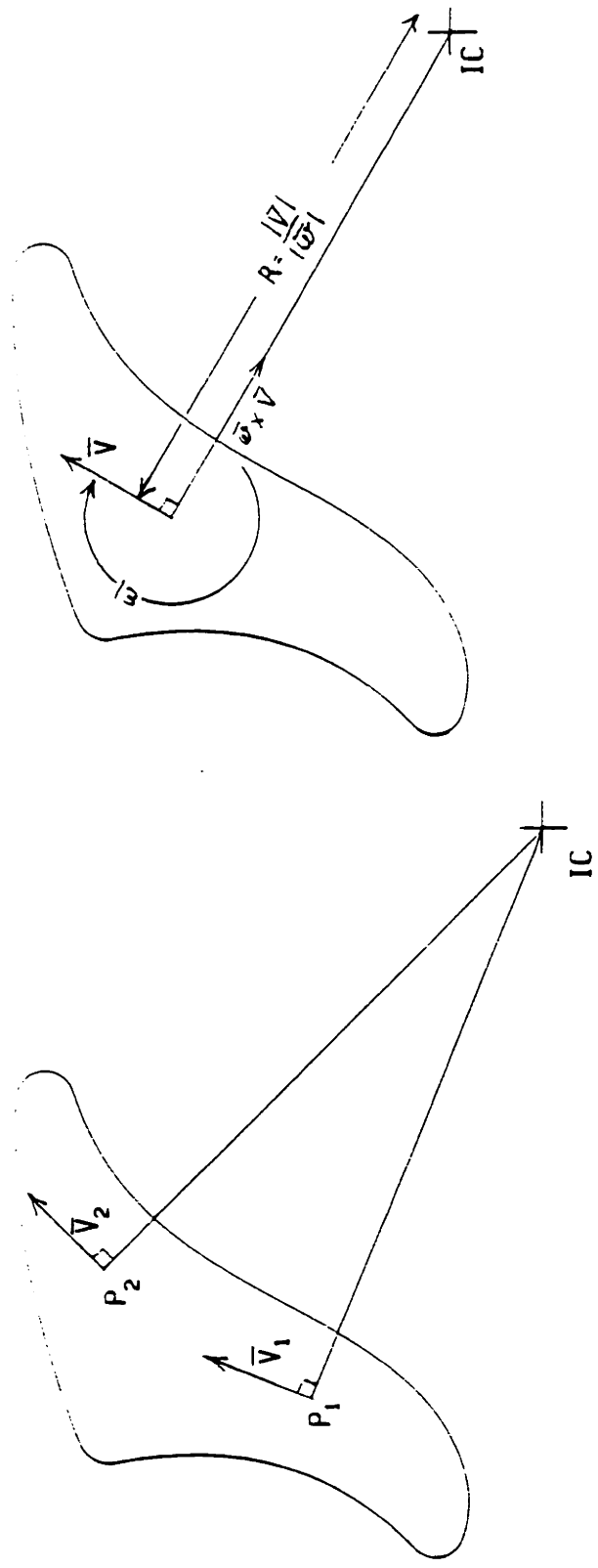
Figure 66. Graphical demonstrations of the 2 different Reuleaux methods

These techniques work well if very high quality velocity data are available, as in graphical or analytical analyses of mechanisms. The presence, however, of small measurement error or noise will introduce very large errors in the location of the instant center.
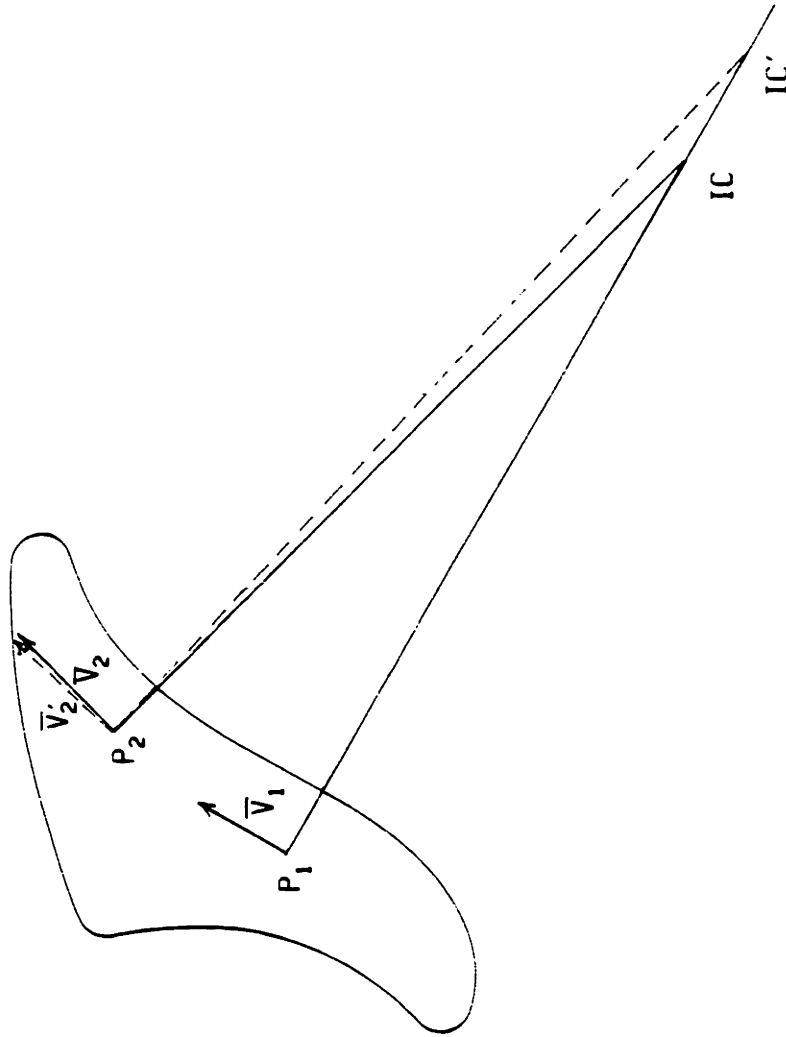
Figure 67. Error magnification using Reuleaux's method

Several researchers have examined the kinematics and patho-kinematics of the knee joint using a modification of Reuleaux's method (Frankel, et al., (1971) [60], Panjabi (1979) [120], and Soudan, et al., (1979) [148]). The modification they apply is simple, but greatly decreases the ability of their method to resolve axes (or centers) of rotation correctly. In Reuleaux's method, the instantaneous relative velocities of two points on one body with respect to another are used. Their modification uses two points on one body as well, but instead of using the relative velocity of either point, they use another (not infinitesimally) displaced position to determine an average velocity of each point. The method proceeds by determining the displacement vector for one point between time: t, and time: t plus delta t. This average velocity vector is perpendicularly bisected. The velocity determination and bisection is then performed for another point on the body. The intersection of the two bisectors is taken to be the instant center of rotation.
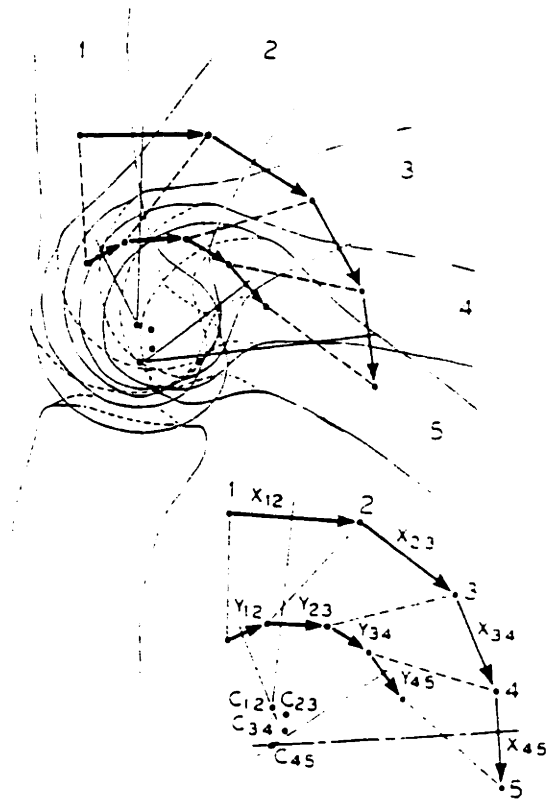
Fig. 4

Construction of the instant center pathway. In this example, the two points X, Y on the femoral axis are placed at arbitrary points six and ten centimeters above the articular surface of one of the femoral condyles. The lines X^{12} and Y^{12} are the vector displacements of points X and Y between positions 1 and 2 of the femur. C^{12} is the centrode for the displacements X^{12} and Y^{12}. The lines X^{23} and Y^{23} and so on are subsequent displacements and C^{23} and so on are the corresponding centrodes.

Figure 68. Figure 4 from Frankel et al. (1971) [60] page 949
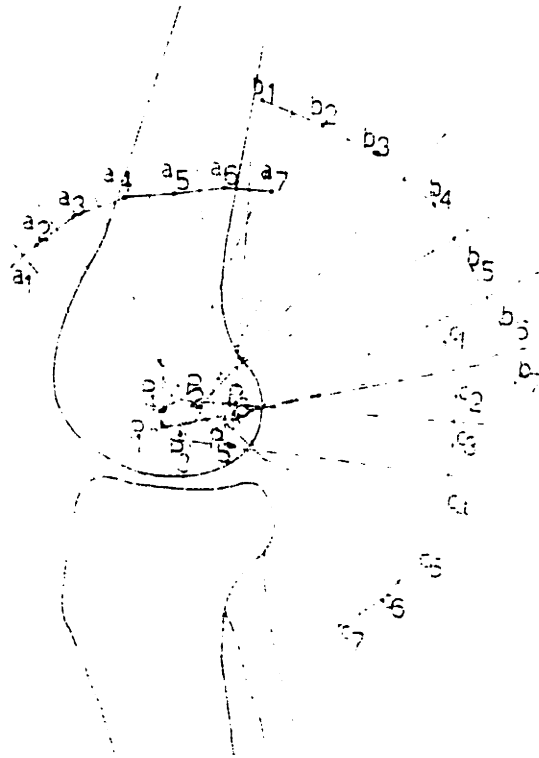
Fig. 1. Instant center pathway of knee-motion is found by the Reuleaux-method from sequential radiographs. A number of dispersed points were found while one would expect a continuous pathway of the instant center.

Figure 69. Figure 1 from Soudan et al. (1979) [148] page 27

It can readily be seen that small errors in the
positions of any of the points will result in large errors
in the center of rotation, and even with highly accurate
position data, the true kinematics of the joint will not be
determined by this piecewise discretization of the motion.

> "... we want to point out that by taking
> sequential radiographs of the joint during its
> movement, the motion is defined discretely,
> providing an 'idea' of the real movement. From
> that 'idea' only an estimation of the instant
> center pathway can be deduced. This estimation
> will be better with more data points, although
> their number is experimentally limited."
> Soudan, et al., (1979) [148] p.29

Soudan, et al., discuss three very important aspects of
the resolution of axes of rotation. First, all methods are
highly sensitive to very small errors in the sampled
kinematics; second, since virtually all relative motion in
humans is spatial (not planar), special care must be taken
in determining axes of rotation; and finally, the
two-point, piecewise average kinematic rotation-center
technique is poor, and a method based on Reuleaux's method
using intersecting perpendiculars to relative velocity
vectors is far superior.

An averaging technique was adopted in this
investigation to minimize the effects of small position
measurement errors. A short trajectory of points (a
"moment" in time) was used to decrease the errors of axis
location. To accomplish this, the best fit plane (in a

least squares sense) was fitted to the trajectory. Then the best fit sphere, subject to the constraint that the center lie on the plane, was found. The axis perpendicular to the plane, and passing through the center of the sphere was taken as the Momentary Axis of Rotation (MAR) for that trajectory. If the trajectory used is shorter than the full range of relative motion, a moving axis can be found. If the entire range is used, the result will be the average location of the axis. This method will only work for joints that are predominantly rotational, with a small ammount of translation (compared to the link lengths). The next section will examine how well this approach, applicable only to a limited class of motion between bodies, applies to the joints in the human lower extremity.

To demonstrate this technique, a pendulum (used earlier in the discussion of calibration verification) was instrumented and placed in the viewing volume of both cameras. The path of the BCS origin fixed in the pendulum from this experiment was fitted to a plane, and then a sphere (once again subject to the constraint that the center of the sphere lie on the best fit plane). For this example, the distance on the physical pendulum from the point of rotation to the origin of the BCS was 340 mm. Figure 70 and 71 display the pendular data viewed normal to, and along, the best fit plane. Notice that the sphere radius (R) is only 1.1 mm [0.044 inches] shorter than the pendulum radius.

The center-of-rotation location was also found to be within 1 mm of the correct height above the floor (Y coordinate) by independent measurement. "URV" in the plot legend stands for: Unit Rotation Vector. This is the unit vector directed along the axis of rotation. For this experiment the vector is almost entirely along the global Z axis, indicating that the pendulum moved almost entirely in a plane parallel to the cameras.

- 210 -

Figure 70. Pendulum data for
axis of rotation along Z,
displayed in the best fit plane

Axis

(0.0198, 0.7760, -0.0344)
R= 0.3390 [meters]
URV=-0.0057, -0.0109, 0.9999
Frames: 1 to 630
TK4: AP1502.DAT:0
Seg# 1 wrt Seg# 0.   ~Z/Y Plane
AP1501 w/ 10Hz Filter

Plane

Data

Sphere

10 mm
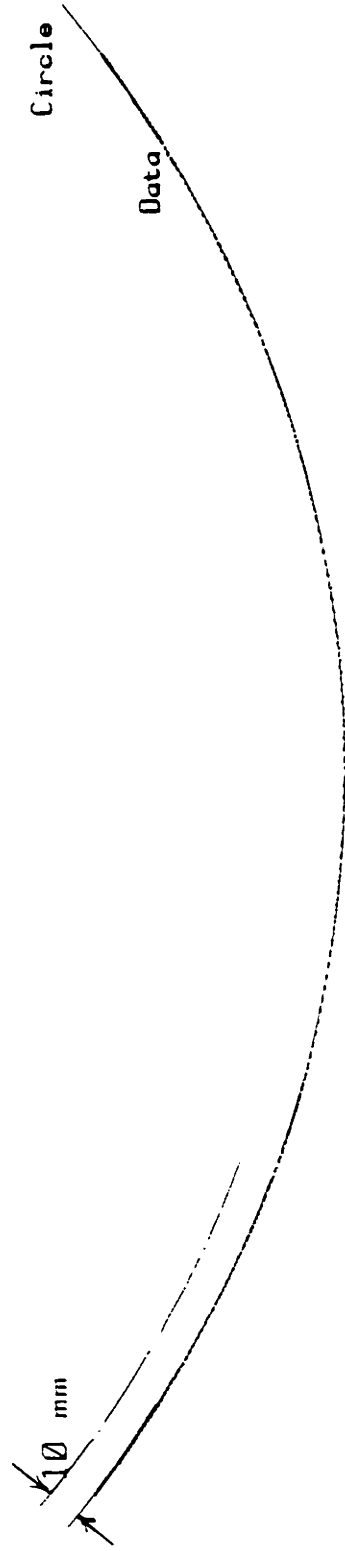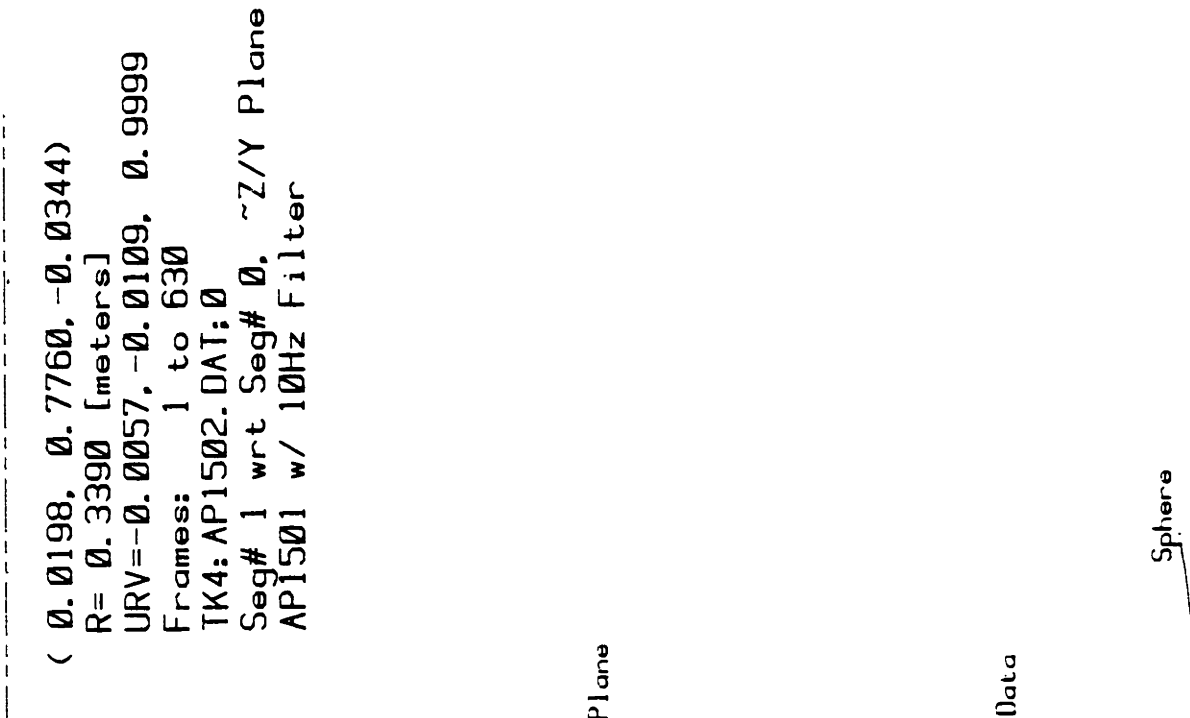
Figure 71. Pendulum data for
axis of rotation along Z,
motion out of the best fit plane

Figures 32 through 37 (in Chapter 3) show similar data for two quite different orientations of the pendulum axis.

There is a flaw in this axis of rotation solution method. To demonstrate it, consider a wheel with a body coordinate system at its center rolling on a plane. The path of the BCS origin will be a straight line, and the axis of rotation will be found at an infinite distance from the plane. Using Reuleaux's method, the instant center of rotation is at the rolling contact point. The different locations occur because none of the data describing the changing orientation of the BCS were taken into account (as in using translational velocity divided by angular, see above). Despite this, the "translational-path-only" technique does provide useful results for a limited class of motion. To describe that class (limiting ourselves to plane motion for the moment), use the same wheel and plane, but now include a bar projecting beyond the rim of the wheel, along a radius. See Figure 72.
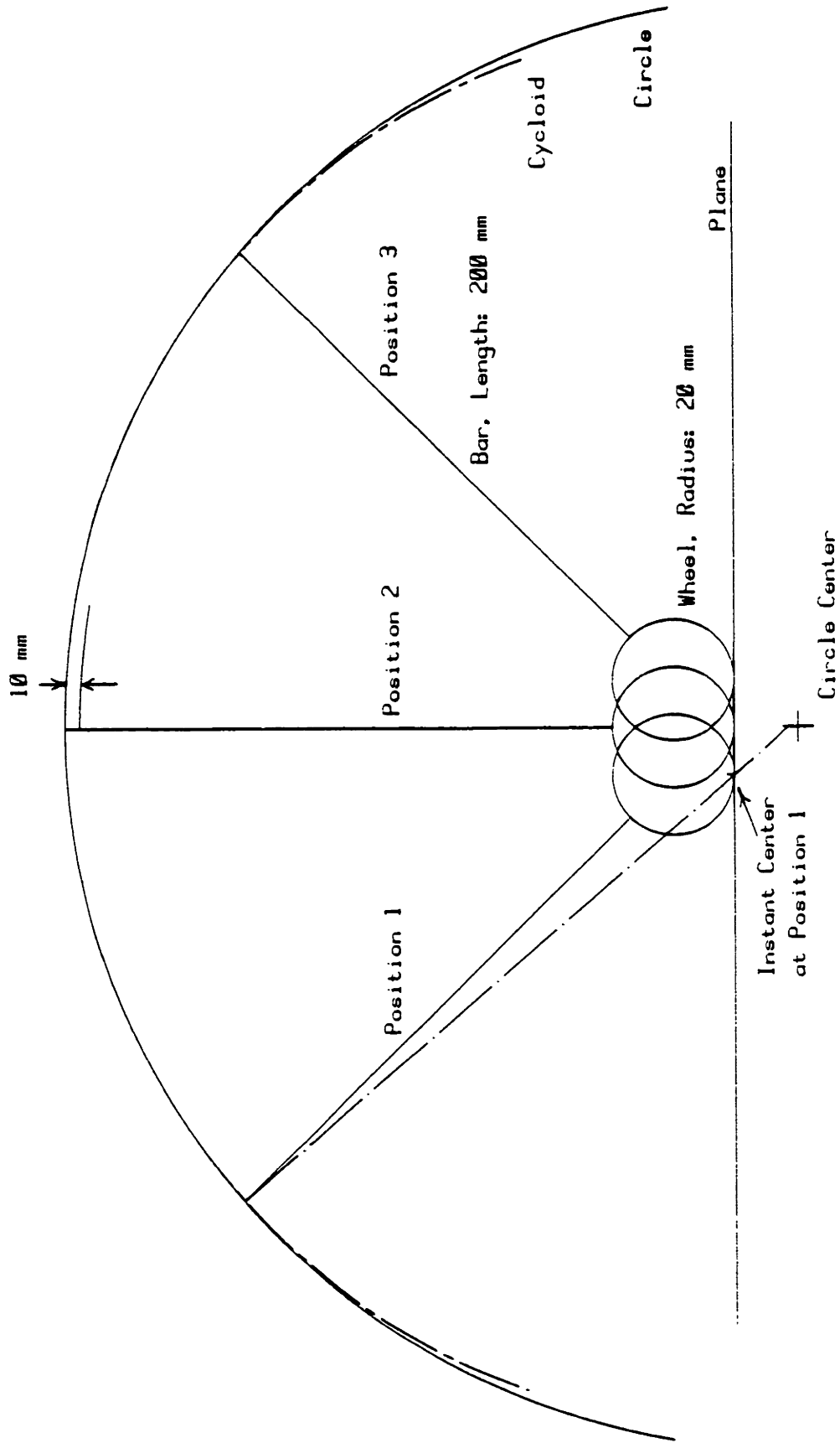
Figure 72. Wheel, Plane, and Bar for prolate cycloid description

The path of any point on the bar is a cycloid. Points inside the wheel rim are <u>curtate cycloids</u>; points outside describe <u>prolate cycloids</u>. For a point on the bar at 10 radii from the wheel center, the wheel can roll +/- 45 degrees and the path will remain virtually circular. This closely represents the physiology of the knee. For a point at mid-thigh, the condyles are represented by a "wheel radius" of approximately 20 mm, and a bar length of 200 mm. For this case the path of the point on the thigh does not deviate from a circle by more than 1 mm for 90 degrees of rotation, and the center of the circle is within 1 cm of the moving center of rotation. See Figure 72.

A relatively straightforward technique can be applied to include the rotational information and eliminate the flaw. Since the path and the velocity along the path are known at each measured point, as well as the angular velocity, a Reuleaux center can be calculated for each point. The calculation proceeds by finding the best fit plane; then the best fit sphere to the relative motion data points subject to the constraint that the center lie on the plane. The the path translational velocity is determined next. This is the velocity at one point in the plane, tangent to the circular intersection of the sphere and plane. Next, the angular velocity normal to the plane is found at that same point. The instant center radius will be at a distance from the data point on the circular path,

equal to the path velocity magnitude divided by the angular velocity. The center will always lie on the radius from the data point to the center of the circle (sphere). This last constraint is chosen because the path has been determined to be very circular, and the center of rotation will always lie on a line normal to the path. Figure 73 shows the calculated trajectory of the Reuleaux centers of rotation for the pendulum used earlier.

(0.0198, 0.7760, -0.0344)
R= 0.3388 [meters]
URV=-0.0057, -0.0109, 0.9999
Frames: 100 to 630
TK4:AP1503.DAT:0
Seg# 1 wrt Seg# 0.   ~X/Y Plane
AP1501w/10HzFltr.DerivsFltrd

Reuleaux Center Path

Figure 73. Pendulum path
and Reuleaux centers path

Unfortunately, noise in the data produces centers that move excessively (20 to 50 cm) from point to point. The presence of noise and small measurement errors precluded the outright use of Reuleaux's method on a point-by-point basis; even with a smoothed, almost circular trajectory, the noise and small inaccuracies in the velocity data once again make the division of translational by rotational velocity highly variable. To demonstrate, Figures 74 and 75 are examples of free swinging knee motion. For this experiment the subject stood on one leg and flexed and extended the instrumented leg at the knee without putting weight on it. The plot shows the motion of a point on the shank (lower leg) moving with respect to the BCS on the thigh.

Figure 74. Reuleaux center of rotation of the knee

(-0.0096, -0.1536, 0.0122)
R= 0.2012 [meters]
URV=-0.3718, -0.2268, 0.9002
Frames: 1 to 630
TK4: AP0422.DAT:0
Seg# 2 wrt Seg# 3, ~X/Y Plane
AP0420w/15HzFltr,DerivsFiltred

10 mm

(0.0045, -0.1435, -0.0132)
R= 0.1674 [meters]
URV=-0.4466, -0.0958, 0.8896
Frames: 1 to 630
TK4:AP0477.DAT:0
Seg# 2 wrt Seg# 3.   ~X/Y Plane
AP0475w/10HzFltr,DerivsFiltred
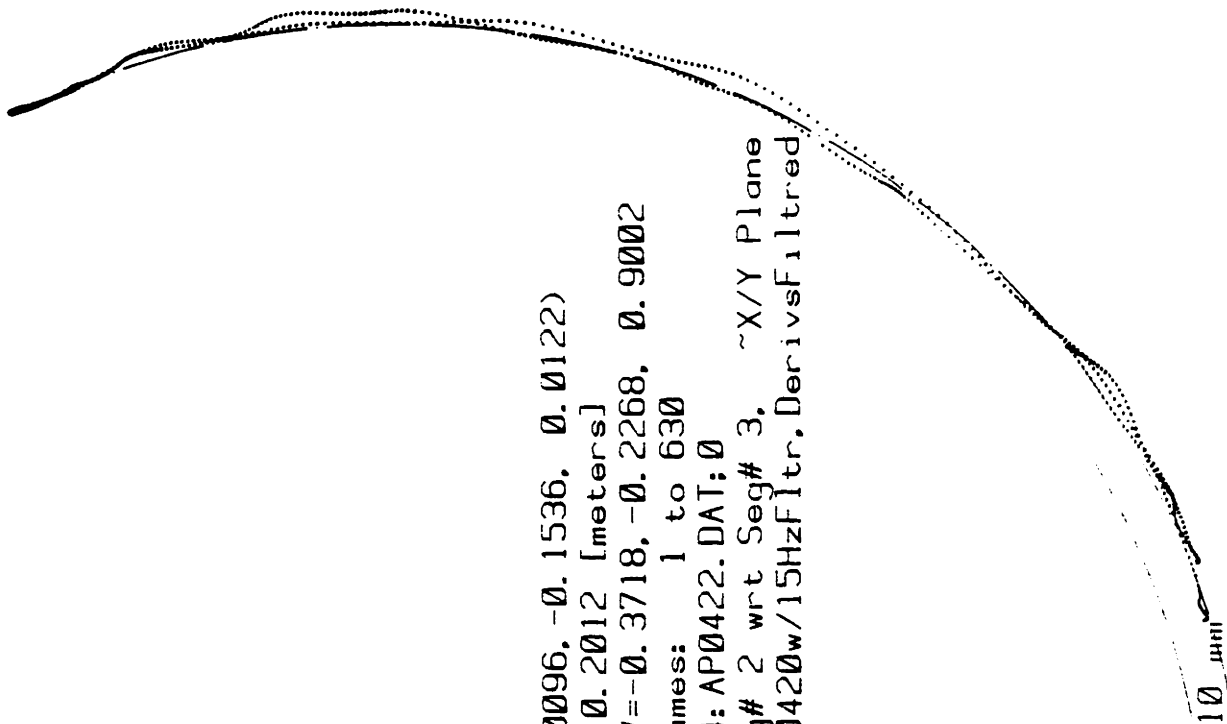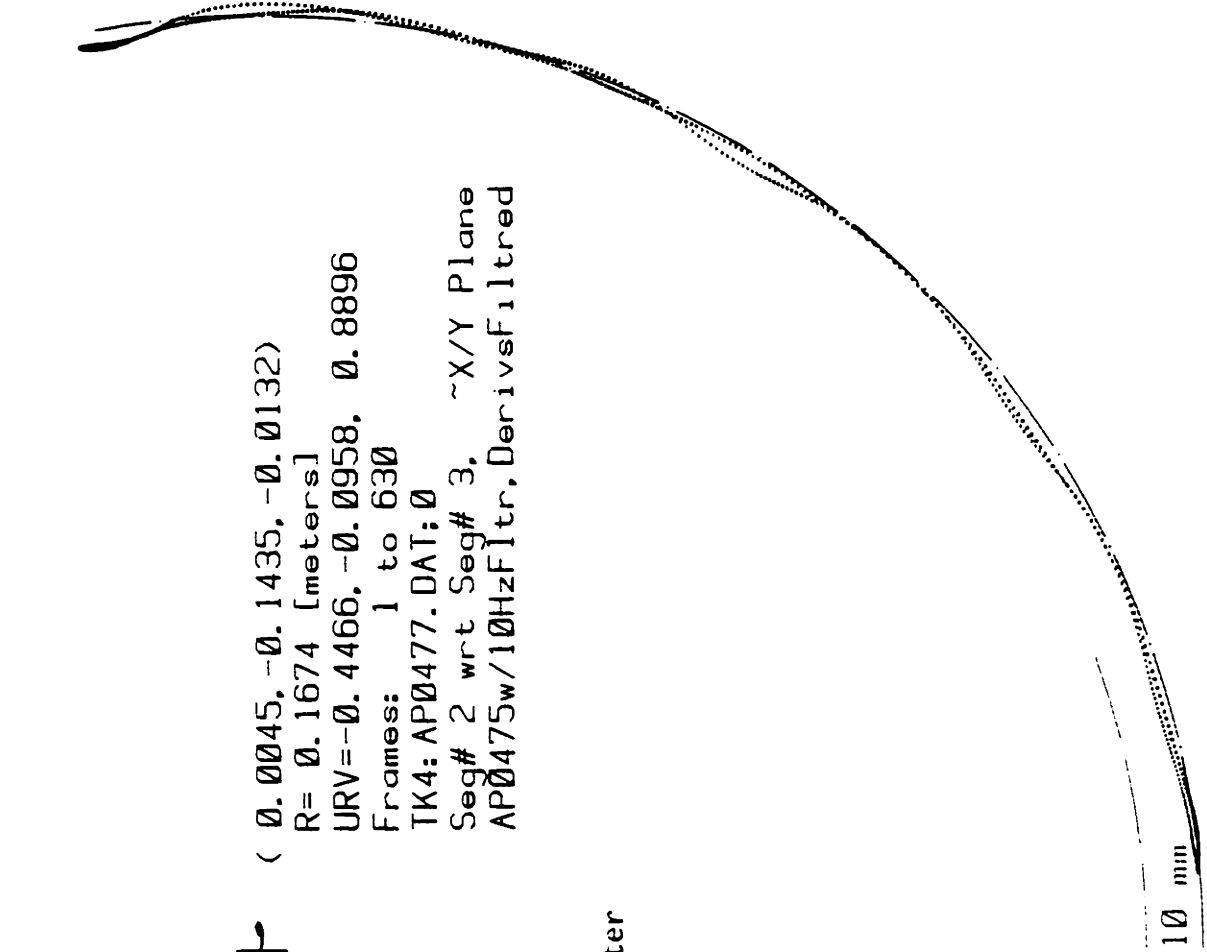
Figure 75. Reuleaux center
of rotation of the knee

Despite the highly circular path of the BCS origin, the center of rotation is found to move over large distances. This example is also confined to a plane; hence centers of rotation are used for clarity in describing the problems of Reuleaux's method.

To make the rolling wheel and bar (used earlier for demonstration) more closely resemble the human knee joint, let us include a variable amount of slip at the contact point between the wheel and the plane to examine how well the "path-only" technique will work for these cases, which include translation. For this discussion, the amount of slip is linearly related to the angle of wheel roll. Or described another way, for a constant wheel angular velocity, the slip velocity is constant. Positive slip is in the direction of rolling, making the center of the wheel move farther than it would have with pure rolling. Negative slip is opposite to the direction of rolling, making the center of the wheel move less far than it would have with pure rolling. Slip is represented in percent. Zero percent slip is pure rolling. Negative 100 percent slip is rotation about the center of the wheel. Negative 50 percent slip means the center of the wheel only translated half as far as it would have if there had been pure rolling. The distance the wheel center will translate is:

$$d=A(Slip+100)/100$$

Where A is the angle the wheel turns [in radians], and Slip is in percent. The paths of points on the bar are no longer cycloids, but again for physiologically reasonable data, the paths are still exceedingly circular.

To demonstrate, once again the 20-mm radius wheel and the 200-mm long bar were used to represent the knee. If the femoral condyles were circular and the cruciate ligaments were tight, there would be -100 percent slip; the center of the circular condyles would remain fixed with respect to the tibial plateau; and a circular path at the end of the bar would be generated. The condyles are not circular, so the path described is not a circle. This fact is the source of much interest, both here and elsewhere, in the trajectory of the axis of rotation at the knee. How much slip is present at the knee? Certainly the range is less than 0 percent to a maximum of -200 percent. That range includes displacement of the femur with respect to the tibia of +/- 31-mm [a total possible range of motion of 2.5 inches]. For this discussion, let us assume that possible slip is bounded by the range -50 percent to -150 percent. For a radius of 20-mm and a range of motion of 90 degrees, this implies that the range of center motion (if the circular condyles are rolling on a flat plane) is +/- 15 mm. Different rates of slip affect the path of a point on the "bar"; however, just as in the cycloids, the path shape is still very circular. See Figures 76, 77, and 78. In fact, the slip changes the

center of rotation position, but the best fit circle center and the Reuleaux center are always within 1 cm of each other for this geometry, and the path does not deviate from the best circle by more than 1 mm. For smaller amounts of slip (should it turn out upon a very detailed analysis that the knee kinematics contain slip closer to 100 percent, as suggested by Kettelkamp 1970, page 779 [87]), the best circle center is closer to the Reuleaux center.
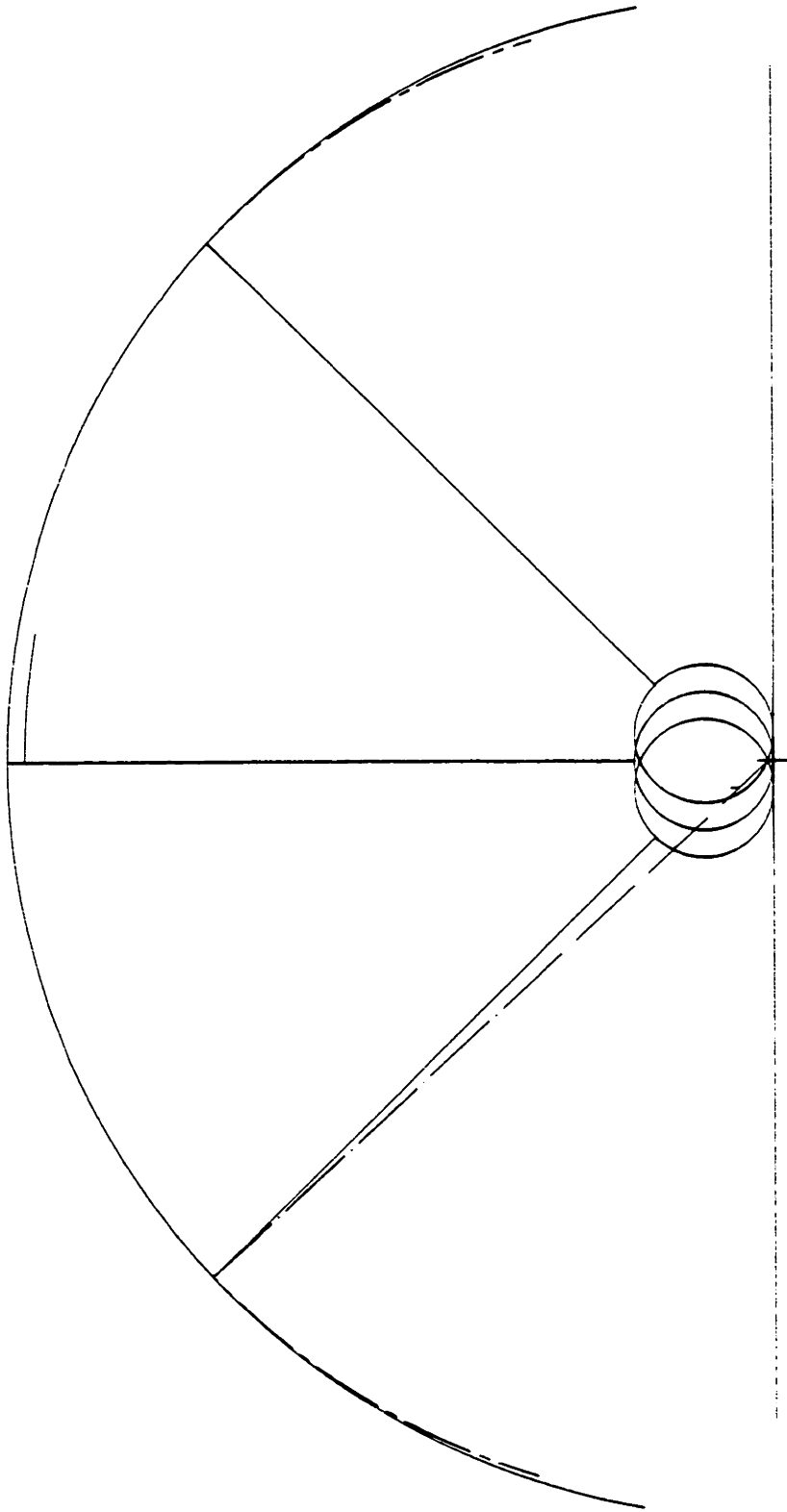
Figure 76. Cycloid with -50% slip and the best fit circle
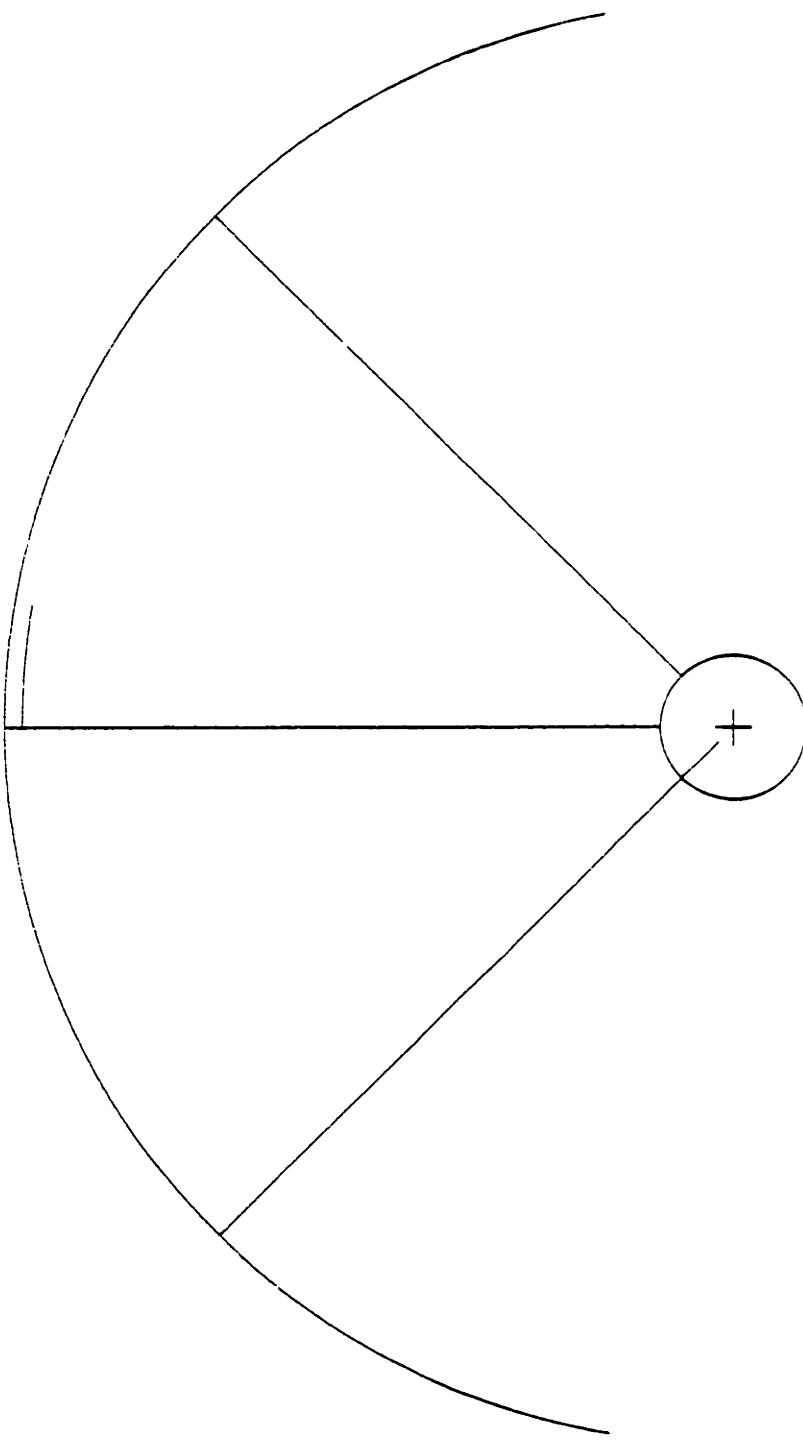
Figure 77. Cycloid with -100% slip and the best fit circle
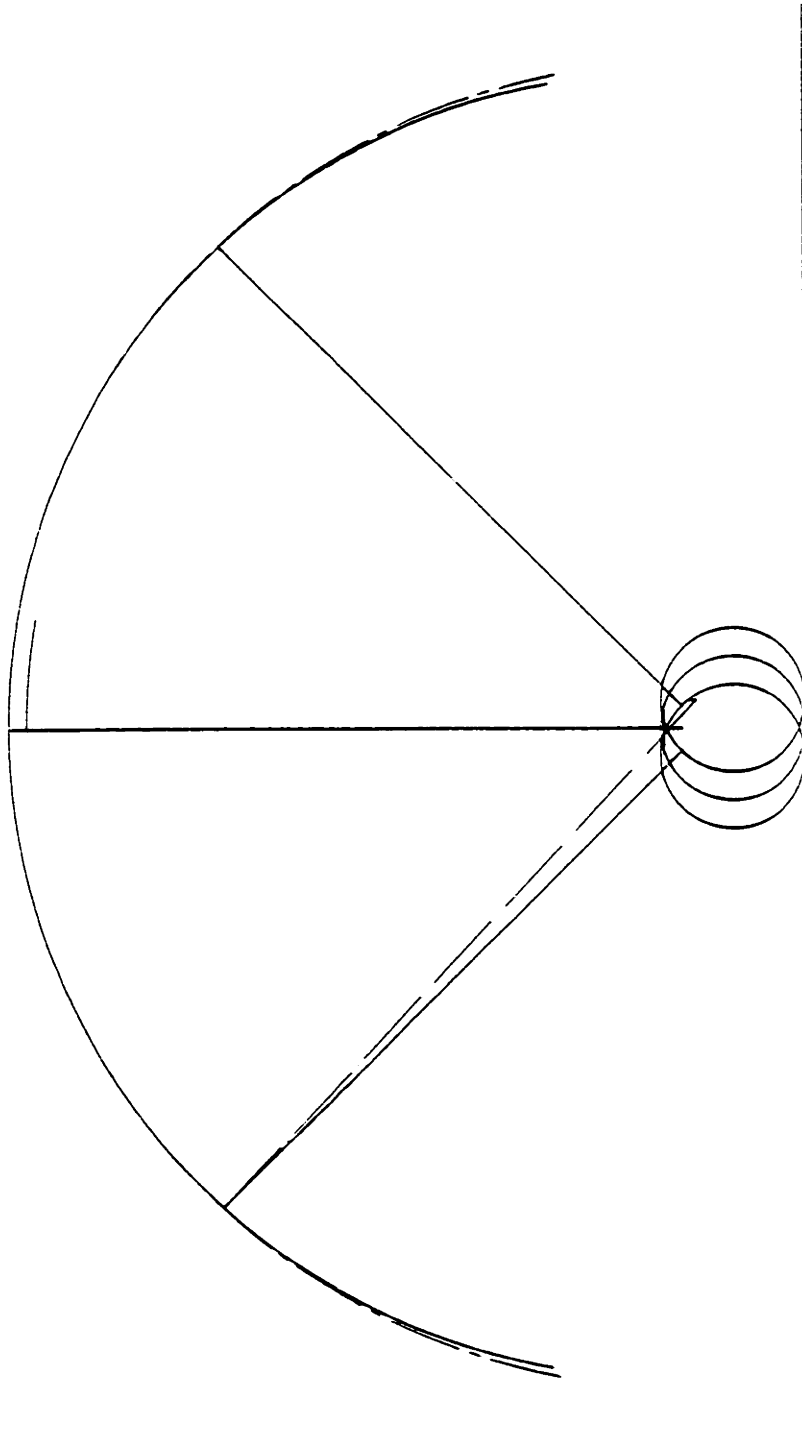
Figure 78. Cycloid with -150% slip and the best fit circle

To sum up Momentary Axes of Rotation: For data accumulated with the subject 3 meters from the cameras (so that the viewing volume will contain one full stride), despite the very high quality of position data, the best that the axes of joint rotation can be found is within 1 cm for the assumed kinematics (slip within -50 percent to -150 percent of rolling, see earlier disucssion) of the lower extremity joints. This is still a considerable improvement, however, over the conventional skin-mounted, body-prominence-marker approach where the joint centers are chosen a-priori by the experimenter.

The following six plots describe the axes calculated for one subject. The data were accumulated in a series of unloaded experiments where the subject flexed and extended each joint separately while standing on the opposite leg. This portion of the experimental protocol will be discussed more fully in a later section. The purpose of those experiments was to locate joint axes to be used in the geometrical definition of each link for the dynamic analysis. The spatial relationship of each body coordinate system (BCS) and the global coordinate system (GCS) are shown in Figure 79 along with the standard numbering of segments used throughout this thesis. Each of the six following axis plots shows the motion of one BCS origin with respect to an adjacent link's origin.
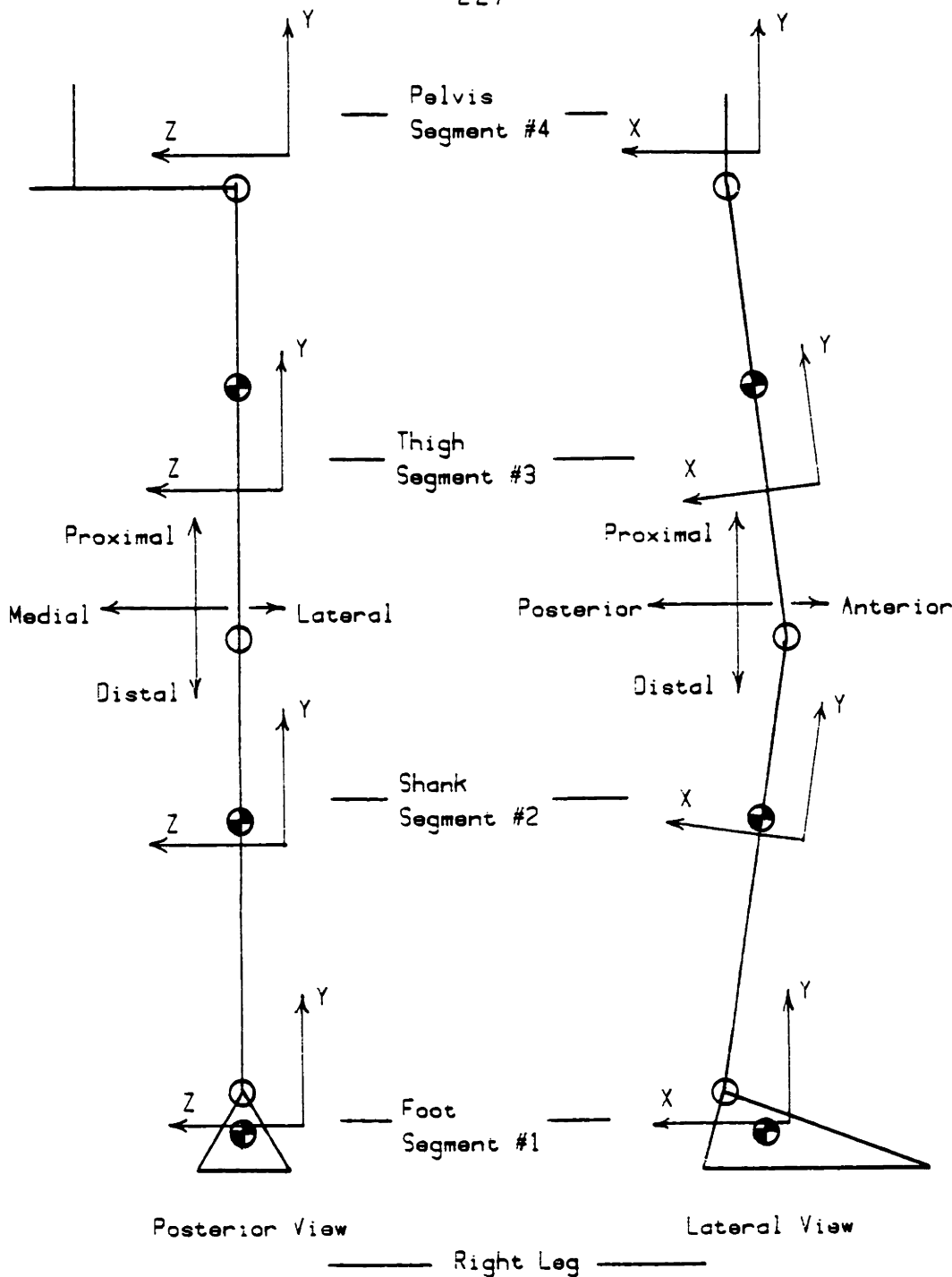
e. g. Flexion at the knee is represented by a positive rotation (right handed coordinate systems) of the shank (#2) with respect to the thigh (#3) (mostly about the Z axis).

Figure 79. Segment Body Coordinate System arrangement and numbering

(-0.0118, -0.2569, 0.0162)
R= 0.0541 [meters]
URV=-0.0756, 0.0661, 0.9949
Frames: 1 to 630
TK4: AP0417.DAT; 0
Seg# 1 wrt Seg# 2. ~X/Y Plane
AP0415/w3HzFltr (AnkleFlexion)

10 mm

Dorsiflexion

Plantarflexion
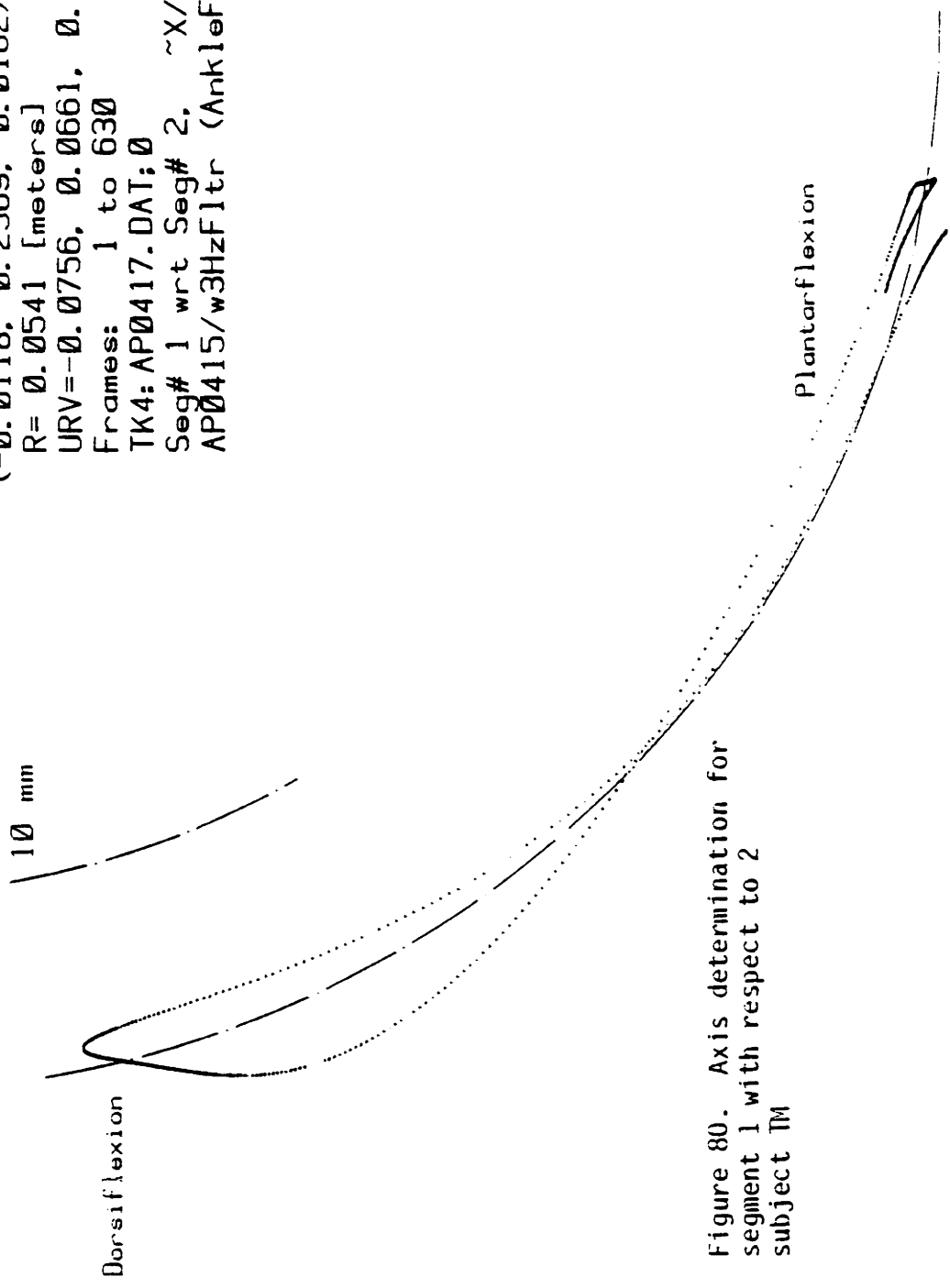
Figure 80. Axis determination for segment 1 with respect to 2 subject TM

Plantarflexion

Dorsiflexion

10 mm
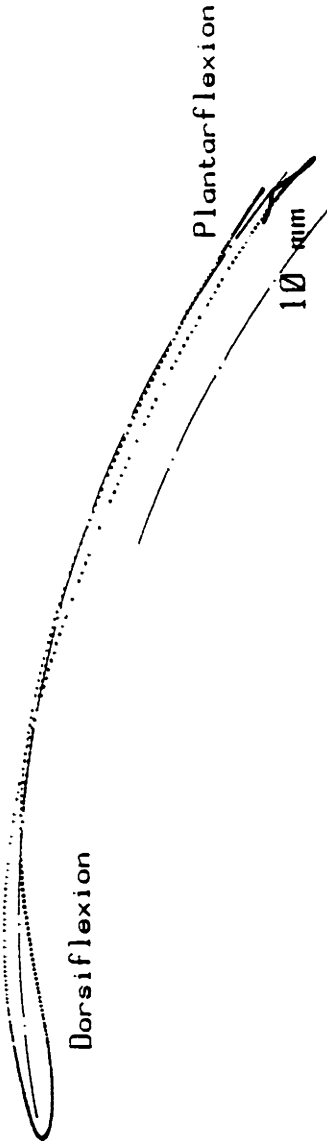
(0.0762, 0.0676, -0.0147)
R= 0.2113 [meters]
URV=-0.0125, -0.1017, 0.9947
Frames: 1 to 630
TK4: AP0417.DAT:0
Seg# 2 wrt Seg# 1, ~X/Y Plane
AP0415/w3HzFltr (AnkleFlexion)

Figure 81. Axis determination
segment 2 with respect to 1
subject TM

Flexion

Extension

(.. 0.0097, -0.1532, 0.0108)
R= 0.2025 [meters]
URV=-0.3774, -0.2227, 0.8989
Frames: 1 to 630
TK4: AP0421.DAT:0
Seg# 2 wrt Seg# 3. ~X/Y Plane
AP0420w/10HzFltr (KneeFlexion)

10 mm
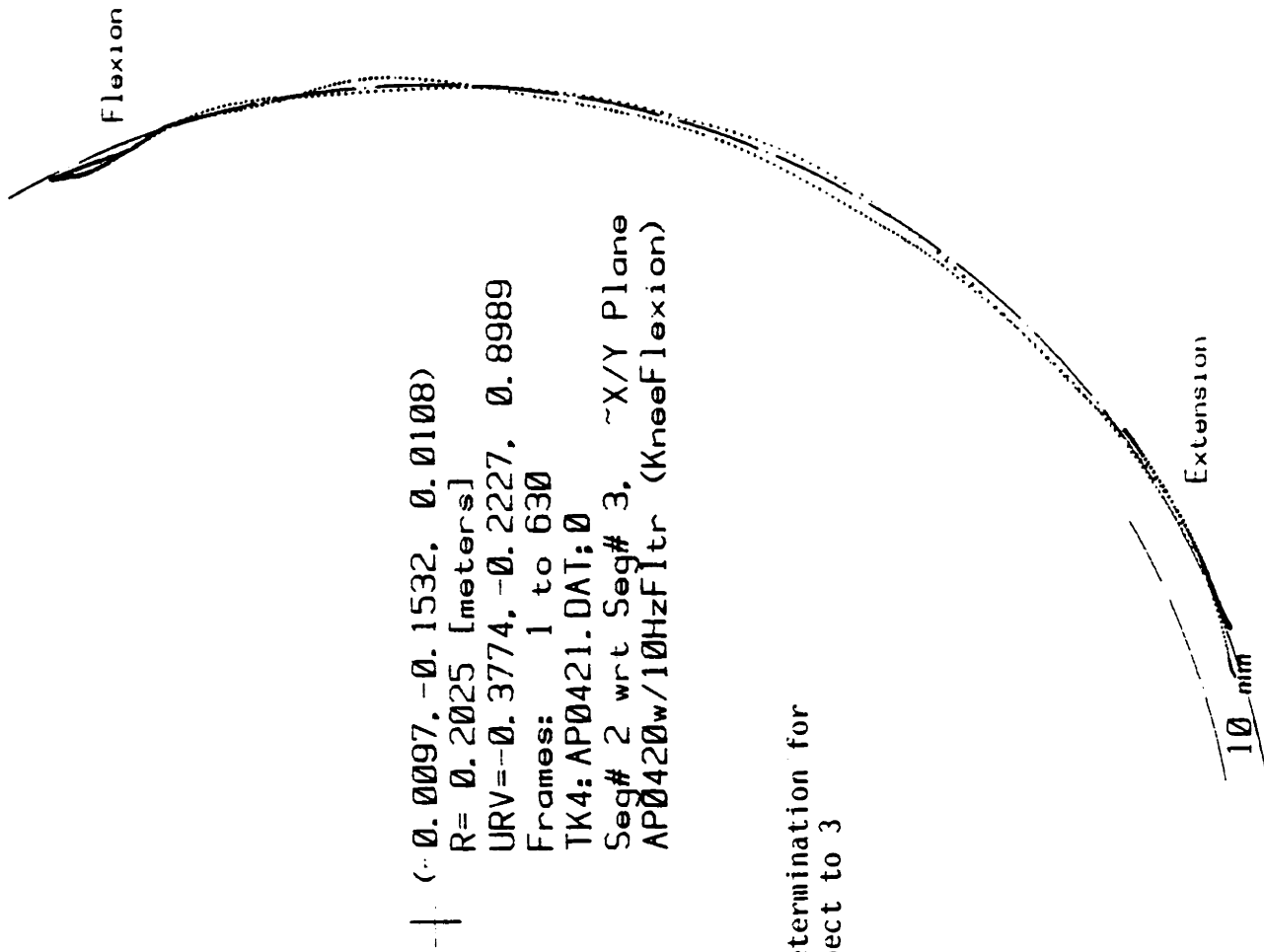
Figure 82. Axis determination for
segment 2 with respect to 3
subject TM

Extension

Flexion

10 mm
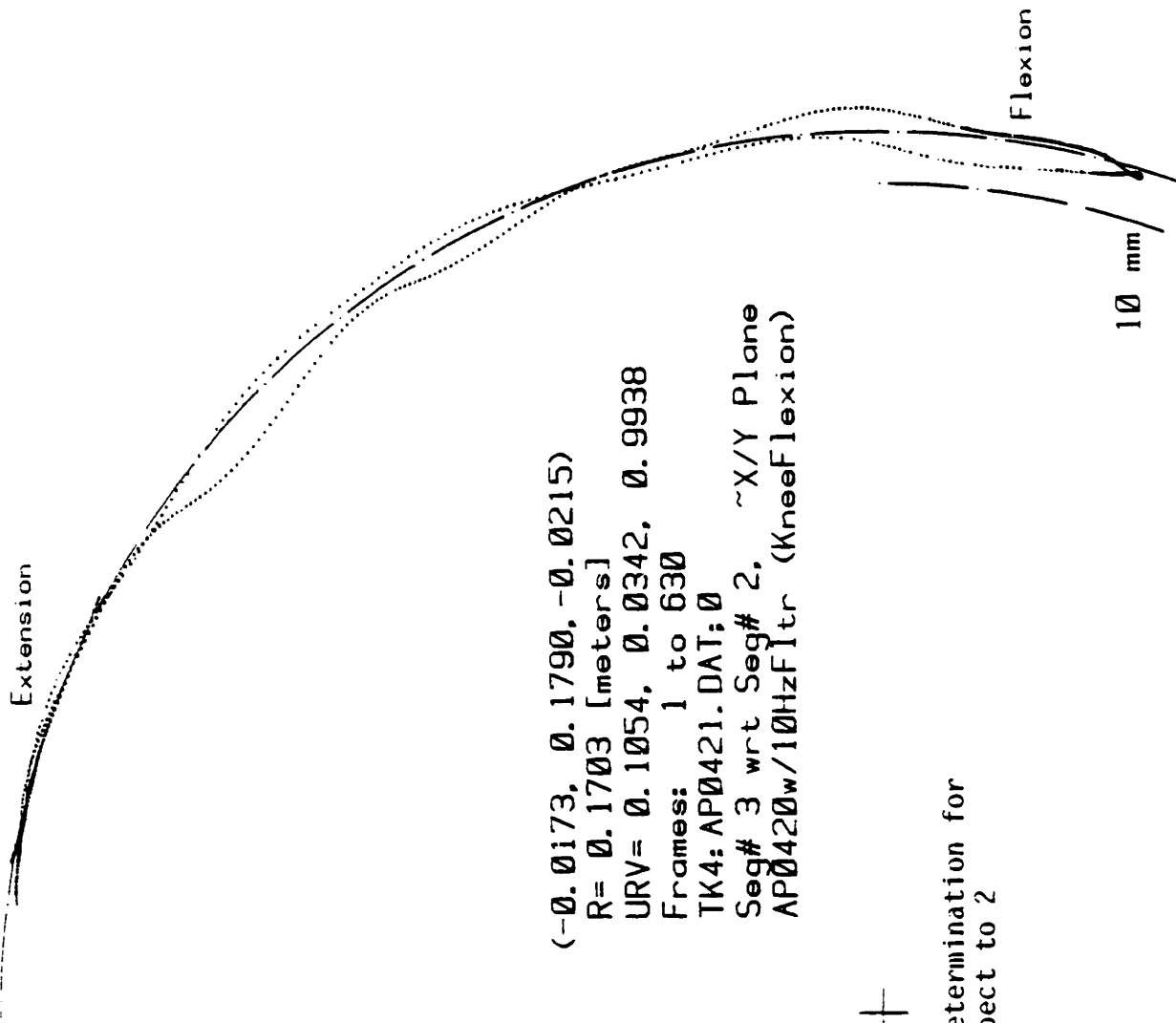
(-0.0173, 0.1790, -0.0215)
R= 0.1703 [meters]
URV= 0.1054, 0.0342, 0.9938
Frames: 1 to 630
TK4:AP0421.DAT:0
Seg# 3 wrt Seg# 2, ~X/Y Plane
AP0420w/10HzFltr (KneeFlexion)

Figure 83. Axis determination for
segment 3 with respect to 2
subject TM

(-0.0532, -0.2095, -0.0108)
R= 0.2260 [meters]
URV=-0.0794, -0.0311, 0.9964
Frames: 1 to 630
TK4: AP0427.DAT;0
Seg# 3 wrt Seg# 4, ~X/Y Plane
AP0425 w/5HzFltr (Hip Flexion)

10 mm

Flexion

Extension
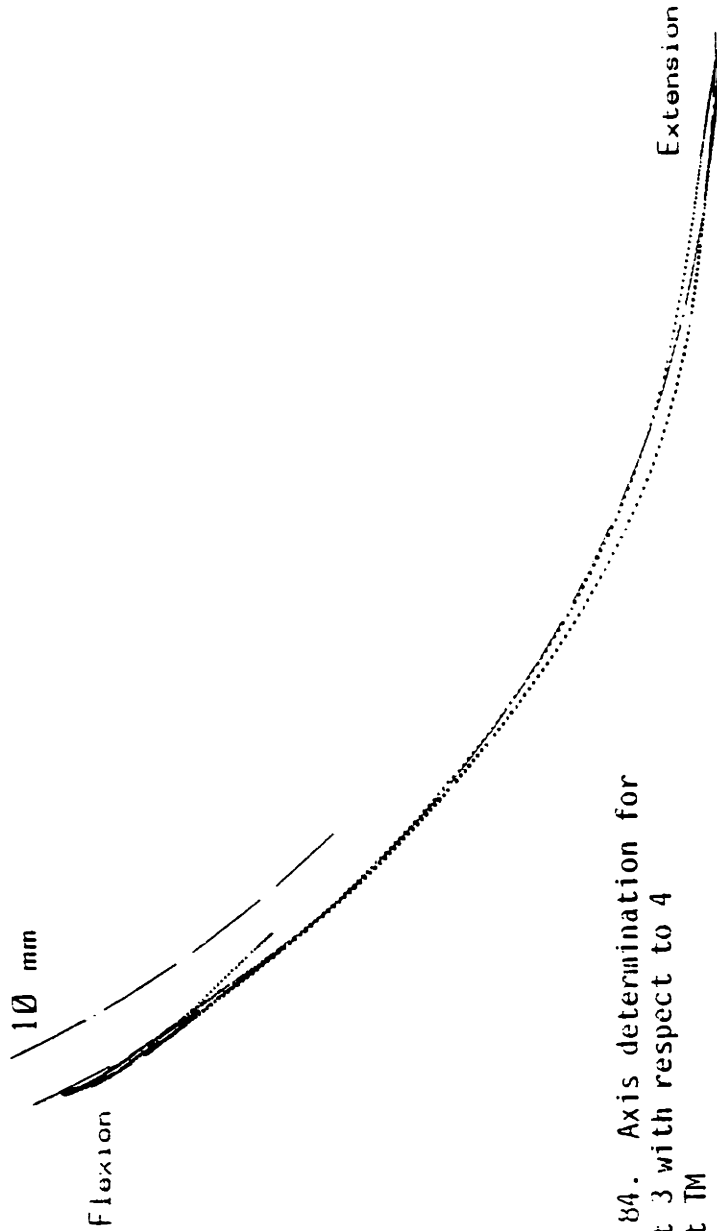
Figure 84. Axis determination for
segment 3 with respect to 4
subject TM

Extension

Flexion

10 mm

(-0.0350, 0.3633, 0.0660)
R= 0.0715 [meters]
URV=-0.1806,-0.1928, 0.9645
Frames: 1 to 630
TK4: AP0427.DAT:0
Seg# 4 wrt Seg# 3. ~X/Y Plane
AP0425 w/5HzFltr (Hip Flexion)
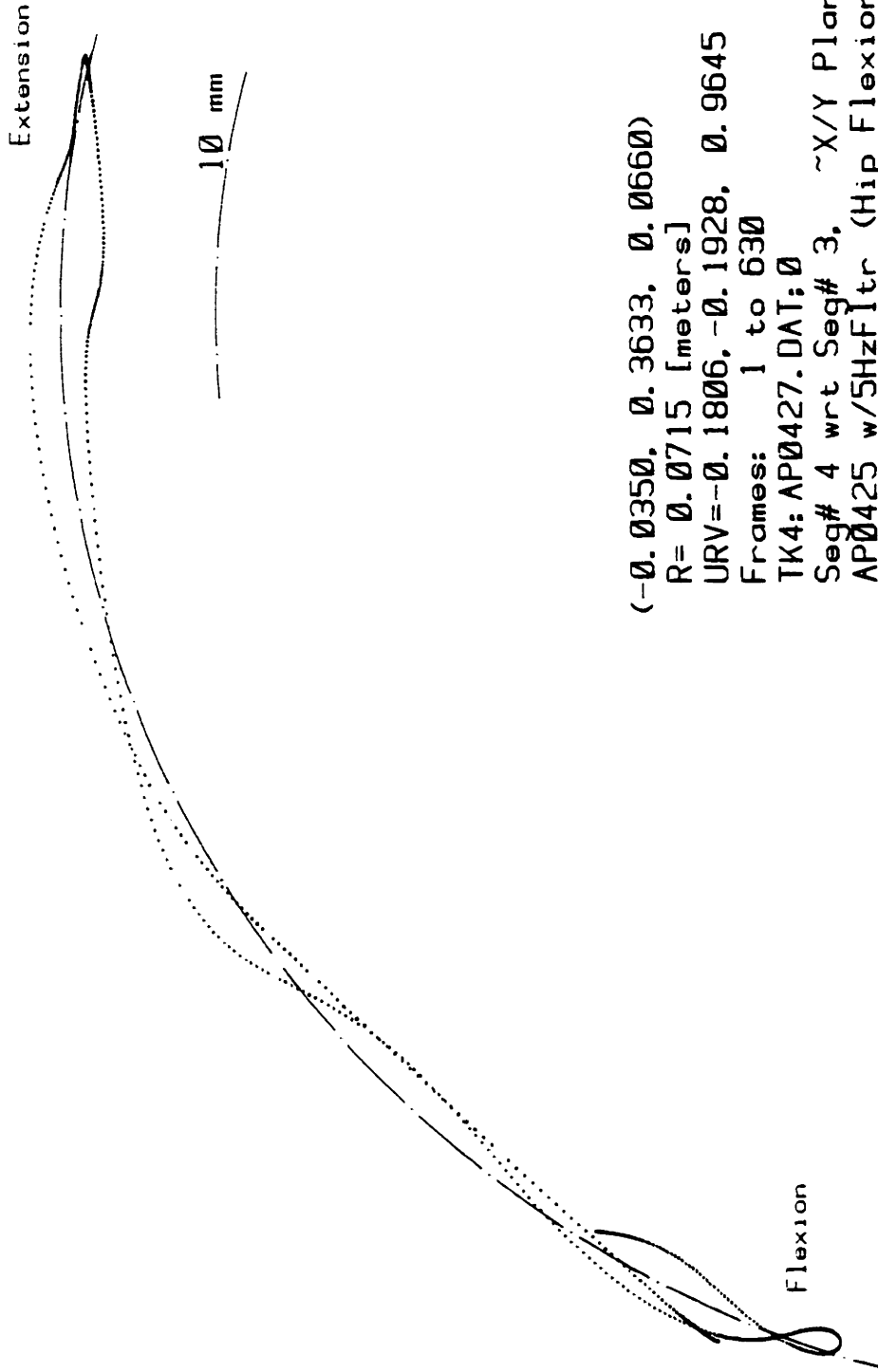
Figure 85. Axis determination for
segment 4 with respect to 3
subject TM

To determine all of the geometry necessary for the dynamic estimation, each axis location with respect to both articulating links must be found; thus the knee axis location with respect to the thigh is shown in Figure 82, and the same axis location with respect to the shank is shown in Figure 83.

MARKER MOUNTING APPROACH

As discussed in Chapter 2 the human body does not lend itself to precise kinematic analysis. The links are not rigid; the skin, soft tissue, and therefore the center of mass move with respect to the underlying skeleton. But rigid mounting of segment markers as with bone pins cannot be used for ethical and experimental reasons. Every effort must be expended to achieve markers (in this case arrays of LEDs) which will remain fixed with respect to the skeleton. If perfect kinematic fidelity were so attained, the only unaccounted variables would be the moving centers of mass. This will be discussed in a later section.

Four marker supports were designed and constructed to mount arrays of seven or eight LEDs on human, lower extremity body segments, one each for the foot, shank (lower leg), thigh, and pelvis. The goal was to instrument fully one leg through one stride of gait (one full swing phase,

and one full stance phase). Photograph 10 depicts the foot segment and the corresponding array of LEDs is shown in Figure 86.

Beyond the physical mounting of the marker array on the subject's body segment, the configuration of the LED array itself is an important design consideration. Several criteria guide the number and shape of an array. First, more LEDs on a single segment produce higher quality position data, a consequence of the averaging effect on both the measurement noise and small residual errors. Secondly, more LEDs produce a vernier effect increasing the effective spatial resolution beyond the resolution for a single LED alone. And finally, more than three LEDs in an array provide for the occlusion of some (so long as three are still in view) while assuring valid data for that segment. The shape of the array influences the rotational resolution of the Body Coordinate System (BCS) associated with each segment; the further separated the LEDs, the higher the rotational resolution.

Clearly, there are competing design requirements, to keep the segments as small and light as possible, and not exceed the maximum of thirty LEDs available to the Selspot I. The compromise solutions presented here all achieve 1 mm and better than 15 milli-radians of positional resolution while encumbering the subject minimally. Part of the

success of these segments is due to the very small size and

mass of the individual LED and the ease with which they can

be mounted directly into the plexiglas fixtures.
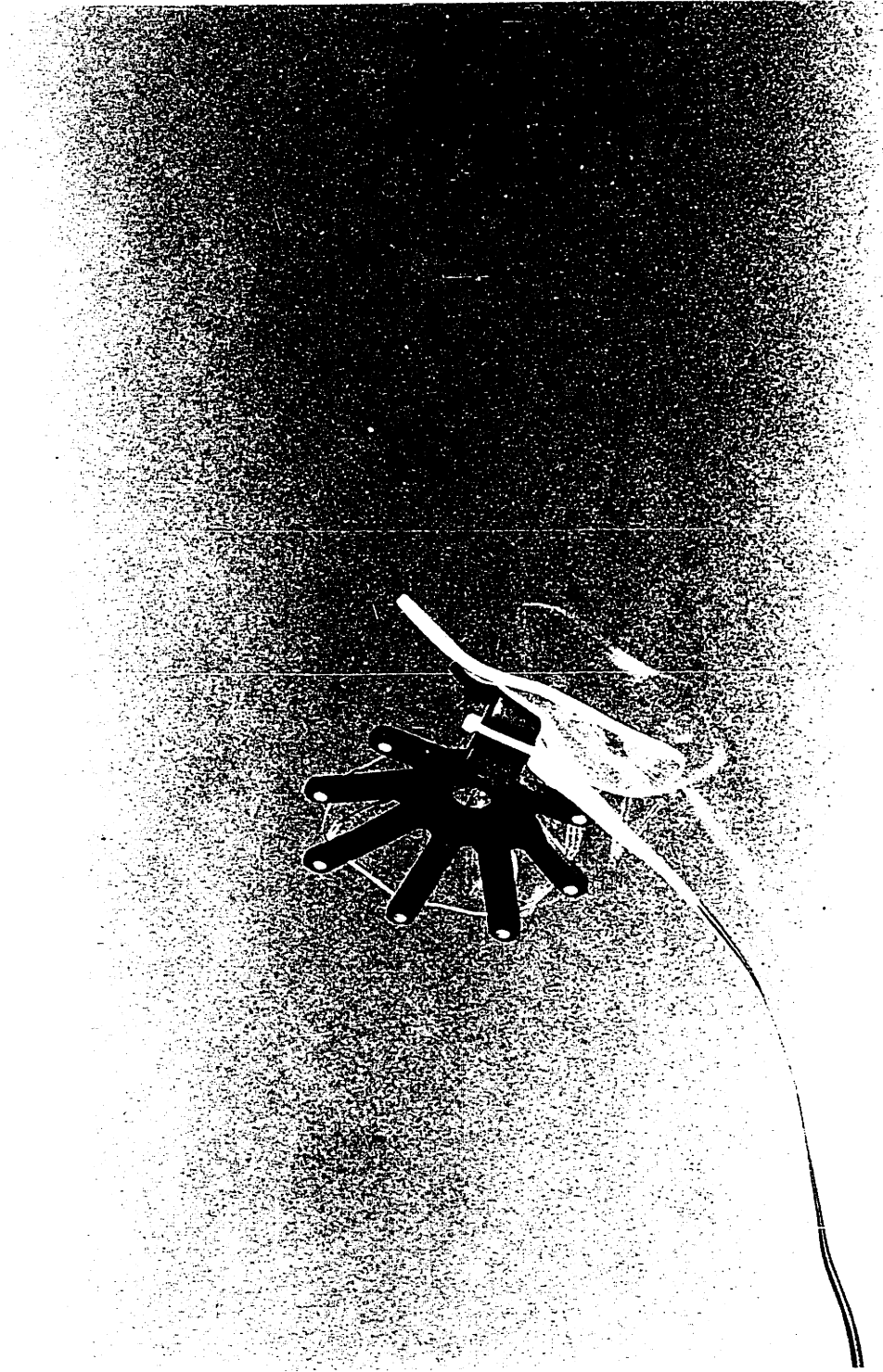
Photograph 10. The foot segment
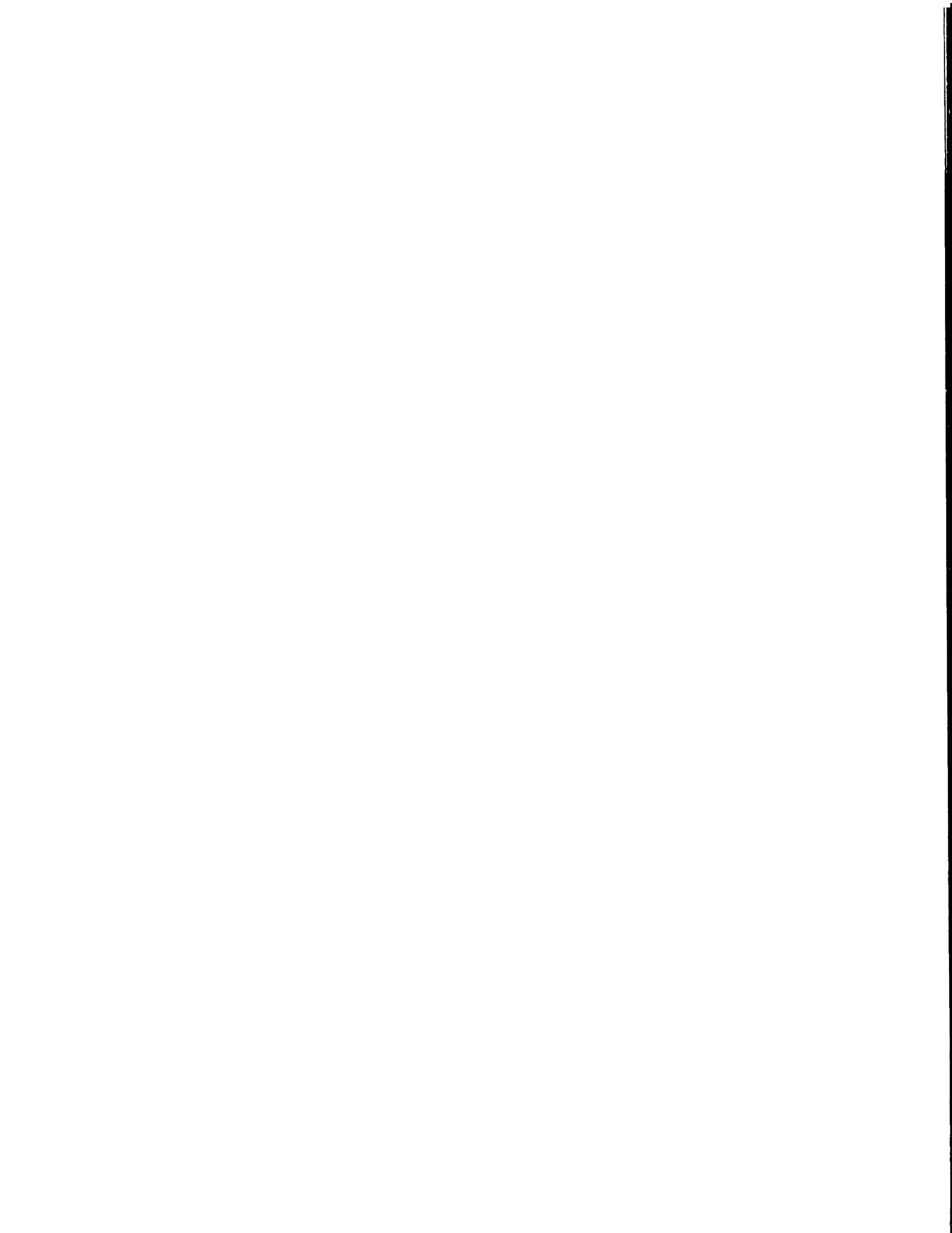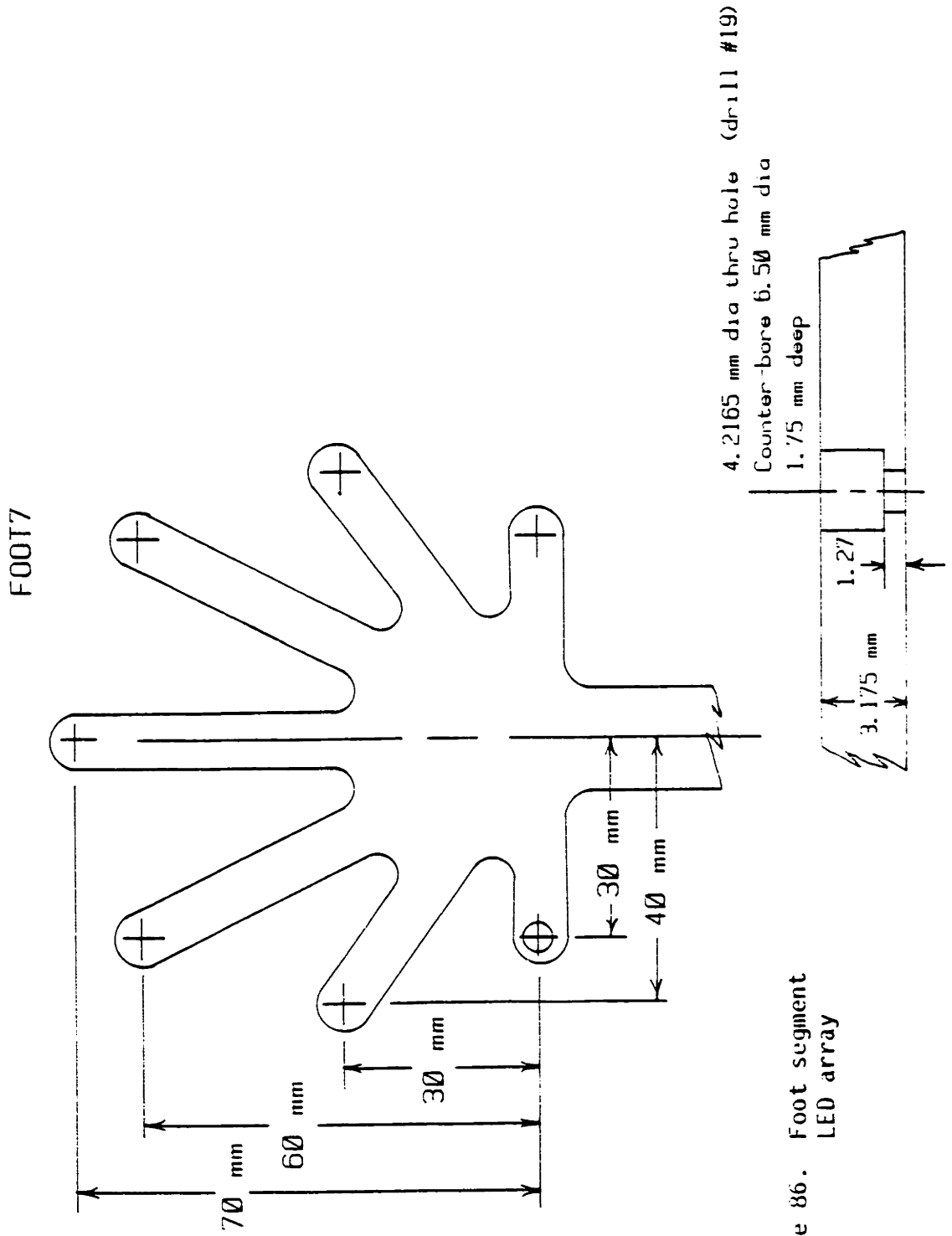
Figure 86.   Foot segment
LED array

```
Name      Nchn   Nseg      Description
FOOT7      7.     1.        Foot Seg 7 Fingers w/ V194P

LED segment coordinate data (meters)
Segment   Channel          X                Y                Z
  1         1            0.0300           0.0000           0.0000
  1         2            0.0400           0.0300           0.0000
  1         3            0.0300           0.0600           0.0000
  1         4            0.0000           0.0700           0.0000
  1         5           -0.0300           0.0600           0.0000
  1         6           -0.0400           0.0300           0.0000
  1         7           -0.0300           0.0000           0.0000
```

Table 5   Foot segment LED array specifications

Since the human foot is not a rigid body and no single bone defines its skeleton, the segment was designed to maintain fixation with the major mass of the foot, which also constitutes the most rigid of all its elements. Attachment was achieved by means of a plexiglas plate 3-mm thick molded to the shape of the lateral-superior aspect of the foot. An elastic strap wrapped in a "figure-8" pattern around the foot and ankle holds the segment in place. Photograph 14 is of the entire instrumented leg.

Shank fixation was achieved by mounting a 3-mm thick plate 25 mm by 100 mm to the medial border of the tibia with an elasic strap 80-mm wide. The LEDs were attached to this plate by means of a curved bar that reached around in front of the tibia to the lateral side of the shank. Photograph 11 and Figure 87 show the tibial segment and the array of LEDs. Notice that the LED array is the same as for the foot, oriented differently.

SHANK7

70 mm

60 mm

30 mm

30 mm

40 mm
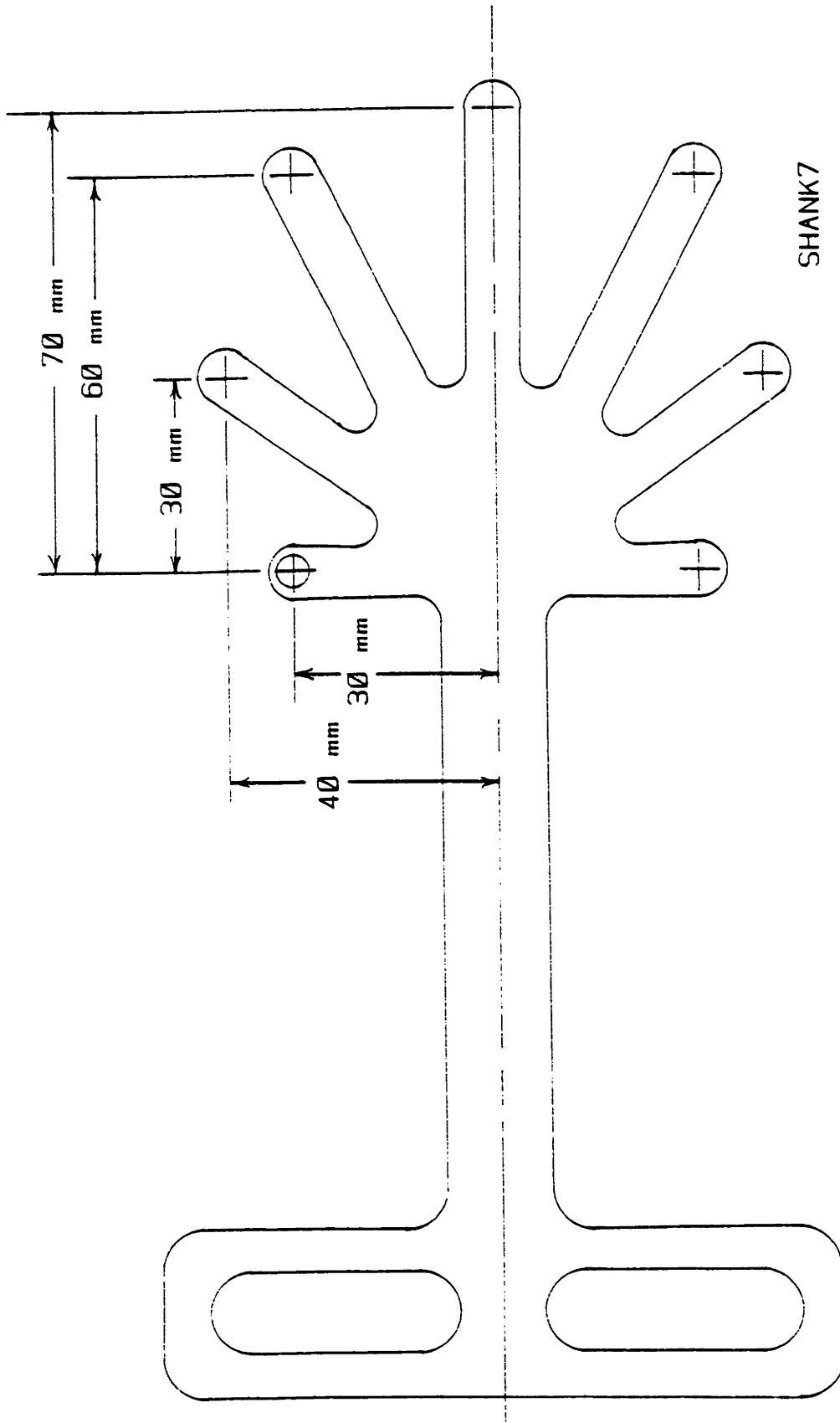
Figure 87.   Shank segment LED array
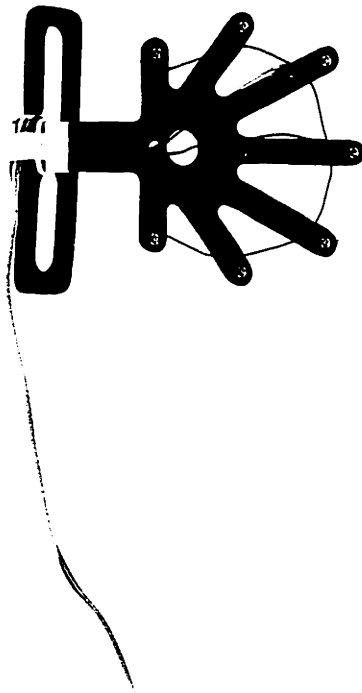
Photograph 11. The tibial segment

```
Name     Nchn   Nseg      Description
SHANK7    7.     1.       7 Finger array w/ V194P LEDs

LED segment coordinate data (meters)
Segment   Channel          X              Y                Z
  1          1           0.0000        -0.0300         0.0000
  1          2           0.0300        -0.0400         0.0000
  1          3           0.0600        -0.0300         0.0000
  1          4           0.0700         0.0000         0.0000
  1          5           0.0600         0.0300         0.0000
  1          6           0.0300         0.0400         0.0000
  1          7           0.0000         0.0300         0.0000
```

Table 6   Shank segment LED array specifications

Thigh fixation proved to be the most difficult. Much palpation indicated that the two, most useful, bony prominences least affected by underlying soft tissue motion were the medial epicondyle of the femur and the greater trochanter, at least in ectomorphs. Once again plexiglas was used to construct the segment, now 6 mm thick because of the large size and need for rigidity. Photograph 12 is of the segment, and Figure 88 is the array of LEDs. Notice that there are eight LEDs on the thigh in contrast to seven on each of the two more distal segments. The Selspot I (as mentioned earlier) has a maximum of thirty LEDs; an even distribution among four segments would place seven on each with two excess. One each of the extra two was placed on the thigh and pelvis segments given their susceptability to occlusion by the free-swinging arm during gait. One of the primary goals of the experimental protocol was to enable the subject to walk as normally as possible, unimpeded by the experimental apparatus. Thus if the subject were walking in a direction approximately parallel to a line joining the cameras, his arm would pass in front of the proximal two segments (thigh and pelvis) during a portion of the stride. For most gait kinematics systems (with one or two markers per segment), this is intolerable. TRACK's ability, however , to eliminate markers that disappear from view and still produce high quality results (if data from three or more LEDs on each segment persists) guaranteed that by simply

allocating excess LEDs to the segments likely to be
obscured, useful data would be acquired without hampering
the subject by enforcing unnatural arm positions during gait
thus the larger arrays of eight LEDs on the thigh and the
pelvic segments.
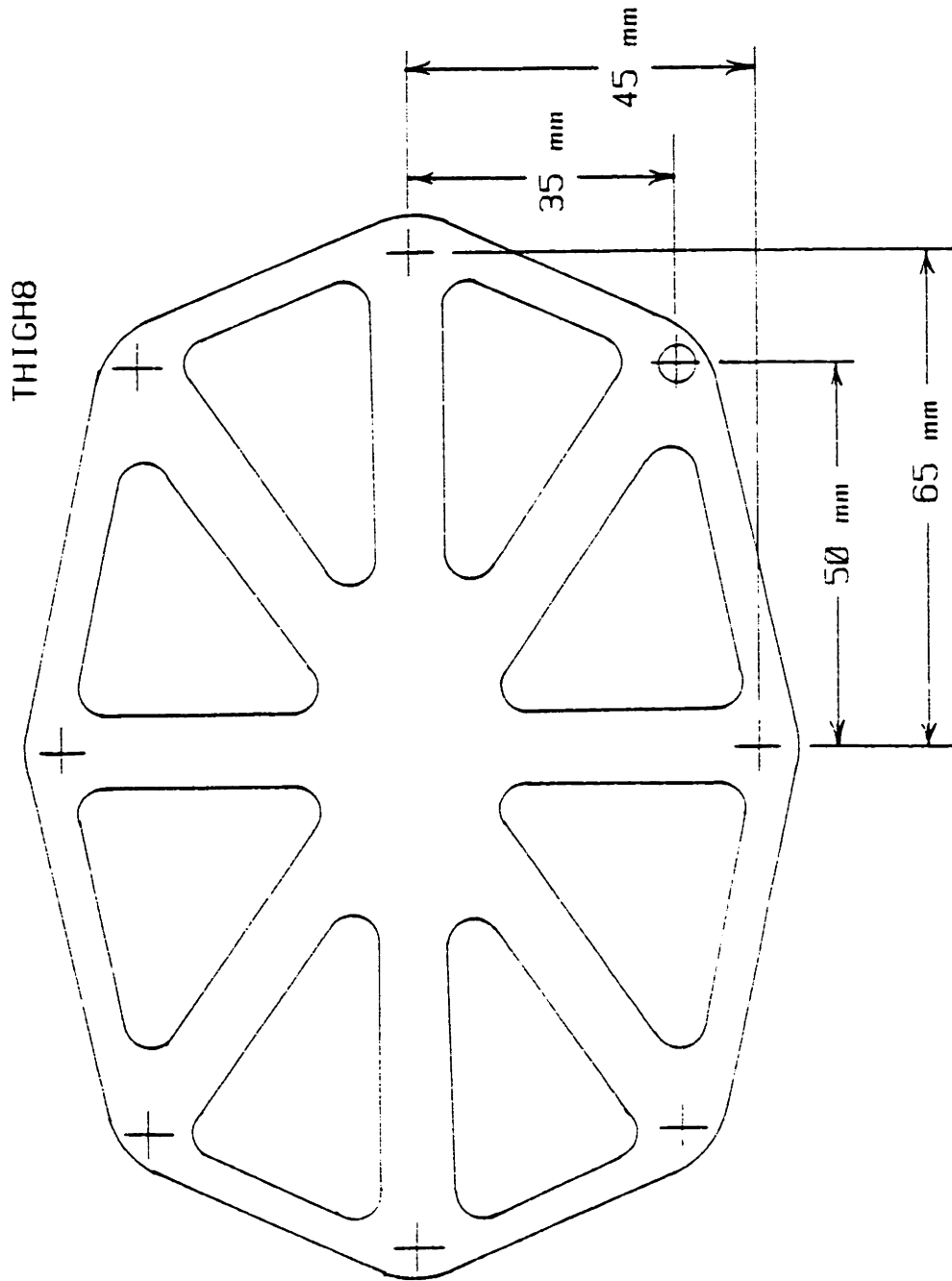
Figure 88. LED array for the thigh segment
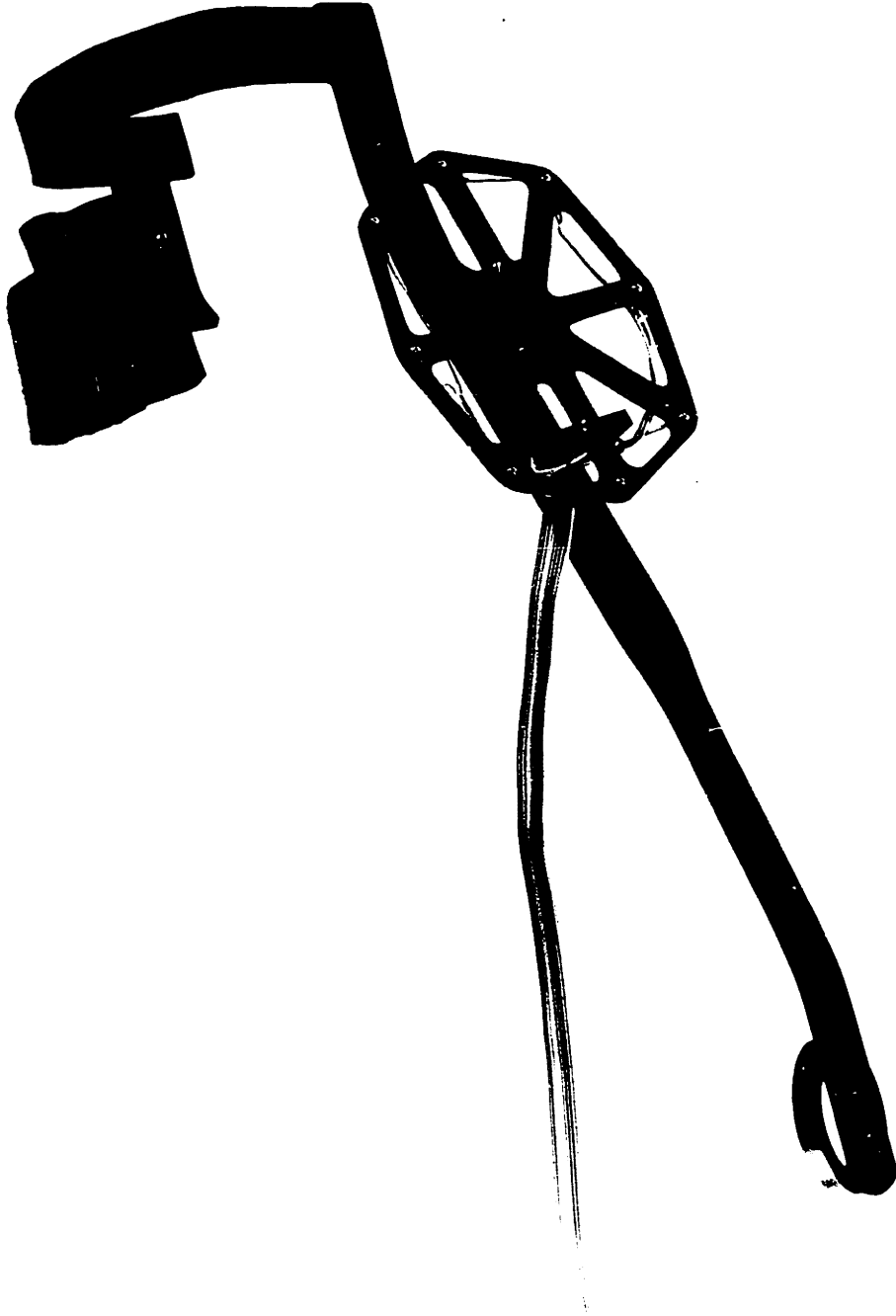
Photograph 12. The thigh segment

```
Name      Nchn   Nseg      Description
THIGH8     8.     1.       Octagon segment w/ V194P LEDs

LED segment coordinate data (meters)
Segment   Channel          X               Y               Z
1         1               -0.0350         -0.0500         0.0000
1         2               -0.0450          0.0000         0.0000
1         3               -0.0350          0.0500         0.0000
1         4                0.0000          0.0650         0.0000
1         5                0.0350          0.0500         0.0000
1         6                0.0450          0.0000         0.0000
1         7                0.0350         -0.0500         0.0000
1         8                0.0000         -0.0650         0.0000
```

Table 7 Thigh segment LED array specifications

The final body segment to be instrumentated was the pelvis. The inspiration for the design came from Professor T. McMahon of Harvard University. Fixation is achieved by a curved segment that reaches around the instrumented side of the pelvis from the iliac crest to the sacrum. A notch in the segment locates on the crest, while a raised section sits on the sacrum. A flexible section of 1.5-mm thick plexiglas reaches around from the sacrum toward the other iliac crest, where another notch at that crest completes the segment. The ends of the segment at the anterior of the pelvis are held together by means of an elastic strap. This elastic also holds securely the three fixation points (two iliac crests and the sacrum). See Photograph 13 and Figure 89.
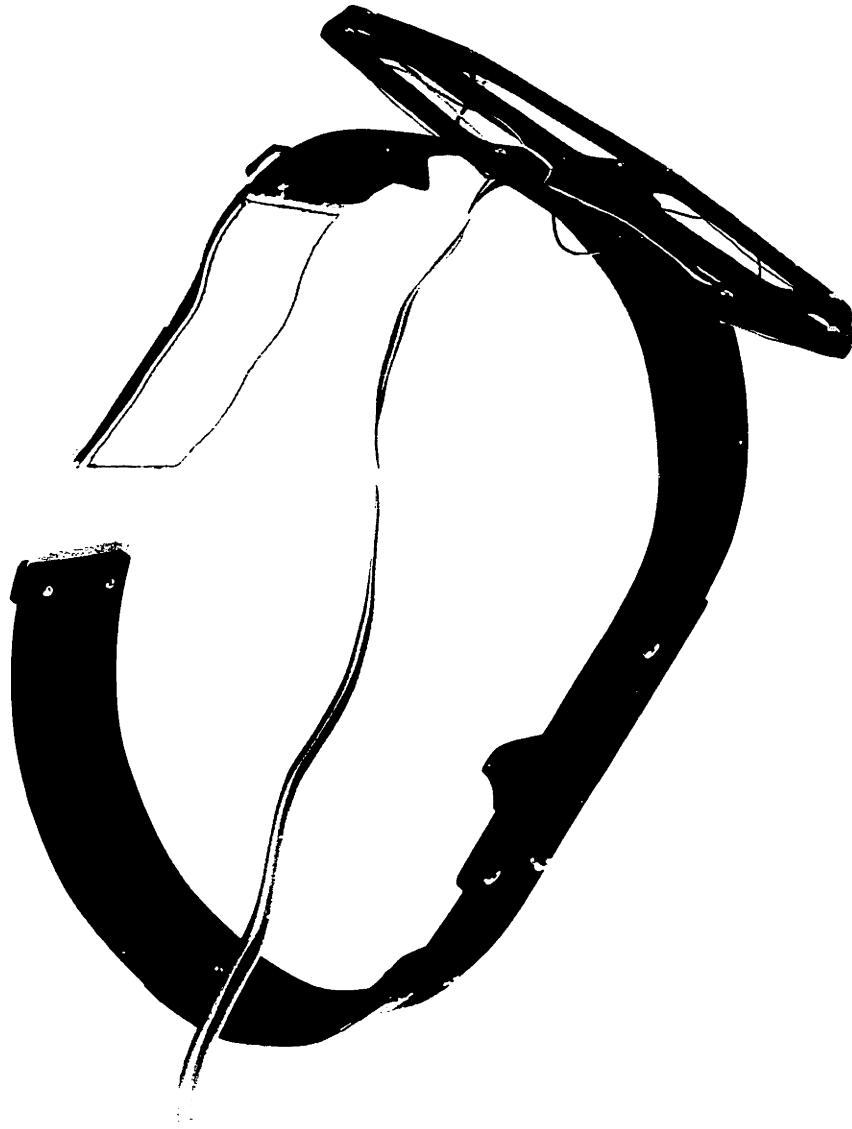
Photograph 13.   The pelvic segment

PELVI8
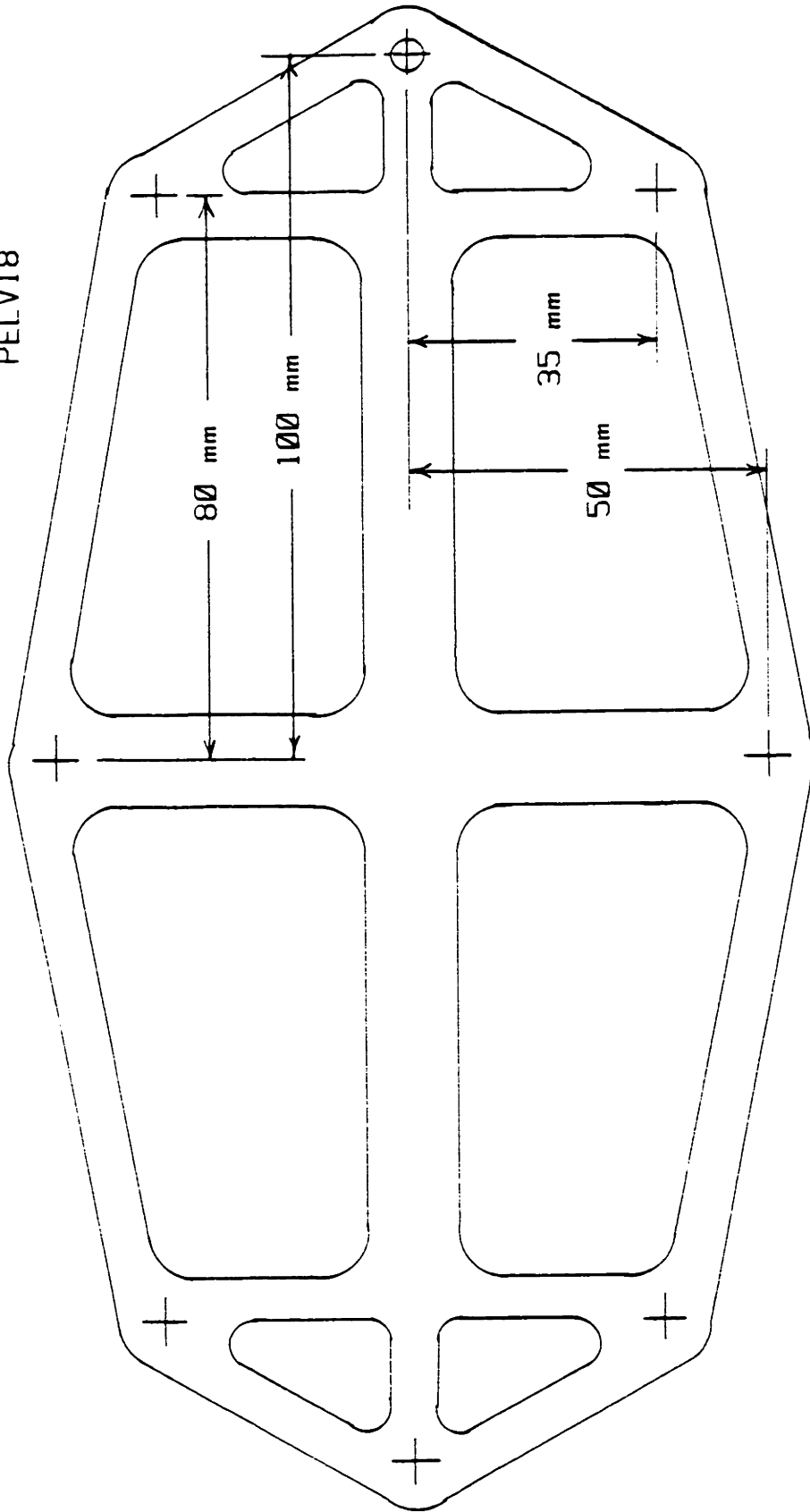
4.2165 mm dia thru hole (drill #19)
Counter-bore 6.50 mm dia
1.75 mm deep
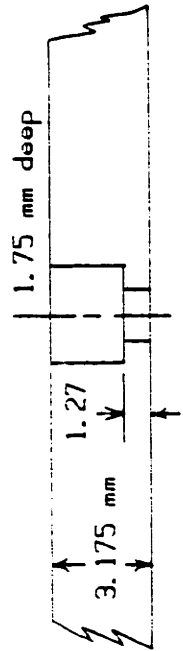
Figure 89. LED array for the pelvic segment

```
Name      Nchn   Nseg        Description
PELVI8     8.     1.         Large Pelvis Segmnt w/ V194P

LED segment coordinate data (meters)
Segment   Channel         X                Y                Z
  1          1          0.0000           0.0500           0.0000
  1          2         -0.0800           0.0350           0.0000
  1          3         -0.1000           0.0000           0.0000
  1          4         -0.0800          -0.0350           0.0000
  1          5          0.0000          -0.0500           0.0000
  1          6          0.0800          -0.0350           0.0000
  1          7          0.0990           0.0000           0.0000
  1          8          0.0800           0.0350           0.0000
```

Table 8 Pelvic segment LED array specifications

Notice that the large size of the LED array, anterior to posterior, insures that arm swing will not completely obscure the segment.

Photograph 14.   The instrumented leg
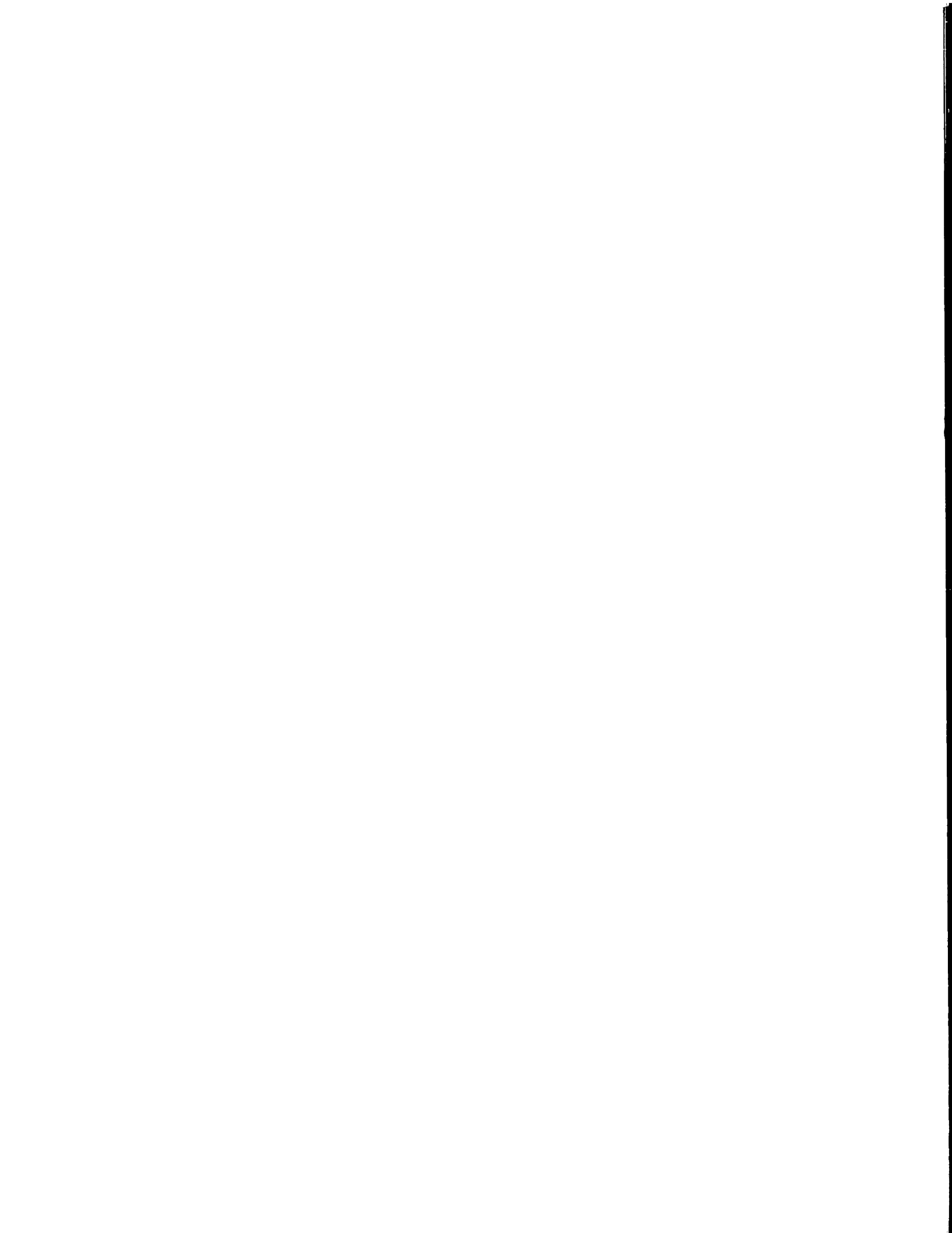
Photograph 15.   The lateral aspect of an instrumented subject

Photograph 16. The posterior aspect
of an instrumented subject

The marker mounting system as described achieves substantially stable mounting of the LEDs with respect to the skeleton; however, it is not perfect. To determine the extent and frequency content of segmental soft tissue movement, the following experiment was conducted. A subject stood stationary on the forceplatform in the middle of the viewing volume, rose up on his toes and then dropped abruptly to his heels. This experimental protocol was developed to embrace the highest possible accelerations experienced in gait, both in magnitude and direction, without introducing the complexities of walking. Figure 90 shows the unfiltered, vertical force measured at the floor during the rise and fall. Figure 91 is the region just after heel strike with the force axis expanded. The subject's weight was 700 Newtons.
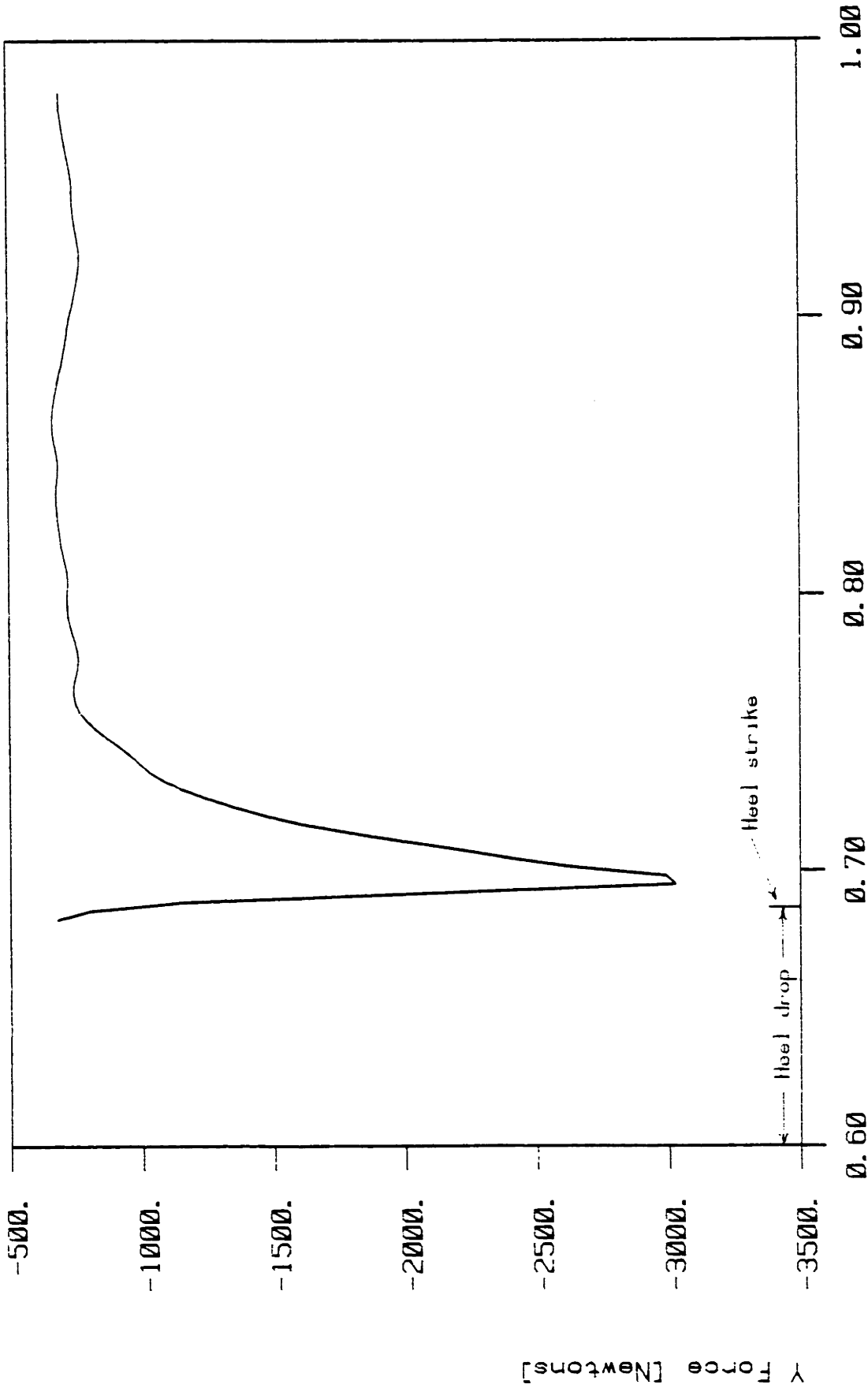
Figure 90. Unfiltered vertical force record of heel strike following heel rise for soft tissue motion evaluation
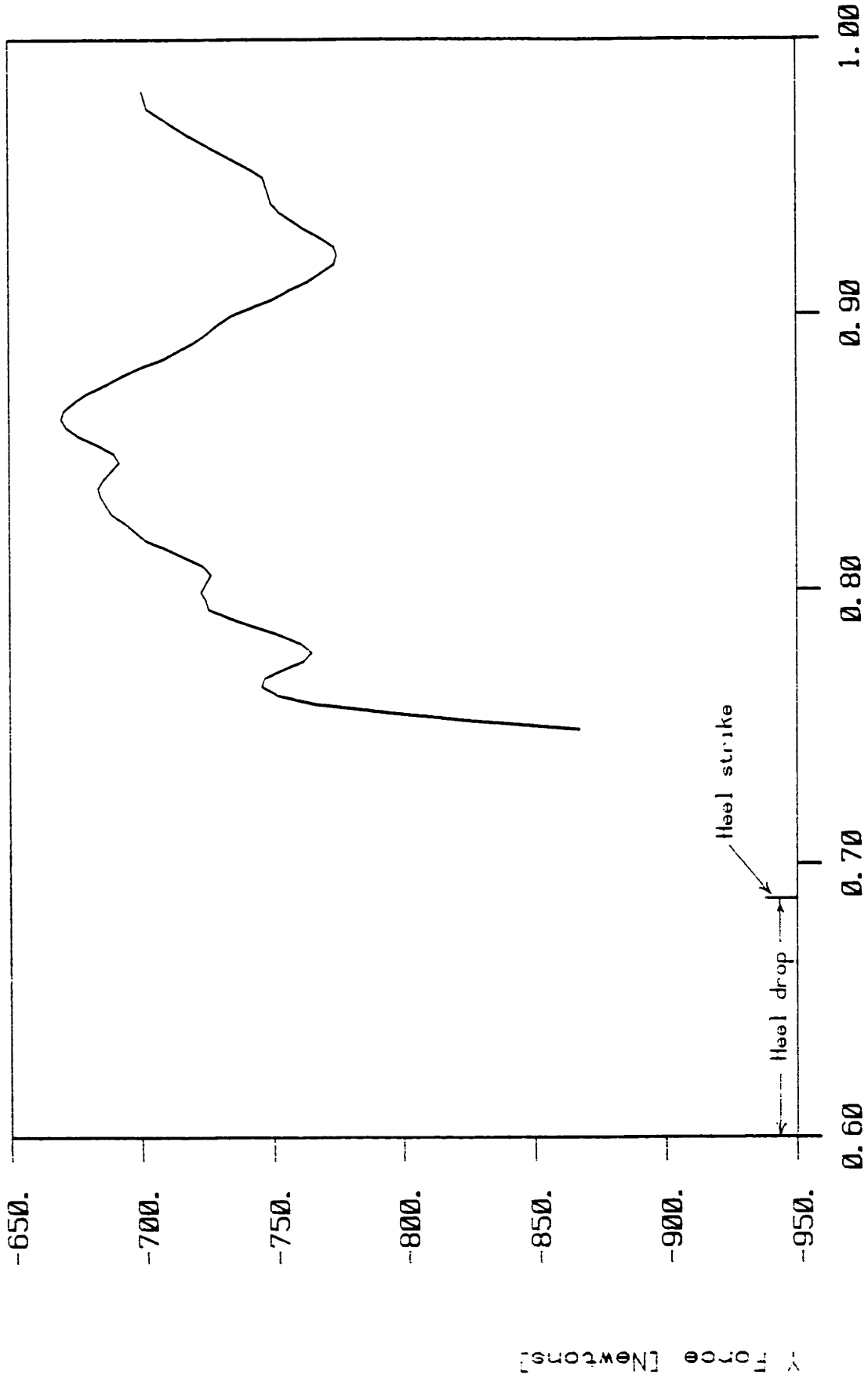
Figure 91. Expanded vertical scale (force) region from figure 90

Note the oscillations in vertical force of approximately 50-newtons amplitude, at a frequency of about 20 Hz. These are due primarily to the soft tissue, vertical oscillation (together with their respective LED segments), in combination with the whole body mass bouncing on soft tissue under the skeleton at the foot and on the springiness of the bones. While this experiment may not be the worst case loading applied to the lower extremity to excite soft tissue motion, it does represent a situation worse than that of gait. The peak vertical force generated at heel contact is more than three times that seen during walking.

The unfiltered, vertical excursion of the BCS associated with the foot is shown in Figure 92, and that of the thigh in Figure 93. The same signals subsequent to the application of a 25-Hz, low-pass, 6th-order, 2-pass Butterworth filter are shown compositely in Figure 94 together with the data from the pelvis and the shank. Figure 94 also includes the excitation data, the foot-floor force as recorded by the forceplate.
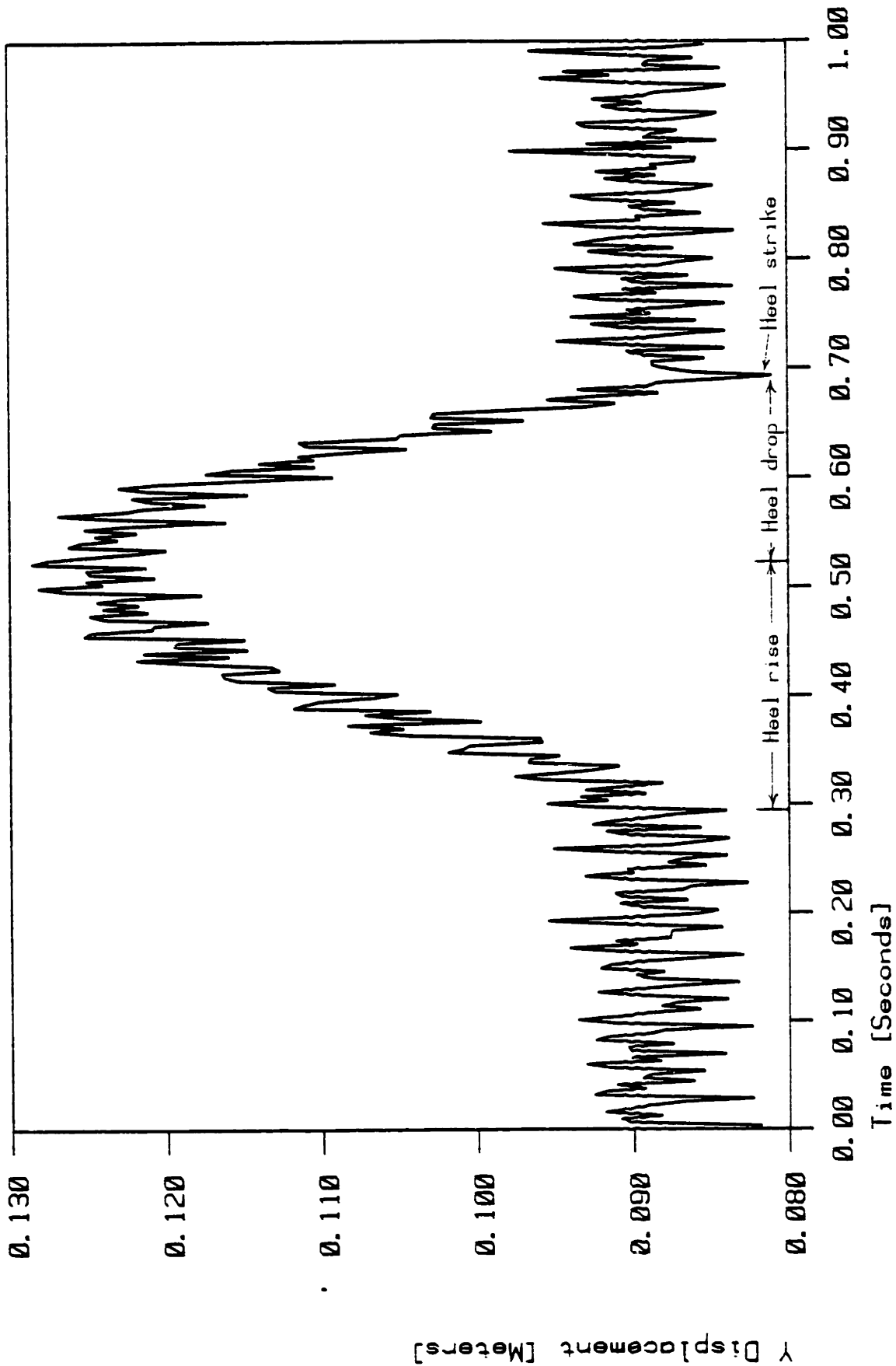
Figure 92. Unfiltered vertical excursion of the foot BCS
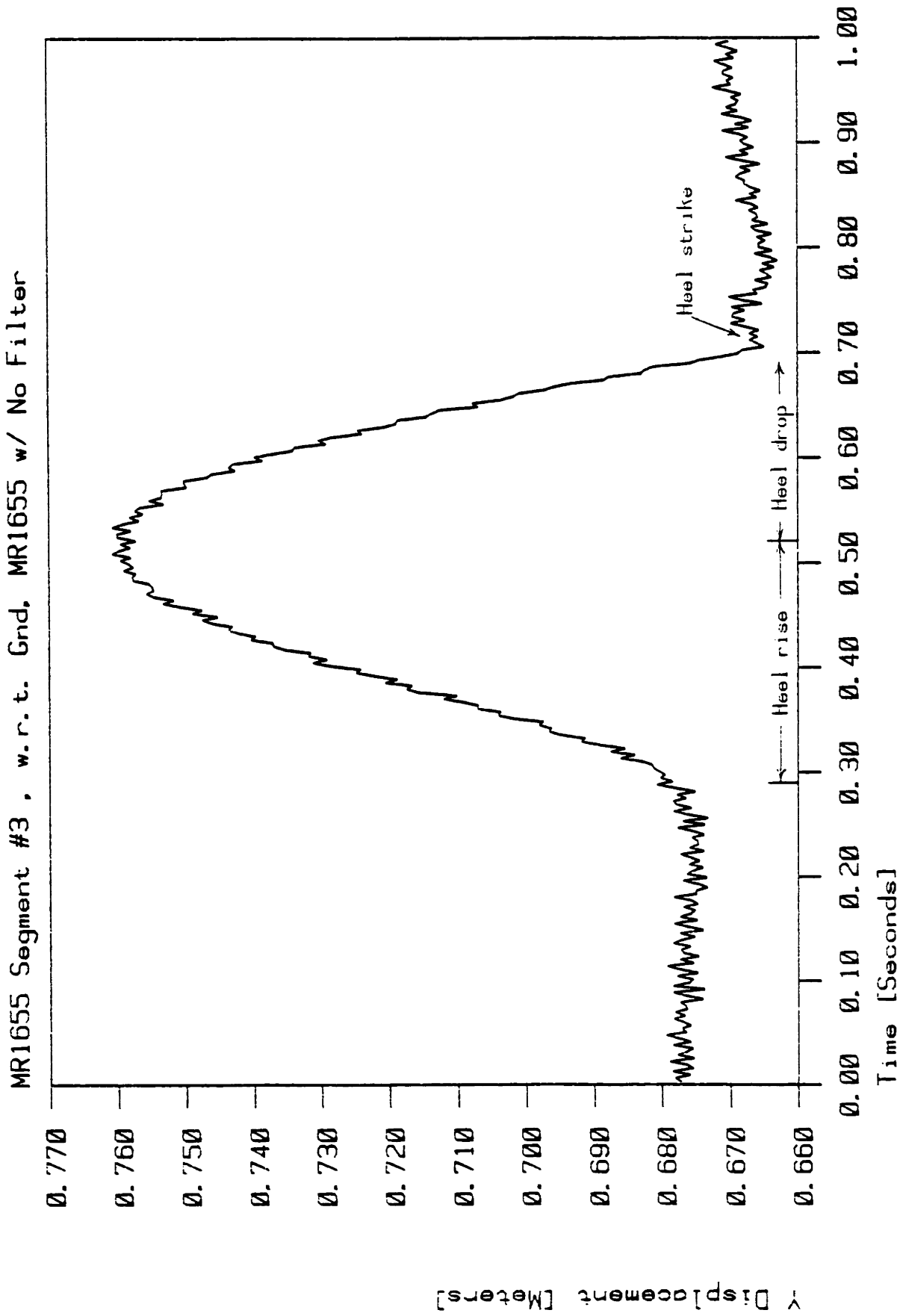
Figure 93.   Unfiltered vertical excursion of the thigh BCS
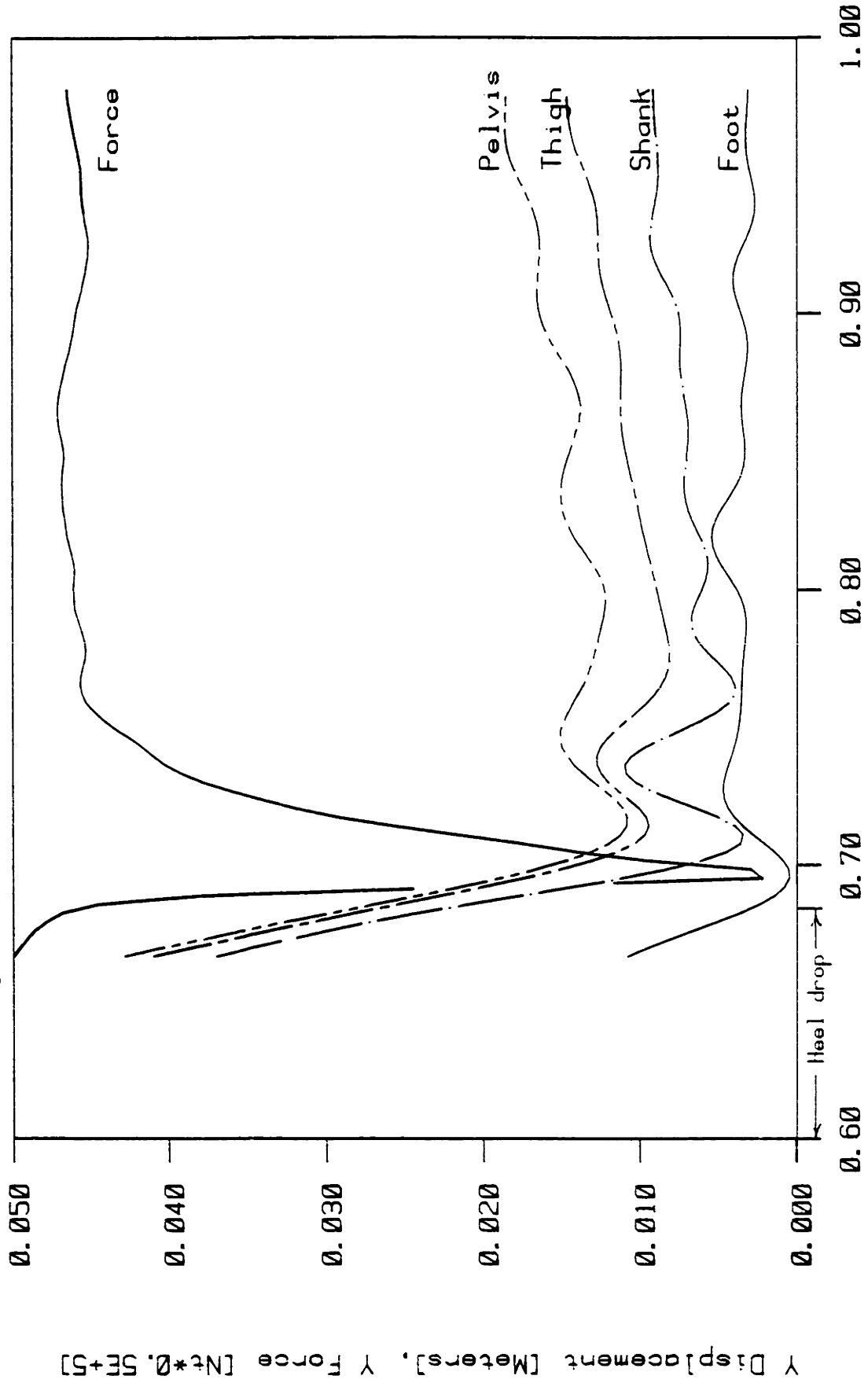
Figure 94. Composite 25 Hz filtered vertical excursion of the: foot, shank, thigh, and pelvis BCS subsequent to heel strike

Notice the phase lag as the soft tissue excitation traverses up the lower extremity. Light, et al., (1980) [95] in a paper devoted to a discussion of heel strike in gait, shows good temporal agreement between accelerations measured on the surface of the skin of the tibia and those measured with tibial bone pins, with some attenuation of the signal peaks in the skin measurements. Mizrahi and Susak (1982) [110] report the tension required in belts supporting transducers about body segments in order that accelerometer response becomes insensitive to belt tension. Presumably, lower tensions result in relative movement, while tensions above the reported value keep the accelerometer fixed with respect to the bone. The tensions they report are naturally dependent on transducer mass and belt material.

The frequency of vertical soft tissue motion is lower than that of force, approximately 12 Hz. Referring back to the frequency domain plots in Figures 62 and 64 which give filtered and unfiltered kinematic data for vertical motion of the lower leg, one of the initial hopes was that soft tissue motion would be distinct from gait motion in frequency, and thus could be treated as noise and eliminated by filtering. The data does not substantiate that hope, however; in fact, the frequency content of the heel-bounce experiment is significant in exactly the same range as is gait.

Fortunately the amplitude is no greater than 5 mm for any of the segments, which indicates a high level of kinematic fidelity of LED markers with the skeleton, even under a peak force of three times that experienced during gait.

The momentary axis of rotation (MAR) technique discussed earlier in Chapter 3 provides another way of evaluating soft tissue motion during gait. Figures 80 through 85 (shown earlier in this chapter) depict the motion of each segment relative to its adjacent segments during free-swinging flexion and extension. The knee in particular shows deviations from a circular trajectory of less than 3 mm. Figure 95 shows the motion of the shank relative to the thigh during gait. Notice the extend - flex slightly - extend - flex trajectory and the larger deviations from a circular path. These are caused by motion of the LED arrays with respect to the skeleton.
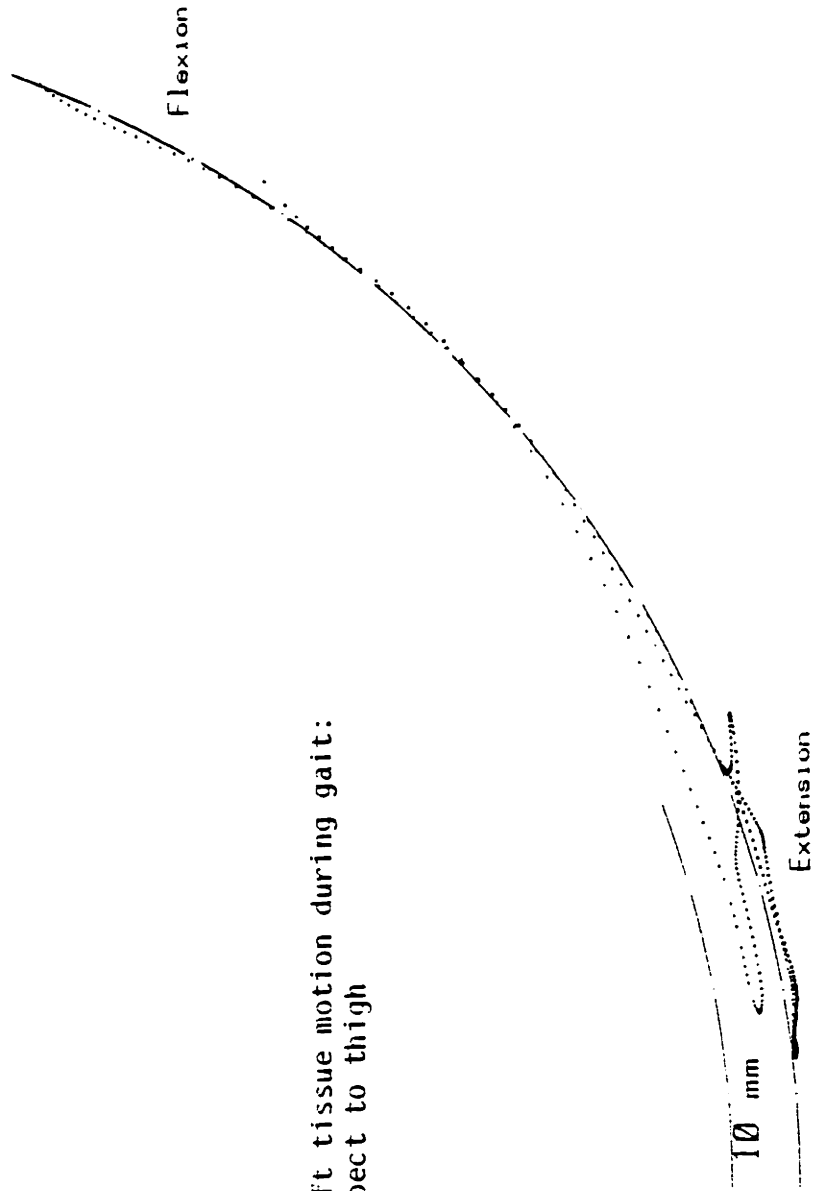
Figure 95.   Soft tissue motion during gait:
Shank with respect to thigh

The maximum deviation from a circle is 5 mm which coincides well with the soft tissue motion data derived from the heel-bounce experiment, even though gait represents quite a different physiologic condition. In the heel bounce, there was very little muscle activity at heel contact, in contrast to gait where muscle activity is quite high at heel strike. Muscle belly distension during contraction will affect the marker-segments location with respect to the skeleton; mounting the segments at palpable bony prominences minimizes muscle motion influence -- one of the major design goals for the segments. Quantifying the contribution of muscle activity to the total soft tissue motion would probably require a radiographic technique, not implemented in this investigation.

The marker position measurement system per se is capable of producing kinematic data with an accuracy of 1 mm at 3-meters range; however, the influence of compliant limb segments reduces the ability to place skeletal links to an accuracy of 5 mm. This finding represents novel information. Marker motion with respect to the skeleton has previously been difficult or impossible to measure because of limitations on the number of markers available (which eliminates the possibiliity of a full 6-degree-of-freedom kinematic description), or marker mounting schemes with less kinematic fidelity relative to the skeleton.

BODY SEGMENT INERTIA PROPERTIES

To perform a three-dimensional dynamic analysis of any mechanism, the mass of each link, the location of the center of mass, and the inertia tensor must all be known. Since dynamics are an integral part of this investigation, the mass and inertial properties of the lower limb segments had to be established. Much research has gone into measuring these properties, and there are many citations in the literature. A partial list includes: Braune and Fischer (1899) [16]; Dempster (1955) [48]; Drillis and Contini (1964) [52]; Moffatt, Harris and Haslam (1969) [112]; Clauser, et al., (1969) [33]; Casper, et al., (1971) [25]; Contini (1972) [35]; Cavanagh and Gregor (1974) [27]; Brooks and Jacobs (1975) [19]; Allum and Young (1976) [5]; Williams and Lissner (1977) [165]; Jensen (1978) [82]; Zatsiorsky and Seluyanov (1979) [183]; Hatze (1980) [70]; and Huang (1980) [79].

The independent determination of inertial properties for the individual subjects used in this investigation was deemed to be worthy of a separate project. Contini's 1972 paper [35] describes a method to estimate body segment parameters based on populations of subjects (both live and cadaverous) of several researchers. This technique was used to provide estimates of the mass and inertial properties of the thigh, shank, and foot of the two involved in this

thesis. Since both subjects had the same mass (71.5 kg),
and were the same height (1.88 m), one set of inertial
properties served for both. The results of applying
Contini's methodology are shown in Table 9. Williams and
Lissner (1977) [165] also published data on the principle
moments of inertia from Dempster (1955) and Miller and
Nelson (1974); these are included for comparison.

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
L thigh [m]	.4343						.461	.432
L shank	.4089						.455	.434
L foot	.0813						.081	*
M thigh [kg]	7.014	7.581	7.298		7.302	7.437	6.550	6.550
M shank	3.260	3.401	3.047		2.895	3.286	2.979	2.979
M foot	.992	1.204	1.063		.917	1.030	.950	.950
L cg thigh[%]	43.3	44.0	37.2		38.8	38.8	42.9	46.8
L cg shank	43.3	42.0	37.1		42.2	40.5	43.1	45.2
L cg foot	42.9	44.4	44.9		43.7	43.2	66.8	-
Ix thigh [kg-m²]				.0587				.0872†
Iy thigh				.020				.0184†
Iz thigh	.109(.086)			.0587	.1281	.1186	.0872	.0872
Ix shank				.0465				.0429†
Iy shank				.0038				.0030†
Iz shank	.0423(.0144)			.0465	.0424	.0466	.0429	.0429
Ix foot				.0058				.006†
Iy foot				.0056				.006†
Iz foot	.003(.0005)			.0058	.0032	.0044	.0060	.006

(1) Dempster (as reported by Williams and Lissner [158], and Vaughn [155]) subject mass 70.92 kg
    (values in parentheses are Dempster's tolerance) subject mass 70.92 kg
(2) Braune and Fischer (from Williams and Lissner [158])
(3) Clauser (from Williams and Lissner [158])
(4) Miller and Nelson (from Williams and Lissner [158])
(5) Chandler (as reported by Williams and Lissner [158])
(6) Vaughn (optimized from long jump) [155] subject mass 70.99 kg, 1.75 m tall
(7) Contini [34]
(8) Values used here for subjects with body mass 71.48 kg, 1.88 m tall

* This value (foot limb length) is not needed, the forceplate provides a vector from the foot body coordinate system to the distal force during stance.

"L cg" is the ratio of the length from the proximal end of the segment to the cg, to: the total limb segment length.

Table 9. Mass and inertial properties of the thigh, shank, and foot

The values in the table indicated with a dagger (+) were not calculated from Contini's paper, but had to be estimated by other techniques. Essentially, Contini only provides inertias about axes normal to the saggital plane. This is sufficient in studies of motion limited to that plane; however, the complete inertia tensor was required here. To obtain estimates of the terms not provided, each segment was assumed to be a frustrum of a cone, with height provided by Contini (segment length) and diameters measured physically. The density was taken from Contini and assumed to be uniform.

Using these somewhat broad estimates and not pursuing the determination of mass and inertial properties further was considered justified for two reasons: First, gait is gravity dominated. There are inertial effects to be sure, but the quasi-static maintenance of the center of mass off the ground is the predominant source of load. Second, sophisticated techniques exist using computer tomography (CAT scans) to measure density at a number of closely spaced "slices" of a living subject and determine inertias on a individual basis to a high degree of accuracy. See Casper, et al., (1971 [25]); Brooks and Jacobs (1975) [19], and Huang (1980) [79]. This technique was not available for this investigation, but could be for subsequent research.

SEGMENT GEOMETRY

In the process of determining the inertias of limb segments, Contini produced population-based estimates of link lengths. These were used in estimation of inertias about axes not discussed by Contini. The link lengths were also used as a verification that the joint axes and vectors to link endpoints were determined correctly. Subsequent to the momentary axis of rotation solution, a point on the axis inside the joint is chosen to represent the link endpoint. This is the point of application of forces and moments at the end of that link. Vectors from the BCS to the link endpoints comprise the geometry needed for the dynamic analysis, and since these vectors are calculated from the kinematic data, there is no need to independently measure the LED segment locations on the subject by referencing them to bony landmarks. The distance between endpoints should be the same as Contini's link lengths, if the subjects employed here are representative samples of the populations used to compile his data.

Table 9 also lists the geometry data for one of the subjects used in this investigation (column 8). The point on the axis of rotation to be used as the segment endpoint was chosen to be in the geometric center of the joint by an external independent measurement. Notice there is no vector defined distal to the foot BCS. The position on the

forceplate of the center of net force is used in the dynamic estimates as the point of application of distal foot force.

Since Contini only provides center of mass location as a ratio of limb length, each three-dimensional C.G. location was assumed to be on the straight line connecting the segment endpoints.

> "CONCEPT OF BODY LINKS
> In kinematic analysis of human motion, the important moving units are not the various bones, which support the surrounding soft tissue structures, but rather the total mass of the segments that turn about the joint axes. The rational axes are not located at the junctures of bones. For example the axis at the hip lies within the femoral head... Both the elbow and knee axes are proximal to the respective articular surfaces."
>
> "The central straight line, which extends between two axes of rotation, is termed a 'link.' Thus, the femur and humerus are longer than their respective links, and the tibia and radius are shorter. The bones form the rigid support required by the segment, but the bone itself is not the link. Link systems are interconnected by joints..."
>
> "Centers of mass of the limb segments lie along link lines (lines connecting adjacent joint axes)."
> Williams and Lissner (1977) [165] pages 206 and 208.

INDIVIDUAL SEGMENT FREE BODY DYNAMICS

Once the kinematics, geometry, and inertias have been measured or estimated, all of the necessary information exists to perform a dynamic analysis and determine the forces and moments that must have impinged on each link for the observed motion to have occurred. The technique is straightforward. Given a two force link* with known kinematics, geometry, and inertias, and the force and moment vectors known at one end, Newton's equations can be applied to determine the force and moment vectors at the other end for each time step. This is a system of six equations in six unknowns and is easily solved. Let us use the foot as a representative link to demonstrate. During the swing phase the forces on the "distal end" of the foot are known; they are always zero. The "distal end" is any point on the inferior surface of the foot. With a forceplatform, the forces and moments at the distal end are also known during stance provided only one foot is on the plate. For one stride, therefore, the complete force and moment history on the distal end of the foot is known; thus the forces at the "proximal end" of the foot can be determined. The "proximal

---

* Two force means that there are only two points of application of external force on the segment. Here they will be described as "proximal" and "distal" to represent the joints at opposite ends of each link in the lower extremity chain.

end" of the foot is a point in the ankle joint determined from the geometry derived from the momentary axes of rotation discussed earlier. These proximal foot forces are the negative distal shank forces, and the process can begin again for the shank. This incremental segment, dynamic solution can proceed along a kinematic chain including all instrumented segments, in this case up to and including the pelvis. See Antonsson (1978) [7], particularly Chapter 4, for a more detailed development of the dynamic solution technique. The dynamic-estimation routine is called "NEWTON" and uses the equations shown in Figure 96.

$$\sum \text{Forces} = F_p + mg - ma + F_d = 0$$

$$\sum \text{Moments} = T_p - I\dot{w} - w \times Iw + F_p \times r_p + T_d + F_d \times r_d = 0$$

Where:

m = the segment mass

I = the segment inertia tensor

g = the global gravity vector

a = the global acceleration vector

$\dot{w}$ = the global angular acceleration vector

w = the global angular velocity vector

F_p = the proximal force vector

F_d = the distal force vector

T_p = the proximal torque vector

T_d = the distal torque vector

r_p = the vector from the BCS origin to the point of application of the proximal force

r_d = the vector from the BCS origin to the point of application of the distal force

$\times$ = the vector cross product operator

Figure 96. Dynamic estimation equations

As a verification that the dynamics are correctly estimated, an experiment with a simple machine was performed. The instrumented linkage was a rigid steel hollow bar, 50 mm by 50 mm in cross-section and 1.2 meters long. A 44.5-mm diameter, stainless steel ball was press-fit in one end. An array of LEDs was attached to the bar with its body-coordinate system origin at the center of mass of the bar and ball. The instrumented bar was placed in the field of view of both Selspot cameras, with the ball end on the forceplate surface, see Figure 97.

Direction of
unconstrained
motion

LED array

BCS

GCS

Forceplatform
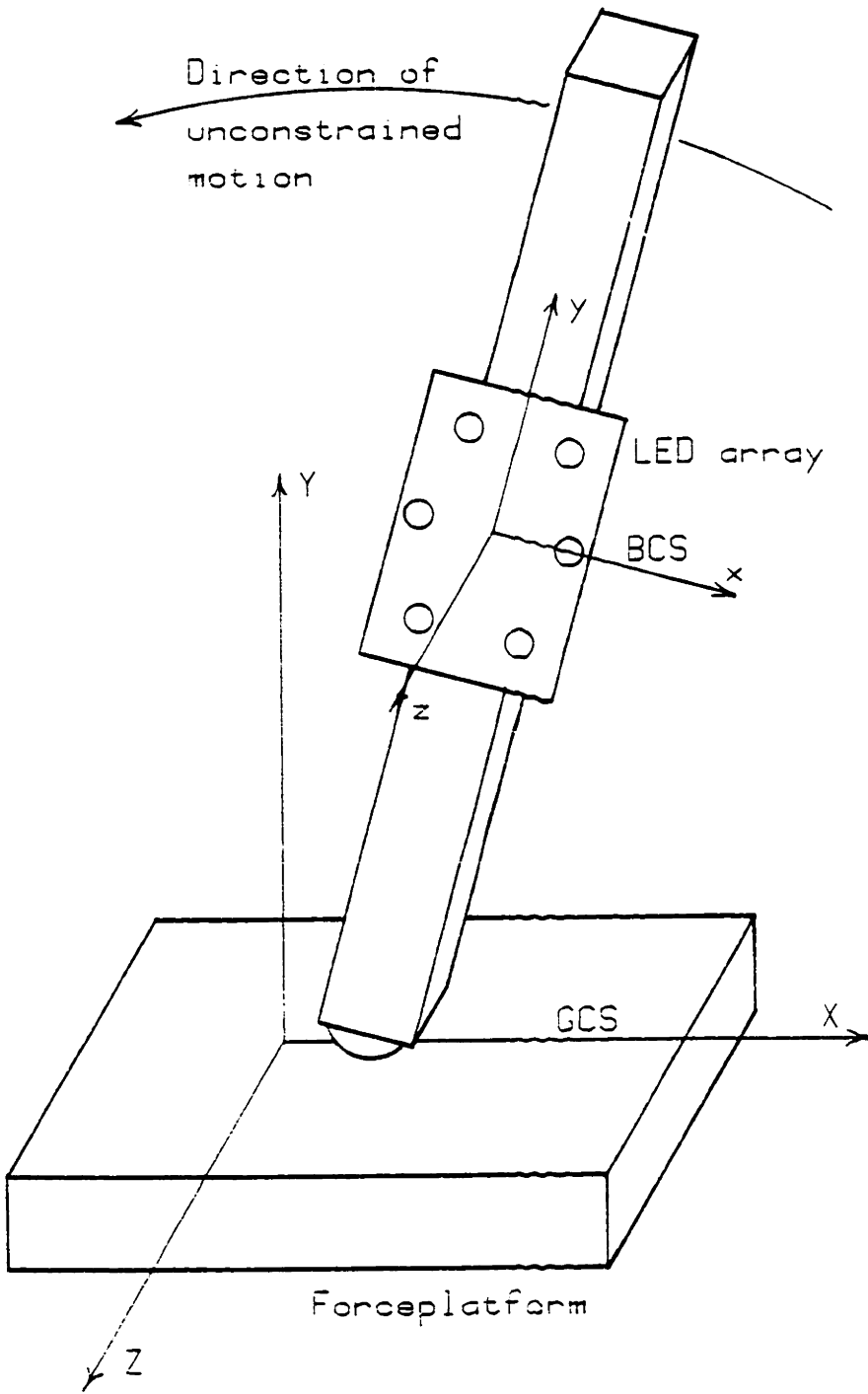
Figure 97. The inverted pendulum experiment

TK4:FB2235.DAT;0

EXPERIMENTAL TRACK PARAMETERS: 12:51:25   22-FEB-82
SAMPLING PARAMETERS:
  Sampling frequency=   315.00
  Total number of frames =   512:   1.63 seconds of data
  Segment file name = SQPEND: 6 channels,   1 segments
Camera positions: X   Y   Z [Meters]   AX   AY   AZ [Degrees]
  Camera 1   0.000   0.000   0.000   0.000   -26.000   0.000
  Camera 2   3.353   0.000   0.000   0.000   26.000   0.000
Camera calibration factors: Camera 1   Camera 2
  Focal distance (SU)   2315.16   2295.02
PROCESSING PARAMETERS:
  Forceplate scale factors: X   Y   Z   [Newtons/Volt]
                            2.00E+01   1.00E+02   1.00E+01
  Skew ray max error= 10 [Selspot Units],   Errmax= 20.00 %
Data Smoothing: Interpolation   Low-pass Filter
  Raw Selspot   None   None
  3-D Selspot   Yes   12.0 Hz Cutoff
  BCS Selspot   Yes   None
  Derivative   None Available   None
  Forceplate   None Available   12.0 Hz Cutoff
Global axes origin: (1.695,-0.617, 3.466) [meters]
Approximate data processing time: 3[min] 4[sec]
PROCESS RESULTS:
  Raw Data file storage name: FB2230 in Directory: ANTON1
  Data Window includes frames: 53 thru 564
  Number of Selspot hardware bad points: 0 Frame-Channels
  Number of bad points eliminated by 3-D skew ray check: 1 Frame-Channels
  Number of bad points eliminated by inter-LED length check: 0 Frame-Channels
                                                    and: 0 Frame-Segments

The worst inter-LED length error was: 16.87%

Table 10.   The inverted pendulum experimental parameters

```
Mass of bar =            5.857  kg
Mass of LED segment = 0.0281 kg
Total Mass =            5.8851 kg
```

Inertia about the long axis (Y) = 0.00429 kg-m^2
Inertia about the X and Z axes = 0.7910 kg-m^2

Total bar length =   1.250 m
Diameter of the "distal" ball =   0.046 m
Distance from the "distal" end to the c.g. = 0.585 m

Table 11.   Mass, inertia, and geometry properties
of the inverted pendulum

The bar was held in the position shown by one person and thrown to another person with the ball maintaining constant contact with the forceplate. The object of the experiment was to estimate the forces acting on the upper (proximal) free end of the bar during its free flight phase. If TRACK and then NEWTON produced correct results, the forces would be zero. The measured kinematics of the bar motion in the global X-Y plane are shown in Figure 98, and the motion in the Z-Y plane are shown in Figure 99. The momentary axis of rotation technique was applied to these kinematics. Figure 100 shows the motion in the best fit plane, and Figure 101 shows deviations from that plane.
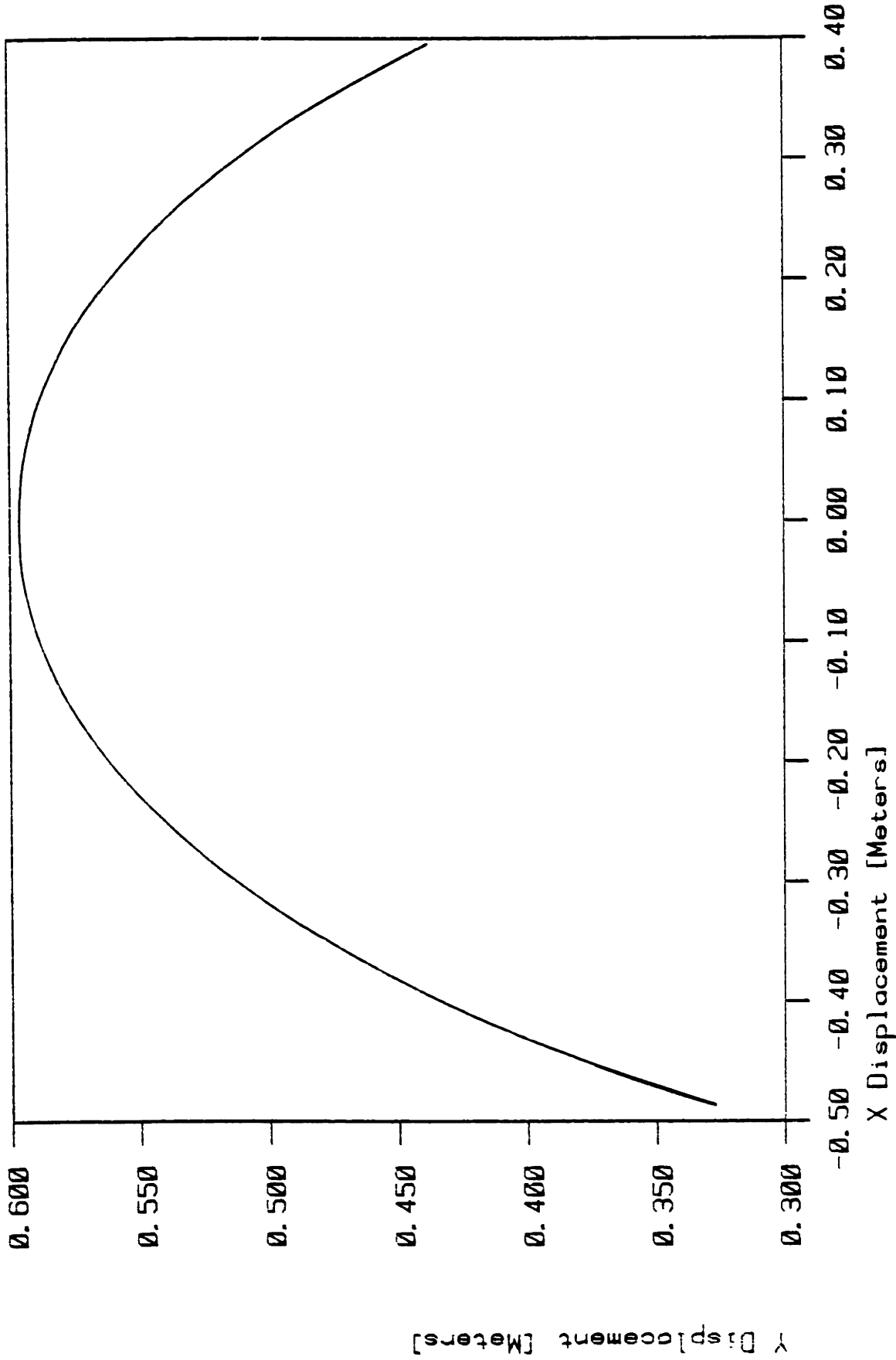
Figure 98. Global X-Y plane inverted pendulum motion
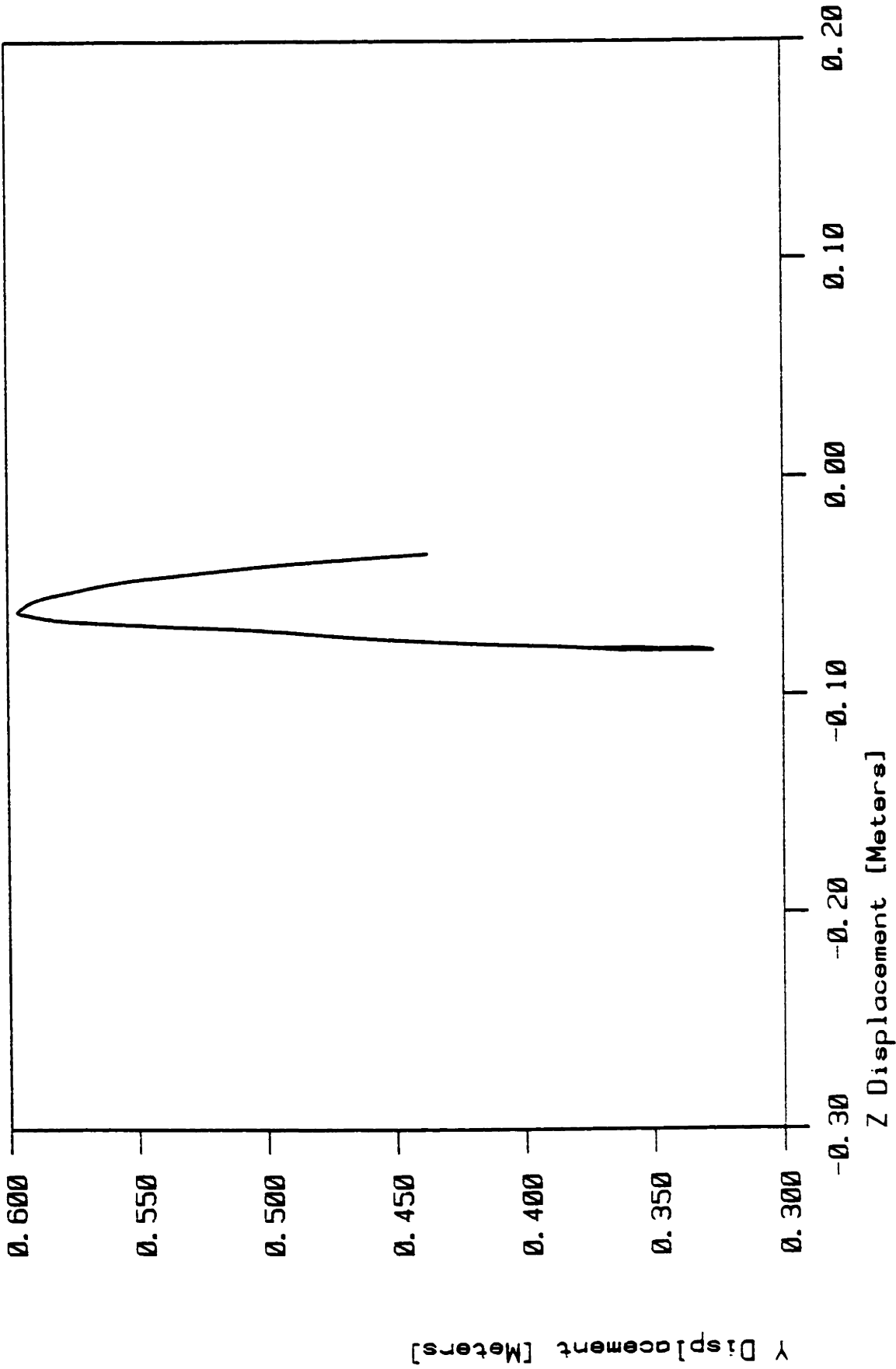
Figure 99. Global Z-Y plane inverted pendulum motion

(-0.0011, 0.0242, -0.0481)
R= 0.5728 [meters]
URV=-0.0504, 0.0195, 0.9985
Frames: 1 to 512
TK4:FB2235.DAT:0
Seg# 1 wrt Seg# 0.   ~X/Y Plane
FB2230 12 Hz Filter, Windowed

10 min
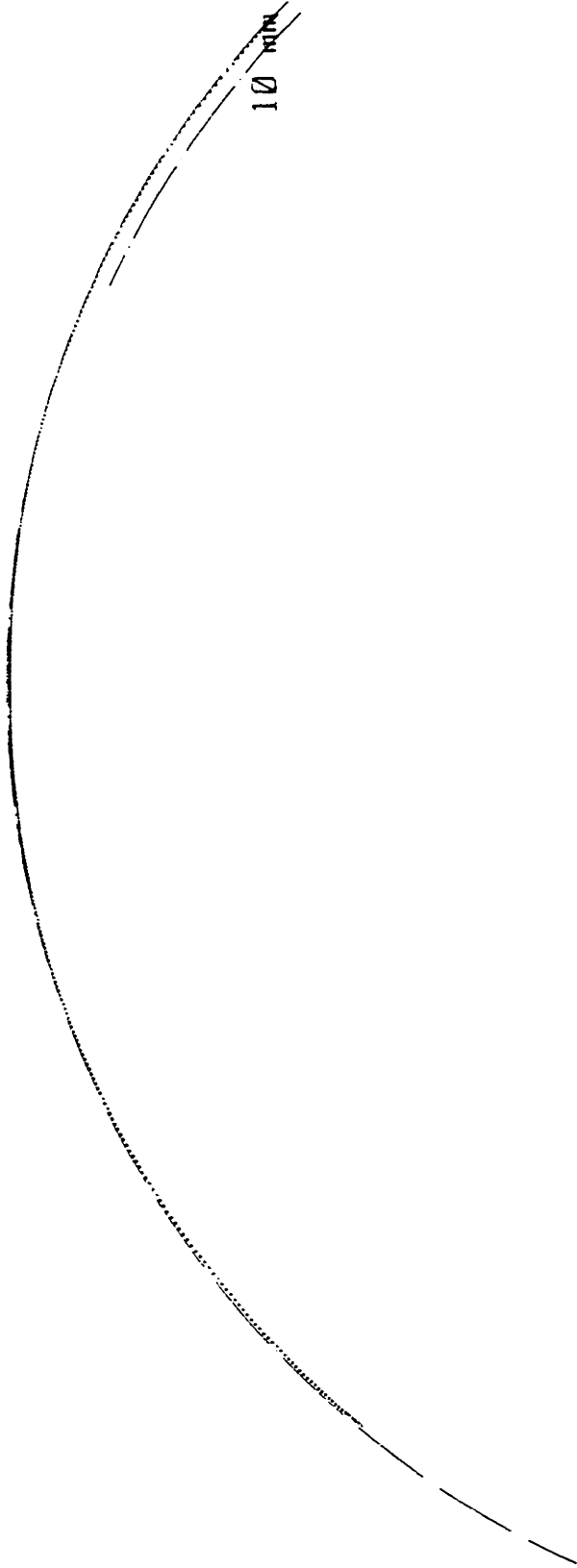
Figure 100. Inverted pendulum
motion in the best fit plane

10 mm

(-0.0011, 0.0242, -0.0481)
R= 0.5728 [meters]
URV=-0.0504, 0.0195, 0.9985
Frames: 1 to 512
TK4:FB2235.DAT;0
Seg# 1 wrt Seg# 0. ~Z/Y Plane
FB2230 12 Hz Filter, Windowed
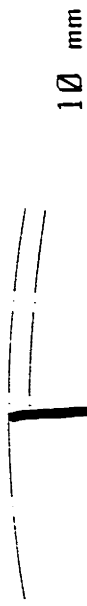
Figure 101. Inverted pendulum motion
deviations from the best fit plane

The kinematics were particularly interesting in that the axis of rotation was found to be above the floor. If pure rolling had been taking place, the real axis of rotation would always have been at the contact point between the ball and the floor, and the momentary axis solution would have found a center slightly below the floor (See Figure 77). The forceplate data for this experiment demonstrated that the contact point did not move as far as it should have had pure rolling occurred, which indicated that some "negative slip" took place. The amount of slip correctly explained the location of the calculated momentary axis location 24 mm above the floor.

Since the measured force on the lower end of the bar used as input to NEWTON, was produced by the forceplate, the measured location of the contact point was also used as the point of application of the "distal" reaction force, which meant that the presence or absence of slip was automatically included. Figure 105 is a plot of the results of that dynamics estimation.
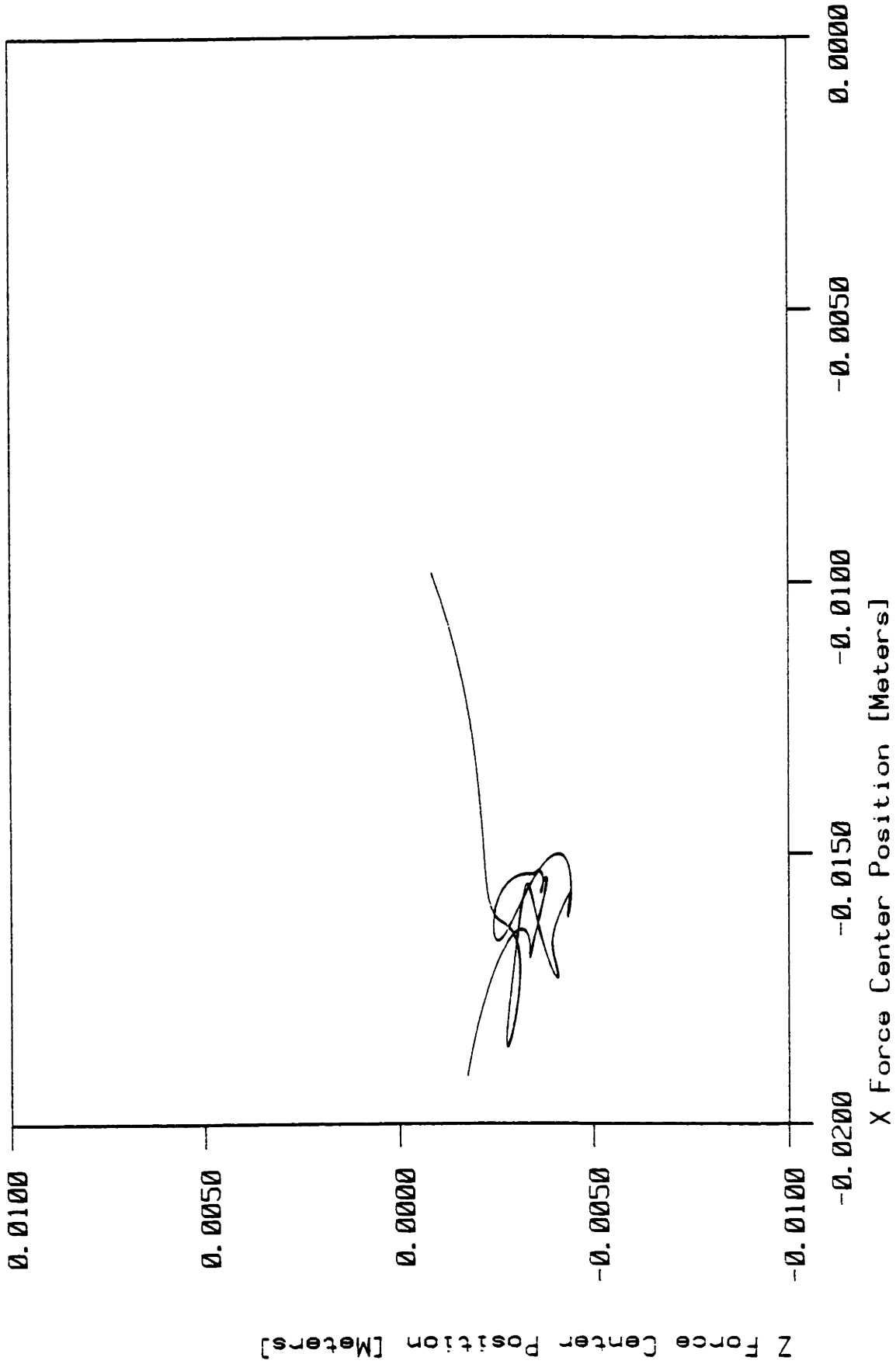
Figure 102. Measured forceplate net force center position
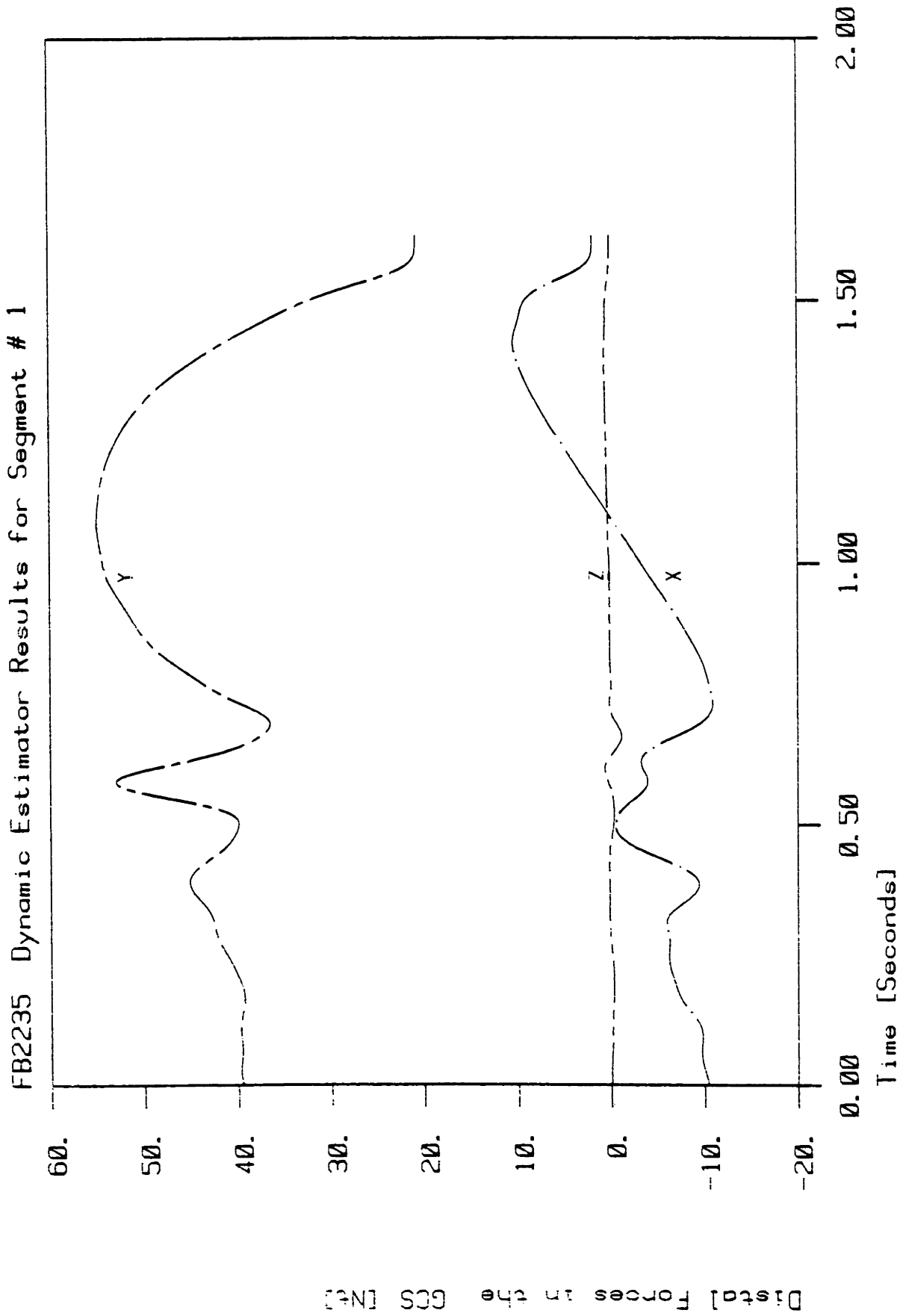
Figure 103. Measured forceplate forces
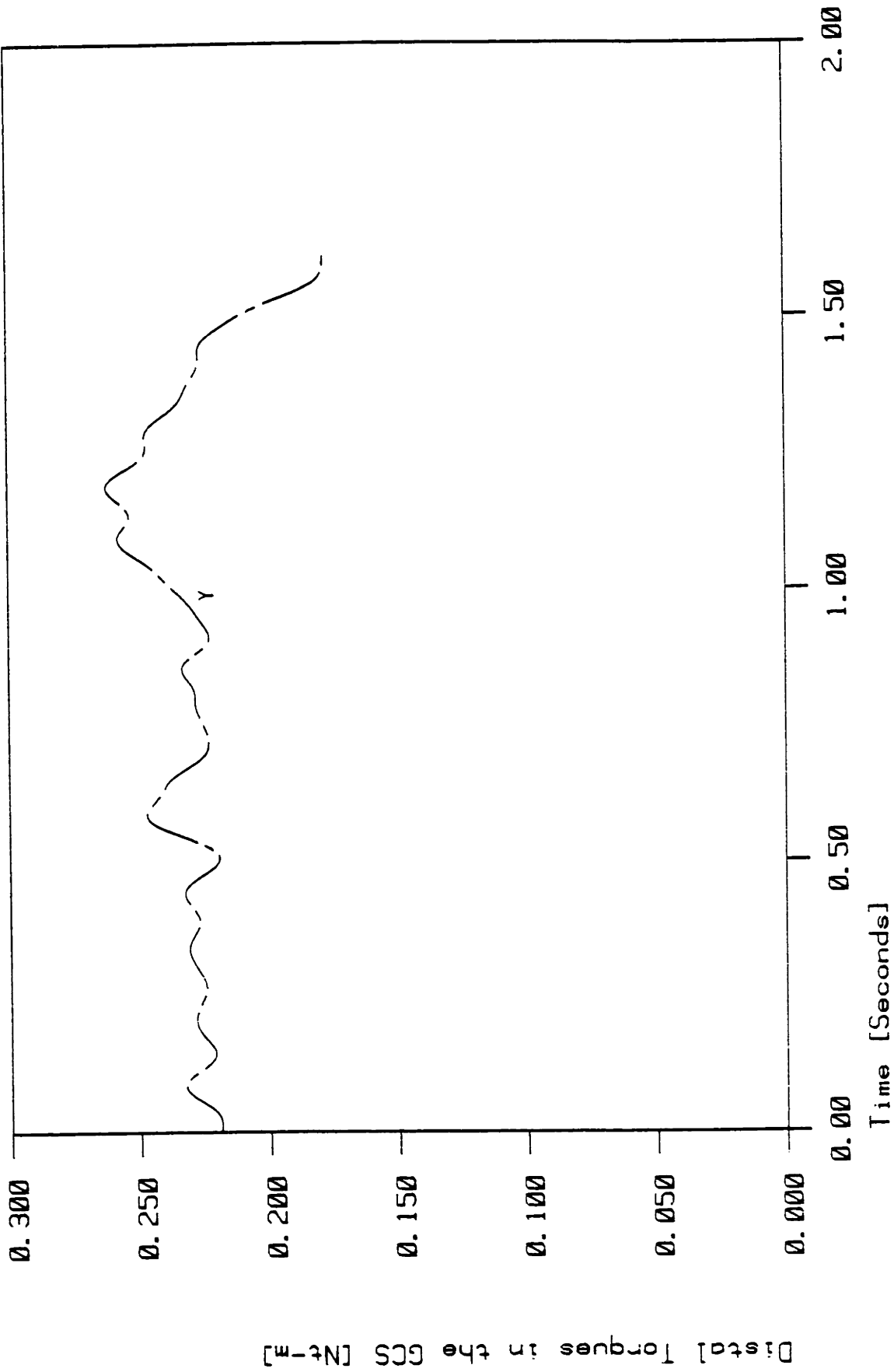
Figure 104. Measured forceplate moment
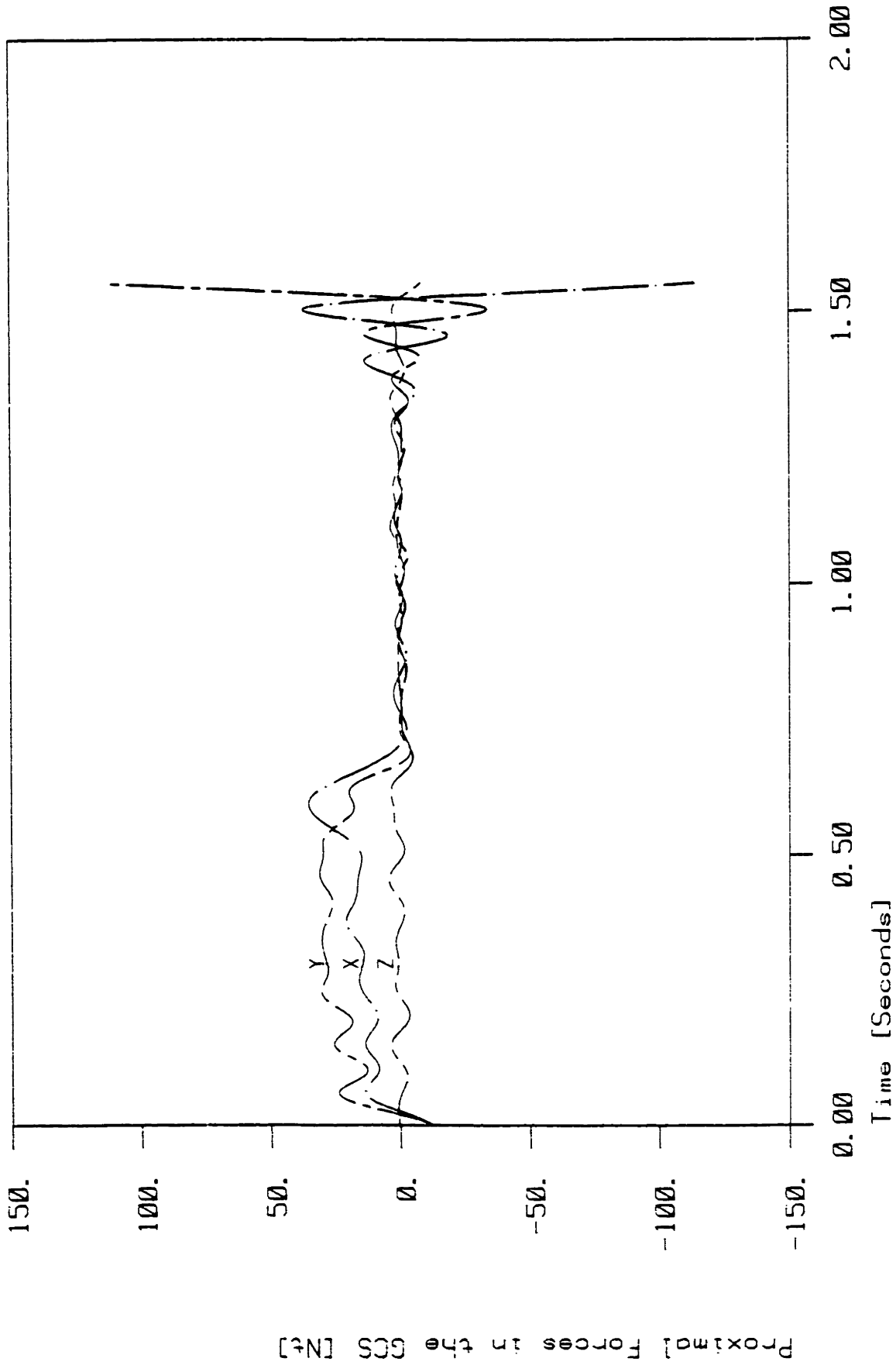
Figure 105. Dynamics results for the inverted pendulum

During the "time of flight" (that time when the bar was moving freely), the forces on the upper end of the bar were estimated to be 0 plus or minus 4 Newtons [1 pound]. The "flight" time was from approximately 0.7 seconds to 1.5 seconds. At 1.5 seconds the pendulum was caught by one of the experimenters, and the data acquisition ended. The force in the global X direction dominates just before release. Positive X is the direction towards which the bar was thrown. The oscillations at the end of the data set are produced by filter "ring" due to the sharp change in velocity and acceleration when the bar was caught. Since the filter is implemented forward and backward in time (to eliminate phase shift), the ringing contaminates the signal "upstream" of the catch. Figure 106 is the estimated moment at the free end of the bar.
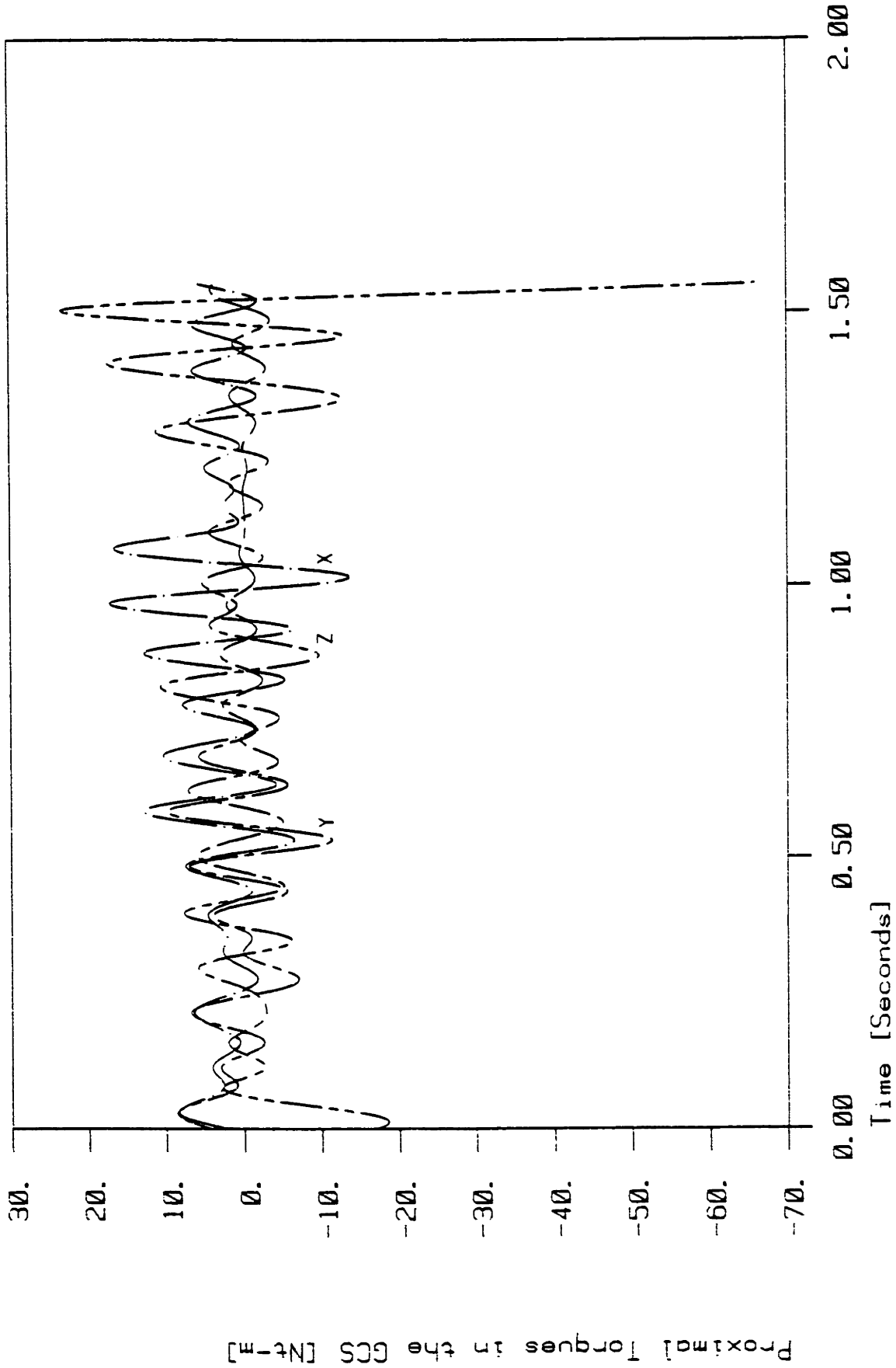
Figure 106. Estimated moments at the upper (free) end of the inverted pendulum

Notice that there is no change in the amplitude of oscillation of these signals during "flight." This is expected because there were no applied moments either before, or during the bar toss. The moments are estimated to be 0 plus or minus 10 [Newton-meters] (7.4 foot-pounds).

If perfect kinematic data (no measurement noise, infinitesimally small resolution, and perfect accuracy) were available, the force and moment estimates would be exactly zero during the unconstrained motion "time of flight". Since, however, the dynamic results are based on double-differentiated, kinematic data with residual noise present, noise will contaminate the answers. Recall the dynamic equations used in the estimator (see Figure 96). For this example, with one rigid link with very well-known mass and inertia properties, the only two possible terms representing noise introduction are from the forceplate and the double-differentiated position histories. The forceplate has been shown earlier to have a signal-to-noise ratio of better than 2000, and represents the least likely noise introduction factor. The residual kinematic noise at low frequencies once double differentiated is much more likely. Figures 107 and 108 show the remaining noise (as determined by a stationary segment, see Chapter 3) and Figures 109 and 110 show the influence of the same noise on the accelerations.
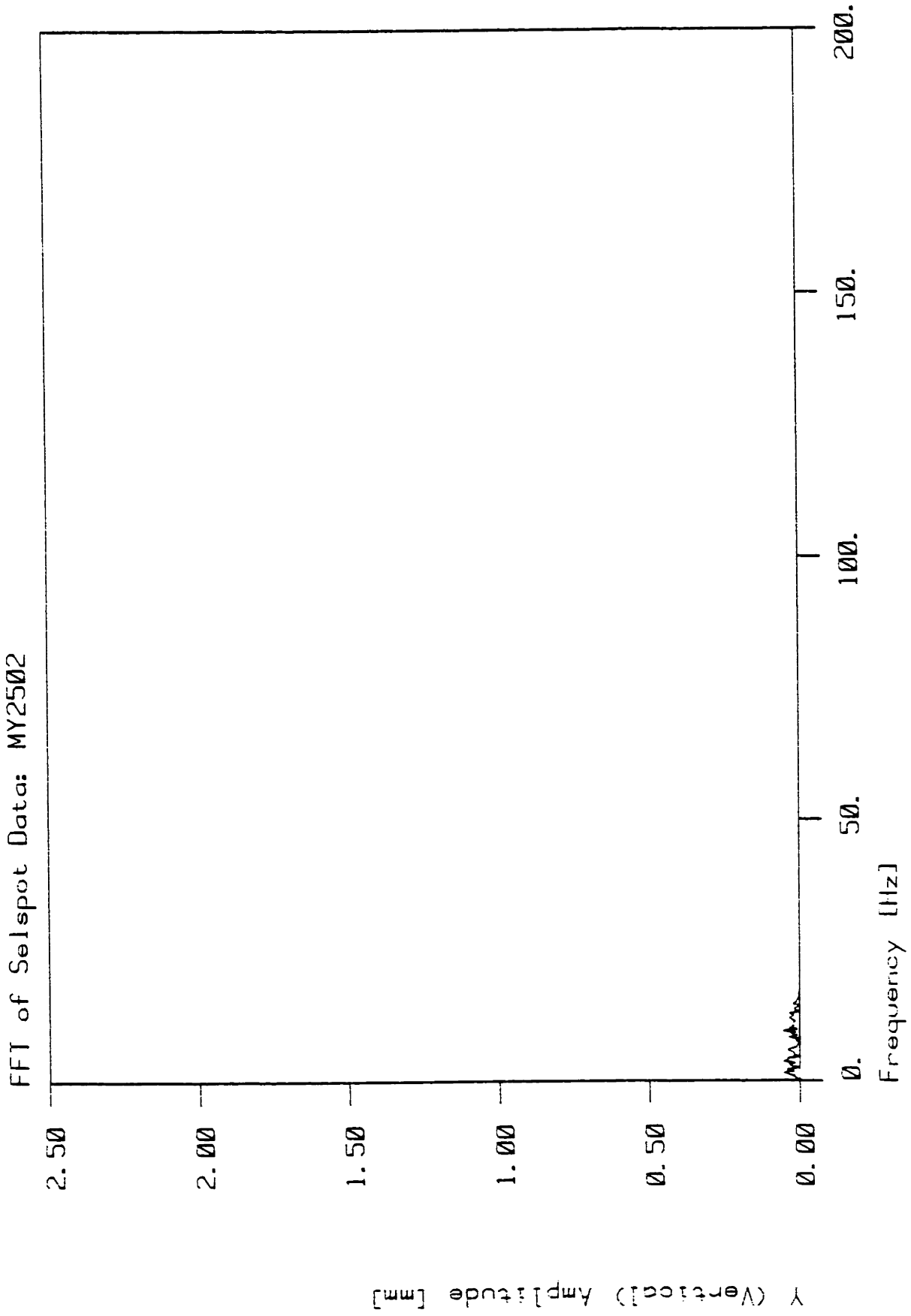
Figure 107. Residual translational position noise

FFT of Selspot Data: MY2502

AZ Amplitude [milli-Radians]
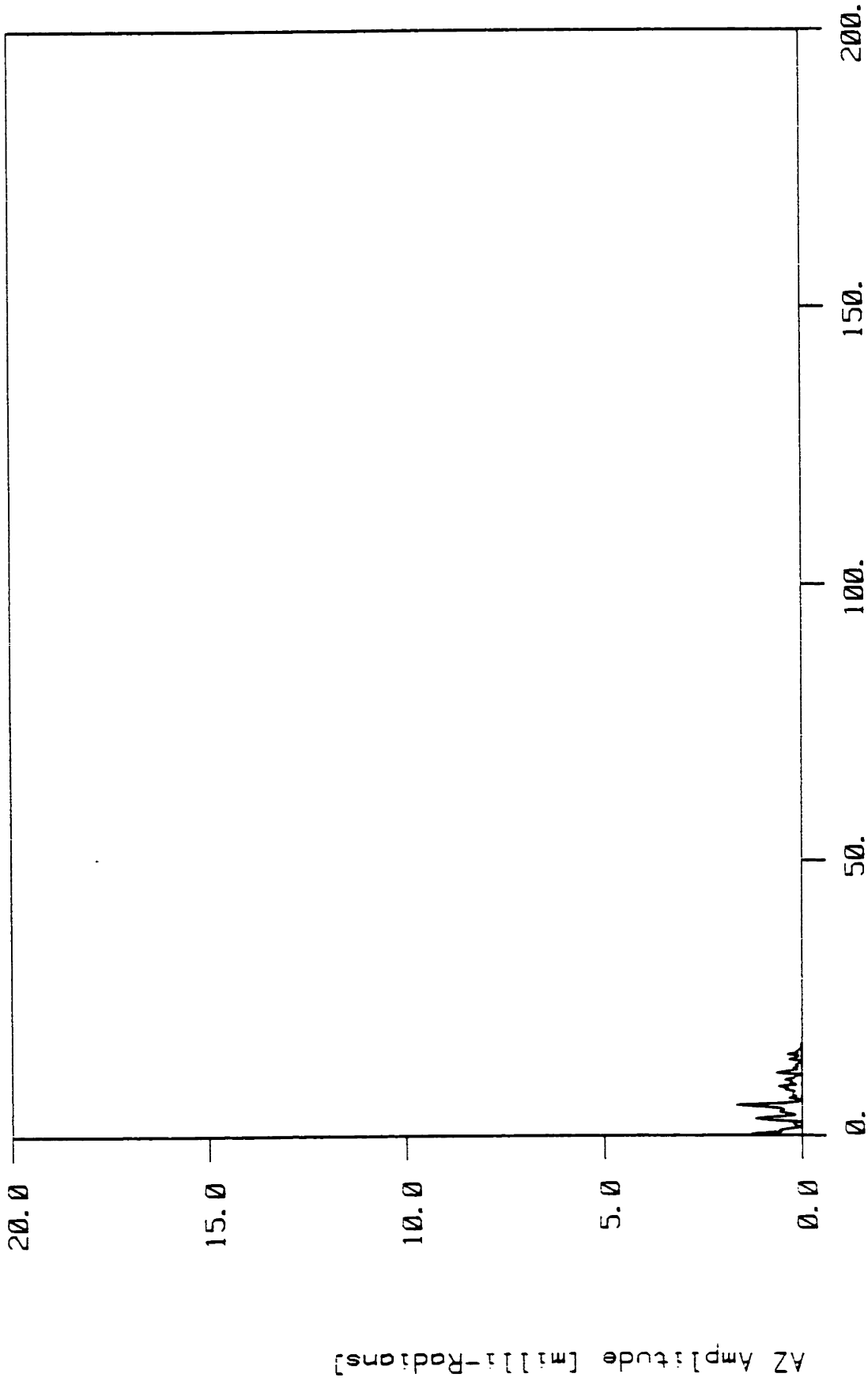
Frequency [Hz]

Figure 108. Residual rotational position noise

- 304 -

FFT of Selspot Data: MY2502

Y Accel (Vertical) Amplitude [Percent Gravity]
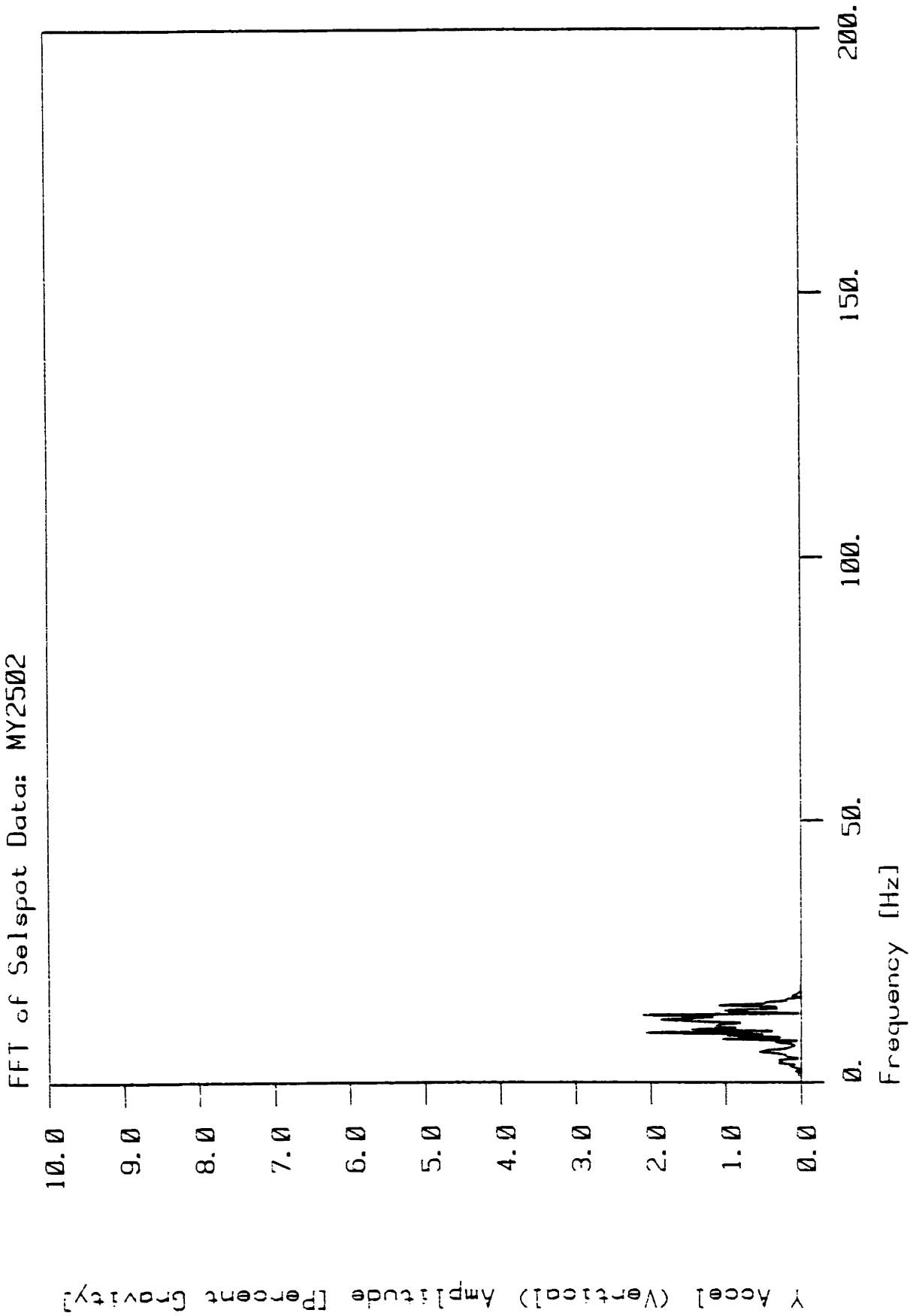
Frequency [Hz]

Figure 109. Residual translational acceleration noise
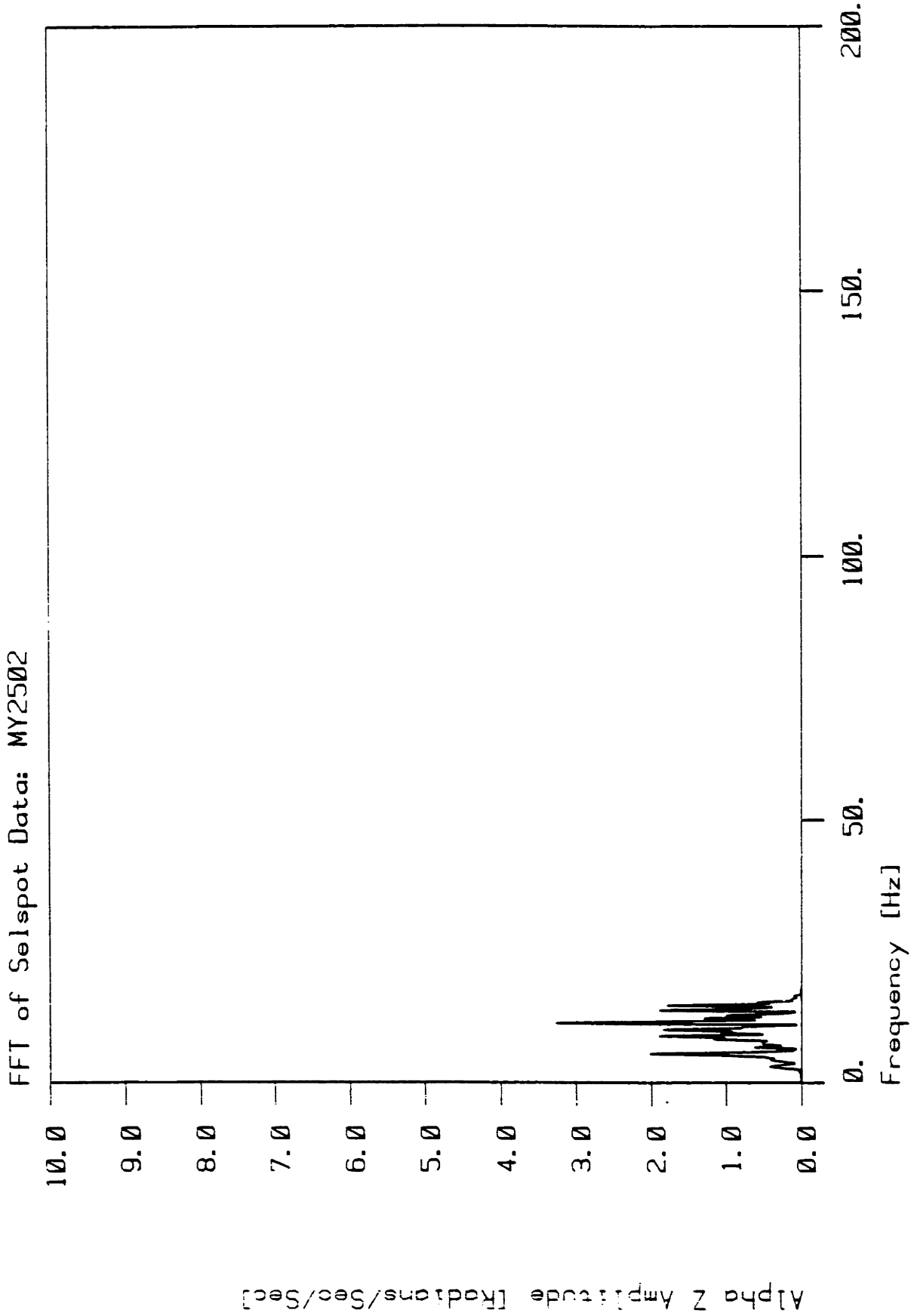
Figure 110. Residual rotational acceleration noise

The amplitude and frequency content of the residual acceleration noise agree with the oscillations about zero force and moment in the inverted pendulum results. The rotational acceleration noise appears as a more significant influence than the translational for two reasons: First, there were no significant moments applied to the pendulum before it was "launched," in contrast to the applied forces necessary to "throw" it; thus the resultant torque level doesn't change during flight. Second, since the major component of force acting on the pendulum during all of its motion is gravity, the influence of acceleration noise at a level of only 2 percent of gravity is small. Rotational action has a much smaller gravity component as the inverted pendulum passes through its vertical position. Thus the moments are much more strongly influenced by the estimation of rotational acceleration, and hence by contaminating noise.

To compare these results with the peviously published art of segment-force estimation, consider the influence of noise, which has yet to be included in any other gait analysis, on the twice-differentiated position signals derived from a typical 30-Hz system (one-tenth the sampling rate used here) with typically five times worse positional resolution -- a completely unexplored frequency content, and an arbitrarily selected filtering protocol. There is no chance that those systems can produce results with the same

high quality as TRACK, and comparisons of quality are virtually impossible, due to the lack of published error analyses.

The data analysis method is based fundamentally on the kinematics of imbedded body-coordinate systems. These provide geometry information on the instrumented link assemblage, and with mass and inertial properties, the linkage dynamics can be estimated. To evaluate quantitatively the overall quality of the dynamics, the quality of each component used in Newton's equations must be determined. These component determinations have been discussed in this chapter, and the next chapter will aggregate this information to provide a "bound of confidence" on the dynamic results.

To even further improve the TRACK results, several approaches can be taken. A filter less inclined to "ring" should be considered. The Butterworth filter used throughout has excellent frequency domain characteristics, but is prone to oscillating at its cutoff frequency if excited. A Chebyshev filter may prove a better choice. Second, the higher resolution (12 bits instead of 10) Selspot II should also represent a decrease of noise level by a factor of 4. This will benefit the dynamic results directly.

Several novel techniques have been developed and described: 1) automation of the kinematic acquisition and reduction process, particularly the Selspot, for automatic, real-time kinematic data acquisition; 2) the imbedded body coordinate system approach for complete kinematic description Conati (1976) [34]; and 3) the momentary axis of rotation solution, to determine joint axis locations, and link endpoint definitions, in addition to the quantification of the influence of position measurement noise on the estimated dynamics results.

CHAPTER 5

RESULT DATA AND CONFIDENCE LIMITS

If any experimental data acquisition and reduction
system is to be used with confidence, the quality of the
resulting data must be assessed. Since there exists no
independent measurement of correct gait dynamics,* no direct
comparison can be made to determine absolute accuracy in
this case. As an alternative, a bound of confidence, or
confidence limits, about the estimated data is produced.

> "Data on the accuracy of cinematographic and force
> plate methods (of human motion analysis) are very
> scarce, Cappozzo, et al., (1976) [21] is one of
> the few sources of these data."
> Hof and Van den Berg, 1981 [77]

---

* The only internally measured joint forces are the 1967
study by Rydell [135] where a hip prosthesis with
hard-wired, strain gauges on the femoral neck was
implanted, and Kilvington and Goodman, (1981) [88] who
also used strain guages but telemetered the data outside
the body. A multitude of problems with the experimental
protocol has led many gait researchers to question the
validity of these results as they apply to normal gait.
Furthermore, in neither case was an explicit attempt made
to compare hip joint forces calculated from kinematic gait
data with the direct measurements. Professor R. W. Mann
(M.I.T.) [24] expects to have an in-vivo instrumented
hemi-arthroplasty implanted this year to measure the
pressure distribution in the hip joint. These pressures
will also produce an independent internal estimate of the
total load carried across the joint. But note that since
the inverse dynamics calculations based on kinematics only
estimate net motion producing forces and moments, the
internally measured forces will differ by the amount of
force generated by muscles (including agonist-antagonist
coactivation) while they are producing net joint torques.

Since the dynamics estimator (NEWTON) produces force estimates, not exact forces, the bound of confidence is a region about the estimated data in which the correct result must lie, based on information concerning errors accumulated throughout the experimental data acquisition and reduction process. Possible sources of these errors have been enumerated in earlier sections; however, a brief list of those that pertain to this system will be repeated here.

1. Sampling noise in the kinematics and foot-floor forces

2. Elimination of small amplitude, high frequency components of the signal by filtering to remove noise

3. Contamination of the signal in the space-time domain by filtering

4. Limited positional measurement accuracy and resolution

5. Motion of measurement markers mounted on soft tissue, with respect to the skeletal links

6. Inability to resolve joint axes exactly

7. Errors in the segment mass and inertial properties

Errors of the nature described here will always contaminate results from a human motion analysis. Any system should be designed to minimize their impact, and their influence should be measured to determine quantitatively the quality of the final data. All of the elements of the list above have been assessed in the course of this investigation, and this chapter will aggregate those

results to determine conservative bounds of confidence. Conservative means that every estimate of error contribution is calculated from the maximum possible components resulting in a bound that is greater than the largest possible error.

The only significant attempt to determine the bounds of confidence on resultant gait data found in the literature was Cappozzo, et al., (1975) [22], (Hof and Van den Berg notwithstanding). In that work, rather than concerning themselves with determining the absolute accuracy of the components of their human motion analysis system, they determined the variance of the sampled gait data with respect to a least-squares, best-fit of the first five terms of a Fourier series. This approach, coupled with their relatively low kinematic sampling rate (20-30 Hz) and use of the same population-based, literature-reported, inertial data used in this investigation, led them to the following conclusions about the dynamic data quality of their results:

> "The contribution of the body segment masses, moments of inertia, and location of the centre of mass uncertainty are negligible. The inaccuracies induced by the uncertainty on the ground reaction components Rx and Rz are of relatively less importance.
> "On the basis of the results obtained in 13 tests on walking upstairs and 5 tests on level walking, the further following consideration can be made. When the speed of progression of the gait goes beyond certain values, depending on the locomotory act performed, the inaccuracy on the moments is significantly affected by the acceleration variables errors."
> Cappozzo, et al., (1975) [22], page 317

Cappozzo's error bound estimates reflect how well the experimental data fits his five-term Fourier series "model" of gait, and not the absolute accuracy of the collected data.

$$\sum \text{Forces} = F_p + mg - ma + F_d = 0$$

$$\sum \text{Moments} = T_p - I\dot{w} - w \times Iw + F_p \times r_p + T_d + F_d \times r_d = 0$$

Where:

m = the segment mass
I = the segment inertia tensor
g = the global gravity vector
a = the global acceleration vector
$\dot{w}$ = the global angular acceleration vector
w = the global angular velocity vector
F_p = the proximal force vector
F_d = the distal force vector
T_p = the proximal torque vector
T_d = the distal torque vector
r_p = the vector from the BCS origin to the
    point of application of the proximal force
r_d = the vector from the BCS origin to the
    point of application of the distal force
x = the vector cross product operator

Figure 111. Dynamic estimation equations

To begin the error contribution examination, the
equations used to estimate the dynamics are presented again
in Figure 111. From the inverted pendulum study described
earlier, the error bound around zero force (acting on the
free end during "flight") was found to be plus or minus 1 Nt
(see Figure 105 in Chapter 4). This was generated largely
from the residual noise in the position measurements not
removed by filtering, since there were no soft tissue motion
or joint axes to be resolved, and the mass and inertial
properties were calculated to a much higher degree of
precision than is possible to obtain for human segments (at
least with higher precision than is possible for segment
inertias determined from the population-based literature).
The weight of the bar was 57.4 Nt, resulting in an error
magnitude of about 2 percent of the weight of the segment,
or equivalently, an acceleration error equal to 2 percent of
gravity. Normal gait can produce peak translational
accelerations of approximately 10 meters per second squared,
or 1 gravity. Errors equal to 2 percent of the signal peak
agree well with the residual acceleration noise remaining
after filtering (Figure 109 in Chapter 4, and also the last
section of Chapter 3).

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
L thigh [m]	.4343						.461	.432
L shank	.4089						.455	.434
L foot	.0813						.081	*
M thigh [kg]	7.014	7.581	7.298		7.302	7.437	6.550	6.550
M shank	3.260	3.401	3.047		2.895	3.286	2.979	2.979
M foot	.992	1.204	1.063		.917	1.030	.950	.950
cg thigh[%]	43.3	44.0	37.2		38.8	38.8	42.9	46.8
cg shank	43.3	42.0	37.1		42.2	40.5	43.1	45.2
cg foot	42.9	44.4	44.9		43.7	43.2	66.8	-
Ix thigh [kg-m]				.0587				.0872+
Iy thigh				.020				.0184+
Iz thigh	.109(.086)			.0587	.1281	.1186	.0872	.0872
Ix shank				.0465				.0429+
Iy shank				.0038				.0030+
Iz shank	.0423(.0144)			.0465	.0424	.0466	.0429	.0429
Ix foot				.0058				.006+
Iy foot				.0056				.006+
Iz foot	.003(.0005)			.0058	.0032	.0044	.0060	.006

(1) Dempster (as reported by Williams and Lissner [158], and Vaughn [155]) (values in parentheses are Dempster's tolerance) subject mass 70.92 kg
(2) Braune and Fischer (from Williams and Lissner [158])
(3) Clauser (from Williams and Lissner [158])
(4) Miller and Nelson (from Williams and Lissner [158])
(5) Chandler (as reported by Vaughn [155])
(6) Vaughn (optimized from long jump) [155] subject mass 70.99 kg, 1.75 m tall
(7) Contini [34] subject mass 71.48 kg, 1.88 m tall
(8) Values used here for subjects with body mass 71.48 kg, 1.88 m tall

* This value (foot limb length) is not needed, the forceplate provides a vector from the foot body coordinate system to the distal force during stance.

"L cg" is the ratio of the length from the cg, to: the total limb segment length.

Table 12. Mass and inertial properties of the thigh, shank, and foot

Contini's [35] tables (used here to estimate segment mass and inertia properties, and included again in Table 12) show that one standard deviation for masses and inertias is always less than 20 percent. (A range of plus or minus one standard deviation for a normally distributed stochastic variable includes 68 percent of the samples). This results in an additional error of 20 percent of the weight of the segment.

A third contribution arises from errors in the segment mass times the segment's translational acceleration. As stated earlier the maximum amplitude of acceleration during gait is approximately 1 gravity, which results in a maximum error contribution of an additional 20 percent of the weight of the segment.

To gain a feeling for the relative importance of these errors to the final result, an examination of the contribution to the final results shows that the inertial loads always contribute less than 10 percent of the total load during the stance phase of gait. The gravity terms (due to the leg holding the rest of the body off the floor) dominate and are included in the data from the foot-floor force interaction.

The fourth term is the contribution of error from the next most distal segment, which results in an accumulation of errors as the dyanmics calculations proceed along the kinematic chain. The first segment (the foot) will only have the errors from the forceplate data added. The forceplatform manufacturer claims a 0.3 percent accuracy throughout the load range of the platform. Using 0.3 percent of the highest load experienced in gait (approximately 1000 Nt) yields an additional contribution for the foot forces of 3 Nt; however, this term will grow as the calculations proceed from segment to segment.

Finally, the possible influence of soft tissue motion can be estimated from the results of the "heel-bounce" experiment. Assuming half of each segment's total segment mass oscillates with the amplitude and frequency shown in Figure 94 in Chapter 4, the following forces will be generated. For the thigh, half of 6.5 is 3.2 kg, oscillating at about 12 Hz with an amplitude of 1-mm yields 36 Nt. The shank has a mass of 3 kg and appeared to oscillate 1.5 mm at 20 Hz, producing 35 Nt. Finally, the foot's mass is 1 kg, producing a force of about 8 Nt. These possible oscillatory forces will only occur immediately subsequent to heel strike, and Figure 94 shows they decay rapidly as a function of time, so these "possible" forces represent a conservative estimate of the influence of soft tissue motion on the total dynamic results. How much of the

heel-strike motion is real motion of portions of the lower extremity, and how much is motion of the LED markers with respect to the segment? The forceplate record from the same experiment can lend some insight into that question. Immediately subsequent to heel contact, there is an oscillating force recorded with an amplitude of about 15 Nt, and a frequency between 15 and 20 Hz. This is the total result of soft tissue oscillation in the whole body just after heel contact. The estimated contribution from one leg alone (the sum of foot, shank, and thigh, above) is about 80 Nt; therefore, the majority of the motion observed in the heel bounce experiment results from markers moving with respect to the segments, rather than soft tissue itself moving with the markers retaining kinematic fidelity with the tissue. The resulting possible contribution to the dynamic errors are 36, 35, and 8 Newtons for the thigh, shank, and foot, respectively; however, these relatively high frequencies of marker motion with respect to the segments will only be generated at heel contact. The majority of other soft tissue influence will be at substantially lower frequencies (on the order of the 1 Hz fundamental of gait), and those soft tissue contributions to the error bounds are very conservative. This is demonstrated by the rapid decay of the 15- to 20-Hz motion in time shown in Figure 94. Figure 112 contains the error

contributions for the proximal end of the foot, shank, and thigh (ankle, knee, and hip, respectively).

FORCE ERRORS

$$\sum E_f = (m \times E_a) + (g \times E_m) + (a \times E_m) + E_{F_d}$$

E_m = mass errors

E_m = 20% of the mass (from Contini)

E_a = acceleration errors

E_a = 2% of gravity (from the inverted pendulum)

E_{F_d} = accumulation of distal force errors

E_{F_d} = 0.3% of the Forceplate load

m = 6.5 kg maximum

a = 10 m/sec/sec maximum

F_d = 1000 nt maximum

MAXIMUM ERRORS ON THE PROXIMAL JOINT

[Nt]	Foot(ankle)	Skank(knee)	Thigh(hip)
g E_m	1.9	5.8	13.0
a E_m	1.9	5.8	13.0
m E_a	0.2	0.6	1.3
	4.0	12.2	27.3
E_{F_d}	3.0	7.0	19.2
Total	7.0	19.2	46.5 [Nt]
+@Heel		8.0	43.0
Strike	8.0	35.0	36.0
	15.0	63.0	125.5 [Nt]

Figure 112. Force error contributions

To complete the determination of the bounds of
confidence in the dynamic results, the estimated moments
must also be considered. Once again from the inverted
pendulum study, the error bound around zero moment (on the
free end during "flight") was found to be +/- 15 Nt-m, due
to residual noise (see Figure 106 Chapter 4). Just as for
the forces, the inertia tensor was known to a very high
degree of precision for the inverted pendulum (compared to
the precision in the data for humans), and the dynamics
results errors from that experiment are all attributed to
noise in the kinematic data and its effect on both the
second and third terms of the sum of moments equation in
Figure 113. To determine the influence of differentiated
position noise on other experiments, the error was divided
by the moment of inertia (because both the second and third
terms are linearly related to moment of inertia) yielding an
effective acceleration amplitude of 20
radians-per-second-squared from both the "omega dot" angular
acceleration term, and the "omega cross I omega" angular
velocity term. This amplitude agrees well with Figure 110
when the effect of both acceleration and velocity as well as
the frequency "side bands" are taken into account. The
resulting error-bound contribution from noise is thus
20-radians-per-second squared times the moment of inertia of
the segment being analyzed. Using the population-based data
once again, an error bound of 40 percent (to be

conservative) was applied to the inertia values. To be able to determine the contribution of inertia errors to the total error bound, the total angular velocity and acceleration must be known. Typical maximum angular velocity amplitude in gait is approximately 3 radians per second, and typical maximum angular acceleration amplitude is 100-radians-per-second squared. So this error contribution is the moment of inertia error (40 percent), times the maximum acceleration, plus the maximum velocity squared. In all of these error determinations the maximum values for all of the terms are used to produce truly ungenerous error bounds.

Static positional and geometrical errors in the joint axis location are less than or equal to 1 cm, producing an additional error equal to the error in the vector, to the joint center, times the force acting on that joint. This really should be a vector cross product; however, a simple multiplication of the two vector magnitudes will produce an error bound greater than or equal to the real error.

Just as for the error bounds on the forces, there is an accumulation of errors as the calculations traverse along the kinematic chain. The most obvious term is the error in the torque acting on a segment from the next most distal one. For the foot, the only contribution to this term is from the measured moments from the forceplate, which again,

are equal to 0.3 percent of the maximum. Typical maximum measured forceplate moments are 2.5 Newton-meters, resulting in an error-bound contribution of 0.008 Nt-m for the foot. The total error in the moment estimate at the proximal end of the foot will be added to the next link's (the shank) moment error-bound calculation, and so on along the linkage.

There is also a moment error term generated from the cross product of the vector to the joint center and the error in the force on the joint. To maintain the uncharitable nature of the error-bound calculation, a simple product of the magnitudes of the vectors will be used. A typical value for the length of the vector from the segment's body-coordinate system (BCS) origin to the joint center (point of interest on the joint axis) is approximately 0.2 meters. The force errors used in this calculation come from the earlier discussion of forces. Figure 113 aggregates the moment error-bound calculation information and results for the proximal ends of the foot (ankle), shank (knee), and thigh (hip).

MOMENT ERRORS

$$\sum E_M = (I\ E_{\dot{w}}) + (I\ 2E_w) + (E_I\ \dot{w}) + (E_I\ w^2) + (F_d\ E_{r_d}) + (E_{F_d}\ r_d) + E_{T_d}$$

E_I = moment of inertia errors

E_I = 40% of moment of inertia (from Contini)

$(E_{\dot{w}} + 2E_w)$ = aggregated acceleration errors

$(E_{\dot{w}} + 2E_w)$ = 20 radians/sec^2 (from the inv. pendulum)

E_{r_d} = errors in location of the joint centers

E_{r_d} = 1 cm

E_{T_d} = accumulation of distal moment errors

E_{T_d} = 0.3% of the Forceplate moment

$\dot{w}$ = 100 radians/sec^2 maximum

w = 3 radians/sec maximum

I = 0.09 kg-m^2 maximum

r_d = 0.2 m maximum

F_d = 1000 Nt maximum

MAXIMUM ERRORS ON THE PROXIMAL JOINT

[Nt-m]	Foot(ankle)	Shank(knee)	Thigh(hip)
$E_I\ \dot{w}$	0.24	1.72	3.50
$E_I\ w^2$	0.024	0.172	0.35
$(E_{\dot{w}} + 2E_w)\,I$	0.12	0.86	1.75
$E_{F_d}\ r_d$	0.60	1.40	3.84
$F_d\ E_{r_d}$	1.0	1.0	1.0
	1.98	5.15	10.44
E_{T_d}	0.008	1.99	7.14
	1.99	7.14	17.58
+@Heel			1.6
Strike		1.6	7.0
		8.7	26.2 [Nt-m]

Figure 113. Moment error contributions

As a comparison Cappozzo, et al., (1975) [22] estimated an error bound on hip moments between +/- 30 Nt-m and +/- 40 Nt-m. This, however, should not be taken as a comparison of the quality of the two techniques. Cappozzo's error bound is a representation of how well his experimental data fit the 5-Fourier-series terms he chose, not how well his data represent reality. The error bounds produced as part of this investigation are a representation of the absolute accuracy of the data, based on determinations of the accuracy of the elements of the kinematic acquisition, and dynamic estimation system.

EXPERIMENTAL PROTOCOL

A complete gait analysis experiment was conducted on two subjects, including kinematic analysis of marker arrays attached to the foot, shank, thigh, and pelvis, along with foot-floor force data. Before any experimentation was conducted, the subject read and signed, the "Use of Humans as Experimental Subjects" consent form. Copies of the forms used and signed can be found in Appendix F. The experiment proceeded by having the subject dress in a skin-tight, non-reflective, black nylon, body stocking. A black knit glove extending to the armpit for the right arm was also used. All of the black apparrel was used to minimize the infra-red reflections from the LED kinematic markers. The

four segments with LED marker arrays were then placed on the subject. The mounting arrangement for each of these has been explained previously. This resulted in four instrumented segments: the foot; shank; thigh; and pelvis of the right, side lower extremity. A small plexiglas backboard with the LED Control Unit (LCU) to power and multiplex the LEDs was placed over the subject's shoulders and on his back. Refer to Photograph 16 (Chapter 4) for a view of an instrumented subject. Even though the option does exist with the Selspot kinematic acquisition system for battery power and optical synchronization of the LEDs, the hard-wire umbilical was used throughout these experiments to ensure the previously calibrated accuracy. This meant that the subject towed nine stranded wires in a single unconstrained flexible cable less than 6 mm in diameter.

All gait experiments used only one instrumented leg, always the right. Since this study was the first measured gait with the fully developed TRACK system, one leg was felt to be a useful beginning. Studies involving more segments, multiple forceplates, or instrumented shoes, and even more LEDs (with the Selspot II) will follow naturally from these experiments. Certainly instrumenting two legs has many advantages, including the ability to compare results from both, as well as providing an internal verification on dynamic results in much the same way that the inverted

pendulum force results were used earlier, e.g., establishing through dynamic linkage calculation the forces and moments on the distal end of the swing phase leg.

All experiments were conducted with the subject barefoot. The most distal LED array is mounted directly to the foot with no shoe. This enabled the subjects to walk as naturally and normally as possible without the influence of footwear style or condition.

The experiment itself began with the subject 20 to 25 feet from the viewing volume. As the subject approached the viewing volume (taking at least four full strides to reach it) the kinematic and forceplate data acquisition began. The subject then entered the viewing volume while continuing his gait, passed through it, and emerged from the other side. Only after the subject had passed completely out of the viewing volume did the data acquisiton cease. Thus in all of the collected data, there is a period of time at the beginning and the end where the subject is out of the viewing volume. The total duration of data sampling was usually 2.0 seconds, always at 315 Hz. This completed the gait data acquisition portion of the experiment. Photograph 17 shows an experiment in progress. The subject's LEDs are facing away from the Selspot cameras for illustration only. Table 13 is the full specification of all thirty LEDs on four segments used for gait analysis.

```
Name      Nchn   Nseg      Description
FSTP30    30.    4.        FOOT7,SHANK7,THIGH8,PELVI8

LED segment coordinate data (meters)
Segment   Channel        X              Y              Z
   1         1         0.0300         0.0000         0.0000
   1         2         0.0400         0.0300         0.0000
   1         3         0.0300         0.0600         0.0000
   1         4         0.0000         0.0700         0.0000
   1         5        -0.0300         0.0600         0.0000
   1         6        -0.0400         0.0300         0.0000
   1         7        -0.0300         0.0000         0.0000
   2         8         0.0000        -0.0300         0.0000
   2         9         0.0300        -0.0400         0.0000
   2        10         0.0600        -0.0300         0.0000
   2        11         0.0700         0.0000         0.0000
   2        12         0.0600         0.0300         0.0000
   2        13         0.0300         0.0400         0.0000
   2        14         0.0000         0.0300         0.0000
   3        15        -0.0350        -0.0500         0.0000
   3        16        -0.0450         0.0000         0.0000
   3        17        -0.0350         0.0500         0.0000
   3        18         0.0000         0.0650         0.0000
   3        19         0.0350         0.0500         0.0000
   3        20         0.0450         0.0000         0.0000
   3        21         0.0350        -0.0500         0.0000
   3        22         0.0000        -0.0650         0.0000
   4        23         0.0000         0.0500         0.0000
   4        24        -0.0800         0.0350         0.0000
   4        25        -0.1000         0.0000         0.0000
   4        26        -0.0800        -0.0350         0.0000
   4        27         0.0000        -0.0500         0.0000
   4        28         0.0800        -0.0350         0.0000
   4        29         0.0990         0.0000         0.0000
   4        30         0.0800         0.0350         0.0000
```

Table 13 All 4 LED array segment specifications for Gait

Photograph 17. Gait experiment

```
TK4:AP0432.DAT;0

                                                    09:23:38   4-APR-82

EXPERIMENTAL TRACK PARAMETERS:
SAMPLING PARAMETERS:
    Sampling frequency=        315.00
    Total number of frames =   630:         2.00 seconds of data
    Segment file name =    FSTP30:     30 channels,    4 segments
Camera positions: X        Y        Z        AX       AY       AZ [Degrees]
                                 [Meters]
    Camera 1    0.000    0.000    0.000    0.000    -26.000    0.000
    Camera 2    3.353    0.000    0.000    0.000    -26.000    0.000
Camera calibration factors:    Camera 1    Camera 2
    Focal distance (SU)         2315.16     2295.02
PROCESSING PARAMETERS:
    Forceplate scale factors:    X        Y        Z    [Newtons/Volt]
                                        1.00E+02   1.00E+01
    Skew ray max error= 10 [Selspot Units],   Errmax=  10.00 %
                                2.00E+01
Data Smoothing:     Interpolation    Low-pass Filter
    Raw Selspot     None             None
    3-D Selspot     Yes              15.0 Hz Cutoff
    BCS Selspot     Yes              15.0 Hz Cutoff
    Derivative      None Available   15.0 Hz Cutoff
    Forceplate      None Available   None
Global axes origin: ( 1.695,-0.617, 3.466} [meters]
Approximate data processing time: 22[min] 15[sec]
PROCESS RESULTS:
    Raw Data file storage name: AP0430  in Directory: TMAC02
    No time window
    Number of Selspot hardware bad points:                   7575  Frame-Channels
    Number of bad points eliminated by 3-D skew ray check:   7069  Frame-Channels
    Number of points eliminated by inter-LED length check:  10555  Frame-Channels
                                                   and:        901  Frame-Segments

    The worst inter-LED length error was:     3922.77%
```

Table 14.   Experimental parameters for a gait experiment

To be able to perform the geometrical analysis to determine the momentary axes of rotation (MAR), an ancillary set of experiments with the instrumented subject was required. This was generally performed after one or two gait experiments, as described above, were conducted. During the entire time from the start of the first gait trial through the geometry experiments, the subject was instructed not to adjust the position of the LED marker arrays on his limb segments. In order for the geometrical analysis to produce useful data to be used in conjunction with gait data, the LED arrays must not have been moved. The geometry data were accumulated by first having the subject stand stationary in the center of the camera's viewing volume with all joints of the instrumented lower extremity fully extended. Data were accumulated for 1 second to provide a full geometrical position and orientation reference. Then, with the subject still standing in the center of the viewing volume, facing in the direction of progression during gait, he was asked to flex and extend the hip only. The kinematic data for all four segments were recorded. He was then asked to flex and extend the knee, then the ankle. An independent set of data were accumulated for each joint flexure, and all motion was performed unloaded while standing on the opposite leg. Typically, one more gait session was then recorded, and the experimental session came to a close. Because the kinematic

data processing time was on the order of 10 to 15 minutes, one or two of the experiments were usually processed while the subject was still instrumented to ensure that valid data had been accumulated. A simple empirical algorithm has been developed to estimate the total off-line processing time of an experiment. The equation is:

$$Time=Frames*(.04+(Chan*F1*.02)+(Seg*F2*.02)+(FP*.04))$$

Where Time is in seconds, Frames are the number of frames of data accumulated, Chan is the number of LEDs used, Seg is the number of independent instrumented segments, and FP is zero if the forceplate is not being sampled, and 1 if it is used. F1 is zero, one, or two, depending on the number of raw or three-dimensional filters applied. F2 is zero, one, or two, depending on the number of result, or derivative filters applied. Thus an experiment accumulating 500 frames with thirty LEDs on four segments with all the low pass filters active, with forceplate data accumulation and reduction, will take approximately 13.3 minutes to completely process from the instant the experiment ended, including dynamics. The same experiment with the same data reducion, except without the filters, would take about 40 seconds to process. All of the off-line processing described in earlier sections was performed including the application of a 15-Hz low-pass filter to the kinematic data and its derivatives.

Many of the kinematic results were presented in earlier sections. The mass and inertial properties described in Chapter 4, as well as the segment geometry, were incorporated into the processing, and the dynamics estimator (NEWTON) produced the following results for one of the subjects.

DYNAMIC RESULTS

Figure 114 is a plot of the trajectories of the origins of the four body coordinate systems in a global Z = constant plane. Figure 115 is a plot of the vertical force measured by the forceplate. Gait timing information was taken from this record. The instrumented right foot stepped on the plate, and heel contact occurred at .511 seconds from the start of the experiment. Heel contact is taken to be at the time the forceplate vertical load began changing from zero. Right toe-off was found to be at 1.214 seconds by the same technique. Left heel contact can be seen as the slight break in the vertical force curve at 1.101 seconds. The next right heel contact can be seen at 1.703 seconds as the 10 Nt oscillations well after the plate was unloaded. From this timing information the cadence for this step was determined to be 100 steps per minute (where a step is a foot-fall of either foot). The subject's mass was 70.8 kg, and his height was 1.88 meters.

AP0432 Segment 1-4, w.r.t. Gnd. AP0430 w/15Hz Fltr Gait

Pelvis

Thigh

Shank

Foot

Y Displacement [Meters]

X Displacement [Meters]
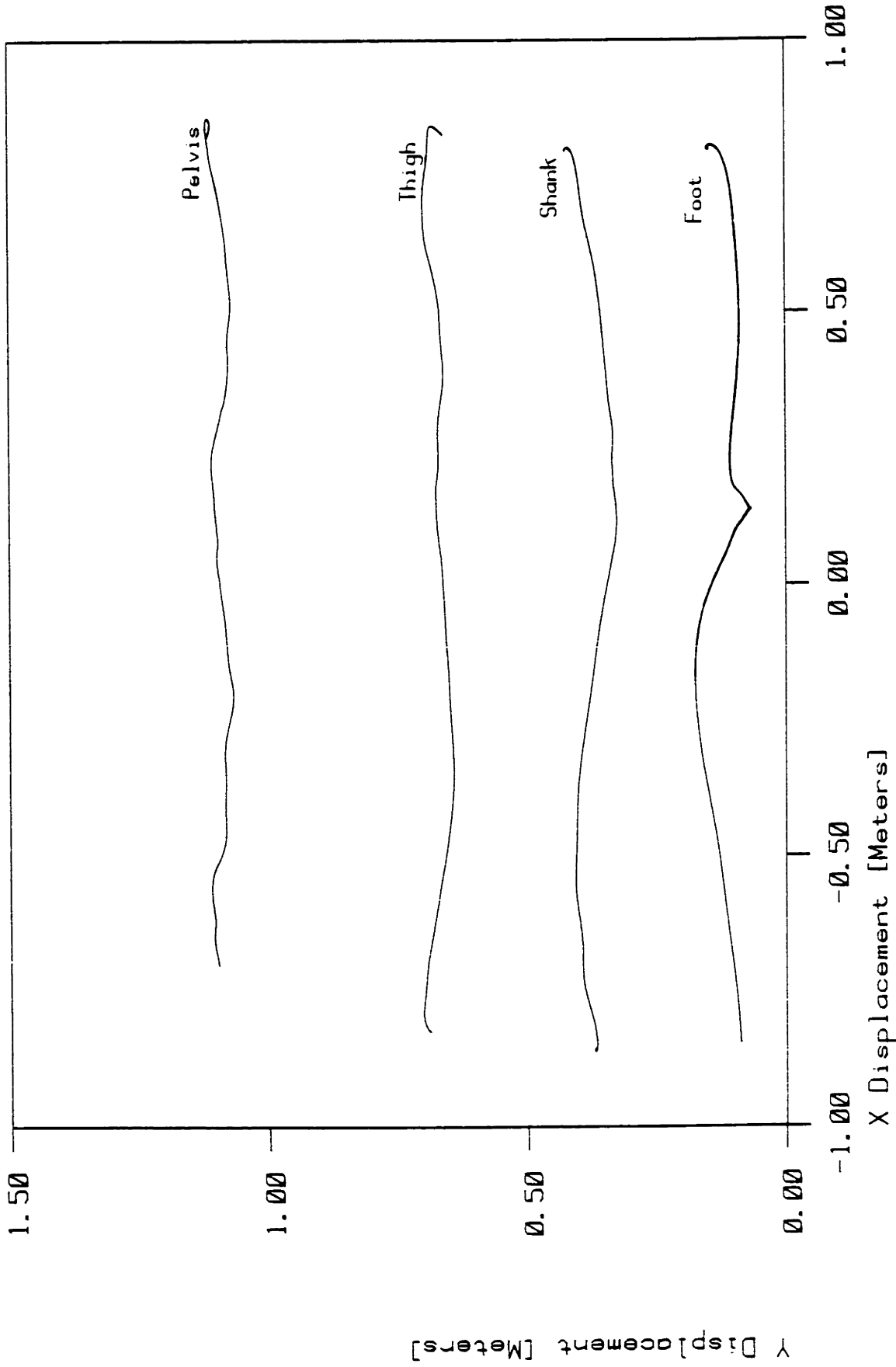
Figure 114. Motion of 4 BCSs in the Z = constant plane
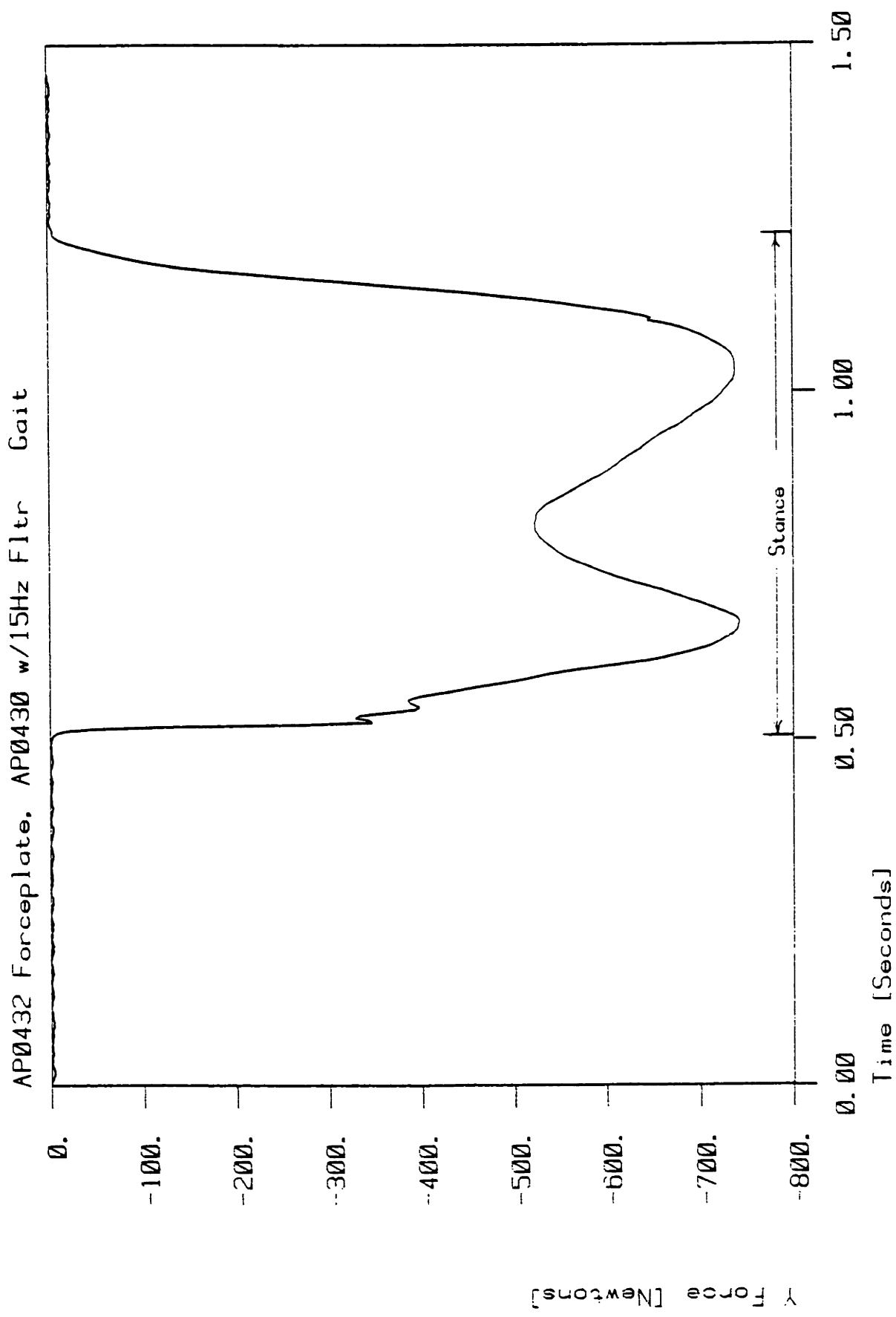
Figure 115.   Vertical force measured at the forceplate
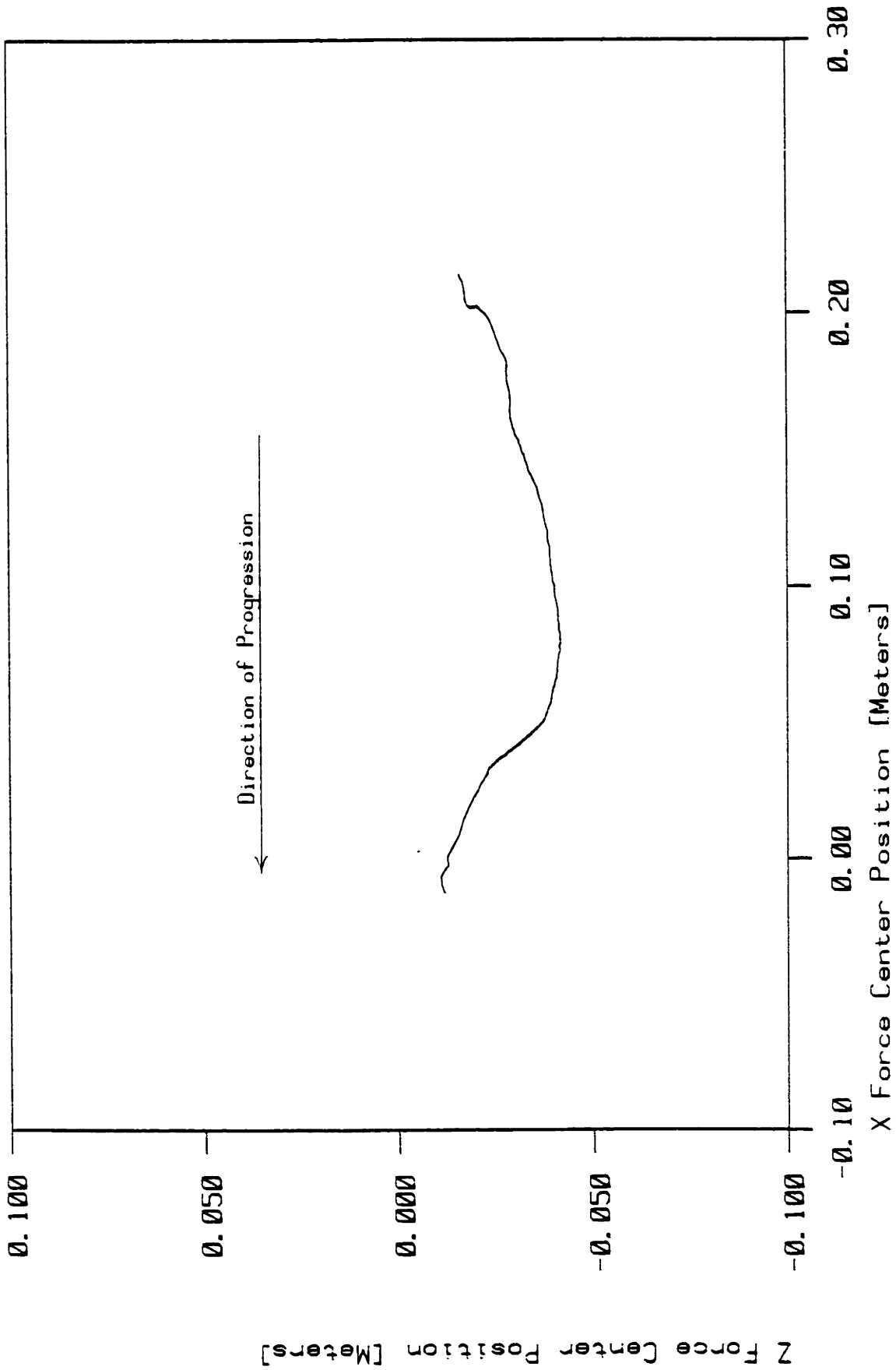
Figure 116. Path of the center of net force on the forceplate during stance

Figure 118 is a plot of the distal forces on the foot in the global coordinate system. This record is the negative of the data recorded by the forceplate. Figure 119 is a plot of the distal moments on the foot also in the global coordinate system.

The coordinate systems used throughout this thesis are arrayed as follows. The origin of the Global Coordinate system (GCS) is at the front node point of camera number 1. The X axis passes through the node of camera number 2. The Y axis is vertical, and Z is toward the viewing volume. See Figure 28 in Chapter 3. The body-coordinate system is similarly oriented. Positive Y is vertical when the lower extremity is fully extended. Positive X is posterior, and postiive Z is medial. Each segment is assigned a number as follows: the Foot is segment number 1, the shank is number 2, the thigh is number 3, and finally the pelvis is number 4. This numbering holds throughout this thesis and is shown in Figure 117.
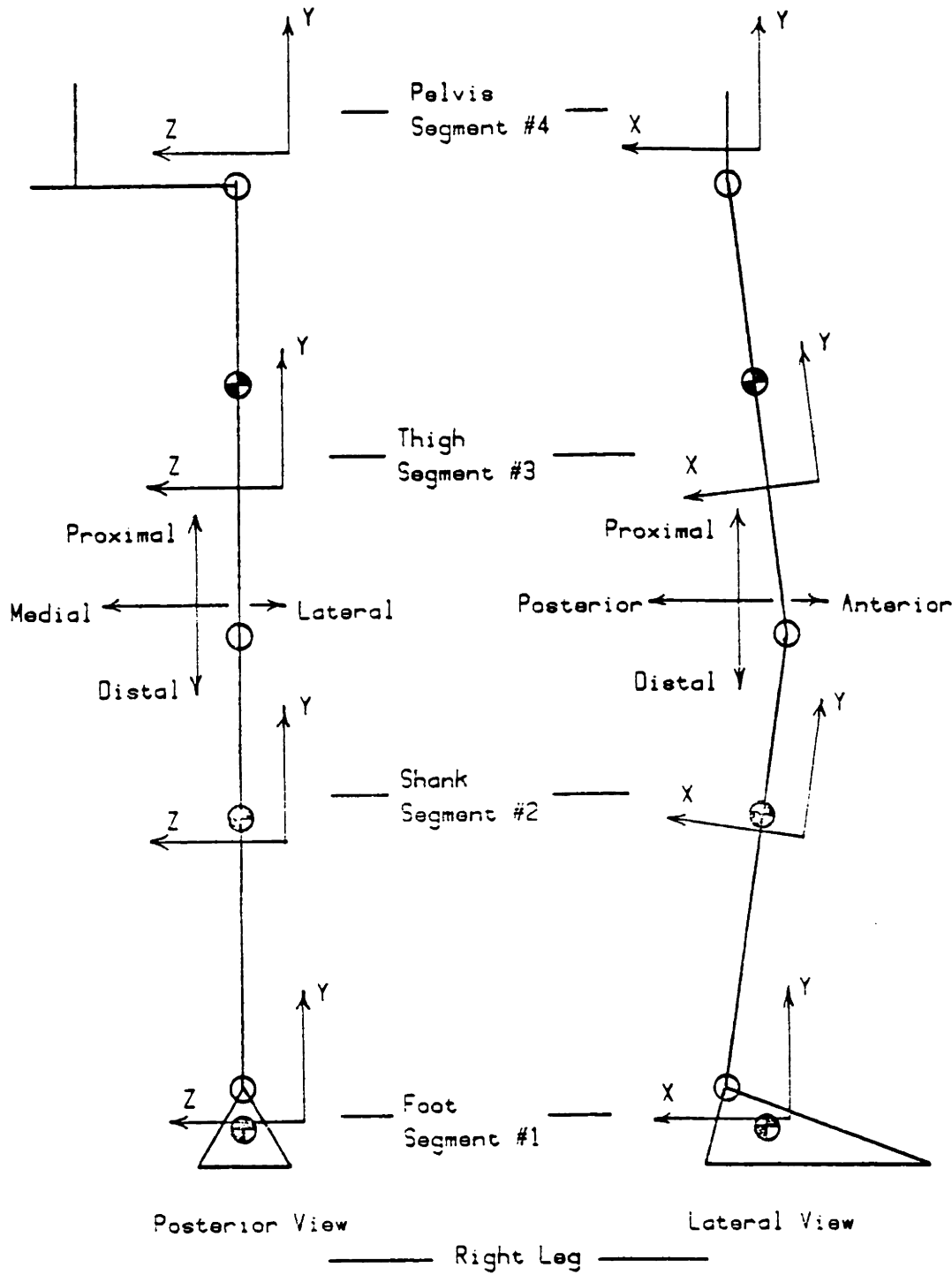
e. g. Flexion at the knee is represented by a positive rotation (right handed coordinate systems) of the shank (#2) with respect to the thigh (#3) (mostly about the Z axis).

Figure 117. Segment BCS arrangement and numbering
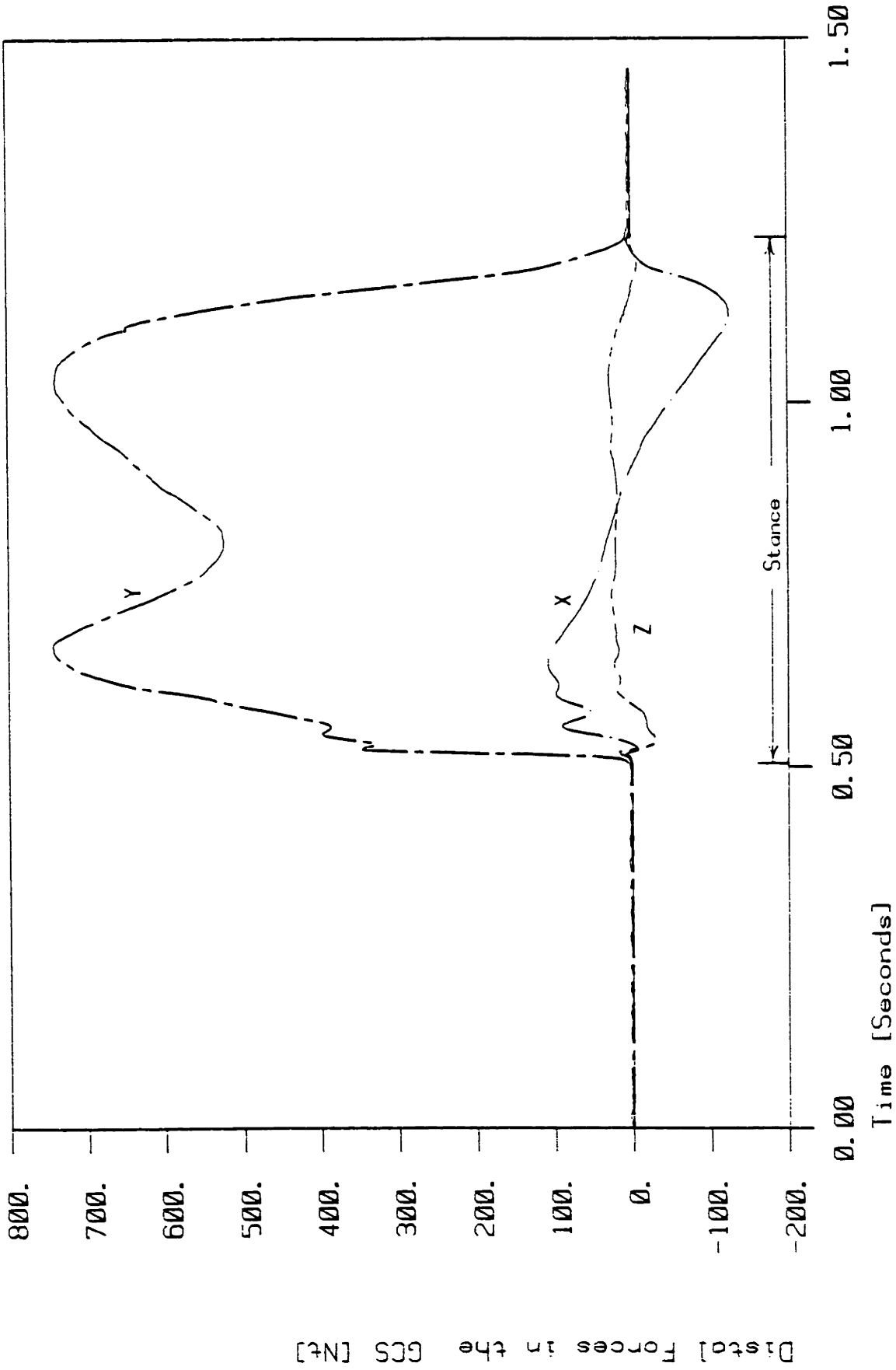
Figure 118. Global Coordinate System distal foot forces
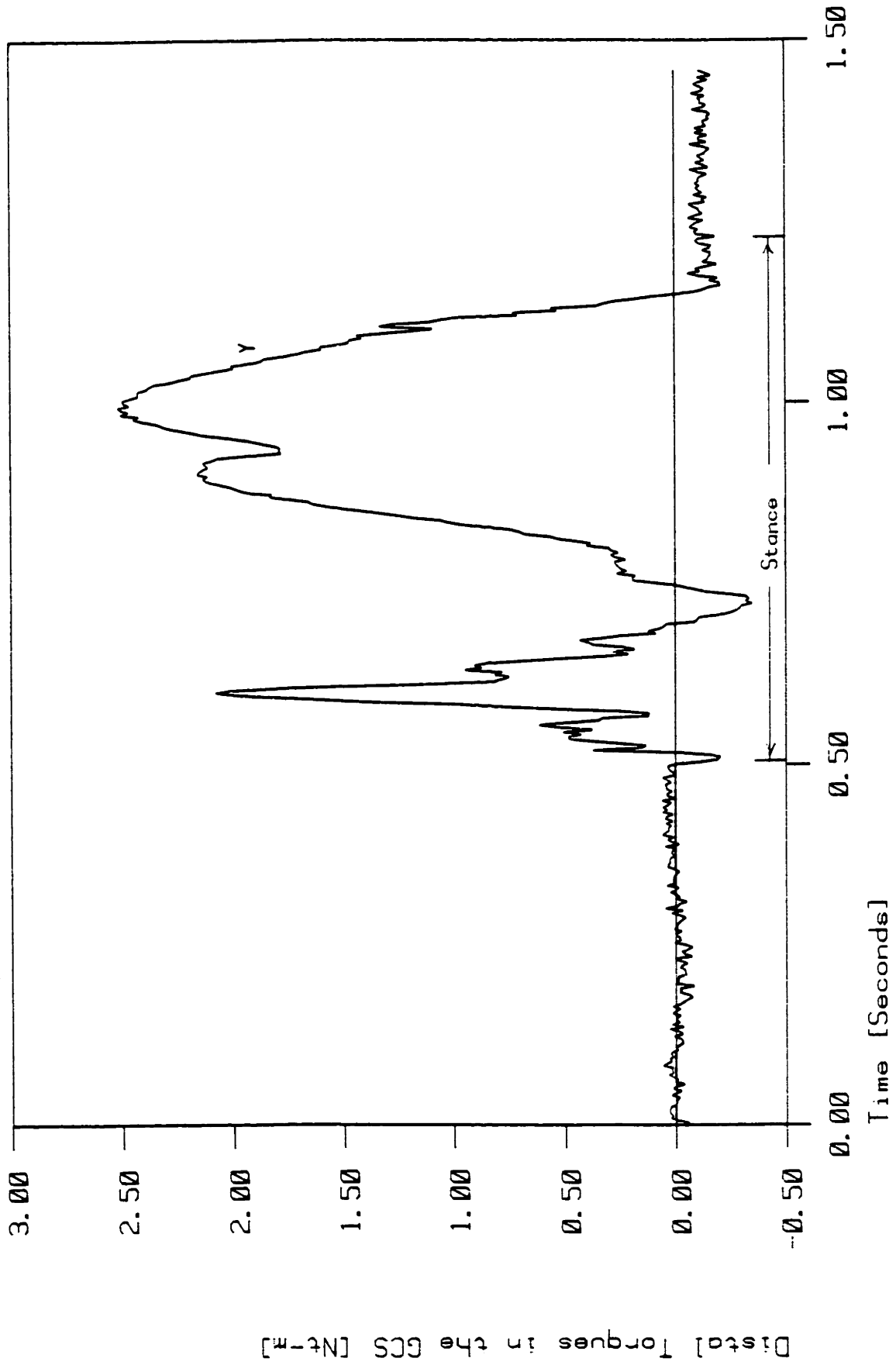
Figure 119. Global Coordinate System distal foot moments

There is only one non-zero moment distal to the foot, (the one about a vertical axis) because the foot cannot generate moments about the other two axes at the surface of a flat floor.

Despite the fact that data were collected for two subjects, the results for only one subject are presented here, because of difficulties encountered in determining the joint center vectors for the foot of the other subject.

Figures 120 and 121 are the forces and moments acting on the distal end of the shank segment (ankle) in the body-coordinate system (BCS) of the shank. Figures 122 and 123 are the forces and moments acting on the distal end of the thigh segment (knee) in the BCS of the thigh, and finally Figures 124 and 125 are the forces and moments acting on the "distal end" of the pelvis (hip) in the BCS of the pelvis.
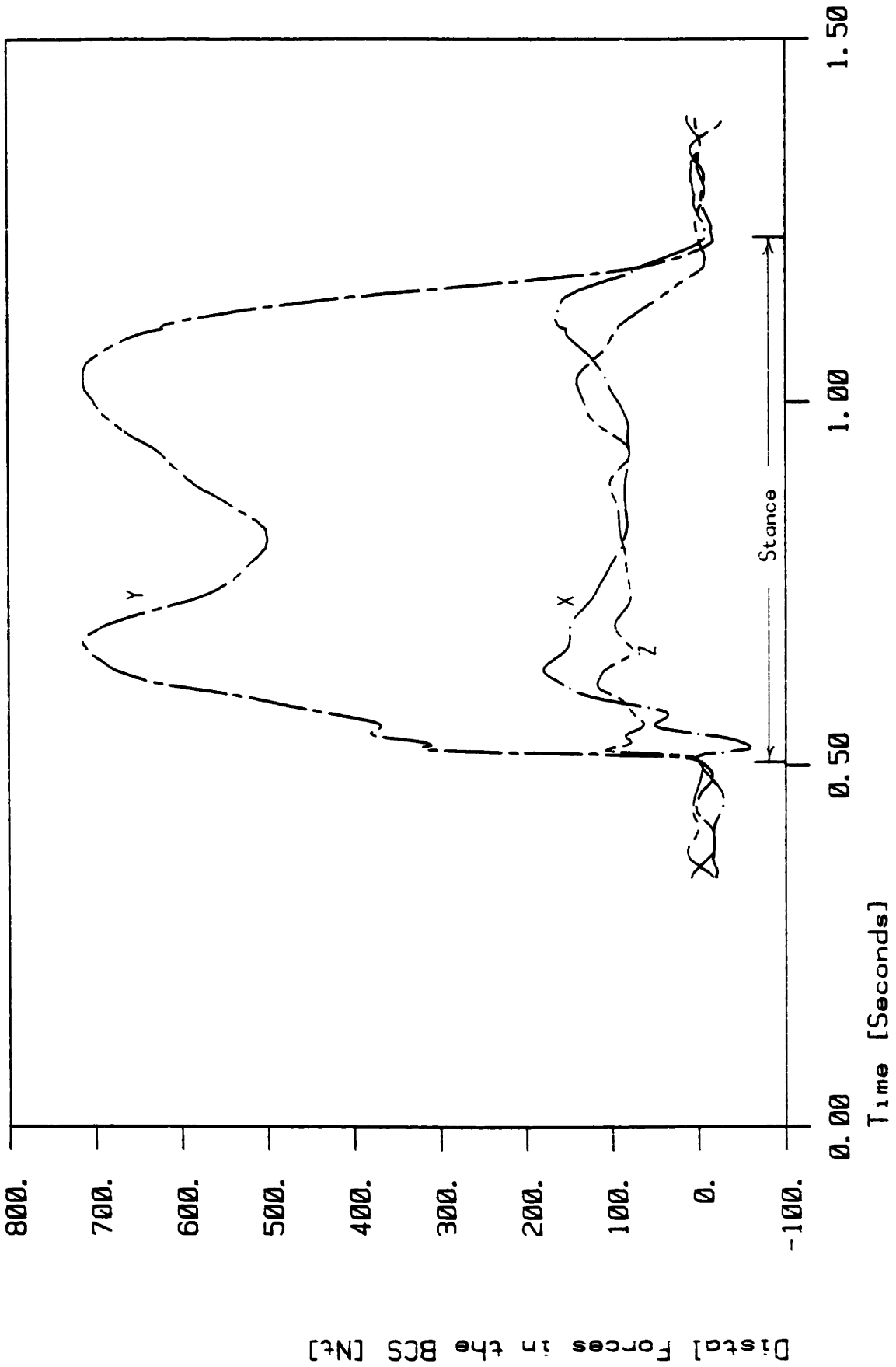
Figure 120.   Forces acting on the distal end of the shank
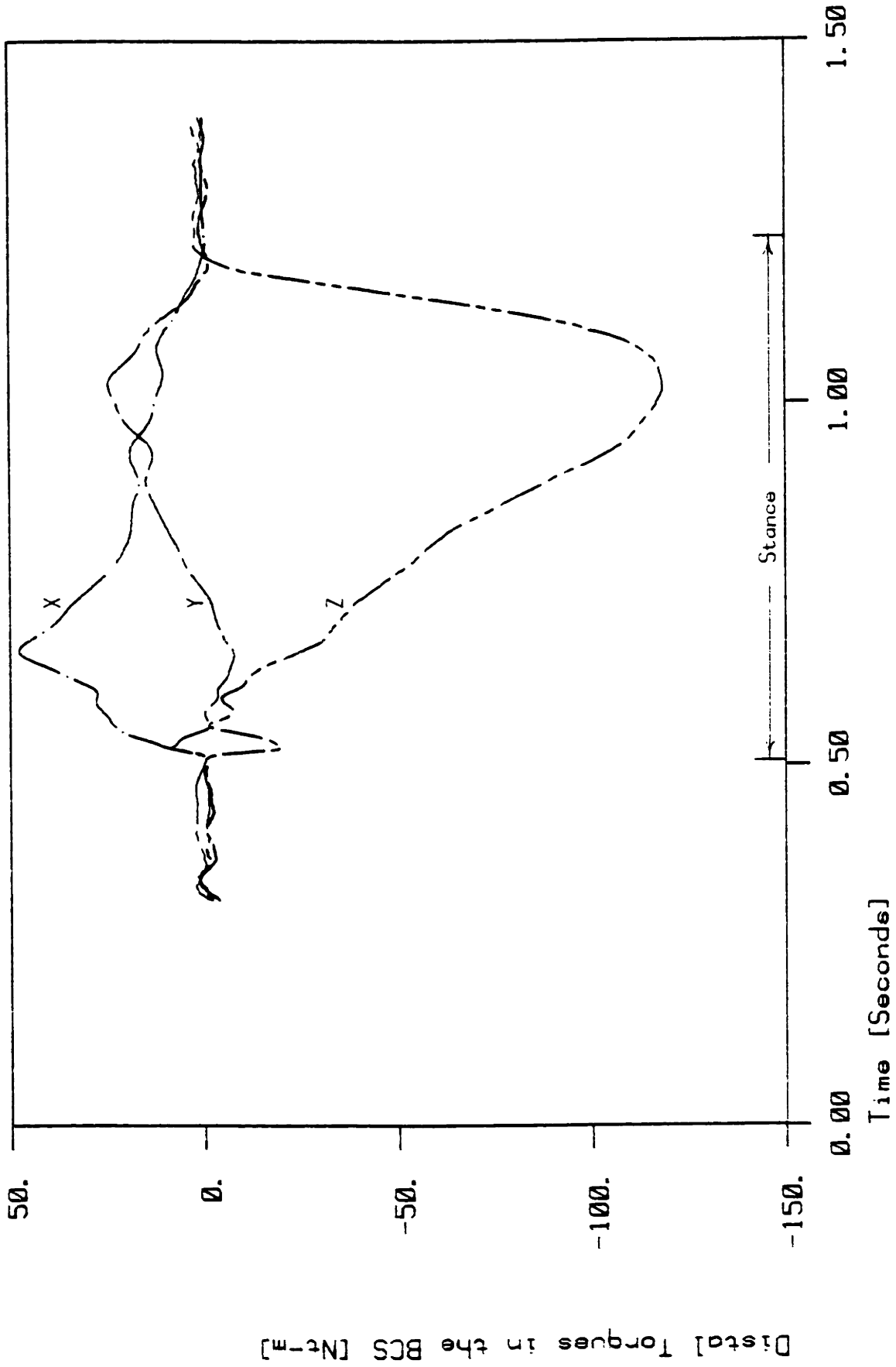
Figure 121. Moments acting on the distal end of the shank
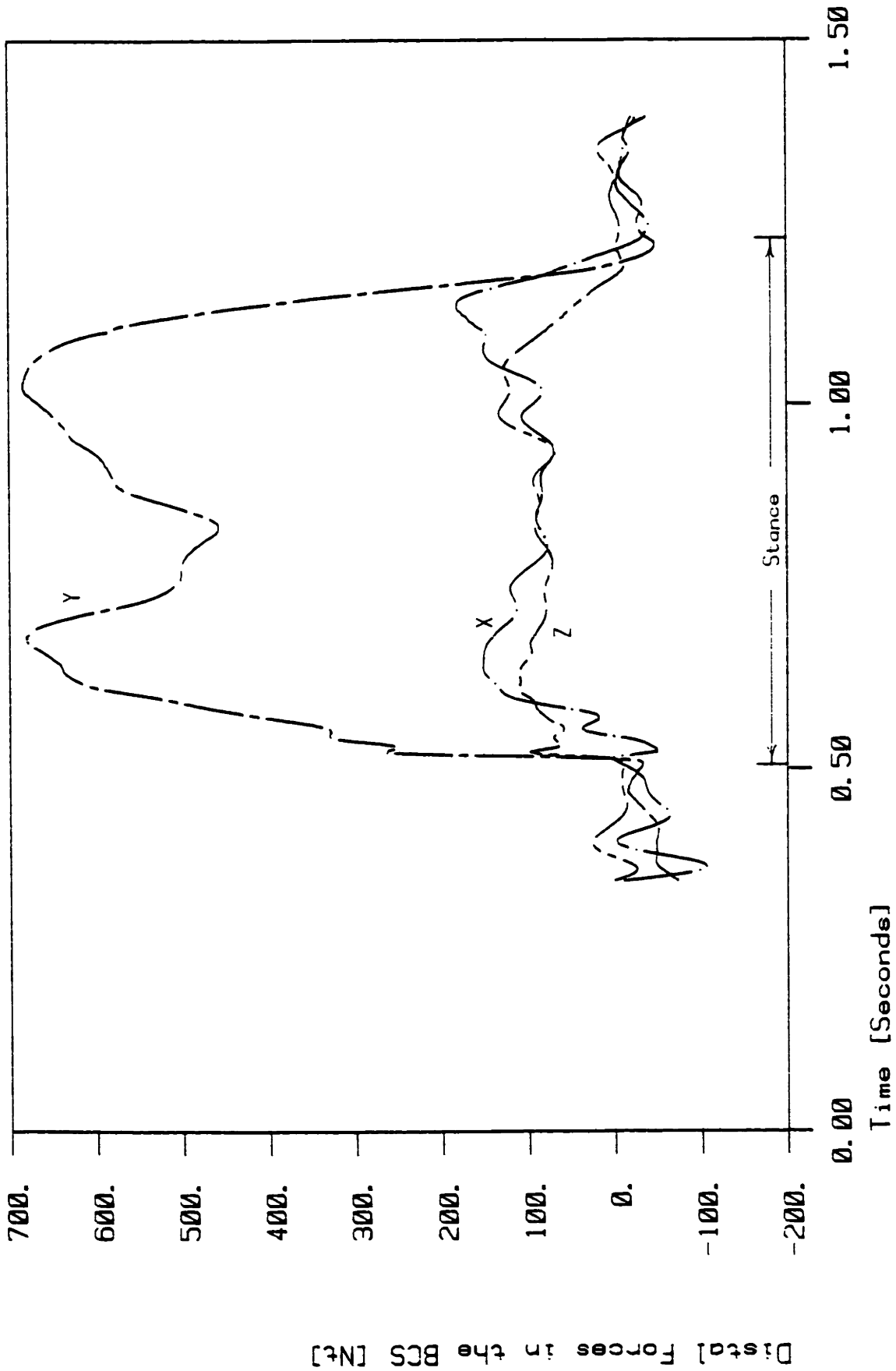
Figure 122.   Forces acting on the distal end of the thigh
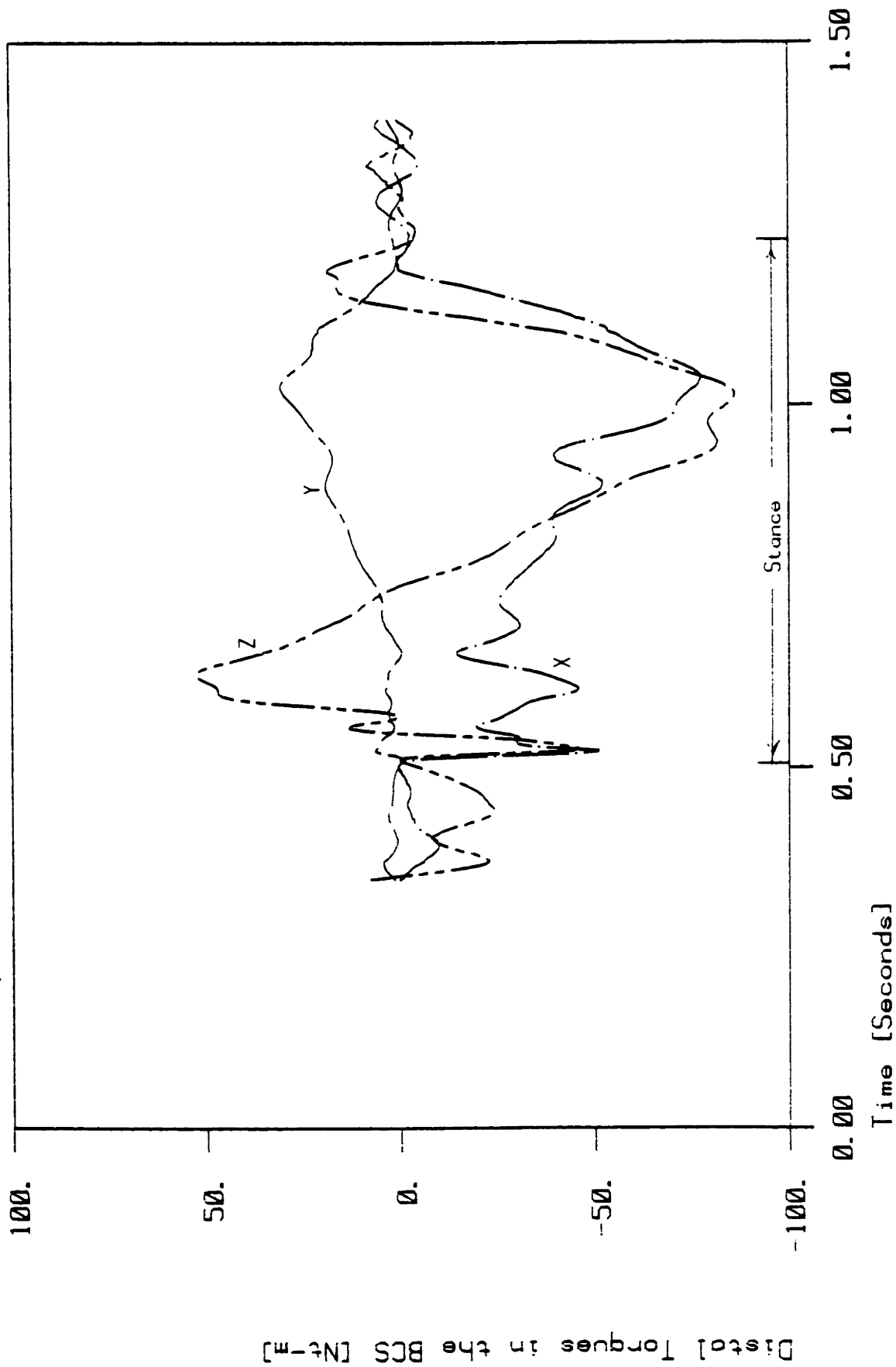
Figure 123. Moments acting on the distal end of the thigh
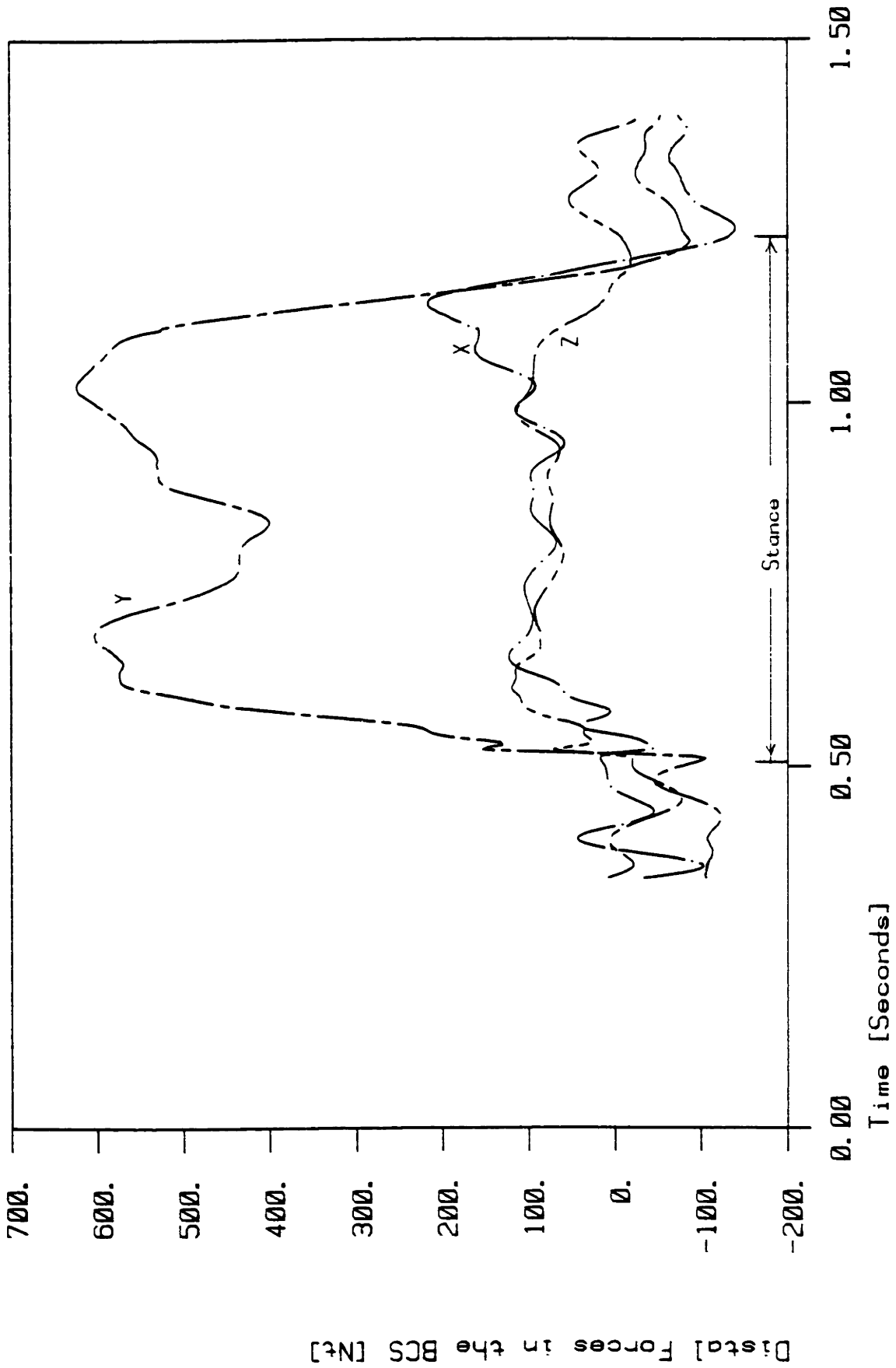
Figure 124.   Forces acting on the distal end of the pelvis (hip)

Figure 125. Moments acting on the distal end of the pelvis (hip)

As a comparison, the results of Cappozzo, et al., (1975) [22] for the moments acting at the ankle, knee, and hip, are included in Figures 126 through 131, with their bound of confidence. Since they do not explain the coordinate systems they use, it can only be assumed that the moments they resolve are about the medial-lateral, flexion-extension axis, the Z axis used here.
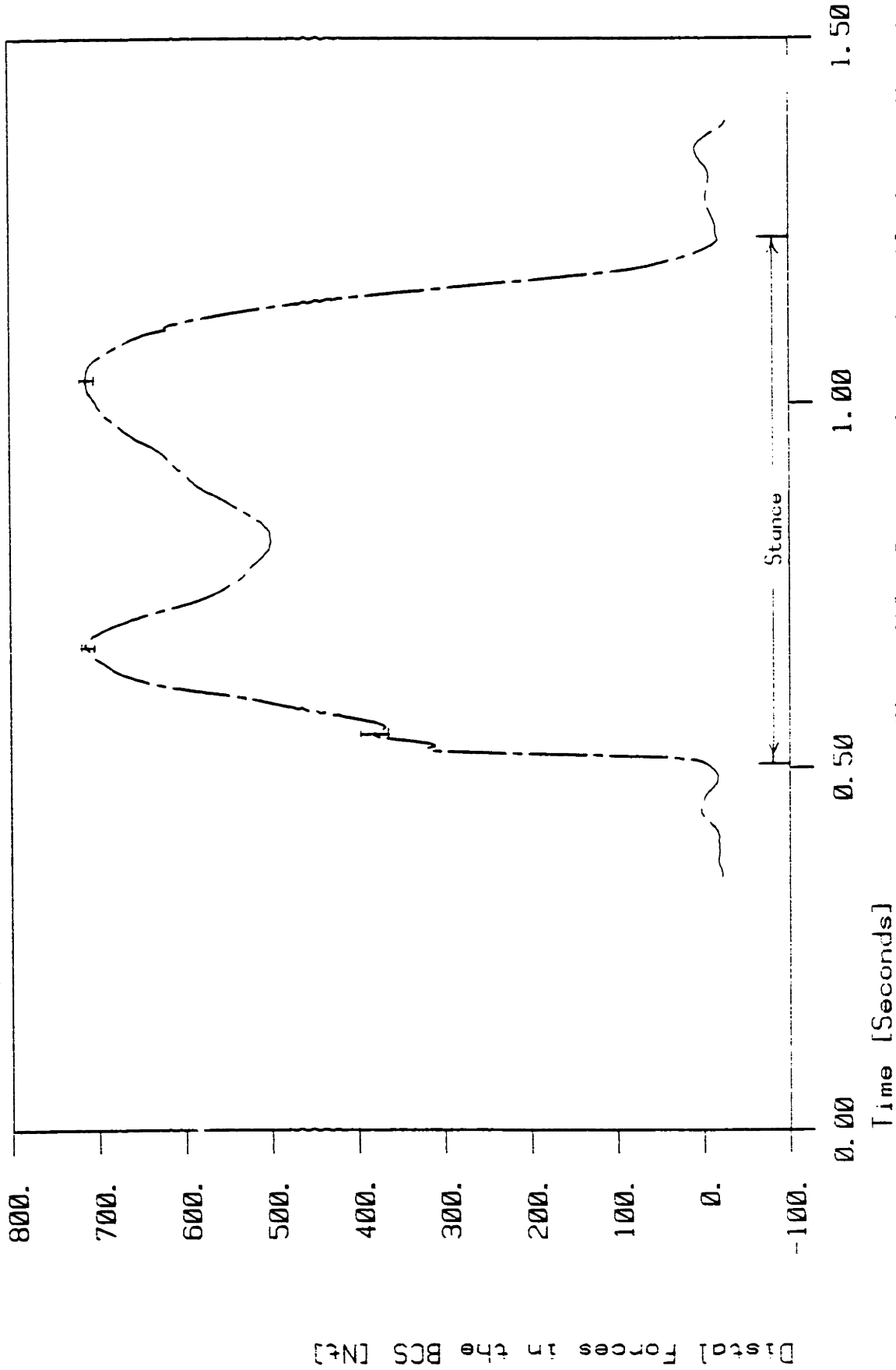
Figure 126. Forces acting at the ankle in the direction of the long axis of the shank (y axis) with sample error bounds
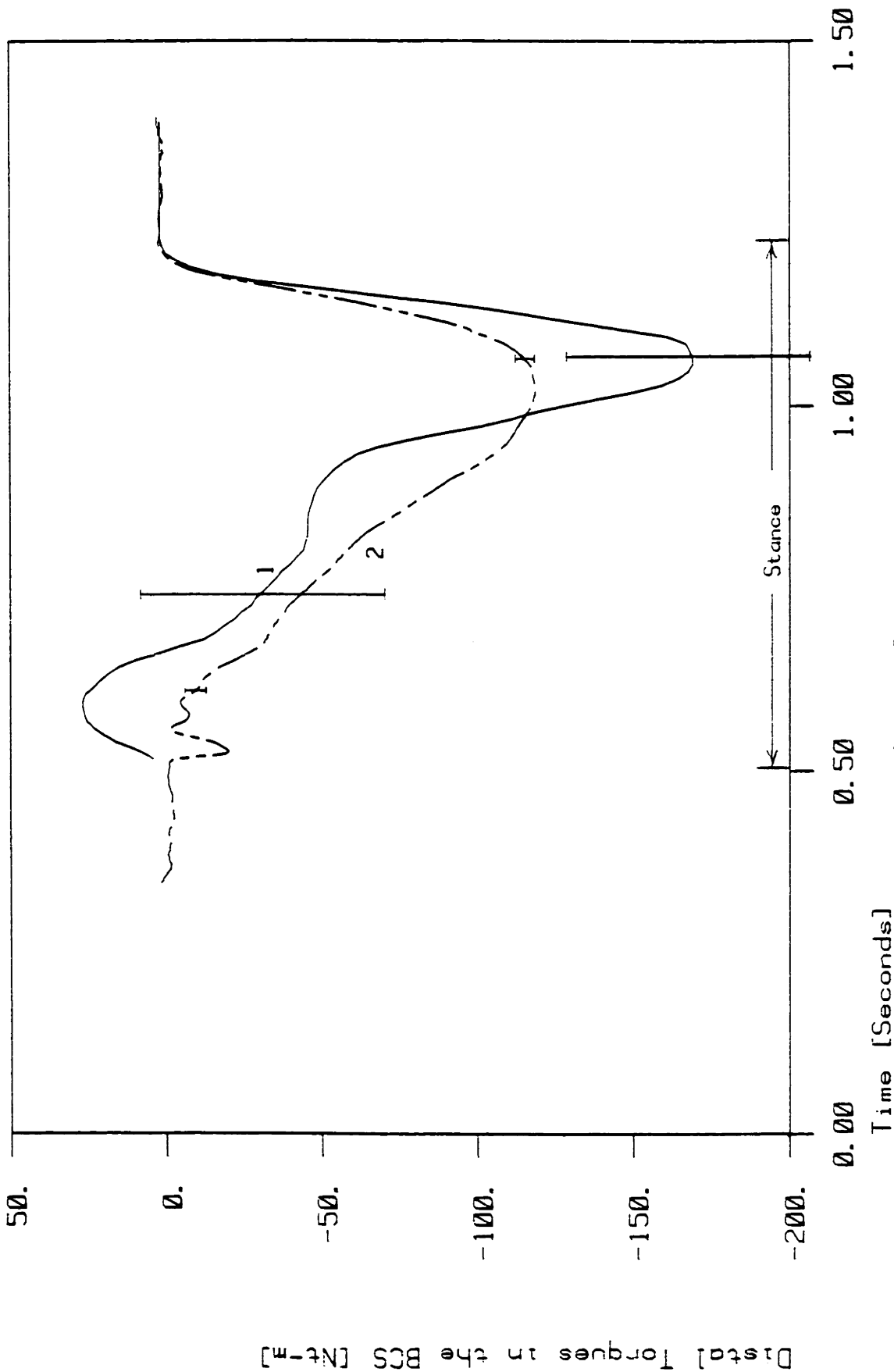
Figure 127. Moments acting about the flexion - extension
axis of the ankle, with error bounds: 1) Cappozzo, 2) Antonsson
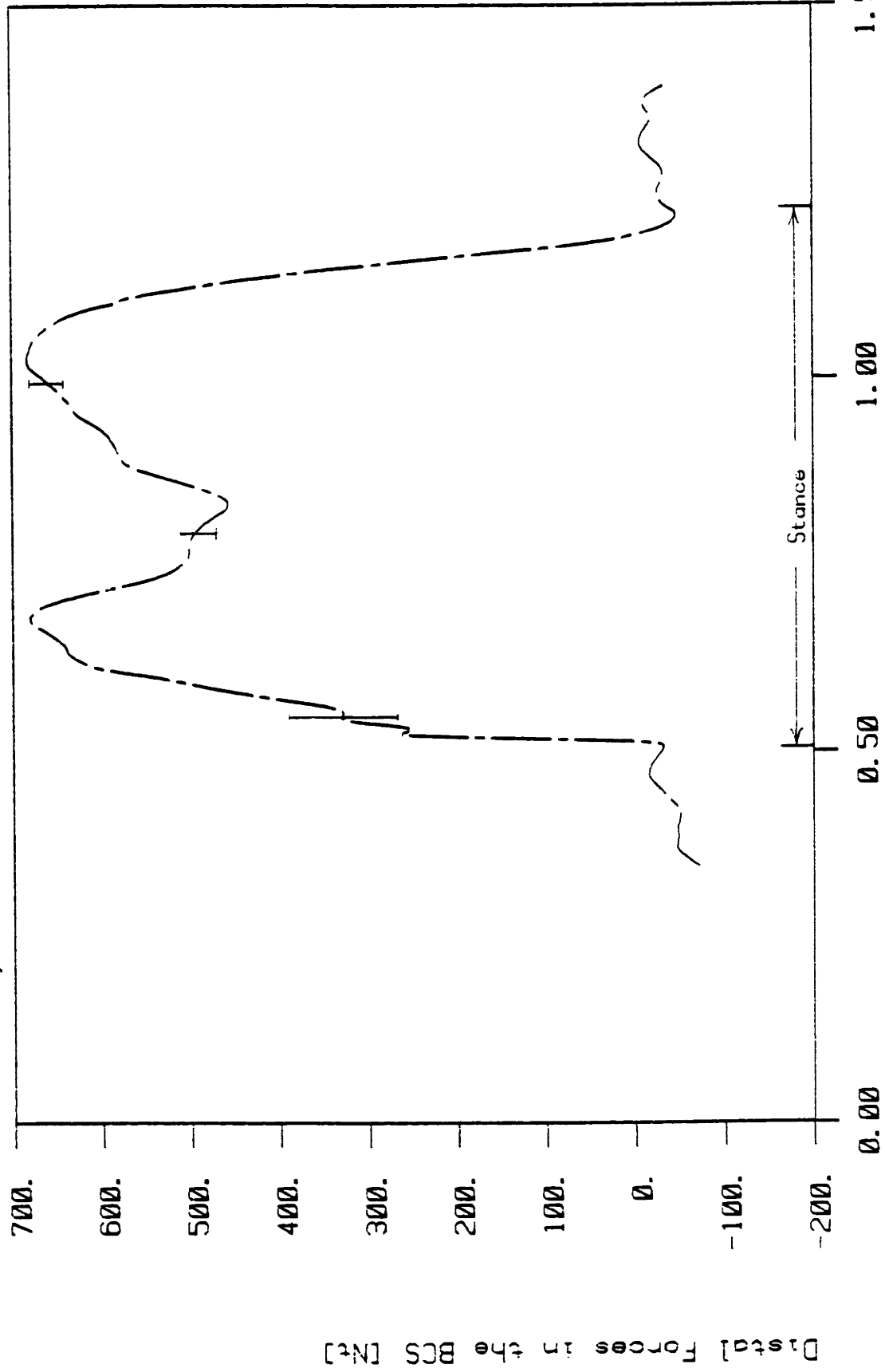
Figure 128. Forces acting at the knee in the direction of the long axis of the thigh (y axis) with sample error bounds
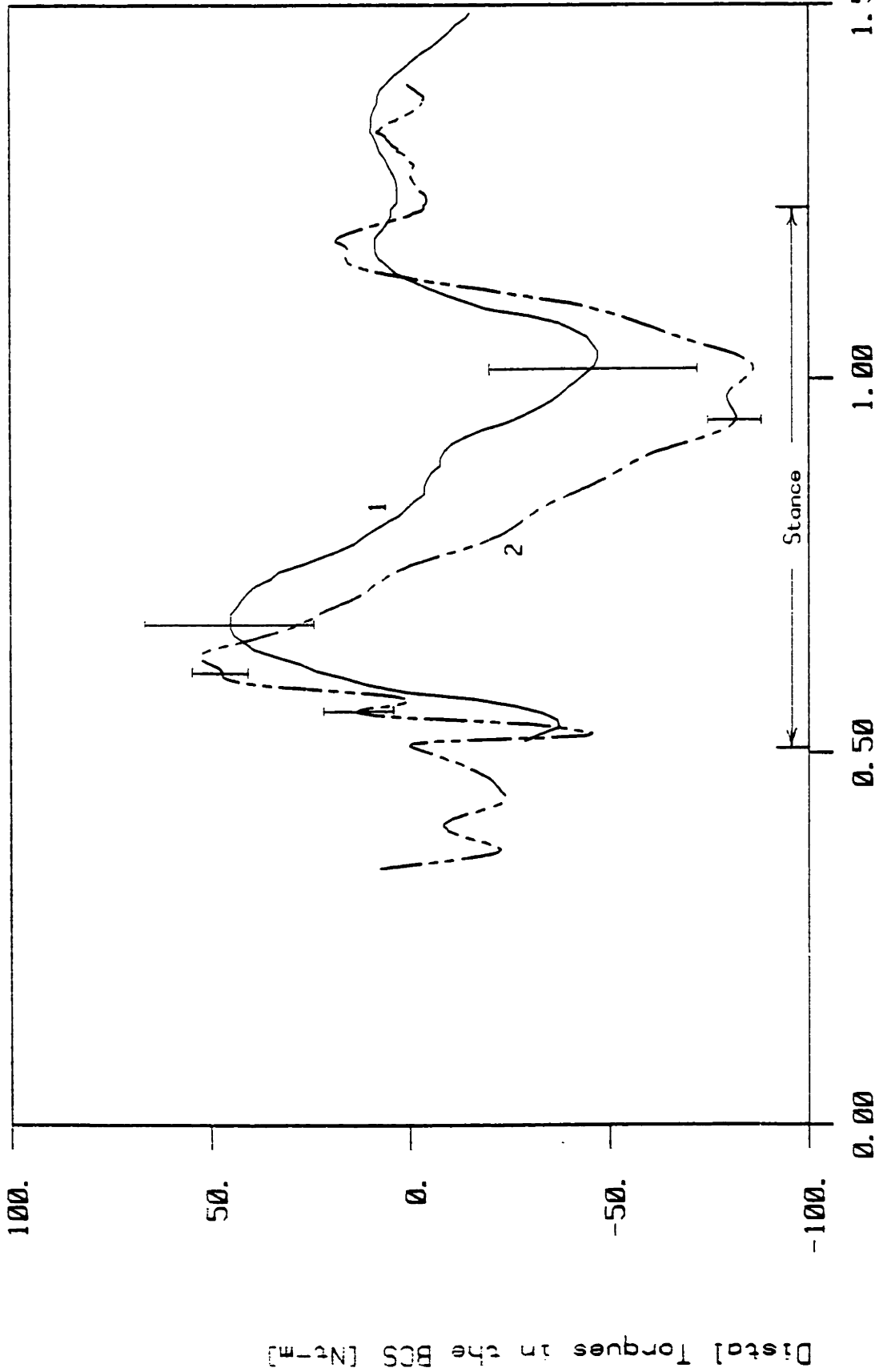
Figure 129. Moments acting about the flexion - extension
axis of the knee, with error bounds: 1) Cappozzo, 2) Antonsson
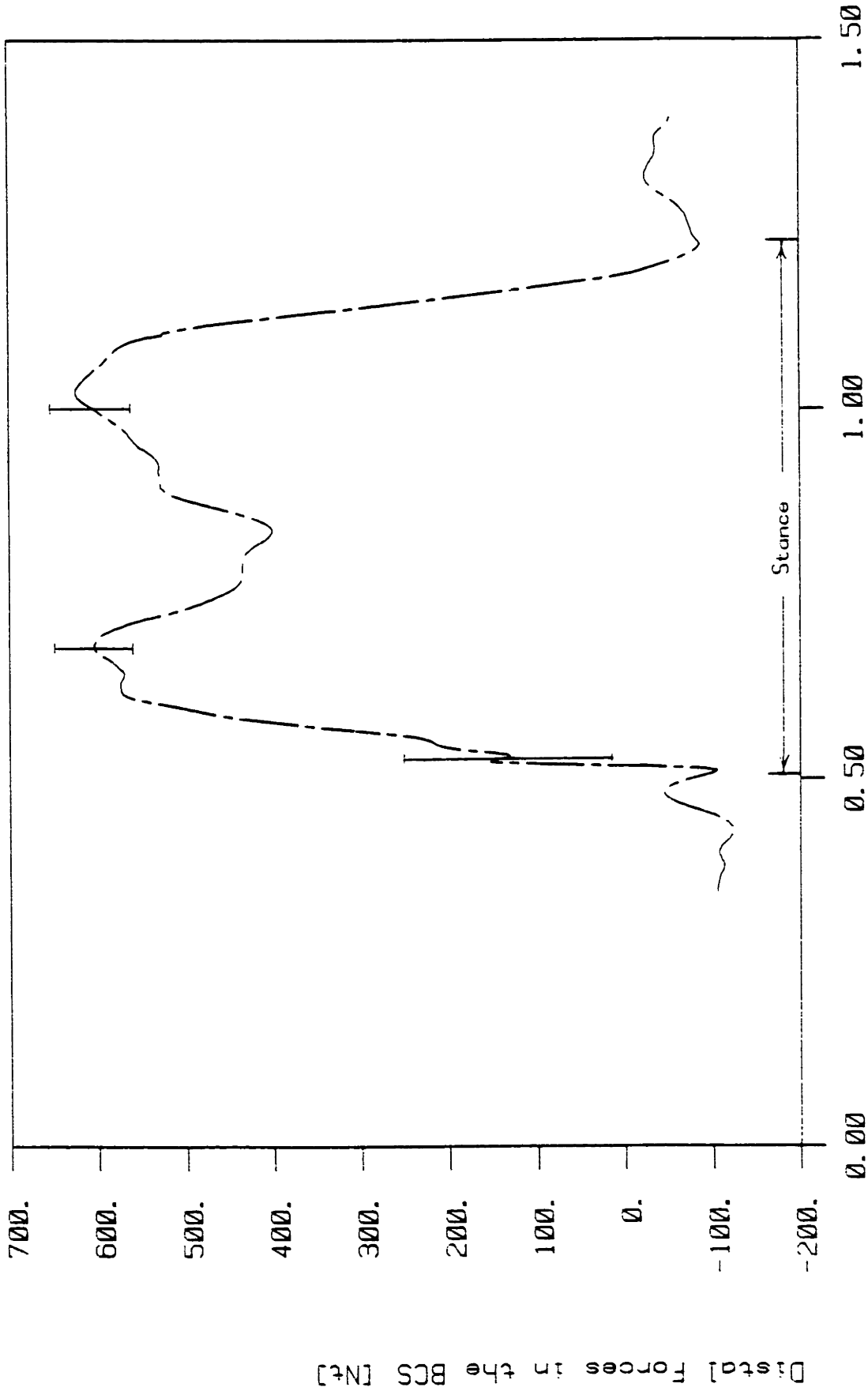
Figure 130. Forces acting at the hip in the direction of
the y axis (proximal) of the pelvis with sample error bounds
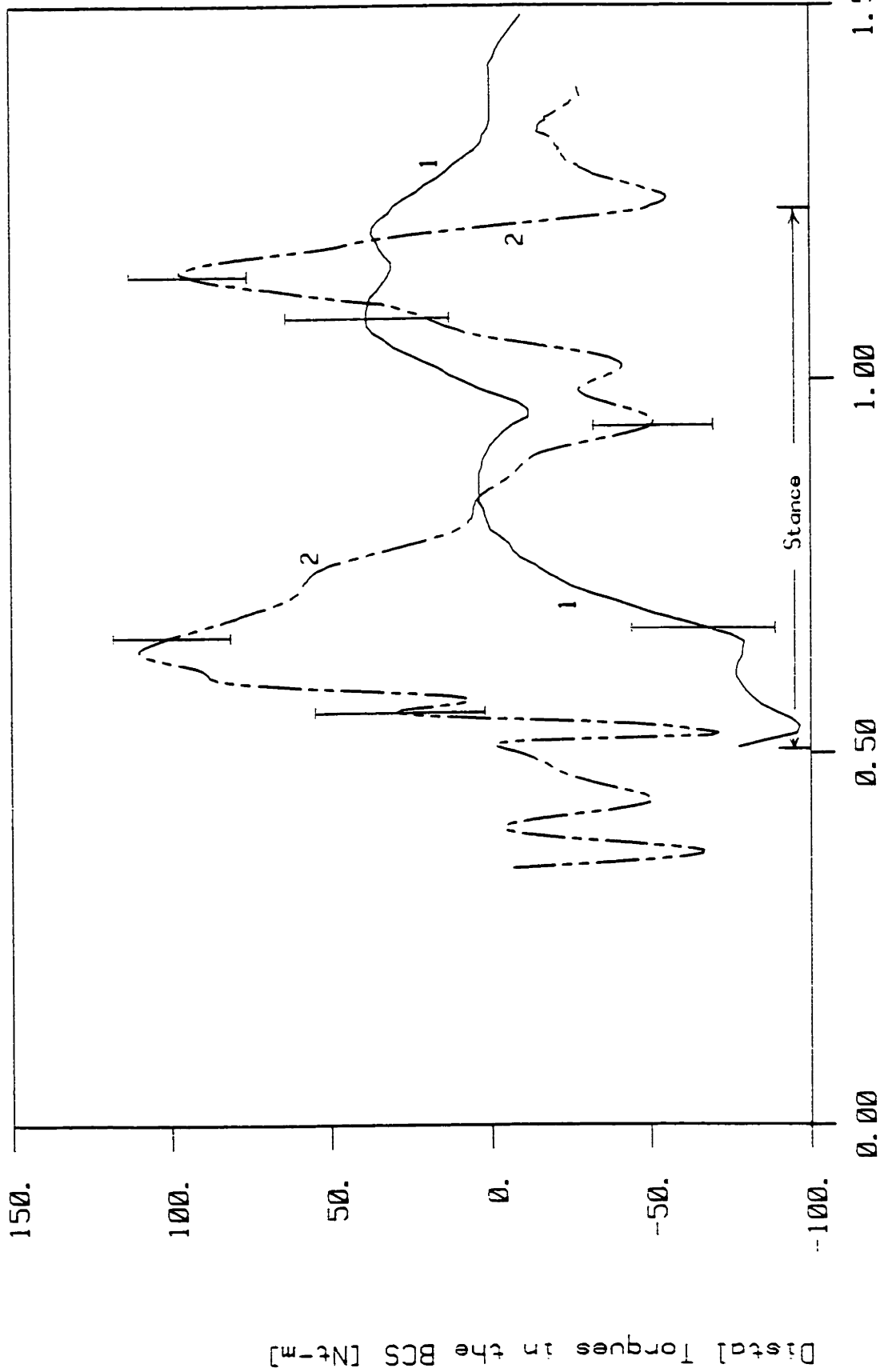
Figure 131. Moments acting about the flexion - extension axis of the hip, with error bounds: 1) Cappozzo, 2) Antonsson

There is good agreement between the two techniques for the moments about the ankle and hip and relatively poor agreement for the moments about the hip joint. For any technique that determines dynamics by proceeding along the linkage from a force reference (in this case the foot), the farther along the chain, the larger the accumulated errors. The larger disagreement at the hip is more likely than at other joints.

The comparison of my results to Cappozzo's has not been made to confirm the quality of my data, but rather to show that this fully automated technique produces results qualitatively not radically different from those currently being published; however, some vital differences must be noted. First, the bulk of my results fall within the error bound suggested by Cappozzo, but my error bound is smaller than his. However, his bound does not represent an envelope within which the correct result lies, rather it represents plus and minus one standard deviation of his data points with respect to the function fit, and as such gives no information on how close the data are to the correct result. Cappozzo's result curves are very smooth functions of time, as a result of choosing (somewhat arbitrarily) only a very few low harmonics to represent the kinematic data. Second, they provided no information about the subject's height, weight, or cadence, making a comparison of dynamic results difficult.

CHAPTER 6

DISCUSSION, CONCLUSIONS AND RECOMMENDATIONS

A system to automatically acquire kinematic data from, and analyze the motion of, a moving human has been designed and implemented. The original design goals of speed, objectivity, and accuracy have all been met and demonstrated. The system has been used in hundreds of experiments ranging from hand motion in American Sign Language, to studies of running efficiency, to instrumentation of an electro-hydraulic above-knee prosthesis simulator, and of course, normal gait. The need for a system with this samplng speed and accuracy has been demonstrated by the frequency domain analysis of noise influence.

The kinematics results have been used to locate average axes of rotation for each joint instrumented. Comparison with published data on limb segment lengths has verified the usefulness of the technique. The data have also been used to estimate the net motion producing dynamics, once double differentiation has been performed.

Since the error contributions to the dynamics have been evaluated at each stage in the discussion, the summation of errors, used as a bound of confidence on the dynamics

results, has been presented. Errors in the accumulated data, as well as other constants, all contribute to the size of the error bounds. The three major contributions are:

1. Double-differentiated residual kinematic noise contaminating the accelerations.

2. Soft tissue motion causing motion of the LED markers with respect to the skeleton and the center of mass

3. Uncertainties in the mass and inertial properties.

A system with the precision shown here is capable of locating an average axis of rotation for each joint; however, the fine kinematic details are lost even to this system when measuring motion at 3-meters range. Naturally, increased spatial resolution would be achieved by a closer camera spacing and a smaller viewing volume for those experiments where the fine details of joint kinematics are vital.

The experimental determination of the locations of joint centers provides several novel and useful functions for dynamic analyses. First, a measure of the soft tissue motion of LED markers can be determined, unlike typical gait analysis systems where the joint centers are fixed by marker location before the experiment, and second, errors in the joint axes locations have been shown to be no more than 1 cm, unlike pre-chosen joint centers where a measure of how close the marker is to the center is only obtainable from a radiographic analysis, and errors greater than 1 cm are

easily possible considering that the geometry of the joint must be estimated by external palpation alone.

A comparison of the results produced by TRACK and some published gait dynamics results show good agreement. The comparison of error bounds shows the confidence limits on my data to be narrower than the only available published values. The other published bounds of confidence only indicate the variance of their estimated data with respect to an (arbitrarily) fitted function, compared to the bounds found here that reflect the range in which the correct result must lie.

Despite the remaining problems (hardly unique to this technique), the TRACK system represents a substantial improvement over conventional human motion analysis techniques, not only in speed, automaticity, objectivity, and accuracy, but also in that each of the contributions to possible errors in the results has been examined, and a conservative estimate of the error contribution assessed. This is useful information to have in addition to the results for human motion analysis, both to enable proper interpretation of the results, as well as to indicate the most important aspects of the system susceptable to improvement in the future.

RECOMMENDATIONS FOR FUTURE RESEARCH

If higher quality inertia data were available for individual subjects, the quality of the resulting dynamic data would be improved. Except for the radiation hazard, the most promising technique appears to be gamma radiation computer tomography. The body is "sliced" into thin sections by the radiation, and the resulting density information can be assembled into a density distribution of the entire body, eventually even of individual muscles and bones themselves (Huang (1980) [79]). The density distribution can be integrated to yield mass and moments of inertia for each body segment and, ultimately, also account for changing mass and inertia properties as a segment changes position with muscle contraction.

The two other areas listed above as major contributors to error are also prime candidates for further investigation. Perhaps a radiographic study would illuminate the extent to which the segment retain kinematic fidelity with the skeleton, confirming the earlier estimates in this study, or provide vital information to aid the design of future generations of LED marker mounts.

Noise contamination of the signal will always be present; however, the Selspot II system should provide a better signal-to-noise ratio. The new kinematic acquisition system has four times finer resolution, and one can only speculate that if those extra bits of information (beyond the 10 bits currently provided) yield any useful data, the noise level must also have been reduced by a commensurate amount.

High frequency noise and resonances from the forceplate can easily be eliminated by pre-aliasing analog filters. The Kistler charge amplifiers currently have 180 kHz filters, however, lower frequency filters are available. Perhaps 100 Hz would be appropriate to avoid aliasing in the time-sampled results for gait experiments conducted with the Selspot where the highest sampling rate is 315 Hz.

An additional improvement that would greatly enhance the ability of the system to analyze more strides of normal gait, or to track a free-ranging human, would be to provide a controllable camera positioning system to permit the cameras to follow a subject anywhere in the laboratory.

Further development of the TRACK human motion measurement system itself should address the following specific questions. The florescent room lighting has been shown to introduce substantial high frequency noise in the measured position signals. Would incandescent bulbs produce

less noise? Is lighting powered by direct current (instead of 60 cycle alternating current) available? Would optical filters over the light bulbs, or the lenses (in addition to the filters already in place over the detector plates) reduce the influence of ambient light without attenuating the infra-red signal light too much? Since light intensity is proportional to the signal-to-noise ratio, the prospects for reducing the incident light with an optical filter to remove ambient seem doubtful.

A related area requiring further investigation is reflection of 940 nm wavelength light from surfaces in the experimental area. Human skin is quite reflective, as are most ordinary surfaces. Reflections can cause the measurement system to produce erroneous results, and reflectivity should be studied to effectively minimize its influence.

The calibration performed on Selspot I cameras used in this study was performed in a polar-coordinate system. A calibration implemented in cartesian coordinates (nomimally a simple transformation can be applied to convert between the two) would produce better results with the intrinsically cartesian detector, if for no other reason than the distribution of calibration points used could be uniform over the entire lateral-photo-effect diode. Presumably calibration will have to be repeated for the Selspot II, and

using a cartesian system (with very high accuracy) to position the LEDs (such as the Gerber flat-bed plotter) will simplify the procedure, and should provide more uniform results.

The 6th order 2 pass Butterworth filter used throughout the preceeding analysis has optimal characteristics in the frequency domain (as do all Butterworth filters); however, it is underdamped, which causes it to "ring" in the time domain, and thereby contaminate the signal. Other filters (e.g., Chebyshev) may provide adaquate results in the frequency domain and not ring as much in the time domain when subjected to sharp changes in signal (i.e., when a subject moves out of view of the Selspot cameras).

The addition of both lower extremities to the analysis will provide a more complete view of gait. Initially, this would not require another forceplate, but to analyze the stance phase dynamics of both legs, two plates would be needed.

As mentioned earlier in this chapter, confidence in the dynamics results would be improved by a more detailed, closeup analysis of the kinematics of the joints being studied. This kind of study would also provide novel data on both the unloaded and the loaded detailed kinematics of the lower extremity joints (or any skeletal joints).

Future uses of this kinematic and dynamic linkage analysis tool are almost limitless. Research already in progress includes a study of the detailed kinematics of the knee, both in normal and pathological knees. The net dynamic data, in conjunction with the muscle recruitment optimization work also proceeding in the Laboratory for Biomechanics and Human Rehabilitation, can yield data on the contact forces at joints -- the hip in particular -- with high accuracy and known bounds of confidence. Validations of those "muscle models" can be carried out by accumulating myoelectric signal activity simultaneously with gait kinematics and a breath-by-breath analysis of metabolic efficiency.

A preliminary study, illustrated in Photograph 2, has already been undertaken to determine if TRACK could be used to augment ballet.

Substantial interest in the application of quantitative measurement of athlete's motion has been accumulating for several years. TRACK's automaticity, high resolution and precision, and high speed all make it ideal for sport training and research. Injury prevention and rehabilitation, performance assessment, training, style, as well as education, are all areas of athletic endeavor to which this system could contribute.

Less glamorous perhaps, but certainly as important, is research concerning mobility impairment, both in augmenting the ability of a blind traveler to interact with his environment, and in providing insight into replacement limb design.

REFERENCES

1. AEG-Telefunken, Optoelectronic Selection Guide, Edition 1.81, Route 22, Orr Dr., Somerville, NJ 08876, Telephone: (201) 722-9800.

2. Aleshinsky, S. Y., and Zatsiorsky, V. M., "Human Locomotion in Space Analyzed Biomechanically Through a Multi-Link Chain Model," Journal of Biomechanics, 1978, 11, 101-108.

3. Alexander, M. J. L., Colbourne, J., "A Method of Determination of the Angular Velocity of a Limb Segment," Technical Note, Journal of Biomechanics, 1980, 13, 1089-1093.

4. Alexander, R. M., and Jayes, A. S., "Fourier Analysis of Forces Exerted in Walking and Running," Journal of Biomechanics, 1980, 13, 383-390.

5. Allum, J. H. J., and Young, L. R., "The Relaxed Oscillation Technique for the Determination of the Moment of Inertia of Limb Segments," Journal of Biomechanics, 1976, 9, 21-25.

6. Andriacchi, T. P., Hampton, S. J., Schultz, A. B., and Galante, J. O., "Three Dimensional Coordinate Data Processing in Human Motion Analysis," Journal of Biomechanical Engineering, Transactions of the ASME, 1979, 101:4, 279-283.

7. Antonsson, E. K., "The Derivation and Implementation of a Dynamic 3-D Linkage Analysis Technique," S.M. Thesis, M.I.T., Cambridge, MA, May 1978.

8. Antonsson, E. K., and Mann, R. W., "Automatic 3-D Gait Analysis Using a Selspot Centered System," 1979 Advances in Bioengineering, ASME, New York, 1979, 51.

9. Antonsson, E. K., "Automatic 3-D Gait Analysis Using a Selspot-Based System," Proceedings of the 4th Congress of the International Society of Electrophysiological Kinesiology, Boston, MA, August, 1979, 122.

10. Antonsson, E. K., and Mann, R. W., "Quantitative Three-Dimensional Dynamic Analysis of Human Movement," 8th Annual Northeast Bioengineering Conference, M.I.T., Cambridge, MA, March 27-28, 1980, 187.

11. Arvikar, R. and Seireg, A., "Pressure Distribution Under the Foot During Static Activities," Engineering in Medicine, 1980, 9:2, 99-103.

12. Baumann, W., "New Chronophotographic Methods for Three-Dimensional Movement Analysis," in: Nelson, R. C., and Morehouse, C. A., eds., Biomechanics IV. Baltimore: University Park Press, 1974, 464-468.

13. Bisshopp, K. E., "Rodrigues' Formula and the Screw Matrix," Journal of Engineering for Industry, Transactions of the ASME, New York, February 1969, 179-185.

14. Boccardi, S., Pedotti, A., Rodano, R., and Santambrigio, G. C., "Evaluation of Muscular Moments at the Lower Limb Joints by an On-Line Processing of Kinematic Data and Ground Reaction," Journal of Biomechanics, 1981, 35-45.

15. Bottema, O., and Roth, B., Theoretical Kinematics. New York: North-Holland Publishing Company, 1979.

16. Braune, W., and Fischer, O., "Uber den Schwerpunkt des Menschlichen Korpers mit Rucksicht auf die Austrustung des Deutschen Infanteristen. Abh. d. Math.-Phys. cl. d. K. Sachs. Gesellsch. der Wiss., 26:561-672. 1889, in W. M. Krogman and F. E. Johnston, eds., Human Mechanics - Four Monographs abridged, Wright-Patterson Air Force Base, Ohio, 1963 (AMRL-TDR-63-123).*

17. Bresler, B., and Frankel, J. P., "The Forces and Moments in the Leg During Level Walking," Transactions of the ASME 1950, 72:27, 27-35.

18. Broerman, J. G.,"Nonlinear Background to Signal Coupling in Lateral Photoeffects," IRE Trans., 1968, ED-15, 860-864.*

19. Brooks, C. B., and Jacobs, A. M., "The Gamma Mass Scanning Technique for Inertial Anthropometric Measurement," Med. Sci. Sports, 7:290-294, 1975.*

20. Brown, T. R. M., Paul, J. P., Kelly, I. G., and Hamblen, D. L., "Biomechanical Assessment of Patients Treated by Joint Surgery," Journal of Biomedical Engineering, October, 1981, Volume 3, 297-304.

21. Cappozzo, A., Figura, F., Marchetti, M., and Pedotti, A., "The Interplay of Muscular and External Forces in Human Ambulation," Journal of Biomechanics, 1976, 9, 35-43.

22. Cappozzo, A., Leo, T., and Pedotti, A., "A General Computing Method for the Analysis of Human Locomotion," Journal of Biomechanics, 1975, 8, 307-320.

23. Cappozzo, A., Maini, M., Marchetti, M., and Pedotti, A., "Analysis by Hybrid Computer of Ground Reactions in Walking," in: Nelson, R. C., and Morehouse, C. A., eds., Biomechanics IV. Baltimore: University Park Press, 1974, 496-501.

24. Carlson, D., Mann, R., and Harris,W., "A Radio Telemetry Device for Monitoring Cartilage Surface Pressures in the Human Hip," IEEE Transactions, 1974, BME-21 No.4, 257-264.

25. Casper, R. M., Jacobs, A. M., Kenney, E. S., and McMaster, I. B., "On the use of Gamma Ray Images for Determination of Human Body Segment Parameters," Paper presented at Quantitative Imagery in Biomedical Sciences, Houston Texas, 1971.*

26. Cavanagh, P. R., "A Technique for Averaging Center of Pressure Paths from a Force Platform," Technical Note, Journal of Biomechanics, 1978, 11, 487-491.

27. Cavanagh, P. R., and Gregor, R. J., "The Quick-release Method for Estimating the Moment of Inertia of the Shank and Foot," in: Nelson, R. C., and Morehouse, C. A., eds., Biomechanics IV. Baltimore: University Park Press, 1974, 524-530.

28. Cavanagh, P. R. and Lafortune, M. A., "Ground Reaction Forces in Distance Running," Journal of Biomechanics, 1980, 13, 397-406.

29. Chao, E. Y.,"Experimental Methods for Biomechanical Measurements of Joint Kinematics," in B. N. Feinberg and D. G. Flemming, eds., CRC Handbook of Engineering in Medicine and Biology. Section B, "Instruments and Measurements," Volume I, West Palm Beach, Florida; CRC Press, 385-411.

30. Chao, E. Y., "Justification of Triaxial Goniometer for the Measurement of Joint Rotation," Journal of Biomechanics, 1980, 13, 989-1006.

31. Chao, E. and Rim, K., "Application of Optimization Principles in Determining the Applied Moments in the Human Leg Joints During Gait," Journal of Biomechanics, 1973, 6, 497.

32. Chao, E., et al., "Theoretical Justification and Clinical Application of Triaxial Goniometric Method in Gait Analysis," 25th Annual ORS, San Francisco, California, February 20-22, 1979.

33. Clauser, C. E., McConville, J. T., and Young, J. W., "Weight, Volume, and Center of Mass of Segments of the Human Body," Wright-Patterson Air Force Base, Ohio, 1969, (AMRL-TR-69-70).*

34. Conati, F., "Real Time Measurement of 3-D Multiple Rigid Body Motion," S.M. Thesis, M.I.T., Cambridge, MA, June 1977.

35. Contini, R., "Body Segment Parameters, Part II," Artificial Limbs, 1972, 16:1, 1-19.

36. CRC Handbook of Engineering in Medicine and Biology. B. N. Feinberg and D. G. Flemming, eds., West Palm Beach, Florida: CRC Press.

37. Crowninshield, R. D., and Brand, R. A., "A Physiologically Based Criterion of Muscle Force Prediction in Locomotion," Journal of Biomechanics, 1981, 14:11, 793-801.

38. Crowninshield, R. D., and Brand, R. A., "Kinematics and Kinetics of Gait," in B. N. Feinberg and D. G. Flemming, eds., CRC Handbook of Engineering in Medicine and Biology. Section B, "Instruments and Measurements," Volume I, West Palm Beach, Florida; CRC Press, 385-411.

39. Crowninshield, R. D., Johnston, R. C., Andrews, J. G., and Brand, R. A., "A Biomedical Investigation of the Human Hip," Journal of Biomechanics, 1978, 11, 75-85.

40. Crowninshield, R. D., Pope, M. H. and, Johnson, R., "An Analytical Model of the Knee," Journal of Biomechanics, 1976, 9, 397-405.

41. Crowninshield, R. D., Pope, M. H., Johnson, R., and Miller, R., "The Impedance of the Human Knee," Journal of Biomechanics, 1976, 9, 529-535.

42. Dainis, A., "Whole Body and Segment Center of Mass Determination from Kinematic Data," Journal of Biomechanics, 1980, 13, 647-651.

43. Dapena, J., "A Method to Determine the Angular Momentum of a Human Body about Three Orthogonal Axes Passing Through Its Center of Gravity," Journal of Biomechanics, 1978, 11, 251-256.

44. Dapena, J., "Simulation of Modified Human Airborne Movements," Journal of Biomechanics, 1981, 14, 81-89.

45. Denavit, J., and Hartenberg, R. S., "A Kinematic Notation for Lower-Pair Mechanisms Based on Matricies," Journal of Applied Mechanics, Transactions of the ASME, June, 1955, 215-221.

46. Denavit, J., Hartenberg, R. S., Razi, R., and Uicker, J. J., "Velocity, Acceleration, and Static-Force Analysis of Spatial Linkages," Journal of Applied Mechanics, Transactions of the ASME, December, 1965, 903-910.

47. Dempster, W. T., "Free-body Diagrams as an Approach to the Mechanics of Human Posture and Locomotion," in F. G. Evans, ed., Biomechanical Studies of the Musculoskeletal System, Springfield, Ill., Charles C. Thomas, 1961.*

48. Dempster, W. T., "Space Requirements of the Seated Operator," Wright-Patterson Air Force Base, Ohio, 1955, (WADCTR 55-159).*

49. Dimnet, J., "The Improvement in the Results of Kinematics of in vivo Joints," Journal of Biomechanics, 1980, 13, 653-661.

50. Dorland's Illustrated Medical Dictionary. Twenty-fifth Edition, Philadelphia: W. B. Saunders, 1974.

51. Dostal, W. F., and Andrews, J. G., "A Three-Dimensional Biomechanical Model of Hip Musculature," Journal of Biomechanics, 1981, 14:11, 803-812.

52. Drillis, R., Contini, R., and Bluestein, M., "Body Segment Parameters: A Survey of Measurement Techniques," Artificial Limbs, 1967, 8:1, 44-66.

53. Eberhart, H. D., Inman, V. T., and Bresler, B., "The Principle Elements in Human Locomotion," in: Klopsteg, P. E., and Wilson, P. D., Human Limbs and their Substitutes. New York: Hafner Publishing Company, 1968, 437-471.

54. Eckhouse, R. H., and Morris, L. R. MINICOMPUTER SYSTEMS Organization, Programming, and Applications (PDP-11). Second Edition, Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1979.

55. Elftman, H., "The Functional Structure of the Lower Limb," in: Klopsteg, P. E., and Wilson, P. D., Human Limbs and their Substitutes. New York: Hafner Publishing Company, 1968, 411-436.

56. Elftman, H., "The Measurement of the External Force in Walking," Science, August 12, 1938, 88:2276, 152-153.

57. Engin, A. E., "On the Biomechanics of Major Articulating Human Joints," in Akkas, N., ed., Progress in Biomechanics, Netherlands: Sijthoff and Noordhoff, 1979, 157-187.

58. Fieret, J., Kwakernaak, A., and Middelhoek, S., "Large Linear Area Light-spot-position-sensitive Photodiode," Electronic Letters, July 7, 1977, 19:14, 422-424.

59. Foley, C. D., Quanbury, A. O., and Steinke, T., "Kinematics of Normal Child Locomotion - A Statistical Study Based on TV Data," Journal of Biomechanics, 1979, 12, 1-6.

60. Frankel, V. H., Burstein, A. H., and Brooks, D. B., "Biomechanics of Internal Derangement of the Knee," Journal of Bone and Joint Surgery, July 1971, 53-A:5,945-962.

61. Ghista, D. N., Toridis, T. G., and Srinivasan, T. M., "Human Gait Analysis: Determination of instantaneous joint reactive forces, muscle forces and the stress distribution in bone segments, Part I," Biomedizinische Technik, 1975, 20:6, 201-213.

62. Ghista, D. N., Toridis, T. G., and Srinivasan, T. M., "Human Gait Analysis: Determination of instantaneous joint reactive forces, muscle forces and the stress distribution in bone segments, Part II," Biomedizinische Technik, 1976, 21:3, 66-74.

63. Gray, H. Anatomy, Descriptive and Surgical. New York: Bounty Books, 1977.

64. Groden, C. M., and Richards, J. A., "On the Non-uniformly Illuminated p-n Junction II," Solid-State Electron, 1969, 12, 813-821.*

65. Gustaffson, L. and Lanshammar, H., "Enoch -- An Integrated System for Measurement and Analysis of Human Gait," Tekinikum Institute of Technology, Uppsalla University, UPTEC 77-23-R, April 1977.

66. Hardt, D. E. "Determining Muscle Forces in the Leg During Normal Human Gait--An Application and Evaluation of Optimization Methods", ASME Journal of Biomechanical Engineering, 1978, Volume 100, 72-78.

67. Hardt, D.,"A Minimum Energy Solution for Muscle Force Control During Walking," Ph.D. Thesis, M.I.T., Cambridge, MA, Sept. 1978.

68. Hardt, D. E., and Mann, R. W., "A Five Body, Three-Dimensional Dynamic Analysis of Walking," Journal of Biomechanics, 1980, 13:5, 455-457.

69. Hatze, H., "A Mathematical Model for the Computational Determination of Parameter values of Anthropomorphic Segments," Journal of Biomechanics, 1980, 13, 833-843.

70. Hatze, H. "A New Method fo the Simultaneous Measurement of the Moment of Inertia, The Damping Coefficient and the Location of the Centre of Mass of a Body Segment in Situ," Biomechanics Laboratory, Department of Physical Education, University of Stellenbosch, Stellenbosch, South Africa.

71. Hatze, H., "Neuromusculoskeletal Control Systems Modeling - A Critical Survey of Recent Developments," IEEE Transactions on Automatic Control, AC-25:3, June 1980, 375-385.

72. Hatze, H., "The Complete Optimization of Human Motion," University of Stellenbosch, S. Africa, Biomechanics Laboratory.

73. Hatze H., The Inverse Dynamics of Three-dimensional Hominoid Motion, Special Report, National Research Institute for Mathematical Sciences, April, 1981, Pretoria, Republic of South Africa.

74. Hatze, H., "The Use of Optimally Regularized Fourier Series for Estimating Higher-Order Derivatives of Noisy Biomechanical Data," Journal of Biomechanics, 1981, 14, 13-18.

75. Hatze H., and Venter, A., HOMSIM: A Simulator of Three-dimensional Hominoid Dynamics, Special Report, National Research Institute for Mathematical Sciences, April, 1981, Pretoria, Republic of South Africa.

76. Heglund, N. C., and Taylor, C. R., and McMahon, T. A., "Scaling Stride Frequency and Gait to Animal Size: Mice to Horses," Science, 1974, 186, 1112-1113.

77. Hof, A. L., and Van den Berg, Jw., "EMG to Force Processing III: Estimation of Model Parameters for the Human Triceps Surae Muscle and Assesment of the Accuracy by Means of a Torque Plate," Journal of Biomechanics, 1981, 14:11, 771-785.

78. Hollerbach, J. M., "A Recursive Lagrangian Formulation of Manipulator Dynamics and a Comparative Study of Dynamics Formulation Complexity," IEEE Transactions on Systems, Man, and Cybernetics, 1980, SMC-10:11, 730-736.

79. Huang, H. K., "A Data Base of Anatomical Cross-Sectional Geometry and Mass Distribution for Children," Technical Report I, (3 YF), DOT-HS-7-01661, Department of Physiology and Biophysics, Georgetown University Medical School, Washington, D.C., February 1980.

80. Huston, R., Passerello, C. E., "On Human Body Dynamics," Annals of Biomedical Engineering, 1976, 4, 25-43.

81. Jacob, S. W., Francone, C. A., and Lossow, W. J. Structure and Function in Man. Philadelphia: W. B. Saunders Co., 1978.

82. Jensen, R. K., "Estimation of the Biomechanical Properties of Three Body Types using a Photogrammetric Method," Journal of Biomechanics, 1978, 11, 349-358.

83. Johnston, R. C., and Smidt, G. L., "Measurement of Hip-Joint Motion during Walking," Journal of Bone and Joint Surgery, September, 1969, 51-A:6, 1083-1094.

84. Kairento, A., and Hellen, G., "Biomechanical Analysis of Walking," Journal of Biomechanics, 1981, 14:10, 671-678.

85. Kane, T. R., "Inertial Parameters Required for the Dynamic Analysis of Human Motions," Mechanics and Sport, J. Bleustein, ed., New York:ASME, AMD-Volume 4, 1973.

86. Kelley, D. L., Kinesiology: Fundamentals of Motion Description. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1971.

87. Kettelkamp, D. B., Johnson, R. J., Smidt, G. L., Chao, E. Y. S., Walker, M., "An Electrogoniometric Study of Knee Motion in Normal Gait," Journal of Bone and Joint Surgery, June, 1970, 52-A:4, 775-790.

88. Kilvington, M., and Goodman, R. M. F., "In Vivo Hip Joint Forces Recorded on a Strain Gauged 'English' Prosthesis Using an Implanted Transmitter," Engineering in Medicine, 1981, 10:4, 175-187.

89. Kinzel, G. L., Hall, A. S., and Hillberry, B. M., "Measurement of the Total Motion Between Two Body Segments - I Analytical Development," Journal of Biomechanics, 1972, 5, 93-105.

90. Kinzel, G. L., Hillberry, B. M., Hall, A. S., Van Sickle, D. C., and Harvey, W. M., "Measurement of the Total Motion Between Two Body Segments - II Description of Application," Journal of Biomechanics, 1972, 5, 283-293.

91. Lamoreux, L. W., "Kinematic Measurements in the Study of Human Walking," Bulletin of Prosthetics Research, Spring 1971, 3-84.

92. Langrana, N. A., and Bartel, D. L., "An Automated Method for Dynamic Analysis of Spatial Linkages for Biomechanical Applications," Journal of Engineering for Industry, Transactions of the ASME, 1975, Paper Number 74-DET-42.

93. Lanshammar, H., "On Practical Evaluation of Differentiation Techniques for Human Gait Analysis," Journal of Biomechanics, 1982, 15:2, 99-105.

94. Lenox, J. B., "Six Degree of Freedon Human Eyeball Movement Analysis Involving Stereometric Techniques," Ph.D. Thesis, Stanford University, Palo Alto, CA, 1976.

95. Light, L. H., McLellan, G. E., and Klenerman, L., "Skeletal Transients on Heel Strike in Normal Walking with Different Footwear," Journal of Biomechanics, 1980, 13, 477-480.

96. Lindholm, L. E., "An Optoelectronic Instrument for Remote On-line Movement Monitoring," in: Nelson, R. C., and Morehouse, C. A., eds., Biomechanics IV. Baltimore: University Park Press, 1974, 510-512.

97. Lopez-Antunez, L. Atlas of Human Anatomy. Philadelphia: W. B. Saunders Co., 1971.

98. Lucovsky, G., "Photoeffect in Nonuniformly Irradiated p-n Junction," J. Appl. Phys., 1960, 31, 1088-1095.*

99. Mann, R. W., Rowell, D., Dalrymple, G., Conati, F., Tetewsky, A., Ottenheimer, D., and Antonsson E., "Precise, Rapid, Automatic 3-D Position and Orientation Tracking of Multiple Moving Bodies," Biomechanics VIII, Champaign, Illinois: Human Kinetics Publishers, 1982.

100. Marks' Standard Handbook for Mechanical Engineers. Second Edition, T. Baumeister (editor in chief) New York:McGraw Hill, 1978.

101. Marsolais, E. B., and Schulz, E., "Engineering Applications in Orthotic and Prosthetic Treatment of Musculoskeletal Defects," Bulletin of Prosthetics Research, Fall 1977, 195-214.

102. McGhee, R., Koozenkanani, S., Gupta, S., and Cheng, I., "Automatic Estimation of Joint Forces and Moments in Human Locomotion from Television Data," Procceedings of the IV IFAC Symposium on Identification and Parameter Estimation, Georgian SSR, USSR, September 1976.

103. McKechnie, R. E., and Cousins, S. J., "A Parallelogram Chain Mechanism for Measuring Human Joint Motion," ASME 76-DET-71.

104. McMahon, T. A., and Greene, P. R., "Fast Running Tracks," Scientific American, December 1978, 148-163.

105. Mena, D., Mansour, J. M., and Simon, S. R., "A Kinematic Model of the Leg during Swing Phase and It's Applicability to Gait Pathology," 25th Annual ORS, San Francisco, California, February 20-22, 1979, 265.

106. Mena, D., Mansour, J. M., and Simon, S. R., "Analysis and Systhesis of Human Swing Leg Motion During Gait and Its Clinical Applications," Journal of Biomechanics, 1981, 14:12, 823-832.

107. Meriam, J. L., Dynamics. New York:John Wiley and Sons, Inc., 1971.

108. Miller, D. I., and Nelson, R. C., Biomechanics of Sport, Philadelphia, Lea and Febiger, 1973.*

109. Miller, N. R., Shapiro, R., and McLaughlin, T. M., "A Technique for Obtaining Spatial Kinematic Parameters of Segments of Biomechanical Systems from Cinematographic Data," Journal of Biomechanics, 1980, 13, 535-547.

110. Mizrahi, J., and Suzak, Z., "In-Vivo Elastic and Damping Response of the Human Leg to Impact Forces," Journal of Biomechanical Engineering, (Transactions of the ASME), February, 1982, Vol. 104, 63-66

111. Mochon, S., and McMahon, T. A., "Ballistic Walking," Journal of Biomechanics, 1980, 13, 49-57.

112. Moffatt, C. A., Harris, E. H., and Haslam, E. T., "An Experimental and Analytic Study of the Dynamic Properties of the Human Leg," Journal of Biomechanics, 1969, 2, 373-387.

113. Nathan, R., "Location of the Resultant Force and the Instantaneous Centre of Rotation of Body Joints," Digest of the 11th International Conference on Medical and Biological Engineering, Ottawa, Canada, 1976, 90-91.

114. Nissan, M., "Review of Some Basic Assumptions in Knee Biomechanics," Journal of Biomechanics, 2980, 13, 375-381.

115. Oberg, K., "Mathematical Modeling of the Human Gait: An Application of the SELSPOT-System," in: Nelson, R. C., and Morehouse, C. A., eds., Biomechanics IV. Baltimore: University Park Press, 1974, 79-84.

116. Onyshko, S., and Winter, D. A., "A Mathematical Model for the Dynamics of Human Locomotion," Journal of Biomechanics, 198, 13, 361-368.

117. Oppenheim, A. V., Schafer, R. W. Digital Signal Processing. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1975.

118. Otis, J. C., Fabian, D. F., and Insall, J. N., "Evaluation of the VA Rancho Gait Analyzer Using Total Knee Arthroplasty Patients Preoperatively and Postoperatively," 25th Annual ORS, San Francisco, California, February 20-22, 1979, 76.

118a. Ottenheimer, D. E., "Dynamic Simulation of Auditory Spatial Displays," S.M. Thesis, M.I.T., Cambridge, MA, January, 1982.

119. Padgaonkar, A. J., Krieger, K. W., and King, A. I., "Measurement of Angular Acceleration of a Rigid Body Using Linear Accelerometers," Journal of Applied Mechanics, Transactions of the ASME, September, 1975, 552-556.

120. Panjabi, M. M., "Centers and Angles of Rotation of Body Joints: A Study of Errors and Optimization," Journal of Biomechanics, 1979, 12, 911-920.

121. Panjabi, M. M., Walters, S., and Goel, V. K., "Guidelines for Optimal Experimental Design to Study Joint Kinematics Using Instant Axis Concept," Van Buskirk, W. C., and Woo, S. L. Y., eds., Proceedings of the ASME, AMD-Vol. 43, 1981 Biomechanics Symposium, University of Colorado, Boulder, Colorado, June 22-24, 1981, 185-188.

121a. Patriarco, A. F., "The Prediction of Individual Muscle Forces During Human Movement," Ph.D. Thesis, Harvard University, Cambridge, MA, June, 1982.

122. Patriarco, A. F., Mann, R. W., Simon, S. R., and Mansour, M. J., "An Evaluation of the Approaches of Optimization Models in the Prediction of Muscle Forces During Human Gait," Journal of Biomechanics, 1981, 14:8, 513-525.

123. Paul, J. P., "Forces Transmitted by Joints in the Human Body," Proc. Inst. of M.E., 1967, 181:31, 8-15.

124. Payne, A. H., "A Force Platform System for Biomechanics Research in Sport," in: Nelson, R. C., and Morehouse, C. A., eds., Biomechanics IV. Baltimore: University Park Press, 1974, 502-509.

125. Pezzack, J. C., Norman, R. W., and Winter, D. A., "An Assessment of Derivative Determining Techniques Used for Motion Analysis," Technical Note, Journal of Biomechanics, 1977, 10, 377-382.

126. Plagenhoef, S. C., "Computer Programs for Obtaining Kinetic Data on Human Movement," Journal of Biomechanics, Technical Note, 1968, 1, 221-234.

127. Plagenhoef, S. C., "Methods for Obtaining Data to Analyze Human Motion," Res. Q. Am. Assoc. Health Phys. Educ., 37:103-112, 1966.*

128. Plagenhoef, S. C., Patterns of Human Motion: A Cinematographical Analysis, Englewood Cliffs, N.J.: Prentice-Hall, Inc. 1971.*

129. Ramey, M. R., and Yang, A. T., "A Simulation Procedure for Human Motion Studies," Journal of Biomechanics, 1981, 14, 203-213.

130. Reuleaux, F., The Kinematics of Machinery. Translated and Edited by A. B. W. Kennedy, New York: Dover, 1963.

131. Robertson, D. G. E., and Winter, D. A., "Mechanical Energy Generation, Absorption and Transfer Amongst Segments During Walking," Journal of Biomechanics, 1980, 13, 845-854.

132. Roth, B., "Finite-Position Theory Applied to Mechanism Synthesis," Journal of Applied Mechanics, Transactions of the ASME, September, 1967, 599-605.

133. Roth, B., "On the Screw Axes and Other Special Lines Associated with Spatial Displacements of a Rigid Body," Journal of Applied Mechanics, Transactions of the ASME, February, 1967, 102-110.

134. Roth, B., "The Kinematics of Motion Through Finitely Separated Positions," Journal of Applied Mechanics, Transactions of the ASME, September, 1967, 591-598.

135. Rydell, N., "Intravital Measurements of Forces Acting on the Hip Joint," in Evans, F.G., (ed.), Studies on the Anatomy and Function of Bone and Joints, New York: Springer Verlag, 1966.

136. Saunders, J., and O'Malley, C. The Illustrations from the Works of Andreas Vesalius. New York: World Publishing Co., 1950.

137. Schut, G. H., "On Exact Linear Equations for the Computation of the Rotational Elements of Absolute Orientation," Photogrammetria, 1960/61, 17:1.

138. Scranton, P. E., and McMaster, J. H., "Momentary Distribution of Forces under the Foot," Journal of Biomechanics, 1976, 9, 45-48.

139. Seireg, A., and Arvikar, R., "A Mathematical Model for Evaluation of Forces in Lower Extremities of the Musculo-Skeletal System," Journal of Biomechanics, 1973, 6, 313-326.

140. Seireg, A., and R. J. Arvikar, "The Prediction of Muscular Load Sharing and Joint Forces in the Lower Extremities During Walking," Journal of Biomechanics, 1975, 8, 89-102

141. Selvik, G., "A Roentgen Stereophotogrammetric Method for the Study of the Kinematics of the Skeletal System. Principles and Applications," Lecture delivered at the symposium: Three-dimensional Position and Displacement Measurements in the Human Body, October 28, 1977, Department of Orthopaedic Surgery, University of Nijmegen, Netherlands.

142. Senturia, S. D., and Wedlock, B. D., Electronic Circuits and Applications, New York: John Wiley and Sons, Inc., 1975.

143. Simon, S. R., Paul, I. L., Mansour, J., Munro, M., Abernethy, P. J., and Radin, E. L., "Peak Dynamic Force in Human Gait," Journal of Biomechanics, 1981, 14:12, 817-822.

144. Slama, C. C., ed. Manual of Photogrammetry. Falls Church, Virginia: American Society of Photogrammetry, 1980.

145. Sommer, H. J., and Miller, N. R., "A Technique for the Calibration of Instrumented Spatial Linkages Used For Biomechanical Kinematic Measurements," Journal of Biomechanics, 1981, 91-98.

146. Sorbie, C., and Zalter, R., "Bio-Engineering Studies of the Forces Transmitted by Joints, (1) The Phasic Relationship of the Hip Muscles in Walking," Proc. Symposium on Biomechanics and Related Bioengineering Topics. R. Kenedi, ed., Oxford: Pergamon Press, 359-367.

147. Soudan, K. and Dierckx, P., "Calculation of Derivatives and Fourier Coefficients of Human Motion Data, While Using Spline Functions," Journal of Biomechanics, 1979, 12, 21-26.

148. Soudan, K., Van Audekercke, R., and Martens, M., "Methods, Difficulties and Inaccuracies in the Study of Human Joint Kinematics and Pathokinematics by the Instant Axis Concept. Example: The Knee Joint," Journal of Biomechanics, 1979, 12, 27-33.

149. Spoor, C. W., and Veldpaus, F. E., "Rigid Body Motion Calculated from Spatial Co-ordinates of Markers," Journal of Biomechanics, Technical Note, 1980, 13, 391-393.

150. Stearns, S. D., Digital Signal Analysis. Rochelle Park, New Jersey: Hayden Book Company, Inc., 1975.

151. Stevens, T. D., "Ultrasonic Location of Bone During Human Gait," S.B. Thesis, M.I.T., Cambridge, MA, May 1979.

152. Tachi, S., Mann, R. W., and Rowell, D., "A Quantitative Comparison Method of Display Scheme in Mobility Aids for the Blind," Biomechanics VIII, Champaign, Illinois: Human Kinetics Publishers, 1982.

153. Tetewsky, A. K. "Implementing a Real Time Computation and Display Algorithm for the Selspot System," S.M. Thesis, Dept. of Mechanical Engineering, M.I.T., 1978.

154. The TTL Data Book for Design Engineers. U.S.A.: Texas Instruments Incorporated, 1981.

155. Thornton-Trump, A. B., "Gait Analysis," in: Akkas, N., ed., Progress in Biomechanics, Netherlands: Sijthoff and Noordhoff, 1979, 127-155.

156. Thornton-Trump, A. B. and Daher, R., "The Prediction of Reaction Forces from Gait Data," Journal of Biomechanics, 1975, 8, 173-177.

157. Townsend, M. A., "Dynamics and Coordination of Torso Motions in Human Locomotion," Journal of Biomechanics, 1981, 14:11, 727-738.

158. Tyson, H. N., Kinematics. New York: John Wiley and Sons, Inc., 1966.

159. Uicker, J. J., Jr., "Dynamic Behaviour of Spatial Linkages, Part I - Exact Equations of Motion," Journal of Engineering for Industry, Transactions of the ASME, New York, February, 1969, 251-257.

160. Van Gheluwe, B., "A New Three-Dimensional Filming Technique Involving Simplified Alignment and Measurement Procedures," in: Nelson, R. C., and Morehouse, C. A., eds., Biomechanics IV. Baltimore: University Park Press, 1974, 476-481.

161. Vaughn, C. L., Andrews, J. G., and Hay, J. G., "Selection of Body Segment Parameters by Optimization Methods," Journal of Biomechanical Engineering, (Transactions of the ASME), February, 1982, Vol. 104, 38-44.

162. Voloshin, A., Wosk, J., and Brull, M., "Force Wave Transmission Through the Human Locomotor System," Journal of Biomechanical Engineering, Transactions of the ASME, February 1981, 103, 48-50.

163. Wallmark, J. I., "A New Semi-conductor Photocel Using Lateral Photoeffect," Proc. IRE, 1957, 45, 474-484.*

164. Wells, R. P., Winter, D. A., "Assesment of Signal and Noise in the Kinematics of Normal, Pathological and Sporting Gaits," Proceedings of the Special Conference of the Canadian Society for Biomechanics, London, Ontario, Canada, 27-29 October, 1980, 92-93.

165. Williams, M., and Lissner, H. R., Biomechanics of Human Motion, Saunders, 1977.

166. Winter, D. A. Biomechanics of Human Movement. New York: Wiley-Interscience, John Wiley and Sons, 1979.

167. Winter, D. A., "Overall Principle of Lower Limb Support During Stance Phase of Gait," Journal of Biomechanics, 1980, 13, 923-927.

168. Winter, D. A., and Robertson, D. G. E., "Joint Torque and Energy Patterns in Normal Gait," Biological Cybernetics, 1978, 29, 137-142.

169. Winter, D. A., Quanbury, A. O., Hobson, D. A., Sidwall, H. G., Reimer, G., Trenholm, B. G., Steinke, T., and Shlosser, H., "Kinematics of Normal Locomotion -- A Statistical Study Based on Television Data," Journal of Biomechanics, 1974, 7, 479-486.

170. Winter, D. A., Sidwall, H. G., and Hobson, D. A., "Measurement and Reduction of Noise in Kinematics and Locomotion," Journal of Biomechanics, 1974, 7, 157-159.

171. Woltring, H. J., "Calibration and Measurement in 3-Dimensional Monitoring of Human Motion by Optoelectronic Means," Biotelemetry, 1976, 3, 65-97.

172. Woltring, H. J., "Comment on the Paper 'A Technique for Obtaining Spatial Kinematic Parameters of Segments of Biomechanical Systems from Cinematographic Data'," Journal of Biomechanics, Letter to the Editor, 1980, 14:4, 277.

173. Woltring, H. J., "Measurement and Control of Human Movement," Ph.D. Thesis, University of Nijmegen, May 1977.

174. Woltring, H. J., "New Possibilities for Human Motion Studies by Real-Time Light Spot Position Measurement," Biotelemetry, 1974, 1, 132-146.*

175. Woltring, H. J., "Planar Control in Multi-Camera Calibration for 3-D Gait Studies," Journal of Biomechanics, 1980, 13, 39-48.

176. Woltring, H. J., "Single- and Dual- Axis Lateral Photodetectors of Rectangular Shape," IEEE Transactions on Electronic Devices, 1975, 22, 581-590, and 1101.*

177. Woltring, H. J., "State Estimation from Multi-Camera Landmark Observations in 3-D Reconstruction of Locomotion Patterns," Proceedings of 4th Congress of the International Society of Electrophysical Kinesiology, Boston, Massachusetts, 1979.*

178. Woltring, H. J., and Marsolais, E. B., "Optoelectric (Selspot) Gait Measurement in Two- and Three- Dimensional Space - A Preliminary Report," Bulletin of Prosthetics Research, Fall 1980, BPR 10 -34, 17:2, 46-52.

179. Wright, D. G., Desai, S. M., and Henderson, W. H., "Action of the Subtalar and Ankle-Joint Complex During Stance Phase Walking," Journal of Bone and Joint Surgery, March, 1964, 46-A:2, 361-382.

180. Zarrugh, M. Y., "Kinematic Prediction of Intersegment Loads and Power at the Joints of the Leg in Walking," Journal of Biomechanics, 1981, 14:10, 713-725.

181. Zarrugh, M. Y., "Power Requirements and Mechanical Efficiency of Treadmill Walking," Journal of Biomechanics, 1981, 14:3, 157-165.

182. Zarrugh, M. Y., and Radcliffe, C. W., "Computer Generation of Human Gait Kinematics," Journal of Biomechanics, 1979, 12, 99-111.

183. Zatsiorsky, V. M., and Seluyanov, V. N., "The Mass and Inertia Characteristics of Main Segments of Human Body," A paper presented to the VII International Congress of Biomechanics, Warsaw, September 1979.

END OF FILM

EASE REWIND

A THREE-DIMENSIONAL KINEMATIC ACQUISITION
AND INTERSEGMENTAL DYNAMIC ANALYSIS SYSTEM
FOR HUMAN MOTION

by

Erik Karl Antonsson

B.S. Cornell University
(1976)

S.M. Massachusetts Institute of Technology
(1978)

SUBMITTED TO THE DEPARTMENT OF MECHANICAL ENGINEERING
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 1982

VOLUME II

APPENDICIES
Eng.

The appendicies included here deal with the hardware and software designed and built in the course of this investigation. Since this is an Engineering-Design thesis the bulk of the work centered on the TRACK III system design and fabrication, however, the work reported in the earlier chapters is of paramount interest and therefore comprises the main body of the thesis; the appendicies cover the design. Each of the three major components of TRACK III (version 3) are discussed in the appendicies: the Selspot, the Forceplate, and the Software package.

APPENDIX A

KISTLER FORCEPLATFORM SPECIFICATIONS

OPERATING INSTRUCTIONS for
PLATFORM TYPES 9281A...

Note

The operating instructions for Platform Types 9281A... are not yet in the usual form available.

The present issue will be replaced by the complete new instructions as soon as possible. With a few exceptions the operating instructions 86.9261, ed. 12.74, also apply to the Type 9281A11.

The following deals with these exceptions (referring to the sections concerned).

1.2.   Technical data

See data-sheet 6.9281A11.

1.4.   Construction of the measuring platform

Instead of 8 TNC connectors, Type 9281A11 has a 9-pole Fischer connector.

2.2.   Mounting frame

The platform must be mounted on a rigid base. The mounting frame Type 9423 must be used unless a mounting surface (steel) of similar quality is available (e. g. table of a large machine tool).

The platform has 4 circular feet, concentric with the transducers.

The mounting frame provides 4 flat, ground surfaces onto which the platform can be fixed. The mounting frame is ideally grouted into a solid concrete block.

Procedure for concreting-in the mounting frame:

● Insert in each of the 4 threaded holes on the mounting frame a M12-eyebolt (supplied with platform).

● Place frame on prepared concrete foundation.

● Level the frame and adjust to proper height, so top surface of platform will match surrounding surface (take into account tartan or other coverings, if any).

● Grout frame in, up to the top surfaces of the 4 sides of the frame. Thus, the 4 ground supporting areas still protrude slightly above the concrete surface.

Important: If the platform is installed outdoors, the concrete block must be bedded in gravel at a frost-proof depth.

● A cable duct for the special cable type 1681A... must be provided. A plastic tube of 50 mm internal diameter is recommended. The cable leaves the platform at the (−x/+y) corner (see data-sheet 6.9281 for exact position).

● Make sure that cable duct and mounting area for platform have a proper drainage.

● The required cut-out in the floor must be at least 404 by 604 mm, centered with the mounting frame.

2.3.   Mounting the platform

- After concrete has set, unscrew eyebolts from mounting frame and carfully clean the 4 supporting areas.

- Screw the 4 eyebolts into the holes in the corners of the platform.

- Using the small M8-eyebolt (also supplied with the platform), lift out the 4 circular covers in the top plate of the platform by screwing in the eyebolt and pulling straight up.

- Lift   the platform on the eyebolts in the corners and place it on the mounting frame. Check whether there is any discernible wobble over a diagonal by tapping with the fists alternatively near 2 diagonally opposite corners of the top plate. If there is any play, this can be readily heard.

- If there is discernible play, lift platform out and use one of the shims supplied with the platform to eliminat the play. Should the thinnest shim supplied be too thick, you may use a piece of household aluminum foil.

- Once the proper shiming has been determined, lift out the platform and connect the cable coming out of the cable-duct.

Note: It does not matter which end of the cable 1681A... goes to the platform, the cable works properly both ways! Put cover of cable plug onto cover of socket on platform to prevent them from getting dirty.

- Carefully lower platform onto mounting frame, taking care that the previously determined shims are in place. Make sure that the cable is not jammed between platform and frame!

- Lightly grease threads of the 4 M12-fixing bolts supplied with the platform and insert them through the 4 holes in the platform.
  Before tightening them, recheck that there is absolutely no discernible play (wobble) diagonally.
  Furthermore, carefully center the platform so there is an even gap of at least 2 mm all around the top plate from the edges of the cutout in the floor.

- Tighten the 4 M12-bolts with a torque wrench with 90 Nm.

- Replace the 4 circular covers. Press down on them with the palm of your hand and rotate them. This will relieve the tension in the O-ring, so there is no play between the covers and their seats.

- Remember to shortcircuit the cable (already attached to the platform) before connecting it to the electronic This is simply done by touching all contact pins in the connector with a paper clip or a piece of blank wire (see operating instructions 311.5001, Charge Amplifier).

2.4.   Special mounting

In certain applications maximum forces occur that are well below the maximum ranges. This is particularly true with investigations outside the field of sport, such as in orthopedics, rehabilitation, neurology etc.

In such cases the platform may also be laid on floors, such as linoleum. The following points must be strictly observed, however:

- The floor must be the same under all 4 supporting surfaces (same material, same elasticity).

- The platform must rest only on its 4 supporting areas.

- It must be installed following exactly the procedure mentioned above under 2.3. (shimming), except that the fixing bolts are not used.

- The forces applied must not exceed 25 % of the ranges.

If there is any doubt, consult our representative.

For best results, the platform should always be mounted on the frame Type 9423.

3.2./3.4 Switching possibilities on the platform

The Type 9281A... platform does not have built-in switches.

For obtaining the various combinations shown on Fig. 10, a distribution box Type 5405 can be used. Instead through switches, the desired connections are made with soldered connections.

Caution: When soldering in such connections, be sure that there are no charge amplifiers connected. Also keep interior of box dry and clean, do not touch any insulator with bare fingers, otherwise the insulation resistance will be lowered and drift will result. Make sure that there are never 2 charge amplifiers paralleled at their inputs, otherwise they will immediately saturate.

For further details, consult our distributor.

4.1. Nomenclature

Note that for the Type 9281A11 Platform, the following dimensions are different from the ones shown for former Type 9261A:

$a = 120$ mm
$b = 200$ mm
$a_{z_0} = -54$ mm

Otherwise, the same analysis is also valid for the Type 9281A11.

Enclosures:

— Data-sheet 6.9281
— Drawing ZS9423 (mounting frame)
— Operating Instructions 86.9261Ae

MEHRKOMPONENTEN-MESSPLATTFORM FÜR BIOMECHANIK UND TECHNIK
MULTICOMPONENT MEASURING PLATFORM FOR BIOMECHANICS AND INDUSTRY

9281A11 6.9281de 6.77

Vorläufige Daten

Quarzkristall-Mehrkomponenten-Messplattform zum Messen einer beliebig wirkenden Kraft in 3 orthogonalen Komponenten. Ebenso können die Koordinaten des momentanen Kraftangriffspunktes und das freie Moment um eine Achse normal zur Plattform bestimmt werden. Weiter Messbereich, hohe Eigenfrequenz.

Tentative Data

Quartz Multicomponent Measuring Platform for measuring the 3 orthogonal components of any force acting on it. The coordinates of the instantaneous point of force application and the torque with respect to an axis normal to the platform. Wide measuring range, high natural frequency.
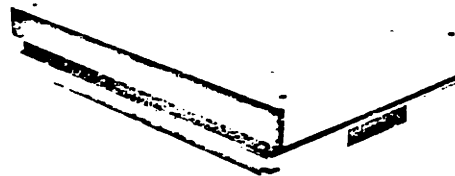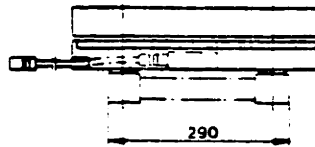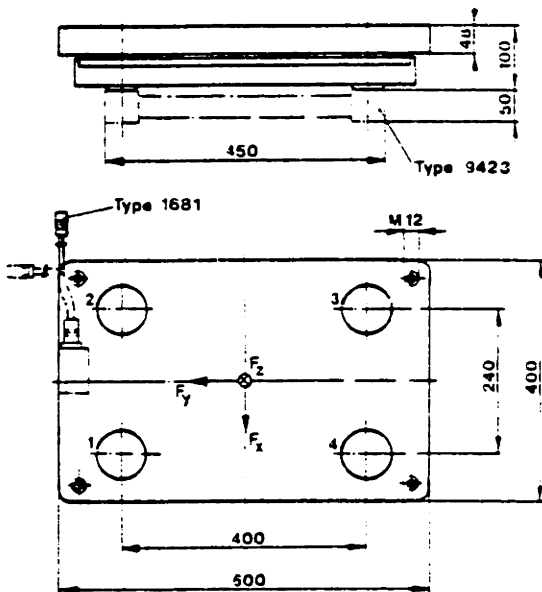
Technische Daten (Typ 9281A11)   Technical data (Type 9281A11)

Bereiche: / Ranges:			
F_x, F_y je / each	kN	-10 ... 10	
F_z	kN	-10 ... 20	
Überlast: / Overload:			
F_x, F_y je / each	kN	-15/15	
F_z	kN	-15/30	
Ansprechschwelle / Threshold	mN	<20	
Empfindlichkeiten: / Sensitivities:			
F_x, F_y je / each	pC/N	-5	
F_z	pC/N	-3.8	
Empfindlichkeitsänderung innerhalb Deckplattenfläche / Variation of sensitivities within surface of top plate	%	≤±1	
Linearität / Linearity	%FSO	≤±0.5	
Hysterese / Hysteresis	%FSO	≤0.5	
Übersprechen: / Cross talk:			
$F_x \rightleftarrows F_y$, $F_z \rightarrow F_{x,y}$	%	≤±1	
$F_{x,y} \rightarrow F_z$	%	≤±2	
Eigenfrequenz / Natural frequency	kHz	>1	
Isolationswiderstand / Insulation resistance	TΩ	>10	
Betriebstemperaturbereich / Operating temperature range	°C	-20 ... 70	
Anschluss für Kabel / Connector for cable	Type	1631A...	
Anzugsdrehmoment der M12 Montageschrauben / Tightening torque for M12 mounting bolts	Nm	90	
Gewicht / Weight	kg	41	

1 N (Newton) = 1 kg · m · s⁻² = 0.1019...kp = 0.2248...lbf; 1 kp = 1 kgf = 9.80665 N; 1 lbf = 4.448...N; 1 in = 25.4 mm; 1 TΩ = 10¹² Ω

Beschreibung

Die Messplattform besteht aus einem Grundrahmen, auf den vier 3-Komponenten-Kraftaufnehmer unter hoher Vorspannung montiert sind. Auf diese vier Aufnehmer ist eine auswechselbare Deckplatte montiert.

Die Standardausführung Typ 9281A11 besitzt eine massive Aluminium-Deckplatte. Deckplatten aus Stahl, mit oder ohne T-Nuten, sind ebenfalls erhältlich.

Die Ausgangssignale stehen an einem 9-poligen Anschlussstecker zur Verfügung. Das Instrument ist rostfrei und spritzwasser-geschützt.

Anwendungsbeispiele

- Biomechanik: Ganganalyse, Tierstudien
- Sport: Bewegungsstudien, Training
- Orthopädie: Rehabilitation, Anpassen von Prothesen
- Neurologie: Posturography, Mikrovibrationen
- Technik: Radkrafte im Fahrzeugbau

Montage

Die Plattform wird mit vier M12-Schrauben auf den Montagerahmen Typ 9423 oder eine starre und plane Stahlplatte montiert. Der Montagerahmen Typ 9423 eignet sich vor allem für das Eingiessen in ein Betonfundament und ist separat zu bestellen.

Elektronik

Mit den Elektronik-Einheiten Typen 9805 und 9851 können die drei Komponenten F_x, F_y und F_z sowie die Koordinaten a_x und a_y des momentanen Angriffspunktes der wirkenden Kraft sowie das freie Moment M'_z bestimmt werden (siehe Datenblätter 10.9805 und 10.9851). Angeschlossen wird die Plattform mittels eines 8-adrigen Kabels Typ 1681A5 (5 m lang, andere Längen erhältlich).

Description

The measuring platform consists of a base frame onto which four 3-component force transducers are mounted under high prestress. An interchangeable top plate is then mounted on the four transducers.

The standard version Type 9281A11 has a solid aluminium top plate. Steel top plates, with or without T-slots, are available also.

The output signals are available from a 9-pole connector. The instrument is rustproof and splashwater proof.

Examples of Applications

- Biomechanics: Gait analysis, animal studies
- Sport: studies of movements, training
- Orthopedics: rehabilitation, fitting of prosthesis
- Neurology: posturography, microvibrations
- Industry: wheel forces in vehicle research

Mounting

The platform can be fixed with four M12-bolts on a mounting frame Type 9423 or a rigid and flat steel plate. The mounting frame Type 9423 is especially suitable for being grouted into a concrete base and has to be ordered separately.

Electronics

With the electronic units Types 9805 and 9851 the three components F_x, F_y and F_z as well as the instantaneous point of application of the acting force as well as the torque M'_z can be determined (see data sheets 10.9805 and 10.9851). The platform is connected with a 8-pole cable Type 1681A5 (5 m long, other lengths available).
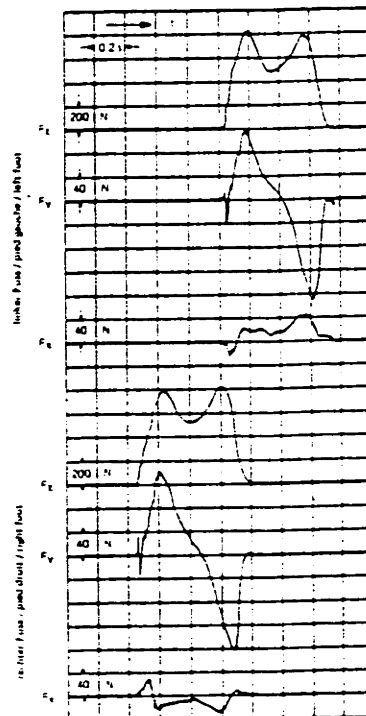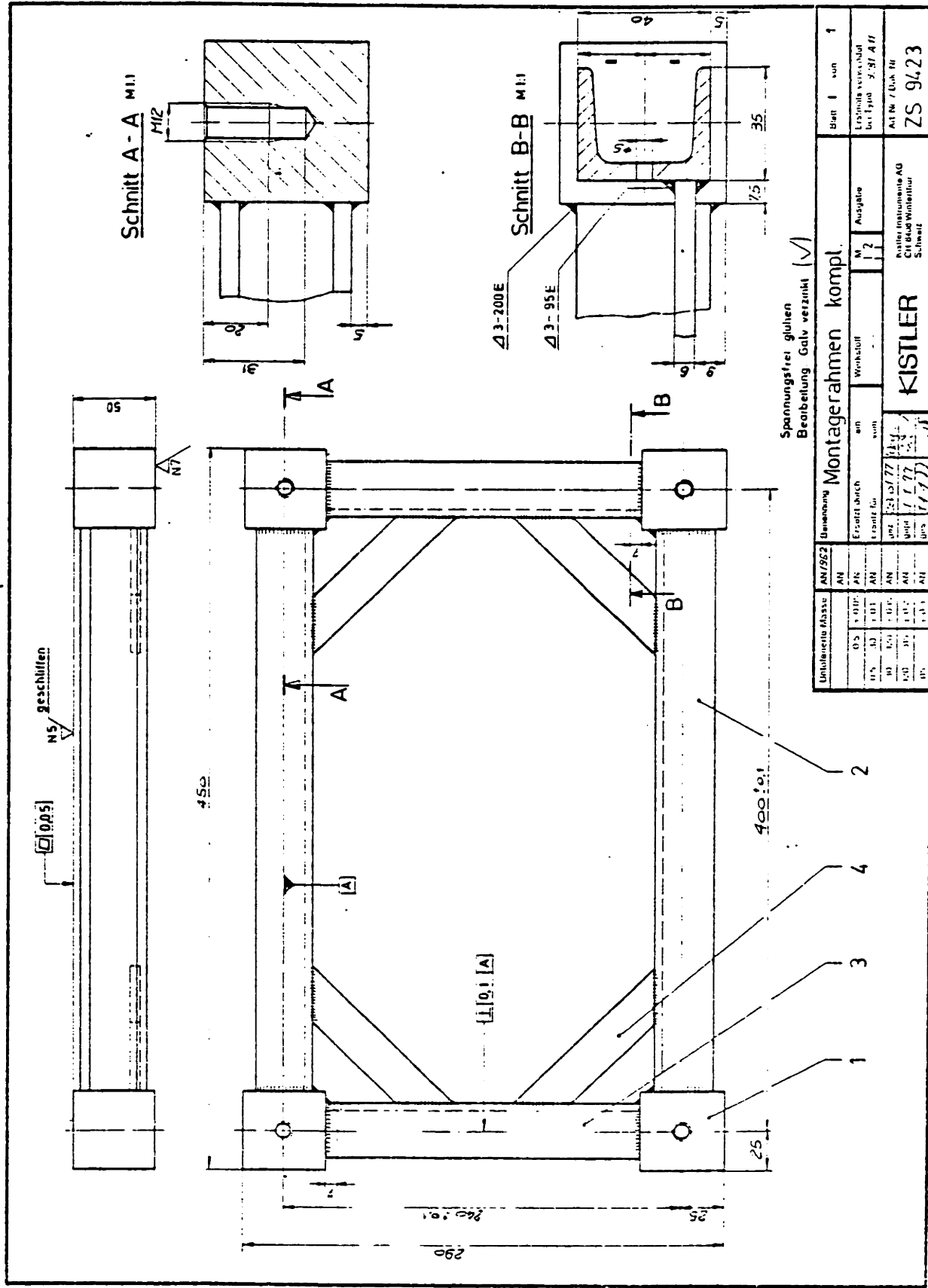
Beispiel:

Fusskräfte beim normalen Gehen. Das Diagramm wurde mit zwei nebeneinanderliegenden Plattformen aufgenommen, sodass jeder Fuss einzeln erfasst werden konnte. F_z ist die Vertikalkomponente, F_y die Horizontalkomponente in Gehrichtung ($-F_y$ nach vorn) und F_x die seitliche Komponente ($+F_x$ nach links). Die Versuchsperson wog 75 kg.

Example:

Forces exerted on feet during normal walking. The diagram was recorded with two platforms side-by-side in order to observe each foot singly. F_z is the vertical component, F_y the horizontal component in direction of walk ($+F_y$ forward) and F_x the lateral one ($+F_x$ to the left). Weight of test subject: 75 kg.

KISTLER Piezo-Messtechnik

KIAG SWISS®

3-Komponenten-Dynamometer
Dynamomètre à 3 composantes
3-component dynamometer

Type **9281 A** SN **104177**

Kalibrierter Bereich Gamme étalonnee Calibrated range [N]	F_x 0...1 000	F_y 0...1 000	F_z 0...2 000
Empfindlichkeit Sensibilité Sensitivity [pC/N]	- 8,16	- 8,19	- 3,86
Linearität Linearite Linearity < ± % FSO	0,3	0,3	0,3

Betriebstemperaturbereich
Gamme de temp. d'utilisation [°C] -20...70
Working temperature range

Kalibriert bei
Etalonne a 20 °C
Calibrated at by He Date 6.7.78

1N(Newton) = 1kg·m·s⁻¹ = 0.1019...kp = 0.2248...lbf
1kp = 1kgf = 9.80665 N
1lbf (pound force) = 4.448...N

$$1N(Newton) = 1kg \cdot m \cdot s^{-1} = 0.1019...kp = 0.2248...lbf$$
$$1kp = 1kgf = 9.80665\ N$$
$$1lbf\ (pound\ force) = 4.448...N$$
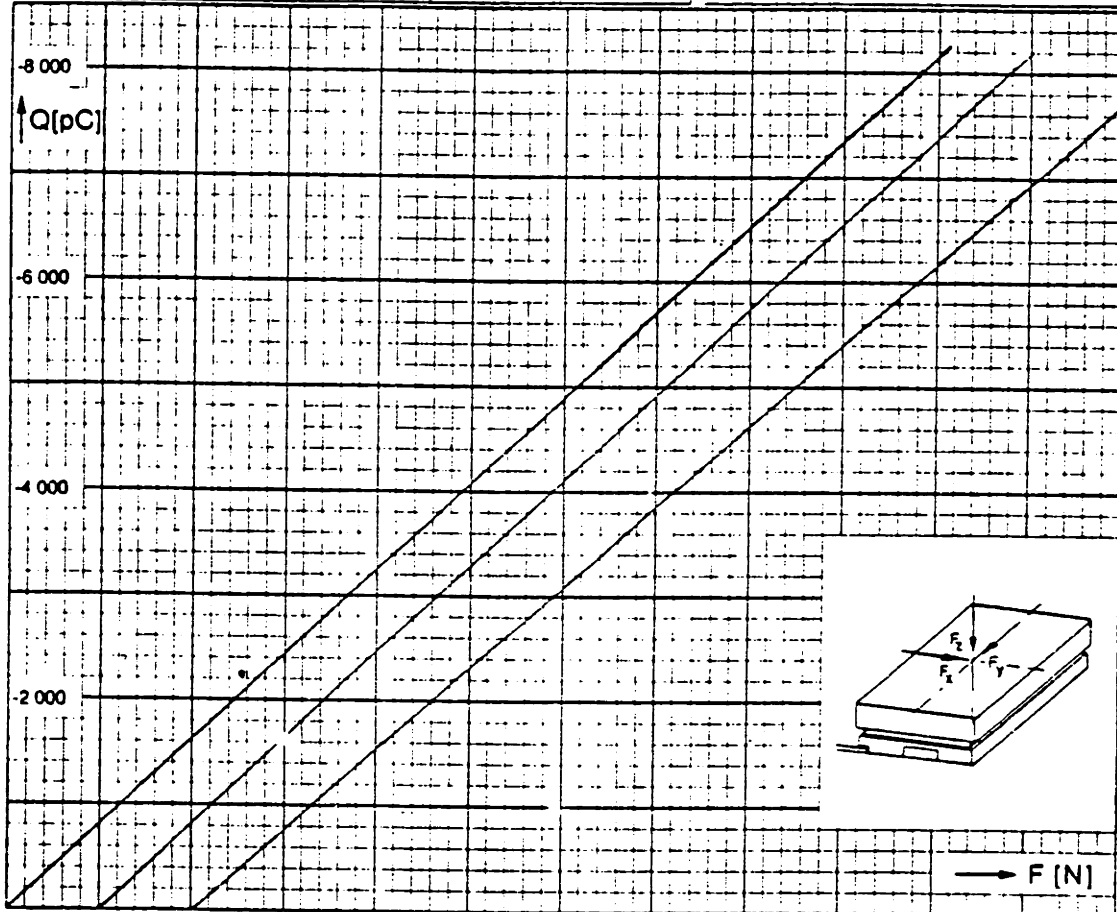
x	0	100	200	300	400	500	600	700	800	900	1 000	
y	0	100	200	300	400	500	600	700	800	900	1 000	
z	0	200	400	600	800	1 000	1 200	1 400	1 600	1 800	2 000	

Übersprechen / Cross talk

	%
x → y	
y → x	
x → z	
y → z	
z → x	
z → y	

KISTLER Piezo-Messtechnik

KIAG SWISS

3-Komponenten-Dynamometer
Dynamomètre à 3 composantes
3-component dynamometer

Type 9281 A   SN 104177

Kalibrierter Bereich Gamme etalonnée Calibrated range	[N]	F$_x$ 0...10 000	F$_y$ 0...10 000	F$_z$ 0...20 000
Empfindlichkeit Sensibilité Sensitivity	[pC/N]	- 8,13	- 8,18	- 3,95
Linearität Linearite Linearity	< ± % FSO	0,3	0,3	0,3

Betriebstemperaturbereich
Gamme de temp. d utilisation [°C]   -20...70
Working temperature range

Kalibriert bei
Etalonne à   20 °C
Calibrated at   by He Date 6.7.78

1N(Newton) = 1kg·m·s^{-1} = 0,1019...kp = 0.2248...lbf
1kp - 1kgf - 9.80665 N
1lbf (pound force) = 4.448...N
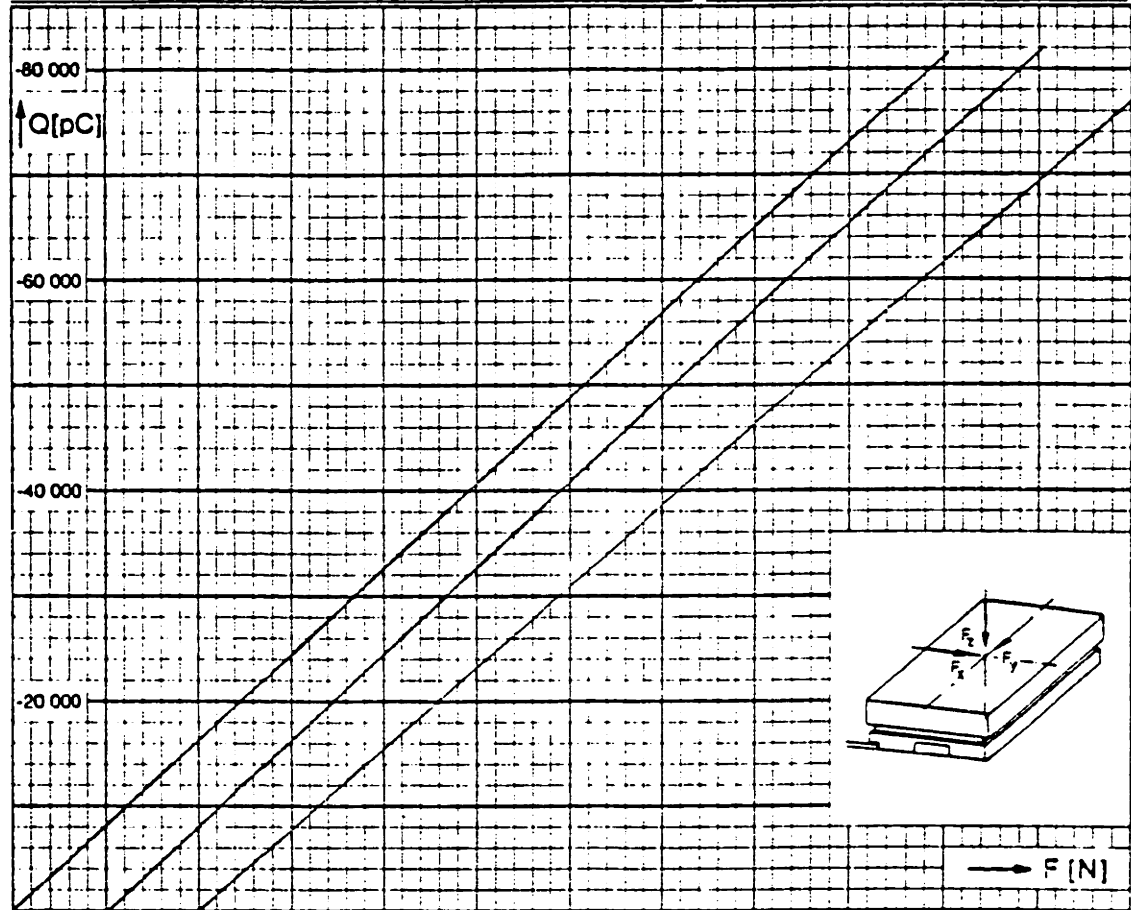
0	1 000	2 000	3 000	4 000	5 000	6 000	7 000	8 000	9 000	10 000	→ x
y → 0	1 000	2 000	3 000	4 000	5 000	6 000	7 000	8 000	9 000	10 000	
z →		0	2 000	4 000	6 000	8 000	10 000	12 000	14 000	16 000	18 000 20 000

Übersprechen / Cross talk

x → Y - 0,1 %
y → x 0,5 %
x → z 0,8 %
y → z 0,5 %
z → x - 0,1 %
z → Y 0,2 %

4.   SIGNIFICANCE AND PROCESSING OF THE OUTPUT SIGNAL

4.1.   Nomenclature

a Distance between transducer axes and y-axis (for 9261A: a = 132 mm)

a_x ⎫
a_y ⎬   Coordinates of the intersection point of the line of action of the resultant with the (x, y)-plane translated parallel by a_z (working plane)

a_z Distance between working plane and (x, y)-plane

b Distance between transducer axes and y-axis (for 9261A: b = 220 mm)

F Resultant of all forces acting on the platform

F_x ⎫
F_y ⎬   Components of the resultant
F_z ⎭

$F_{x_1} ... F_{x_4}$ Force components in x-direction acting on transducers 1 ... 4
$F_{y_1} ... F_{y_4}$ Force components in y-direction acting on transducers 1 ... 4
$F_{z_1} ... F_{z_4}$ Force components in z-direction acting on transducers 1 ... 4

M Resultant moment vector after reducing all acting forces and moments to the origin of coordinates

M_x ⎫
M_y ⎬   Components of the resultant moment vector
M_z ⎭

M' Free moment vector resulting from all acting free moments

M'_x ⎫
M'_y ⎬   Components of the resultant free moment vector
M'_z ⎭

O Origin of coordinates

σ_z Normal mechanical stresses in the test object/measuring platform interface ($\sigma > 0$ corresponds to compression

x ⎫
y ⎬   Reference coordinate system
z ⎭
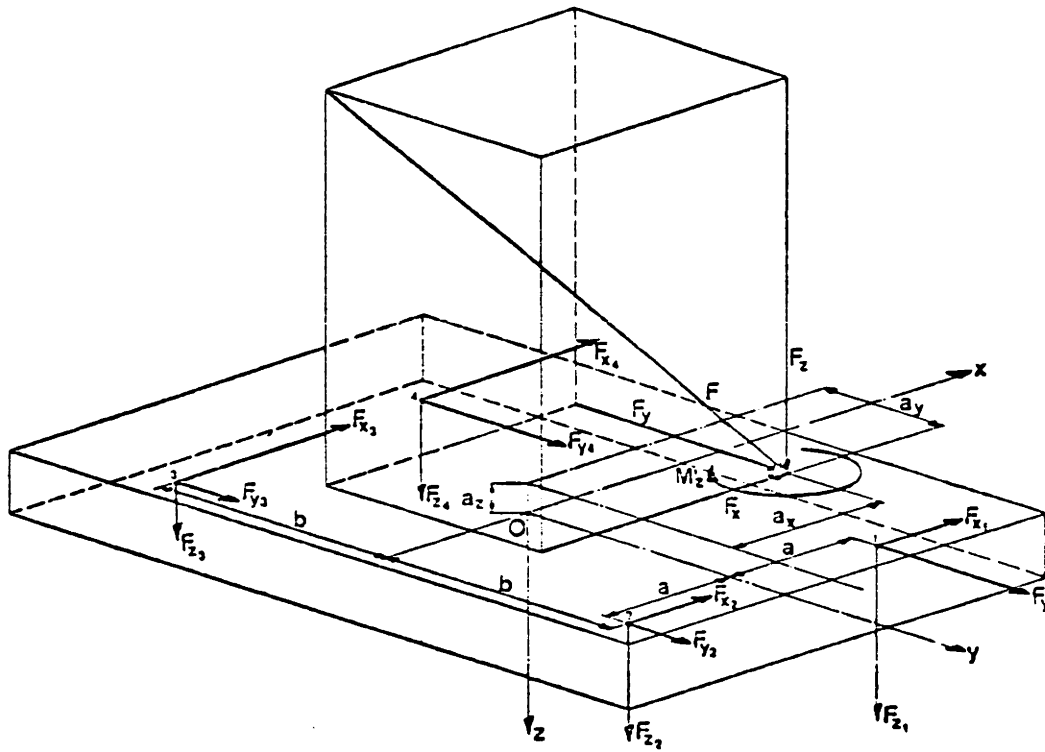
Fig. 9: Definition of the measurands

4.2.   Reference coordinate system

A positive Cartesian coordinate system is used as reference coordinate system. The z-axis is normal to the platform surface, and the positive direction points into the platform. The positive y-axis points towards the end with the connections.

A positive Cartesian coordinate system is defined as follows:

> Looking in the direction of the positive x-axis, a rotation of the positive y-axis through 90° into the positive z-axis appears clockwise. Similarly the positive moment vector M_x constitutes a clockwise rotation looking in the direction of the positive x-axis.

The same is true of the y- and z-axes by cyclic rotation.

4.3.   Correlations between forces and moments acting on the measuring platform and output signals

Note: The following individual transducer outputs are paralleled permanently and are designated and defined as measuring platform outputs as follows:

Output:

X_{1+2} : $F_{x_{1+2}} = F_{x_1} + F_{x_2}$

X_{3+4} : $F_{x_{3+4}} = F_{x_3} + F_{x_4}$

Y_{1+4} : $F_{y_{1+4}} = F_{y_1} + F_{y_4}$

Y_{2+3} : $F_{y_{2+3}} = F_{y_2} + F_{y_3}$

Components of the resultant:

$$F_x = F_{x_{1+2}} + F_{x_{3+4}} \tag{1}$$

$$F_y = F_{y_{1+4}} + F_{y_{2+3}} \tag{2}$$

$$F_z = F_{z_1} + F_{z_2} + F_{z_3} + F_{z_4} \tag{3}$$

Resultant:

$$F = (F_x^2 + F_y^2 + F_z^2)^{\frac{1}{2}} \tag{4}$$

determined by the angle γ' to the (x, y)-plane

$$\sin \gamma' = \frac{F_z}{|F|} \tag{5}$$

and the angle δ of the projection of F onto the (x, y)-plane to the positive x-axis:

$$\tan \delta = \frac{F_y}{F_x}$$

or determined by the direction cosines

angle α between the resultant and the positive x-axis

$$\cos \alpha = \frac{F_x}{|F|} \tag{7}$$

angle β between the resultant and the positive y-axis

$$\cos \beta = \frac{F_y}{|F|} \tag{8}$$

angle γ between the resultant and the positive z-axis

$$\cos \gamma = \frac{F_z}{|F|} \tag{9}$$

(Note: $\cos^2 \alpha + \cos^2 \beta + \cos^2 \gamma = 1$)

Components of the resultant moment vector (referred to origin of coordinate)

$$M_x = b \ (F_{z_1} + F_{z_2} - F_{z_3} - F_{z_4}) \tag{10}$$

$$M_y = a \ (-F_{z_1} + F_{z_2} + F_{z_3} - F_{z_4}) \tag{11}$$

$$M_z = b \ (-F_{x_{1+2}} + F_{x_{3+4}}) + a \ (F_{y_{1+4}} - F_{y_{2+3}}) \tag{12}$$

Generally a multicomponent force and moment measuring system yields not more than the 3 components of the resultant, its direction in space (though not its position!) and the 3 components of the resultant moment vector referred to the origin of coordinates.

In other words:

Only the resultant translated through the coordinate origin and the resulting moment vector referred to the origin of coordinates are obtained.

Let us take the equations for determining the resultant moment vector M (see Fig. 9):

$$M_x = a_y \cdot F_z - a_z \cdot F_y + M'_x \qquad (13)$$

$$M_y = -a_x \cdot F_z + a_z \cdot F_x + M'_y \qquad (14)$$

$$M_z = a_x \cdot F_y - a_y \cdot F_x + M'_z \qquad (15)$$

and assume for the time being that no free moments are acting on the measuring system, i.e.:

$$M'_x = M'_y = M'_z = 0$$

Since F_x, F_y, F_z, M_x, M_z and M_z are measured by the system, for determining a_x, a_y and a_z equations (13), (14) and 15 can be rewritten as follows:

$$0 \cdot a_x + F_z \cdot a_y - F_y \cdot a_z = M_x$$

$$-F_z \cdot a_x + 0 \cdot a_y + F_x \cdot a_z = M_y \qquad (16)$$

$$F_y \cdot a_x - F_x \cdot a_y + 0 \cdot a_z = M_z$$

The determinant of this system equals 0, because all coefficients of the main diagonal are equal to 0, i.e. the system is indeterminate for a_x, a_y and a_z.

Because it is a linear equation system, however. it determines a straight line (i.e. the action line of the resultant) with a_x, a_y and a_z as the coordinates of a point running on it. The point of force application is thus simple to find: it is enough to determine the intersection of the action line with a given plane.

Example: For the working plane of the measuring platform Type 9261A the equation is:

$$a_z = -37 \text{ mm} \qquad (17)$$

This value has only to be substituted in equation (16) to obtain the coordinates a_x and a_y of the force application point in the working plane.

In the general case, however, there are free moments acting on a multicomponent measuring system as well. System (16) then appears as follows:

$$0 \cdot a_x + F_z \cdot a_y - F_y \cdot a_z = M_x - M'_x = ?$$

$$-F_z \cdot a_x + 0 \cdot a_y + F_x \cdot a_z = M_y - M'_y = ? \qquad (18)$$

$$F_y \cdot a_x - F_x \cdot a_y + 0 \cdot a_z = M_z - M'_z = ?$$

The measuring system can measure only F_x, F_y, F_z, M_x, M_y and M_z (see equations (1), (2), (3), (10), (11) and (12).

Though the components M'_x, M'_y and M'_z of the free moment vector are contained in the components M_x, M_y and M_z of the resultant moment vector, their magnitude is not known.

In other words:

Since M'_x, M'_y and M'_z are not known, the constants of system (18) also remain unknown. The equation system (18) not only determines the action line of the resultant therefore, but also comprises all straight lines parallel to the action line.

Or to put it in another way yet:

> In the general case it is impossible to say of a measured moment whether it is generated only by a force eccentric to the origin of coordinates, or only by a free moment, or by any combination of the two.

Special case of measuring platform Type 9261A:

No free moments M'_x and M'_y can act on the platform, thus:

$$M'_x = M'_y = 0$$

Why?

In the intended applications of the measuring platform (standing, walking, running, driving, rolling, jumping on and off it) no tensile stresses are imposed on its surface, i.e. the interface test object/measuring platform:

$$\sigma_z > 0 \tag{19}$$

from which it also follows that

$$F_z = \int \sigma_z \cdot dA \geq 0 \tag{20}$$

On the other hand a free moment M'_x or M'_y would impose tensile stresses too($\sigma_z < 0$), because a free moment can always be represented by a force couple (two parallel forces, equal but of opposed signs). But as long as the condition $\sigma_z \geq 0$ is fulfilled, no force couple forming M'_x or M'_y can arise.

For the measuring platform Type 9261A the equation system (18) can therefore be written as follows:

$$
\begin{aligned}
0 \cdot a_x + F_z \cdot a_y - F_y \cdot a_z &= M_x \\
-F_z \cdot a_x + 0 \cdot a_y + F_x \cdot a_z &= M_z \\
F_y \cdot a_x - F_x \cdot a_y + 0 \cdot a_z &= M_z - M'_z
\end{aligned}
\tag{21}
$$

From the first two equations the coordinates a_x and a_y of the intersection of the action line of the resultant with the (x, y)-plane shifted by a_z can now be determined directly:

$$a_x = \frac{F_x \cdot a_z - M_y}{F_z} \tag{22}$$

$$a_y = \frac{F_x \cdot a_z + M_x}{F_z} \tag{23}$$

The value of a_z is known: it is the distance of the working plane (parallel to the (x, y)-plane) from the coordinate origin.

With a_x and a_y now known, from the appropriately resolved third equation of (21) the free moment about an axis parallel to the z-axis can also be determined.

$$M'_z = M_z - F_y \cdot a_x + F_x \cdot a_y \tag{24}$$

4.4.   Processing the output signals

Depending on the number of components desired, up to eight charge amplifiers may be needed to convert the output signals of the measuring platform into analog voltages.

Here it may be recalled that the individual outputs can be added together directly by means of the switches provided, e.g. for (1), (2) and (3), i.e. the three force components. In this case only three charge amplifiers are needed (see Fig. 10, A).

For processing (10), (11) and (12) on the other hand, one charge amplifier each must be used for outputs whose sum and difference are required.

The various formulae derived under 4.3. can then be calculated either analogously or digitally.

For the analog calculation, special summing amplifiers Type 5217 and analog dividers Type 5215 are available.

If all six variables (F_x, F_y, F_z, a_x, a_y and M'_z) are to be determined, the Electronic Unit for 6-component Force Measurement Type 9803 may be used. This contains all the necessary equipment: 8 charge amplifiers, 2 summing amplifiers and 1 analog divider (see Fig 10, D).

The various extension phases of a measuring system are shown in Fig. 10.

A: for measuring the three force components F_x, F_y and F_z. Primarily for simpler measurements in sport and gait analysis.

B: for determining the coordinates a_x and a_y of the center of gravity of a person standing on the platform and measuring the force component F_z. Employed mainly for Romberg tests and other investigations into posture control (posturography, statokinesimetry).

    Note: 1. The unused outputs for F_x and F_y on the platform are to be short-circuited (e.g. with short circuit cover Type 1875). See also under 3.2.

                 2. This system is suited only for the applications named, where the horizontal forces F_x and F_y are very small. In all other cases expansion to configuration C is necessary.

C: Combination of A and B. Measures the three force components F_x, F_y and F_z as well as the coordinates a_x and a_y of the momentary force application point. Suited for all application areas.

D: Like C, but measures the free moment M'_z in addition. Versatile system with unlimited application potential .

For further information about these instruments the appropriate operating instructions should be consulted.

The analog output voltages of the charge amplifiers (range ± 10 V) may also be processed digitally. When choosing A/D-converters, make sure in particular that they are sufficiently fast, so that the frequency range is not narrowed.

OPERATING INSTRUCTIONS
CHARGE AMPLIFIER 5001

5001   B 11. 8.70   1 - 12r
5001e

CONTENTS

OPERATING INSTRUCTIONS
CHARGE AMPLIFIER 5001

5001	B 11.	8.72	0.1
	5001e		

Modification Supplement

Beginning with serial number SN 15615 the polarity of the control voltage for remote control has been changed.

NEW is a positive control voltage (+15V) for the charge amplifier 5001. As a distinguishing mark, the designation of the corresponding BNC connector has been changed

      NEW: "REMOTE OPERATE" (+15V, positive)

      (OLD: "RESET" (-15V, negative)).

It is not possible to connect amplifiers with positive control voltage in parallel with amplifiers having negative control voltage for remote control operation.

The change has been made to enable the remote control of measuring installations by data loggers or computers (which today use nearly exclusively positive logic).

The operating instructions B 11.5001e, Ed. 8.70 remain valid with the exception of the above mentioned modification.

Enclosure: B 11.5001e, Ed. 8.70

B 11 .5001 e   8.70   1r

Figures

Tables

Fig. 1: Charge Amplifier 5001

B 11.5001e   8.70   2

I   DESCRIPTION

I - 1   Introduction

The Charge Amplifier Type 5001 is a mains-operated DC amplifier of very high input impedance with capacitive negative feedback, intended to convert the electric charge from a piezoelectric transducer into a proportional voltage on the low impedance amplifier output.

The calibration factor setting (adjustment of transducer sensitivity at the amplifier) makes possible standardized amplifier sensitivities of, for example, 1, 2, 5, etc. mV per mechanical unit (at, psi, kp, lb, g; the mechanical unit concerned is that shown on the calibration sheet of the relevant transducer). The carefully designed calculating disc enables the reciprocal value of the sensitivity to be shown directly as a measuring range (e.g. 100 at/Volt). When a recorder with a deflection sensitivity of 1 V/cm is used, the measuring range (e.g. 100 at/V is transformed to the recording scale (e.g. 100 at/cm). This facilitates working with the amplifier and helps to avoid incorrect adjustments.

The equipment has been tested and approved by the SEV (Swiss Electrotechnical Association) for protection class II, ungrounded operation.

I - 2   Technical data (table 1)

(See also Section I - 3, Characteristics)

Measuring ranges, 12 steps, 1 : 2 : 5 : ...,		
continuous 1 : 10	pC	±10 - 500'000
Range capacitors 1 : 2 : 5 : ...	pF	10 - 50'000
Calibration factor setting	pC/M.U.	0,1 - 110'000
(transducer sensitivity, M.U. = mech. unit,		
e.g. at, psi, kp, lb, g)		
Output voltage	V	±10
Voltage saturation, no load	V	< ±15
with load of 50 mA	V	> ±10
Output current (sum of A1 + A2)	mAeff.	50
Output impedance A1 (short circuit protected)	Ω	100
A2 (including 50 mA slow-blow fuse)	Ω	0,5 (35)
Max. input voltage without damage	V	±125, short peak
Insulation at input	Ω	$> 10^{14}$
Frequency range with standard filter (-3dB)	kHz	0 - 180
(with input cables up to 15 m, independent of range)		
Time constant "Long"	s	1'000 - 100'000
"Medium"	s	1 - 5'000
"Short"	s	0,01 - 50
Bleeder resistance (Long, Medium, Short)	Ω	$10^{14}, 10^{11}, 10^{9}$
Linearity (max. error)	%	< 0,05
Accuracy of ranges	%	±1
(of the two most sensitive)	%	±3
Calibration input (calibration capacitor)	pF	1'000 ±0,5%
Sensitivity (max. 30V)	pC/mV	1
Hum and noise	mVrms	max. 2
Cable noise signal (due to capacity, per 1'000 pF)	pCrms	0,02 max.
Max. capacity at input	pF	100'000

B 11.5001e   8.70   2r

Zero stability (long term)	mV	$\leq \pm 20$
Zero stability (dependent on power line)	mV	$\leq 0,2$
Max. drift (due to leakage current)	pC/s	$\pm 0,03$
Temperature drift ("Transd. Sens." setting 10-00/1-00)	mV/°C	max. $\pm 0,5/5$
Cable connectors (receptacles)	Type	BNC neg.
Power connection, Protection Class II	V	220/110 $\pm 20\%$
for ungrounded operation	Hz/VA	50 - 60/5
Permissible ambient temperature	°C	0 - 50
Dimensions (with case)	mm	74 x 145 x 210
Weight	kg	2,1

I - 3   Characteristics

Below are details of the special characteristics of the charge amplifier so far as they require particular explanation. (Terms taken from the table of technical data are underlined continuously.)

a) Measuring ranges: In the technical data, the overall measuring ranges of the charge amplifier are given in pC for each 10V output voltage. The smallest range is that of a 10 pF range capacitor and the 10-turn potentiometer in position 1-00, the largest is that of a 50'000 pF range capacitor and potentiometer position 10-00.

In measuring technology, the range is in practice given differently, as mechanical units per Volt of output voltage, making allowance for the sensitivity of the connected transducer (calibration factor, expressed in pC per M.U.). The mechanical unit concerned at the time is given by the calibration sheet: pC/at, pC/kp, pC/g, pC/psi, etc.

For an unaltered electrical measuring range (pC), the mechanical measuring range (mechanical units) alters according to the transducer sensitivity. See Fig. 2a.

The range indication on the amplifier represents a scale: number of mechanical units per Volt of output voltage. The utilizable measuring range (10V output voltage) corresponds to ten times the range indication. For further details, see section II - 1, Operating philo-
· sophy.

b) Calibration factor setting: This adjusts the amplifier to the sensitivity of the connected transducer. The adjustment is twofold:
On the 10-turn potentiometer "Transd.Sens.", only the numerical order of the transducer sensitivity is set, regardless of the decimal point (e.g., sensitivity of 82.4 pC/at is set at Pos. 8-24).
The calculating disc "Transd.Sens.Range" is used to set the decade of the transducer sensitivity; in the above example, for 82.4 pC/at, the decade "10 - 100" must be set. Shifting the position of the calculating disc has no effect on the amplifier setting: it merely gives the correct decimal point for the range indication. (See also Fig. 2a.)

c) Voltage saturation: The beginning of voltage limitation is sharply defined, which means that the shape of the curve of the recorded signal shows clearly if the amplifier is saturated. This eliminates incorrect interpretation.

d) Output current, output impedance: The two outputs can be loaded to 50 mArms.
Output A1, "OUTPUT 100 Ω", is provided for the connection of high-impedance recorders, e.g. an oscilloscope. To avoid measuring errors due to voltage cross at the 100Ω resistor, the input impedance of the connected equipment should be high (100kΩ give ~ 0,1%, 10kΩ give ~ 1% error). The 100Ω resistance protects the output stage of the amplifier and safeguards it against long-term short-circuit. The output A1 is not influenced by loading output A2.
Output A2, "OUTPUT 50 mA", is provided for the connection of low-impedance recorders, e.g. a galvanometer of UV recording equipment. The output is protected by a 50 mA slow-

blow fuse. The output impedance is made up of less than 0.5Ω internal resistance and about 35Ω fuse resistance. (Special fuses with 10Ω resistance are obtainable.) A load on this output has practically no effect on the output voltage of the 100Ω output.

e) Maximum input voltage without damage to the Dual-MOSFET. The given maximum voltage may be applied only for a short time, as, for example, when connecting a statically charged cable. For testing the insulation, only about 22V may be applied; moreover, a special instrument is required. The electrostatic voltmeter 149 (150V) must not be used for this purpose.

f) Frequency range with standard filter: In principle, the frequency range is dependent on the measuring range (size of negative feedback capacitor = range capacitor) and the cable capacity at the amplifier input. In the case of input cables up to about 15 m (50 ft) in length, the dependence on range shows only above the limit frequency of the standard filter, so that if such cables are used the frequency range is independent of the range set. For cable lengths above 15 m (50 ft) see Fig. 2c. Filters see paragraph II - 2 d.

g) Time constants "Long","Medium", "Short": The time constants are determined by the bleeder resistance and the range capacitor, and so are dependent on the measuring range. The time constant in the negative feedback circuit causes an exponential discharge of the range capacitor.

Bleeder resistance (Long, Medium, Short): In the "Long" position, the insulation resistance of the relevant range capacitor which is switched in acts as a leakage resistance. The resistance is more than $10^{14}\Omega$ for the small capacitors and some $10^{13}\Omega$ for the large ones.

In the "Medium" position, a $10^{11}\Omega$ resistance is connected in parallel with the range capacitor. There is a corresponding time constant for each size of range capacitor.

In the "Short" position, a $10^9\Omega$ resistance is connected.

In the event of faulty currents at the amplifier input - if, for example, a zero deviation causes a faulty current (drift current) to flow through the insulation resistance - the range capacitor becomes charged and the output voltage rises correspondingly.

This charging of the range capacitor, resulting from the fact that the insulation resistance at the input is not infinite and from the zero deviation, is counteracted by the discharge of the range capacitor through the bleeder resistance.

If the insulation resistance at the input is considerably lower than the bleeder resistance, the amplifier will drift into positive or negative saturation, depending on the polarity of the zero deviation.

If, on the other hand, the insulation resistance at the input is greater than the bleeder resistance, the amplifier does not saturate as the discharge prevails.

The advantage of this concept is as follows: the insulation resistance of the transducer and the connecting cable determine the choice of medium or short time constants. Although the time constant alters with the measuring range, it is possible to obtain automatically in each range the highest possible time constant at which the amplifier operates stably, i.e. does not drift to saturation.

B 11.5001e   8.70   3r

	Transd. Sens. Range							
	0,1-1,1	1-11	10-110	100-1,1k	1k-11k	10k-110k	Range	z.B.√p.ex.√e.g.
1,2,5, x	10^4	10^3	10^2	10	1	10^{-1}		82,7 pC/at :
1,2,5, x	10^3	10^2	10	1	10^{-1}	10^{-2}	Mech.	min. 1×10^{-1} at/V
1,2,5, x	10^2	10	1	10^{-1}	10^{-2}	10^{-3}	Units/V	max. 5×10^2 at/V
1,2,5, x	10	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}		

a C_g(pF)
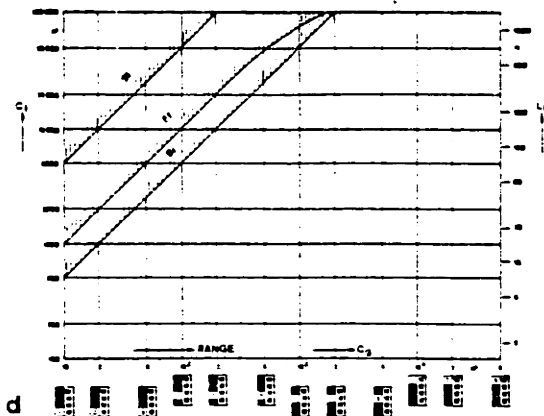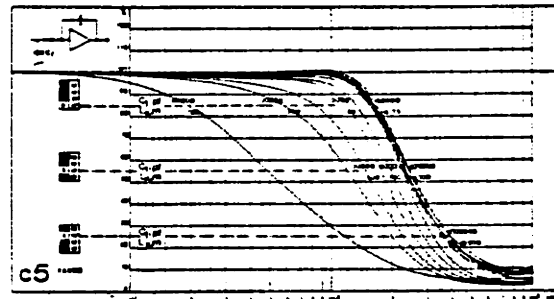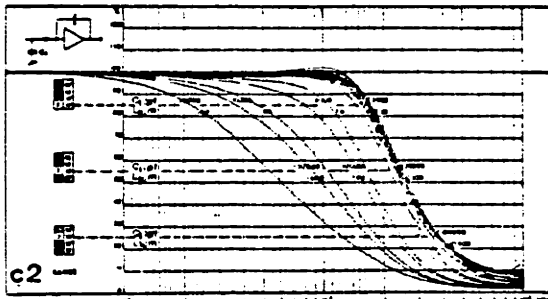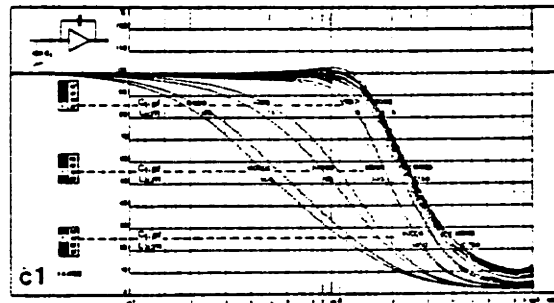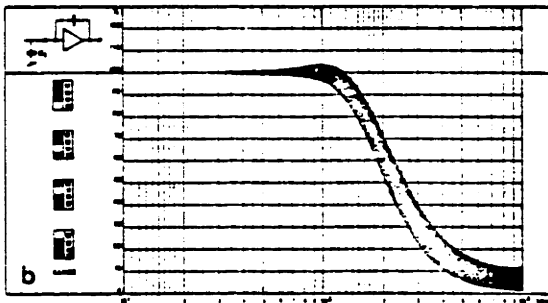
Legend:

a) Table of available ranges

b) Frequency response with standard filter (180 kHz) for all ranges and input capacities < 1000 pF (cable length < 15 m).

c) Frequency response as under b) but for capacities > 1000pF (cable length > 15 m).

d) Limitung factors for the input cable capacity (cable length).

C_i = Input cable capacity (pF)
C_g = Feed-back capacitor (pF)
L_k = Cable length (m)
F_f = Frequency response error -5% at 100 kHz
B_r = Hum and noise 1 mV/10mV at "transd.Sens." 10-00/1-00
S_t = Static error -1%

Fig. 2: Ranges and influence of the input cable capacity

h) Accuracy of the ranges: The accuracies quoted are valid for the time-constant position "Medium". In the "Long" position, an additional measuring error of ÷2% occurs in the most sensitive range (range capacitor 10 pF). In the larger measuring ranges this additional error is correspondingly reduced.

When the transducer sensitivity is set to the 10-turn potentiometer, a maximum setting error of 0.5% should be allowed for.

i) Maximum capacity at the input. The absolute maximum value is given, that is determined by the amplifier characteristics. For a measuring error of 1% at low frequencies, the input capacity should not exceed 500 times the value of the range capacitor; this limit is shown on Fig. 2 d. The influence of the input capacity at higher frequencies is shown in Fig. 2c. Hum and noise also depend on the input capacity; Fig. 2d shows the limit for 1 mV (Transd. Sens. on 10-00) or 10 mV (Transd. Sens. on 1 - 00).

k) Zero jump: When the reset switch is released, a zero jump of 0,2 to 0,5 pC (depending on input capacity, upper value for 10'000 pF) may occur.

I - 4   Principles of operation (see Fig. 3)

The Charge Amplifier 5001 consists of a DC amplifier of high input impedance with capacitive negative feedback, followed by an operational amplifier with adjustable resistive negative feedback. The extremely high input resistance ($10^{14}\Omega$) on the charge amplifier is achieved by using a high voltage (125V) Dual-MOSFET in the input stage.

Assuming that a piezoelectric transducer MA produces a negative charge Q of a given value: without negative feedback, there would be a voltage of $U_e = Q/C$ (Volts) on the amplifier input, where C is the sum of all the capacitances at the input (transducer, connecting cable and other capacitances in parallel). This voltage U_e, amplified by the amplification factor (> 50'000) would appear at the amplifier output. With a negative feedback capacitance, the conditions are different. The output voltage U_g is oppositephase to the input voltage, i.e. for a negative input voltage a positive output voltage appears, and this transmits via the negative feedback capacitor C_g, a positive charge to the input, thus compensating the negative charge of the transducer. The input remains virtually at null potential, while the negative feedback capacitor receives a charge equal to that arising from the transducer, but of opposite polarity.
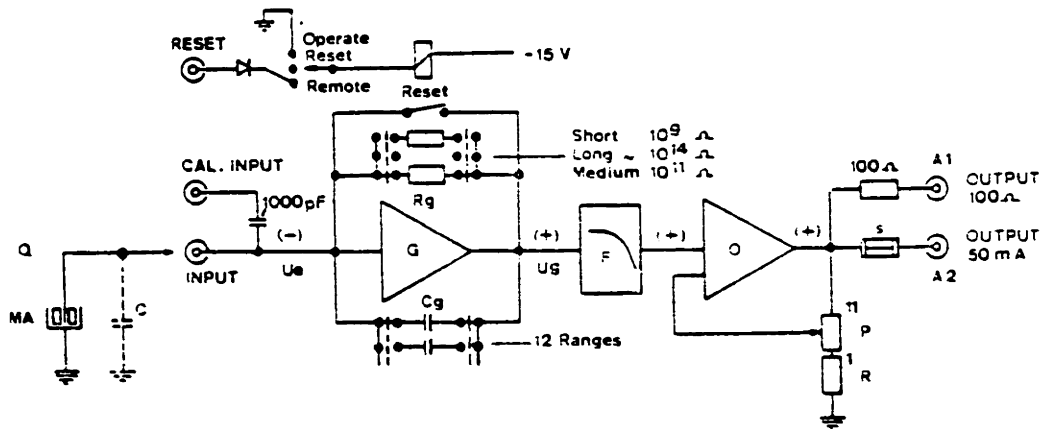
Fig. 3: Schematic diagram

The electrical charge Q on the transducer thus appears to be transmitted, as it were, directly to the negative feedback capacitor C_g.

The voltage on C_g is easy to calculate: $U_g = Q/C_g$. Since the input of the charge amplifier remains virtually at zero potential, then U_g is the output voltage on the charge amplifier. Alteration of the size of the negative feedback capacitor (range capacitor) gives the desired "amplification" (conversion of charge to voltage).

The output voltage of the charge amplifier is further amplified in an operational amplifier with adjustable negative feedback P + R, and is thus adjusted to the transducer sensitivity. The negative feedback is adjusted by the precision 10-turn potentiometer P. A portion k of the output voltage (variable between approx. 0.1 and 1 and determined by the position of P) is fed back to the input, thus varying the amplification between 10 and 1. Considering the performance of both amplifiers, the output voltage is $Q/C_g \cdot k$. In practice, k is adjusted to the sensitivity of the selected transducer.

Example: Without a voltage divider (k = 1), and with a given range capacitor of 100 pF and a transducer sensitivity of, e.g. 82.4 pC/at, the output voltage per at is:

$$U_a = \frac{Q}{C_g \cdot k} = \frac{82,4 \text{ pC/at}}{100 \text{ pF} \cdot 1} = \underline{0,824 \text{ V/at}}.$$

If the potentiometer is then set to factor k = 0.824, we have:

$$U_a = \frac{Q}{C_g \cdot k} = \frac{82,4 \text{ pC/at}}{100 \text{ pF} \cdot 0,824} = \underline{1 \text{ V/at}}.$$

The great practical advantage of this adjustment is obvious. Instead of the output voltage per mechanical unit (amplifier sensitivity e.g. 1V/at), the range scale shows the reciprocal value i.e., the measuring range per Volt (e.g. 1 at/V). If an oscilloscope is connected to the charge amplifier and set to a sensitivity of 1V cm, the indicated range becomes the diagrammatic scale; in the example given this is $\underline{1 \text{ at/cm}}$.

If required, a free selection of factor k makes it possible to set any desired output voltage and/or any desired range.

The 10-turn potentiometer P can normally be varied between 0 and 1 (setting knob is marked 0-00 to 10-00). For practical reasons (accuracy and stability), it can, however, be used only in the range 1-00 to 10-00. In addition, it is often desirable to extend the range on the upper limit, in order, for example, to span the tolerance deviation with transducers of a nominal sensitivity of 1, 10 or 100 pC/M.U. For this reason, a fixed resistance of 10% is connected in series with the 10 turn potentiometer, the dial of which can then be set to the range 1-00 to 11-00- The whole resistance of the potentiometer (factor k = 1) is then reached, not at position 10-00, but at position 11-00. Since, however, according to calculation, position 11-00 corresponds to an amplification of 10/11 = 0.91, a corresponding attenuation has to be effected elsewhere; in the case of the present amplifier, this attenuation takes place in the plug-in filter.

A great advantage of the capacitive negative feedback is that the amplification factor of the amplifier proper does not enter into the measuring accuracy, provided that it is sufficiently high. The only essential parts are the negative feedback capacitance C_g (range capacitor) and the position of the potentiometer P.

Parallel to the negative feedback capacitor are two high-value resistances, ($10^9\Omega$ and $10^{11}\Omega$, able to be connected by means of a switch) and a short-circuit Reed relay "Reset" contact. A short-circuit on the negative feedback capacitor (range capacitor) causes it to discharge completely and brings the output of the amplifier to 0 Volts (for zero suppression by carrying away a charge corresponding to a preload or for zero setting before a measurement or for adjusting the amplifier). The high-value resistances, on the other hand, discharge the range capacitor slowly, with a time constant of $T = RC$. This avoids drift phenomena and carries away any spurious charges. Due to this reduced time constant, quasistatic measurements are inhibited.

I - 5   Controls and connections

All the operating controls are on the front of the amplifier, the connectors being on the back. The zero adjustment is accessible through a hole in the inner casing of the amplifier.
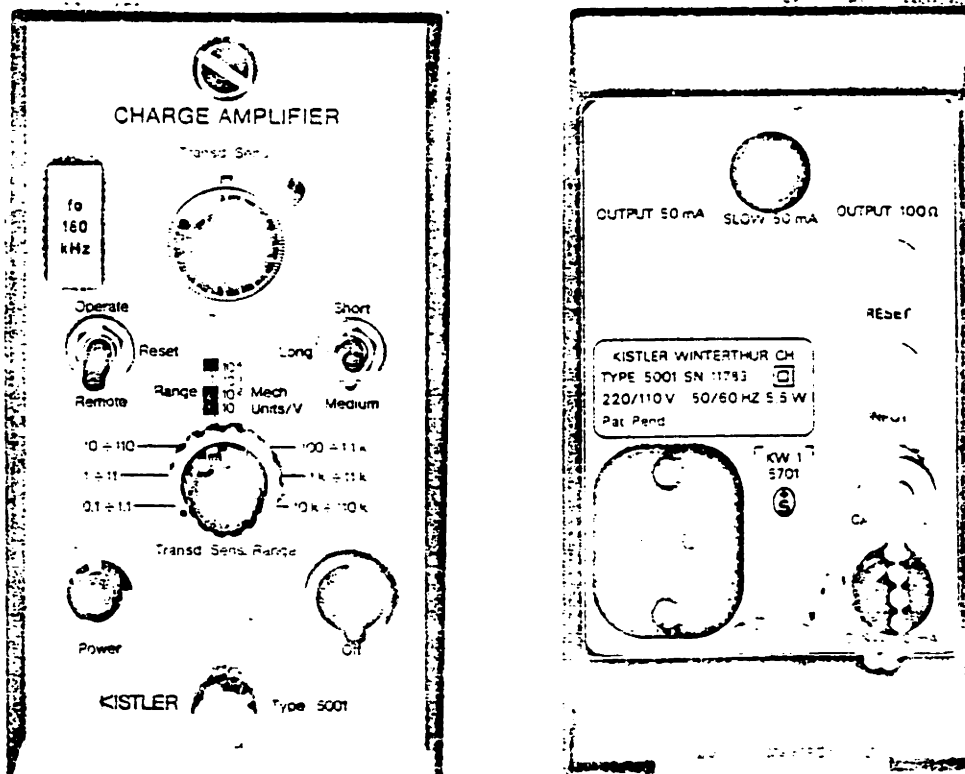
Fig. 4: Arrangement of controls and connections

B 11.5001e   8.70   5r

Operating controls and their function (Table 2)

Designation / Control	Function
Front of amplifier	
Off Toggle switch	In the "Off" position, the amplifier is switched off.
Power Glow lamp	When this lamp glows, the amplifier is switched on.
Transd.Sens.Range (Transducer sensitivity range) Calculating disc.	Calculating disc for automatic range indication; no alteration of amplifier setting. The decade of the transducer sensitivity is dialed on the calculating disc. (The numerical order - regardless of decimal value - is to be set on the 10-turn potentiometer "Transd.Sens."). These procedures ensure that the correct measuring range always appears, without any need for conversion. In the position 0.1 to 1.1, the range indication also corresponds to the relevant range capacitance in pF.
Range Mech. Units/V 12-step switch	Range switch. The use of this switch selects the appropriate range capacitor. The figures shown give mechanical units/Volt. The indicator shows on the other hand the scale (e.g. when using a recording instrument with a sensitivity of 1V/cm: M.U./cm) or, if multiplied by 10, the maximum measuring range of the amplifier (corresponding to 10V max. output voltage.)
Short - Long - Medium (time constant) Toggle switch	Time constant switch: In the "Short" position, the range capacitor is bridged with a $10^9\Omega$ resistance; in the "Medium" position, with a $10^{11}\Omega$ resistance. The smaller time constants thus obtained ensure drift-free measurement of dynamic processes. For quasistatic measurements or calibration, the switch must be set to the "Long" position.
Operate - Reset - Remote Toggle switch	"Grounding" or reset switch. This switch replaces the former grounding pushbutton GND. The "Operate" position is the normal operating position. In the "Reset" position, the amplifier is "grounded", i.e., the range capacitor is short-circuited. If no measurements are being made, the switch should be set to the "Reset" position. In the "Remote" position, the amplifier is controlled by the position of the remote control switch. In the event of power failure, or if the amplifier is switched off, the switch is positioned automatically at "Reset" (principle of normally closed-circuit).
Transd. Sens. 10-turn potentiometer	Wirewound precision-built potentiometer for stepless alteration of negative feedback for adjustment to the transducer sensitivity. The numerical sequence is always set, regardless of the decimal point (e.g., if the value is 82.4 pC/at, the position 8-24 is set). The decimal point is determined by the calculating disc "Transd.Sens.Range" by setting it to the range 10 to 110. (See Fig. 1).

<u>Plug-in filter</u>	Depending on the measuring problem, plug-in low-pass filters of different limit frequencies or special filters can be inserted. The amplifier cannot be used without a filter, as this supplies the connecting component between the charge amplifier and the operational amplifier. When ordered normally, the charge amplifier contains, unless otherwise specified, a low-pass filter with a limit frequency of 180 kHz. . For other filters consult data sheet 12.011.
<u>Back of amplifier</u>	
<u>Power plug, 2-pole,</u> ungrounded	For connection to power line of 220V, 50 - 60 Hz, adaptable to 110V (solder connection). Protection Group II, 4 kV test voltage.
<u>SLOW 50 mA</u> <u>1) power fuse</u> 2) fuse for power output	Both fuses contain a 50 mA slow-blow element. The same power fuse is used for 220 and 110 Volt. In certain circumstances, it may be necessary to use at the output a special insert with a lower resistance.
<u>CAL. INPUT</u> <u>BNC-neg. connector</u>	This calibration input leads via a 1000 pF calibration capacitor to the normal input, and is used for calibration and checking. It can also, however, be used as a voltage input; in this case, the amplifier acts as an impedance converter with capacitive input. In contrast to the charge input (pC), the calibration input is in mV, since 1 mV · 1000 pF = 1 pC. The maximum voltage which may be applied is 30 V.
<u>INPUT</u> <u>BNC-neg. connector</u>	Normal input, to which the quartz transducer is connected (highly insulating).
<u>RESET</u> <u>BNC-neg. connector</u>	Remote control of amplifier reset ("grounding"). The remote control is in operation when the "Operate-Reset-Remote" switch is in the "Remote" position. When the connecting circuit is open, the amplifier is "grounded" (principle of normally closed contact). In order to open the ground contact (for normal operation, i.e., measurement), the remote connection must be short-circuited: open circuit voltage -15V, short-circuit current approx. 10 mA. If required, several amplifiers of this type can be connected in parallel to the remote control and operated by a single switch.
<u>OUTPUT 100Ω</u> <u>BNC-neg. connector</u>	Amplifier output ±10V, protected from short-circuits. To avoid measuring errors caused by voltage drop, the input impedance of any instrument connected to the amplifier must be high. (100 kΩ ≙ 0,1%; 10 kΩ ≙ 1% error.)
<u>OUTPUT 50 mA</u> <u>BNC neg.-connector</u>	Amplifier output, ±10V, maximum load 50 mA, taken out via fuse. Impedance less than 0.5Ω - resistance of fuse (10 to 35Ω, according to make).

B 11.5001e   8.70   6r

I - 6   Typical applications
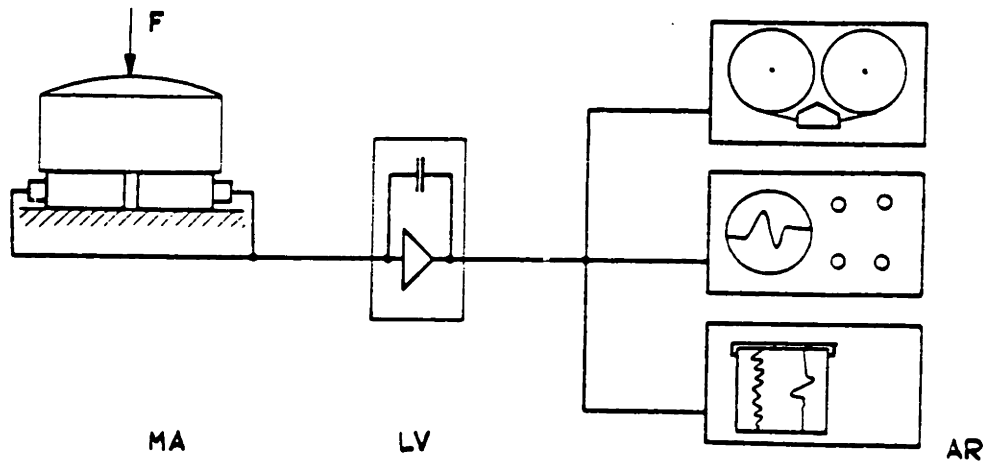
MA = transducer
LV = charge amplifier
AR = indicating or recording equipment

Fig. 5: Typical example of application

A typical application for the measurement of mechanical quantities (at, kp, g) is the combination of the charge amplifier with a suitable quartz transducer and an indicating or recording instrument.

The parallel connection of several transducers gives the sum of the electrical charges. Application: force measurements by parallel connection of several load washers.

B 11.5001e   8.70   7r

In the event of power failure, or when the amplifier is switched off, the range capacitor is automatically short-circuited, providing reliable protection for the input transistor.

If no measurements are made for a long period, it is recommended to set the switch to the "Reset" position. It is recommended to leave the amplifier switched on over the whole period of measurement, in order to attain increased stability.

c) The "Short-Long-Medium" switch

This 3-position toggle switch is used to select the time constant. The long time constant gives virtually static behaviour of the amplifier. Because of erratic currents, however, the time available for measuring is limited normally to the order of minutes, as the erratic currents cause drift (gradual charging of range capacitor), which eventually drive the amplifier into saturation.

Short or medium time constants mean that a high-value bleeder resistor is connected in parallel with the range capacitor. The gradual discharge (time constant $T = R \cdot C$) of the range capacitor prevents its being charged by erratic currents and thus prevents the zero from drifting and the amplifier from saturating. This type of operation is particularly suitable for dynamic measurements, but not for quasi-static measurements or calibration.

The value of the time constant depends on the product of resistance (Short: $10^9 \Omega$, Medium: $10^{11} \Omega$) and capacitance (size of range capacitor), i.e. $T = R \cdot C$.
The lower cut-off frequency is: $fg = 1/(2\pi \cdot T)$.
The size of the range capacitor is determined as follows: the calculating disc "Transd.Sens. Range" is set to the range 0.1 - 1.1 (without moving the range switch), and the range indicated then corresponds to the capacitance of the relevant range capacitor in pF ($1pF = 10^{12}F$).

Example: For a range capacitor of e.g. 50 pF, in the "Short" position ($R = 10^9 \Omega$) the time constant is: $T = R \cdot C = 10^9 \Omega \cdot 50 \cdot 10^{-12} = 50 \cdot 10^{-3}$ s.
From this is calculated a lower cut-off frequency (-3dB) of:
$fg = 1/(2\pi \cdot T) = 159/50 = 3,18$ Hz.
For the same measuring range, the lower cut-off frequency in the "Medium" position ($R = 10^{11} \Omega$) is 0.0318 Hz. (Other values see table 3)

Time constant and lower cut-off frequency (Table 3)

Range												
Cg (pF)	$1 \cdot 10^1$	$2 \cdot 10^1$	$5 \cdot 10^1$	$1 \cdot 10^2$	$2 \cdot 10^2$	$5 \cdot 10^2$	$1 \cdot 10^3$	$2 \cdot 10^3$	$5 \cdot 10^3$	$1 \cdot 10^4$	$2 \cdot 10^4$	$5 \cdot 10^4$
T (s) Short	0,01	0,02	0,05	0,1	0,2	0,5	1	2	5	10	20	50
fu (Hz)	$16 \cdot 10^0$	$3 \cdot 10^0$	$32 \cdot 10^{-1}$	$16 \cdot 10^{-1}$	$3 \cdot 10^{-1}$	$32 \cdot 10^{-2}$	$16 \cdot 10^{-2}$	$3 \cdot 10^{-2}$	$32 \cdot 10^{-3}$	$16 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$32 \cdot 10^{-4}$
T (s) Medium	1	2	5	10	20	50	100	200	500	1000	2000	5000
fu (Hz)	$16 \cdot 10^{-2}$	$3 \cdot 10^{-2}$	$32 \cdot 10^{-3}$	$16 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$32 \cdot 10^{-4}$	$16 \cdot 10^{-4}$	$3 \cdot 10^{-4}$	$32 \cdot 10^{-5}$	$16 \cdot 10^{-5}$	$3 \cdot 10^{-5}$	$32 \cdot 10^{-6}$
Long	on "Long" the amplifier behaviour is not determined by time constant but by drift.											

Cg =   Negative feedback (range) capacitor
T =   Storage time constant of Cg for "Short" or "Medium"
fu =   Corresponding lower cut-off frequency

II   OPERATION

II - 1   Operating philosophy

a) Measuring range and scale

The operating concept with KISTLER is directed to the scale, i.e. the amplifier is set to the sensitivity of the transducer being used and the range switch is set to the required scale (number of mechanical units per Volt of output voltage or per cm deflection on the recording instrument). No conversion of pC into Volts or centimetres is necessary.
Any oscillogram, to be evaluated, requires a scale. This is usually given in M.U./cm, and the diagram is so labelled (e.g. 10 at/cm). If the quantity being measured increases or decreases, the next larger (e.g., 20 at/cm) or the next smaller (e.g. 5 at/cm) scale is selected.
Modern measuring techniques now use only fixed measuring ranges (scales) in the steps 1 : 2 : 5 : ... . The time when the measuring range (scale) which had to be used was an uneven value, merely in order to be able to make full use of the available measuring range of a piece of equipment, is long past. There is no longer time to evaluate diagrams with uneven measuring ranges (scales). The objective is attained by KISTLER as follows: the charge amplifier is set to the sensitivity of the transducer in use (numerical sequence and decimal point), in accordance with the calibration sheet provided, e.g. 79.8 pC/at. When this has been done, the range selector switch can be set to the required measuring range (scale), calibrated in mechanical units per Volt (e.g. 10 at/V). If for indication or recording purposes an oscilloscope or recording equipment with a scale of 1 V/cm is used, the scale obtained is, for example, 10 at/cm.

Advantage:	indication on range switch:	10 at/V
	gives on oscilloscope or recording equipment	10 at/cm

If a higher sensitivity (lower scale), e.g. 0,2V/cm has to be selected at the indicating or recording equipment (e.g. when the lowest range of the amplifier is set, but not used to full scale) the overall scale is obtained by multipiying the amplifier scale with the recording scale.

Example:	Charge amplifier:	10^{-1} at/V	= 0,1 at/V
	Oscilloscope:		0,2 V/cm
	Overall-scale:	0,1 at/V x 0,2 V/cm	= 0,02 at/cm

b) The "Operate-Reset-Remote" switch

This 3-position toggle switch replaces the earlier "grounding" push button GND. When it is operated, the range capacitor is short-circuited and so fully discharged. In the new Charge Amplifier 5001 this is done by a high-insulation Reed relay, with a normally closed contact.

In the position "Operate", the relay is energized, which brakes the short circuit.

In the position "Reset" the relay draws no current, and the relay contact short-circuits the range capacitor; this brings the amplifier output voltage to 0.

In the position "Remote", it is the setting of the remote control switch that determines the setting of the amplifier (remote control switch open = Reset, i.e., the amplifier is "grounded"; the output voltage is 0. Remote control switch short-circuited = Operate).

The remote control terminals of any number of amplifiers can be connected in parallel. Each single amplifier can then be set locally to "Operate" or "Reset", independently of the position of the remote control switch.

II - 2   Operating instructions

a) General directions

The amplifier is ready for operation as soon as it is switched on. If greater stability is required for quasi-static measurements, some time must be allowed for the apparatus to warm up (20 - 60 min.). It is not necessary to ground the apparatus electrically. Construction to Protection Group II standards (4 kV test Voltage) means that there is no danger to the operators. The amplifier as delivered ex works is set for a line voltage of 220V AC. Adaptation to 110 V AC is made by removing the left-hand cover and resoldering as described on the diagram fixed to it. (Warning: before opening the instrument, remove the power line plug!)

Connection of the transducer to the amplifier must be by means of special cables only, i.e., high-insulation coaxial cable, which do not give off electrostatic charges when moved. See data sheet 15.011. Normal connecting cables may be used at the amplifier output, the remote control connection and the calibration input.

Polarity: KISTLER transducers usually supply a negative signal: pressure transducers for pressure increase, force transducers for force increase, force-measuring elements for pressure, and accelerometers for positive acceleration (mounting surfaces are stressed for pressure).

Negative charge signals are converted by the charge amplifier into positive output voltage.

b) Connection of high impedance instruments

All high impedance instruments, e.g., oscilloscopes, special amplifiers, digital voltmeters, magnetic tape equipment, etc., must be connected to the "OUTPUT 100Ω". Because of the low load, there is no risk of signal reduction, and the scale corresponds to the selected range. For a load resistance of 100 kΩ, the signal attenuation is 0.1%. The "OUTPUT 100Ω" is long-term short-circuit-proof.

c) Connection of low-impedance instruments

Low-resistance apparatus, e.g., galvanometers of UV-recorders must be connected to the "OUTPUT 50 mA". This output is protected by a 50 mA slow-blow fuse. The output impedance is composed of the internal resistance of less than 0.5Ω and the resistance of the fuse (10 to 35Ω, depending on the make). In order to protect the galvanometer (subsequently referred to as G) from overloading and consequent damage, a sufficiently large series resistance must be provided to ensure that at the maximum possible output voltage (10 to 12V, depending on the load) the current is limited to the maximum permissible for the G in use. Details on adjustment of the G (determination of series and/or parallel resistances) are given fully in the operating instructions for the relevant recording equipment. Some special information on calculations with particular reference to the specifications of the present amplifier, is attached.
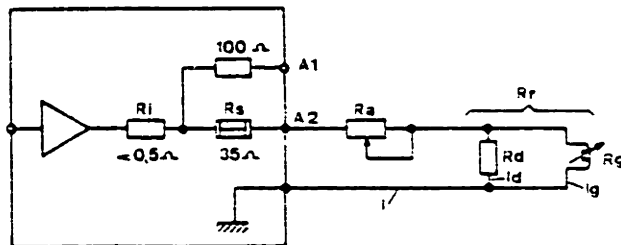
Fig. 6: Connection of a recording galvanometer (schematic diagram)
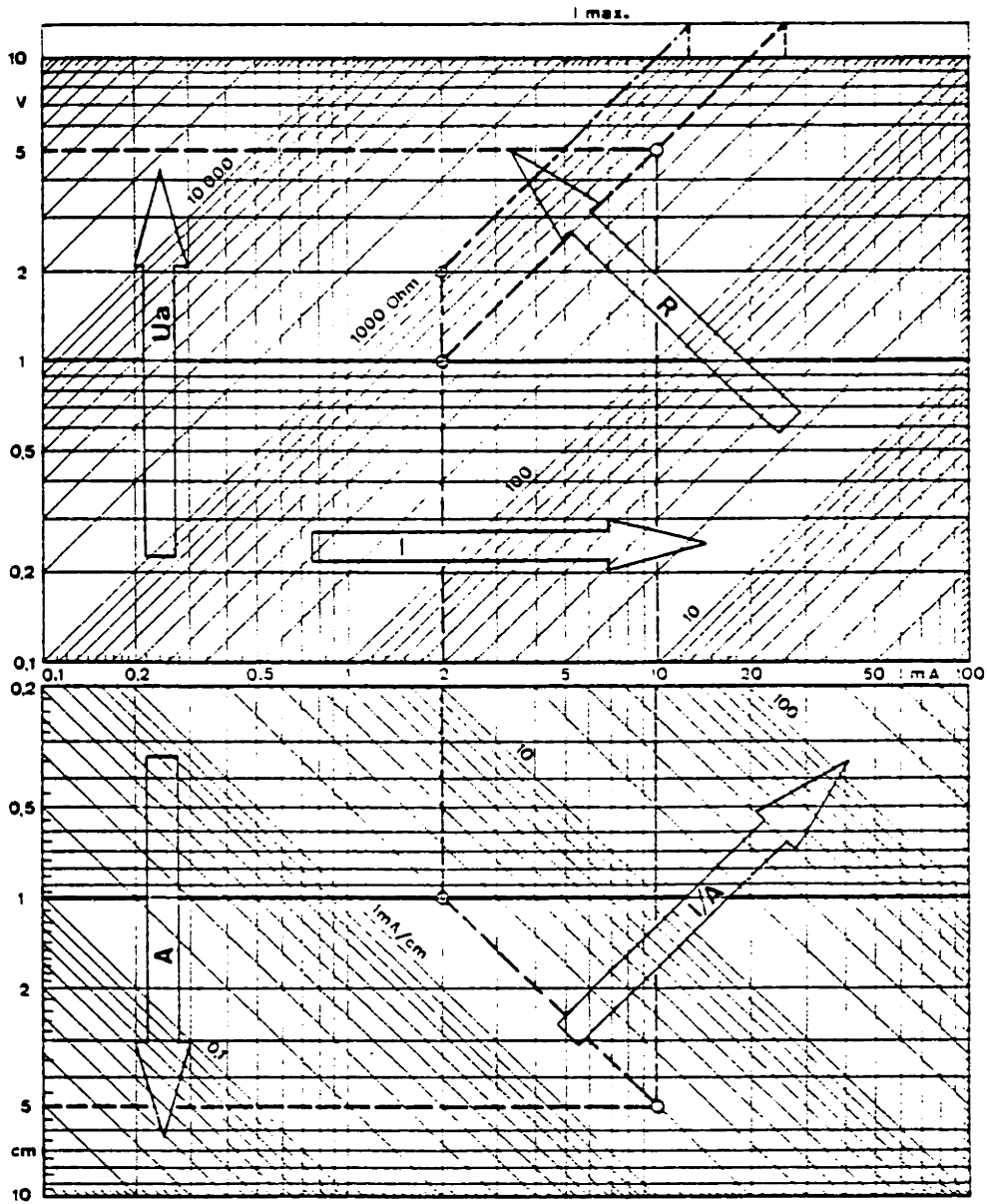
A	=	Galvanometer deflection (cm)
I/A	=	Current sensitivity of the galvanometer with its parallel damping resistor Rd (mA/cm)
I	=	Output current = Ig + Id (mA)
R	=	Total resistance (Ω)
Ua	=	Amplifier output voltage (V)

Fig. 7: Nomogram for galvanometer matching

Matching and operation of recording galvanometers are highly simplified by the use of the Galvo-Amplifier Type 5211. This amplifier features an adjustable over-load protection and a calibration signal.

High-frequency Gs require much current, they are fluid damped (no Rd). Low-frequency Gs, on the other hand, require very little current for deflection; they are damped electro-magnetically (by the damping resistance Rd).

The nomogram in Fig. 7 shows the relationships between the following quantities:

A   -   desired galvanometer deflection (cm)
1/A   -   galvanometer sensitivity (current factor), (mA/cm, incl. Rd).
Ig   -   galvanometer current for the desired deflection (mA) (incl. Rd)
R   -   total resistance of the load circuit (Ω)
Ua   -   amplifier output voltage (V)

In accordance with the KISTLER operating philosophy, it is recommended to undertake adjustment of the G in such a way that 1V of the amplifier output voltage causes 1 cm deflection on the recording instrument. The advantage of this adjustment is that the range indication on the amplifier (e.g. 20 at/V) then also represents the scale of the oscillogram, in the present case 20 at/cm (20 at/V · 1 V/cm = 20 at/cm).

This adjustment is carried out according to the nomogram as follows:
On the line "A = 1 cm", find the intersection point with the G-sensitivity line "1/A: e.g. 2 mA/cm". From this point, draw a perpendicular upwards until it cuts the line "Ua = 1V". This intersection gives the total resistance of the load circuit required for matching (e.g. 500Ω). Trace the line "R = 500Ω" until it intersects the line "I max." (at about 12V): this shows the maximum current that can flow when R = 500Ω if the amplifier is "saturated" (in the present example, 24 mA).
If this value is greater than the maximum current permissible for the G in use, then, if the G is to be reliably protected, the total resistance R must be doubled (in the example, to 1000Ω), and this naturally also doubles the scale (2V/cm = range set on the amplifier x 2). Intermediate values, e.g., increasing R by 50%, are not recommended.

For the chosen example, it also appears from the nomogram that if the recording equipment is used for deflections up to 5 cm, the resulting current is 10 mA and the amplifier output voltage rises to 5V.

Accuracy: The nomogram is not for the purpose of giving the matching resistance within a few per cent, but for providing a general view of the possibilities of matching. Exact adjustment (e.g., on the resistance Ra) can, for example, be made as follows:
Given the following amplifier setting:

Transd.Sens.	Pos. 10-00
Transd.Sens.Range	0,1 - 1,1
Range	$1 \cdot 10^3$ Mech.Units/V

1 cm deflection must be produced for 1 Volt at the calibration input. The adjustment (calibration) is made by altering Ra.
The connection of a highly sensitive G is given in the example below:

Sensitivity of G alone	1/A	=	0.1 mA/cm
G resistance	Rg	=	120 Ω
Required damping resistance	Rd	=	250 Ω
Desired deflection	A	=	10 cm

For full deflection, 1 mA will be required (0.1 mA/cm · 10 cm).

Highly sensitive galvanometers require a damping resistance Rd, through which flows part (Id) of the current from the amplifier.

$$\frac{\text{(Damping current)}}{\text{(Galvanometer current)}} \quad \frac{Id}{Ig} = \frac{Rg}{Rd} \quad \frac{\text{(Galvanometer resistance)}}{\text{(Damping resistance)}}$$

The total current required at the amplifier output is thus:

$$I = Ig + Id = Ig \; \frac{Ig \cdot Rg}{Rd}$$

In the present example:

$$I = 1 \text{ mA} + \frac{1 \text{ mA} \cdot 120 \, \Omega}{250 \, \Omega}$$

$$= 1 \text{ mA} + 0{,}48 \text{ mA} = 1{,}48 \text{ mA}$$

The current required for the full deflection of 10 cm is thus not 1 mA, but 1.48 mA, i.e. the sensitivity of G with its parallel damping resistor is 0,148 mA/cm instead of 0,1 mA/cm for G alone.
If the current of 1.48 mA should flow at 10 Volts, the total resistance must be

$$R = \frac{10'000 \text{ mV}}{1.48 \text{ mA}} = 6755 \, \Omega$$

Rd + Rg give about 80 Ω, Rs + Ri about 35 Ω. Thus, Ra (external series resistor) must be adjusted to 6640 Ω.

As the value of Ra (6640 Ω), as seen from the galvanometer, is connected parallel to the damping resistance, this is therefore slightly reduced, being no longer 250 Ω but 240 Ω, which results only in greater damping and can almost always be neglected.

At 12 V (voltage limitation) the total flow of current is only about 2 mA (1.3 mA through the G). This value lies below the usual safety limit, i.e., there is no danger that the G will be damaged by excessive current.

d) Filters

The 5001 amplifier is delivered with a plug-in low-pass filter (LC filter, cut-off frequency 180 kHz, 12 dB/oct), unless otherwise ordered. The filter is connected between the charge amplifier and the operational amplifier. In place of the standard filter, LC filters with different cut-off frequencies or special filters may be used. Special filters, for example, may be tuned to a particular transducer, in order to maintain the flattest possible frequency response and to suppress the resonant frequency of the transducer. Special filters can also be constructed as active filters.
For available filters consult data sheet 12.011.

Input filters (RC filters), in contrast to the plug-in (output) filters are not mainly for the purpose of damping high-frequency noise signals, but to prevent short charge peaks (caused, for example, by shock loads) from saturating the amplifier.
See data sheet 12.543.

II - 3   Precautions during operation

WARNING

The amplifier INPUT must not be short-circuited nor connected to a low-
impedance voltage source. In the event of short-circuit, the amplifier
immediately saturates. Calibration voltages may be applied to the INPUT
only via a high-insulation capacitor or directly to the CAL.INPUT. If
the CAL.INPUT is not used, the attached cover should be fitted.

While work is being done on the installation (connecting transducers and
cables), and during long pauses in measurement, the amplifier should be
switched to "Reset", either, in the case of single amplifiers, by means
of the "Operate-Reset-Remote" switch on the front panel or, in the case
of remote control of several amplifiers, (switch position "Remote"), by
setting the common "Operate-Reset" switch to "Reset".

Before being connected to the amplifier, input cables should be dis-
charged. (Only use special cables, see section II-2a). This is simply
done by short-circuiting the central pin to the shield with a screwdriver
or similar tool. This prevents any high electrostatic charges on the cable
from producing voltage peaks, which can cause damage to the Dual-
MOSFET input transistor.

All connections to the input side (connector on the transducer, to the connecting and ex-
tension cables and to the amplifier) must be protected from dirt and must, if necessary, be
cleaned with Freon TF or pure gasoline. It is strongly recommended to cover the connections
with protective caps after use.

If ground-loops occur, the transducers should, if possible, be built insulated into the object
to be measured, or insulating couplings Type 1739 and separate ground lines should be used.

Overloading: Overloading by an excessive charge signal does not normally damage the
amplifier. At a tenfold overload, the amplifier recovers after 50 µs. At very high overloads,
the charge can give rise to a considerable voltage, as it can no longer flow into the charge
amplifier when it is saturated. The level of voltage is dependent on the charge and the in-
put capacitance (transducer capacitance and cable capacitance).

$$U = Q/C \quad (Volt = Coulomb/Farad)$$

Example: A load washer 903A (40 pC/kp), loaded with 6000 kp, gives a charge of
240'000 pC.
The total capacitance with short connecting cable on the most sensitive
range is, for example, 240 pF. In this example, the voltage produced is
1000 Volts, and this would destroy the costly input transistor.

In order to avoid short overloads due to charge peaks (caused, for example, by shock or
frame noise in the case of accelerometers), the use of an input filter (RC filter) is recommen-
ded.

The time constant switch should normally (for measurement of dynamic phenomena) be set to
"Medium" or "Short". This prevents the amplifier from drifting to saturation. The time constant
switch should be set to "Long" only for quasi-static measurements or calibration.

B 11.5001e   8.70   10r

III   CALIBRATION AND MAINTENANCE
═══════════════════════════════════════

III - 1   Calibration and adjustment

The calibration of the amplifier is dependent only on the negative feedback capacitor (range capacitor) in the charge amplifier and on the negative feedback resistance (10 turn potentiometer "Transd.Sens.") in the operational amplifier. Because of the negative feedback and the high open-loop gain, the amplification factor has virtually no effect on the measuring accuracy.
A simple method of control is given by the built-in calibration capacitor (1000 pF $\pm$ 0.5%), at the CAL.INPUT. Voltages applied to this input are converted via the capacitor into charges. The charge obtained is 1 pC per mV (1mV · 1000 pF = 1 pC). At the maximum voltage of 30 V a maximum charge of 30'000 pC is obtained.
In order to check the calibration of the amplifier, a voltage is applied to the CAL.INPUT of the same number of mV as would be given as charge (pC) by any transducer for a specific mechanical load. It can then be checked whether the output voltage corresponds to the simulated mechanical quantity.

Example: Assuming that the pressure transducer 701A, with a sensitivity of 81.6 pC/at, is being used, and that the amplifier is set for the measurements of 100 at (10 at/V), the pressure transducer would thus give a charge of 8160 pC for 100 at. To check the amplifier, 8160 mV (8.16V) are applied to the CAL.INPUT. This calibration voltage should give 10 V output voltage. A positive calibration voltage gives a negative output voltage and vice-versa.

With the amplifier set as follows: 10 turn potentiometer "Transd.Sens.Range" position 0.1 - 1.1, "Range" switch set to 1000 M.U./V, the amplification obtained is -1, i.e., -1V at the CAL.INPUT appears as -1V at the OUTPUT.

Calibration of a whole measuring installation. This calibration is in general unnecessary, since all KISTLER devices have calibrated sensitivity.
In the case of extensive and complicated measuring installations, it is recommended to carry out a charge calibration. This consists of producing an electrical charge in the charge calibrator, the said charge be exactly the same as that which would be given by the transducer used in the installation for a specific mechanical load, e.g., corresponding to the whole measuring range.
This calibrates the whole measuring channel from the input of the charge amplifier right through to the recording equipment or data-processing or evaluation equipment. Only the transducer is not included in this calibration. The advantages of this charge calibration are: its low cost (as compared with a mechanical calibration installation), greater measuring accuracy (only the errors of the charge calibrator and the transducer enter into the calibration), and the simultaneous control of function, which ensures that all the operating controls of the linked devices are correctly set.

For extremely high measuring accuracy, mechanical calibration must be used and the relatively higher cost must be accepted. For this calibration, the transducer must be removed from the object to be measured and installed in the calibration equipment (e.g. dead weight tester). In the case of this type of calibration, the output voltage or the measuring record is compared directly with the calibration quantity (e.g., pressure on the dead weight tester).

Zero adjustment: The zero adjustment is the only adjustment that can be made from outside the amplifier. Adjustments requiring removal of the inner casing of the amplifier should be carried out only by the makers. A drift adjustment is not provided, as the amplifier is fully transistorized, and drift is caused only by fault currents, which cannot normally be compensated.

Parts List (Table 4)

Designation	Part No.	Description	Quantity
-	Type 5001	Charge Amplifier complete	-
-	Type 5711	Case	1
-	3.511.019	Fastening screw	2
-	5.111.004	Rubber ring for screw	2
-	Type 1505	Power cord	1
-	Type 1511	Output cable	1
7.611.001	7.611.001	Printed circuit board including components: "power supply"	1
7.611.009	7.611.009	Printed circuit board including components: "amplifier"	1
-	5.811.019	MOSFET input transistor	1
7.611.997	7.611.997	Printed circuit board including components: "Reed relay"	1
Type 5311	Type 5311	Standard filter 180 kHz	1
C1-C12	-	Range capacitors 10 - 50'000 pF	1 each
C13	3.630.017	Calibration capacitor, 1000 pF $\pm$ 0.5%	1
C14	5.616.029	Capacitor, 100 pF	1
D1	5.810.019	Diode 1N916	1
F1	5.321.003	Fuse, power supply, 50 mA slow-blow	1
F2	5.321.003	Fuse, OUTPUT 50 mA slow-blow	1
-	5.320.001	Fuse holder	2
L1	5.340.002	Neon lamp, "Power", complete	1
N1	5.510.005	Power receptacle	1
N2-N6	5.550.019	BNC chassis connector, neg.	5
-	Type 1853	BNC shielding cover with chain	1
P1	5.712.010	10-turn potentiometer 5 kΩ	1
-	5.313.004	10-turn dial	1
R1	5.721.083	Resistor, 33 kΩ	1
R2	5.721.063	Resistor, 320 Ω	1
R3	5.721.061	Resistor, 560 Ω	1
R4	5.739.007	Resistor, 10^{11} Ω	1
R5	5.739.006	Resistor, 10^{7} Ω	1
R6	5.721.089	Resistor, 100 kΩ	1
R7	5.721.001	Resistor, 5,6 Ω	1
S1	5.330.006	Power switch	1
S2	3.331.007	Range switch, without components	1
-	5.313.014	Knob for S2, black	1
S3	7.630.009	Time constant switch	1
S4	5.330.007	Reset switch	1
T1	3.630.002	Power transformer	1

A zero error causes a drift that is greater,the poorer is the insulation resistance of the trans-
ducer and the input cable. The zero error is measured in the "Reset" position.

If the zero error exceeds the value given in the Technical Data, adjustment must be made.
The ZERO potentiometer is accessible via a hole in the inner casing of the amplifier, on the
right.

Adjustment is made by connecting a sensitive voltmeter or an oscilloscope to the OUTPUT
(1mV should still be readable). The amplifier should be set to "Reset" and the 10-turn poten-
tiometer to Pos. 1-00. The position of the "Range" switch is unimportant. Zero is then adjusted
with a small screwdriver.

III - 2   Maintenance and replacement of components

No periodical maintenance is necessary. Inspections are limited to examination for loose
connections, the cleaning of dirty high-insulation Teflon insulators, especially in the connec-
tors, and the replacement of fuses.

It is recommended to limit repairs to the replacement of printed circuits. Other repairs should
be entrusted to the makers, who have trained expert staff and the appropriate laboratory in-
stallations.

IV   DIAGRAM, PARTS LIST

The diagram contains the most important main components: amplifier-relay- and power supply
printed board, power transformer with details for converting the 220 Volt supply to 110 Volts,
all the operating controls and the connections as well as the plug-in low-pass filter and the
wiring diagram.
The parts list contains only the components given in the circuit diagram (Fig. 8) plus case,
cables and a few other spare parts.
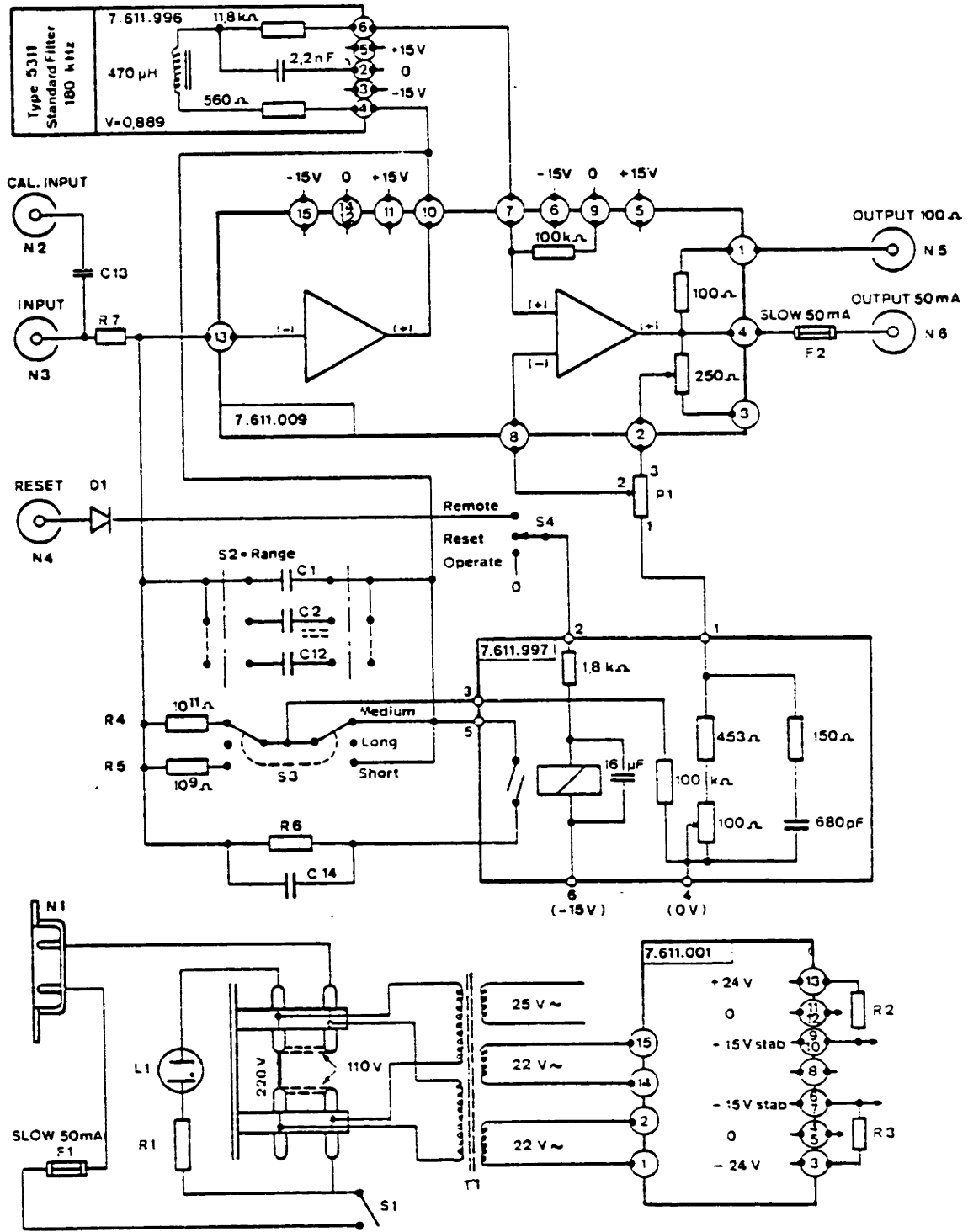
Fig. 3: Circuit diagram, Charge Amplifier 5001

APPENDIX B

FORCEPLATFORM ANALOG TO DIGITAL
CONVERSION SYSTEM AND INTERFACE

     The forceplate (FP) interface utilizes a remote analog
to digital (A/D) converter and a Direct Memory Access (DMA)
digital interface to the PDP 11/60 computer. Since the FP
only transfers 8 words of data per frame and the Selspot
transfers up to 120, the FP has a handshake to hold the
digital data on the lines until the computer has a chance to
read it in. The Selspot is a free running device with no
handshake, thus it always has first priority on data word
transfer (priority BR7) and the FP has a slightly lower
priority (BR6).   This means that occasionally the FP
handshake will be used to hold the FP data valid until the
11/60 has serviced the current Selspot data word transfer.

     A feature of the FP charge amplifiers provides for
remote RESET/OPERATE control.   When the amps are RESET,
their inputs are gounded to reset the output to zero at the
present forceplate loading.   When the amps are in OPERATE
mode, changes in the piezoelectric charge at the plate are
reflected as changes in voltage at the outputs from the
amplifiers. The FP interface makes use of the remote
RESET/OPERATE feature through a programmable bit in the
Command and Status Register (CSR).   In fact there are 3
different programmable "FP enable" lines. When any one of
them is high (1) the FP is enabled and put into OPERATE
mode.   if all of the FP enable lines are low (0), the FP is
returned to RESET mode. One enable comes from the Selspot
interface, which enables the FP when ever Selspot data is
being taken. The other 2 enables come from the user
programmable bits and lines on the FP DMA interface itself
(FNCT2 and FNCT3). FNCT2 enables FNCT1 to strobe the FP to
start frames. This is a "programmed I/O" mode of operation.
FNCT3 enables the KW11-K clock in the 11/60 to strobe the FP
frame start every time the clock overflows. A final feature
of the enale logic: if either of the external enables
(FNCT2 of FNCT3) are on (1) (enabled), the Selspot strobe of
the FP frame start is disabled. This allows independent use
of the Selspot and the Forceplate.

MEMO
To: File
From: Erik Antonsson
RE: Kistler Force Plate - PDP-11/60 DMA Interface.

      Application Notes for the ANALOG DEVICES DAS1128 used
in the KISTLER FORCE PLATE interfaced to the PDP 11/60
through an MDB DR-11B.

      A frame of FP (Force Plate) data is initiated by three
different things. If EXT ENB (FNCT1) is high (1) a positive
pulse on EXT FS1 (External Frame Start 1) or a negative
pulse on EXT FS2 (External Frame Start 2) will start a
frame. FNCT1 is a user bit in the CSR of the DR-11B.   EXT
FS2 is connected to the overflow output tab on the KW-11K.
A short negative pulse is generated at each CLOCK A
overflow.   With EXT ENB high, the third input (from the
Selspot) is disabled. If EXT ENB (FNCT1) is low and SS ENB
is high (from the Selspot interface) then SSFS (Selspot
Frame Start) will initate a frame. Care must be exercised
if both SSENB and EXT ENB are high. Then any one of the 3
frame pulses will start a frame of conversions.

      The SS ENB signal is connected to the ON sampling
indicator on the Selspot MDB-11B. It is high while the SS
DMA is active. The SSFS signal comes from a 74161 modulo 16
preset counter used to count the number of LED to be sampled
by the SS. The next to most significant bit is used as SSFS
to provide one positive pulse per SS frame sampled. Because
of the origin of this signal it does not occur in the same
place in the SS frame for different number of channels. The
positive transition occurs mid-frame for 1,15,or 30 channels
sampled.   It succesively moves towards the end of the frame
for the intervening numbers of channels.

      A frame of data consists of 8 conversions of the 8
differential analog inputs. It is started as above, and
ended when the multiplexer counts to 7. The count is then
reset to 0 and the converter system is ready for another
frame initate pulse. A flip-flop output is used to drive
the TRIG signal as an ON-OFF control. It is started by the
frame initate pulse, and cleared by the count reaching 7 AND
the conversion beginning for that channel (not-EOC). This
TRIG signal is interrupted any time the BUSYO signal from
the DR-11B is high. BUSYO indicates that a request has been
made to the CPU to read the data on the lines and is held
high until the memory cycle to read the data is complete.
With TRIG interrupted this way the data is sure to remain

valid until the computer indicates that it has actually completed the read. Not-EOC (End of Conversion) is delayed by 1 micro second and used as a data available (CYCLE REQUEST A) signal to the DR-11B. The delay is used to insure that the data has settled before requesting the DMA read.

The DAS is set up for 8 double ended differential analog channels (11B,12B),(17B,18T,+5v), (The DAS 1128 pin numbers enclosed in parentheses and separated by commas are wired together externally) input with a range of +/-10.24 volts (2T,12T),(14T,15T),(13T,14B), (15B,16B) discretized into 12 bits (28T,35B). This results in 5 mV per bit. It is wired for 2.08 micro-seconds per bit conversion (26B,GND) and 20 micro seconds delay (23B,GND). It is sequential channel free running with TRIG used as a RUN/STOP control (24B,25B,+5v),(24T,27B),(23T,27T). The output digital coding is 2's complement (17T,-15v),(28B is the MSB).

The MDB DR-11B DMA interface is at address 772430 and vector 130 with an interupt priority of BR6. The BUS ADDRESS 32k boundry overflow increments the Extended Address Bits (29H-J and 30H-J both cut and 29J-K and 30J-K installed). Another NPR request can take control during an interrupt latency (31H-J is cut and 31J-K is installed). The DMA is in DATO configuration (C1 CONTROL=1, C0 CONTROL=0) with word addressing (A001=0), word count increment (WC INC ENB=1), and bus address increment (BA INC ENB=1). It also releases the Bus after each memory transfer cycle to allow other devices to interleave bus cycles (SINGLE CYCLE=1). DSTATA, DSTATB, and DSTATC are all grounded at the Force Plate.

NOTE that NO LOCK and CYC REQ B are both grounded in the P2 connector (P2-28,P2,30),(P2-37,P2-38). NO LOCK MUST be grounded for the DR-11B to operate at all.

NOTE: In all of the schematics the connector pin numbers are circled and the IC pin numbers are not. Connector numbers: n/50 are on the 50 pin connector at the forceplate end. Connector numbers terminated in B or T are on the DAS1128 itself. N.B. The 50 pin connector on the card with the DAS has its even and odd rows INVERTED because of the interconnection to the BACK of the 50 pin connector through the front panel of the force plate electronics box.

FORCEPLATE INTERFACE ACTIVE CSR BITS

15	ERROR	(RO)	
14	NEX	(R/W)	Non Existent Memory
09	DSTATC	(RO)	DAS Power On
07	READY	(RO)	Done filling last buffer, ready for next
06	IE	(R/W)	Interrupt Enable
05	XBA17	(R/W)	Extended bus address
04	XBA16	(R/W)	Extended bus address
03	FNCT3	(R/W)	FP Ext. Enable #2 (Clock)
02	FNCT2	(R/W)	FP External Enable #1
01	FNCT1	(R/W)	FP Ext. Frame Start #1
00	GO	(WO)	Start filling buffer
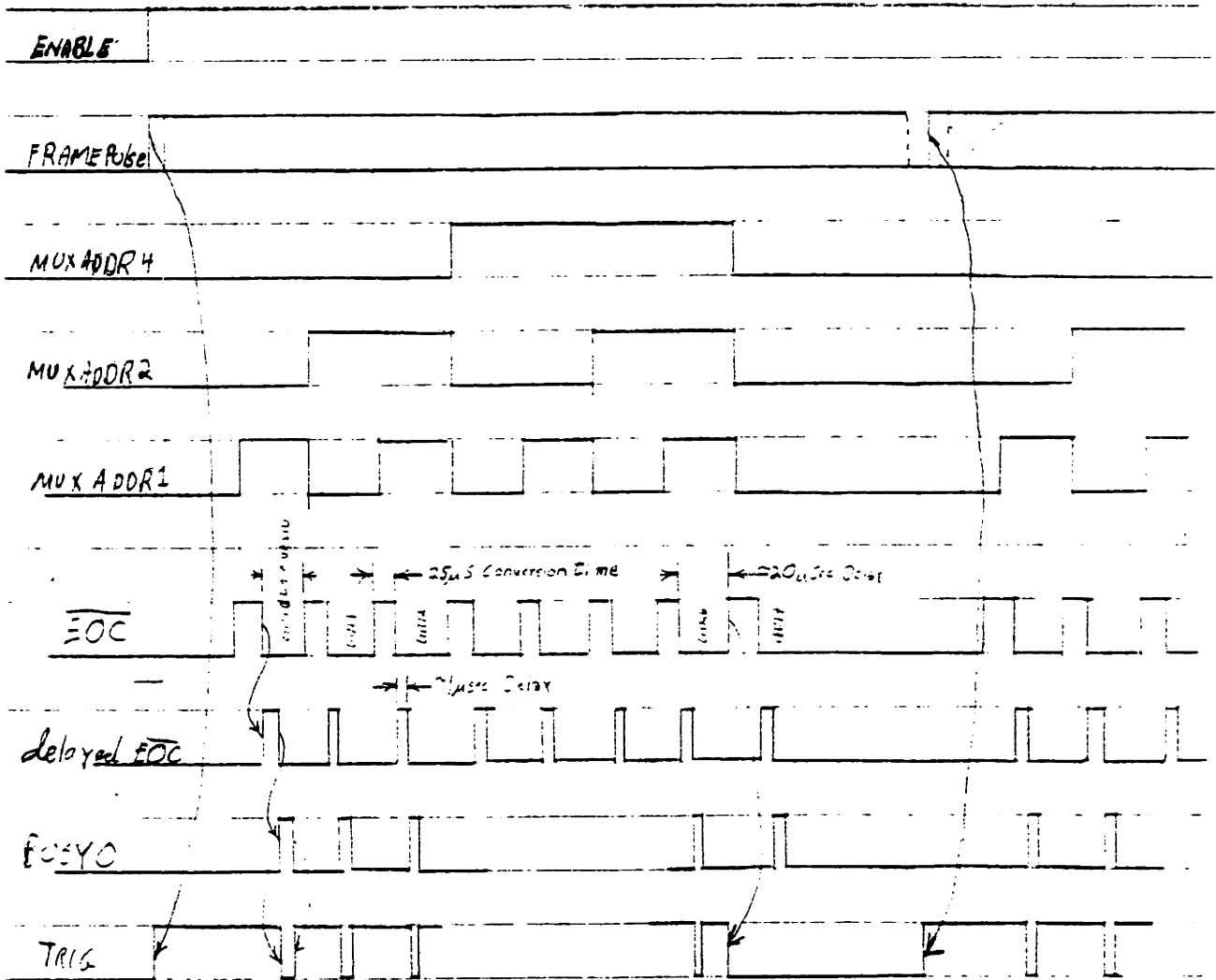
The falling edge of delayed EOC starts the 11/60 DMA transfer cycle throught the MDB DR11-B.
Data is valid while EOC is low.
Forceplate is RESET (not enabled) with ENB low, it is in OPERATE (enabled) mode with ENB high.
BUSY0 indicates a DMA cycle is in progress.
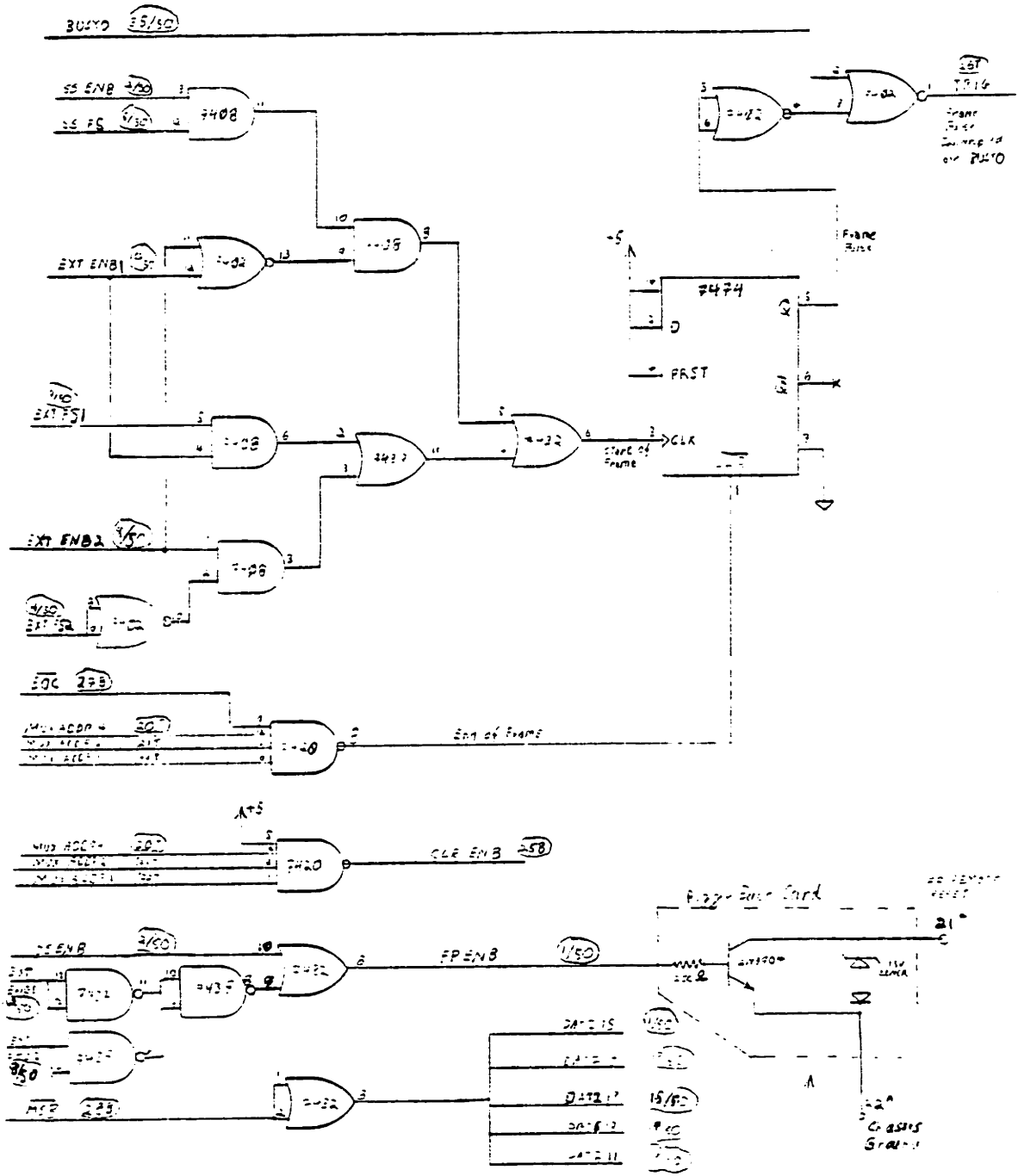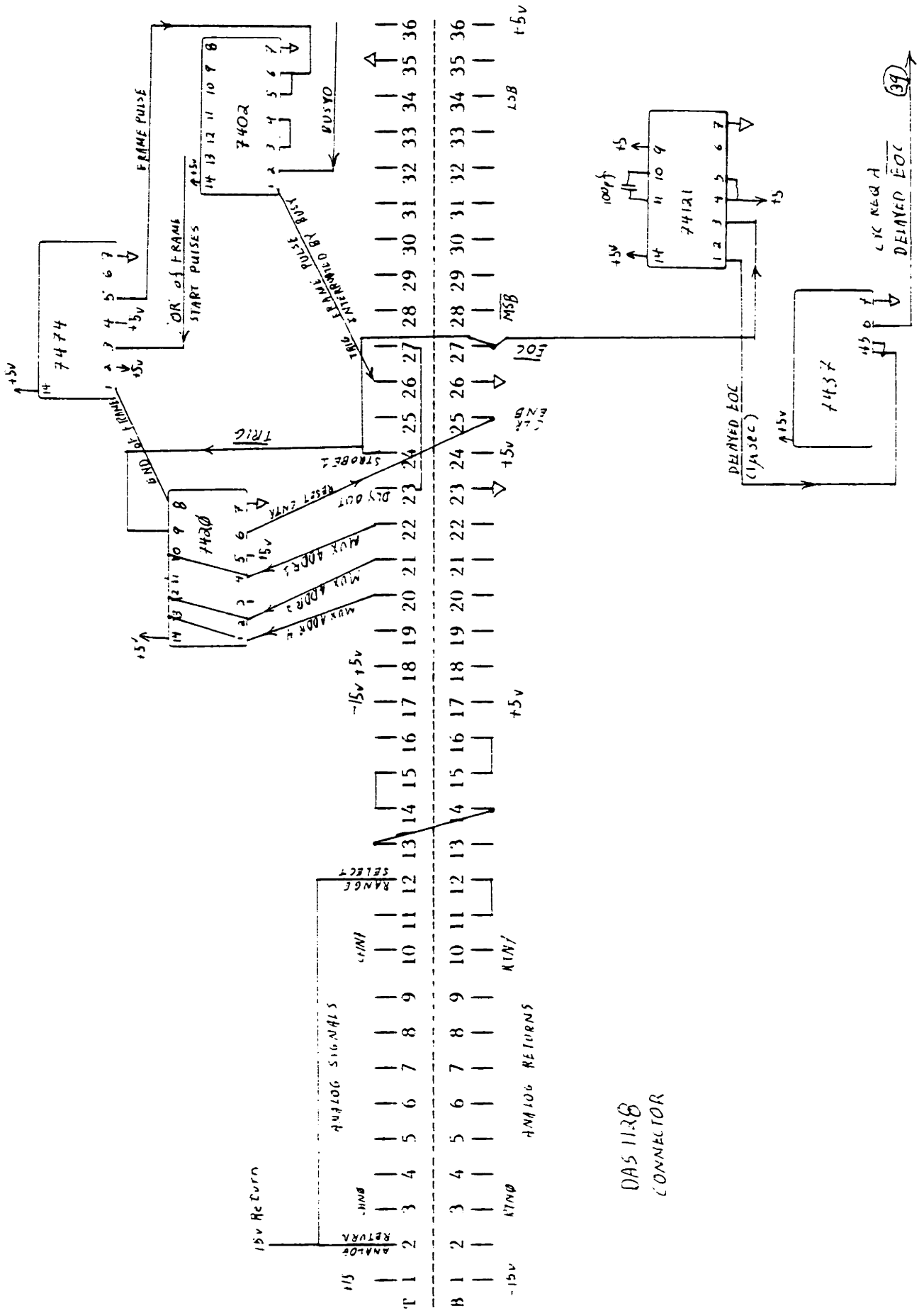
DAS 1128
CONNECTOR

FP SIGNAL NAME	DHS SIGNAL NAME	50 pin #	DMA INTERFACE SIGNAL NAME	INTERFACE PIN #
GND		1	To Front Panel Only (Cut Wire)	X
External Enable (SS)		2		8
GND		3		10
External Frame Start #2		4	KW11-K Clock A Overflow (Fast-on Tab)	X
GND		5	GND	14
SS Frame Start		6		16
GND		7	GND	18
Ext. Enable #2 (clock)		8	FNCT3	35
External Frame Start #1		9	FNCT1	27
External Enable #1 (FP)		10	FNCT2	31
$\overline{MSB}$	28B	11	DATAI15	1
LSB	34B	12	DATAI00	2
$\overline{MSB}$	28B	13	DATAI14	3
Bit 11	34T	14	DATAI01	4
$\overline{MSB}$	28B	15	DATAI13	5
Bit 10	33B	16	DATAI02	6
$\overline{MSB}$	28B	17	DATAI12	7
Bit 9	33T	18	DATAI03	8
$\overline{MSB}$	28B	19	DATAI11	9
Bit 8	32B	20	DATAI04	10
Bit 2	29B	21	DATAI10	11
Bit 7	32T	22	DATAI05	12
Bit 3	30T	23	DATAI09	13
Bit 6	31B	24	DATAI06	14
Bit 4	30B	25	DATAI08	15
Bit 5	31T	26	DATAI07	16
GND*		27	C0 CONTROL IN (to GND on DMA)	17
GND*		28	A001 IN (to GND on DMA)	18
Panel switch to +5/Gnd		29	ATTN	19
GND		30	GND	20
N/C*		31	BA INC ENB (to +5v on DMA)	21
GND		32	GND	22
GND		33	N/C	23
GND		34	GND	24
BUSY		35	BUSY0	25
GND		36	GND	26
N/C*		37	C1 CONTROL IN (to +5v on DMA)	27
GND		38	GND	28
Power ON indicator		39	DSTATC	29
GND		40	GND	30
GND		41	DSTATB	31
GND		42	GND	32
N/C*		43	SINGLE CYCLE (to +5v on DMA)	33
GND		44	GND	34
GND		45	DSTATA	35
GND		46	GND	36
N/C		47	G00	37
GND		48	GND	38
Delayed $\overline{EOC}$		49	CYCLE REQ A	39
GND		50	GND	40

*Means that wire is cut at the DMA interface in the 50 conductor cable

CARD CAGE BACKPLANE FOR A/D AT THE FORCEPLATE

Signal Name	Card Pin #	DAS 1128 Pin #
FP Remote Reset	21	
FP Remote Reset Ground	22	
(FP Chassis Ground)		
5v Ground (Digital Return)	A	35B
(Computer Ground)		
-15v	C	1B
15v Ground	19	2T/2B
(Analog Return)		
+15v	X	1T
+5v	Z	36T/36B
Analog in channel:		
0 Signal	11	3T
0 Return	M	3B
1 Signal	10	4T
1 Return	L	4B
2 Signal	9	5T
2 Return	K	5B
3 Signal	8	6T
3 Return	J	6B
4 Signal	7	7T
4 Return	H	7B
5 Signal	6	8T
5 Return	F	8B
6 Signal	5	9T
6 Return	E	9B
7 Signal	4	10T
7 Return	D	10B
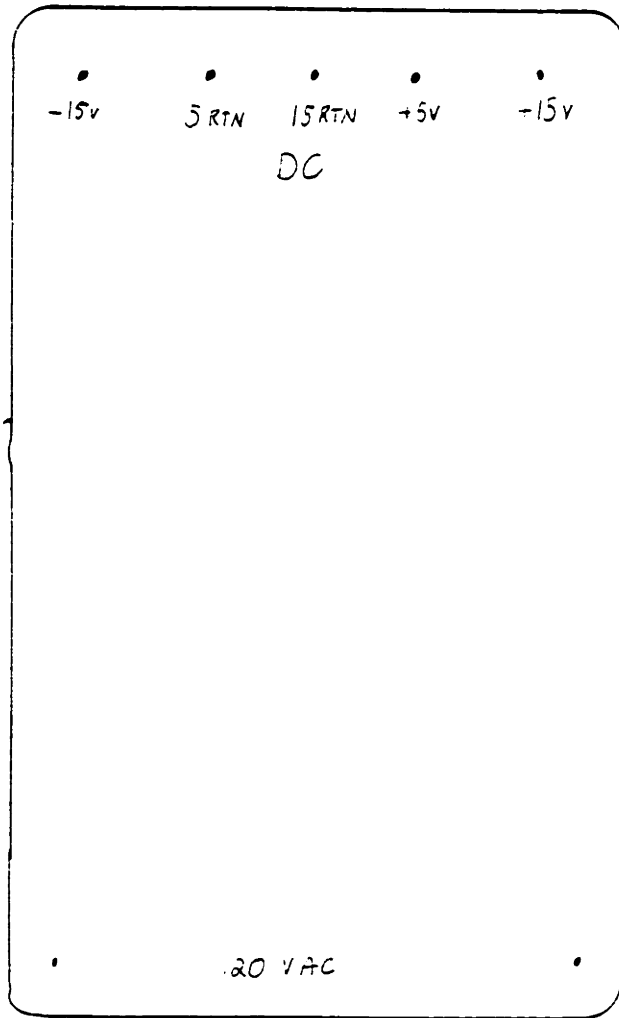
FP DAS 1128 Power Supply as viewed from the bottom

Sequential free running of te DAS 1128 uses:

$$\overline{\text{STROBE}} \rangle \qquad \langle \overline{\begin{array}{c} \text{DELAY OUT} \\ \overline{\text{TRIG}} \end{array}}$$

$\overline{\text{EOC}}$ → $\overline{\text{STRIBE}}$ → DELAY OUT → $\overline{\text{TRIG}}$ → $\overline{\text{EOC}}$ ...

EOC = end of conversion

17-September 1981

The enable logic was changed slightly to use line #8/50, external enable #2 (for the clock on external frame start #2). Thus WC INC ENB (P2-21 on DMA) is not on line 8/50 anymore, but is jumpered to +5v on the DMA board (blue wire wrap wire jumper)

18-November 1981

A 200 ohm resistor was added in series with a 470 pico-farad capacitor on the cycle request A line on the MDB DR11-B board at location R18. This is a simple filter to clean up the CYC REQ A line and help avoid spurious triggering.

CYC REQ A

220Ω

470
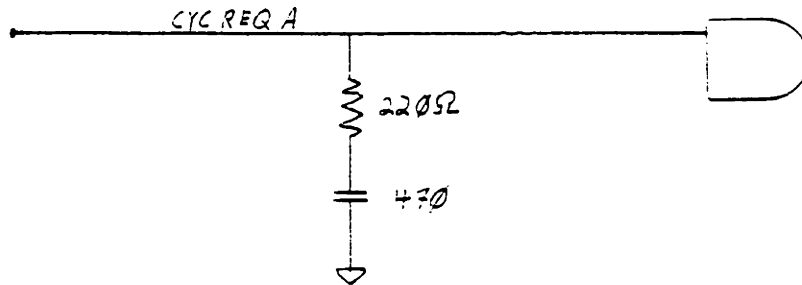
20-November 1981

All ground lines from the MDB DR11-B are also connected to +5v Ground at the FP, and the chassis ground (FP pin #22) is connected to FP +5v Gound (FP pin #A). This should cause ground loop problems, but actually seems to clean up the digital signals.

16-September 1981

1. The Delayed EOC was moved. It used to come from a 74121
    Q directly.   It is now taken from $\overline{Q}$ and inverted by a
    7437 NAND/Driver to help eliminate noise on the data
    strobe $\overline{EOC}$/CYC REQ A line.

2. 6 signals on the DMA previously strapped to +5v or
    Ground at the FP are now strapped on the DMA board
    directly using blue wire wrap wire jumpers.   To return
    the DMA board to its original state, remove all BLUE
    jumpers.

3. Those 6 signal lines were cut in the cable to the FP to
    help eliminate noise.   Probably an unnecessary move.
    See the cable conductor assignment in this document for
    details.

4. The +5v power supply at the FP formerly was grounded
    only through the cable to the DMA. It was found that
    signal noise was reduced dramatically by also grounding
    it at the FP chassis (through the front panel to pin #30
    on the 50 pin connector).

5. As of 16-September the FP and its interface works fine
    at up to 2.5 kHz frames per second (8 16-bit words per
    frame).

8-August-1981

        An error led to the connection of 120 VAC to the
chassis of the FP electronics unit. This destroyed the
Analog Devices DAS 1128, and the MDB DR11-B.   To eliminate
the possibility of this re-occurring, the chassis is now
grounded through a 3 wire power cable. This same ground is
used as a signal ground from the charge amplifiers, but the
inputs to the DAS 1128 A/D are all differential double ended
so this creates no problems.   The only possible problem
could come from the RESET/OPERATE control circuitry.
However, chassis ground is used as a reference for the
transistor and computer ground for the TTL logic.   This
should be OK.

        Care should be taken to preserve this isolation and NOT
connect +/- 15v or +5v grounds in the DAS circuitry to the
FP chassis ground.

FORCEPLATE DATA BUFFER ADDRESSES:

     FPSCOM   330500
Contains:

     FPBUF0 (768)   330500
     FPBUF1 (768)   333500
   (768. words = 3000 [octal]) bytes

768 words per buffer allows 96 frames to fill each buffer.
768 was chosen to allow an integral number of frames to fill
a buffer, and to be divisible by 256 to permit an integral
number of disk blocks to be written each time a buffer is
filled and then written to disk. For more details on the
double buffered DMA to disk software, see Appendix H.

    Forceplate DR11-B device register addresses

        772430   DRWC
        772432   DRBA
        772434   DRST
        772436   DRDB

    Interrupt vector: 130,   Priority BR6

FP transducer	FP charge amp	DAS 1128 channel	Transducer sensitivity
X1	X12	0	10 Nt/Volt
X2			
X3	X34	1	10 Nt/Volt
X4			
Y1	Y14	2	20 Nt/Volt
Y4			
Y2	Y23	3	20 Nt/Volt
Y3			
Z1	Z1	4	100 Nt/Volt
Z2	Z2	5	100 Nt/Volt
Z3	Z3	6	100 Nt/Volt
Z4	Z4	7	100 Nt/Volt

Note that the forceplate (Kistler) and the TRACK III global coordinate systems are different:

Kistler	TRACK III	
-Y	X	Anterior/Posterior
-Z	Y	Vertical
X	Z	Medial/Lateral

The TRACK III software corrects for this difference.

The Kistler charge amplifiers produce a +/- 10 volt output, and the DAS 1128 is set up for +/-10.24 volt input. This yeilds a 5 millivolt per discrete level in the digital output from the A/D converter.

ANALOG DEVICES

Low Cost, High Speed
Data Acquisition Module

DAS1128

FEATURES
Complete Data Acquisition System
12 Bit Digital Output
16 Single or 8 Differential Analog Inputs
High Throughput Rate
Selectable Analog Input Ranges
Versatile Input/Output/Control Format
Low 3 Watt Power Dissipation
Small 3" x 4.6" x 0.375" Module

GENERAL DESCRIPTION
The DAS1128 is a complete self-contained miniature high speed data acquisition system. The compact 3" x 4.6" x 0.375" module provides the designer with an easily implemented solution to the data acquisition problem. It contains an analog input signal multiplexer, a sample-and-hold amplifier, a 12 bit A/D converter, and all of the programming, timing and control circuitry needed to perform the complete data acquisition function.

The DAS1128 is a high performance device which can digitize an analog signal to an accuracy of ±½LSB out of 12 bits, relative to full scale. It has ±8ppm/°C gain temperature coefficient, and the maximum throughput rate can be varied from 50,000 conversions/second for a 12 bit conversion from different analog input channels, to 200,000 conversions/second for a successive 4 bit conversion made on a single channel.
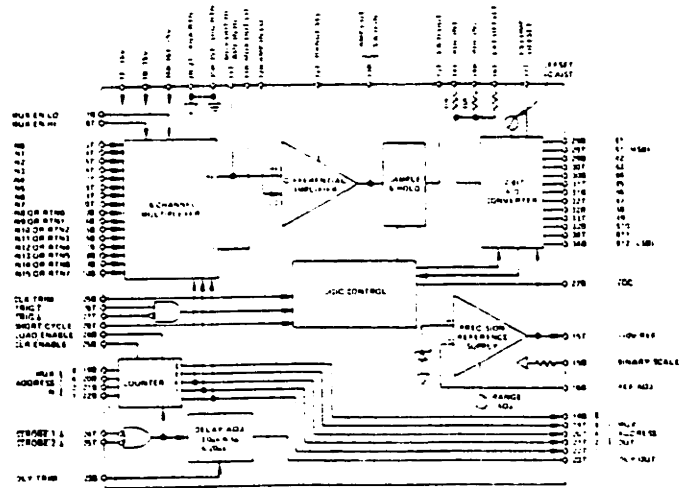
Figure 1. Functional Block Diagram

SPECIFICATIONS (typical @ +25°C and ±15V unless otherwise noted)

ANALOG INPUTS

Number of Inputs to Multiplexer	16 Single Ended, 8 True-Differential, 16 Pseudo-Differential
Input Voltage (Full Scale Range)	-10V to +10V, 0V to +10V, -5V to +5V, 0V to +5V, -10.24V to +10.24V, 0V to +10.24V, -5.12V to +5.12V, or 0V to +5.12V
Maximum Input Voltage	±15V
Input Current (per channel)	5nA max
Input Impedance	>10^8 ohms
Input Capacitance	10pF for "OFF" channel, 100pF for "ON" channel
Input Fault Current (power off or MUX failure)	Internally limited to 20mA
Direct ADC Input Impedance	10kΩ for each input line

ACCURACY[1]

Resolution	12 Bits
Error Relative to F S	±½LSB
Quantization Error	±½LSB
Differential Nonlinearity Error	
@ 33kHz throughput rate	±½LSB, 1LSB max
@ 50kHz throughput rate	±1LSB
Noise Error	±½LSB
-FS to +FS Error Between Successive Channel Transitions	±1LSB

TEMP. COEFFICIENTS

Gain	8ppm/°C, 20ppm/°C max
Offset	5ppm/°C, 15ppm/°C max
Differential Nonlinearity	2.5ppm/°C, 6ppm/°C max

SIGNAL DYNAMICS

Throughput Rate (12 Bits)	50kHz (max) (includes 5µs for MUX and SHA settling time plus 15µs for ADC)
MUX Crosstalk ("OFF" channels to "ON" channel)	>80dB down @ 1kHz
Differential Amplifier CMRR	70dB to 1kHz
SHA Acquisition Time to 0.01%	4.5µs max
SHA Aperture Uncertainty	10ns
SHA Feedthrough	70dB down w 1kHz

DIGITAL INPUT SIGNALS

Compatibility	Standard DTL/TTL logic levels, 1 unit load/line
MUX Address Inputs (8, 4, 2, 1; Pins 19B through 22B)	Positive true natural binary coding selects channel for random addressing mode. Must be stable for 100ns after STROBE.
MUX ENABLE HI (Pin 18T)	High (Logic "1") input enables MUX "HI" output (for inputs 0 through 7)
MUX ENABLE LO (Pin 17B)	High (Logic "1") input enables MUX "LO" output (for inputs 8 through 15)
STROBE (Pin 24T or 25T)	Negative going transition (Logic "1" to Logic "0") updates MUX address register. STROBE 1 must be a Logic "1" to enable STROBE 2. STROBE 2 must be at Logic "1" to enable STROBE 1.
LOAD ENABLE (Pin 24B)	High (Logic "1") input allows next STROBE command to sequentially advance MUX address register. Low (Logic "0") input allows next STROBE command to update MUX address register according to external address inputs.
CLEAR ENABLE (Pin 25B)	Low (Logic "0") input allows next STROBE command to reset MUX address to channel "0" overriding LOAD ENABLE.
TRIGGER (Pin 26T)	Positive going transition (Logic "0" to Logic "1") initiates A/D conversion (even during conversion): TRIGGER (Pin 27T) must be at Logic "0" to allow TRIGGER function.
TRIGGER (Pin 27T)	Negative going transition (Logic "1" to Logic "0") initiates A/D conversion: Pin 26T (TRIGGER) must be at Logic "1" to allow TRIGGER function.

DIGITAL OUTPUT SIGNALS

Compatibility	Standard DTL/TTL logic levels; 5 unit loads/line.
Parallel Outputs	BT, B1 through B12
Coding	Natural binary, two's complement, offset binary, or one's complement. Pin selectable.
MUX Address Outputs (8, 8, 4, 2, 1; pins 18B, 19T through 22T)	Positive true natural binary coding indicates channel selected
DELAY OUT (Pin 23T)	Negative going transition (Logic "1" to Logic "0") occurring normally 5µs (adjustable from 3.0µs to 20µs) after STROBE command initiates A/D conversion automatically when connected to the TRIGGER.
EOC (Pin 27B)	High (Logic "1") output during A/D conversion.

ADJUSTMENTS & TRIMS

Offset Adjust	
Internal Adjustment (Externally Accessible)	±10LSB's (min)
Remote External Adjustment (Pin 16T)	±10LSB's (min)
Range Adjust	
Internal Adjustment (Externally Accessible)	±10LSB's (min)
Remote External Adjustment (Pin 16B)	±10LSB's (min)
Clock Trim (Pin 26B)	
Factory Setting (Pin 26B "OPEN")	1.25µs/Bit
External Adjustment Range	1.25µs/Bit to 2.08µs/Bit
Delay Trim (Pin 23B)	
Factory Setting (Pin 23B "OPEN")	3.0µs
External Adjustment Range	3.0µs to 20µs

CONTROLS

SHORT CYCLE (Pin 28T)	Connect to ground for full 12 bit resolution. Connect to B$_n$ output for resolution to B$_{n-1}$ bits
Channel Selection Mode MUX Address Loading Mode)	Random, sequential continuous, and sequential triggered. Pin selectable.
A-D Conversion/Channel-Select Sequences	Normal (input channel remains selected during its A/D conversion) and overlap (next channel selected during A/D conversion). Pin selectable.
Range Select (Pin 12T)	Differential Amplifier gain control: connect to ANA RTN (Pin 2T) for X1 gain; connect to AMP OUT (Pin 13B) for X2 gain. This control is used in FSR selection procedure.
BINARY SCALE (Pin 15B)	Connect to REF ADJ (Pin 16B) to set reference to 10.24V. This control is used in FSR selection procedure, see Table II.
OUTPUT CODING (Pin 17T)	Ground for 1's complement output code; connect to -15V dc for other available codes.

POWER REQUIREMENTS

-15V ±3%	40mA, 50mA max
+5V ±3%	70mA, 100mA max
+5V ±5%	250mA, 500mA max
Power Supply Sensitivity[2]	
Gain	±2.0mV/V
Offset	±4.0mV/V
Ref	±0.5mV/V

ENVIRONMENT & PHYSICAL

Operating Temperature	0 to +70°C
Storage Temperature	-25°C to +85°C
Relative Humidity	Up to 95% non-condensing
Electrical Shielding	RFI & EMI 6 sides (except connector area)
Packaging	Insulated steel cased module 5.00" x 4.00" x 0.375"
PRICE	$295.00 (1-9), price includes mating right-angle connector.

[1] Warmup time to rated accuracy is 5 minutes
[2] Specification applies only when tracking +15V and -15V supplies are used, and for slowly occurring variations in power supply voltages.

Specifications subject to change without notice

-2-

THEORY OF OPERATION

A block diagram of the DAS1128 is shown in Figure 1. Analog input signals are applied to the various inputs of the 16 channel CMOS multiplexer. This multiplexer in conjunction with the differential amplifier that follows it, can be configured by the user to accept 16 single ended analog inputs, or 8 fully differential analog inputs. It can also be connected as a 16 channel "pseudo-differential" input device, which permits some of the benefits of differential operation while maintaining a 16 channel input capability.

The differential buffer amplifier is gain programmable by the user via jumpers at the module pins. This feature, along with the selectable reference voltages, permits the user to set up the DAS1128 to operate on any of 8 input voltage ranges. The differential amplifier drives a sample-and-hold amplifier, whose function it is to hold the selected analog input signal at a constant level while the A/D converter is making a conversion.

The A/D converter is a high speed 12 bit successive approximation device that has been designed using the Analog Devices' AD562, 12 bit integrated circuit D/A. The reference voltage for the conversion is supplied by an adjustable precision reference circuit that has a temperature coefficient of 5ppm/°C.

In addition to these basic functional blocks, the DAS1128 also contains all of the clock circuitry necessary to perform the complete data acquisition function. The internal clock can be externally adjusted to provide various throughput rates at different accuracies. Input channel addressing logic is provided, as is the capability to short cycle the A/D converter (i.e. perform conversions of less than 12 bits resolution). It is also possible for the user to adjust the time interval between input channel selection and the commencement of a conversion. The user can thus trade off speed vs. accuracy in the settling time of the multiplexer and sample-and-hold amplifier, as well as speed versus accuracy of the A/D converter.
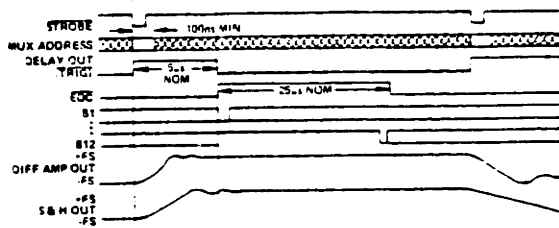
Figure 2. Simplified Timing Diagram, Showing Time-Interval Assignments and Constants

INPUT CONNECTIONS

As shown in Figure 3, three input configurations can be used. 16 single-ended inputs (3a) can be connected to the multiplexer, all referenced to analog gnd. In the second configuration (3b), the inputs are connected individually as 8 true differential pairs. In this case the differential amplifier is connected "Differentially" with the output of the MUX. Finally, a "Quasi-Differential" connection (3c) can be realized under favorable ground path conditions. In this configuration the differential amplifier Lo terminal is used as the ground return

for all sensors. In each of these input schemes, it should be noted that the input multiplexer has been designed to protect itself and signal sources from both overvoltage failure and from fault currents due to power-off loading or MUX failure.
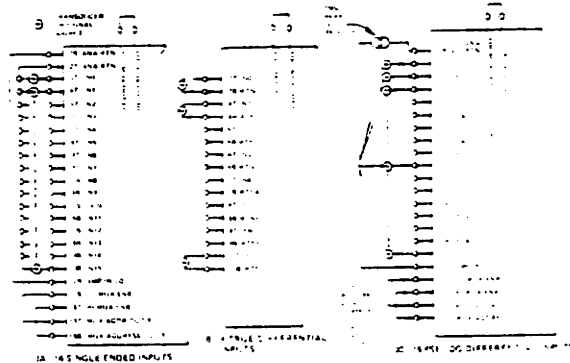
Figure 3. Signal Input Connections for Three Different Configurations

Full scale range of the DAS1128 may be set by appropriate jumper connections for 8 different ranges: 0 to +10V; 0 to +5V; 0 to +10.24V; 0 to +5.12V; -10 to +10V; -5 to +5V, -10.24 to +10.24V; -5.12 to +5.12V.

Note that 10.24 and 5.12 ranges are commonly used since conversion increments become 5mV/bit, 2.5mV/bit, and 1.25mV/bit.

MUX AND S/H DYNAMICS – OVERLAP MODE

The overlap mode is defined as the ability of MUX to accept a new channel address thereby selecting the next channel to be sampled while the previously acquired sample is being held by the S/H for conversion. The dynamic characteristics of the S/H circuit are shown in Figure 4. Maximum throughput rates are obtainable when a single channel is held at a single address and the channel is sampled repeatedly. In a dynamic condition, data-throughput rates obtainable are shown in Figure 5.
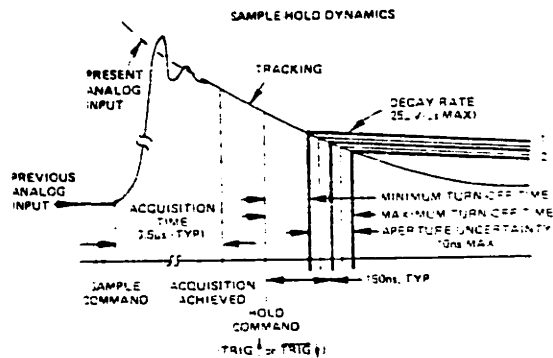
Figure 4. Sample-Hold Parameters Defined and Specified

SHORT CYCLE

It is possible to short cycle the DAC1128, i.e., stop the conversion after less than 12 bits. This can be done by connecting an external jumper between short cycle terminal and one of the output terminals. With shorter cycles the attainable throughput rate increases, see Figure 5. In short cycle operation the EOC will decrease proportionately to the number of bits selected. Note the short cycle terminal *must* be grounded for full 12-bit operation.
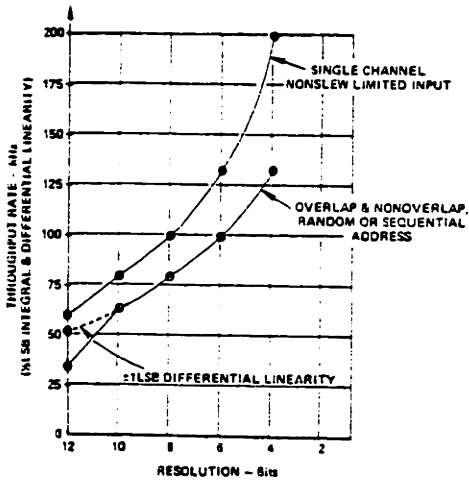
Figure 5. DAS1128 Throughput Rates

MUX ADDRESSING

External terminals have been provided for the address counter. Thus the address counter can be configured to produce the following modes: Continuous sequential scanning (free running), sequential scanning with external step command, abbreviated scan continuously, random channel selection. See Figure 6 and set up procedure for details.
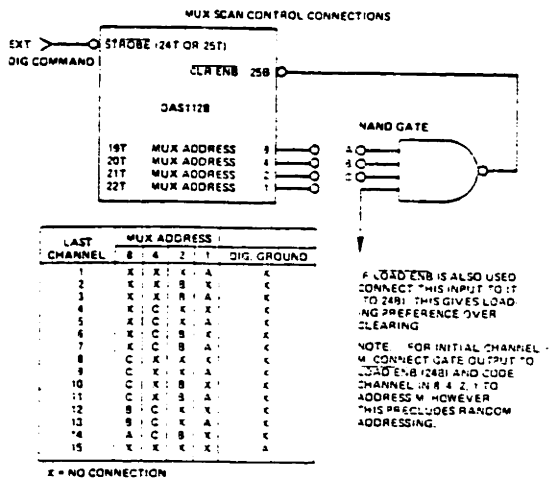
Figure 6. To shorten scanning sequency of multiplexer channels, make the appropriate connections. (as shown in the chart) between an external NAND gate and MUX ADDRESS terminals 19T to 21T

GROUNDING CONSIDERATIONS

Attention should be given to the methods of connection for electrical returns and voltage reference points. Analog return (ANA RTN) and digital return (DIG RTN) are provided. The following rules should be applied when integrating the DAS1128 into the system.

1. If the ±15V power supply is floating (for optimum analog accuracy), connect its return to ANA RTN (Pin 2B or 2T). If the ±15V power supply is *not* floating, connect its return to DIG RTN (Pin 35T or 35B).

2. Connect the +5V supply return to DIG RTN (Pin 35T or 35B). If this supply also powers additional equipment, run separate, parallel returns to the equipment ground and to DIG RTN (Pin 35T or 35B).

3. To minimize signal grounding problems, single-ended input signals should only be returned to ANA RTN (Pin 2B or 2T). If this is not possible, then connect the input signals in either the "true differential" or "pseudo-differential" configurations (see Figure 3).

4. Connect computer ground to DIG RTN (Pin 35T or 35B). Use heavy wire or ground planes.

5. The computer chassis should be connected to the computer and power supply grounds at only one point.

6. Connect the third-wire ground from main ac power input to the computer power supply return.

GAIN AND OFFSET ADJUSTMENTS

The DAS1128 is calibrated with external gain and offset adjustment potentiometers connected as shown in Figure 7 and 8. The offset adjustment potentiometer has an adjustment range of at least ±10LSB's, and the gain range adjustment potentiometer has an adjustment range of at least ±10LSBs.

Offset calibration is not affected by changes in gain calibration, and should therefore be performed prior to gain calibration. Proper gain and offset calibration requires great care and the use of extremely sensitive and accurate reference instruments. The voltage standard used as a signal source must be very stable. It should be capable of being set to within ±1/10LSB of the desired value at any point within its range.

These adjustments are not made with zero and full scale input signals, and it may be helpful to understand why. An A/D converter will produce a given digital word output for a small range of input signals, the nominal width of the range being one LSB. If the input test signal is set to a value which should cause the converter to be on the verge of switching between two adjacent digital outputs, the unit can be calibrated so that it does switch at just that point. With a high speed convert command rate and a visual display, these adjustments can be performed in a very accurate and sensitive way. Analog Devices *Analog-Digital Conversion Notes* gives more detailed information on testing and calibrating A/D converters.

OFFSET CALIBRATION

For unipolar +10V operation set the input voltage precisely to +0.0012V and adjust the offset potentiometer until the converter is just on the verge of switching from 000000000000 to 000000000001.

For ±5V bipolar operation set the input voltage precisely to -4.9988V for ±10V units set it to -9.9976V. Adjust the offset

(handwritten) -4- for ±10.24V set it to -10.2375V.
½ LSB from most Negative Voltage.

potentiometer, Figure 7, until Offset Binary coded units are just on the verge of switching from 000000000000 to 000000000001 and Two's Complement coded units are just on the verge of switching 100000000000 to 100000000001.

INT. OFFSET ADJUSTMENT
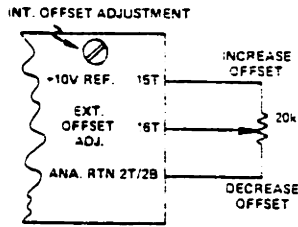
Figure 7. Ext. Offset Adjustment

GAIN CALIBRATION

Set the input voltage precisely to +9.9963V for unipolar operation, +4.9963V for inputs of ±5V or +9.9926V for inputs of ±10V. Note that these values are 1½LSB's less than nominal full scale. Adjust the 20k variable gain resistor, Figure 8, until Binary and Offset Binary coded units are just on the verge of switching from 111111111110 to 111111111111 and Two's Complement coded units are just on the verge of switching from 011111111110 to 011111111111.

For ±10.24V set it to +10.2325V.

INT. RANGE ADJUSTMENT
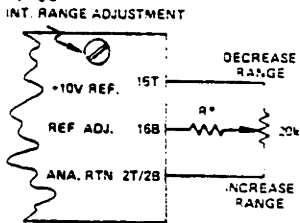
*R SHOULD BE AT LEAST 47kΩ AND NO MORE THAN 470kΩ REDUCING R INCREASES ADJUSTMENT RANGE.

Figure 8. Ext. Ref. Adjustment

CLOCK RATE ADJUSTMENT

The clock rate may be adjusted for best conversion time/accuracy trade-off. The conversion time is varied by means of the external circuitry shown in Figure 9. An open CLK TRIM terminal (Pin 26B) results in 1.25μs/bit nominal conversion time. A grounded CLK TRIM terminal (for highest accuracy) results in 2.08μs/bit conversion.
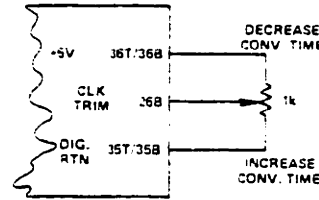
Figure 9. Clock Trim

DELAY TIME ADJUSTMENT

The DLY OUT signal may be adjusted to vary the A/D converter triggering time by means of the external circuitry shown in Figure 10. An open DLY TRIM terminal (Pin 23B) results in a nominal delay time of 3.0μs. A grounded DLY TRIM terminal (for highest-accuracy) results in 20μs delay time nominal.
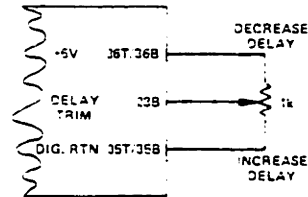
Figure 10. Delay Trim

TABLE I

INPUT CONFIGURATION	ANALOG INPUT CONNECTIONS	ANALOG INPUT RETURN	JUMPER CONNECTIONS
16 Single-Ended Inputs (Figure 3a)	3T thru 10T and 3B thru 10B	All input returns to 2B or 2T	11B to 11T 12B to 2B or 2T 17B to 19T 18T to 18B
8 Differential Inputs (Figure 3b)	3T thru 10T	3B thru 10B	11B to 12B 17B to 18T to "1"
16 Pseudo-Differential Inputs (Figure 3c)	3T thru 10T and 3B thru 10B	Common Input return to 12B	11B to 11T 17B to 19T 18T to 18B

RECOMMENDED SET-UP PROCEDURE

1. Select input configuration, see Table I.

2. Select MUX address mode.
The method of addressing the multiplexer can be selected by connecting the unit as follows:

RANDOM. Set Pin 24B (LOAD ENB) to Logic "0". The next falling edge of STROBE will load the address presented to Pins 19B through 22B (8, 4, 2, 1). The code on these lines must be stable during the falling edge of STROBE plus 100ns.

SEQUENTIAL FREE RUNNING. Set to Logic "1", Pin 24B (LOAD ENB) and 25B (CLR ENB). Connect Pin 27B (EOC) to Pin 24T (STROBE 1). Connect Pin 23T (DLY OUT) to Pin 27T (TRIG). Use Pin 26T (TRIG) as a run/stop control (i.e., A/D conversion will continue while TRIG is high and will stop while TRIG is low).

SEQUENTIAL TRIGGERED. Set to Logic "1", Pins 24B (LOAD ENB) and 25B (CLR ENB). Connect Pin 24T (STROBE) to external triggering source. The multiplexer address register will automatically advance by one channel whenever a STROBE command is received. The initial channel can be selected by setting Pin 24B (LOAD ENB) to Logic "0" during only one STROBE command. The multiplexer address will then be determined by the logic levels on Pins 19B through 22B (the external MUX address lines). Channel "0" can be selected as the initial channel by setting Pin 25B (CLR ENB) to Logic "0" during only one STROBE command. The final channel can be selected by following the procedure presented in Figure 6.

3. Select A-D conversion/channel select sequence (see Figure 5).

   (1) NORMAL (input channel remains selected during its A/D conversion). Connect Pin 23T (DLY OUT) to Pin 27T (TRIG).

   (2) OVERLAP (next channel is selected during A/D conversion). Connect Pin 27B (EOC) to TTL compatible inverter input. Connect inverter output to Pin 24T (STROBE). Connect Pin 23T (DLY OUT) to Pin 27T (TRIG). Adjust the delay to at least 4µs greater than EOC, 20µs max (see Figure 10). The signal on Pin 26T (TRIG) serves as RUN/STOP control.

   (3) REPETITIVE SINGLE CHANNEL. After selecting the input channel to be repetitively sampled (see MUX ADDRESS MODE, above), set Pin 27T (TRIG) to Logic "0". Connect Pin 26T (TRIG) to a triggering source. Conversion process is initiated by positive edge of TRIG command.

4. Select output resolution.
   a. Full 12 bit resolution: connect Pin 28T (SHT CYC) to Pin 35B (DIG RTN).
   b. Bn (Bn < 12) bit resolution: connect Pin 28T to the output pin for Bn + 1.

5. Select optimum throughput rate.
The system clock frequency and the STROBE to TRIG delay (if used) can be trimmed to optimize the accuracy/throughput rate trade-off. See Figures 9 and 10.

6. Select input voltage full scale range. See Table II.

7. Select output digital coding. See Table III.

TABLE II

FOR FULL SCALE RANGE OF:	MAKE THE FOLLOWING CONNECTIONS
0 to +10V	12T to 2T; 14T to 14B to ADC Source*
0 to +10.24V	same as 0 to +10V, plus 15B to 16B.
0 to +5V	12T to 13B; 14T and 14B to ADC Source*
0 to +5.12V	same as 0 to +5V, plus 15B to 16B
-10V to +10V	12T to 2T; 14T to 15T; and 14B to ADC Source*.
-10.24V to +10.24V	same as -10V to +10V, plus 15B to 16B
-5V to +5V	12T to 13B; 14T to 15T and 14B to ADC Source*.
-5.12V to +5.12V	same as -5V to +5V, plus 15B to 16B.

*ADC Source is usually Sample and Hold Output (13T), but may be any signal source including Diff. Amp. Output (13B) if Sample and Hold is not desired.

TABLE III

OUTPUT CODE	CONNECTIONS
Unipolar Binary	Connect 17T to -15V Use 29T (B1) for MSB
2's Complement	Connect 17T to -15V Use 28B (B̄1) for MSB
Offset Binary	Connect 17T to -15V Use 29T (B1) for MSB
1's Complement	Connect 17T to 2B Use 28B (B̄1) for MSB
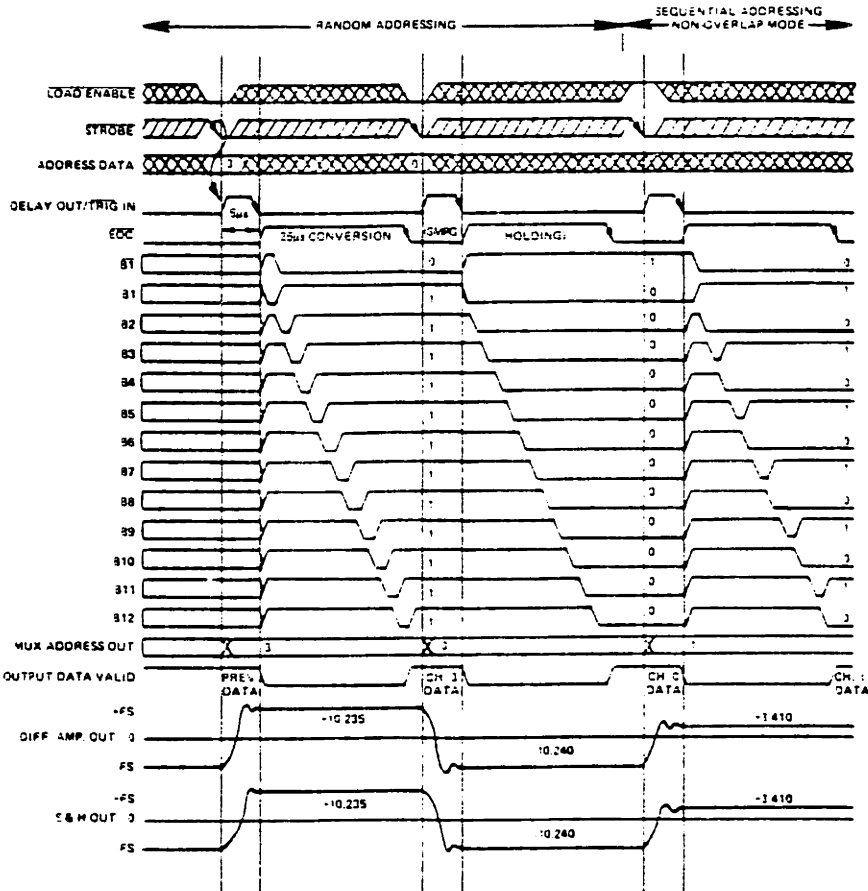
Figure 11. Timing for Non-Overlap Operation in Both Random and Sequential Addressing Modes. For Status Keys and Signal Condition Data, Refer to Box Below.

Figure 12. Timing Diagram for Overlap Operation in the Sequential Addressing Mode. For Status Keys and Signal Condition Data, See Box at Right.

SIGNAL CONDITIONS AND STATUS KEYS FOR FIGURES 11 AND 12.

CH. 2 = -3.415V CODE 010 101 010 101
CH. 3 = +10.235V CODE 111 111 111 111
CH. 0 = -10.240V CODE 000 000 000 000
CH. 1 = +3.410V CODE 101 010 101 010

ADC SET UP FOR ±10.24V INPUT, OFFSET BINARY. (FOR TWO'S COMPLEMENT, USE $\overline{B1}$ FOR M.S.B.)

KEY	INPUTS	OUTPUTS
XXX	May change	Don't know
ZZZ	May change 0 to 1	Changes 0 to 1
SSS	May change 1 to 0	Changes 1 to 0
OR	Must be stable	Will be stable

MDB-DR11B
DIRECT MEMORY ACCESS MODULE

INTRODUCTION

The MDB-DR11B Direct Memory Access Module is an interface for the direct-memory-access transfer of data between a Digital Equipment Corporation (DEC) PDP-11 computer memory, and the user's peripheral device (figure 1).

The MDB-DR11B consists of a single quad module. The module fits into only one SPC slot of a DD11-A, B, or C assembly, unlike the DEC DR11B which occupies four slots. The MDB module works with all standard DEC software for the DR11B and performs all DEC DR11B functions.

Logic on the module includes the following facilities:

a.   bus drivers and receivers;

b.   four registers, as follows:
           Data Register
           Word Count Register
           Address Register
           Command/Status Register;

c.   a multiplexer to select the bus address, word count, input data, or command/status information onto the Unibus;

d.   logic to decode the received device address; and
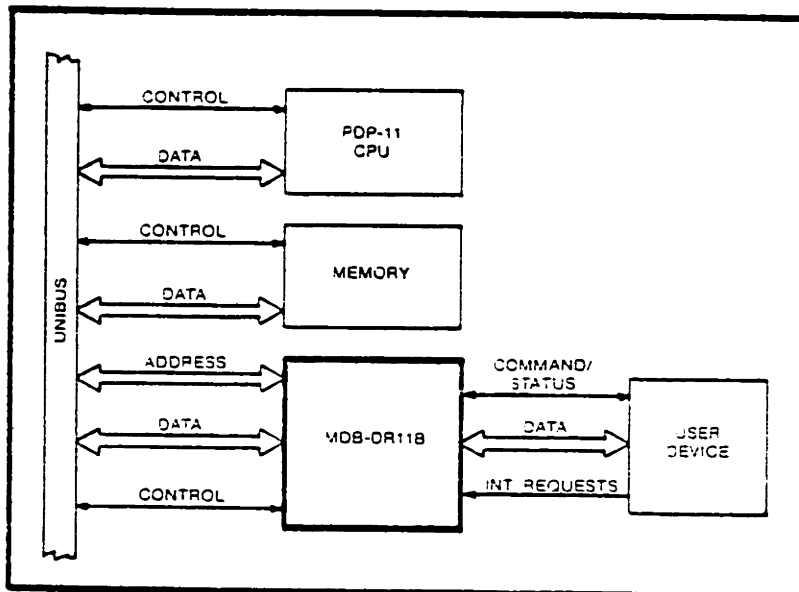
Figure 1.   System Block Diagram

e.   interrupt and bus master control logic.

Operation is always begun under program control which does the following:

a.   loads the required word count into the Word Count Register;

b.   specifies the starting memory address or bus address at which the block transfer is to begin; and

c.   loads function bits into the Command, Status Register.

The user device responds to the function bits by setting-up controls to the DR11B. If the user device is to receive data, the DR11B performs a DATI operation, loading its data register with information at the specified bus address. Outputs of the Data Register are presented to the user device.

If the user device is to send data to the memory, the DR11B performs a DATO operation, transferring data words from the user device to the specified bus address. Note that input data is not buffered and must be retained for the entire Unibus transfer.

Data and control signals at the MDB-DR11B user interface are at standard TTL levels.

PHYSICAL DESCRIPTION

The MDB-DR11B module is a quad module fitting into one slot in the systems assembly. The interface to the Unibus is through the assembly backplane. The module is connected to the user device through two 40-pin Berg connectors, J1 and J2. User logic may be built on an MDB-M91WW wire-wrap board, connected to the MDB-DR11B through short ribbon jumpers.

The MDB-DR11B is powered by the +5V dc supply at the systems assembly, and requires 2.4 amperes from that supply.

INSTALLATION

The following paragraphs contain instructions and information for installing the MDB-DR11B module, and for installing or removing jumpers that configure the module for its application.

INSTALLING MODULE

Plug the module into any available small peripheral slot in the PDP-11 processor, or into one of the four slots in a DD11 Peripheral Mounting Panel (figure 2). Figure 2 shows the backplane wiring that may be required to maintain the non-processor grant (NPG) chain. Note that in some newer processors and DD11 system units a wire jumper is installed between CA1 and CB1. In these cases, it is only necessary to remove that wire jumper in the slot in which the MDB-DR11B is to be installed.

CABLING

The module has two Berg connectors for the user device interface. Cables from the device may be brought directly to these connectors, or to an MDB-M91WW wirewrap board

Table 1. Interface Cable Connectors, Pin Assignments

Connector P1		Connector P2	
Pin	Signal	Pin	Signal
1	DATI15	1	ODAT15
2	DATI00	2	ODAT00
3	DATI14	3	ODAT14
4	DATI01	4	ODAT01
5	DATI13	5	ODAT13
6	DATI02	6	ODAT02
7	DATI12	7	ODAT12
8	DATI03	8	ODAT03
9	DATI11	9	ODAT11
10	DATI04	10	ODAT04
11	DATI10	11	ODAT10
12	DATI05	12	ODAT05
13	DATI09	13	ODAT09
14	DATI06	14	ODAT06
15	DATI08	15	ODAT08
16	DATI07	16	ODAT07
17	CO CONTROL IN	17	GND
18	A00IN	18	GND
19	ATTN	19	INIT0
20	GND	20	GND
21	BA INC ENB	21	WC INC ENB
22	GND	22	GND
23	no connection	23	READY0
24	GND	24	GND
25	BUSY0	25	470 OHMS TO +5V DC
26	GND	26	GND
27	C1 CONTROL IN	27	FNCT10
28	GND	28	NO LOCK
29	DSTATC	29	FNCT10
30	GND	30	GND
31	DSTATB	31	FNCT20
32	GND	32	GND
33	SINGLE CYCLE	33	FNCT30
34	GND	34	GND
35	DSTATA	35	FNCT30
36	GND	36	GND
37	GO0	37	CYCLE REQ B
38	GND	38	GND
39	CYCLE REQ A	39	END CYCLE 0
40	GND	40	GND

JUMPER CONNECTIONS

Certain jumper connections may be prepared on the module in order to configure the module for its application. The module is furnished with certain preferred configurations strapped by printed circuit etch. To change configurations, cut etch and or install wire jumpers as required.

Device Address

Ten bits of the device address are strapped at jumpers numbered 13 through 22 (zone 13A on the module). The module is furnished strapped for address 772410.

A connection from pad J to pad K selects a logic "1", and a connection from pad J to pad H selects a logic "0". Jumpers are related to address bits as follows:

Bit	Jumper
A03	15
A04	14
A05	22
A06	13
A07	16
A08	19
A09	21
A10	20
A11	18
A12	17

Address assignments determined by standard DEC software are as follows:

No. of DR11B Module	Register Address
First	772410 - 772416
Second	772430 - 772436
Third	772450 - 772456
Fourth	772470 - 772476

Vector Address

The 6-bit vector address is strapped at jumper locations 23 through 28 (zones 6A and 5A). The module is furnished strapped for vector address 124. Where more than one module is used, the user must assign a different address to each module, with 124 assigned to the first module.

A connection from pad J to pad H selects a logic "1", and a connection from pad J to pad K selects a logic "0". Jumpers are related to data bus bits as follows:

Bit	Jumper
DATA02	28
DATA03	27
DATA04	26
DATA05	24
DATA06	23
DATA07	25

Bus Master Control Level

The module is furnished with etched connections to select level 5. That is, BR5L, BG5INH, and BG5OUT are connected to driver receiver circuits. Inputs for BG4INH, BG6INH, and BG7INH are etch-jumpered to corresponding BG4OUTH, BG6OUTH, and BG7OUTH lines.

To select a level other than level 5, cut the etch for level 5, and connect wire jumpers for the selected level, as follows (be sure to remove any etched or wire jumpers not listed for the selected level):

Level 4	Level 5	Level 6	Level 7
4/H-J	3/H-J	2/H-J	1/H-J
11/H-J	9/H-J	7/H-J	5/H-J
12/H-J	10/H-J	8/H-J	6/H-J
9/H-10/H	11/H-12/H	11/H-12/H	11 H-12 H
7/H-8/H	7/H-8/H	9/H-10/H	9 H-10/H
5/H-6/H	5/H-6/H	5/H-6/H	7/H-8/H

Bus Address Overflow Control

The module is furnished with etched jumpers that cause the logic to set READY and ERROR if the bus address overflows the 32K boundary set by some DEC software. These jumpers are 29/H-J and 30/H-J.

If your application permits disregarding the 32K boundary, jumpers may be reconfigured so that each 32K overflow simply increments the Extended Bus Address count, and sets READY and ERROR only when the address has overflowed the limit of system addressing. To select this mode of operation, cut etch jumpers 29 H-J and 30/H-J, and connect wire jumpers at 29/J-K and 30/J-K.

Non-Processor Request Control

The module is furnished to permit an interrupt cycle to be performed without interruption by an NPR request from some other device. That is, there is an etch connection at 31/H-J.

If an NPR request is to be permitted to take control during an interrupt latency period (between the interrupt request and the bus grant), cut the etch at 31/H-J and connect a wire jumper at 31/J-K.

REGISTERS

There are four registers available to the Unibus. Each register is described in the following paragraphs.

DATA REGISTER

The 16-bit Data Register is used both to read, and write, data as follows:

    a.   **Write Data.** The register stores the output data word for presentation to the user device. The register is loaded under program control.

    b.   **Read Data.** A data word from the user device is transferred to the Unibus without buffering. That is, the user device must hold data on the lines until it is read under program control or transferred directly to memory.

The preferred Data Register address is 772416.

WORD COUNT REGISTER

The Word Count Register is a 16-bit read write register. It is loaded from the Unibus, under program control, with the 2's-complement of the number of words to be transferred, and normally is incremented one count towards zero as each word is transferred. Incrementing may be inhibited by a WC INC ENB signal from the user device.

When the word count reaches zero, the READY bit is set in the command status word to stop the bus cycle.

The preferred Word Count Register address is 772410.

BUS ADDRESS REGISTER

The Bus Address Register is a 15-bit register, as bit 0 is furnished by the user device. The contents of the register are used, along with bits XBA16 and XBA17 (in the Command Status Register) to specify the bus address.

The Bus Address Register is normally incremented after each word to transferred. advancing the address to the next word location. The user device may inhibit incrementing by asserting BA INC ENB.

The ERROR bit is set in the Command Status Register if the address overflows (changes from all-"1's to all-"0's"). The error (BAOF) is cleared either by INIT or by loading a new number into the Bus Address Register.

The preferred Bus Address Register address is 772412.

COMMAND/STATUS REGISTER

The contents of the Command Status Register are commands to control the user device. and bits giving user device status to the Unibus. Table 2 lists and defines bits in the Command Status Register.

The preferred Command Status Register address is 772414. Figure 3 shows the contents of the Command Status Register.
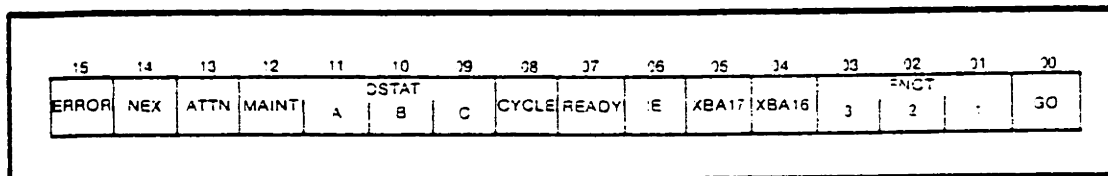
Figure 3. Contents of Command/Status Register

Table 2. Command/Status Bits

Bit	Name	Description and Effect
15	ERROR (read only)	a. Indicates error as follows: 1. NEX (bit 14), or 2. ATTN (bit 13), or 3. Interlock error (module/connector discontinuity), or No Lock signal (P2-28) not grounded, or 4. Bus address overflow (BAOF) as bus address changes from all-"1's" to all-"0's". b. Sets READY (bit 7) and causes interrupt if IE (bit 6) has been set. c. ERROR is cleared by clearing all error conditions, as follows: 1. Module is seated in connector. 2. Bus address is cleared or reloaded. 3. Bit 14 is loaded with a "0". 4. Bit 13 is cleared by the user device.
14	NEX (read)	a. Non-existent Memory. Indicates that the module, acting as bus master, failed to receive a SSYN response within 20 microseconds after asserting MSYN. b. NEX sets ERROR bit. c. Cleared by INIT or by loading "0".
13	ATTN (read only)	a. Attention. Shows state of user device ATTN signal. b. Sets ERROR for device-initiated interrupt. c. Set and cleared only by user device.
12	MAINT (read/write)	a. Maintenance. Used to enable execution of diagnostic programs. b. Cleared by INIT.

Table 2. Command/Status Bits (cont'd)

Bit	Name	Description and Effect
11 10 09	DSTATA DSTATB }(read only) DSTATC	a. **Device Status.** Indicate state of user-designated DSTATA, DSTATB, and DSTATC signals. b. Set and cleared only by user device.
08	CYCLE (read/write)	a. If set when GO is issued, enables an immediate bus cycle. b. Cleared by INITI, or start of bus cycle.
07	READY (read only)	a. Indicates the MDB-DR11B is able to accept a new command. b. Set by INIT or ERROR, or by word count overflow. c. Cleared by GO. d. If bit 6 is set, READY causes an interrupt, forcing module to release the Unibus.
06	IE (read/write)	a. **Interrupt Enable.** Enables either ERROR or READY to set an interrupt. b. Cleared by INIT.
05 04	XBA17 }(read/write) XBA16	a. **Extended Bus Address.** Along with contents of Bus Address Register, specify address for **indirect** memory transfers. b. Cleared by INIT. c. Bits XBA17 and XBA16 are not incremented when Bus Address Register overflows, but ERROR is set.
03 02 01	FNCT3 FNCT2 (read/write) FNCT1	a. **Function.** Bits available to user device for assignment by user. b. Cleared by INIT.
00	GO (write only)	a. Causes MDB-DR11B to signal user device that a command has been issued. b. Clears READY.

USER DEVICE INTERFACE

Signals at the user device interface are at standard TTL levels. Output drivers are 74367
bus drivers with "enable" inputs strapped to ground. Table 4 lists and defines signals at
the user device interface. Unless otherwise stated, logic is asserted at the high level.

Table 4. User Device Interface Terms

Signal	Description
Inputs to MDB-DR11B from User Device	
DAT100-DAT115	Sixteen data lines. Data must be held until transferred to memory in a DATO cycle.
C1 CONTROL IN C0 CONTROL IN	Two signals specify type of Unibus cycle to be performed, and have the same meanings (but opposite logic levels) as Unibus signals C1 and C0 (refer to table 1). (High true, 0 = unasserted, 1 = asserted) C1H C0H Cycle 0 0 DATI 0 1 DATIP 1 0 DATO 1 1 DATOB
CYCLE REQ A. CYCLE REQ B	Either signal sets CYCLE bit to initiate bus request and subsequent Unibus cycle. Must be pulsed positive for at least 100 nanoseconds (negative-going transition is active).
WC INC ENB	**Word Count Increment Enable.** To permit counting each bus cycle, this line must be held high. For certain operations, however, such as a read-modify-write sequence, the line may be brought low for the DATIP cycle and raised for the following DATO cycle.
BA INC ENB	**Bus Address Increment Enable.** To permit the Bus Address Register to increment following each bus cycle, this line must be held high. For certain operations, however, such as a read-modify-write sequence, the line may be brought low for the DATIP cycle and raised for the following DATO cycle.
A00IN	**Bus Address Bit 00.** Used as bit 0 of the Bus Address Register. For sequential word addressing A00 is held low. The line may be controlled to permit byte addressing.
DSTATA, DSTATB. DSTATC	**Device Status Bits.** User-assigned status signals. Levels on these lines appear as bits 09, 10, and 11 in the Command Status Register.
ATTN	**Attention.** The signal level on this line becomes bit 13 in the Command Status Register. ATTN causes an error condition and prevents further bus cycles. If the IE bit is set, ATTN causes an interrupt. If not used, ATTN must be held low.

Table 4. User Device Interface Terms (cont'd)

Signal	Description
SINGLE CYCLE	When held high, SINGLE CYCLE causes MDB-DR11B to release bus after each cycle, permitting Unibus to interleave MDB-DR11B cycles with cycles of other devices. In this case, MDB-DR11B requests bus master for each cycle. For burst-mode or read-modify-write operations, SINGLE CYCLE is held low, causing MDB-DR11B to control the bus until SINGLE CYCLE goes high, or until READY is set. Bus cycle does not begin until CYCLE is set.
Outputs from MDB-DR11B to User Device.	
ODAT00-ODAT15	Data to user device. Data is buffered in Output Data Register, which is loaded under program control, or by an MDB-DR11 DATI cycle. INIT clears all lines to "0".
INITO	**Initialize.** Goes high when Unibus is initialized.
FNCT1, FNCT2, FNCT3	**Function.** Bits 01, 02, and 03 in Command Status Register, output through drivers. Defined by user to control device operation. Cleared by INIT.
READYO	Bit 07 in Command Status Register. Set high by INIT. Goes low as GO bit is loaded to indicate that a command has been received. Set high again by word count overflow or error.
BUSYO	High level while bus cycle is in progress. Set high as CYCLE is set, and goes low when cycle is complete. When CYCLE is controlled by program, BUSYO follows CYCLE.
END CYCLE O	Positive-going pulse (approx. 100 nsec) output as bus cycle is ended.
GOO	Positive-going pulse (approx. 200 nsec) output as GO is set in Command/Status Register. Indicates start of new operation.
NO LOCK	Must be grounded. Used to set ERROR in CSR. Indicates that cable is not attached to P2.

The sequence of operation at the MDB-DR11B user device interface is, generally, as follows:

    a.   The GO bit is set, and READY goes low, indicating that a cycle is to begin. FNCT bits may define the command.

    b.   The user device then provides the following signals: DAT100-DAT115, C1 CONTROL, C0 CONTROL, WC INC ENB, and A001N. These signals must be held on the lines throughout the bus cycle.

BUS MASTER MODE

In the Bus Master mode of operation, the user device causes the MDB-DR11B to take control of the bus and perform either an interrupt operation, or a data transfer operation.

Interrupt Operation
The MDB-DR11B can initiate an interrupt cycle if both bit 06 (Interrupt Enable) and bit 07 (READY) are set in the Command, Status Register. The resulting bus request (BRnL) requests control of the bus.

The master device then sends the bus grant signal (BGnINH) which the module acknowledges by returning SACKL. When the bus is free, the module puts the vector address on the bus, along with the signal INTRL. This directs the program to the subroutine that controls the user device.

Data Transfer Operation
The module will perform DATI, DATIP, DATO, and DATOB operations.

The program first loads Bus Address and Word Counter Registers, and then loads bit 0 (GO) into the Command Status Register to clear READY. A CYCLE REQUEST (A or B) from the user device initiates transfer by setting CYCLE, and consequently, BUSY. A bus request is then sent to the master device which responds with the bus grant signal, permitting the module to become bus master.

The module makes its request on the NPR line and, when the NPR bus grant is received, becomes bus master, asserting BUS BBSY. The address is then put on the bus. C0 CONTROL and C1 CONTROL lines from the user device determine subsequent operation.

MAINTENANCE

MAINTENANCE MODE OPERATION

The MAINT bit (bit 12) in the Command, Status Register enables the MDB-DR11B to operate with existing DEC software for DR11-B diagnostics. To configure the MDB-DR11B for diagnostic operation (with outputs looped to inputs), connect jumper cable (supplied) between connectors J1 and J2 (refer to table 1 for pin connections).

The MAINT bit causes the FNCT bits to act as an octal counter which counts bus cycles. Note that with FNCT1 connected to the C1 CONTROL line, the MDB-DR11B performs alternating DATI and DATO cycles. If J1 and J2 are jumpered together but the MAINT bit is not set, either all DATI cycles, or all DATO cycles, will be performed.

Because FNCT3 is looped to the SINGLE CYCLE line, the MAINT bit causes a burst mode sequence of four cycles, followed by four single cycles.

Refer to appropriate DEC documentation for details of diagnostics.

APPENDIX C

SELSPOT SPECIFICATIONS AND TIMING

The Selspot hardware manual provides a good treatment of the internal electronics, and Conati (1977) [34] discusses the function of the Selspot. A timing diagram, and the modifications to the Selspot backplane are included here.

The Selspot is a free running device. There is no "handshake" with an external device. Thus it always runs at 315 Hz, and puts out 37500 words of data per second. There is no provision for holding a data word valid until the computer can read it. Thus the Selspot DMA interface has the highest priority, to ensure that it will be serviced before any other device, and minimize the risk of missing a data word.

Signal to noise in the Selspot is linearly related to LED intensity (until the detectors are overdriven). Thus one should always use the brightest LEDs possible for a given experiment.

All TRACK III LED array segments used the following wire color coding:

Ground	Brown
Channel 1	Red
2	Orange
3	Yellow
4	Green
5	Blue
6	Purple
7	Grey
8	White

Originally the Selspot LED Control Unit (LCU) had no
provision to allow single LEDs to be plugged in. The LCU
only had a 50 pin connector to access all channels. I built
a mainfold for one end of the LCU which incorporated 30
"mini-phono" jacks. All jacks have a common ground
(sheild). The jacks are connected the LED channels as
follows:

TIMING DIAGRAM FOR DIGITAL OUTPUTS
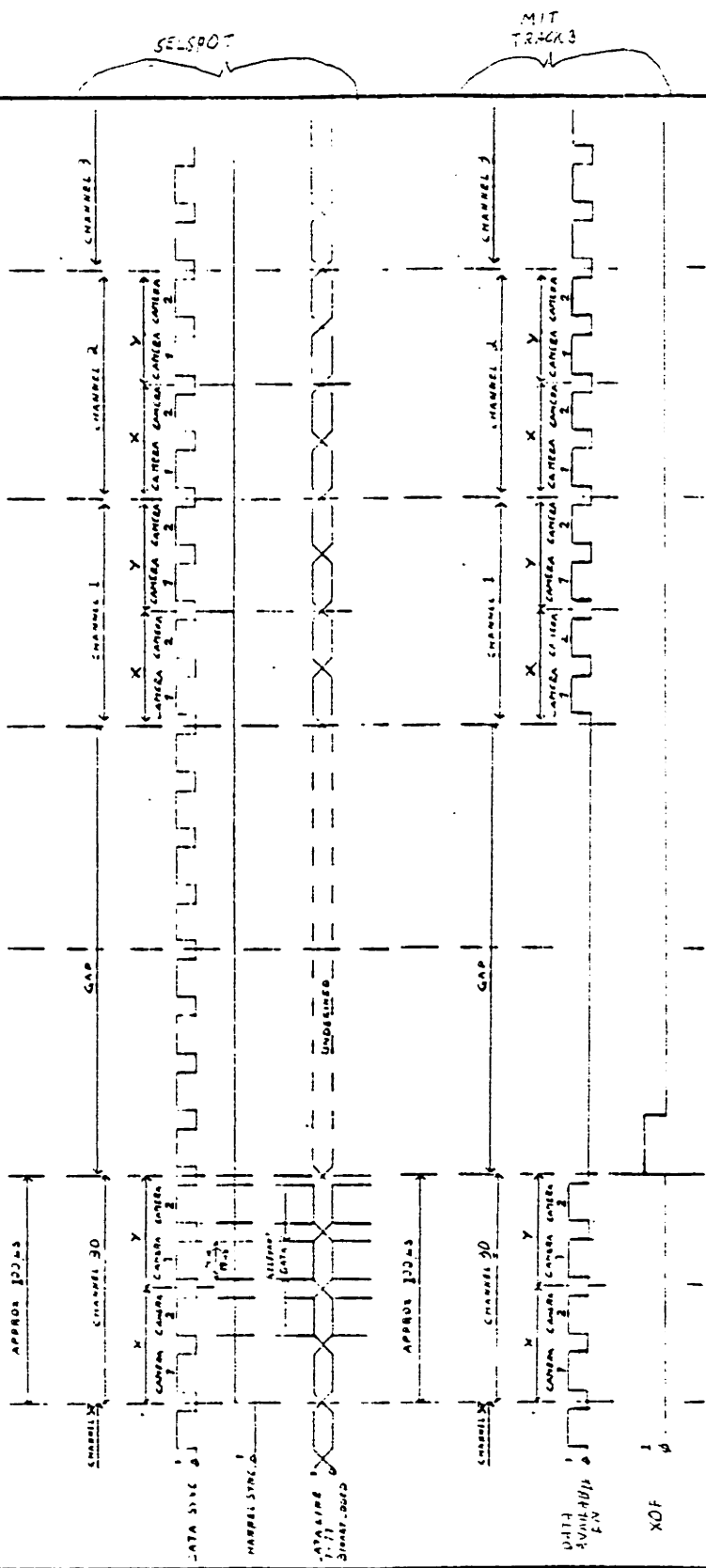
SELSPOT SYSTEM
TIMING DIAGRAM

SELCOM AB
SWEDEN

4000-04
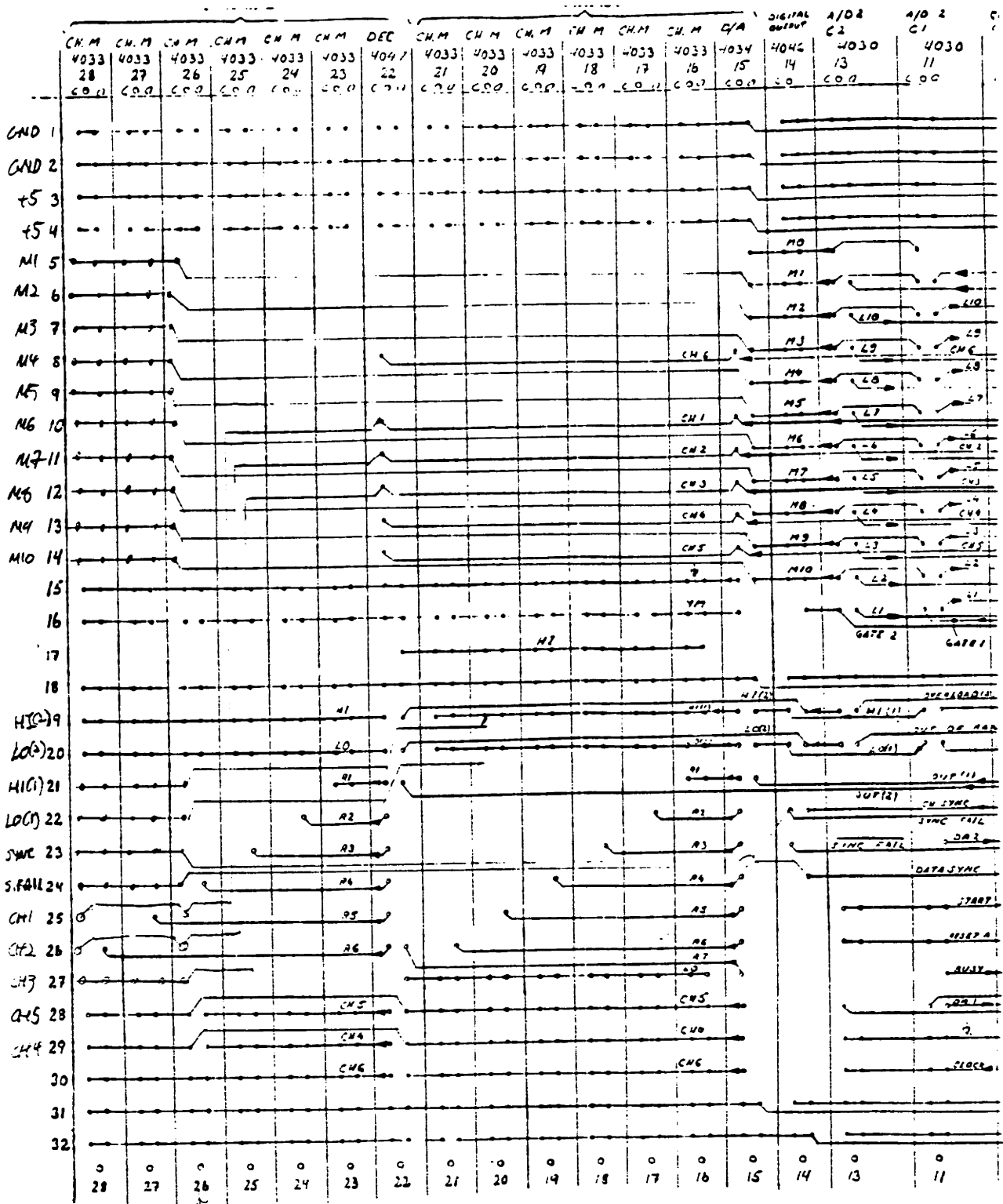
Computer
Interface Slots

CI · CAMERA I
C2 · CAMERA 2

CH M · CHANNEL MODULE
DEC · DECODER MODULE

LED CONTROL UNIT (LCU) MANIFOLD

All mini-phono jacks have a common shield ground

30	29	28	27	26	25	24	23	22	21
20	19	18	17	16	15	14	13	12	11
10	9	8	7	6	5	4	3	2	1

Pattern of LED channel numbers on the LCU, viewed from the manifold end.

APPENDIX D

SELSPOT-COMPUTER INTERFACE

MASSACHUSETTS INSTITUTE OF TECHNOLOGY   *CAMBRIDGE, MASSACHUSETTS 02139*

HARVARD-M.I.T. REHABILITATION ENGINEERING CENTER
Building 31-063

June 12, 1980

Memorandum
To: File
From: George F. Dalrymple
Subject: Selspot/PDP11-60 DMA Interface

The interface makes possible rapid acquisition of data from the Selspot system. It is capable of selecting data from 1 to 30 LEDs, and from every Selspot frame to one frame out of 256 frames.

The interface is constructed on a MDB 11B general purpose DMA controller. The wire wrap area of the board contains: (1) circuits to synchronize the interface to both the Selspot and the PDP-11; (2) a frame counter that enables transfer from one frame from N frames, where N is software selectable; (3) a data word or LED counter which produces 4M transfers per enabled frame, where M is software controllable; (4) an interface control logic to interrupt the system when the transfer is complete; and (5) a control section which allows the user to start and stop the data transfer via push buttons or software control.

The interface output data register is used as the control register. Bits 0-7 contain the frame division number, bits 8-12 the number of LEDs for the LED counter, and bits 13-14 are decoded for reset and enable signals and for control of the panel lamps on the control box. Bit 15 is the interrupt enable bit.

The address jumpers are selected for a base address of 772410_8 and a Bus request level of 5. The interrupt vector address is 240_8. A jumper to C1 control from DMAADR makes the unit an input device only.

Frame Counter. The frame counter consists of two 74161s (C-1, D-1), cascaded synchronously with the terminal count (TC) output used to reload the counter and enable the LED counter. When all outputs are high, the TC signal is high. In this circuit it remains high for one frame period. The counter is loaded on the leading edge of the clock pulse (pin 2) when the Parallel Enable ($\overline{PE}$; Pin 9) is low, i.e. when TC is high, regardless of the state of the count enable signals.

The counter is loaded with the two's complement of the frame division number. The counter is enabled by a signal synchronized with the frame synch pulse (XOF) from the Selspot and disabled immediately upon Word Count Zero signal.

LED Counter. The LED counter provides a gate signal to control the various DMA control signals and consists of two 7461 cascaded synchronous counters (E-2, F-2). Bits 0 and 1 are always loaded with zeros to multiply the count by four. Bits 2-7 are loaded with the two's complement of the number of LEDs. Bit 8 is always loaded high and the MSB is used as a gate pulse to enable DMA transfers.

The clock for the LED counter is the Data Available signal. The load signal ($\overline{PE}$) is the complement signal of the load flip flop, a D-type FF. The Load FF is set by the frame pulse and set low by the clock for the LED counter. The LED counter is loaded by each frame pulse, regardless of whether or not it will count during that frame.

Synchronizer. The signals for controlling the counters and bus control logic are synchronized by a D-type FF (B-2) and a monostable (B-1). The frame synch pulse is derived from the trailing edge of the XOF pulse from the Selspot. The trailing edge is used to maximize the time available between frames for servicing the interface. A 74121 monostable, set at approximately 1us, generates the frame synch pulse. The leading edge of the pulse turns on the initiate FF if the Word Count Zero signal and pin 4 of the decoder (H-2) are low. The initiate FF enables the interface, i.e. BUSMSTR FF, Frame Counter and output gates. The trailing edge of the synch pulse clocks the Frame Counter and sets the LED Counter Load FF. When the Word Count Zero goes high, the initiate FF is immediately reset.

The BUSMSTR signal is generated by ANDing Bit 8 of the LED counter, the TC signal from the Frame Counter, Data Available from the Selspot, and the output of the BUSMSTR FF. The BUSMSTR FF is set by every Data Available pulse if the initiate line is high and is reset by the BBSY signal from the computer.

The Bus Address Increment, Word Count Increment and Input Data Register Load signals are derived from a similar gate using the first three signals for the BUSMSTR signal, except the 4 gate signal is the initiate signal.

Control. The control section provides externally-controlled flags for system control, status panel lamps and the run signal.

The "Start" and "Stop" pushbuttons set their respective FFs (H-1) which control bits CS01 and CS00. These FFs are cleared by lows on DATO 13 and 14.

The RUN signal and RUN light are asserted by highs on DATO 13 and 14. Should the system be run without the control panel, the RUN signal has an independent pull-up resistor.

Interrupt Control. The device interrupt signal is asserted when the word count goes to zero if Bit DATO 15 is asserted. It is reset by the BBSY signal.

GFD:lsw

copies: Erik Antonsson
         Prof. R.W. Mann
         S. Tachi
         Prof. D. Rowell

Data Out Register

Bit	Function
DATO 0-7	2 complement
	Frame Division
DATO 8-12	2 complement of no. of LEDs

	Reset Latch	Ready Lamp	Off Lamp	Initiate and on Lamp
DATO 13	0	1	0	1
DATO 14	0	0	1	1
DATO 15	Interrupt Enable			

Control and Status Register

Bit	Function
CS0	On Flag
CS1	Off Flag
CS7	Done

Register Address

Address	Register
772410_8	Word Count
772412_8	Bus Address
772414_8	Status
772416_8	Data I/O

June 12, 1980

74161
E-2
P3-16
P3-14   DATO 12
P3-33   DATO 11
P3-45   DATO 10

74161
E 2
P3-39   DATO 09
P3-27   DATO 08
82 6
G

BUSMSTR
F1
P3-1
BR INC
P3-11
WC INC
P2-13
10 PLD
F1
P2-301

74161
D2
P3-29   DATO 07
P3-28   DATO 06
P3-25   DATO 05
P3-7   DATO 04

74161
C2
P3-6   DATO 03
P3-26   DATO 02
P3-5   DATO 01
P3-47   DATO 00

XOF
P2-14

High
MIR
P2-41

74
91
B2
C1
74
4
B1
Q
1K   +5

DATA
AVAILABLE
P2-16

INITIATE
High
BBSY

CUT GROUNDS P2-14,16
INVERTERS PAS 2.1

SHEET 1 OF 3
PAGE 11B
SEL SEQ / PDP 11/EC INTERFACE
24 JAN 80

TOLERANCES
DECIMAL
FRACTIONAL
ANGULAR

REVISION RECORD

Cut Ground P3-8

- 483 -

IDR15   CS15
P2--6   P2-41

IDR14   CS14
P2--4   P2-17

IDR13   CS13
P2-7   P2-3-

IDR12   CS12
P2-45   P2-39

IDR11   CS11
P2-36   P2-50

IDR10   CS10
P2-32   P2-42

IDR09
P2-35

IRO08
P2-31

IDR07
P2-28

IDR06
P2-7

IDR05
P2-27

IDR04
P2-24

IDR03
P2-5

IDR02
P2-6

IDR01
P2-4

IDR00
P2-3

SHEET 3 OF 3
MDB 11B

SELSPOT / PDP 11/60 INTERFACE

DATE 27 JAN 80

SELSPOT DMA INTERFACE NOTES

P3-16 Ground, Etch cut.
F-2 pin 12 connected to P3-16 for
digital trigger timing input for
Selspot synchronized forceplate
frame start pulse.

P3-8 Ground, Etch cut.
D1-9 to E1-3   E1: 7414, Hex inverter
E1-4 to P3-8

P3-8 ON: Signal used to enable forceplate
data conversion when the Selspot is
sampling.

SELSPOT INTERFACE
CARD

50 pin connector

To computer (11/60)

22
+5

19
READY

21
ON

20
DONE

17
STOP

18
START

12
GND

220Ω

5
GND   BARE   BLACK

220Ω

6
START   GREEN   GREEN

7
STOP   RED   WHITE
                    BLACK

330Ω   4
DONE   BLUE   BLUE

330Ω   3
ON   ORANGE   BROWN

330Ω   2
READY   YELLOW   White

1
+5   BLACK   RED

330Ω   330Ω   330Ω

9 PIN CINCH
CONNECTOR

CONTROL
BOX

8   RED/BLACK

9   SHEILD
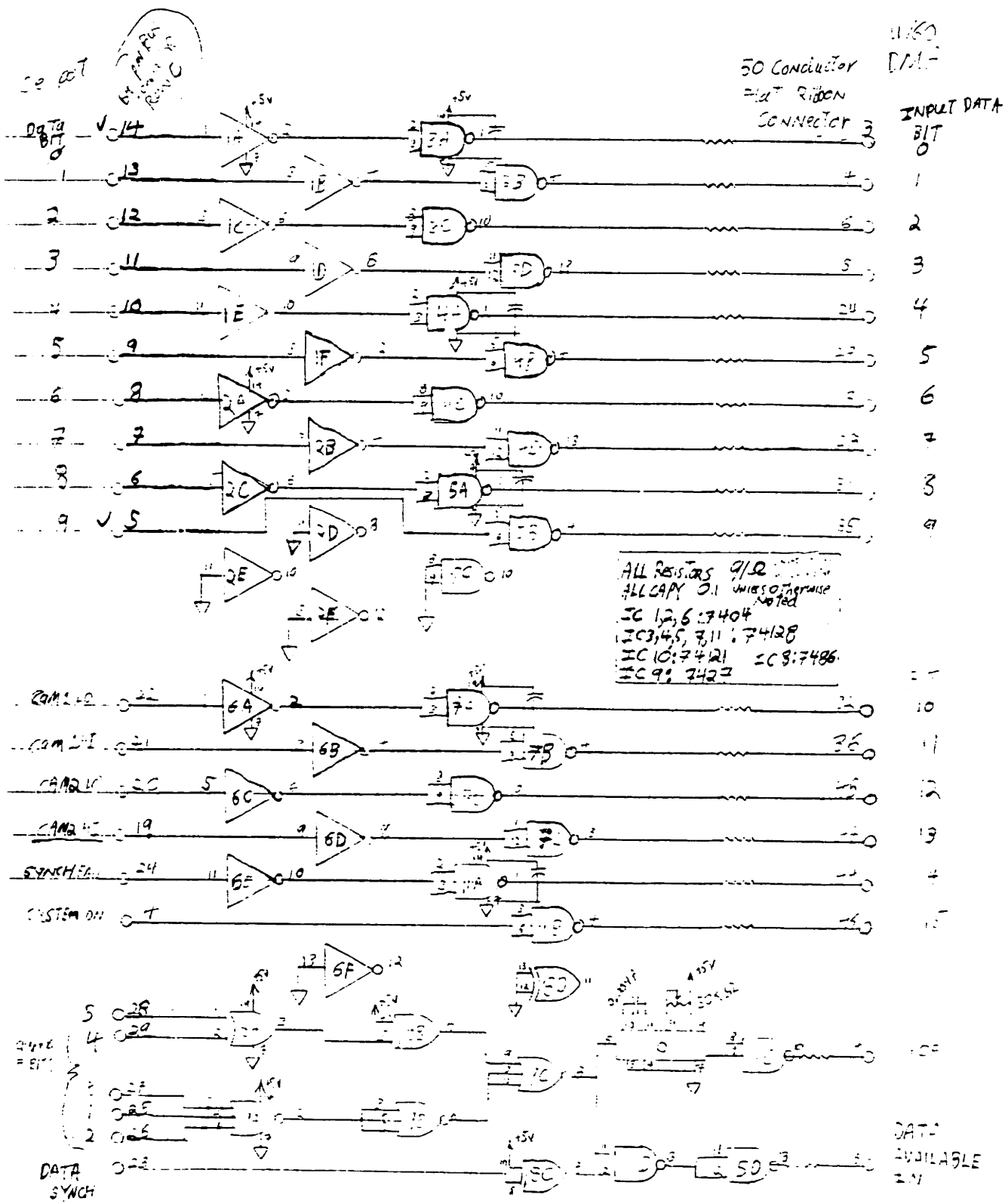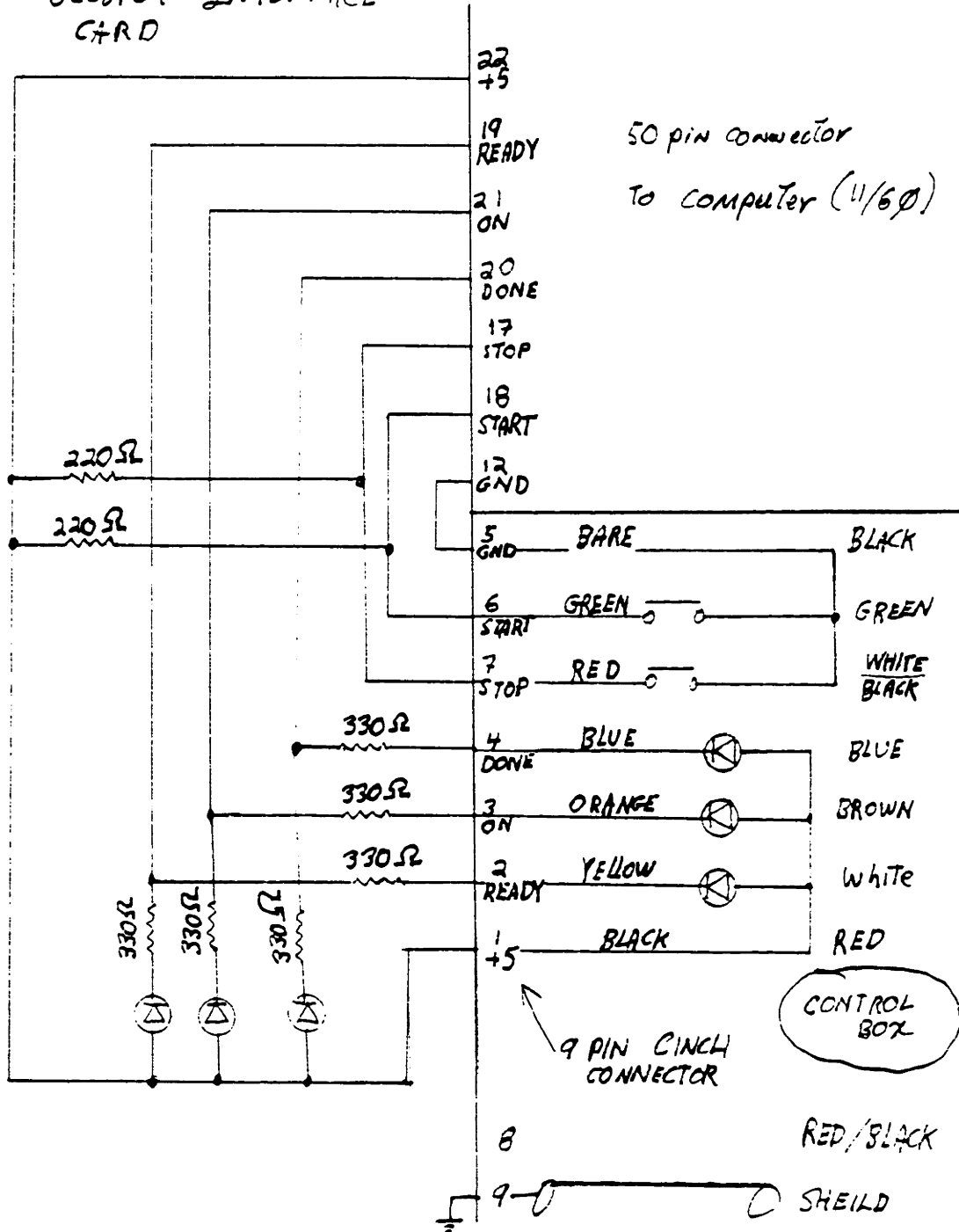
SELSPOT INTERFACE REGISTER BIT ASSIGNMENTS

DRWC: 772410

```
 _____
|                          :                     |
|_____:_____|
```
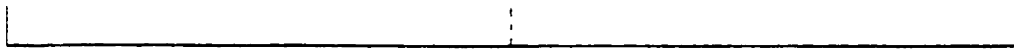
2's Complement of the number of words to transfer in DMA
mode: 16 bits

DRBA: 772412

```
 __ __ _____
|Ø |1 |                    :                       |
|__|__|_____:_____|
```

Starting Buffer address for storage of DMA transferred
words: 18 bits, the top 2 (Ø,1) are hardwired at the
interface and not selectable under program control

DRST: 772414

```
 _____ __ _____ __ __
|                              |Ø7|                     |Ø1|ØØ|
|_____|__|_____|__|__|
```
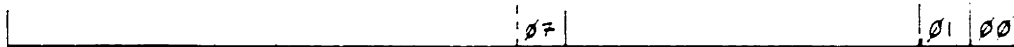
    Ø7: READY: Set to indicate that DMA transfers
        are DONE (Word count has incremented to zero)

    Ø1: Set when the STOP switch is closed
    ØØ: Set when the START switch is closed
        (Ø1 and ØØ are reset when bits 14 and 13
         in DRDB are cleared)

DRDB: 772416

| 15 | 14 | 13 | 12 ←———— NCHN ————→ 08 | 07 ←———— NSKIP ————→ 00 |
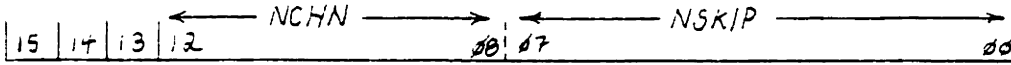
15: INTERRUPT ENABLE

14:	13:	
0	0	RESET
0	1	"READY" Light ON
1	0	"STOP" Light ON
1	1	"ON" Light ON, and Enable DMA transfers

08:-12:   NCHN: 2's Complement of the number of LEDs
00:-07:   NSKIP: 2's Complement of:
            [(Number of frames to skip between frames
               of data stored)+1]

DATA WORDS FROM THE SELSPOT

SYS ON	SYNCH FAIL	CAMERA +I	-0	+I	-0			←———— DATA ————→	
15	14	13	12	11	10	09	08	07	00
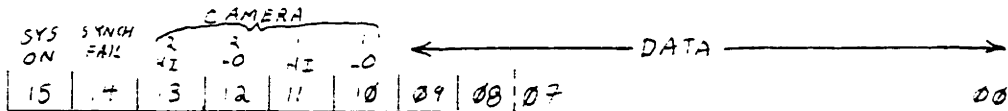
APPENDIX E

KLINGER SCIENTIFIC OPTICAL BENCH
SPECIFICATIONS

110-20 Jamaica Avenue
Richmond Hill, New York   11418
(212) 846-3700

SYSTEM X 95

A new construction system for optical and precision engineering applications. The most important constructional element is a hollow tubular profile made of aluminum alloy, whose longitudinal ribs provide its extraordinary stiffness. In addition, these ribs are the four guideways around the profile. These guides are also compatible with the carriages of the GN 95 granite benches. Three dimensional constructions are possible by use of suitably designed connecting elements. The optical components are mounted on carriages with the aid of holders and mounting elements.

Advantages: Low weight but stiff hollow profile, easy use, table and rack construction possible; construction of plane parallel optical arrangements also possible. Vibration damping can be supplied.

Applications: Experimental construction of optical arrangements without high requirements with respect to guiding accuracy. Also suitable for three-dimensional or plane constructions, as tables or racks. Sensitive to resonance.
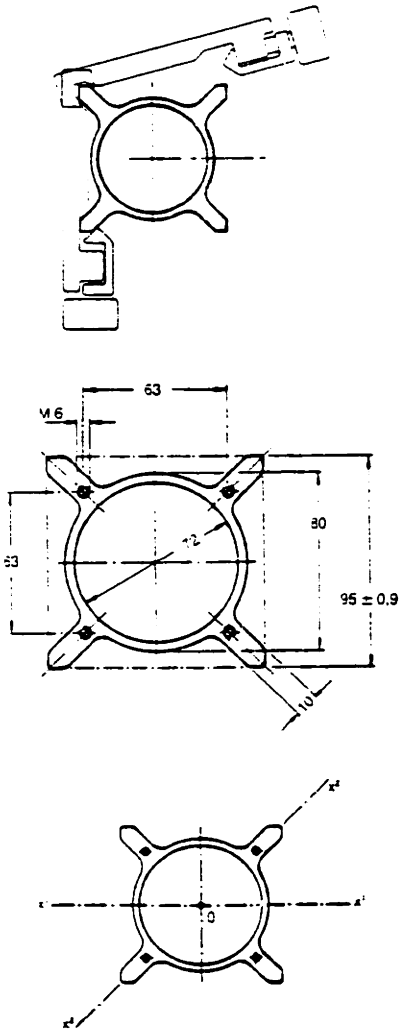
Caracteristics of the X 95 profile

Moment of inertia: (averaged experimental values)
$Jx^1 \approx 215$ cm⁴
$Jx^2 \approx 265$ cm⁴
$J^0 \approx 225$ cm⁴

$$f = \frac{P \cdot L^3}{48 E \cdot J}$$

Modulus of elasticity
$E = 7250$ kg/mm²

Tolerances
Bending: 0.5 mm/m
Twisting: 0.6 mm/m
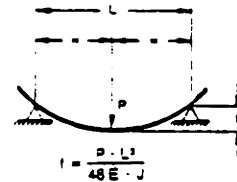
f = Bending
P = Load
L = Profile length
J = Moment of inertia
E = Modulus of elasticity

X 95 Profile

Length (mm)	Weight (kg)	Order No.
X 95– 250	1.4	02 6102
X 95– 500	2.7	02 6105
X 95– 640	3.4	02 6106
X 95– 750	4.1	02 6107
X 95–1000	5.4	02 6110
X 95–1500	8.1	02 6115
X 95–2000	10.8	02 6120
X 95–2000	13.5	02 6125
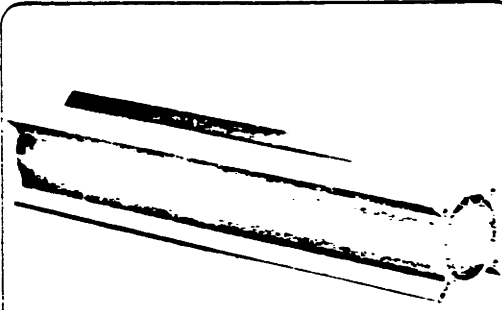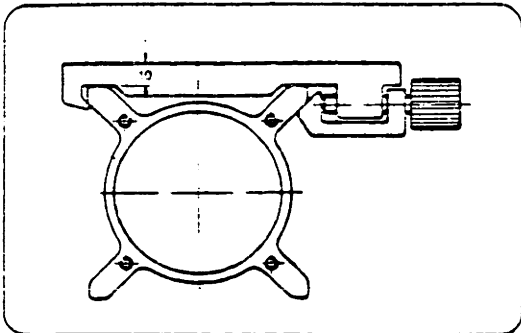
X 95 carriages

The X 95 carriages mate with the guide ribs of the X 95 profile.

Locking is provided by a knurled screw which operates a clamping wedge.

The clamping screw permits the following settings:

- completely loosened: the carriage can be removed from the profile.
- lightly tighten: the carriage slides smoothly over the whole length of the profile and the clamping wedge is held in position by spring pressure.
- tightened: the carriage is locked in the required position.

Clamping by the clamping wedge is carried out on the lower side of the guide rib, which means that damage to the supporting surface is impossible.

The sliding surfaces of the carriages are coated with teflon to provide easy motion and to prevent damage to the surfaces of the guide ribs.
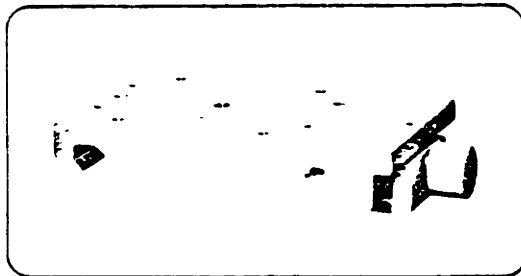
	Width (mm)	Weight (kg)	Order No.
Carriage X 95–50		0.4	02 6401
Carriage X 95–80		0.6	02 6402
Carriage X 95–120		0.8	02 6403

Hole pattern on carriages
X 95–50, –80, –120
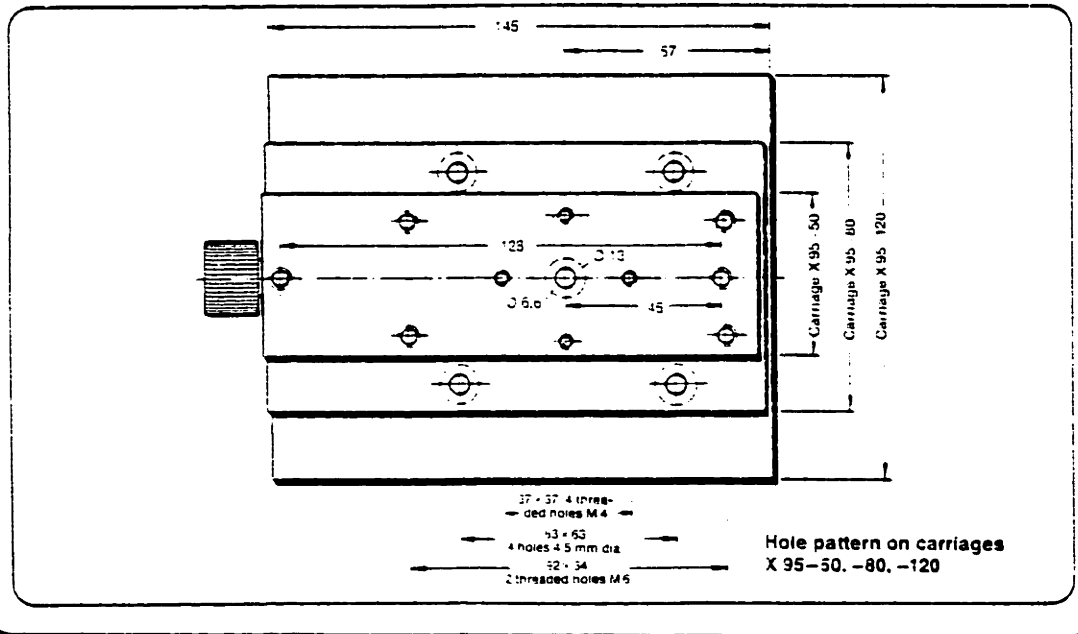
accuracy and mechanical systems

The mechanical accuracy is the difference between a theoretical value and a real dimensional value.

In the case of micropositioning the dimension measured is the distance between two successive positions of the same object. The admissible positioning tolerance can be represented by a sphere (depicted below). In practice we can only measure the projection of the difference from a coordinate reference. In other words we must consider tolerance represented as a cube rather than only a sphere.

$\pm T_x$

radius T

Ao origin

sphere within which the actual point A₁ must lie
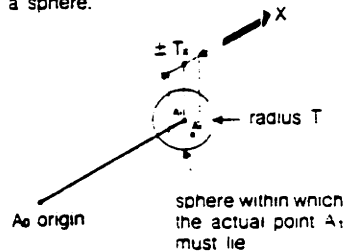
|Ao A₁| = distance (measured dimension)|

The tolerance can be expressed as absolute value and/or relative value to the travel range.
Example:
Tolerance on X axis = ± Tx
where $Tx = 10^{-4} |A_0 A_1| + 0.01$ mm
(this is common value for 100 mm) travel range.

The total tolerances are a function of many randomly distributed parameters. In addition, our parts are manufactured in large production runs. Therefore, these two factors result in the higher probability of being within the sphere than on the peripheral.

Within a normally distributed sample a point A₁ randomly chosen will yield a standard deviation of:

$$\sigma x = T \frac{\sqrt{3}}{6}$$

Example:
For travel ranges of 100 - 200 mm most of the mechanical systems will guarantee a tolerance of ± 0.1 mm at 1 σ. To obtain a standard deviation of 0.01 mm it becomes necessary to utilize precision micropositioners in the realm of 1 μm or better and require an in depth understanding of the proposed application. The resultant accuracy is a combination of the number of parameters including stability, repeatability ... etc.

stabilty

The relative position of the mechanical parts is determined by the interaction of various constraints. This can normally be considered as an elastic equilibrium phenomena. In a stable environment the position repeatability will be ideal with the exception of minimal drift.

The exhibited drift is caused by factors of unequal importance. They include:

— Wear and tear: Several hundred hours of operation can create worn parts resulting in a few microns of drift. This operating time is a function of the supported load on the stage and the speed of movement.

— Lubricants: The thickness of the lubricant film on our stages varies with the contact pressure and the viscosity of the product. This thickness can vary by a few microns whether the stage is in operation or stopped, loaded or not. The use of either very fluid or solid lubricants to avoid this phenomena is often incompatible with other conditions of operation such as speed, loading, cleanliness, etc.

— Thermal equilibrium: All the major components of our stages are constructed from steel. At 20°C the differential of the coefficient of thermal expansion is approximately 10^{-5} per C°. Thus variations in the ambient temperature have minimal effect on the mechanical components. One should note however, the effect of sporatic localized heating can yield somewhat different results and therefore, time should be allowed for stabilization of the new thermal equilibrium.

Examples:
Thermal deviation between the carriage and body of an MT50.16 translation stage measured when the stepping motor is powered.

not powered ►◄ powered ◄— —► waiting time 5 hours

disturbed field ●—————● ◄— stable field
drift = 5 μm/hr   drift = 3 μm/7 hrs.

Measurement made by Michelson interferometer 1 fringe = λ/2 = 0.3164 μm.

 time

20 μm

$\theta = 20°C$

1/4 hour
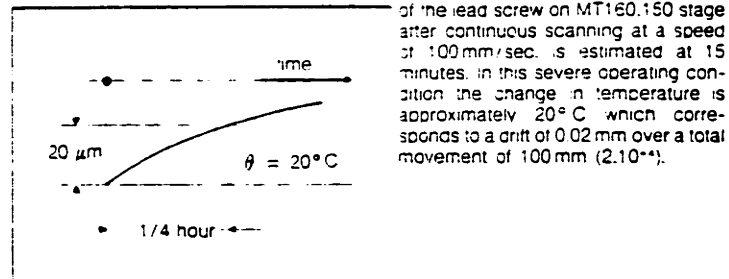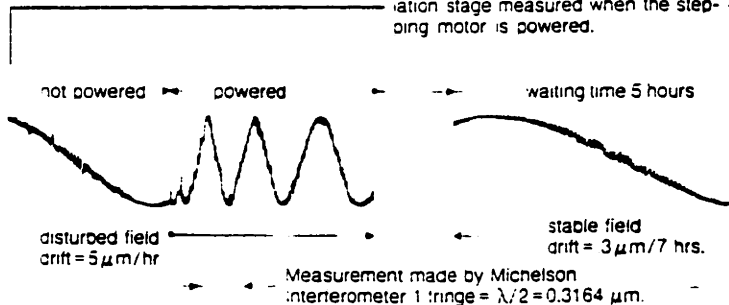
The time for temperature stabilization of the lead screw on MT160.150 stage after continuous scanning at a speed of 100 mm/sec. is estimated at 15 minutes. In this severe operating condition the change in temperature is approximately 20° C which corresponds to a drift of 0.02 mm over a total movement of 100 mm (2.10^{-4}).

translation stages type MR

MR stages consist of a carriage sliding within a U-shaped frame. A micrometer mount assisted by a pair of springs provides accurate positioning. The carriage movement has zero play insured by the ball bearing slides employed. These stages are specified by the letters MR followed by two groups of numbers indicating: the first, the module size, the second, the range of translation. All stages belonging to the same module have identical carriages with the same load capacity and use the same mounting pattern.

Displacement:

Under constant loading approximates a straight line to within 1 μ m.

trajectory of a MR table
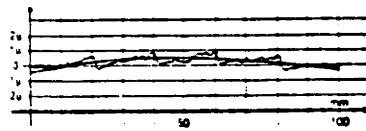
Movement control and accuracy:

Play and backlash: zero
Sensitivity: 1 μ m
Absolute accuracy:

displacement x $10^{-4} \pm$ 1 μ m

accuracy of MR micrometers
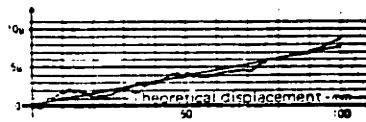
Load carrying capacity:

The load carrying capacity is designed for the center of the carriage (C_z). For canter-leavered loading situations in the accompanying drawing we have:

$C = Q(1+d/a)$ with:

Q = actual load (kg)
d = distance to the carriage center (mm)
a = a constant defined by the modular size and related to the dimensions of the ball bearing track.

The accuracy of the carriage is guaranteed provided that:

$C \leqslant C_z$
$Q_x \leqslant \pm C_x$ •

• + C_x: the load capacity of the micrometer mount
  − C_x: restoring force of the return springs
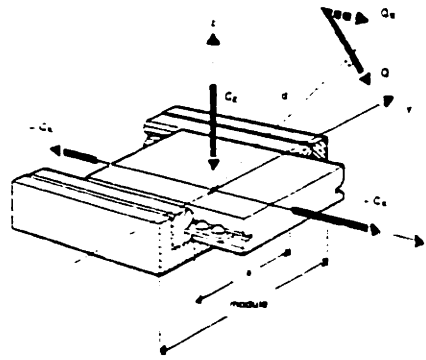
	C_z (kg)	+ C_x (kg)	− C_x (kg)	a (mm)
MR 32	20	5	0.15	12
MR 50	40	12	0.6	30
MR 90	100	20	2	45
MR 120	200	45	3	70
MR 160	300	80	5	90

specifications for type MR stages

MR stages consist of a carriage sliding within a U-shaped frame. A micrometer mount assisted by a pair of springs provides accurate positioning. The carriage movement has zero play insured by the ball bearing slides employed. These stages are specified by the letters MR followed by two groups of numbers indicating: first, the module size, second, the range of translation. All stages having the same module number have identical carriages with the same load capacity and utilize identical mounting hole patterns.

accuracy

Average accuracy .... 3 μm
Sensitivity .... 1 μm

The sensitivity specification pertains to stages equipped with a micrometer. Some models using a differential micrometer have a sensitivity of 0.1 μm.

Readability .... 1 div. = 10 μm
Straightness of trajectory
  Pitch αy .... 10⁻⁴ rad.
  Yaw αz .... 10⁻⁴ rad

load capacity
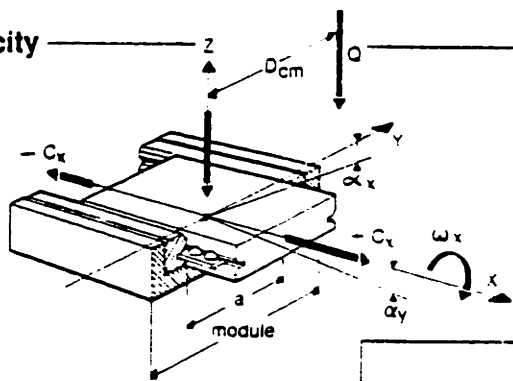
$C = Q(1 + \frac{D}{a})$ must be $\leq Cz$, with:

Cz = Nominal load capacity equivalent to the vertical load capacity
C = Load capacity
Q = Actual load
D = Distance from center of carriage
a = A constant defined by the module size and directly related to the width of the ball guidage track

	module					
	32	50	80	120	160	
Cz	5	10	25	50	75	kgf
a	1.2	3	4.5	7	9	cm
Transversal deviation (roll) O.D. = cmN. αx =	5	3	0.1	0.03	0.01	10⁻⁶ rad.
Longitudinal deviation (pitch) O.D. = 1cmN. αy =	15	9	0.2	0.1	0.03	10⁻⁶ rad.
Load capacity + Cx direction	5	12	20	45	80	kgf
Load capacity – Cx direction	1	1	3	4	4	kgf
Flexion torque: ω x width - Cx = 1 kgf	0.4	0.5	2	4	8	cmN

- -

standard translation stage type MR.

holes are accessible only at end of carriage range

 tapped holes $\Oslash$ V (ISO) ⌀ E

Section A

 tapped holes ⌀ V (ISO) / ⌀ U

 clearance holes for screws ⌀ V / ⌀ E
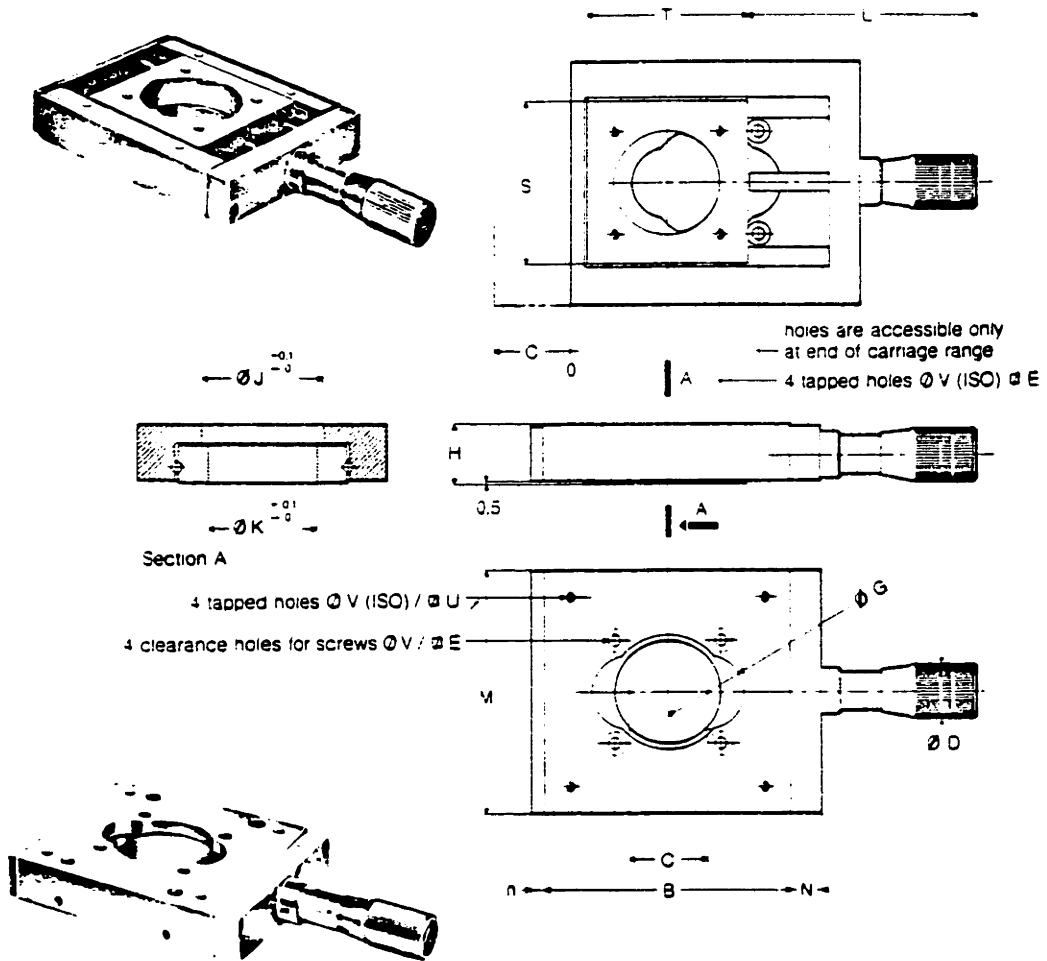
	B	Range	D	E	G	H	J	K	L	M	n	N	S	T	U	V	Weight kg	CODE	PRICE
MR .					•												0.3		.
MR .									.5				.6				0.9		.
MR .																	3.2		.
MR .																	4.2		.
MR .																	9.6		.

NOTE: MR. has a mm hole in frame and a mm hole in the carriage

rotation stages type TR

Type TR rotation stages provide rotational movement about one axis. Essentially their structure consists of two coaxial cylinders, the inner one supporting the upper plate, the outer supporting the base.

knob for disengaging the fine movement

360° rotatable mount with vernier, 1 arc minute reading

fixed part

fine adjustment ± 2° range
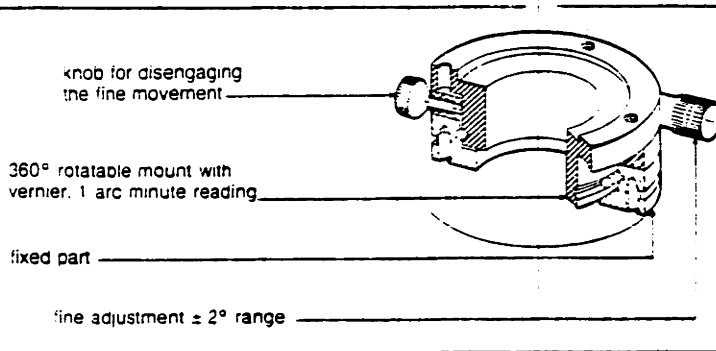
accuracy

	TR 46	TR 80	TR 120	TR 160	
average accuracy	30	1	1	1	arc minutes
sensitivity	5	3	2	1.5	arc seconds
ball race wobble	5	3	2	1.5	10^{-5} radian

load capacity
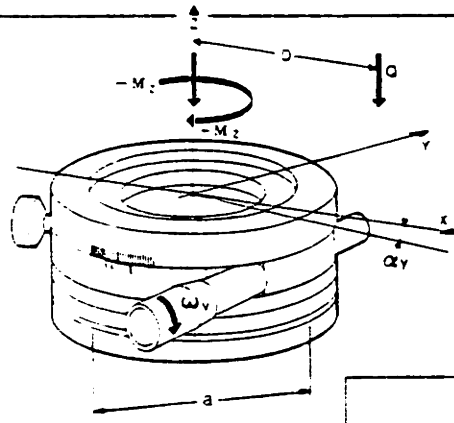
$C = Q (1 + \dfrac{D}{a})$ must be $\leqslant Cz$, with:

Cz = nominal load capacity equivalent to the vertical load capacity
C = load capacity
Q = actual load
D = distance from center of carriage
a = a constant defined by the module size and directly related to the width of the ball guide track

	module				
	46	80	120	160	
Cz = nominal load capacity equivalent to the vertical load capacity	10	20	50	80	DaN
a = a constant	2	4	6	8	cm
Transversal deviation Q.D. = 1 cmN. α =	9.5	0.15	0.08	0.02	10^{-6} rad.
Load capacity clockwise direction. + Mz	0.65	3.6	10	30	mN
Load capacity counterclockwise direction. - Mz	0.30	0.7	1.5	3.5	mN
Driving torque ωz. with a load + Mz = 1cmN	1	1	2	5	cmN

locking of
fine adjustment

TOP VIEW

CODE TR 46   **338 123**
TR 90   **338 124**
TR 90S   **338 324**
TR 120   **338 125**
TR 160   **338 126**
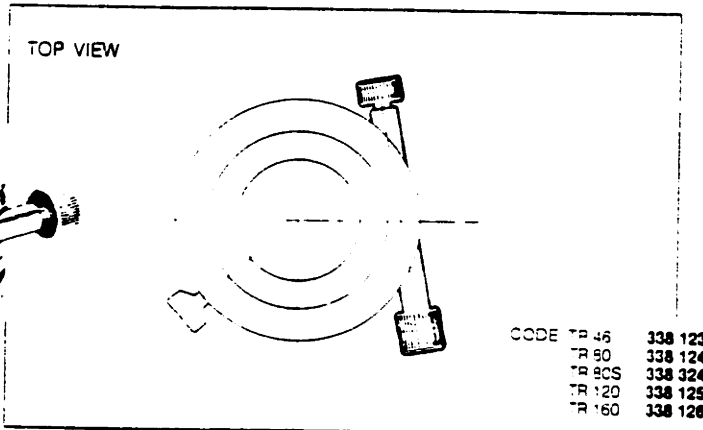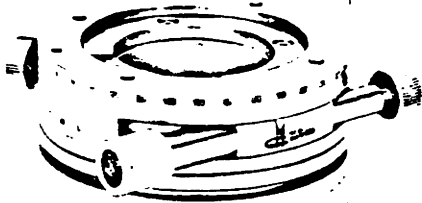
mounting units with circular bases

Hole patterns for mounts with four
bridles correspond to module "U" sizes.

bridles BR

screw Ø V. length L
(not supplied)
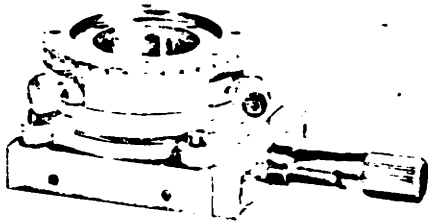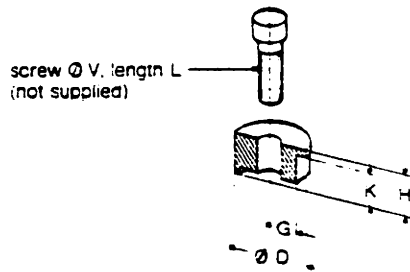
	Module	D	G	H	K	L	V
BR 3	50	9	2.2	5.5	5	10	3
BR 4	90	13	4.5	7.5	6	12	4
BR 5	120	15	4.5	10.5	8	15	5
BR 6	160	26	10	12.5	10	20	6

CODE BR 3 **338 263**
BR 4 **338 264**
BR 5 **338 265**
BR 6 **338 266**

APPENDIX F

COMMITTEE ON THE USE OF HUMANS AS
EXPERIMENTAL SUBJECTS INFORMED
CONSENT FORMS

INFORMED CONSENT DOCUMENT
Automatic Spatial Dynamic Human Mobility Analysis

The purpose of this study is to measure the motion during normal walking. Four light-weight pieces of plexiglas will be attached to one of your legs with elastic straps and/or a skin adhesive. It may be necessary to shave the hair from small areas of your skin to insure good attachment. A waist belt with an electronic box (for powering the motion measurement LEDs) will be attached. You will then be asked to walk normally across the lab. For some of the trials you will walk across a force measuring platform imbedded in the floor. The platform is as stiff as the surrounding floor, and is covered with the same floor tiles.

The plexiglas structures have small infra-red Light Emitting Diodes (LEDs) imbedded that are detected by a pair of infra-red sensitive cameras to record motions of your leg. The forceplate will record the forces between your foot and the floor.

At no time will you be subject to any electrical charge, or any external forces. Every effort will be made to allow you to walk naturally and normally.

There is no benefit to you beyond the satisfaction of contributing to the knowledge of human motion.

Throughout the experiments, we welcome and encourage any comments, suggestions or inquiries you have concerning the the experiments. Should you wish to remain anonymous, we will arrange for that. You are free to withdraw from the experiment at any time.

I understand, that in the event of any injury resulting from the research procedure, medical care is available through the M.I.T. Medical Department. The costs of that care will be borne by my own health insurance or other personal resources. Information about the resources available at the M.I.T. Medical Department is available from Laurance Bischoff at 253-1774.

There is no other compensation, financial or insurance, furnished to research subjects merely because they are research subjects. Further information may be obtained by calling Kimball Valentine at 253-2882.

I have read the above and agree to participate in this experiment.

Signature _____

Dated _____4- April- 1982_____

INFORMED CONSENT DOCUMENT
Automatic Spatial Dynamic Human Mobility Analysis

The purpose of this study is to measure the motion during normal walking. Four light-weight pieces of plexiglas will be attached to one of your legs with elastic straps and/or a skin adhesive. It may be necessary to shave the hair from small areas of your skin to insure good attachment. A waist belt with an electronic box (for powering the motion measurement LEDs) will be attached. You will then be asked to walk normally across the lab. For some of the trials you will walk across a force measuring platform imbedded in the floor. The platform is as stiff as the surrounding floor, and is covered with the same floor tiles.

The plexiglas structures have small infra-red Light Emitting Diodes (LEDs) imbedded that are detected by a pair of infra-red sensitive cameras to record motions of your leg. The forceplate will record the forces between your foot and the floor.

At no time will you be subject to any electrical charge, or any external forces. Every effort will be made to allow you to walk naturally and normally.

There is no benefit to you beyond the satisfaction of contributing to the knowledge of human motion.

Throughout the experiments, we welcome and encourage any comments, suggestions or inquiries you have concerning the the experiments. Should you wish to remain anonymous, we will arrange for that. You are free to withdraw from the experiment at any time.

I understand, that in the event of any injury resulting from the research procedure, medical care is available through the M.I.T. Medical Department. The costs of that care will be borne by my own health insurance or other personal resources. Information about the resources available at the M.I.T. Medical Department is available from Laurance Bischoff at 253-1774.

There is no other compensation, financial or insurance, furnished to research subjects merely because they are research subjects. Further information may be obtained by calling Kimball Valentine at 253-2882.

I have read the above and agree to participate in this experiment.

Signature _Thomas Mocinowski_

Dated ___4 APRIL 1982___

APPENDIX G
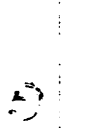
TELEFUNKEN INFRA-RED LIGHT EMITTING DIODE
SPECIFICATIONS

Route 22 - Orr Drive
Somerville, New Jersey   08876
(201) 722-9800

INFRARED DIODES

EMITTING DIODES

TYPE		PACKAGE	ELECTRICAL CHARACTERISTICS							PACKAGE	
			η	Φ_e mW	Φ_e and I_e at I_F mW/sr	I_F mA	t_r ns	t_f ns	λ_p nm	Outline	Dimensions Fig Nr.
V 194 P		TO 18	>60	10	1.6	100	500	600	950		22
CQY 31		TO 18	>40	15	1	100	150	120	900		23
CQY 33N	E / F	TO 18	>40	8	4.5 / 7	100	400	450	950		24
CQY 34N	E / F	TO 18	>12	8	12 / 18	100	400	450	950		25
CQY 35N	E / F	TO 18	>5	8	24 / 36	100	400	450	950		
CQY 32N		TO 18	>5	15	10	100	400	450	900		26
CQX 18	A / B	TO 92	>75	1	0.15-0.3 / 0.25	20	400	450	950		28
CQX 19		TO 39 clear plastic lens	>20	20	40	250	700	830	950		27
CQY 36N		18 mm clear plastic	>40	5	15	50	400	400	950		3
CQY 37/N		18 mm clear plastic	>12	5	4.5	50	400	450	950		2
CQX 46		30 mm clear plastic	>25	15	10	100	400	450	950		

Note: A, B, E, F denote radiant intensity groups.   *At I_F = 1 A

SYMBOL DESIGNATION

A_{VC} — Open loop voltage gain

CTR — Current transfer ratio
(also called coupling factor):
Ratio of output to into in a
photoelectric (optoelectronic)
coupling device

E_A — Standard illumination level defined
by IEC 306-1 and DIN 5033

I_C — Collector current

I_E — Emitter current

I_F — Diode forward current

I_K — Short circuit current of photovoltaic
device under specifies
radiation/illumination condition

I_O — DC output current

I_V — Luminous intensity

I_{ca} — Collector current which flows for a
specified radiation/illumination
condition

I_{on} — Portion of total photoelectric detector
device current which is produced by
the photoelectric effect

I_{ra} — $I_{on} - I_{ro}$

I_{ro} — Reverse dark current which flows
through a photoelectric device
without radiation/illumination

I_{RO} — Current which flows when reverse bias
is applied to a semiconductor junction
without radiation/illumination

I_{ROI} — Laser diode threshold current

klx — 1000 Lux

lux — SI Unit of illumination

mcd — SI Unit of luminous intensity
I_V (mcd = 0.001 cd)

P_V — Power dissipation

P_{ot} — Total power dissipation

t_f — Fall time

t_r — Rise time

t_{off} — Turn-off time

t_{on} — Turn-on time

V_F — Diode forward voltage

V_{is} — Isolation voltage

V_o — Open circuit voltage generated across
terminals of photovoltage device
under radiation/illumination condition

V_o/V_{no} — Signal-to-noise ratio

V_{CC} — Supply voltage

V_R — Reverse voltage

V_{CEO} — Collector to emitter breakdown
voltage with base open

φ — Off-axis angle at which the luminous
intensity is one-half the on-axis
luminous intensity

$\Delta\lambda$ — Wavelength interval between
wavelenghts

η — Efficiency

λ_o — Peak wavelength of sensitivity
or emission

$\lambda_{1/2}$ — Bandwidth at 50% of maximum
sensitivity or emission

Φ_e — Radiant flux

PACKAGE DIMENSIONS

All dimensions in mm
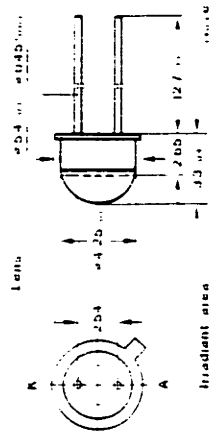
Fig. 22
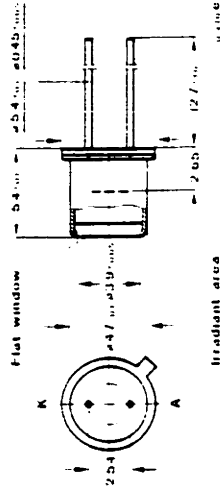
Fig. 23
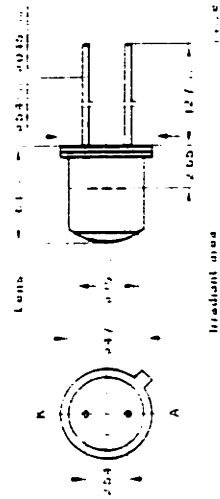
Fig. 24

Fig. 25

APPENDIX H

THE SOFTWARE PACKAGE

The TRACK III software package is large.   It   consists
of several thousand lines of source code in both FORTRAN IV
and PDP-11 Assembly language (MACRO). The vast majority of
that package is not listed here both for reasons of bulk,
and also because of the exclusive licensing agreement
between MIT and Selcom AB of Partille, Sweden to sell the
software, however, several of the more interesting routines
from TRACK III have been included, and as a reference to the
entire package, a list of the program section names and
brief functional descriptions follows, along with a list of
disk data file usage.

TRACK III PROGRAMS   SECTION   (All are FORTRAN except where noted)

TRACK3.CMD	1.0	Batch Control
T3BFZD	2.0	Forceplate Zero collection
GTPHAD.MAC	2.1	Get Physical Address of an array
FPINIT	2.2	Initialize the FP Interface
FPINIM.MAC	2.3	Initialize the FP Interface
T3FPM1.MAC	2.4	Get one frame of FP data
T3BSFA	3.0	Selspot and Forceplate data acquire
T3SSFP	3.1	F4P to MAC arg translation SS and FP
T3SSM3.MAC	3.2	SS and FP DMA synchronous data collect
T3SFTM	3.3	Fortran to Macro argument translation
T3SSM2.MAC	3.4	Selspot Only DMA data collection
T3FBDI	4.0	Selspot Hardware Bad data interrogate
T3BFDT	5.0	FP data translate to canonical form
T3BRSV	6.0	Raw data save
T3BRDP	7.0	Reclaim for proces prev saved raw data
T3SPFC		
T3SPRD		
T3SPPR		
T3BWND	7.5	Data "Window" in Time
T3BCCR	8.0	Selspot camera correction
T3SCCR	8.1	Selspot camera correction

TRACK III PROGRAMS   SECTION   (All are FORTRAN except where noted)

Programs	Section	Description
T3BRFT	9.0	Selspot Raw data l-p filter and interp
T3SFLD	9.1	Low-pass filter design
T3SIFR	9.2	Interpolate and filter
T3SFPI	9.3	First pass interpolator
T3SIIC	9.4	Interpolator Index Corrector
T3SSPI	9.5	Second pass interpolator
T3S2PF	9.6	Two pass filter
T3BFDR	9.8	Forceplate data reduction
T3FFFT	10.0	Forceplate data l-p filter
T3SLPF	10.1	Low-pass filter
T3S2PF		
T3BPCA	11.0	Selspot 3-D point calculate
T3SPIN	11.1	3-D process initialize
T3SPCA	11.2	3-D calculation process
T3FBAD	11.5	Prints of bad frame-channels for each LED
T3BPFT	12.0	Selspot 3-D data l-p filter
T3SFLD		
T3SIFR		
T3SFPI		
T3SIIC		
T3SSPI		
T3S2PF		
T3BPSV	12.3	3-D Point data save
T3SPEL	12.5	Selspot bad data eliminate
T3SPEL	12.6	Selspot bad data eliminate

TRACK III PROGRAMS   SECTION   (All are FORTRAN except where noted)

T3READ

T3ROCA	13.0	Orientation calculation
T3SOCA (T3SGSC)	13.1	Orientation calculation
T3RAPF	13.5	Result interpolate and filter (prior to deriv)
T3SFLD		
T3SIFR		
T3SFPI		
T3SIIC		
T3SSPI		
T3S2PF		
T3BACL	14.0	Velocity and acceleration calculation
T3EAFL	14.5	Velocity and acceleration filtering
T3SIPF		
T3S2PF		
T3BDSV	15.0	Result data save

TA DATA DISPLAY SECTION

T3RFGF   16.0   Raw data graph
    T3SPRD
    T3SPPR
    SVGRAF

T3DFGF   17.0   Result data file graph
    T3SDFG   17.1   Result data file graph
    T3SPRD
    T3SPPR
    T3SPPP
    VGRAPH

T3 UTILITIES	SECTION	
T3PFCG	18.0	Parameter change
T3SPFC	18.1	Parameter file change
T3SPRD	18.2	Parameter file read
T3SPPR	18.3	Parameter print
T3PHEX	18.5	Parameter Header Examine
T3SPHD		
T3SPPR		
T3SPPP	18.6	Post Processing Parameter print
T3PSEL	19.0	Print Selspot data
T3PFCP	20.0	Print Forceplate data
T3DDIR	21.0	Print and change data directory
T3LFMP	22.0	IED file manipulate
T3POSN	23.0	Position the Selspot cameras
T3SSM1.MAC	23.1	Selspot single cycle macro routine
STOP	24.0	Routine to ABORT: TRACK3 and AT.

T3 DATA PARTITIONS SECTION

IOPAGE   28.1
FPSCOM   28.2
SSCOM1   28.3

T3 CANONICAL DATA COMMON AND DATA FILE HEADER

PARAM.CMN   29.0

UTILITY SUBROUTINES SECTION

ASCIIP	30.1	Checks a string for only printable ASCII
QUERY	30.2	Asks a Yes or No question
STHING	30.3	Moves a character string into an array
UPPERC	30.4	Changes lowercase ASCII to uppercase
DELETE	31.1	Deletes a specified file
ERRMSG	31.2	Prints an error message from ERRMSG.COM
FIND	31.3	Searches for a file on a disk
FLFIND	31.4	Finds a file on a disk, Prnt error if not there
RENAME	31.5	Renames a file to a new name
T3SDSH	31.6	Searches a T3 directory for a specified entry

FCS disk I/O and directory manipulation routines

CLOSES.MAC	32.01
DELETM.MAC	32.02
FINDM.MAC	32.03
INIFDB.MAC	32.04
OPENM.MAC	32.05
OPENW.MAC	32.06
PARSE.MAC	32.07
READS.MAC	32.08
RENAM.MAC	32.09
WAITS.MAC	32.10
WRITES.MAC	32.11

FREQUENCY DOMAIN ANALYSIS

SSFFT	40.0	Calculates Fourier Transform of Selspot data
TFPFFT	41.0	Calculates Fourier Transform of Forceplate data
FFT.MAC	41.1	Fast Fourier Transform

TRACK III LOGICAL DEVICE ASSIGNMENTS

TK0: (DR0:)   Selspot raw data
TK1: (DL0:)   Forceplate raw data

N.P. TK0: and TK1: MUST be different physical devices
     w/ DIFFERENT CONTROLLERS!

TK2: (DR0:)   Experimental parameters
TK3: (DR2:)   LED files and directory
TK4: (DR0:)   Results, manipulated data, scratch files,
              data directories, saved canonical form raw data.
TK5: (DR1:)   Camera correction matricies
TK7: (DR1:)   Track III Tasks

2 CHARACTER MONTH ABBREVIATIONS

```
JA    January
FE    February
MR    March
AP    April
MY    May
JE    June
JL    July
AG    August
SP    September
OC    October
NV    November
DC    December
```

The standard data file names are of the form:

MNDY

Where:

MN is a 2 character abbreviation for the current month
DY is the day of the month
   is the number of the experiment on that day (00 to 99)

Thus JA1205 is the 5th experiment on the 12th of January.

*** TRACK III FILE USAGE ***

TRACK III PROGRAMS	SECTION	IN	OUT	MODIFY
TRACK3.CMD	1.0			
T3PFZD	2.0			TK2:T3BPRF.PRF;1
T3BSFA T3SSM3.MAC	3.0 3.2	TK2:T3BPRF.PRF;1		
T3SSM2.MAC	3.4		TK0:SLSPOT.DAT TK1:FCEPLT.DAT TK0:SLSPOT.DAT	
T3BEDI	4.0	TK2:T3BPRF.PRF;1 TK0:SLSPOT.DAT	TK4:T3B000.DAT TK4:T3B001.DAT	
T3BFDI	5.0	TK2:T3BPRF.PRF;1 TK1:FCEPLT.DAT	TK4:T3BF00.DAT TK4:T3BF01.DAT	
T3PRSV	6.0	TK4:T3B000.DAT TK4:T3BF00.DAT	TK4:namfil.RAW TK4:namfil.FPR	TK4:namdir.DIR
T3BHDP	7.0	TK4:namfil.RAW TK4:namfil.FPR	TK4:T3B001.DAT TK4:T3BF01.DAT	
T3EWND	7.5	TK4:T3B001.DAT;n TK4:T3BF01.DAT;n	TK4:T3B001.DAT;n+1 TK4:T3BF01.DAT;n+1	
T3BCCR	8.0	TK5:ERRMTX.IN1 TK5:ERRMTY.IN1 TK5:ERRMTX.IN2 TK5:ERRMTY.IN2		TK4:T3B001.DAT
T3BRFT	9.0		TK4:INTFLT. CR	TK4:T3B001.DAT
T3BFDR	9.8	TK4:T3BF01.DAT	TK4:T3BF02.DAT	

TRACK III PROGRAMS	SECTION	IN	OUT	MODIFY
T3EFFT	10.0		TK4:INTFLT. CR	TK4:T3BF02.DAT
T3BPCA	11.0	TK4:T3B001.DAT	TK4:T3BF02.DAT	
T3BBAD	11.5	TK4:T3B002.DAT		
T3BPFT	12.0		TK4:INTFLT. CR	TK4:T3B002.DAT
T3BRPSV	12.3	TK4:T3B002.DAT TK4:T3BF02.DAT	TK4:namfil.3DP TK4:namfil.FPD	TK4:namdir.DIR
T3BPEL	12.5	TK4:T3F002.DAT		TK4:T3B002.DAT
T3BEAD	11.5	TK4:T3B002.DAT		
T3BOCA	13.0	TK4:T3B002.DAT	TK4:T3B003.DAT	
T3BAPF	13.5		TK4:INTFLT.SCR	TK4:T3B003.DAT
T3BACL	14.0	TK4:T3B003.DAT	TK4:T3B003.ACL	
T3BAFL	14.5		TK4:INTFLT. CR	TK4:T3B003.ACL
T3BDSV	15.0	TK4:T3B003.DAT TK4:T3B003.ACL TK4:T3BF02.DAT	TK4:namfil.DAT TK4:namfil.ACL TK4:namfil.FPD	TK4:namdir.DIR

T3 DATA DISPLAY   SECTION   IN   OUT   MODIFY

		IN	OUT
T3REGF	16.0	TK4:randir.DIR TK4:namfil.RAW	GH0:PLOTTR.DAT GH0:PLOTTR.DAT
T3DEGF	17.0	TK4:namdir.DIR TK4:namfil.DAT TK4:namfil.ACL TK4:namfil.FPD	

UTILITIES	SECTION	IN	OUT	MODIFY
T2PFCG	18.0			TK2:T3BPRF.PRF;1
T3PHEX	18.5	TK4:namfil.DAT TK4:namfil.RAW		
T3PSEL	19.0	TK4:T3B000.DAT TK4:T3B001.DAT TK4:T3B002.DAT TK4:T3B003.DAT TK4:namfil.DAT		
T3PFCP	20.0	TK4:T3BF01.DAT TK4:T3EF02.DAT TK4:namfil.FPD		
T3DDIR	21.0			TK4:namdir.DIR
T3LFMP	22.0	TK3:lsfnam.SEG	TK3:lsfnam.SEG	TK3:SEGFIL.DIR
T3POSN	23.0			
STOP	24.0			

Acquisition

Acquisition of data from the Selspot and forceplate systems is the fundamental task of the software and there are two essential aspects that make it work: Direct Memory Access (DMA) and double buffered direct to disk data transfer. The use of the DMA for data transfer from an external device to memory will be covered first.

A DMA allows transfer of data words (in this case 16 bits wide) from an external device directly into main memory without central processor intervention. The user simply gives the DMA a number of words to transfer, a base address of a data buffer to fill with consecutive data words, and a command to start transfers. Each time the external device indicates it has a new data word ready (usually with a control line transition from high (1) to low (0)) the DMA moves that word from the data lines to the next consecutive memory location in the data buffer. Typically once the data buffer is full the DMA will issue an interrupt. This is exactly the mode that all data is transferred from the external devices to the computer in the TRACK III system.

DMA transfer differs from "programmed Input/Output (I/O)" in that programmed I/O requires the central processor to intervene and move each word of data from an interface register into the appropriate memory location. This is an intrinsically slower process.

Several additional features should be noted. Since the Selspot always runs as if it had all 30 LEDs active at 315 Hz a software of hardware selection process must exist if fewer LEDs or fewer frames per second are desired. The latest hardware interface performs a hardware selection of frames and channels (LEDs). See Appendix D for details. Essentially the operator instructs the DMA how many channels in each frame to accumulate and how many frames to skip between frames of data transferred to memory. The greatest advantage of doing this selection (gating) in hardware is that the central processor is totally free during the filling of one buffer.

Filling a buffer with data is an intrinsically "off-line" process. To accumulate and manipulate data in real-time, each frame (or each data word) has to be processed as it is recieved by the computer. None of TRACK

III deals with real-time data processing, because the gait experiments didn't require it. However the power of the system can be seen by the daily use of the Selspot in real-time at a maximum kinematic calculation rate of 100 frames per second for one body segment with 3 LEDs. A sample routine for getting one frame of data at a time included here is T3SSM1.MAC

It is easily seen that in the off-line buffer filling mode accumulation of data with 30 LEDs at 315 Hz will require 37,500 (decimal) words of memory for each second of data desired. For long experiments (especially if the forceplate is also storing data) the memory requirements quickly become prohibitive for a 16 bit computer. For this reason a routine to fill a buffer of data, begin filling a different buffer while simultaneously writing the first full buffer to mass storage (disk), and continue to "swap" buffers in this manner, was written. This is a classic "double buffered" data transfer routine. It is especially effective when using a DMA because the processor is idle while one buffer is filling, thus it can be put to work writing out a previously filled buffer to enable it to be filled again. Fortunately the process of writing a buffer of data to disk is also a DMA process (out instead of in) and the central processor is relatively free during that process too.

One condition must exist to enable the double buffered DMA-to-memory-to-disk process to work: the writing of a buffer must take less time than the filling of a buffer. If this condition is not met. The first buffer would not be availble to be filled again after the second buffer had finished filling. Since we know that the Selspot produces 37,500 words per second, how fast can the data be written to disk? For buffers 7680 (decimal) words long, data can be written to disk at an average rate of just over 100,000 (decimal) words per second using the assembly language File Control Services (FCS) directives. (This discussion of data transfer software is relatively general, however the routines themselves, particularly where interaction with data on a disk is concerned is very operating system dependent. In this case the computer is a PDP 11/60 and the operating system is RSX11-M Version 3.2 Patch D). Therefore there is a margin of more than 2 to 1 in speeds which should enable the double buffered process to work. Why are the buffers 7680? There are two reasons. The first is that large buffers can be written to disk with a higher average speed than small buffers, because each time a new buffer is begun, a disk latency is encountered. This latency is the time for the disk head to position itself over the correct track on the disk, and then the additional time for the disk to spin around to the correct place for data writing to

begin. Naturally this is not a completely predictive
process, latencies are not always the same amount of time,
and that is why an average disk writing speed is quoted.
The burst speed, once the head is positioned correctly is
much greater. So in general it is beneficial to use the
largest buffers possible.

The second reason that 7860 word long buffers were
chosen is that 7680 is integrally divisible by 256, which is
the length of a disk block. To use the FCS routines whole
disk blocks must be written. 7680 is also integrally
divisible by a very large group of numbers between 1 and 30
allowing a great latitude in the number of LEDs used by the
system and still insuring that a buffer will never complete
filling except at the end of a complete frame of the
Selspot. This is a necessary condition because the DMA
interrupts the system to "swap" buffers (start filling the
other one, and start writing the one just filled) once a
buffer is completely full. The Selspot produces data in 120
word bursts (30 LEDs with 4 words of data for each: X and Y
from each of 2 cameras) with 200 micro seconds between the
bursts. This 200 micro seconds is sufficient time to enable
the computer to service the DMA interrupt and swap the
buffers and never miss the next word of data. If a buffer
finished filling in the middle of a frame of data, there
would not be time (about 14 micro seconds) to interrupt and
swap buffers before the next word was missed. Remember that
the Selspot is a free running device. There is no
handshake, thus there is no way to keep a word of data valid
on the lines until the computer can transfer it. If the
computer is busy while a data word is valid, it is lost.

The process of events during data transfer by the double buffered routine is as follows:

1. Tell the DMA the number of LEDs desired and the number of frames to skip between frames to transfer.

2. Set up the DMA to begin filling buffer number zero (0).

3. Start filling buffer 0

4. When buffer 0 completes filling, an interrupt will be issued, and the central processor will start executing the interrupt service routine. This routine does nothing but to set up the DMA to begin filling the other buffer (not the one just completed filling).

5. Then as the interrupt service routine is exiting it issues an Asynchronous System Trap (AST) which is a software interrupt. The processor then begins executing the AST routine (meanwhile the next buffer has already begun to fill). The AST routine simply queues a request to start writing the buffer that just completed filling.

6. The AST routine exits and the processor returns to a wait loop to wait for the next interrupt and buffer swap. Meanwhile one buffer is being written to disk (by the disk controller DMA) and the other buffer is being filled by the Selspot DMA.

7. After a buffer finishes being written to disk another AST routine is entered to keep a runnig total of the number of disk blocks written. This number is compared with the number of words transferred by the Selspot DMA once the entire transfer operation is complete.

The routine T3SSM2.MAC is an assembly language routine that works in exactly the manner just described.

T3SSM2
FLOW CHART

T3SSM2
EXAMPLE: 242 Frames, with 30 LEDs
29040 words
3 Full buffers: NFBUFS=3
1 Partial buffer: NFBUF1=4
6000. extra words in the
last partial buffer: NEXWRD=6000.
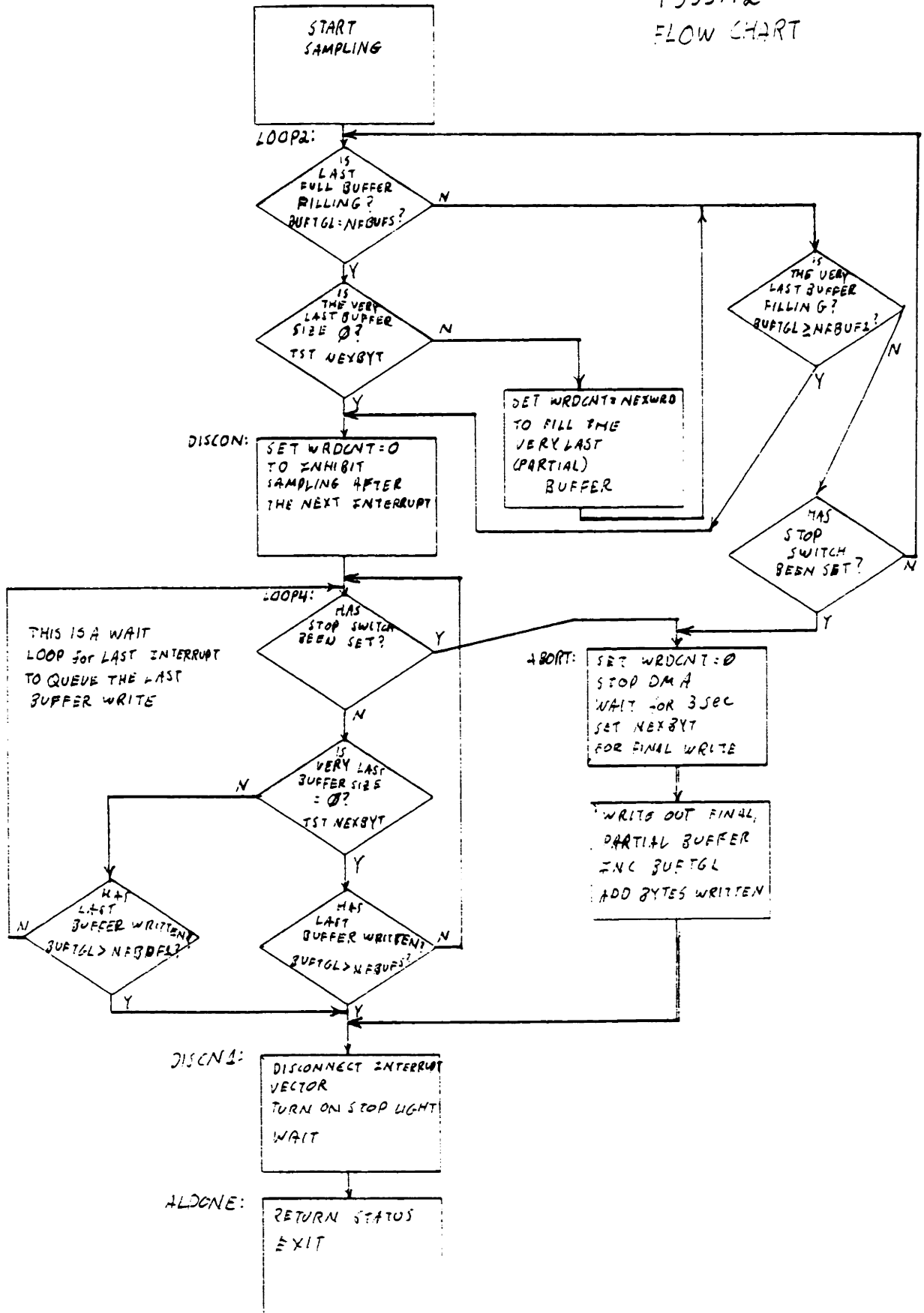
T3SSM2 NOTES:

1) Interrupt everytime DRWC (word count) goes to zero (0)

2) AST routine Queued by the Interrupt Service Routine

Interrupt Service Routine Functions:

1) BUFTGL to set up next buffer to fill

2) WRDCNT to set up DMA

AST Routine Functions:

1) WRDCNT to check if DMA is still active

2) NEXWRD to check if the last buffer should be
   partially written

3) BUFTGL to flag last buffer

4) NFBUF1 to flag last buffer

5) Increment BUFTGL to toggle buffers
   and to count interrupts

The forceplate data acquisition operates in a very similar manner. It uses a separate DMA, however the interface is considerably simpler because the forceplate operation will always transfer all 8 words of data per frame, thus none of the hardware gating is needed. As described in Appendix B there are 3 different ways to strobe the forceplate to get it to send data to its DMA. The forceplate DMA is a standard PDP 11 DR11-B. When the forceplate is used alone, the data transfer process operates in exactly the same manner as for the Selspot. See T3FPM1.MAC as an example of a routine to transfer a single frame of forceplate (FP) data, and see T3FPM2.MAC as an example of double buffered transfer of data from the forceplate to memory to disk using 768 word long buffers. The maximum data through-put rate of this routine with the forceplate and the DAS 1128 conversion system (described in Appendix B) is 2,500 8-word frames of data per second. The 768 word buffers were chosen to be integrally divisible by 256 (again to write full disk blocks).

Several important differences between the forceplate and Selspot interfaces should be noted. First, the Selspot DMA only has the lower 16 bits of its data buffer address selectable under program control. The top two are hard wired. At this time they are set to 01, indicating that the buffers must always be in the second 32 k-word section of memory. The forceplate DMA has all 18 bits of buffer address selectable under program control, thus in some instances, the buffers need not be fixed in memory, but rather can be set up a run-time, and a routine used to find the buffer's base addresses, and these sent to the DMA. The Selspot can only operate in this way in a very limited set of circumstances.

Second, the forceplate is a free-running device, however it does have a handshake, and if the DMA is not ready to recieve a data word, it is held on the lines by the DAS 1128 until the forceplate DMA is ready to recieve it. This enables the forceplate interface to operate at a slightly lower priority than the Selspot DMA, and if the processor is off servicing the Selspot, the forcplate data is not lost. This is also advantageous because the forceplate only produces 8 words of data per frame, compared to a maximum of 120 for the Selspot. The CPU will always find time to take 8 words from the forcplate inbetween the 120 for the Selspot.

Finally, to use the Selspot (SS) and the forceplate (FP) simultaneously and synchronously a fifth routine had to be written. Essentially T3SSM3.MAC is T3SSM2.MAC and T3FPM2.MAC combined. It has 2 interrupt service routines and 2 disk write AST routines, one for each interface. Everything executes asynchronously, with each device's interrupts and AST's occurring whenever their respective buffers are full. Comments that begin with a ;S; are for lines that pertain to the Selspot data transfer, those that begin with ;F; are for the FP.

A final pair of routines are included here. They are the translation routines (written in fortran for simplicity) that take the user desired number of frames and channels and convert them to the number of full buffers and length of final partial buffers, etc. that the double buffered routines need. T3SFTM.FTN is used with T3SSM2.MAC for the Selspot alone, and T3SSFP.FTN is used with T3SSM3.MAC for the SS and the FP together. Notice that those fortran conversion routines also have extensive error reporting from the data transfer routines.

Once again let me stress that the reason these sophisticated double buffered data transfer routines were written was to allow "long" data sets to be accumulated. As little as 10 seconds of data from the Selspot and the forceplate together amounts to almost 1 mega-byte of memory (400,200 [decimal] 16-bit words). A substantial effort was put forth throughout the software package design to eliminate any section that required the entire experiment in memory at one time, to avoid restrictions on experiment length due to computer main memory size. The routines included in this appendix are all programs that usually require all data from one experiment to be memory resident, and have been re-coded to "page" data in and out to enable "long" experiments to be acquired and processed.

Nowhere in the TRACK III software package are there any references to physical devices, logical devices are used throughout. This adds the flexibility of having the data written to any of the disks on the computer, or it allows transfer of the software to a machine with completely different disks. All that need be done is an assignment from the Monitor Console Routine (MCR) before any TRACK III programs are executed.

To ensure that disk latencies to not preclude sampling from both the Selspot and the forceplate simultaneously, the logical devices TK0: and TK1: must be assigned to disk devices with different controllers. TK0: is the logical device that the Selspot data is written to during data acquisition. TK1: is the logical device that the forceplate data is written to. It is not enough to assign them to two different RL01's (for instance) that have the same controller. The through-put to a disk depends on how busy the controller is, and the total through-put to two disks on one controller will be about the same as to one disk. Disks with different controllers can realize higher through-put rates. While the central processor is waiting for one disk transfer, the other can be active.

Different device types are always on different controllers. A pair of RM03's usually has one controller, as does a pair of RL01's, and as long as different device types are assigned to TK0: and TK1: (as opposed to simply different unit numbers) the system will operate correctly.

In general TK0: should be the highest speed device available. The acquisition of Selspot data occurs at the highest rate of all data accumulation in TRACK III, and occasionally a relatively slow disk like an RL01 will not respond quickly enough to maintain data integrity.

```
T3SSM1 -- TRACK III SUBROUTINE SINGLE SAMPLE MACRO ROUTINE
T3SSM1
T3SSM1
T3SSM1

    20-OCT-80      by Erik Antonsson

Copyright (C) 1980  Massachusetts Institute of Technology
                          Cambridge, Massachusetts

This software is furnished under an agreement for use only on a
single computer system and may be copied only with the inclusion
of the above copyright notice.  This software, or any other copies
thereof, may not be provided or otherwise made available to any
other person except for use on such system and to one who subscribes
to these agreement terms.  Title to and ownership of the software
shall at all times remain in MIT.

The information in this document is subject to change without notice
and should not be construed as a commitment by MIT.

MIT assumes no responsibility for the use or reliability of its
software.

        .TITLE  T3SSM1 -- TRACK III SINGLE SAMPLE SUBROUTINE
        .IDENT  /V01A/
        .PSECT  T3SSM1
T3SSM1::CALL    $SAVAL
        CLR     SSDB
        MOV     #-120.,SSWC     ;Transfer 120 words (1 full frame)
        MOV     #FADD,SSBA      ;Base address of SSCOM1 (310500)
        MOV     #061377,SSDB    ;Go bits, 30 chn, and take next frame
LOOP1:  TSTB    SSST            ;Wait for frame to complete
        BPL     LOOP1
        CLR     SSDB            ;Reset the interface
        RETURN
        .END
```

```
T3SSM2  -- INTERRUPT SERVICED SELSPOT SAMPLING WITH DISK DATA STORAGE
T3SSM2
T3SSM2
T3SSM2

        by Erik Antonsson        13-JUN-80

--------------------------------------------------------------------

Copyright (C) 1980  Massachusetts Institute of Technology
                       Cambridge, Massachusetts

This software is furnished under an agreement for use only on a
single computer system and may be copied only with the inclusion
of the above copyright notice.  This software, or any other copies
thereof, may not be provided or otherwise made available to any
other person except for use on such system and to one who subscribes
to these agreement terms.  Title to and ownership of the software
shall at all times remain in MIT.

The information in this document is subject to change without notice
and should not be construed as a commitment by MIT.

MIT assumes no responsibility for the use or reliability of its
software.

--------------------------------------------------------------------

This routine acquires data from the LR11-B DMA (direct memory access
device) and with double buffers writes it out to a disk file.

The FORTRAN CALL is as follows:

    CALL T3SSM2(NCHN,NSKIP,NFBUFS,NLLWRD,NTBLKS,IDSW)
```

Where:

NCHN is the number of active LEDs.
   N.B. The only allowable values for NCHN are:
   1-6,8,10,12,15,16,20,24,32 to ensure that a FRAME
   boundry and disk write block (7680 words) coincide.
NSKIP is the number of frames to skip +1 between sampled frames.
   NSKIP is returned as the SELSPOT/DMA control word (octal).
NFBUFS is the total number of FULL 7680. word buffers to fill.
   NFBUFS is returned with the number of buffers written.
NEXWRD is the number of words in the very last buffer (NFBUFS+1).
   If NEXWRD is non zero, the TOTAL number of buffers filled, either
   fully or partially is NFBUFS+1.
NEXWRD is returned with the write error code if an error occured.
   See pages 5-7 through 5-9, or E-1 through B-3 of the I/O Drivers
   Reference Manual in Volume 5A RSX-11M V3.2.
NTELKS(1) is the TOTAL number of 256 word disk blocks to be written.
   NTELKS(1), and NTELKS(2) are returned with the number of bytes
   actually written (double precision integer).
IDSW is the directive status word returned for error decoding.
   See Appendix B of the Executive Reference Manual in Volume 3B
   RSX-11M V3.2.

SSWC, SSBA, SSST, SSDB are the SELSPOT DMA registers defined in SSDEV

SSWC   172410   transfer word count
SSBA   172412   data buffer bus address register
SSST   172414   status register
SSDB   172416   data register (used for control)

BUF0 and BUF1 physical addresses are retrieved from the base address of the installed common partition SSCOM0 (242500) and SSCOM1 (300500) obtained at VMR. These partitions and installed COMMONs are necessary only because the physical address of the data buffers must be passed to the DMA and this particular DMA has the extended address bits set to 01, necessatating the leading 2 in the address and requiring that the buffers be fixed (more or less) in memory. A fixed partition is the only way to insure this location, and it eliminates the chore of finding the physical base address of the buffers by the program.

It is worth noting that only the first word of these two buffer partitions need be included in COMMON, because the base address of the buffer is all that any DMA device needs to fill/empty a buffer, even if the higher addresses in the buffer aren't in program space. Both the DR11-B acquisition device and the disk are DMA and since the data is only put in memory by the DR11-B to be written out to the disk, these buffers nee not be wholly included in the COMMON.

The extended address bits are hardwired to :01 in the SEISPOT DMA interface thus only the lower 16 bits are moved to SSBA.

This subroutine uses an AST routine queued from the ISR to write the data to disk. The data is stored in disk file: TK0:SLSPOT.DAT using logical unit number (LUN) 14. Note that this LUN is unavailable for use in the calling programs.

The usual calling program should be something like T3SFTM.FTN a program written by Erik Antonsson to simplify the input to this routine without having to do extensive arithmetic manipulations in MACRO.

```
        .TITLE  T3SSM2 -- INTERRUPT SERVICED SELSPOT SAMPLING
        .IDENT  /V01C/
        .MCALL  DIR$,CINT$,ASTX$S
        .MCALL  FDBDF$,FDAT$A,FDRC$A,FDBK$A,FDOP$A
        .MCALL  FDAT$R,OPEN$W,WRITE$,CLOSE$,FDBK$R,WAIT$
M$$MGE  =       0

FDBOUT: FDBDF$                          ;block disk write set up macros
        FDAT$A  R.FIX,,512.,30.
        FDRC$A  FD.RWM
        FDBK$A  .15360.,7.,IOSTAT,WRTAST
        FDOP$A  14.,DATAST,,FO.WRT

T3SSM2::CALL    $SAVAL                  ;save all registers
        MOV     2(R5),LIST+2            ;save argument list
        MOV     4(R5),LIST+4
        MOV     6(R5),LIST+6
        MOV     10(R5),LIST+10
        MOV     12(R5),LIST+12
        MOV     14(R5),LIST+14
        MOV     @2(R5),NCHN             ;get argument values
        MOV     @4(R5),NSKIP
        MOV     @6(R5),NFBUFS
        MOV     @10(R5),NEXWRD
        MOV     @12(R5),NTBLKS
        MOV     NCHN,NBYTES
        ASL     NBYTES
        ASL     NBYTES
        ASL     NBYTES                  ;NCHN*8 = number of bytes in each record
        MOV     NFBUFS,NFBUF1
        TST     NEXWRD                  ;Dont increment NFBUF1 if NEXWRD is zero
        BEQ     CONT@A
        INC     NFBUF1                  ;NFBUF1=NFBUFS+1 to flag last buffer
CONT@A: MOV     NEXWRD,NEXBYT           ;NEXBYT is the # of extra bytes in last buf
        ASL     NEXBYT                  ;(=NEXWRD*2)
```

```
        NEGB    NCHN                    ;these 3 arguments need to be in 2's comp
        NEGB    NSKIP
        NEG     NEXWRD
        MOVB    NSKIP,CNTRL             ;low byte is 2's comp f <frames to skip +1>
        MOVB    NCHN,CNTRL+1            ;next 5 bits: 2's comp of # of active LEDs
        BIS     #16000,CNTRL           ;hi-bit set: interrupt enable,bits 13,14:start
        MOV     #-7600.,WRDCNT          ;normally set up 1st buffer to be a full one
        TST     NFBUFS
        BNE     CONT1
        MOV     NEXWRD,WRDCNT           ;if 0 full buffers, set up 1st to be the last
CONT1:  CLR     BYTES                   ;init double precision running sum of bytes
        CLR     BYTES+2                 ;written to the disk
        FDAT$R  #FDBOUT,,,NBYTES,NTBLKS ;adjust file params to number of blcks
                                        ;to be written, and record size
        OPEN$W  #FDBOUT                 ;open TK0:SLSPOT.DAT
        BCC     AA                      ;error if C bit set
        JMP     CPERR                   ;jump to error handling routine
AA:     MOV     #ISR,C2INT+C.INIS       ;set up interrupt service routine
                                        ;address for CINT$
        MOV     #124,C2INT+C.INVE       ;set up interrupt vector for CINT$
DIR$    #C2INT                          ;execute directive to connect to interrupt
        BCC     AB
        JMP     CONERR                  ;C bit set if directive error
AB:     CLR     SSDB                    ;reset switch registers
        MOV     #20000,SSDB             ;turn on READY status light
        CLR     SSWC                    ;clear SELSPOT DMA word count register
        CLR     SSST                    ;clear the status register
        MOV     #1,BUFTGL               ;set up buffer toggle to write BUF0 out first
        MOV     #1,BFWTGL
        MOV     #1,DONE
LOOP1:  BIT     #2,SSST                 ;Test for STOP switch hit
        BEQ     LOOP1D                  ;If not hit, goto test START switch
        JMP     DISCON                  ;If hit, goto DISCON
LOOP1F: BIT     #1,SSST                 ;test for START switch
        BEQ     LOOP1                   ;if not set yet, test again
```

```
        MOV     #52500,SSBA     ;fill BUF0 first
        MOV     CNTRL,SSDB      ;control word for the SELSPOT/DMA
        MOV     WRDCNT,SSWC     ;2's comp of # of words to be transferred
```

Note that BUFTGL is the number of the buffer being filled.
Also note that the very
last buffer may have a size of 0 in which case no samples will be
taken to fill it, and it will not be written out to the disk. It is
called the "very last buffer" only for identification purposes, and
sometimes it will not be used at all.

```
        CMP     BUFTGL,NFBUFS   ;is the last full buff r filling?
        BLT     LOOP2
        FGT     MVLAST
        TST     NEXWRD
        BEQ     MVLAST
        MOV     NEXWRD,WRDCNT
        JMP     LOOP2
MVLAST: CLR     WRDCNT
LOOP2:  TST     DONE
        BNE     CONTA
        JMP     DISCON
CONTA:  TSTB    IOSTAT
        BPL     CONTB
        JMP     WRTERR
CONTB:  BIT     #2,SSST
        BEQ     LOOP2
```

```
;
; Stop sampling when STOP switch is set.  Mark end of data in the disk file.
;
;
ABORT:  CLR     WRDCNT          ;inhibit further sampling
        MOV     #40000,SSDB     ;turn on STOP status light, stop sampling,
                                ;disable interrupts
;
;
; Find number of words sampled before sampling was stopped
;
;
        MOV     SSWC,R1         ;save word count from this interrupted buffer
        NEG     R1              ;SSWC is 2's comp. of words left to transfer
        MOV     #7680.,R2       ;normal full buffer size
        CMP     BUFTGL,NFBUF1   ;if not filling very last buffer when stopped
        BLT     CONT3           ;then use full buffer size minus SSWC
        MOV     NEXWRD,R2       ;if very last, use partial buffer size minus
        NEG     R2              ;SSWC
CONT3:  SUB     R1,R2           ;buffer size minus SSWC is number of samples
        MOV     R2,NEXBYT
        ASL     NFXBYT          ;*2 for number of bytes
        FDBKSR  #FDBOUT,,NEXBYT
        BIT     #1,BFWTGL
        BNE     CNELST
WRITE$  #FDBOUT,#BUF1
WAIT$   #FDBOUT,#7.
        BCC     CC              ;error if C bit set
        JMP     WRERR           ;jump to error handling routine
        BR      ENDLST
CC:
ONELST: WRITE$  #FDBOUT,#BUF0
WAIT$   #FDBOUT,#7.
        BCC     ENDLST          ;error if C bit set
        JMP     WRERR           ;jump to error handling routine
ENDLST: INC     BFWTGL
        ADD     IOSTAT+2,BYTES
        ADC     BYTES+2
        JMP     DISCN1          ;disable interrupts and exit
```

```
; Error reporting
;
CONERR: MOV  #-1,BUFTGL     ;connect to interrupt error
        JMP  ALDONE
ASTERR: MOV  #-3,BUFTGL     ;ISR FORK2 queued AST exit error
        JMP  DISCON
OPERR:  MOV  #-4,BUFTGL     ;file open error
        JMP  AIDONE
WRERR:  MOV  #-5,BUFTGL     ;block disk write directive error
        JMP  DISCON
WRTERR: MOV  #-6,BUFTGL     ;block disk I/O error (flagged by IOSTAT)
        JMP  DISCON
WASTER: MOV  #-10,BUFTGL    ;I/O queued AST exit error
        JMP  DISCON
CLSERR: MOV  #-11,BUFTGL    ;file close error
        JMP  ALDONE
;
; Finished, disable sampling, disconect interrupt, wait for last buffer to
; write, clean up:
;
DISCON: CLR  WRDCNT         ;inhibit further sampling
LOOP4:  BIT  #2,SSST        ;test for STOP switch
        BNE  ABORT          ;if set, stop sampling
        TSTB IOSTAT         ;any block I/O errors?
        BMI  WRTERR         ;if so flag them and exit
        TST  NEXBYT         ;is the last buffer size 0?
        BNE  CONT4          ;if not, continue
        CMP  BFWTGL,NFBUFS  ;if so, wait for last full buffer to write
        BLE  LOOP4          ;if STOP switch is not set
        JMP  DISCN1         ;once written, disconnect interrupts and exit
CONT4:  CMP  BFWTGL,NFBUF1  ;wait for final disk write if STOP switch
        BLE  LOOP4          ;is not set
```

```
DISCN1: CLR     C2INT+C.INIS        ;clear isr address in CINT$
        DIR$                        ;disconnect to the SELSPOT/DMA interrupt vect
        CLOSE$  #C2INT              ;close TK0:SISPOT.DAT
        CLOSE$  #FDBOUT             ;error if C bit set
        BCC     DD                  ;jump to error handling routine
        JMP     CLSERR
DE:     MOV     #40000,SSDB         ;turn on STOP status light, disable interrupts
        CLR     R2                  ;short wait to allow STOP status light to
WLOOP1: CLR     R1                  ;turn on.
WLOOP2: INC     R1
        CMP     R1,#200.
        BLE     WLOOP2
        INC     R2
        CMP     R2,#500.
        BLE     WLOOP1
;-------------------------------------------------------------------
; Finished.  Write out return data to the FORTRAN main program.
;-------------------------------------------------------------------
ALDONE: DEC     BUFTGL              ;BUFTGL-1 contains the number of interrupts
        CLR     SSDB                ;turn off all status lights
        MOV     #2,(R5)             ;Restore argument list for passage to FORTRAN
        MOV     LIST+2,2(R5)
        MOV     LIST+4,4(R5)
        MOV     LIST+6,6(R5)
        MOV     LIST+10,10(R5)
        MOV     LIST+12,12(R5)
        ADD     #2,LIST+12
        MOV     LIST+14,14(R5)
        MOV     CNTRL,@4(R5)        ;for diagnostics pass DMA control word back
        MOV     BUFTGL,@6(R5)
        MOV     IOSTAT,@10(R5)      ;IOSTAT contains the block WRITE error code
        MOV     BYTES,@12(R5)       ;BYTES and BYTES+2 contain the total number of
        MOV     BYTES+2,@LIST+12    ;bytes written to the disk(double precision)
        MOV     $DSW,@14(R5)        ;$DSW contains the directive status word
        RETURN                      ;return to calling program (usually T3SFTM)
```

```
; AST routine queued from the ISR FORK2 routine.

ASTRTN: TST    DONE            ;is further sampling disabled?
        BNE    WRITEB          ;if not write out this buffer
        TST    NEXBYT          ;is the very last buffer size 0?
        BEQ    WRITEF          ;if not write out last full buffer
        CMP    FUFTGL,NFBUF1   ;has the last buffer not been reached?
        BLT    WRITEE          ;if not write out this last full buffer
        MOV    #40000,SSDE     ;turn on STOP status light, disable interrupts
        FDBK$R #1,BFWTGL,NEXBYT ;set up to write last buffer
WRITEF: BIT    #1,BFWTGL       ;test which buffer to write
        BNE    ONEAST
        WRITE$ #FDBOUT,#BUF1   ;write out BUF1
        WAIT$  #FDBOUT,#7.
        BCC    EE              ;error if C bit set
        MOVB   #-1,IOSTAT      ;error code =-1, undetermined I/O error
        BR     ENDAST
EE:
ONEAST: WRITE$ #FDBOUT,#BUF0   ;write out BUF0
        WAIT$  #FDBOUT,#7.
        BCC    ENDAST          ;error if C bit set
        MOVB   #-1,IOSTAT      ;error code =-1, undetermined I/O error
ENDAST: INC    BFWTGL          ;increment BUFTGL to toggle buffers
        TST    (SP)+           ;Normal AST exit requires stack pop
        ASTX$S ASTERR          ;AST exit directive (if error go to ASTERR)

; AST routine queued after each disk write completes.  It keeps a
; running total of bytes written to disk.

WRTAST: ADD    IOSTAT+2,BYTES  ;keep running total of disk bytes written
        ADC    BYTES+2         ;double precision result
        TST    (SP)+           ;must pop stack before AST exit
        ASTX$S WASTER          ;AST exit directive
```

```
; Data
;
C2INT:  CINT$  124,ISR,0,ENADIS,PR7,ASTRTN  ;Connect to interrupt directive
LIST:   .BYTE  6,0                          ;FORTRAN argument list storage area
        .BLKW  6
CNTRL:  .WORD  0
NCHN:   .WORD  0
NSKIP:  .WORD  0
NEXBYT: .WORD  0
NTBLKS: .WORD  0
BYTES:  .BLKW  2
NBYTES: .WORD  0
IOSTAT: .BLKW  2
BFWTGL: .WORD  1
;
; Block disk I/O file name storage.
;
DATAST: .WORD  DVNMSZ,DEVNAM
        .WORD  DRNMSZ,DIRNAM
        .WORE  FLNMSZ,FILNAM
        .EVEN
DEVNAM: .ASCII  /TK0:/
DVNMSZ=.-DEVNAM
        .EVEN
DIRNAM: .ASCII  //
DRNMSZ=.-DIRNAM
        .EVEN
FILNAM: .ASCII  /SLSPOT.DAT/
FLNMSZ=.-FILNAM
        .EVEN
```

```
;*****************************************************************
; Interrupt service routine
;*****************************************************************
        .PSECT  ISR,RW,D,GBL,REL,OVR
ISR:    BIT     #1,BUFTGL       ;test toggle to determine which buffer to fill
        BNE     ONE
        MOV     #52500,@#172412 ;move BUF0 address to SSBA
        MOV     WRDCNT,@#172410 ;# of words to transfer to SSWC
        INC     BUFTGL          ;Increment BUFTGL to toggle buffers
        BR      ISDONE
ONE:    MOV     #110500,@#172412 ;move BUF1 address to SSBA
        MOV     WRDCNT,@#172410 ;# of words to transfer to SSWC
        INC     BUFTGL          ;toggle buffers
ISLONF: CMP     BUFTGL,NFBUFS   ;test for filling last full buffer
        BLT     ISREXT          ;if not exit ISR
        BGT     VLAST           ;if beyond last full, must be VERY last buffer
        TST     NEXWRD
        BEQ     VLAST           ;if very last is empty
        MOV     NEXWRD,WRDCNT
        JMP     ISREXT
VLAST:  CLR     WRDCNT          ;very last is filling,inhibit further sampling
        CMP     BUFTGL,NFBUF1   ;if interrup AFTER very last
        BLE     ISREXT          ;then:
        MOV     #40000,@#172416 ;Disable interrupts. Turn on stop light SSDB
        CLR     DONE            ;Indicate sampling is finished
ISREXT: CALL    @#$FORK2        ;FORK2 routine to allow directive execution
        CLR     @R3             ;This is required for FORK2 routine
        CALL    @#$QASTC        ;Queue an AST
ENADIS: RTS     PC              ;Return
BUFTGL: .WORD   1
WRDCNT: .WORD   0
NFBUFS: .WORD   0
NFBUF1: .WORD   0
NEXWRD: .WORD   0
DONE:   .WORD   0
        .END
```

```
; T3FPM1 -- TRACK III SUBROUTINE SINGLE SAMPLE MACRO ROUTINE
; T3FPM1
;
;       SEP-81          by Erik Antonsson
;
;
        .TITLE  T3FPM1 -- TRACK III SINGLE SAMPLE SUBROUTINE
        .IDENT  /V01A/
        .PSECT  T3FPM1
T3FPM1::CALL    $SAVAL
        CLR     SSDB            ;Reset the Selspot Interface
        BIT     #1000,DRST      ;Test for DAS power on
        BEQ     ERROR           ;Quit if off
        TST     ERST            ;Test for ATTN
        BMI     ERROR           ;Quit if set
        TSTB    DRST            ;Is the DMA READY?
        BPL     ERROR           ;If not, exit.
        CLR     DRST            ;Be sure FNCT1 is low
        BIS     #4,DRST         ;Enable external FP frame start FNCT2
T3FPAC::MOV     BFADD,DRBA      ;Data Buffer Address
        BIS     BFEXA,DRST      ;Extended Address Bits
        MOV     #-6.,DRWC       ;Transfer 8 words (1 full frame)
        BIS     #3,DRST         ;FNCT1,GO
LOOP1:  TST     DRST            ;Test for ATTN
        BMI     ERROR           ;Quit if set
        BIT     #1000,DRST      ;Test for DAS power on
        BEQ     ERROR           ;Quit if off
        TSTB    DRST            ;Wait for frame to complete
        BPL     LOOP1           ;Loop to wait for sampling to finish
        BIC     #3,DRST         ;Reset the interface
        CLR     @2(R5)          ;Clear Error Flag
        RETURN
ERROR:  MOV     LRST,@2(R5)     ;SET ERROR FLAG
T3FPCT::CLR     ERST            ;Clear F/P interface
        RETURN
        .END
```

```
;  T3FPM2 -- INTERRUPT SERVICED FCEPLT SAMPLING WITH DISK DATA STORAGE
;  T3FPM2

        .TITLE  T3FPM2 -- INTERRUPT SERVICED FCEPLT SAMPLING
        .IDENT  /V01A/
        .MCALL  DIR$,CINT$,ASTX$S
        .MCALL  FDBDF$,FDAT$A,FDRC$A,FDBK$A,FDOP$A
        .MCALL  FDAT$R,OPEN$W,WRITE$,CLOSE$,FDBK$R,WAIT$
MESSAGE =       @                       ;block disk write set up macros
FDFOUT: .CSECT
        FDBDF$
        FDAT$A  R.FIX,,16.,30.
        FDRC$A  FD.RWM
        FDBK$A  ,1536.,20.,IOSTAT,WRTAST
        FDOP$A  15.,DATAST,,FO.WRT
T3FPM2::CALL    $SAVAL          ;save all registers
        MOV     2(R5),LIST+2    ;save argument list
        MOV     4(R5),LIST+4
        MOV     6(R5),LIST+6
        MOV     10(R5),LIST+10
        MOV     12(R5),LIST+12
        MOV     14(R5),LIST+14
        MOV     @2(R5),NBBUFS   ;get argument values
        MOV     @4(R5),NEXWRD
        MOV     @6(R5),NTBLKS
        MOV     @10(R5),ARATE
        NEG     ARATE
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
        BIT     #1000,DRST      ;Test for DAS power on
        BNE     CONTM1
        JMP     HDWERR
CONTM1: TST     DRST            ;Test for FCrPLT hdwe error
        BPL     CONTM
        JMP     HDWERR
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
CONT&:   MOV    NFBUFS,NFBUF1        ;If very last buffer is empty
         TST    NEXWRD              ;dont increment NFBUFS
         BEQ    CONT@A              ;NFBUF1=NFBUFS+1 to flag last buffer
         INC    NFBUF1              ;NEXBYT is the # of extra bytes in last buf
CONT@A:  MOV    NEXWRD,NEXBYT       ;(=NEXWRD*2)
         ASL    NEXBYT
         NEG    NEXWRD
         MOV    #-768.,WRDCNT       ;normally set up 1st buffer to be a full one
         TST    NFBUFS
         BNE    CONT1
         MOV    NEXWRD,WRDCNT       ;if @ full buffers, set up 1st to be the last
CONT1:   CLR    BYTES               ;init double precision running sum of bytes
         CLR    BYTES+2             ;written to the disk
         FDAT$R #FDROUT,.,.,NTBLKS  ;adjust file params to number of blcks
                                    ;to be written
         OPEN$W #FDROUT             ;open TK1:FCEPLT.DAT
         BCC    &A                  ;error if C bit set
         JMP    OPERR               ;jump to error handling routine
AA:      MOV    #ISR,C2INT+C.INIS   ;set up interrupt service routine
         MOV    #132,C2INT+C.INVE   ;address for CINT$
                                    ;set up interrupt vector for CINT$
         DIR$   #C2INT              ;execute directive to connect to interrupt
         BCC    CONT12
         JMP    CONERR              ;C bit set if directive error
CONT12:  CLR    SSLF                ;reset switch registers
         MOV    #20060,SSDR         ;turn on READY status light
         CLR    IRWC                ;clear FCEPIT DMA word count register
         CLR    DRST                ;clear the status register
         MOV    #1,EUFTGL           ;set up buff toggle to write FPBUF@ out first
         MOV    #1,BFWTGL           ;Set up buf write toggle
         MOV    #131,CNTRL          ;DMA CSR bits for Int enb, Ext Addr,FNCT3,GO
         MOV    #1,DONE             ;Set up done sampling flag
IOOP1:   BIT    #2,SSST             ;Test for STOP switch hit
         BEQ    LOOP1A              ;If STOP not hit, test for START
         JMP    DISCCN              ;If STOP hit before START, goto DISCON
```

```
LOOP1A: BIT   #1,SSST            ;test for START switch
        BEQ   LOOP1              ;if not set yet, test again
        MOV   #44500,DRBA        ;fill FPEUF0 first
        MOV   WRDCNT,DRWC        ;2's comp of # of words to be transferred
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
        MOV   CNTRL,DRST         ;Int Enb,Extended Address Bits,FNCT2,GO
        MOV   ARATE,KWABUF       ;Clock A overflow every ARATE Counts
        MOV   #405,KWACSR        ;Count continuously 100kHz starting now
        CLR   SSDR               ;Turn off READY status light
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; Note that BUFTGL is the number of the buffer being filled.
; Also note that the very
; last buffer may have a size of 0 in which case no samples will be
; taken to fill it, and it will not be written out to the disk.  It is
; called the "very last buffer" only for identification purposes, and
; sometimes it will not be used at all.
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
        CMP   BUFTGL,NFBUFS      ;is the last full buffer filling?
        BLT   LOOP2              ;If not,goto LOOP2
        BGT   MVLAST             ;goto MVLAST if the VERY last buffer filling
        TST   NEXWRD             ;Is the size of the VERY last 0?
        BEQ   MVLAST             ;If so,goto MVLAST
        MOV   NEXWRD,WRDCNT      ;If not, set up WRDCNT to fill partial buffer
        JMP   LOOP2
MVLAST: MOV   #4,CNTRL           ;Halt sampling on next interrupt
        CLR   WRDCNT
LOOP2:  TST   LONE               ;Has sampling finished?
        BNE   CONTA              ;If not, goto CONTA
        JMP   DISCON             ;If so, goto DISCON
CONTA:  TSTB  IOSTAT             ;Were there any I/O errors?
        BPL   CONTB              ;If not, goto CONTB
        JMP   WRTERR             ;If so, goto WRTERR
CONTB:  BIT   #2,SSST            ;Test for STOP switch hit
        BEQ   LOCP2              ;Loop again if not hit.
```

```
;----------------------------------------------------------------
; Stop sampling when STOP switch is set.  Mark end of data in the disk file.
;----------------------------------------------------------------
ABORT:  CLR     WRDCNT          ;inhibit further sampling
        CLR     CNTRL           ;Clear control word
        MOV     #40000,SSDE     ;turn on STOP status light, stop sampling,
                                ;disable interrupts
        CLR     DRST            ;Clear DR11-E CSR
;----------------------------------------------------------------
; Find number of words sampled before sampling was stopped
;----------------------------------------------------------------
        MOV     DRWC,R1         ;save word count from this interrupted buffer
        NEG     R1              ;SSWC is 2's comp. of words left to transfer
        MOV     #768.,R2        ;normal full buffer size
        CMP     BUFTGL,NFBUF1   ;if not filling very last buffer when stopped
        BLT     CONT3           ;then use full buffer size minus DRWC
        MOV     NEXWRD,R2       ;if very last, use partial buffer size minus
        NEG     R2              ;DRWC
CONT3:  SUB     R1,R2           ;buffer size minus DRWC is number of samples
        MOV     R2,NEXBYT
        ASL     NEXBYT          ;*2 for number of bytes
        FDBK$R  #FDBOUT,,NEXBYT
        BIT     #1,BFWTGL
        BNE     ONELST
        WRITE$  #FDBOUT,#FPBUF1
        WAIT$   #FDBOUT,#20.
        BCC     CC
        JMP     WRERR           ;error if C bit set
                                ;jump to error handling routine
CC:     BR      ENDLST
ONELST: WRITE$  #FDBOUT,#FPBUFG
        WAIT$   #FDBOUT,#20.
        BCC     ENDLST          ;error if C bit set
        JMP     WRERR           ;jump to error handling routine
ENDLST: INC     BFWTGL
        ADD     IOSTAT+2,BYTES
        ADC     BYTES+2
        JMP     DISCN1          ;disable interrupts and exit
```

- 555 -

```
;---------------------------------------------------------------------
;   Error reporting
;---------------------------------------------------------------------
;
CONERR: MOV     #-1,BUFTGL          ;connect to interrupt error
        JMP     ALDONE
DISERR: MOV     #-2,BUFTGL          ;disconnect to interrupt error
        JMP     ALDONE
ASTERR: MOV     #-3,BUFTGL          ;ISR FORK2 queued AST exit error
        JMP     DISCON
OPERR:  MOV     #-4,BUFTGL          ;file open error
        JMP     ALDONE
WRERR:  MOV     #-5,BUFTGL          ;block disk write directive error
        JMP     DISCON
WRTERR: MOV     #-6,BUFTGL          ;block disk I/O error (flagged by IOSTAT)
        JMP     DISCON
WASTER: MOV     #-10,BUFTGL         ;I/O queued AST exit error
        JMP     DISCON
CLSERR: MOV     #-11,BUFTGL         ;file close error
        JMP     ALDONE
;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
HDWERR: MOV     #-12,BUFTGL         ;FCEFLT hardware error
        MOV     LIST+4,4(R5)
        MOV     DRST,G4(R5)
        CLR     LRST
        CLR     KWACSR
        JMP     AIDONE
;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

```
;
;       Finished, disable sampling, disconnect interrupt, wait for last buffer to
;       write, clean up.
;
DISCON: CLR    WRDCNT        ;inhibit further sampling
        MOV    #124,CNTRL    ;inhibit further sampling, leave int enb
LOOP4:  BIT    #2,SSST       ;test for STOP switch
        BEQ    LOOP42
        JMP    ABORT
LOOP42: TSTB   IOSTAT        ;if set, stop sampling now
        BMI    WRTERR        ;any block I/O errors?
        TST    NEXBYT        ;if so flag them and exit
        FNE    CONT4         ;is the last buffer size 0?
        CMP    BFWTGL,NFBUFS ;if not, continue
        BLE    LOOP4         ;if so, wait for last full buffer to write
        JMP    DISCN1        ;if STOP switch is not set
CONT4:  CMP    BFWTGL,NFBUF1 ;once written, disconnect interrupts and exit
        BLE    LOOP4         ;wait for final disk write if STOP switch
                            ;is not set
DISCN1: CLR    C2INT+C.INIS  ;clear ISR address in CINT$
        DIR$   #C2INT        ;disconnect the FP/DMA interrupt vector
        CLOSE$ #FDEOUT       ;close TK0:DK11HL01.DAT
        BCC    DD            ;error if C bit set
        JMP    CLSERR        ;jump to error handling routine
DD:     MOV    #4000,SSDB    ;turn on STOP status light, disable interrupts
        ;=================================================================
        MOV    DRST,CNTRL    ;Save DR11-F CSR for F4b return
        CLR    DRST          ;Reset the interface
        CLR    KWACSR        ;Stop Clock A
        ;=================================================================
        CLR    R2
WLOOP1: CLR    R1            ;short wait to allow STOP status light to
WLOCP2: INC    R1            ;turn on.
        CMP    R1,#20.
        BLE    WLOOP2
        INC    R2
        CMP    R2,#50.
        BLE    WLOOP1
```

```
; Finished.  Write out return data to the FORTRAN main program.
;
ALDONE: DEC    BUFTGL               ;BUFTGL-1 contains the number of interrupts
        CLR    SSDB                 ;turn off all status lights
        MOV    #2,(R5)              ;Restore argument list for passage to FORTRAN
        MOV    LIST+2,2(R5)
        MOV    LIST+4,4(R5)
        MOV    LIST+6,6(R5)
        MOV    LIST+10,10(R5)
        MOV    LIST+12,12(R5)
        ADD    #2,LIST+12
        MOV    LIST+14,14(R5)
        MOV    CNTRL,@4(R5)         ;for diagnostics pass DMA control word back
        MOV    BUFTGL,@6(R5)
        MOV    IOSTAT,@10(R5)       ;IOSTAT contains the block WRITE error code
        MOV    BYTES,@12(R5)        ;BYTES and BYTES+2 contain the total number of
        MOV    BYTES+2,@LIST+12     ;bytes written to the disk (double precision)
        MOV    $DSW,@14(R5)         ;$DSW contains the directive status word
        RETURN                      ;return to calling program (usually T3SFTM)
```

```
; AST routine queued from the ISR FORK2 routine.

ASTRTN: TST    DONE           ;is further sampling disabled?
        BNE    WRITEB         ;if not write out this buffer
        TST    NEXBYT         ;is the very last buffer size 0?
        BEQ    WRITEB         ;if not write out last full buffer
        CMP    BUFTGL,NFBUF1  ;has the last buffer not been reached?
        BLT    WRITEB         ;if not write out this last full buffer
        MOV    #40000,SSDB    ;turn on STOP status light, disable interrupts
        CLR    ERST           ;disable interrupts
        FDBK$R #FDBOUT,,NEXBYT ;set up to write last buffer
WRITEB: BIT    #1,BFWTGL      ;test which buffer to write
        BNE    ONEAST
        WRITE$ #FDBOUT,#FPBUF1 ;write out BUF1
        WAIT$  #FDBOUT,#20.
        BCC    EE             ;error if C bit set
        MOVB   #-1,IOSTAT     ;error code =-1, undetermined I/O error
        BR     ENDAST
EE:
ONEAST: WRITE$ #FDBOUT,#FPBUF0 ;write out BUF0
        WAIT$  #FDBOUT,#20.
        BCC    ENDAST         ;error if C bit set
        MOVB   #-1,IOSTAT     ;error code =-1, undetermined I/O error
ENDAST: INC    BFWTGL         ;increment BUFTGL to toggle buffers
        TST    (SP)+          ;Normal AST exit requires stack pop
        ASTX$S ASTERR         ;AST exit directive (if error go to ASTERR)

; AST routine queued after each disk write completes.  It keeps a
; running total of bytes written to disk.

WRTAST: ADD    IOSTAT+2,BYTES ;keep running total of disk bytes written
        ADC    BYTES+2        ;double precision result
        TST    (SP)+          ;must pop stack before AST exit
        ASTX$S WASTER         ;AST exit directive
```

```
;
; Data
;
C2INT:  CINT$   130,ISR,0,ENADIS,PRE,ASTRTN;Connect to interrupt directive
LIST:   .BYTE   6,0         ;FORTRAN argument list storage area
        .BLKW   6
NCHN:   .WORD   0
NSKIP:  .WORD   0
ARATE:  .WORD   0
NEXBYT: .WORD   0
NTBLKS: .WORD   0
BYTES:  .BLKW   2
NBYTES: .WORD   0
IOSTAT: .BLKW   2
;
; Block disk I/O file name storage.
;
DATAST: .WORD   DVNMSZ,DEVNAM
        .WORD   DRNMSZ,DIRNAM
        .WORD   FLNMSZ,FILNAM
        .EVEN
DEVNAM: .ASCII  /TK1:/
DVNMSZ=.-DEVNAM
        .EVEN
DIRNAM: .ASCII  //
DRNMSZ=.-DIRNAM
        .EVEN
FILNAM: .ASCII  /FCEPLT.LAT/
FLNMSZ=.-FILNAM
        .EVEN
```

```
;**************************************************************************
;       Interrupt service routine
;**************************************************************************
;
;
;
        .PSECT  ISR,RW,D,GBL,REL,OVR
ISR:    BIT     #1,BUFTGL          ;test toggle to determine which buffer to fill
        BNE     ONE
        MOV     #44500,@#1724432   ;move BUF0 address to DRBA
        MOV     WRDCNT,@#1724430   ;# of words to transfer to DRWC
        MOV     CNTRL,@#1724434    ;Int enb,ext addr,FNCT2,GO
        INC     BUFTGL             ;Toggle buffers
        BR      ISDONE
ONE:    MOV     #47500,@#1724432   ;move BUF1 address to DRBA
        MOV     WRDCNT,@#1724430   ;# of words to transfer to DRWC
        MOV     CNTRL,@#1724434    ;Int enb,ext addr,FNCT2,GO
        INC     BUFTGL             ;Toggle buffers
ISDONE: CMP     BUFTGL,NFBUFS
        BLT     ISREXT
        BGT     VLAST
        TST     NEXWRD
        BEQ     VLAST
        MOV     NEXWRD,WRDCNT
        JMP     ISREXT
VLAST:  MOV     #124,CNTRL
        CLR     WRDCNT
        CMP     BUFTGL,NFBUF1
        BLE     ISREXT
        CLR     @#1724434
        CLR     DRST
        CLR     DCNE
ISREXT: CALL    @#$FORK2           ;FORK2 routine to allow directive execution
        CLR     @R3                ;This is required for FORK2 routine
        CALL    @#$QASTC           ;Queue an AST
ENADIS: RTS     PC                 ;Return
```

```
BUFTGL: .WORD   1
BFWTGL: .WORD   1
WRDCNT: .WORD   0
CNTRL:  .WORD   0
NFBUFS: .WORD   0
NFBUF1: .WORD   0
NEXWRD: .WORD   0
DONE:   .END
```

T3SSM3 -- INTERRUPT SERVICED SS AND FP SAMPLING WITH DISK DATA STORAGE
T3SSM3

by Erik Antonsson   12-JUN-81

Copyright (C) 1981 Massachusetts Institute of Technology
Cambridge, Massachusetts

This routine acquires data from 2) DR11-Bs DMA (direct memory access
device) and with 2 double buffers writes it out to 2 disk files.
One DMA, set of buffers and one disk file are dedicated to the
Selspot device, and the other DMA, set of buffers and disk file
are for the Kistler forceplate (via the Analog Devices DAS 1128).

The FORTRAN CALL is as follows:

    CALL T3SSM2(NCHN,NSKIP,NFBUFS,NEXWRD,NTBLKS,FCNTRL,
    *FNFBFS,FNXWRD,FNTBLKS,IDSW)

Where:

NCHN is the number of active LEDs.
    N.B. The only allowable values for NCHN are:
    1-6,8,10,12,15,16,20,24,30 to ensure that a FRAME
    boundry and disk write block (7680 words) coincide.

NSKIP is the number of frames to skip +1 between sampled frames.
    NSKIP is returned as the SELSPCT/DMA control word (octal).

NFBUFS is the total number of FULL 7680. word SS buffers to fill.
    NFBUFS is returned with the number of SS buffers written.

NEXWRD is the number of words in the very last SS buffer (NFBUFS+1).
    If NEXWRD is non zero, the TOTAL number of SS buffers filled, either
    fully or partially is NFBUFS+1.
    NEXWRD is returned with the SS write error code if an error occured.
    See pages 5-7 through 5-9, or B-1 through B-3 of the I/O Drivers
    Reference Manual in Volume 5A RSX-11M V3.2.

NTBLKS(1) is the TOTAL number of 256 word SS disk blocks to be written
    NTBLKS(1), and NTBLKS(2) are returned with the number of SS bytes
    actually written (double precision integer).

FCNTRL is returned as the FP/LMA control word (octal).

FNFBFS is the total number of FULL 768. word buffers FP buffers to
    fill.

FNFFS is returned with the number of FP buffers written.

FNXWRD is the nuber of words in the very last FP buffer (FNFBFS+1).
    If FNXWRD is non-zero, the TOTAL number of FP buffers
    filled, either fully or partially is NFBUFS+1.
    FNXWRD is returned with the FP write error code if an error occured.
    See pages 5-7 through 5-9, or B-1 through B-3 of the I/O Drivers
    Reference Manual in Volume 5A RSX-11M V3.2.

FNTBIK(1) is the TOTAL number of 256 word FP disk blocks to be
    written.
    FNTBIK(1) and FNTBIK(2) are returned with the number of FP bytes
    actually written (double precision integer).

IDSW is the directive status word returned for error decoding.
See Appendix B of the Executive Reference Manual in Volume 3B
RSX-11M V3.2.

SSWC,SSBA,SSST,SSDB are the SELSPOT DMA registers defined in IOPAGE

SSWC	172410	transfer word count
SSBA	172412	data buffer bus address register
SSST	172414	status register
SSDE	172416	data register (used for control)

DRWC,DRBA,DRST,DRDB are the FORCE PLATE DMA registers defined in IOPAGE

DRWC	172430	transfer word count
DRBA	172432	data buffer bus address register
DRST	172434	status register
DRDB	172436	data register (used for control)

BUF0 and BUF1 physical addresses are retrieved from the base address of
the installed common partition SSCOM0 (242500) and SSCOM1 (300500) obtained
at VMR. These partitions and installed C (MMONs are necessary only because
the physical address of the data buffers must be passed to the DMA and
this particular DMA has the extended address bits set to 01, necessatating
the leading 2 in the address and requiring that the buffers be fixed
(more or less) in memory. A fixed partition is the only way to insure
this location, and it eliminates the chore of finding the physical
base address of the buffers by the program.
It is worth noting that only the first word of these two buffer
partitions need be included in COMMON, because the base address of
the buffer is all that any DMA device nee s to fill/empty a buffer, even
if the higher addresses in the buffer arent in program space. Both the

DR11-B acquisition device and the disk are DMA and
since the data is only put in memory by the DR11-B to be written out to
the disk, these buffers nee not be wholly included in the COMMON.

FPBUF0 and FPBUF1 physical addresses are retrieved from the base address
of the installed common partition FPCOM obtained at VMR or by the MCR
function FAR. These partitions and intalled COMMON are necessary only
because the physical address of the data buffers must be passed to the
DMA. (FPBUF0: 234500, FPBUF1: 237500)

The extended address bits are hardwired to :01 in the SELSPOT DMA interface
thus only the lower 16 bits are moved to SSBA.
The extended address bits for the FP/DMA buffer addresses are set in the
DRST (CSR) and are selectable under program control.

This subroutine uses an AST routine queued from the ISR to write the
data to disk. The data is stored in disk file: TK0:SLSPOT.DAT
using logical unit number (LUN) 14 and event flag 7. The FP also uses
an AST routine queued from the FPISR to write data to disk. The FP
data is stored in disk file: TK1:FCEPLT.DAT using logical unit number
(LUN) 15 and event flag 20.
Note that these LUNS and event flags are unavailable
for use in the calling programs.

The usual calling program should be something like T3SSFP.FTN a program
written by Erik Antonsson to simplify the input to this routine without
having to do extensive arithmetic manipulations in MACRO.

The end use of this routine is to be imbedded in a calling program
that issues a sampling sequence for the SS/DMA to acquire 8 channels
of A/D data (from the force plate) synchronously with this Selspot/DMA
data. The synchrony is provided by a pulse (from the SS/DMA interface
everytime a frame of data is taken) fed into the digital trigger for
the A/D sweeps of the DAS 1128.

```
        .TITLE  T3SSM3  -- INTRPT SERVICED FP/SS SAMPLING
        .IDENT  /V01A/
        .MCALL  DIR$,CINT$,ASTX$S
        .MCALL  FDBDF$,FDAT$A,FDRC$A,FDBK$A,FDOP$A
        .MCALL  FDAT$R,OPEN$W,WRITE$,CLOSE$,FDBK$R,WAIT$
M$$MGE  =       0

        .CSECT
FDBOUT: FDBDF$                                  ;S;block disk write set up macros
        FDAT$A  R.FIX,,512.,30.                 ;S
        FDRC$A  FD.RWM                          ;S
        FDBK$A  ,15360.,,7-.,IOSTAT,WRTAST       ;S
        FDOP$A  14.,DATAST,,FO.WRT              ;S

FPLFDB: FDBDF$                                  ;F;FP block disk write macros
        FDAT$A  R.FIX,,16.,30.                  ;F
        FDRC$A  FD.RWM                          ;F
        FDBK$A  ,1536.,,20.,FOSTAT,FWRAST        ;F
        FDOP$A  15.,FDATAS,,FO.WRT              ;F
T3SSM3::CALL    $SAVAL                          ;save all registers
        MOV     2(R5),LIST+2                    ;save argument list
        MOV     4(R5),LIST+4
        MOV     6(R5),LIST+6
        MOV     10(R5),LIST+10
        MOV     12(R5),LIST+12
        MOV     14(R5),LIST+14
        MOV     16(R5),LIST+16
        MOV     20(R5),LIST+20
        MOV     22(R5),LIST+22
        MOV     24(R5),LIST+24
        MOV     @2(R5),NCHN                     ;S;get argument values
        MOV     @4(R5),NSKIP                    ;S
        MOV     @6(R5),NFBUFS                   ;S
        MOV     @10(R5),NEXWRD                  ;S
        MOV     @12(R5),NTBLKS                  ;S
```

```
           MOV    @16(R5),FNFBFS        ;F;get FP argument values
           MOV    @20(R5),FNXWRD        ;F
           MOV    @22(R5),FNTBLK        ;F
           BIT    #1000,DRST            ;F;Test for F/P DAS power on
           BNE    FCNTM1                ;F
           JMP    FHDWER                ;F
FCNTM1:    TST    DRST                  ;F;Test for FCEPLT hdwe error
           BPL    FCNT0                 ;F
           JMP    FHDWER                ;F
FCNT0:     MOV    FNFBFS,FNFBF1         ;F
           TST    FNXWRD                ;F;If Very last buffer is Zero in size
           BEQ    FCNT0A                ;F;Then Very last is last full buffer
           INC    FNFBF1                ;F;FNFBF1=FNFBFS+1 to flag last buffer
FCNT0A:    MOV    FNXWRD,FNXBYT         ;F;FNXBYT is the # of extra bytes in last buf
           ASL    FNXBYT                ;F;(=FNXWRD*2)
           NEG    FNXWRD                ;F
           MOV    #-768.,FWRDCT         ;F;normally set up 1st buffer to be a full one
           TST    FNFBFS                ;F
           BNE    FCONT1                ;F
           MOV    FNXWRD,FWRDCT         ;F;if 0 full buffers, set up 1st to be last
FCONT1:    CLR    FBYTES                ;F;init double precision running sum of bytes
           CLR    FBYTES+2              ;F;written to the disk
           FDAT$R #FPLFDB,,,,FNTBLK     ;F;adjust file params to number of blk
                                        ;F;to be written
           OPEN$W #FPLFDB               ;F;open TK1:FCEPLT.DAT
           BCC    FAA                   ;F;error if C bit set
           JMP    FOPERR                ;F;jump to error handling routine
FAA:       MOV    #FPLISR,FC2INT+C.INIS ;F;set up interrupt service routine
           MOV    #130,FC2INT+C.INVE    ;F;address for CINT$
           DIR$   #FC2INT               ;F;set up interrupt vector for CINT$
           BCC    FCNT12                ;F;execute directive to connect to interrupt
           JMP    FCNERR                ;F;C bit set if directive error
```

```
FCNT12: CLR   DRWC              ;F;clear FCEPLT DMA word count register
        CLR   DRST              ;F;clear the status register
        MOV   #1,FBFTGL         ;F;set up buffer toggle to write BUF0 out 1st
        MOV   #1,FBFWTG         ;F;Set up buf write toggle
        MOV   #121,FCNTRL       ;F;DMA CSR bits for Int enb, Ext Addr,FNCT2,GO
        MOV   NCHN,NBYTES       ;S
        ASL   NBYTES            ;S;NCHN*6 = number of bytes in each record
        ASL   NBYTES            ;S
        ASL   NBYTES            ;S
        MOV   NFBUFS,NFBUF1     ;S
        TST   NEXWRD            ;S;Is the very last buffer size zero?
        BEQ   FCT12A            ;S;If so, Very last buffer is last full buffer
        INC   NFBUF1            ;S;NFBUF1=NFBUFS+1 to flag last buffer
FCT12A: MOV   NEXWRD,NEXBYT     ;S;NEXBYT is the # of extra bytes in last buf
        ASL   NEXBYT            ;S;(=NEXWRD*2)
        NEGB  NCHN              ;S;these 3 arguments need to be in 2's comp
        NEGB  NSKIP             ;S
        NEG   NEXWRD            ;S
        MOVB  NSKIP,CNTRL       ;S;low byte is 2's comp of <frames to skip +1>
        MOVB  NCHN,CNTRL+1      ;S;next 5 bits: 2's comp of # of active LEDs
        BIS   #16000,CNTRL      ;S;hi-bit set: interrupt enab,bits 13,14:start
        MOV   #-7680.,WRDCNT    ;S;normally set up 1st buffer to be a full one
        TST   NFBUFS            ;S
        BNE   CONT1             ;S
CONT1:  MOV   NEXWRD,WRDCNT     ;S;if 0 full buffs, set up 1st to be the last
        CLR   BYTES             ;S;init double precision running sum of bytes
        CLR   BYTES+2           ;S;written to the disk
FDAT$R  #FDBOUT,,,NBYTES,NTBLKS ;S;adjust file params to # of blcks
                                ;S;to be written, and record size
OPEN$W  #FDBOUT                 ;S;open TK0:SLSPOT.LAT
        BCC   AA                ;S;error if C bit set
        JMP   OPERR             ;S;jump to error handling routine
```

```
AA:     MOV    #ISR,C2INT+C.INIS    ;S;set up interrupt service routine
        MOV    #124,C2INT+C.INVE    ;S;address for CINT$
        DIR$   #C2INT               ;S;set up interrupt vector for CINT$
        BCC    CONT12               ;S;execute directive to connect to interrupt
        JMP    CONERR               ;S
CONT12: CLR    SSDB                 ;S;C bit set if directive error
        MOV    #20000,SSDB          ;S;reset switch registers
        CLR    SSWC                 ;S;turn on READY status light
        CLR    SSST                 ;S;clear SELSPOT DMA word count register
        MOV    #1,BUFTGL            ;S;clear the status register
        MOV    #1,BFWTGL            ;S;set up buffer toggle to write BUF0 out 1st
        MOV    #1,SDONE             ;S
        MOV    #1,FDONE             ;S;Set SS sampling flag
LOOP1:  BIT    #2,SSST              ;F;Set FP sampling flag
        BEQ    LOOP1C               ; ;Test for STOP switch hit
        JMP    DISCON               ; ;If STOP switch not hit, test START switch
LOOP1C: BIT    #1,SSST              ; ;If STOP switch hit, goto DISCON
        BEQ    LOOP1                ;S;test for START switch
        MOV    #44500,DKBA          ;S;if not set yet, test again
        MOV    FCNTRL,DRST          ;F;fill FPBUF0 first
        MOV    FWRDCT,DKWC          ;F;Int Enb,Extended Address Bits,FNCT2,GC
        MOV    #52500,SSBA          ;F;2's comp of # of words to be transferred
        MOV    CNTRL,SSDB           ;S;fill BUF0 first
        MOV    WRDCNT,SSWC          ;S;control word for the SELSPOT/DMA
        CMP    FBFTGL,FNFBFS        ;S;2's comp of # of words to be transferred
        BLT    LOOP1A               ;Is the last full buffer filling?
        BGT    MFVLST               ;if not goto wait loop
        TST    FNXWRD               ;if beyond, the very last buffer is filling
        BEQ    MFVLST               ;is very last buffer size zero?
        MOV    FNXWRD,FWRDCT        ;if so, stop sampling
        JMP    LOOP1A               ;if not, set up wordcount for next interrupt
```

```
MFVLST: MOV   #4,FCNTRL       ;Stop sampling on next interrupt
        CLR   FWRDCT          ;Stop sampling on next interrupt
LOOP1A: CMP   BUFTGL,NFBUFS   ;test for filling last full buffer
        BLT   LOOP2           ;if not exit this loop
        BGT   MVLAST          ;if beyond last full, must be VERY last buffer
        TST   NEXWRD          ;if very last is empty
        BEQ   MVLAST          ;stop sampling
        MOV   NEXWRD,WRDCNT   ;if not, set up wordcount for next interrupt
        JMP   LOOP2

MVLAST: CLR   WRDCNT          ;very last is filling,inhibit further sampling
;--------------------------------------------------------------------
;  Note that BUFTGL is the number of the buffer being filled.
;  Also note that the very
;  last buffer may have a size of 0 in which case no samples will be
;  taken to fill it, and it will not be written out to the disk.  It is
;  called the "very last buffer" only for identification purposes, and
;  sometimes it will not be used at all.
;--------------------------------------------------------------------
LOOP2:  TST   SDONE           ;S;Is SS sampling done?
        BNE   CONTA           ;S;If not, continue
        TST   FDONE           ;F;Is FP sampling done too?
        BNE   CONTA           ;F;If not, continue
        JMP   DISCON          ;  If both are done, disconnect and exit
CONTA:  TSTB  IOSTAT          ;S;any block I/O errors?
        BPL   CONTB           ;S
        JMP   WRTERR          ;S;if so flag them, and exit
CONTB:  TSTB  FOSTAT          ;F;any block I/O errors?
        BPL   CONTC           ;F
        JMP   FWRTER          ;F;if so flag them, and exit
CONTC:  BIT   #2,SSST         ;  test for STOP switch
        BEQ   LOOP2           ;  if not set yet, test again
```

```
; Stop sampling when STOP switch is set.  Mark end of data in the disk file.
;
ABORT:  CLR     WRDCNT              ;S:inhibit further sampling
        CLR     CNTRL               ;S
        MOV     #40000,SSDB         ;S:turn on STOP status light, stop sampling,
        CLR     FWRDCT              ;S:disable interrupts
        CLR     FCNTRL              ;F:inhibit further sampling
        MOV     #4,DHST             ;F:inhibit interrupts,inhibit SS frame starts
;
; Find number of words sampled before sampling was stopped
;
        MOV     SSWC,R1             ;S:save word count from this interrupted buffr
        NEG     R1                  ;S:SSWC is 2's comp. of words left to transfer
        MOV     #7680.,R2           ;S:normal full buffer size
        CMP     BUFTGL,NFBUF1       ;S:if not filling very last buffr when stopped
        BLT     CONT3               ;S:then use full buffer size minus SSWC
        MOV     NEXWRD,R2           ;S:if very last, use partial buffer size minus
        NEG     R2                  ;S:SSWC
CONT3:  SUB     R1,R2               ;S:buffer size minus SSWC is number of samples
        MOV     R2,NEXBYT           ;S
        ASL     NEXBYT              ;S;*2 for number of bytes
FDBK$R  #FDBOUT,,NEXBYT             ;S
        BIT     #1,BFWTGL           ;S
        BNE     ONELST              ;S
        WRITE$  #FDBOUT,#BUF1       ;S
        WAIT$   #FDBOUT,#7.         ;S
        BCC     CC                  ;S;error if C bit set
        JMP     WRERR               ;S;jump to error handling routine
CC:     BR      ENDLST              ;S
ONELST: WRITE$  #FDBCUT,#BUF0       ;S
        WAIT$   #FDBOUT,#7.         ;S
        BCC     ENDLST              ;S;error if C bit set
        JMP     WRERR               ;S;jump to error handling routine
```

```
ENDLST: INC     BFWTGL              ;S
        ADD     IOSTAT+2,BYTES      ;S
        ADC     BYTES+2             ;S
        JMP     DISCN1              ;S;disable interrupts and exit
;----------------------------------------------------------------
; Find number of FP words sampled before sampling was stopped
;----------------------------------------------------------------
;
        MOV     DRWC,R1             ;F;save word count from this interrupted buffr
        NEG     R1                  ;F;SSWC is 2's comp. of words left to transfer
        MOV     #768.,R2            ;F;normal full buffer size
        CMP     FBFTGL,FNFFF1       ;F;if not filling very last buffr when stopped
        BLT     FCONT3              ;F;then use full buffer size minus DRWC
        MOV     FNXWRD,R2           ;F;if very last, use partial buffer size minus
        NEG     R2                  ;F;DRWC
FCONT3: SUB     R1,R2               ;F;buffer size minus DRWC is number of samples
        MOV     R2,FNXBYT           ;F
        ASL     FNXBYT              ;F;*2 for number of bytes
        FDBK$R  #FPLFDB,,FNXBYT     ;F
        BIT     #1,FBFWTG           ;F
        BNE     F1LST               ;F
        WRITE$  #FPLFDB,#FPBUF1     ;F
        WAIT$   #FPLFDB,#20.        ;F
        BCC     FCC                 ;F;error if C bit set
        JMP     FWRERR              ;F;jump to error handling routine
FCC:    BR      F2LST               ;F
F1LST:  WRITE$  #FPLFDB,#FPBUF0     ;F
        WAIT$   #FPLFDB,#20.        ;F
        BCC     F2LST               ;F;error if C bit set
        JMP     FWRERR              ;F;jump to error handling routine
F2LST:  INC     FBFWTG              ;F
        ADD     FOSTAT+2,FBYTES     ;F
        ADC     FBYTES+2            ;F
        JMP     DISCN1              ;F;disable interrupts and exit
```

```
;----------------------------------------------------------------------
; Error reporting
;----------------------------------------------------------------------
CONERR: MOV     #-1,BUFTGL      ;S;connect to interrup error
        JMP     ALDONE          ;S
ASTERR: MOV     #-3,BUFTGL      ;S;ISR FORK2 queued AST exit error
        JMP     DISCON          ;S
OPERR:  MOV     #-4,BUFTGL      ;S;file open error
        JMP     ALDONE          ;S
WRERR:  MOV     #-5,BUFTGL      ;S;block disk write directive error
        JMP     DISCON          ;S
WRTERR: MOV     #-6,BUFTGL      ;S;block disk I/O error (flagged by IOSTAT)
        JMP     DISCON          ;S
WASTER: MOV     #-10,BUFTGL     ;S;I/O queued AST exit error
        JMP     DISCON          ;S
CLSERR: MOV     #-11,BUFTGL     ;S;file close error
        JMP     ALDONE          ;S
;----------------------------------------------------------------------
; Force Plate Error reporting
;----------------------------------------------------------------------
FCNERR: MOV     #-1,FBFTGL      ;F;connect to interrup error
        JMP     ALDONE          ;F
FASTER: MOV     #-3,FBFTGL      ;F;FPLISR FORK2 queued AST exit error
        JMP     DISCON          ;F
FOPERR: MOV     #-4,FBFTGL      ;F;file open error
        JMP     ALDONE          ;F
FWRERR: MOV     #-5,FBFTGL      ;F;block disk write directive error
        JMP     DISCON          ;F
FWRTER: MOV     #-6,FBFTGL      ;F;block disk I/O error (flagged by FOSTAT)
        JMP     DISCON          ;F
FWASTE: MOV     #-10,FBFTGL     ;F;I/O queued AST exit error
        JMP     DISCON          ;F
FCLSER: MOV     #-11,FBFTGL     ;F;file close error
        JMP     ALDONE          ;F
```

```
FHDWER: MOV    #-12,FBFTGL    ;F;FCEPLT hardware error
        MOV    DRST,FCNTRL    ;F
        CLR    DRST           ;F
        JMP    ALDONE         ;F
;----------------------------------------------------------------
; Finished, disable sampling, discon ect interrupt, wait for last buffer to
; write, clean up.
;----------------------------------------------------------------
DISCCN: CLR    WRDCNT         ;S;inhibit further sampling
        CLR    FWRDCT         ;F;inhibit further sampling
        MOV    #40200,SSDB    ;S;Turn on STOP light
LOOP4:  BIT    #2,SSST        ;test for STOP switch
        BEQ    CONT39         ;
        JMP    ABORT          ;
CONT39: TSTB   IOSTAT         ;S;any block I/O errors?
        BMI    WRTERR         ;S;if so flag them and exit
        TST    NEXBYT         ;S;is the last buffer size 0?
        BNE    CONT4          ;S;if not, continue
        CMP    FFWTGL,NFEUFS  ;S;if so, wait for last full buffer to write
        BLE    LOOP4          ;S;if STOP switch is not set
        JMP    FLOOP4         ;S;once written, check on FP buffer writes
CONT4:  CMP    BFWTGL,NFBUF1  ;S;wait for final disk write if STOF switch
        BLE    LOOP4          ;S;is not set
FLOCP4: BIT    #2,SSST        ;test for STOP switch
        BEQ    FLOP42         ;F
        JMP    ABORT          ;F;if set, stop sampling now
FLOP42: TSTB   FOSTAT         ;F;any block I/O errors?
        BMI    FWRTER         ;F;if so flag them and exit
        TST    FNXBYT         ;F;is the last buffer size 0?
        BNE    FCONT4         ;F;if not, continue
        CMP    FBFWTG,FNFBFS  ;F;if so, wait for last full buffer to write
        BLE    FLOOP4         ;F;if STOF switch is not set
        JMP    DISCN1         ;F;orce written, disconnect irterupts and exit
FCONT4: CMP    FBFWTG,FNFBF1  ;F;wait for final disk write if STOF switch
        BLE    FLOOP4         ;F;is not set
```

```
DISCN1: CLR     C2INT+C.INIS    ;S;clear ISR address in CINT$
        DIR$                    ;S;disconnect to the SS/DMA interrupt vect
CONT41: CLOSE$  #FDBOUT         ;S;close TK0:DR11BRL01.DAT
        BCC     FLOOP5          ;S;error if C bit set
        JMP     CLSERR          ;S;jump to error handling routine
FLOOP5: CLR     FC2INT+C.INIS   ;F;clear FPLISR address in CINT$
        DIR$    #FC2INT         ;F;disconnect the DMA interrupt vector
        CLOSE$  #FPLFDB         ;F;close TK1:FCEPLT.DAT
        BCC     FDD             ;F;error if C bit set
        JMP     FCLSER          ;F;jump to error handling routine
FDD:    MOV     DRST,FCNTRL     ;F;Save DR11-B CSR for F4P return
        CLR     DRST            ;F;Reset the interface
        CLR     R2              ;short wait to allow STOP status light to
WLOOP1: CLR     R1              ;turn on.
WLOOP2: INC     R1
        CMP     R1,#300.
        BLE     WLOOP2
        INC     R2
        CMP     R2,#500.
        BLE     WLOOP1
```

```
;
; Finished.  Write out return data to the FORTRAN main program.
;
        DEC     BUFTGL          ;S;BUFTGL-1 contains the number of interrupts
        DEC     FFFTGL          ;F;FFFTGL-1 contains the number of interrupts
ALDONE: CLR     SSDB            ;turn off all status lights
        MOV     #2,(R5)         ;Restore argument list for passage to FORTRAN
        MOV     LIST+2,2(R5)
        MOV     LIST+4,4(R5)
        MOV     LIST+6,6(R5)
        MOV     LIST+10,10(R5)
        MOV     LIST+12,12(R5)
        ADD     #2,LIST+12
        MOV     LIST+14,14(R5)
        MOV     LIST+16,16(R5)
        MOV     LIST+20,20(R5)
        MOV     LIST+22,22(R5)
        ADD     #2,LIST+22
        MOV     LIST+24,24(R5)
        MOV     CNTRL,@4(R5)    ;S;for diagnostics pass DMA control word back
        MOV     BUFTGL,@6(R5)   ;S
        MOV     IOSTAT,@10(R5)  ;S;IOSTAT contains the block WRITE error code
        MOV     BYTES,@12(R5)   ;S;BYTES and BYTES+2 contain the total # of
        MOV     BYTES+2,@LIST+12 ;S;bytes wrttn to the disk(double precision)
        MOV     FCNTRL,@14(R5)  ;F;for diagnostics pass DMA control word back
        MOV     FBFTGL,@16(R5)  ;F
        MOV     FOSTAT,@20(R5)  ;F;FOSTAT contains the block WRITE error code
        MOV     FBYTES,@22(R5)  ;F;FBYTES and FBYTES+2 contain the total # of
        MOV     FBYTES+2,@LIST+22 ;F;bytes wrttn to the disk(double precision)
        MOV     $DSW,@24(R5)    ;$DSW contains the directive status word
        RETURN                  ;return to calling program (usually T3SFTM)
```

```
; AST routine queued from the ISR FORK2 routine.

ASTRTN: TST     SDONE              ;S;is further sampling disabled?
        BNE     WRITEB             ;S;if not write out this buffer
        TST     NEXBYT             ;S;is the very last buffer size 0?
        BEQ     WRITEB             ;S;if not write out last full buffer
        CMP     BUFTGL,NFBUF1      ;S;has the last buffer not been reached?
        BLT     WRITEB             ;S;if not write out this last full buffer
        FDBK$R  #FDBOUT,,NEXBYT    ;S;set up to write last buffer
WRITEB: BIT     #1,BFWTGL          ;S;test which buffer to write
        BNE     ONEAST             ;S
        WRITE$  #FDBOUT,#BUF1      ;S;write out BUF1
        WAIT$   #FDBOUT,#7.        ;S
        BCC     EE                 ;S;error if C bit set
        MOVB    #-1,IOSTAT         ;S;error code =-1, undetermined I/O error
EE:     BR      ENDAST             ;S
ONEAST: WRITE$  #FDBOUT,#BUF0      ;S;write out BUF0
        WAIT$   #FDBOUT,#7.        ;S
        BCC     ENDAST             ;S;error if C bit set
        MOVB    #-1,IOSTAT         ;S;error code =-1, undetermined I/O error
ENDAST: INC     BFWTGL             ;S;increment BFWTGL to toggle buffers
        TST     (SP)+              ;S;Normal AST exit requires stack pop
        ASTX$S  ASTERR             ;S;AST exit directive (if error go to ASTERR)

; AST routine queued after each disk write completes.  It keeps a
; running total of bytes written to disk.

WRTAST: ADD     IOSTAT+2,BYTES     ;S;keep running total of disk bytes written
        ADC     BYTES+2            ;S;double precision result
        TST     (SP)+              ;S;must pop stack before AST exit
        ASTX$S  WASTER             ;S;AST exit directive
```

```
; AST routine queued from the FPLISR FORK2 routine.
;
;
FASTRT: TST     FDONE               ;F;is further sampling disabled?
        BNE     FWRITB              ;F;if not write out this buffer
        TST     FNXBYT              ;F;is the very last buffer size 0?
        BEQ     FWRITB              ;F;if not write out last full buffer
        CMP     FBFTGL,FNFBF1       ;F;has the last buffer not been reached?
        BLT     FWRITB              ;F;if not write out this last full buffer
        FDBK$R  #FFLFDB,,FNXBYT     ;F;set up to write last buffer
FWRITB: BIT     #1,FBFWTG           ;F;test which buffer to write
        BNE     F1AST               ;F
        WRITE$  #FPLFDB,#FPBUF1     ;F;write out FPBUF1
        WAIT$   #FPLFDB,#20.        ;F
        BCC     FEE                 ;F;error if C bit set
        MOVB    #-1,FOSTAT          ;F;error code =-1, undetermined I/O error
FEE:    BR      F2AST               ;F
F1AST:  WRITE$  #FFLFDB,#FPBUF0     ;F;write out FPBUF0
        WAIT$   #FPLFDB,#20.        ;F
        BCC     F2AST               ;F;error if C bit set
        MOVB    #-1,FOSTAT          ;F;error code =-1, undetermined I/O error
F2AST:  INC     FBFWTG              ;F;increment FBFTGL to toggle buffers
        TST     (SP)+               ;F;Normal AST exit requires stack pop
        ASTX$S  FASTER              ;F;AST exit directive (if error go to FASTER)
;
; AST routine queued after each disk write completes.  It keeps a
; running total of bytes written to disk.
;
FWRAST: ADD     FOSTAT+2,FBYTES     ;F;keep running total of disk bytes written
        ADC     FBYTES+2            ;F;double precision result
        TST     (SP)+               ;F;must pop stack before AST exit
        ASTX$S  FWASTE              ;F;AST exit directive
```

```
; _____
; Data
; _____
C2INT:  CINT$   124,ISR,0,ENADIS,PR7,ASTRTN  ;S;Connect to intrpt directive
FC2INT: CINT$   130,ISR,0,FFNADS,PR6,FASTRT  ;F;Connect to intrpt
LIST:   .BYTE   12,0                ;FORTRAN argument list storage area
        .BLKW   12
CNTRL:  .WORD   0                   ;S
NCHN:   .WORD   0                   ;S
NSKIP:  .WORD   0                   ;S
NEXBYT: .WORD   0                   ;S
NTBLKS: .WORD   0                   ;S
BYTES:  .BLKW   2                   ;S
NBYTES: .WORD   0                   ;S
BFWTGL: .WORD   1                   ;S
IOSTAT: .BLKW   2                   ;S
FNXBYT: .WORD   0                   ;F
FNTBLK: .WORD   0                   ;F
FBYTES: .BLKW   2                   ;F
FOSTAT: .BLKW   2                   ;F
FBFWTG: .WORD   1                   ;F
```

```
;
; Block disk I/O file name storage.
;

DATAST: .WORD    DVNMSZ,DEVNAM        ;S
        .WORD    DRNMSZ,DIRNAM        ;S
        .WORD    FLNMSZ,FILNAM        ;S
        .EVEN                         ;S
DEVNAM: .ASCII   /TK0:/               ;S
DVNMSZ=.-DEVNAM                       ;S
        .EVEN                         ;S
DIRNAM: .ASCII   //                   ;S
DRNMSZ=.-DIRNAM                       ;S
        .EVEN                         ;S
FILNAM: .ASCII   /SLSPOT.DAT/         ;S
FLNMSZ=.-FILNAM                       ;S
        .EVEN                         ;S

;
; FP Block disk I/O file name storage.
;

FDATAS: .WORD    FDVNMS,FDEVNM        ;F
        .WORD    FDRNMS,FDIRNM        ;F
        .WORD    FFLNMS,FFILNM        ;F
        .EVEN                         ;F
FDEVNM: .ASCII   /TK1:/               ;F
FDVNMS=.-FDEVNM                       ;F
        .EVEN                         ;F
FDIRNM: .ASCII   //                   ;F
FDRNMS=.-FDIRNM                       ;F
        .EVEN                         ;F
FFILNM: .ASCII   /FCEPLT.DAT/         ;F
FFLNMS=.-FFILNM                       ;F
        .EVEN                         ;F
```

```
;***********************************************************************
; SELSPOT Interrupt service routine
;***********************************************************************
        .PSECT  ISR,RW,D,GBL,REL,OVR    ;
ISR:    BIT     #1,BUFTGL               ;test toggle to determine which buff to fill
        BNE     ONE                     ;
        MOV     #52500,@#172412         ;move BUF0 address to SSBA
        MOV     WRDCNT,@#172410         ;# of words to transfer to SSWC
        INC     BUFTGL                  ;Increment BUFTGL to toggle buffers
        BR      DONE                    ;
ONE:    MOV     #110500,@#172412        ;move BUF1 address to SSBA
        MOV     WRDCNT,@#172410         ;# of words to transfer to SSWC
        INC     BUFTGL                  ;toggle buffers
DONE:   CMP     BUFTGL,NFBUFS           ;test for filling last full buffer
        BLT     ISREXT                  ;if not exit ISR
        BGT     VLAST                   ;if beyond last full, must be VERY last buff
        TST     NEXWRD                  ;if very last is empty
        BEQ     VLAST                   ;stop sampling
        MOV     NFXWRD,WRDCNT           ;if not, set up wordcount for next interrupt
        JMP     ISREXT                  ;
VLAST:  CLR     WRDCNT                  ;very last is filling,inhibit further sampling
        CMP     BUFTGL,NFBUF1           ;if interrupt AFTER very last
        BLE     ISREXT                  ;then:
        BIC     #100000,@#172416        ;Disable interrupts SSCB
        CLR     SDONE                   ;Clear SS sampling flag
ISREXT: CALL    @#$FORK2                ;FORK2 routine to allow directive execution
        CLR     @R3                     ;This is required for FORK2 routine
        CALL    @#$QASTC                ;Queue an AST
ENADIS: RTS     PC                      ;Return
BUFTGL: .WORD   1
WRDCNT: .WORD   0
NFBUFS: .WORD   0
NFBUF1: .WORD   0
NEXWRD: .WORD   0
SDONE:  .WORD   0
```

```
;****************************************************************************
; FORCE PLATE Interrupt service routine                                    *
;****************************************************************************
;
FPLISR: BIT    #1,FBFTGL          ;test toggle to determine which buff to fill *
        BNE    FONE               ;                                            *
        MOV    #44500,@#172432    ;move FPBUF0 address to DRBA                  *
        MOV    FWRDCT,@#172430    ;# of words to transfer to DRWC              *
        MOV    FCNTRL,@#172434    ;Int enb,ext addr,FNCT2,GO                   *
        INC    FBFTGL             ;Toggle buffers                              *
        BR     FPDONE             ;                                            *
FONE:   MOV    #47500,@#172432    ;move FPBUF1 address to DRBA                  *
        MOV    FWRDCT,@#172430    ;# of words to transfer to DRWC              *
        MOV    FCNTRL,@#172434    ;Int enb,ext addr,FNCT2,GO                   *
        INC    FBFTGL             ;Toggle buffers                              *
FPDONE: CMP    FBFTGL,FNFBFS      ;Is the last full buffer filling?            *
        BLT    FISREX             ;If not, exit the FPLISR                     *
        BGT    FVLAST             ;If beyond, the very last buffer is filling  *
        TST    FNXWRD             ;Is very last buffer size zero?              *
        BEQ    FVLAST             ;If so, stop sampling                        *
        MOV    FNXWRD,FWRDCT      ;If not, set up wordcount for next interrupt *
        JMP    FISREX             ;                                            *
FVLAST: MOV    #4,FCNTRL          ;Stop sampling on next interrupt             *
        CLR    FWRDCT             ;Stop sampling on next interrupt             *
        CMP    FBFTGL,FNFBF1      ;Is this the interrupt AFTER the very        *
        BLE    FISREX             ;last buffer?                                *
        CLR    FCNTRL             ;If so, stop all sampling and inhibit        *
        CLR    @#172434           ;interrupts (DRST)                           *
        CLR    FDONE              ;Clear FP sampling flag                      *
FISREX: CALL   @#$FORK2           ;FORK2 routine to allow directive execution  *
        CLR    @R3                ;This is required for FORK2 routine          *
        CALL   @#$QASTC           ;Queue an AST                                *
FENADS: RTS    PC                 ;Return                                      *
```

```
FBFTGL: .WORD    1
FWRDCT: .WORD    0
FCNTRL: .WORD    0
FNFBFS: .WORD    0
FNFBF1: .WORD    0
FNXWRD: .WORD    0
FDONE:  .WORD    0
;***********************************************************************
        .END
```

```
C     T3SFTM -- TRACK III SUBROUTINE FORTRAN TO MACRO TRANSLATE
C     T3SFTM
C     T3SFTM
C     T3SFTM
C
C              by Erik Antonsson          21-JUN-80
C-----------------------------------------------------------------
C     Copyright (C) 1980  Massachusetts Institute of Technology
C                         Cambridge, Massachusetts
C
C     This software is furnished under an agreement for use only on a
C     single computer system and may be copied only with the inclusion
C     of the above copyright notice.  This software, or any other copies
C     thereof, may not be provided or otherwise made available to any
C     other person except for use on such system and to one who subscribes
C     to these agreement terms.  Title to and ownership of the software
C     shall at all times remain in MIT.
C
C     The information in this document is subject to change without notice
C     and should not be construed as a commitment by MIT.
C
C     MIT assumes no responsibility for the use or reliability of its
C     software.
C-----------------------------------------------------------------
C     This routine translates the sampling parameters from the few
C     passed to it from FORTRAN to the many required by the macro
C     sampling routine T3SSM2.
C
C     The FORTRAN CALL is as follows:
C
C        CALL T3SFTM(LU1,LUO,NCHN,NFREQ,NFRAVE,NTBY)
C
```

```
C   Where:
C       LUI is the terminal input logical unit number
C       LUO is the printing output logical unit number
C       NCHN is the number of active LEDs. [integer:1<=NCHN<=30]
C           N.B. This programme re-adjusts the # of leds sampled
C           to be sure that a FRAME boundry and BLOCK (7680 words)
C           coincide.
C       NCHN is returned with the DMA control word [octal]
C       NFREQ is the desired sampling frequency [integer:1<=NFREQ<=315]
C           N.B. NFREQ can only take on certain values: 315/integer
C       NFREQ is returned with the number of buffers fully or partially
C           filled
C       NFRAME is the total number of frames to acquire
C       NFRAME is returned =-1 if the argument list was rejected
C       NFRAME is returned =-2 if an error occured during DMA sampling
C       NTBY is returned with the total number of bytes transferred
C           in a 2 element array (double integer precision)
C
C       The DMA data is written to the disk file TK0:SLSPOT.DAT
C       using logical unit number (LUN) 14. Note that this LUN 14 is
C       unavailable for use in the calling programs.
```

```
      SUBROUTINE T3SFTM(LUI,LUO,NCHN,NFREQ,NFRAME,NTBLKS)
      DIMENSION NTBLKS(2)
      BYTE IBYTE
3     IF(NCHN.LT.1)NCHN=1
      IF(NCHN.GT.30)NCHN=30
      NCH=NCHN
5     IF(IMOD(1920,NCH).EQ.0)GOTO 10
      NCH=NCH+1
      IF(NCH.GT.30)GOTO 3
      GOTO 5
10    IF(NFREQ.LT.2)NFREQ=2
      IF(NFREQ.GT.315)NFREQ=315
      IF(NFRAME.LT.1)NFRAME=1
      NSKIP=IFIX((315.0/FLOAT(NFREQ))+0.5)
      FREQ=315.0/FLOAT(NSKIP)
      WORDS=FLOAT(NFRAME)*FLOAT(NCH)*4
      NFBUFS=IFIX(WORDS/7680.)
      NEXWRD=IFIX(WORDS-(FLOAT(NFBUFS)*7680.))
      NTBLKS(1)=IFIX(WORDS/256.)
      RWORDS=FLOAT(NTBLKS(1))*256.
      IF(RWORDS.LT.WORDS)NTBLKS(1)=NTBLKS(1)+1
      WRITE(LUO,995)?
995   FORMAT(
     6'$Do you wish to sample with these parameters? [Y/N]:',A1)
      READ(LUI,997)IBYTE
997   FORMAT(A1)
      IF(IBYTE.NE.'Y'.AND.IBYTE.NE.'y')NFRAME=-1
      IF(NFRAME.LE.0)RETURN
      WRITE(IUO,999)?
999   FORMAT(/,' Ready...',A1)
      CALL T3SSM2(NCH,NSKIP,NFBUFS,NEXWRD,NTBLKS,IDSW)
      IF(NFBUFS.IT.0)GOTO 10
```

```fortran
C
C   Return sampling parameters to the calling program
C
      NFREQ=NFBUFS
      NCHN=NSKIP
      RETURN
C
C   Report errors
C
100   WRITE(LUO,1003)7,7,7,7
1003  FORMAT(1X,4A1)
      IF(NFBUFS.EQ.-1)WRITE(LUO,1005)
1005  FORMAT(/,'  *** DMA INTERRUPT CONNECT ERROR ***')
      IF(NFBUFS.EQ.-3)WRITE(LUO,1015)
1015  FORMAT(/,'  *** DMA ISR FORK2 QUEUED AST EXIT ERROR ***')
      IF(NFBUFS.EQ.-4)WRITE(LUO,1020)
1020  FORMAT(/,'  *** DMA FILE OPEN ERROR ***',/,
     *'  The most likely cause of this error is TK0: not MOUNTED')
      IF(NFBUFS.EQ.-5)WRITE(LUO,1025)
1025  FORMAT(/,'  *** DMA BLOCK DISK WRITE DIRECTIVE ERROR ***')
      IF(NFBUFS.EQ.-6)WRITE(LUO,1030)
1030  FORMAT(/,'  *** DMA BLOCK DISK I/O ERROR ***')
      IF(NEXWRD.EQ.-3)WRITE(LUO,1032)
1032  FORMAT(/,'  *** DISK BLOCK I/O DEVICE NOT REAY ***')
      IF(NFBUFS.EQ.-8)WRITE(LUO,1040)
1040  FORMAT(/,'  *** DMA I/O QUFUED AST EXIT ERROR ***')
      IF(NFBUFS.EQ.-9)WRITE(LUO,1045)
1045  FORMAT(/,'  *** DMA FILE CLOSE ERROR ***')
      WRITE(LUO,1055)NEXWRD,IDSW
1055  FORMAT(/,'  THE DMA IOSTAT BLOCK I/O ERROR CODE = ',I6,/,
     *'  See pages 5-7 through 5-9, or F-1 through B-3 of the',/,
     *'  I/O Drivers Reference Manual in Volume 5A RSX-11M V3.2',/,
     *'  THE DMA DIRECTIVE STATUS WORD = ',I6,/,
     *'  See Appendix B of the Executive Reference Manual in Volume',/,
     *'  3B RSX-11M V3.2')
      NFRAME=-2
      RETURN
      END
```

```
C    T3SSFP -- TRACK III SUBROUTINE FORTRAN TO MACRO TRANSLATE
C    T3SSFP
C    T3SSFP
C    T3SSFP
C
C            by Erik Antonsson      21-JUN-80
C
C    The FORTRAN CALL is as follows:
C
C    CALL T3SSFP(LUI,LUO,NCHN,NFREQ,NFRAME,NTBY)
C
C    Where:
C        LUI is the terminal input logical unit number
C        LUO is the printing output logical unit number
C        NCHN is the number of active LEDs. [integer:1<=NCHN<=30]
C            N.B. This programme re-adjusts the # of leds sampled
C            to be sure that a FRAME boundry and BLOCK (7680 words)
C            coincide.
C
C        NCHN is returned with the DMA control word [octal]
C        NFREQ is the desired sampling frequency [integer:1<=NFREQ<=315]
C            N.B. NFREQ can only take on certain values: 315/integer
C
C        NFREQ is returned with the number of buffers fully or partially
C            filled
C        NFRAME is the total number of frames to acquire
C        NFRAME is returned =-1 if the argument list was rejected
C        NFRAME is returned =-2 if an error occured during DMA sampling
C        NTBY is returned with the total number of bytes transferred
C            in a 2 element array (double integer precision)
C
C        The DMA data is written to the disk file TK0:SISPOT.DAT
C        using logical unit number (LUN) 14, and TK1:FC PLT.CAT
C        using logical unit number (LUN) 15.
C        Note that LUNS 14 and 15 are unavailable for use in the
C        calling programs.
C
```

```fortran
      SUBROUTINE T3SSFP(LUI,LUO,NCHN,NFREQ,NFRAME,NTBLKS)
      DIMENSION NTBLKS(2)
      INTEGER FNTBLK(2),FNFBFS,FNXWRD
      BYTE IBYTE
3     IF(NCHN.LT.1)NCHN=1
      IF(NCHN.GT.30)NCHN=30
      NCH=NCHN
5     IF(IMOD(1920,NCH).EQ.0)GOTO 10
      NCH=NCH+1
      IF(NCH.GT.30)GOTO 3
      GOTO 5
10    IF(NFREQ.LT.2)NFREQ=2
      IF(NFREQ.GT.315)NFREQ=315
      IF(NFRAME.LT.1)NFRAME=1
      NSKIP=IFIX((315.0/FLOAT(NFREQ))+0.5)
      FREQ=315.0/FLOAT(NSKIP)
      WORDS=FLOAT(NFRAME)*FLOAT(NCH)*4
      NFBUFS=IFIX(WORDS/7680.)
      NEXWRL=IFIX(WORDS-(FLOAT(NFBUFS)*7680.))
      NTBLKS(1)=IFIX(WORDS/256.)
      RWORDS=FLOAT(NTBLKS(1))*256.
      IF(RWORDS.LT.WORDS)NTBLKS(1)=NTBLKS(1)+1
C
      FNFBFS=NFRAME/96
      FNXWRL=(NFRAME-(FNFBFS*96))*8
      FNTBLK(1)=(FNFBFS*3)
      IF(FNXWRD.GT.0)FNTBLK(1)=FNTBLK(1)+1
      IF(FNXWRD.GT.256)FNTBLK(1)=FNTBLK(1)+1
      IF(FNXWRD.GT.512)FNTBLK(1)=FNTBLK(1)+1
      WRITE(5,2000)FNFBFS,FNXWRD,FNTBLK(1)
C2000 FORMAT(1X,3(I6,2X))
C
```

```
        WRITE(LUO,995)?
995     FORMAT(
     &'$Do you wish to sample with these parameters? [Y/N]:',A1)
        READ(LUI,997)IBYTE
997     FORMAT(A1)
        IF(IBYTE.NE.'Y'.AND.IBYTE.NE.'y')NFRAME=-1
        IF(NFRAME.LE.0)RETURN
        WRITE(LUO,999)?
999     FORMAT(/,' Ready...',A1)
        CALL T3SSM3(NCH,NSKIP,NFBUFS,NEXWRD,NTBLKS,
     *NCH,FNFBFS,FNXWRD,FNTBLK,IDSW)
        WRITE(LUO,2010)NCH,NSKIP,NFBUFS,NEXWRD,NTBLKS,
     *NCH,FNFBFS,FNXWRD,FNTBLK,IDSW
2010    FORMAT(1X,6(I6,2X),/,1X,6(I6,2X))
        IF(NFBUFS.LT.0)GOTO 10
        IF(FNFBFS.LT.0)GOTO 200
C
C  Return sampling parameters to the calling program
C
        NFREQ=NFBUFS
        NCHN=NSKIP
        RETURN
```

```
C
C  Report errors
C
100      WRITE(LUO,1003)7,7,7,7
1003     FORMAT(1X,4A1)
         IF(NFBUFS.EQ.-1)WRITE(LUO,1005)
1005     FORMAT(/,'    *** SELSPOT DMA INTERRUPT CONNECT ERROR  ***')
         IF(NFBUFS.EQ.-3)WRITE(LUO,1015)
1015     FORMAT(/,
        *'    *** SELSPOT DMA ISR FORK2 QUEUED AST EXIT ERROR  ***')
         IF(NFBUFS.EQ.-4)WRITE(LUO,1020)
1020     FORMAT(/,'    *** SELSPOT DMA FILE OPEN ERROR ***',/,
        *' The most likely cause of this error is TK0: not MOUTED')
         IF(NFBUFS.EQ.-5)WRITE(LUO,1025)
1025     FORMAT(/,
        *'    *** SELSPOT DMA BLOCK DISK WRITE DIRECTIVE ERROR  ***')
         IF(NFBUFS.EQ.-6)WRITE(LUO,1030)
1030     FORMAT(/,'    *** SELSPOT DMA BLOCK DISK DISK I/O ERROR  ***')
         IF(NEXWRD.EQ.-3)WRITE(LUO,1032)
1032     FORMAT(/,'    *** DISK BLOCK I/O DEVICE "TK0:" NOT READY  ***')
         IF(NFBUFS.EQ.-8)WRITE(LUO,1040)
1040     FORMAT(/,
        *'    *** SELSPOT DMA I/O QUEUED AST EXIT ERROR  ***')
         IF(NFBUFS.EQ.-9)WRITE(LUO,1045)
1045     FORMAT(/,'    *** SELSPOT DMA FILE CLOSE ERROR  ***')
         WRITE(LUO,1055)NEXWRD,IDSW
1055     FORMAT(/,
        *' THE SELSPOT DMA IOSTAT BLOCK I/O ERROR CODE = ',I6,/,
        *' See pages 5-7 through 5-9, or B-1 through B-3 of the',/,
        *' I/O Drivers Reference Manual in Volume 5A RSX-11M V3.2',/,
        *' THE DMA DIRECTIVE STATUS WORD = ',I6,/,
        *' See Appendix B of the Executive Reference Manual in Volume',/,
        *' 3B RSX-11M V3.2')
         NFRAME=-2
         IF(FNFBFS.IT.0)GOTO 200
         RETURN
```

```
200     WRITE(LUO,1003)7,7,7,7
        IF(FNFBFS.EQ.-1)WRITE(LUO,2005)
2005    FORMAT(/,'   *** FORCE PLATE DMA INTERRUPT CONNECT ERROR ***')
        IF(FNFBFS.EQ.-3)WRITE(LUO,2015)
2015    FORMAT(/,
      * '   *** FORCE PLATE DMA ISR FORK2 QUEUED AST EXIT ERROR ***')
        IF(FNFBFS.EQ.-4)WRITE(LUO,2020)
2020    FORMAT(/,'   *** FORCE PLATE DMA FILE OPEN ERROR ***',/,
      * ' The most likely cause of this error is TK1: not MOU TED')
        IF(FNFBFS.EQ.-5)WRITE(LUO,2025)
2025    FORMAT(/,
      * '   *** FORCE PLATE DMA BLOCK DISK WRITE DIRECTIVE ERROR ***')
        IF(FNFBFS.EQ.-6)WRITE(LUO,2030)
2030    FORMAT(/,'   *** FORCE PLATE DMA BLOCK DISK I/O ERROR ***')
        IF(FNXWRD.EQ.-3)WRITE(LUO,2032)
2032    FORMAT(/,'   *** DISK BLOCK I/O DEVICE "TK1:" NOT READY ***')
        IF(FNFBFS.EQ.-8)WRITE(LUO,2040)
2040    FORMAT(/,
      * '   *** FORCE PLATE DMA I/O QUEUED AST EXIT ERROR ***')
        IF(FNFBFS.EQ.-9)WRITE(LUO,2045)
2045    FORMAT(/,'   *** FORCE PLATE DMA FILE CLOSE ERROR ***')
        WRITE(LUO,2055)NEXWRD,IDSW
2055    FORMAT(/,
      * ' THE FORCE PLATE DMA IOSTAT BLOCK I/O ERROR CODE = ',I6,/,
      * ' See pages 5-7 through 5-9, or B-1 through B-3 of the',/,
      * ' I/O Drivers Reference Manual in Volume 5A RSX-11M V3.2',/,
      * ' THE DMA DIRECTIVE STATUS WORD = ',I6,/,
      * ' See Appendix E of the Executive Reference Manual in Volume',/,
      * '3E RSX-11M V3.2')
        NFRAME=-2
        RETURN
        END
```

Low-pass filtering and interpolation

Digital filters are usually coded such that the entire time history of one variable (as a function of time) must be memory resident during the filtering process. The TRACK III data usually has several (up to 120) different variables (as functions of time), making the problem of loading the data into memory all the more acute, if a relatively "long" data set has been accumulated. To get around the problem a "paging" filter was written. Essentially it reads in 50 frames of data for all of the variables, filters the 50 frames for each one, and saves the condition of the filter at the end of the 50 frames for each variable. The saved condition is held in a Fortran COMMON array. The page (or block) of 50 filtered frames is then written back out to the disk, and the next 50 frames are read in. Since the condition of the filter at the end of the previous page has been saved for each variable the filter can be started again (for each variable) as if the previous 50 frames for that variable had been in memory. This paging process continues through the whole data set.

The problem is made more complex by two things. First digital filters (just as analog ones) don't like to encounter data gaps. A gap in the data (e.g. if an LED was occluded) will make the filter "ring" (large amplitude oscillations at approximately the cutoff frequency of the filter). To avoid gaps the data is always linearly interpolated before filtering. Interpolators are also usually coded to have all of one time history of one variable in memory at once. Again, a paging interpolator was written to execute before filtering.

An additional reason for moving to a paging filter and interpolator, even if a single variable's complete time history could be fit in memory, is that the entire data file would have to be read and written as many times as there are variables to pick out each one separately. This results in a filter routine that is extremely I/O inefficient, the paging filter and interpolator reads and writes the data set 3 times, once for each "pass."

The action of the interpolator is as follows:

1. The parameters of the filter are "designed". (T3SFLD.FTN) This routine is directly from Stearns (1975) [150].

2. A page of 50 frames of all variables is read in. (T3SIFR.FTN)

3. Linear interpolation occurrs within the page and the endpoints of the data gaps crossing page boundries are saved in a data file. (T3SFPI.FTN) First pass interpolator.

4. The page is written out. This occurs for the whole data set.

5. The inter-page gap endpoint data file is manipulated so that correct data is available even if a gap is more than 2 pages long. (T3SIIC.FTN)

6. Starting at the beginning again, a page of 50 frames of data is read in.

7. The inter-page data gaps are linearly interpolated using a page of data and the data file containing the endpoints of the gaps. (T3SSPI.FTN) Second pass interpolator.

8. In order to minimize the amount of disk I/O, once a page has gone through the second pass interpolator, while it is still in memory, the first pass of the low pass filter is applied. (The filter also has 2 passes, one forward and one backward in time, to eliminate all phase shift). (T3S2PF.FTN)

9. The page is written out.

10. Subsequent pages are treated by the second pass interpolator and the first pass of the filter.

11. The second pass of the filter is applied, paging through the data set from back to front.

```
C   T3BPFT -- TRACK III 3-D DATA INTERPOLATE AND FILTER
C   T3BPFT
C   T3BPFT
C   T3BPFT
C
C      25-SEP-80      by Erik Antonsson
C
C   Copyright (C) 1980  Massachusetts Institute of Technology
C                        Cambridge, Massachusetts
C
C   This software is furnished under an agreement for use only on a
C   single computer system and may be copied only with the inclusion
C   of the above copyright notice.  This software, or any other copies
C   thereof, may not be provided or otherwise made available to any
C   other person except for use on such system and to one who subscribes
C   to these agreement terms.  Title to and ownership of the software
C   shall at all times remain in MIT.
C
C   The information in this document is subject to change without notice
C   and should not be construed as a commitment by MIT.
C
C   MIT assumes no responsibility for the use or reliability of its
C   software.
C
C   This program linearly interpolates bad (occluded or eliminated) data
C   and then filters the 3-D TRACK III data.
C
C   Logical Unit 14 is the data file to be filtered, TK4:T3B002.DAT
C
C   The filtered data is written back into the same file to save disk space.
C
```

```fortran
      COMMON/BLKFLD/A(3),B(3),C(3),GRAF(2,10)
      LUO=5
      OPEN(UNIT=14,NAME='TK4:T3B002.DAT;0',ACCESS='DIRECT',TYPE='OLD')
C
C     Read post processing parameters from the data file
C
      READ(14'2)RNCHN,FRAMES,FREQ,RNSEG
      NCHN=IFIX(RNCHN)
      READ(14'9)R91,R92,R93,R94,R95,R96,RPC,R98,R99
      IF(RPC.NE.0.0)GOTO 300
      READ(14'10)RLINF1,RLINF2,SMPTOP,CUTOFF,RMPTOP,RUTOFF,RINTOP,
     *SINTOP
      NFRAME=IFIX(FRAMES)
      NFREQ=IFIX(FREQ)
      NSEG=IFIX(RNSEG)
      ICUTOF=IFIX(CUTOFF)
C
C     If neither raw interpolation nor filtering is desired, stop
C
      IF(SMPTOP.EQ.1.0)SINTOP=1.0
      IFILT=IFIX(SMPTOP)
      IF(SMPTOP.NE.1.0.AND.SINTOP.NE.1.0)GOTO 3
      IF(IFILT.EQ.1)RPC=1.0
      IF(IFILT.EQ.1)WRITE(14'9)R91,R92,R93,R94,R95,R96,RPC,R98,R99
      IF(IFILT.EQ.1)WRITE(LUO,1000)CUTOFF
1000  FORMAT(' Interpolate and low-pass filter (',F5.1,
     &' Hz Cutoff) the 3-D data ')
      IF(IFILT.NE.1.AND.SINTOP.EQ.1.0)WRITE(LUO,1010)
1010  FORMAT(' Interpolate the 3-D data ')
```

```
C
C     Design the filter
C
      CUTOFA=CUTOFF
      T=1.0/FREQ
      NS=3
      IF(IFILT.EQ.1)CALL T3SFLD(CUTOFA,T,NS,A,B,C,GRAF)
C
C     Interpolate obscured points and filter the file
C
      IW=3
      CALL T3SIFR(14,NCHN,NFRAME,IW,IFILT)
    3 CLOSE(UNIT=14)
      CALL EXST(1)
  700 WRITE(LUO,1050)7,7,7,7
 1050 FORMAT(' *** TK4:T3B002.DAT HAS ALREADY BEEN FILTERED ***',4A1)
      CLOSE(UNIT=14)
      CALL EXST(2)
      END
```

```
C     T3SFLD -- TRACK III SUBROUTINE FILTER LOW-PASS DESIGN
C     T3SFLD
C     T3SFLD
C     T3SFLD
C
C     THIS ROUTINE IS FROM:
C       STEARNS,S.D., DIGITAL SIGNAL ANALYSIS,
C       HAYDEN BOOK CO. INC., 1975.
C       COPYRIGHT (C) 1975, STEARNS,S.D.
C
C     LOWPASS BUT EKWORTH DIGITAL FILTER DESIGN SUBROUTINE
C     INPUTS ARE CUTOFF (3-DB) FREQUENCY FC IN HERTZ,
C     SAMPLING INTERVAL T IN SECONDS, AND
C     NUMBER NS OF FILTER SECTIONS.
C     OUTPUTS ARE NS SETS OF FILTER COEFFICIENTS, I.E.,
C       A(K) THRU C(K) FOR K=1 THRU NS, AND
C       10 PAIRS OF FREQUENCY AND POWER GAIN, I.,E.,
C       GRAF(1,K) AND GRAF(2,K) FOR K=1 THRU 10.
C     NOTE THAT A(K),H(K),C(K) AND GRAF(2,10) MUST BE
C     DIMENSIONED IN THE CALLING PROGRAM.
C
C     THE DIGITAL FILTER HAS NS SECTIONS IN CASCADE.    THE KTH
C       SECTION HAS THE TRANSFER FUNCTION
C
C                   A(K)*(Z**2+2*Z+1)
C       H(Z)=---------------------------
C                   Z**2+B(K)*Z+C(K)
C
C     THUS IF F(M) AND G(M) ARE THE INPUT AND OUTPUT OF THE
C       KTH SECTION AT TIME M*T, THEN
C
C       G(M)=A(K)*(F(M)+2*F(M-1)+F(M-2))-B(K)*G(M-1)
C             -C(K)*G(M-2)
C
```

```
      SUBROUTINE T3SFLD(FC,T,NS,A,B,C,GRAF)
      DIMENSION A(3),B(3),C(3),GRAF(2,10)
      RPI=4.0*ATAN(1.0)
      DENOM=COS(FC*RPI*T)
      IF(DENOM.EQ.0.) GO TO 3
      WCP=SIN(FC*RPI*T)/DENOM
      DO 1 K=1,NS
      CS=COS(FLOAT(2*(K+NS)-1)*RPI/FLOAT(4*NS))
      X=1./(1.+WCP*WCP-2.*WCP*CS)
      A(K)=WCP*WCP*X
      B(K)=2.*(WCP*WCP-1.)*X
      C(K)=(1.+WCP*WCP+2.*WCP*CS)*X
    1 CONTINUE
      DO 2 K=1,10
      GRAF(2,K)=.99-.98*FLOAT(K-1)/9.
      X=ATAN(WCP*(1./GRAF(2,K)-1.)**(1./FLOAT(4*NS)))
      GRAF(1,K)=X/(RPI*T)
    2 CONTINUE
      RETURN
    3 TYPE 1000, '7,'7,'7,'7
 1000 FORMAT(' *** ERROR!!! -- CUTOFF FREQUENCY SELECTED HAS CAUSED
     * A FLOATING ',/,' ZERO DIVIDE.  SELECT ANOTHER SLIGHTLY DIFFERENT
     * FREQUENCY ***',4A1)
      CALL EXST(2)
      END
```

```
C   T3SIFR -- TRACK III SUBROUTINE INTERPOLATE AND FILTER ROUTINE
C   T3SIFR
C   T3SIFR
C   T3SIFR
C
C      18-SEP-80        by Erik Antonsson
C ----------------------------------------------------------------
C   Copyright (C) 1980  Massachusetts Institute of Technology
C                       Cambridge, Massachusetts
C
C   This software is furnished under an agreement for use only on a
C   single computer system and may be copied only with the inclusion
C   of the above copyright notice.  This software, or any other copies
C   thereof, may not be provided or otherwise made available to any
C   other person except for use on such system and to one who subscribes
C   to these agreement terms.  Title to and ownership of the software
C   shall at all times remain in MIT.
C
C   The information in this document is subject to change without notice
C   and should not be construed as a commitment by MIT.
C
C   MIT assumes no responsibility for the use or reliability of its
C   software.
C ----------------------------------------------------------------
C
```

```fortran
C
C   A two pass interpolator (to allow very long data sets to be
C   interpolated) is cascaded with a two pass digital filter
C   (for the same long data set) resulting in a three pass routine.
C   The data set is divided up into 50 frame blocks and
C   they are processed one at a time and the transition parameters
C   are saved in a scratch file... This read/writing of the blocks
C   is performed 3 times (hence ...three pass).  The data file can be
C   in either of two cannonical TRACK III forms: raw data with
C   X1,X2,Y1,Y2 for each channel in each frame, or 3-D data with X,Y,Z
C   for each channel in each frame.  Only the logical unit number of the
C   open data file need be passed to identify the proper data to interpolate
C   and filter.  N.B. The routine also uses the previous logical unit
C   number PLUS 1 for the scratch data file needed during processing.
C
C   IW=3 for 3-D (X,Y,Z) data
C   IW=4 for Raw (X1,X2,Y1,Y2) data
C
C   IFILT=0 for Interpolation only
C   IFILT=1 for Interpolation and Filtering
C
      SUBROUTINE T3SIFR(LU,NCHN,NFRAME,IW,IFILT)
      COMMON/BLKFLD/A(3),B(3),C(3),GRAF(2,10)
      COMMON/BLKINT/DATA(120,50),BDG(120,4,3),NF,NFT,NC,IFIFTY
      COMMON/BLKTMP/LFIFTY
      DO 104 IQ=1,120
      DO 101 JQ=1,50
      DATA(IQ,JQ)=0.0
101   CONTINUE
      DO 103 JQ=1,4
      DO 102 KQ=1,3
      BDG(IQ,JQ,KQ)=0.0
102   CONTINUE
103   CONTINUE
104   CONTINUE
```

```
C
C     IW=4 for raw data, IW=3 for 3-D point data
C
      NC=NCHN*IW
      NFT=NFRAME/50
      LFIFTY=MOD(NFRAME,50)
      IF(LFIFTY.NE.0)NFT=NFT+1
      IF(LFIFTY.EQ.0)LFIFTY=50
      LUS=LU+1
      CALL ASSIGN(LUS,'TK4:INTFLT.SCR;0',16,'SCR')
C
C     The first 12 records are for the filter parameter storage
C     all others are for the interpolater.
C
      NRECS=(NFT+5)*4
      NWORDS=NC*2
      DEFINE FILE LUS(NRECS,NWORDS,U,IR)
      ZERO=0.0
      WRITE(LUS'NRECS)(ZERO,I=1,NC)
C
C     KC indicates the 1st or 2nd pass
C
      DO 17 KC=1,2
      DO 16 NF=1,NFT
      IF(KC.EQ.1)GOTO 5
C
C     Read in second pass block endpoint data
C
      DO 1 K1=1,3
      DO 1 K=1,4
      READ(LUS'(NF+K1+1)*4+K)(BDG(I,K,K1),I=1,NC)
    1 CONTINUE
```

```
C
C      Read in 1 block of 50 frames
C
 5         IFIFTY=50
           IF(NF.EQ.NFT.AND.MOD(NFRAME,50).NE.0)IFIFTY=MOD(NFRAME,50)
           DO 6 J=1,IFIFTY
           J2=(J+((NF-1)*50))+10
           READ(LU'J2)(DATA(I,J),I=1,NC)
 6         CONTINUE
           IF(KC.EQ.2)GOTO 7
C
C      First pass interpolator (Intra-block)
C
           CALL T3SFPI
           GOTO 13
C
C      Second pass interpolator (Inter-block)
C
 7         CALL T3SSPI
C
C      First filter pass
C
           IF(IFILT.EQ.0.0)GOTO 13
           IF(NF.NE.1)GOTO 9
           DO 8 I=1,NC
           DO 8 K1=1,3
           DO 6 K=1,4
           BDG(I,K,K1)=DATA(I,1)
 8         CONTINUE
           GOTO 11
 9         DO 10 K1=1,3
           DO 10 K=1,4
           READ(LUS'(K1-1)*4+K)(BDG(I,K,K1),I=1,NC)
 10        CONTINUE
```

```fortran
11      CALL T3S2PF(1)
        IF(NF.EQ.NFT)GOTO 13
        DO 12 K1=1,3
        DO 12 K=1,4
        WRITE(LUS'(K1-1)*4+K)(BDG(I,K,K1),I=1,NC)
12      CONTINUE
C
C       Write data out
C
13      DO 14 J=1,IFIFTY
        J2=(J+((NF-1)*50))+10
        WRITE(LU'J2)(DATA(I,J),I=1,NC)
14      CONTINUE
C
C       If after first pass, write out 2nd pass block endpoint data
C
        IF(KC.EQ.2)GOTO 16
        DO 15 K=1,4
        WRITE(LUS'((NF+3)*4)+K)(BDG(I,K,1),I=1,NC)
15      CONTINUE
16      CONTINUE
C
C       Correct the Inter-Block interpolator index file to work with whole blocks
C       of bad data.
C
        NF=LUS
        IF(KC.EQ.1)CALL T3SIIC
17      CONTINUE
C
C       Second filter pass
C
        IF(IFILT.EQ.0.0)GOTO 25
        DO 22 NF=NFT,1,-1
```

```
C
C     Read in a block of data (the order the blocks are read in is reversed)
C
          IFIFTY=50
          IF(NF.EQ.NFT.AND.MOD(NFRAME,50).NE.0)IFIFTY=MOD(NFRAME,50)
          DO 18 J=1,IFIFTY
          J2=(J+((NF-1)*50))+10
          READ(LU'J2)(DATA(I,J),I=1,NC)
18        CONTINUE
          IF(NF.NE.NFT)GOTO 20
          DO 19 I=1,NC
          DO 19 K1=1,3
          DO 19 K=1,4
          BDG(I,K,K1)=DATA(I,IFIFTY)
19        CONTINUE
20        CALL T3S2PF(2)
C
C     Write out this block
C     The filter parameters (BDG) are held in COMMON
C
          DO 21 J=1,IFIFTY
          J2=(J+((NF-1)*50))+10
          WRITE(LU'J2)(DATA(I,J),I=1,NC)
21        CONTINUE
22        CONTINUE
25        CLOSE(UNIT=LUS,DISPOSE='DELETE')
          RETURN
          END
```

```
C   T3SFPI -- TRACK III SUBROUTINE, FIRST PASS INTERPOLATOR
C   T3SFPI
C   T3SFPI
C   T3SFPI
C
C       18-SEP-80        by Erik Antonsson
C   '''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
C
C   Copyright (C) 1980  Massachusetts Institute of Technology
C                           Cambridge, Massachusetts
C
C   This software is furnished under an agreement for use only on a
C   single computer system and may be copied only with the inclusion
C   of the above copyright notice.  This software, or any other copies
C   thereof, may not be provided or otherwise made available to any
C   other person except for use on such system and to one who subscribes
C   to these agreement terms.  Title to and ownership of the software
C   shall at all times remain in MIT.
C
C   The information in this document is subject to change without notice
C   and should not be construed as a commitment by MIT.
C
C   MIT assumes no responsibility for the use or reliability of its
C   software.
C   '''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
C
```

```
      SUBROUTINE T3SFPI
      COMMON/BLKINT/DATA(120,50),BDG(120,4,3),NF,NFT,NC,IFIFTY
      DO 6 I=1,NC
      DO 1 K=1,4
      BDG(I,K,1)=0.0
1     CONTINUE
C
C     Find the first and last good data points for 2nd pass
C
      DO 2 J=1,IFIFTY
      IF(ABS(DATA(I,J)).GT.512.0)GOTO 2
      IF(BDG(I,1,1).EQ.0.0)BDG(I,1,1)=FLOAT(J)
      BDG(I,3,1)=FLOAT(J)
2     CONTINUE
      IF(BDG(I,1,1).EQ.0.0)GOTO 6
      BDG(I,2,1)=DATA(I,IFIX(BDG(I,1,1)))
      BDG(I,4,1)=DATA(I,IFIX(BDG(I,3,1)))
C
C     Interpolate within the block
C
      INDEX=0
      DO 5 J=IFIX(BDG(I,1,1)),IFIX(BDG(I,3,1))
      IF(ABS(DATA(I,J)).GT.512.0)GOTO 4
      IF(INDEX.EQ.0)GOTO 5
      DAT=(DATA(I,J)-DATA(I,(J-(INDEX+1))))/FLOAT(INDEX+1)
      DO 3 K=1,INDEX
      DATA(I,(J-INDEX+K-1))=DATA(I,(J-(INDEX+1)))+(DAT*K)
3     CONTINUE
      INDEX=0
4     INDEX=INDEX+1
5     CONTINUE
6     CONTINUE
      RETURN
      END
```

```
C     T3SIIC -- TRACK III SUBROUTINE, INTERPOLATOR INDEX CORRECTOR
C     T3SIIC
C     T3SIIC
C
C         15-JAN-82        by Erik Antonsson
C
C     ----------------------------------------------------------------
C
C         Copyright (C) 1982  Massachusetts Institute of Technology
C                           Cambridge, Massachusetts
C
C     This  software  is  furnished   under an  agreement  for use  only on a
C     single computer  system and may    be copied  only with  the inclusion
C     of the  above copyright notice.     This  software, or any other copies
C     thereof, may not be  provided   or otherwise made available  to any
C     other person except for use on such system and to one who subscribes
C     to these agreement terms.   Title to and  ownership of the software
C     shall at all times remain in MIT.
C
C     The information in this document is subject to change without notice
C     and should not be construed as a commitment by MIT.
C
C     MIT  assumes   no responsibility  for the use  or reliability  of its
C     software.
C     ----------------------------------------------------------------
C
C
      SUBROUTINE T3SIIC
      COMMON/BLKINT/DATA(120,50),BDG(120,4,3),NF,NFT,NC,IFIFTY
      LUS=NF
```

```
C
C     First forwards through the BDG file to correct the "last good data
C     point" indicies and values.
C
      ZERO=0.0
      DO 8 NF=0,3
      DO 8 K=1,4
8     WRITE(LUS'((NF*4)+K))(ZERO,I=1,NC)
      ONE=1.0
      DO 9 K=1,4
9     WRITE(LUS'(((NFT+4)*4)+K))(ONE,I=1,NC)
C
C     K1=1 is previous block
C     K1=2 is current block
C
      DO 70 NF=1,NFT+1
      DO 10 K1=1,2
      DO 10 K=1,4
      READ(LUS'(NF+K1+1)*4+K)(BDG(I,K,K1),I=1,NC)
10    CONTINUE
      DO 50 I=1,NC
      IF(BDG(I,3,2).NE.0.0)GOTO 50
      BDG(I,3,2)=BDG(I,3,1)-50.0
      BDG(I,4,2)=BDG(I,4,1)
50    CONTINUE
      DO 60 K=1,4
      WRITE(LUS'(NF+3)*4+K)(BDG(I,K,2),I=1,NC)
60    CONTINUE
70    CONTINUE
```

```
C
C    Then Backwards through the BDG file to correct the "first good data
C    point" indicies and values.
C
     DO 170 NF=NFT+1,1,-1
     DO 110 K1=1,2
     DO 110 K=1,4
     READ(LUS'(NF+K1+1)*4+K)(BDG(I,K,K1),I=1,NC)
110  CONTINUE
     DO 150 I=1,NC
     IF(BDG(I,1,1).NE.0.0)GOTO 150
     BDG(I,1,1)=BDG(I,1,2)+50.0
     BDG(I,2,1)=BDG(I,2,2)
150  CONTINUE
     DO 160 K=1,4
     WRITE(LUS'(NF+2)*4+K)(BDG(I,K,1),I=1,NC)
160  CONTINUE
170  CONTINUE
C
     RETURN
     END
```

```
C     T3SSPI -- TRACK III SUBROUTINE, SECOND PASS INTERPOLATOR
C     T3SSPI
C     T3SSPI
C     T3SSPI
C
C     ------------------------------------------------------------
C       18-SEP-80       by Erik Antonsson
C     ------------------------------------------------------------
C
C     Copyright (C) 1980  Massachusetts Institute of Technology
C                         Cambridge, Massachusetts
C
C     This software is furnished under an agreement for use only on a
C     single computer system and may be copied only with the inclusion
C     of the above copyright notice.  This software, or any other copies
C     thereof, may not be provided or otherwise made available to any
C     other person except for use on such system and to one who subscribes
C     to these agreement terms.  Title to and ownership of the software
C     shall at all times remain in MIT.
C
C     The information in this document is subject to change without notice
C     and should not be construed as a commitment by MIT.
C
C     MIT assumes no responsibility for the use or reliability of its
C     software.
C     ------------------------------------------------------------
C
      SUBROUTINE T3SSPI
      COMMON/ELKINT/DATA(120,50),ELG(120,4,3),NF,NFT,NC,IFIFTY
      COMMON/ELKTMP/LFIFTY
```

```
C
C   Fill in data gap at beginning of the block (if present)
C
      DO 2 I=1,NC
      IF(BDG(I,1,2).LE.1.0)GOTO 2                    !Gap at beginning?
      SPNI=BDG(I,3,1)-(BDG(I,1,2)+50.0)              !If yes, find gap length
      IF(ABS(SPNI).LT.1.0)GOTO 2                     !If gap length.le.1 loop
      SPN=FDG(I,4,1)-BLG(I,2,2)                      !Find gap bndry good data diff
      DINC=SPN/SPNI                    !Delta increment for interpolation
      IBGN=IFIX(EDG(I,1,2))            !Index of 1st good data wrt this block
      IF(IBGN.GT.IFIFTY)DOFSET=(BDG(I,1,2)-50)*DINC  !If gap strtd bef beg
      IF(IBGN.LE.IFIFTY)DOFS T=0.0     !If gap started in this block
      IETST=((NF-1)*50)+IFIX(EDG(I,3,1))             !Gap to very beginning?
      IF(IETST.LE.0)DOFSET=0.0         !If yes, reset to make all gap values
      IF(IETST.LE.0)DINC=0.0           !equal to the first good value
      IETST=(((NF+1)-NFT)*50-IFIFTY)+IFIX(EDG(I,1,3))
      IF(NF.EC.NFT)IETST=IFIX(BDG(I,1,2))-IFIFTY
      IF(IETST.GT.0.AND.IFGN.GT.IFIFTY)GOTO 2
      IF(IBGN.GT.IFIFTY)IBGN=IFIFTY    !Reset interp beg to end of block
      DATA(I,IBGN)=EDG(I,2,2)-DOFSET   !Set end of block to beg interp
      DO 1 J=IBGN-1,1,-1               !Interp from end of gap to beg of blok
      IF(IBGN.IT.1)GOTO 1
      DATA(I,J)=DATA(I,J+1)-DINC
    1 CONTINUE
    2 CONTINUE
```

```
C
C      Fill in data gap at end of block (if present)
C

       DO 4 I=1,NC
       IF(BDG(I,3,2).GE.FLOAT(IFIFTY))GOTO 4
       SPNI=(BDG(I,1,3)+50.0)-BDG(I,3,2)
       IF(ABS(SPNI).LT.1.0)GOTO 4
       SPN=EDG(I,2,3)-BDG(I,4,2)
       DINC=SPN/SPNI
       IBGN=IFIX(BDG(I,3,2))
       IF(IBGN.LT.1)DOFSET=(BDG(I,3,2)+50)*DINC
       IF(IBGN.GE.1)DOFSET=0.0
       IF(IBGN.LT.1)IBGN=1
       IETST=(((NF+1)-NFT)*50-LFIFTY)+IFIX(BDG(I,1,3))
       IF(NF.EQ.NFT)IETST=IFIX(BDG(I,1,2))-IFIFTY
       IF(BDG(I,1,2).GE.FLOAT(IFIFTY).AND.IETST.LE.0)GOTO 4
       IF(IETST.GT.0)DOFSET=0.0
       IF(IETST.GT.0)DINC=0.0
       DATA(I,IBGN)=BDG(I,4,2)+DOFSET
       DO 3 J=IBGN+1,IFIFTY
       IF(IEGN.GT.IFIFTY)GOTO 3
       DATA(I,J)=DATA(I,J-1)+DINC
3      CONTINUE
4      CONTINUE
       RETURN
       END
```

```
C  T3S2PF -- TRACK III SUBROUTINE, 2 PASS FILTER
C  T3S2PF
C  T3S2PF
C  T3S2PF
C
C       18-SEP-80      by Erik Antonsson
C
C  Copyright (C) 1980  Massachusetts Institute of Technology
C                      Cambridge, Massachusetts
C
C  This software is furnished under an agreement for use only on a
C  single computer system and may be copied only with the inclusion
C  of the above copyright notice.  This software, or any other copies
C  thereof, may not be provided or otherwise made available to any
C  other person except for use on such system and to one who subscribes
C  to these agreement terms.  Title to and ownership of the software
C  shall at all times remain in MIT.
C
C  The information in this document is subject to change without notice
C  and should not be construed as a commitment by MIT.
C
C  MIT assumes no responsibility for the use or reliability of its
C  software.
C
      SUBROUTINE T3S2PF(IPASS)
      COMMON/BLKFLC/A(3),B(3),C(3),GRAF(2,10)
      COMMON/BLKINT/LATA(120,50),F(120,4,3),NF,NFT,NC,IFIFTY
      ISTRT=1
      ISTOP=IFIFTY
      ISTEP=1
      IF(IPASS.EQ.1)GOTO 1
      ISTRT=IFIFTY
      ISTOP=1
      ISTEP=-1
```

```
1       DO 5  I=1,NC
        DO 4  J=ISTRT,ISTOP,ISTEP
        F(I,1,3)=DATA(I,J)
        DO 2  N=1,3
        TEMP=A(N)*(F(I,N,3)+2.@*F(I,N,2)+F(I,N,1))
        F(I,N+1,3)=TEMP-B(N)*F(I,N+1,2)-C(N)*F(I,N+1,1)
2       CONTINUE
        DO 3  N=1,4
        DO 3  MM=1,2
3       F(I,N,MM)=F(I,N,MM+1)
        DATA(I,J)=F(I,4,3)
4       CONTINUE
5       CONTINUE
        RETURN
        END
```

First and Second Derivatives

The routine originally used to calculate the first and second derivatives with respect to time of the kinematic data, required that an entire time history of each variable be memory resident. Since the method only uses 5 points along the time history at any one time, a new routine was coded to take advantage of that, and use 5 points of every time history at once. In order to maximize efficiency of data I/O, 9 points are kept in memory at once for each time history, so that once 5 first derivative data points have been calculated, they can in-turn be used to calculate the second derivative. This results in a routine that calculates velocity and accelerations and requires only one pass through the data set. Complications arise in getting the "pyrimad" calculations started, special sections of the code handle both startup and finish to provide reliable derivatives to the ends of the data set.

```
C     T3BACL -- TRACK III BATCH VELOCITY AND ACCELERATION CALCULATION
C     T3BACL
C     T3BACL
C     T3BACL
C
C         4-DEC-80      by Erik Antonsson
C
C-------------------------------------------------------------------
C
C         Copyright (C) 1981  Massachusetts Institute of Technology
C                             Cambridge, Massachusetts
C
C     This software is furnished  under an  agreement  for use  only on a
C     single computer system and may  be copied only with  the inclusion
C     of the  above copyright notice.   This software, or any other copies
C     thereof, may  not be  provided  or otherwise  made available  to any
C     other person except for use on such system and to one who subscribes
C     to these agreement terms.   Title to and  ownership of the software
C     shall at all times remain in MIT.
C
C     The information in this document is subject to change without notice
C     and should not be construed as a commitment by MIT.
C
C     MIT  assumes  no responsibility  for the  use  or reliability  of its
C     software.
C------------------------------------------------------------------
C
C     This routine uses the Lagrangian interpolation polynomial of degree
C     4 relevant to 5 succesive points.  See Hildebrand, F.B.,
C     Introduction to Numerical Analysis, McGraw-Hill, New York,
C     1956, pp. 82-84.  See also routine DET5 from the IBM System/360
C     Scientific Subroutine Package Version III.  The central difference
C     between DET5 and T3BACL is that the former works on a single dimensional
C     array (vector) and the latter works on a multi-dimensional data
C     file.
C     Note that the cartesian angles are stored in the position data file as
C     degrees, and are converted to radians before differentiation.
C
```

```fortran
      DIMENSION D(7,6,10),E(5,6,10),F(4,6,10)
C
C Set up input data result file
C
      PI=4.0*ATAN(1.0)
      DTR=PI/180.0
      LUO=5
      WRITE(LUO,1000)
1000  FORMAT(' Calculate the first 2 derivatives
     & (velocities and accelerations)')
      OPEN(UNIT=10,NAME='TK4:T3E003.DAT',ACCESS='DIRECT',TYPE='OLD',
     *READONLY)
      READ(10'2)NCHN,FRAMES,FREQ,RNSEG
      NFRAME=IFIX(FRAMES)
      NSEG=IFIX(RNSEG)
      NREC=NFRAME+10
      NWORD=12*NSEG*2
C
C Set up output data result acceleration file and transfer parameter header
C
      CALL ASSIGN(9,'TK4:T3B003.ACL',14,'NEW')
      DEFINE FILE 9(NREC,NWORD,U,IR)
      DO 3 J=1,10
      READ(10'J)(((D(I,K,L),I=1,2),K=1,6),L=1,NSEG)
      WRITE(9'J)(((D(I,K,L),I=1,2),K=1,6),L=1,NSEG)
3     CONTINUE
      READ(10'9)R91,R92,R93,R94,R95,R96,RPC,R98,R99
      RPC=0.0
      WRITE(9'9)R91,R92,R93,R94,R95,R96,RPC,R98,R99
```

```fortran
C
C     Read initial data points for startup
C
      HH=FREQ/12.0
      DO 10 J=1,7
      READ(10'J+10)( (D(J,K,L),K=1,6),(DUM,K=1,10)),L=1,NSEG)
      DO 8 L=1,NSEG
      DO 8 K=4,6
      D(J,K,L)=D(J,K,L)*DTR
    8 CONTINUE
   10 CONTINUE
C
C     Startup sequence
C
      DO 20 L=1,NSEG
      DO 19 K=1,6
      E(1,K,L)=HH*(-25.*D(1,K,L)+48.*D(2,K,L)-36.*D(3,K,L)+
     *16.*D(4,K,L)-3.*D(5,K,L))
      E(2,K,L)=HH*(-3.*D(1,K,L)-10.*D(2,K,L)+18.*D(3,K,L)-6.*D(4,K,L)
     *+D(5,K,L))
      E(3,K,L)=HH*(D(1,K,L)-D(5,K,L)+8.*(D(4,K,L)-D(2,K,L)))
      E(4,K,L)=HH*(D(2,K,L)-D(6,K,L)+8.*(D(5,K,L)-D(3,K,L)))
      E(5,K,L)=HH*(D(3,K,L)-D(7,K,L)+8.*(D(6,K,L)-D(4,K,L)))
      F(1,K,L)=HH*(-25.*E(1,K,L)+48.*E(2,K,L)-36.*E(3,K,L)+
     *16.*E(4,K,L)-3.*E(5,K,L))
      F(2,K,L)=HH*(-3.*E(1,K,L)-10.*E(2,K,L)+18.*E(3,K,L)-6.*E(4,K,L)
     *+E(5,K,L))
   19 CONTINUE
   20 CONTINUE
      WRITE(9'11)((E(1,K,L),K=1,6),(F(1,K,L),K=1,6),L=1,NSEG)
      WRITE(9'12)((E(2,K,L),K=1,6),(F(2,K,L),K=1,6),L=1,NSEG)
```

```fortran
C
C     Main single and double differentiation loop
C
      DO 100 J=7,NFRAME
      READ(10'J+10)(((D(7,K,L),K=1,6),(DUM,K=1,10)),L=1,NSEG)
      DO 80 L=1,NSEG
      DO 79 K=1,6
      IF(K.GT.3)D(7,K,L)=D(7,K,L)*DTR
      E(5,K,L)=HH*(D(3,K,L)-D(7,K,L)+8.*(D(6,K,L)-D(4,K,L)))
      F(2,K,L)=HH*(E(1,K,L)-E(5,K,L)+8.*(E(4,K,L)-E(2,K,L)))
      DO 75 I=1,6
      D(I,K,L)=D((I+1),K,L)
75    CONTINUE
      DO 76 I=1,4
      E(I,K,L)=E((I+1),K,L)
76    CONTINUE
79    CONTINUE
80    CONTINUE
      WRITE(9'J+6)((E(2,K,L),K=1,6),(F(2,K,L),K=1,6),L=1,NSEG)
100   CONTINUE
```

```
C
C   End sequence
C
      DO 140 L=1,NSEG
      DO 138 K=1,6
      E(5,K,L)=HH*(-D(2,K,L)+6.*D(3,K,L)-18.*D(4,K,L)+10.*D(5,K,L)+
     *3.*D(6,K,L))
      F(1,K,L)=HH*(E(1,K,L)-E(5,K,L)+8.*(E(4,K,L)-E(2,K,L)))
      DO 120 I=1,5
      D(I,K,L)=D((I+1),K,L)
120   CONTINUE
      DO 121 I=1,4
      E(I,K,L)=E((I+1),K,L)
121   CONTINUE
      E(5,K,L)=HH*(3.*D(1,K,L)-16.*D(2,K,L)+36.*D(3,K,L)-
     *48.*D(4,K,L)+25.*D(5,K,L))
      F(2,K,L)=HH*(E(1,K,L)-E(5,K,L)+8.*(E(4,K,L)-E(2,K,L)))
      F(3,K,L)=HH*(-E(1,K,L)+6.*E(2,K,L)-18.*E(3,K,L)+10.*E(4,K,L)+
     *3.*E(5,K,L))
      F(4,K,L)=HH*(3.*E(1,K,L)-16.*E(2,K,L)+36.*E(3,K,L)-
     *48.*E(4,K,L)+25.*E(5,K,L))
138   CONTINUE
140   CONTINUE
      DO 150 I=1,4
      WRITE(9'NFRAME+6+I)((E((I+1),K,L),K=1,6),(F(I,K,L),K=1,6),
     *L=1,NSEG)
150   CONTINUE
      CALL CLOSE(9)
      CALL CLOSE(10)
      CALL EXST(1)
      END
```

Dynamics estimator

```
C    FORCE1 -- INERTIAL FORCE AND MOMENT ESTIMATES FOR 1 LINK
C    FORCE1
C    FORCE1
C
C         APRIL 4,1978      by Erik Antonsson
C         16-FEB-82         Modifi d by Erik Antonsson to run on the 11/60
C
C    This programme takes position and orientation data,
C    and the first and second derivatives, and the inertial properties
C    of the linkage element, and estimates the forces and
C    moments that must have impinged on the link for the
C    observed motion to have occured.
C
C    FCEPLT Plots the estimator's results 22-APR-82
C
      REAL MASS,INERT(3),IW(3),IWD(3)
      DIMENSION UD(3),UDD(3),WE(3),W(3),G(3),RP(3),RD(3),
     &SEGROT(9),SEGRIT(9),TEMP1(3),TEMP2(3),DUM(10),
     &WCIW(3),RDCFD(3),RPCFP(3),
     &FP(3),FD(3),TP(3),TD(3),FPG(3),FDG(3),TPG(3),TDG(3),
     &NAMFIL(9),NAMACL(9),NAMFPD(9),NAMFCE(9),NDESCR(20)
      DATA NAMACL/'TK',4:' ,'00','00','A','CL',;0',"0/
      DATA NAMFPD/'TK',4:' ,'00','00','F','PD',;0',"0/
      DATA NAMFCE/'TK',4:' ,'00','00','F','CE',"0',0/
      IU=14
      LUI=5
      LUO=5
      WRITE(LUO,993)
993   FORMAT(' Solve the inverse dynamics (1 segment) ')
      LUPR=6
      IR=2
      CALL T3SDSH(LU,LUI,LUO,LUPR,IR,NAMFIL,NDESCR)
```

```fortran
      WRITE(LUO,995)
995   FORMAT(' Working...',/)
      READ(14'2)RNCHN,FRAMES,FREQ,RNSEG,FPISCL
      NFRAME=IFIX(FRAMES)
      NSEG=IFIX(RNSEG)
      IF(NSEG.EQ.1)GOTO 8
      WRITE(LUO,1000)7,7,7
1000  FORMAT(' *** ERROR, THE NUMBER OF SEGMENTS IS NOT "1" ***',
     &4A1)
8     CALL EXST(2)
      DO 10 I=3,5
      NAMACL(I)=NAMFIL(I)
      NAMFPD(I)=NAMFIL(I)
      NAMFCE(I)=NAMFIL(I)
10    CONTINUE
      OPEN(UNIT=15,NAME=NAMACL,ACCESS='DIRECT',TYPE='OLD',
     *READONLY)
      OPEN(UNIT=16,NAME=NAMFPD,ACCESS='DIRECT',TYPE='OLD',
     *READONLY)
      NRECS=NFRAME+10
      NWORD=24*2
      CALL ASSIGN(17,NAMFCE,14,'NEW')
      DEFINE FILE 17(NRECS,NWORD,U,IR)
C
C     Transfer first 10 records from NAMFIL to NAMFCE
C
      DO 12 I=1,10
      READ(14'I)(DUM(J),J=1,10)
      WRITE(17'I)(DUM(J),J=1,10)
12    CONTINUE
```

```fortran
C
C     Assign gravity field
C
      G(1)=0.0
      G(2)=-9.8
      G(3)=0.0
C
C     Assign inertia, mass [kg] and geometry here.
C     Moments of inertias must be about the C.G.
C
1002  WRITE(LUO,1002)7
      FORMAT('$Enter the Mass. [Default: 5.8851] (fl.pt): ',A1)
      READ(LUI,1004)MASS
1004  FORMAT(F14.7)
      IF(MASS.LE.0.0)MASS=5.8851
      INERT(1)=0.7910
      INERT(2)=0.00429
      INERT(3)=0.7910
      RP(1)=0.0
      RP(2)=0.645
      RP(3)=0.0
C
C     Read data
C
      DO 800 K=1,NFRAME
      KREC=K+10
      READ(14'KREC)X,Y,Z,AX,AY,AZ,DUM(7),
     *(SEGR1T(I),I=1,9)
C
C     SEGR1T is the transpose of SEGROT because the rotation
C     matrix is stored row-wise
C
      CALL GMTRA(SEGR1T,SEGROT,3,3)
```

```
C
C      Read accelerations from "namfil.ACL"
C      in this form: XD,YD,ZD,W1,W2,W3,XDD,YDD,ZDD,WD1,WD2,WD3
C
       READ(15'KREC)(UD(I),I=1,3),(TEMP1(I),I=1,3),
      *(UDD(I),I=1,3),(TEMP2(I),I=1,3)
C
C      Change reference frames of rotation data to the BCS
C
       CALL GMPRD(SEGROT,TEMP1,W,3,3,1)
       CALL GMPRD(SEGROT,TEMP2,WD,3,3,1)
C
C      Read forceplate forces in the GCS from "namfil.FPD"
C      in this form: FX,FY,FZ,FAX,MY,FAZ
C
       READ(16'KREC)(FDG(I),I=1,3),FAX,TDG(2),FAZ
       TDG(1)=0.0
       TDG(3)=0.0
       DO 330 L=1,3
       FDG(L)=-FDG(L)
330    CONTINUE
       TDG(2)=-TDG(2)
C
C      Calculate the vector to the forceplate center of force
C      in the BCS
C
       TEMP1(1)=FAX-X
       TEMP1(2)=-Y
       TEMP1(3)=FAZ-Z
       CALL GMPRD(SEGROT,TEMP1,RD,3,3,1)
```

```
C     Process to yeild forces and moments
C
C     All of the following are 3 element arrays; the elements
C     are assigned as follows: X,Y,Z.  i.e. RD(i)= the Distal
C     endpoint vector's X component, RD(2)= the Y component, etc.
C
C     FPG is the force on the Proximal end in the GCS
C     FDG is the force on the Distal end in the GCS
C     TPG is the torque at the Proximal end in the GCS
C     TDG is the torque at the Distal end in the GCS
C     UDD is the acceleration of the CG in the GCS
C
C     FP is the force on the Proximal end in the BCS
C     FD is the force on the Distal end in the BCS
C     TP is the torque at the Proximal end in the BCS
C     TD is the torque at the Distal end in the BCS
C     RP is the vector from the CG to the Proximal end in the BCS
C     RD is the vector from the CG to the Distal end in the BCS
C     W is the angular velocity about the CG in the BCS
C     WD is the angular acceleration about the CG in the BCS
C
      DO 350 L=1,3
      FPG(L)=MASS*UDD(L)-MASS*G(L)-FDG(L)
350   CONTINUE
      CALL GMPRD(SEGROT,FPG,FP,3,3,1)
      CALL GMPRD(SEGRCT,FDG,FD,3,3,1)
      CALL GMPRD(SEGROT,TPG,TD,3,3,1)
      CALL CROSSP(RD,FD,RDCFD)
      CALL CROSSP(RP,FP,RPCFP)
      DO 400 L=1,3
      IW(L)=INERT(L)*W(L)
400   CONTINUE
      CALL CROSSP(W,IW,WCIW)
      DO 500 L=1,3
      TP(L)=-(INERT(L)*WD(L))-WCIW(L)+RDCFD(L)+TD(L)+RPCFP(L)
500   CONTINUE
```

```fortran
C
C  Change reference frames to ground (Global)
C
      DO 600 I=1,3
      TEMP2(I)=-TP(I)
600   CONTINUE
      CALL GMPRD(SEGR1T,TEMP2,TPG,3,3,1)
      WRITE(LUC,1200)K,FPG,TPG
1200  FORMAT('+',I4,3F9.3,2X,3F9.3)
      WRITE(17'KREC)
     &(FD(I),I=1,3),(TD(I),I=1,3),
     &(FP(I),I=1,3),(TP(I),I=1,3),
     &(FDG(I),I=1,3),(TDG(I),I=1,3),
     &(FPG(I),I=1,3),(TPG(I),I=1,3)
800   CONTINUE
      CLOSE(UNIT=14)
      CLOSE(UNIT=15)
      CLOSE(UNIT=16)
      CALL CLOSE(17)
      CALL EXST(2)
      END
```

```
C     FORCE4 -- INERTIAL FORCE AND MOMENT ESTIMATES FOR 4 LINKS
C     FORCE4
C     FORCE4
C
C     APRIL 4,1978     by Erik Antonsson
C     16-FEB-82        Modified by Erik Antonsson to run on the 11/60
C     23-APR-82        Modified by Erik Antonsson to analyze 4 links
C     12-MAY-82        Modified by Erik Antonsson to negate Proximal Torques
C
C ----------------------------------------------------------------------
C
C        Copyright (C) 1982  Massachusetts Institute of Technology
C                            Cambridge, Massachusetts
C
C     This  software  is  furnished   under an   agreement   for use  only on a
C     single computer system and may   be copied  only with   the  inclusion
C     of the  above copyright notice.    This software,  or any other  copies
C     thereof, may  not be  provided   or otherwise made available   to any
C     other person except for  use on such system and to one who subscribes
C     to these  agreement terms.    Title to and   ownership of the software
C     shall at all times remain in MIT.
C
C     The information in this document is subject to change without notice
C     and should not be construed as a commitment by MIT.
C
C     MIT  assures    no  responsibility  for the use   or reliability   of its
C     software.
C ----------------------------------------------------------------------
C
C     This program takes position and orientation data,
C     and the first and second derivatives, and the inertial properties
C     of three linkage elements, and estimates the forces and
C     moments that must have impinged on the link proximal
C     end points for the
C     observed motion to have occured.
C
C     FCEPLT Plots the estimator's results 22-APR-82
C     FORCEP Plots the estimator's results  6-MAY-82
C
```

```fortran
      REAL MASS1,INERT1(3),RP1(3),RD1(3),RCG1(3),RCGG1(3),
     &FP1(3),FD1(3),TP1(3),TD1(3),FPG1(3),FDG1(3),TPG1(3),TDG1(3)
      REAL MASS2,INERT2(3),RP2(3),RD2(3),RCG2(3),RCGG2(3),
     &FP2(3),FD2(3),TP2(3),TD2(3),FPG2(3),FDG2(3),TPG2(3),TDG2(3)
      REAL MASS3,INERT3(3),RP3(3),RD3(3),RCG3(3),RCGG3(3),
     &FP3(3),FD3(3),TP3(3),TD3(3),FPG3(3),FDG3(3),TPG3(3),TDG3(3)
      REAL MASS4,INERT4(3),RP4(3),RD4(3),RCG4(3),RCGG4(3),
     &FP4(3),FD4(3),TP4(3),TD4(3),FPG4(3),FDG4(3),TPG4(3),TDG4(3)
      REAL IW(3),IWD(3)
      DIMENSION UD(3),UDD(3),WD(3),W(3),
     &WCIW(3),RDCFD(3),RPCFP(3),
     &SEGROT(9),SEGR1T(9),TEMP1(3),TEMP2(3),TEMP3(3),TEMP4(3),
     &NAMFIL(9),NAMACL(9),NAMFPD(9),NAMFCE(8),G(3),DUM(10),NIESCR(20)
      DATA NAMACL/'TK',4:' ','00',' ','00',' ','A',' ','CL',' ';0',' ';0/
      DATA NAMFPD/'TK',4:' ','00',' ','00',' ','F',' ','PD',' ';0',' ';0/
      DATA NAMFCE/'TK',4:' ','00',' ','00',' ','F',' ','CE',' ',0/
      LUO=5
      WRITE(LUO,993)
993   FORMAT(' Solve the inverse dynamics (4 segments) ')
      LU=14
      LUI=5
      LUPR=8
      IR=2
      CALL T3SDSH(LU,LUI,LUO,LUPR,IR,NAMFIL,NDESCR)
      WRITE(LUO,995)
995   FORMAT(' Working...',/)
      READ(14',2)RNCHN,FRAMES,FREQ,RNSEG,FPLSCL
      NFRAME=IFIX(FRAMES)
      NSEG=IFIX(RNSEG)
      IF(NSEG.GE.4)GOTO 8
      WRITE(LUO,1000)7,7,7,7
1000  FORMAT(' *** ERROR, THE NUMBER OF SEGMENTS IS NOT .GE. 4 ***',
     &4A1)
      CALL EXST(2)
```

```
8     DO 10 I=3,5
      NAMACL(I)=NAMFIL(I)
      NAMFPD(I)=NAMFIL(I)
      NAMFCE(I)=NAMFIL(I)
10    CONTINUE
      OPEN(UNIT=15,NAME=NAMACL,ACCESS='DIRECT',TYPE='OLD',
     *READONLY)
      OPEN(UNIT=16,NAME=NAMFPD,ACCESS='DIRECT',TYPE='OLD',
     *READONLY)
      NRECS=NFRAME+10
      NWORD=24*2*4
      CALL ASSIGN(17,NAMFCE,14,'NEW')
      DEFINE FILE 17(NRECS,NWORD,U,IR)
C
C     Transfer first 10 records from NAMFIl to NAMFCE
C
      DO 12 I=1,10
      READ(14'I)(DUM(J),J=1,10)
      WRITE(17'I)(DUM(J),J=1,10)
12    CONTINUE
C
C     Assign gravity field
C
      G(1)=0.0
      G(2)=-9.8
      G(3)=0.0
```

```
C
C     Assign mass [kg] and moments of inertia [kg-m**2]
C     Moments of inertias must be about the C.G.
C
      OPEN(UNIT=12,NAME='INERTIA.DAT;0',TYPE='OLD',READONLY)
      READ(12,1002)MASS1
      READ(12,1002)INERT1(1)
      READ(12,1002)INERT1(2)
      READ(12,1002)INERT1(3)
      READ(12,1002)MASS2
      READ(12,1002)INERT2(1)
      READ(12,1002)INERT2(2)
      READ(12,1002)INERT2(3)
      READ(12,1002)MASS3
      READ(12,1002)INERT3(1)
      READ(12,1002)INERT3(2)
      REAL(12,1002)INERT3(3)
      CLOSE(UNIT=12)
1002  FORMAT(F14.7)
C
C     Assign geometry here.  RP is the vector from the CG to the proximal
C     end of the segment in the BCS.
C     RD is the vector from the CG to the distal end of the segment.
C
      OPEN(UNIT=12,NAME='GEOMETRY.DAT;0',TYPE='OLD',READONLY)
C
C     Read Proximal vector in the BCS
C
      READ(12,1002)RP1(1)
      READ(12,1002)RP1(2)
      READ(12,1002)RP1(3)
      REAL(12,1002)RP2(1)
      READ(12,1002)RP2(2)
      REAL(12,1002)RP2(3)
      READ(12,1002)RP3(1)
      REAL(12,1002)RP3(2)
      READ(12,1002)RP3(3)
```

```
C
C   Read Distal vector in the BCS
C

      READ(12,1002)RD1(1)
      READ(12,1002)RD1(2)
      READ(12,1002)RD1(3)
      READ(12,1002)RD2(1)
      READ(12,1002)RD2(2)
      READ(12,1002)RD2(3)
      READ(12,1002)RD3(1)
      READ(12,1002)RD3(2)
      READ(12,1002)RD3(3)

C
C   Read Center of Mass (CG) vector in the BCS
C

      READ(12,1002)RCG1(1)
      READ(12,1002)RCG1(2)
      READ(12,1002)RCG1(3)
      READ(12,1002)RCG2(1)
      READ(12,1002)RCG2(2)
      READ(12,1002)RCG2(3)
      READ(12,1002)RCG3(1)
      READ(12,1002)RCG3(2)
      READ(12,1002)RCG3(3)
      CLOSE(UNIT=12)
```

```
C
C     All of the following are 3 element arrays, the elements
C     are assigned as follows: X,Y,Z.  i.e. RD(1)= the Distal
C     endpoint vector's X component, RD(2)= the Y component, etc.
C
C     FPG is the force on the Proximal end in the GCS
C     FDG is the force on the Distal end in the GCS
C     TPG is the torque at the Proximal end in the GCS
C     TDG is the torque at the Distal end in the GCS
C     UD  is the translational velocity of the CG in the GCS
C     UDD is the translational acceleration of the CG in the GCS
C
C     FP  is the force on the Proximal end in the BCS
C     FD  is the force on the Distal end in the BCS
C     TP  is the torque at the Proximal end in the BCS
C     TD  is the torque at the Distal end in the BCS
C     RP  is the vector from the CG to the Proximal end in the BCS
C     RD  is the vector from the CG to the Distal end in the BCS
C     RCG is the vector from the origin of the BCS to the CG
C     W   is the angular velocity about the CG in the BCS
C     WD  is the angular acceleration about the CG in the BCS
C
C     Read data:
C     Only need X,Y,Z for the link on the FP
C
      DO 800 K=1,NFRAME
      KREC=K+10
      READ(14'KREC)X,Y,Z,AX,AY,AZ,DUM(7),
     *(SEGR1T(I),I=1,9)
C
C     SEGR1T is the transpose of SEGROT because the rotation
C     matrix is stored row-wise
C
      CALL GMTRA(SEGR1T,SEGROT,3,3)
```

```
C
C     Move the BCS origin to the CG
C
      DO 110 I=1,3
110   TEMP1(I)=RCGG1(I)
      CALL GMPRD(SEGR1T,TEMP1,TEMP2,3,3,1)
      DO 115 I=1,3
115   RCGG1(I)=TEMP2(I)
      X=X+RCGG1(1)
      Y=Y+RCGG1(2)
      Z=Z+RCGG1(3)
C
C -----------------------  Segment #1  ------------------------
C
C     Read accelerations from "namfil.ACL"
C     in this form: XD,YD,ZD,W1,W2,W3,XDD,YDD,ZDD,WD1,WD2,WD3
C
      READ(15'KREC)(UD(I),I=1,3),(TEMP1(I),I=1,3),
     *(UDD(I),I=1,3),(TEMP2(I),I=1,3)
C
C     Readjust translational velocity and acceleration to be about the CG
C     (still in the GCS)
C
C     Velocity = Velocity + (Omega cross R)
C
      CALL CROSSP(TEMP1,RCGG1,TEMP3)
      DO 120 I=1,3
120   UD(I)=UD(I)+TEMP3(I)
C
C     Translational acceleration will be adjusted in two steps:
C     Acceleration = Acceleration + (Omega cross (Omega cross R))
C
      CALL CROSSP(TEMP1,TEMP3,TEMP4)
      DO 122 I=1,3
122   UDD(I)=UDD(I)+TEMP4(I)
```

```
C
C      Acceleration = Acceleration + (Alpha cross R)
C
       CALL CROSSP(TEMP2,RCGG1,TEMP4)
       DO 124 I=1,3
124    UDD(I)=UDD(I)+TEMP4(I)
C
C      Change reference frames of rotation data to the BCS
C
       CALL GMPRD(SEGROT,TEMP1,W,3,3,1)
       CALL GMPRD(SEGROT,TEMP2,WD,3,3,1)
C
C      Read forceplate forces in the GCS from "namfil.FPD"
C      in this form: FX,FY,FZ,FAX,MY,FAZ
C
       READ(16'KREC)(FDG1(I),I=1,3),FAX,TDG1(2),FAZ
       TDG1(1)=0.0
       TDG1(3)=0.0
       DO 130 I=1,3
       FDG1(L)=-FDG1(L)
130    CONTINUE
       TDG1(2)=-TDG1(2)
C
C      Calculate the vector to the forceplate center of force
C      in the BCS
C
       TEMP1(1)=FAX-X
       TEMP1(2)=-Y
       TEMP1(3)=FAZ-Z
       CALL GMPRD(SEGROT,TEMP1,RD1,3,3,1)
```

```
C
C      Process to yeild forces and moments
C

          DO 150 L=1,3
          FPG1(L)=MASS1*UDD(L)-MASS1*G(L)-FDG1(L)
150       CONTINUE

C
C      Change forces and torque into BCS
C

          CALL GMPRD(SEGROT,FPG1,FP1,3,3,1)
          CALL GMPRD(SEGROT,FDG1,FD1,3,3,1)
          CALL GMPRD(SEGROT,TDG1,TD1,3,3,1)
          CALL CROSSP(RD1,FD1,RDCFD)
          CALL CROSSP(RP1,FP1,RPCFP)
          DO 160 L=1,3
          IW(L)=INERT1(L)*W(L)
160       CONTINUE
          CALL CROSSP(W,IW,WCIW)
          DO 170 L=1,3
          TP1(L)=-(INERT1(L)*WD(L))-WCIW(L)+RDCFD(L)+TD1(L)+RPCFP(L)
170       CONTINUE

C
C      Change reference frames to ground (GCS)
C

          DO 180 I=1,3
          TP1(I)=-TP1(I)
180       CONTINUE
          CALL GMPRD(SEGR1T,TP1,TPG1,3,3,1)
```

```
C
C
C---------------    Segment #2    ---------------
C
C
C     Change the vector to the CG to the GCS
C
C
         DO 210 I=1,3
210      TEMP1(I)=RCG2(I)
         CALL GMPRD(SEGR1T,TEMP1,TEMP2,3,3,1)
         DO 215 I=1,3
215      RCGG2(I)=TEMP2(I)
C
C     Read accelerations from "namfil.ACL"
C     in this form: XL,YD,ZD,W1,W2,W3,WD1,WD2,WD3
C
         READ(15'KREC)(DUMMY,I=1,12),(UD(I),I=1,3),(TEMP1(I),I=1,3),
     *(UDD(I),I=1,3),(TEMP2(I),I=1,3)
C
C     Readjust translational velocity and acceleration to be about the CG
C     (still in the GCS)
C
C     Velocity = Velocity + (Omega cross R)
C
         CALL CROSSP(TEMP1,RCGG2,TEMP3)
         DO 220 I=1,3
220      UD(I)=UD(I)+TEMP3(I)
C
C     Translational acceleration will be adjusted in two steps:
C     Acceleration = Acceleration + (Omega cross (Omega cross R))
C
         CALL CROSSP(TEMP1,TEMP3,TEMP4)
         DO 222 I=1,3
222      UDD(I)=UDD(I)+TEMP4(I)
```

```
C
C     Acceleration = Acceleration + (Alpha cross R)
C
      CALL CROSSP(TEMP2,RCGG2,TEMP4)
      DO 224 I=1,3
224   UDD(I)=UDD(I)+TEMP4(I)
C
C     Change reference frames of rotation data to the BCS
C
      CALL GMPRD(SEGROT,TEMP1,W,3,3,1)
      CALL GMPRD(SEGROT,TEMP2,WD,3,3,1)
C
C     Determine the Distal forces from the previous link's proximal forces
C
      DO 230 L=1,3
      FDG2(L)=-FPG1(L)
      TDG2(L)=-TPG1(L)
230   CONTINUE
C
C     Process to yeild forces and moments
C
      DO 250 L=1,3
      FPG2(L)=MASS2*UDD(L)-MASS2*G(L)-FDG2(L)
250   CONTINUE
```

```
C
C     Change forces and torque into BCS
C
      CALL GMPRD(SEGROT,FPG2,FP2,3,3,1)
      CALL GMPRD(SEGROT,FDG2,FD2,3,3,1)
      CALL GMPRD(SEGROT,TDG2,TD2,3,3,1)
      CALL CROSSP(RD2,FD2,RDCFD)
      CALL CROSSP(RP2,FP2,RPCFP)
      DO 260 L=1,3
      IW(L)=INERT2(L)*W(L)
260   CONTINUE
      CALL CROSSP(W,IW,WCIW)
      DO 270 L=1,3
      TP2(L)=-(INERT2(L)*WD(L))-WCIW(L)+RDCFD(L)+TD2(L)+RPCFP(L)
270   CONTINUE
C
C     Change reference frames to ground  (GCS)
C
      DO 280 I=1,3
      TP2(I)=-TP2(I)
280   CONTINUE
      CALL GMPRD(SEGR1T,TP2,TPG2,3,3,1)
```

```
C
C   -------------------------------    Segment #3    -------------------------------
C
C   Change the vector to the CG to the GCS
C
         DO 310 I=1,3
316      TEMP1(I)=RCG2(I)
         CALL GMPRD(SEGR1T,TEMP1,TEMP2,3,3,1)
         DO 315 I=1,3
315      RCGG2(I)=TEMP2(I)
C
C   Read accelerations from "namfil.ACL"
C   in this form: XD,YD,ZD,W1,W2,W3,XDD,YDD,ZDD,WD1,WD2,WD3
C
         READ(15'KREC)(DUMMY,I=1,24),(UD(I),I=1,3),(TEMP1(I),I=1,3),
        *(UDD(I),I=1,3),(TEMP2(I),I=1,3)
C
C   Readjust translational velocity and acceleration to be about the CG
C   (still in the GCS)
C
C   Velocity = Velocity + (Omega cross R)
C
         CALL CROSSP(TEMP1,RCGG3,TEMP3)
         DO 320 I=1,3
320      UD(I)=UD(I)+TEMP3(I)
C
C   Translational acceleration will be adjusted in two steps:
C   Acceleration = Acceleration + (Omega cross (Omega cross R))
C
         CALL CROSSP(TEMP1,TEMP3,TEMP4)
         DO 322 I=1,3
322      UDD(I)=UDD(I)+TEMP4(I)
```

```
C
C     Acceleration = Acceleration + (Alpha cross R)
C
      CALL CROSSP(TEMP2,RCGG3,TEMP4)
      DO 324 I=1,3
324   UDD(I)=UDD(I)+TEMP4(I)
C
C     Change reference frames of rotation data to the BCS
C
      CALL GMPRD(SEGROT,TEMP1,W,3,3,1)
      CALL GMPRD(SEGROT,TEMP2,WD,3,3,1)
C
C     Determine the Distal forces from the previous link's proximal forces
C
      DO 330 L=1,3
      FDG3(L)=-FPG2(L)
      TDG3(L)=-TFG2(L)
330   CONTINUE
C
C     Process to yeild forces and moments
C
      DO 350 L=1,3
      FPG3(L)=MASS3*UDD(L)-MASS3*G(L)-FDG3(L)
350   CONTINUE
```

```
C
C     Change forces and torque into BCS
C
      CALL GMPRD(SEGROT,FPG3,FP3,3,3,1)
      CALL GMPRD(SEGROT,FDG3,FD3,3,3,1)
      CALL GMPRD(SEGROT,TDG3,TD3,3,3,1)
      CALL CROSSP(RD3,FD3,RDCFD)
      CALL CROSSP(RP3,FP3,RPCFP)
      DO 360 L=1,3
      IW(L)=INERT3(L)*W(L)
360   CONTINUE
      CALL CROSSP(W,IW,WCIW)
      DO 370 L=1,3
      TP3(L)=-(INERT3(L)*WD(L))-WCIW(L)+RDCFD(L)+TD3(L)+RPCFP(L)
370   CONTINUE
C
C     Change reference frames to ground (GCS)
C
      DO 380 I=1,3
      TP3(I)=-TP3(I)
380   CONTINUE
      CALL GMPRD(SEGR1T,TP3,TPG3,3,3,1)
```

```
C
C
C      --------------------------    Segment #4    -------------------------
C
C      Determine the Distal forces from the previous link's proximal forces
C
          DO 430 L=1,3
          FDG4(L)=-FPG3(L)
          TDG4(L)=-TPG3(L)
430       CONTINUE
C
C      Change forces and torque into BCS
C
          CALL GMPRD(SEGROT,FPG4,FD4,3,3,1)
          CALL GMPRD(SEGROT,TDG4,TD4,3,3,1)
C
C      Set proximal forces on last link to zero
C
          DO 450 L=1,3
          FP4(L)=0.0
          TP4(L)=0.0
          FPG4(L)=0.0
          TPG4(L)=0.0
450       CONTINUE
```

```
C
C-----------------------------------------------------------------------
C
C     Write out force data
C
      WRITE(17'KREC)
     &FD1,TD1,FP1,TP1,
     &FDG1,TDG1,FPG1,TPG1,
     &FD2,TD2,FP2,TP2,
     &FDG2,TDG2,FPG2,TPG2,
     &FD3,TD3,FP3,TP3,
     &FDG3,TDG3,FPG3,TPG3,
     &FD4,TD4,FP4,TP4,
     &FDG4,TDG4,FPG4,TPG4
C
C-----------------------------------------------------------------------
C
      PCNT=(FLOAT(K)*100.0)/FRAMES
      WRITE(LUO,1500)PCNT
1500  FORMAT('+',F6.1,'%     ')
1400  CONTINUE
      WRITE(LUO,1505)7,7
1505  FORMAT('+Finished ',2A1)
      CLOSE(UNIT=14)
      CLOSE(UNIT=15)
      CLOSE(UNIT=16)
      CALL CLOSE(17)
      CALL EXST(1)
      END
C
```

GEOMETRY.DAT

+0.0750
+0.0600
+0.0610
-0.0690
+0.1820
+0.0540
-0.0500
+0.2790
+0.1470
+0.0000
+0.0000
+0.0000
+0.0180
-0.2520
+0.0920
-0.0410
-0.1720
+0.0860
+0.0400
+0.0630
+0.0606
-0.0130
-0.0060
+0.0705
-0.0461
+0.0837
+0.1206

INERTIA.DAT

```
00.9500,   Foot Mass [Kg]
00.0060,   Ixx [Kg-m**2]
00.0060,   Iyy [Kg-m**2]
00.0060,   Izz [Kg-m**2]
02.9790,   Shank Mass [Kg]
00.0429,   Ixx [Kg-m**2]
00.0030,   Iyy [Kg-m**2]
00.0429,   Izz [Kg-m**2]
06.5500,   Thigh Mass [Kg]
00.0872,   Ixx [Kg-m**2]
00.0184,   Iyy [Kg-m**2]
00.0872,   Izz [Kg-m**2]
71.4800.   Total Body Mass [Kg]
```

These inertias apply to both T.Macirowski and E.Antonsson

Calculated from Contini, R., "Body Segment Parameters, Part II," Artificial Limbs, Spring 1972, 16:1, 1-19.

Momentary Axis of Rotation

As explained in the text, the momentary axis of rotation solution proceeds by finding the best fit plane (least squares sense) to a portion of the trajectory of one segment's Body Coordinate System (BCS) with respect to another. (PLANE.FTN) Then the best fit sphere (again least squares sense) is found, subject to the constraint that the center of the sphere lie on the best fit plane. The Momentary Axis of Rotation (MAR) lies perpendicular to the plane, and passes through the center of the sphere.

The Plane solution code is interesting, in that there are 4 different possible cases of the plane equation. Each is solved separately, and the solution with the best index of fit is used. Antonsson (1978) [7] discusses the plane solution in greater detail.

```
C  T3BMAH -- TRACK III BATCH MOMENTARY AXIS OF ROTATION
C  T3BMAR
C  T3BMAH
C
C  10-APR-82      by Erik Antonsson
C
      DIMENSION UR(3),NDESCR(20)
      DIMENSION D(7,10),SEGRNT(9,16),SEGH1T(9),SEGROT(9)
      DIMENSION TEMP1(3),TEMP2(3),REL1(3),REL2(3),RELP(3),DA(3)
      BYTE NAMFIL(18)
      COMMON /BLOCK1/U(3,630),BB(4),BR(4),MS(3)
      COMMON /BLOCK2/RC(3),PTH12,UV(3)
      COMMON /BLOCK3/UC(3,2),RL(2)
      COMMON /BLOCK4/IB,IE,NP,NPMX
      COMMON /BLOCK5/BIN(4)
      PI=4.0*ATAN(1.0)
      PIB2=PI/2.0
      PIT2=PI*2.0
      RTD=180.0/PI
      DTR=PI/180.0
      LU=14
      LUO=5
      LUI=5
      LUP=8
      IR=2
      CALL T3SDSH(LU,LUI,LUO,LUP,IR,NAMFIL,NDESCR)
      READ(14'2)RNCHN,FRAMES,FREQ,RNSEG,FPLSCL
      READ(14'7)ERRMAX,XGLO,YGLO,ZGLO
      NCHN=IFIX(RNCHN)
      NFRAME=IFIX(FRAMES)
      NFREQ=IFIX(FREQ)
      NSEG=IFIX(RNSEG)
      NPMX=630
      IF(NFRAME.GT.630)CALL EXST(2)
      IF(NFRAME.LT.630)NPMX=NFRAME
```

```
C
C   Assign each segment another segment for relative motion calculation.
C
        JR=0
        J=1
        IF(NSEG.EQ.1)GOTO 19
10      WRITE(LUO,1103)NSEG,7
1103    FORMAT(' $Enter a segment to find the MAR for.
     &   [1, R:1-',I2,' D:1]: ',A1)
        READ(LUI,1104)J
1104    FORMAT(I6)
        IF(J.EQ.0)J=1
        IF(J.LT.1.OR.J.GT.NSEG)GOTO 10
        WRITE(LUO,1120)J,NSEG,7
1120    FORMAT(' Enter the segment wrt which you want to find the
     &   MAR of',/,' segment #',I2,
     &   ' The "inertial" ref. frame (ground) is segment #0.',/,
     &   '$[1, R:0-',I2,' D:0]: ',A1)
        READ(LUI,1140)JR
1140    FORMAT(I6)
        IF(JR.GT.NSEG)JR=0
        IF(JR.LT.1)JR=0
```

```
C
C   Set trajectory length
C
   15     NS=1
          NF=NFRAME
          CALL QUERY('Do you wish to window the data in Time',IAR)
          IF(IAR.NE.1)GOTO 20
          WRITE(LUO,1148)7
 1148     FORMAT('$Enter the first and last frames to use: ',A1)
          READ(LUI,1149)NS,NF
 1149     FORMAT(2I6)
          IF(NS.LT.1)NS=1
          IF(NF.LE.1)NF=NFRAME
          IF(NF.GT.NFRAME)NF=NFRAME
          IF(NF.GT.NPMX)NF=NPMX
          IF(NS.GT.NF)NS=1
          NFRAME=(NF-NS)+1
   20     WRITE(LUO,1151)
 1151     FORMAT(' Working... ')
C
C   Read Initial Data (Fill up U(3,NPMX))
C
          IF(J.GE.JR)JM=J
          IF(JR.GT.J)JM=JR
          DO 30 K=1,NFRAME
          IF(JR.EQ.0)GOTO 28
```

```
C
C     Relative to another segment
C
      READ(14'K+NS+9)
     &(((D(J2,J3),J2=1,7),(SEGRNT(J2,J3),J2=1,9)),J3=1,JM)
      DO 21 I=1,9
      SEGR1T(I)=SEGRNT(I,JR)
21    CONTINUE
      CALL GMTRA(SEGR1T,SEGRO1,3,3)
      DO 22 I=1,3
      TEMP1(I)=D(I,J)-D(I,JR)
22    CONTINUE
      CALL GMPRD(SEGRO1,TEMP1,REL1,3,3,1)
      DO 26 I=1,3
      U(I,K)=REL1(I)
26    CONTINUE
      GOTO 30
28    READ(14'K+NS+9)
     &(((D(J2,J3),J2=1,7),(SEGRNT(J2,J3),J2=1,9)),J3=1,JM)
      DO 29 I=1,3
      U(I,K)=D(I,J)
29    CONTINUE
30    CONTINUE
C
C     Find axis for initial NPMX points
C
      IB=1
      IE=NFRAME
      NP=NFRAME
      IERR=0
      INIT=1
      CALL AXIS(IERS,INIT,IERP)
      K1=NFRAME/2
      CALL RODVEC(K1,IERR)
```

```
      WRITE(LUO,2490)(BB(IQJ),IQJ=1,4),IERP
2490  FORMAT(' Plane Solution: ',/,
     &1X,G14.7,' * X +',G14.7,' * Y +',G14.7,
     &' * Z +',G14.7,' =0.0',/,' Plane Calculation IERP=',I3)
      WRITE(LUO,2500)(BR(I),I=1,3),(RC(I),I=1,3),(UV(I),I=1,3),
     &BR(4),IERS,IERR
2500  FORMAT(
     &' Origin:        ',3(2X,F14.7),/,
     &' Termination:',3(2X,F14.7),/,
     &' Unit Vector:',3(2X,F14.7),/,
     &' Radius:         ',2X,F14.7,/,
     &' Sphere Calculation IERS=',I3,'   Rodregues Vector    IERR=',I3)
      WRITE(LUO,2510)NAMFIL,NDESCR,J,JR,NS,NF,7,7
2510  FORMAT(' For Datafile: ',18A1,2X,20A2,/,
     &' Segment:',I2,' w.r.t. Segment:',I2,/,
     &' Frame:',I4,' to Frame:',I4,1X,2A1)
      OPEN(UNIT=9,NAME='MARPLT.DAT',TYPE='NEW')
      DO 55 I=1,18
      IF(NAMFIL(I).EQ.0)NAMFIL(I)="40
55    CONTINUE
      WRITE(9,3000)NAMFIL,NDESCR
3000  FORMAT(18A1,2X,20A2)
      WRITE(9,3002)J,JR
3002  FORMAT(2I6)
      WRITE(9,3002)NS,NF
      WRITE(9,3010)(BR(I),I=1,4)
3010  FORMAT(4G14.7)
      WRITE(9,3020)(FB(I),I=1,3)
3020  FORMAT(3G14.7)
      WRITE(9,3002)250,290
      WRITE(9,3022)IERP,IERS,IERR
3022  FORMAT(3I6)
      CLOSE(UNIT=9)
      CALL EXST(1)
      END
```

```
C***************************************************************************
C
C     AXIS -- Subroutine to find an instant axis
C     AXIS
C
C     9-DEC-81        by Erik Antonsson
C
C     IER=-1:  *** ERRORS IN GRADIENT CALCULATION ***
C     IER= 0:  Convergence was obtained.
C     IER= 1:  *** NO CONVERGENCE IN LIMIT ITERATIONS ***
C     IER= 2:  *** LINEAR SEARCH TECHNIQUE INDICATES IT IS LIKELY THAT
C              THERE EXISTS NO MINUMUM ***
C
C
C     BR(1-3) = Origin of the Rodrigues Vector
C     BR(4)   = Radius
C
C
C
      SUBROUTINE AXIS(IER,INIT,IERP)
      EXTERNAL FUNCT4
      DIMENSION H(9),G(2),B(2)
      COMMON /BLOCK1/U(3,630),BB(4),BR(4),MS(3)
      COMMON /BLOCK2/RC(3),PTH12,UV(3)
      COMMON /BLOCK3/UC(3,2),RL(2)
      COMMON /BLOCK4/IB,IE,NP,NPMX
      COMMON /BLOCK5/EIN(4)
C
C     Set up PLANE solution.  It is found by a least squares fit of NP points
C     (a short trajectory).  The plane is used as a const aint in finding
C     the center of rotation.
C
C     BB(1)*X + BB(2)*Y + BB(3)*Z + BB(4) = 0.0
C
      CALL PLANE(IS,IERP)
```

```
C     Find the largest of BB(1), BB(2), BB(3), (to avoid singular plane
C     descriptions) and permute the subscripts of BB and U to get the largest
C     BB last.  By doing this the same FUNCT4 can be used regardless of the
C     Plane description.
C
      BG=ABS(BB(3))
      IBB=0
      DO 4 I=1,2
      IF(ABS(BB(I)).LE.BG)GOTO 4
      BG=BB(I)
      IBB=I
    4 CONTINUE
      DO 8 I1=1,3
      MS(I1)=I1
    8 CONTINUE
      DO 12 I=1,3
      MS(I)=MS(I)+IBB
      IF(MS(I).LE.3) GOTO 12
      MS(I)=MS(I)-3
   12 CONTINUE
```

```
C     If INIT=1:Take the average of X, Y, Z to initialize the optimization.
C     This ensures that the starting point is on the correct side of the
C     curve of the trajectory.
C     If INIT=0: Use BIN as a starting point.
C

      IF(INIT.NE.1)GOTO 18
      B(1)=0.0
      B(2)=0.0
      IE2=IE
      IF(IE.LT.IB)IE2=IE+NPMX
      DO 16 I2=IB,IE2
      I=I2
      IF(I2.GT.NPMX)I=I2-NPMX
      B(1)=B(1)+U(MS(1),I)
      B(2)=B(2)+U(MS(2),I)
16    CONTINUE
      B(1)=B(1)/NP
      B(2)=B(2)/NP
      GOTO 20
18    B(1)=BIN(MS(1))
      B(2)=BIN(MS(2))
20    N=2
      EST=0.0
      EPS=1.E-4
      LIMIT=500
C
C     This fits a sphere to the data by least squares and constrains the
C     center to be on the plane of the points.  The coordinates of the center
C     are BR(1), BR(2), and BR(3).  BR(4) is the radius.
C

      CALL FMFP(FUNCT4,N,B,F,G,EST,EPS,LIMIT,IER,H)
      RETURN
      END
```

```
C***********************************************************************
C
C      FUNCT4
C      FUNCT4
C
C      16-MAR-78          by E.K.Antonsson
C
C      Calculates the sum of the errors of the length from a given center
C      point to each point on the trajectory minus the radius, and the
C      gradients of the function, w.r.t. B(1),B(2).
C      This uses the plane constraint.
C      The radius for each center point is the RMS value for all the
C      points used in the trajectory.  This means that the radius is
C      always correct for each center point that RMCG generates.
C
C      See main programme for explanation of MS(3).
C
       SUBROUTINE FUNCT4(N,B,F,G)
       DIMENSION B(2),G(2)
       COMMON /BLOCK1/U(3,630),BB(4),BR(4),MS(3)
       COMMON /BLOCK2/RC(3),PTH12,UV(3)
       COMMON /BLOCK3/UC(3,2),RL(2)
       COMMON /BLOCK4/IB,IE,NP,NPMX
       COMMON /BLOCK5/BIN(4)
       BR(MS(1))=B(1)
       BR(MS(2))=B(2)
       BR(MS(3))=-(BB(4)+(BR(MS(1))*BR(MS(1)))+(BB(MS(2))*BR(MS(2))))
     */BB(MS(3))
       BR(4)=0.0
       F=0.0
       G(1)=0.0
       G(2)=0.0
```

```
C
C  Radius value determination, BR(4) is the radius
C
       IE2=IE
       IF(IE.LT.IB)IE2=IE+NPMX
       DO 8 L2=IB,IE2
       L=L2
       IF(L2.GT.NPMX)L=L2-NPMX
       RT=0.0
       DO 4 I=1,3
       RS=U(I,L)-BR(I)
       RT=RT+(RS*RS)
    4  CONTINUE
       BR(4)=BR(4)+SQRT(RT)
    8  CONTINUE
       BR(4)=BR(4)/NP
C
C  Objective function and gradients
C
       DO 16 L2=IB,IE2
       L=I2
       IF(L2.GT.NPMX)L=L2-NPMX
       ZD=U(MS(3),L)-BR(MS(3))
       DSCR=0.0
       DO 12 I=1,3
       DSCRT=U(I,L)-BR(I)
       DSCR=DSCR+(DSCRT*DSCRT)
   12  CONTINUE
       SDSCR=SQRT(DSCR)
       F=F+(SDSCR-BR(4))**2
       GC=2*(SDSCR-BR(4))*(1/SDSCR)
       G(1)=G(1)+GC*(BR(MS(1))-U(MS(1),L)-ZD*(BB(MS(1))
    */BB(MS(3))))
       G(2)=G(2)+GC*(BR(MS(2))-U(MS(2),L)-ZD*(BB(MS(2))
    */BB(MS(3))))
   16  CONTINUE
       RETURN
       END
```

```
C*************************************************************************
C  RODVEC -- Rodrigues Vector Solution
C  RODVEC
C
C    9-DEC-81          by Erik Antonsson
C
C  Rodrigues vector solution, computed for 1 time step: (NPL to NPD+1)
C
C  RC1,RC2,RC3 = Termination of the Rodrigues Vector
C  PTH12 = Angle Rotated (NPD to NPD+1)
C
      SUBROUTINE RODVEC(NPD,IER)
      COMMON /BLOCK1/U(3,630),BB(4),ER(4),MS(3)
      COMMON /BLOCK2/RC(3),PTH12,UV(3)
      COMMON /BLOCK3/UC(3,2),RL(2)
      COMMON /BLOCK4/IB,IE,NP,NPMX
      COMMON /BLOCK5/BIN(4)
      PI=4.0*ATAN(1.0)
      CALL ANGLED(NPD,(NPD+1),RL12,TH12,IER)
      IF(IER.EQ.5.OR.IER.EQ.6)RETURN
C
C  PTH12 is a printing angle.
C
      PTH12=TH12*180.0/PI
      RHO=(SIN(TH12/2))/(COS(TH12/2)+1.E-36)
C
C  UV(1), UV(2), and UV(3) are the components of the unit normal perpendicular
C  to the plane of rotation.
C
      UL=SQRT((BB(1)*BB(1))+(BB(2)*BB(2))+(BB(3)*BB(3)))
      UV(1)=BB(1)/UL
      UV(2)=BB(2)/UL
      UV(3)=BB(3)/UL
```

```
C
C     Termination of the Rodrigues vector.
C
      RC(1)=(UV(1)*RHO)+BR(1)
      RC(2)=(UV(2)*RHO)+BR(2)
      RC(3)=(UV(3)*RHO)+BR(3)
C
      RETURN
      END
```

```
C**********************************************************************
C     ANGLED -- Finds the angle (delta angle) between 2 elements of U(3,630)
C     ANGLED
C
C            9-DEC-81          by Erik Antonsson
C
C     FORTRAN CALL:
C            CALL ANGLED(I1,I2,RL12,TH12,IER)
C
C     WHERE:
C            I1 = Beginning element index
C            I2 = Ending element index
C            RL12 = Average radius squared
C            TH12 = Absolute value of Angle between U(1-3,I1) and U(1-3,I2)
C                   [Radians]
C
C            IER = 5:  *** SINGULARITY, NO RODRIGUES VECTOR CALCULATED
C            IER = 6:  *** DIVIDE BY ZERO, OR SQRT OF NEGATIVE NUMBER IN:
C                   TH12 CALCULATION ***
C
      SUBROUTINE ANGLED(I1,I2,RL12,TH12,IER)
      DIMENSION M(2)
      COMMON /BLOCK1/U(3,630),FB(4),BR(4),MS(3)
      COMMON /BLOCK2/RC(3),PTH12,UV(3)
      COMMON /BLOCK3/UC(3,2),RL(2)
      COMMON /BLOCK4/IB,IE,NP,NPMX
      COMMON /BLOCK5/BIN(4)
      PI=4.0*ATAN(1.0)
      PIB2=PI/2.0
      M(1)=I1
      M(2)=I2
      DO 4 N=1,2
      IF(M(N).LE.NPMX)GOTO 2
      M(N)=M(N)-NPMX
      GOTO 1
    1
```

```
 2      UC(1,N)=U(1,M(N))-BR(1)
        UC(2,N)=U(2,M(N))-BR(2)
        UC(3,N)=U(3,M(N))-BR(3)
        RL(N)=SQRT(UC(1,N)**2+UC(2,N)**2+UC(3,N)**2)
 4      CONTINUE
C
C   RL(1) is the length of the vector from PC to P(NPL).
C   RL(2) is the length from PC to P(NPL+1).
C
        RL12=RL(1)*RL(2)
        IF(RL(1).NE.0.0.AND.RL(2).NE.0.0)GOTO 8
        IER=5
        RETURN
 8      THCS=(UC(1,1)*UC(1,2)+UC(2,1)*UC(2,2)+UC(3,1)*UC(3,2))/RL12
        IF(ABS(THCS).LT.1.0)GOTO 12
        IER=6
        RETURN
 12     TH12=(PIE2)-ATAN(THCS/(SQRT(1.0-THCS**2)))
        RETURN
        END
```

```
C     PLANE -- Solution of a Least-Squares Best-Fit Plane to a set of points
C     PLANE
C     PLANE
C
C     FORTRAN CALL:
C           31-MAR-82        by Erik Antonsson
C
C
C     FORTRAN CALL:
C           CALL PLANE(IS,IERP)
C
C     WHERE:
C           IS     = The solution constraint Case number (see below)
C           IERP   = The error/success parameter from FMFP
C                    =-1: Errors in Gradient calculation
C                    = 0: Convergence Obtained
C                    = 1: No Convergence in Limit Iterations
C                    = 2: Linear search technique indicates it is likely
C                         there exists no minimum.
C
C     COMMON CONTAINS:
C           NP     = The number of points
C           U(3,NP) = The X,Y,Z points (Dimension of U(3,NQ) where NQ.GE.NP)
C           BB(4)  = Plane solution coefficients .
C
C     Solution of the best fitting plane (by least squares) to
C     a set of data.  The input is NP (the number of points) and
C     U(3,630) (X,Y,Z points).  The equation is returned in BB(4):
C     BB(1)*X + BB(2)*Y + BB(3)*Z + BB(4) = 0.0
C     There are 4 different cases of solution and each one is
C     performed here.  The 4 constraint cases are;
C     Let  1) BB(1)=1.0, BB(4)=0.0
C          2) BB(2)=1.0, BB(4)=0.0
C          3) BB(3)=1.0, BB(4)=0.0
C          4)            BB(4)=-1.0
```

```
C     These cover all of the possible singularities in the plane
C     solution.  If BB(4)=0.0 the plane passes through the origin.
C     Then each of the other 3 coeffs is set to 1.0 in turn, to
C     eliminate the degree of freedom in the solution.  The 4th
C     case is a plane that doesnt pass through the origin and
C     BB(4) is set to -1.0, eliminating the degree of freedom.
C     These 4 have to be performed separately because the
C     objective function for each is different, and this
C     solution routine will not tolerate a degree of freedom
C     in the solution.  Hence it has to be eliminated systematically.
C     The one with the smallest objective function is chosen
C     to be the equation of the plane.
C
      SUBROUTINE PLANE(IS,IERP)
      DOUBLE PRECISION BP3(2),GP3(2),H3(9),AI3(4)
      DOUBLE PRECISION BP4(3),GP4(3),H4(15),AI4(9)
      DIMENSION P(4,4),F(4),IER(4),IERSMQ(4)
      COMMON /BLOCK1/U(3,630),BB(4),BR(4),MS(3)
      COMMON /BLOCK4/IB,IH,NP,NPMX
      EXTERNAL CASE1,CASE2,CASE3,CASE4
      N=2
      EST=0.0
      EPS=1.0E-10
      LIMIT=300
      IF(IE.GT.IB)IBE=(IE-IB)/2
      IF(IE.LE.IB)IEE=NPMX
```

```
C
C  CASE1 A=1.0,D=0.0
C
      AI3(1)=U(2,IB+5)
      AI3(3)=U(3,IB+5)
      AI3(2)=U(2,IE-5)
      AI3(4)=U(3,IE-5)
      FP3(1)=U(1,IB+5)
      BP3(2)=U(1,IE-5)
      CALL DSIMQ(AI3,BP3,2,IERSMQ(1))
      CALL DFMFP(CASE1,N,BP3,FP3,GF3,FCT,EPS,LIMIT,IER(1),H3)
      P(1,1)=1.0
      P(2,1)=BP3(1)
      P(3,1)=FP3(2)
      P(4,1)=0.0
      F(1)=FP3
C
C  CASE2: B=1.0,D=0.0
C
      AI3(1)=U(1,IB+5)
      AI3(3)=U(3,IB+5)
      AI3(2)=U(1,IE-5)
      AI3(4)=U(3,IE-5)
      FP3(1)=U(2,IB+5)
      BP3(2)=U(2,IF-5)
      CALL DSIMQ(AI3,BP3,2,IERSMQ(2))
      CALL DFMFP(CASE2,N,BP3,FP3,GF3,FCT,EPS,LIMIT,IER(2),H3)
      P(1,2)=B+3(1)
      P(2,2)=1.0
      P(3,2)=BP3(2)
      P(4,2)=0.0
      F(2)=FP3
```

```
C
C     CASE3:  C=1.0,  D=0.0
C
      AI3(1)=U(1,IP+5)
      AI3(3)=U(2,IP+5)
      AI3(2)=U(1,IE-5)
      AI3(4)=U(2,IE-5)
      FP3(1)=U(3,IB+5)
      FP3(2)=U(3,IE-5)
      CALL DSIMQ(AI3,BP3,2,IPRSMQ(3))
      CALL DFMFP(CASE3,N,FP3,FP3,GP3,EST,EPS,LIMIT,IER(3),H3)
      P(1,3)=BP3(1)
      P(2,3)=BP3(2)
      P(3,3)=1.0
      P(4,3)=0.0
      F(3)=FP3
C
C     CASE4:  D=-0.0001
C
      N=3
      EST=0.0
      EPS=1.0E-10
      D=-0.0001
      AI4(1)=U(1,IB+5)
      AI4(4)=U(2,IB+5)
      AI4(7)=U(3,IB+5)
      AI4(2)=U(1,IBE)
      AI4(5)=U(2,IBE)
      AI4(8)=U(3,IBE)
      AI4(3)=U(1,IE-5)
      AI4(6)=U(2,IE-5)
      AI4(9)=U(3,IE-5)
      BP4(1)=-D
      BP4(2)=-D
      BP4(3)=-D
```

```
      CALL DSIMQ(AI4,BP4,3,IERSMQ(4))
      CALL DFMFP(CASE4,N,BP4,FP4,GP4,EST,EPS,LIMIT,IER(4),H4)
      P(1,4)=BP4(1)
      P(2,4)=BP4(2)
      P(3,4)=BP4(3)
      P(4,4)=D
      F(4)=FP4
C
C     Check to see if the IER parameter for the set of coefficients with
C     the smallest objective function is zero.  If not, reset the objective
C     function to greater than the largest, then try again.
C     Check to see if the set of coefficients with the smallest objective
C     function are all zero.  If they all are, reset the objective function
C     to greater than the largest, then try again.
C
7     FG=F(4)
      IG=4
      FL=F(4)
      IS=4
      DO 20 I=1,3
      IF(F(I).GT.FL)GOTO 10
      FL=F(I)
      IS=I
      GOTO 20
10    FG=F(I)
      IG=I
20    CONTINUE
      IF(IER(IS).NE.0)GOTO 22
      IF((P(1,IS).NE.0.0).OR.(P(2,IS).NE.0.0).OR.
     &(P(3,IS).NE.0.0).OR.(P(4,IS).NE.0.0))GOTO 27
22    F(I)=FG*1.1
      GOTO 7
```

```
C
C   Assign best set of coeffs to BB
C
27      DO 30 I=1,4
30      BB(I)=P(I,IS)
        IERP=IER(IS)
        RETURN
        END
```

```fortran
C
C ----------------------------------
C
C      CASF1
C      CASE1
C
       SUBROUTINE CASE1(N,BP,FP,GP)
       DOUBLE PRECISION BP(2),GP(2),FP,A,B,C,D,FPQT,FPQ,DENOM
       DOUBLE PRECISION GT1,GT2,UBV2
       COMMON /BLOCK1/U(3,630),BH(4),BH(4),MS(3)
       COMMON /BLOCK4/IB,IE,NP,NPMX
       A=1.0
       B=BP(1)
       C=BP(2)
       D=0.0
       FPQ=0.0
       GT1=0.0
       GT2=0.0
       DENOM=(A*A)+(B*B +(C*C)
       IE2=IE
       IF(IE.LT.IB)IE2=IE+NPMX
       DO 10 I2=IB,IE2
       L=L2
       IF(L2.GT.NPMX)L=L2-NPMX
       X=U(1,L)
       Y=U(2,L)
       Z=U(3,L)
       FPQT=(A*X)+(B*Y)+(C*Z)+D
       FPQ=FPQ+((FPQT*FPQT)/DENOM)
       GT1=GT1+(((A*Y*X +(B*Y*Y)+(C*Y*Z)+(D*Y))*2.0/DENOM)
       GT2=GT2+(((A*Z*X +(B*Z*Y)+(C*Z*Z)+(D*Y))*2.0/DENOM)
10     CONTINUE
       FP=FPQ
       UBV2=(FP/DENOM)
       GP(1)=GT1-(UBV2*2.0*B)
       GP(2)=GT2-(UBV2*2.0*C)
       RETURN
       END
```

```
C ----------------------------------------
C
C   CASE2
C   CASE2
C
      SUBROUTINE CASE2(N,BP,FP,GP)
      DOUBLE PRECISION BP(2),GP(2),FP,A,B,C,D,FPQT,FPQ,DENOM
      DOUBLE PRECISION GT1,GT2,UBV2
      COMMON /ELOCK1/U(3,630),BR(4),BR(4),MS(3)
      COMMON /ELOCK4/IB,IF,NP,NPMX
      A=BP(1)
      P=1.0
      C=BP(2)
      D=0.0
      FPQ=0.0
      GT1=0.0
      GT2=0.0
      DENOM=(A*A)+(B*B +(C*C)
      IE2=IE
      IF(IE.LT.IB)IE2=IE+NPMX
      DO 10 I2=IB,IE2
      L=L2
      IF(L2.GT.NPMX)L=L2-NPMX
      X=U(1,L)
      Y=U(2,L)
      Z=U(3,L)
      FPQT=(A*X)+(B*Y)+(C*Z)+D
      FPQ=FPQ+((FPQT*FPQT)/DENOM)
      GT1=GT1+(((A*X*X+(B*X*Y)+(C*X*Z)+(D*X))*2.0/DENOM)
      GT2=GT2+(((A*Z*X +(B*Z*Y)+(C*Z*Z)+(D*Y))*2.0/DENOM)
10    CONTINUE
      FP=FPQ
      UBV2=(FP/DENOM)
      GP(1)=GT1-(UBV2*2.0*A)
      GP(2)=GT2-(UBV2*2.0*C)
      RETURN
      END
```

```
C  -------------------------------------------------------
C
C  CASE3
C  CASE3
C
       SUBROUTINE CASE3(N,BP,FP,GP)
       DOUBLE PRECISION BP(2),GP(2),FP,A,E,C,D,FPQT,FPQ,DENOM
       DOUBLE PRECISION GT1,GT2,UBV2
       COMMON /BLOCK1/U(3,630),BR(4),BR(4),MS(3)
       COMMON /BLOCK4/IB,IE,NP,NPMX
       A=BP(1)
       B=BP(2)
       C=1.0
       D=0.0
       FPQ=0.0
       GT1=0.0
       GT2=0.0
       DENOM=(A*A(+(B*B +(C*C)
       IE2=IE
       IF(IE.LT.IB)IE2=IE+NPMX
       DO 10 L2=IB,IE2
       L=L2
       IF(L2.GT.NPMX)L=L2-NPMX
       X=U(1,L)
       Y=U(2,L)
       Z=U(3,L)
       FPQT=(A*X)+(B*Y)+(C*Z)+D
       FPQ=FPQ+((FPQT*FPQT)/DENOM)
       GT1=GT1+(((A*X*X +(B*X*Y)+(C*X*Z)+(D*X))*2.0/DENOM)
       GT2=GT2+(((A*Y*X +(B*Y*Y)+(C*Y*Z)+(E*Y))*2.0/DENOM)
10     CONTINUE
       FP=FPQ
       UBV2=(FP/DENOM)
       GP(1)=GT1-(UBV2*2.0*A)
       GP(2)=GT2-(UBV2*2.0*E)
       RETURN
       END
```

```
C   -----------------------------------------------------------
C
C   CASE4
C   CASE4
C
      SUBROUTINE CASE4(N,BP,FP,GP)
      DOUBLE PRECISION EP(3),GP(3),FP,A,B,C,D,FPQT,FPQ,DENOM
      DOUBLE PRECISION GT1,GT2,GT3,UBV2
      COMMON /BLOCK1/U(3,630),BB(4),BR(4),MS(3)
      COMMON /BLOCK4/IB,IE,NP,NPMX
      A=BP(1)
      B=BP(2)
      C=BP(3)
      D=-0.0001
      FPQ=0.0
      GT1=0.0
      GT2=0.0
      GT3=0.0
      DENOM=(A*A)+(B*B +(C*C)
      IE2=IE
      IF(IE.LT.IB)IE2=IE+NPMX
      DO 10 L2=IB,IE2
      L=L2
      IF(L2.GT.NPMX)L=L2-NPMX
      X=U(1,L)
      Y=U(2,L)
      Z=U(3,L)
      FPQT=(A*X)+(B*Y)+(C*Z)+D
      FPQ=FPQ+((FPQT*FPQT)/DENOM)
      GT1=GT1+(((A*X*X +(B*X*Y)+(C*X*Z)+(D*X))*2.0/DENOM)
      GT2=GT2+(((A*Y*X +(B*Y*Y)+(C*Y*Z)+(D*Y))*2.0/DENOM)
      GT3=GT3+(((A*Z*X +(B*Z*Y)+(C*Z*Z)+(D*Z))*2.0/DENOM)
10    CONTINUE
      FP=FPQ
      UBV2=(FP/DENOM)
      GP(1)=GT1-(UBV2*2.0*A)
      GP(2)=GT2-(UBV2*2.0*B)
      GP(3)=GT3-(UBV2*2.0*C)
      RETURN
      END
```

APPENDIX I

THE PLOTTING SOFTWARE PACKAGE

The host computer for the TRACK III system has always had minimal graphics capability. An older computer in the same lab was equiped with a VT-11 stroke refresh display processor and a VR17 17 inch diagonal screen. The VT-11 has disadvantages in that the display list is held in the central processor's main memory, so that the dislpay processor must steal memory cycles to refresh the display. For this reason, it was felt that a distributed network co-processing application of display would gain the highest performance. The Host processor issues plot commands and arguments to a channel of a local network. The satelite is "listening" to that channel and responds to the commands by passing the arguments to the proper subroutine. To begin the explanation, lets start with the satelite.

The VT-11 is simple to control. There are only a few different instructions that can be put in the display list to move the beam either with the beam on or off.

.

SET GRAPHIC MODE
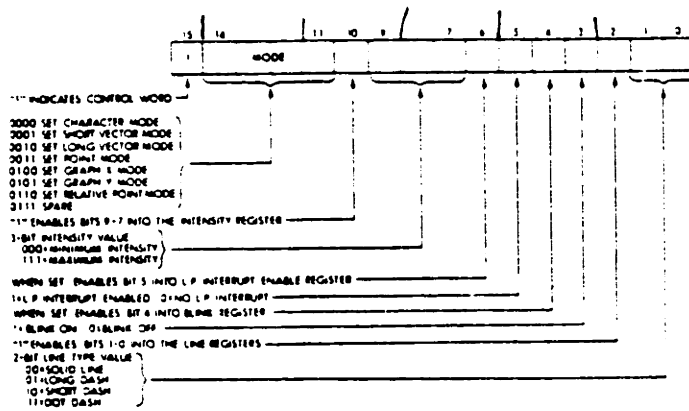
JUMP
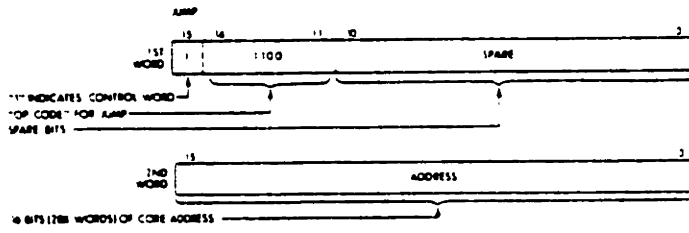
NO-OP
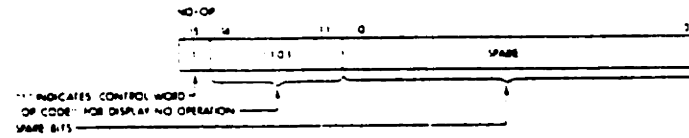
LOAD STATUS REGISTER A
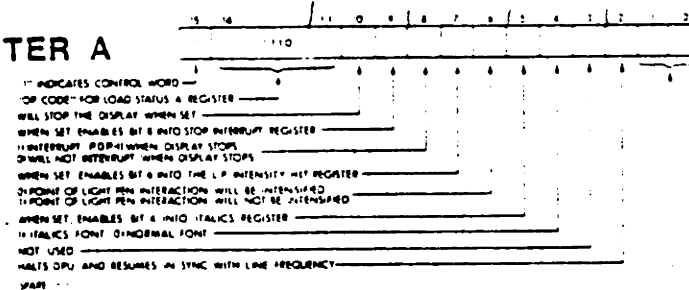
LOAD STATUS REGISTER B
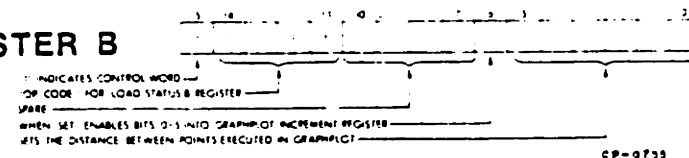
Figure 2-8 Instruction Word Functions

VT-11 Display Processor Instructions

CHARACTER DATA FORMAT- Mode 0000
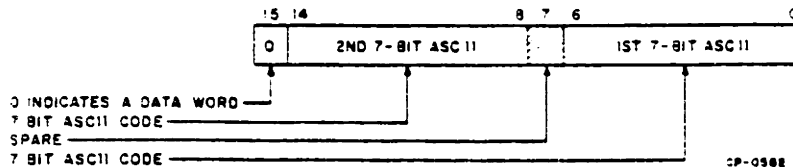
SHORT VECTOR MODE- Mode 0001
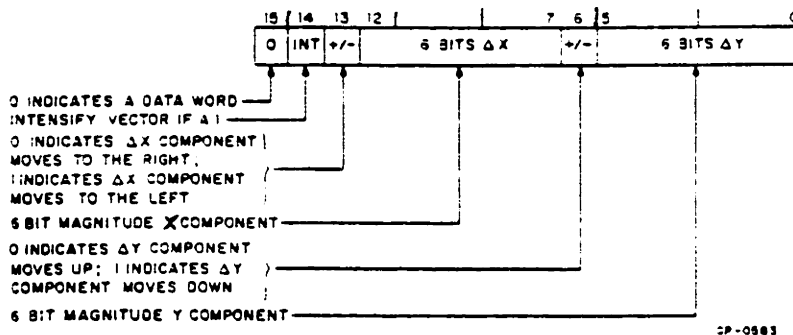
LONG VECTOR DATA FORMAT- 0010

Figure 2-18 Data Word Formats (Sheet 1 of 2)

VT-11 Display Processor Graphic Data
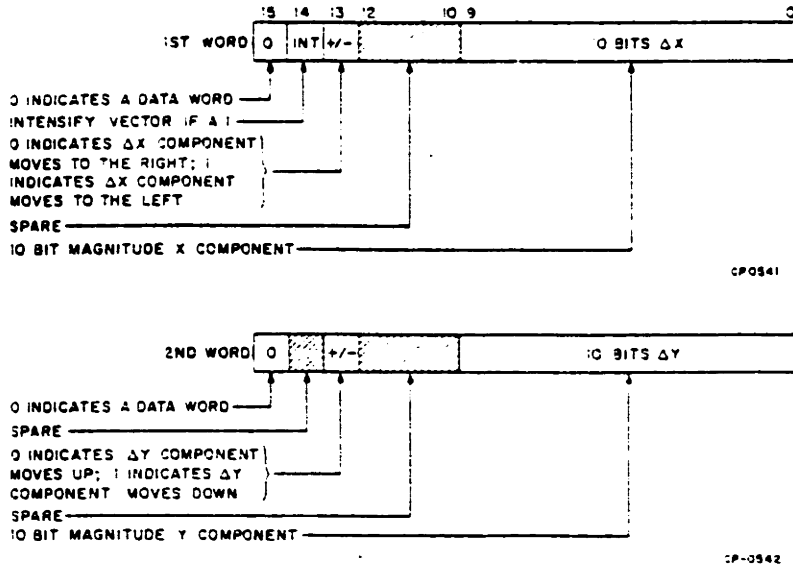
**POINT DATA
MODE-
Mode 0011**

1ST WORD

```
 15  14  13        10 9                    0
| O | INT |      | ░░░ |      10 BITS X      |
```

O INDICATES A DATA WORD ──
INTENSIFY POINT IF A 1 ──
SPARE ──
10 BIT X COORDINATE ──

CP 0543

2ND WORD

```
| O |░░░░░░░░░░░|        10 BITS Y         |
```

O INDICATES A DATA WORD ──
SPARE ──
10 BIT Y COORDINATE ──
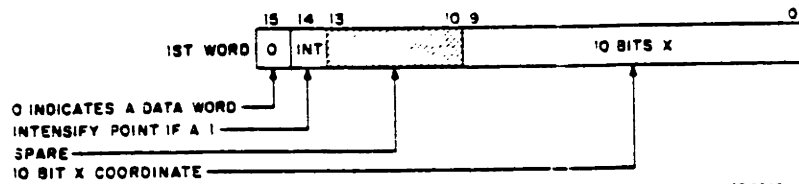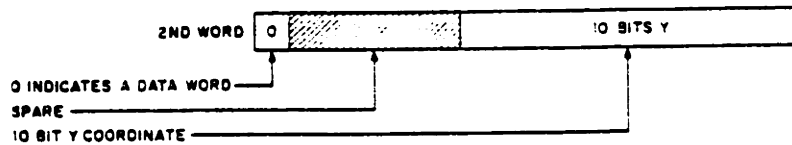
CP-0544

**GRAPHPLOT X-
Mode 0100**

**GRAPHPLOT Y-
Mode 0101**

```
 15  14         10  9                    0
| O |░░░░░░░░░|      10 BITS X (Y)        |
```

O INDICATES A DATA WORD ──
SPARE ──
10 BIT X(Y) COORDINATE ──
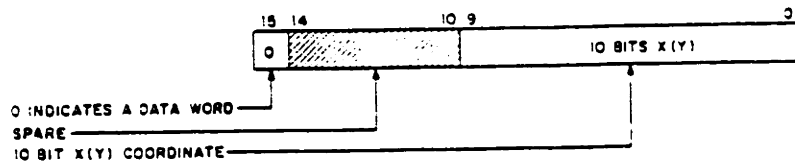
CP-0545

**RELATIVE
POINT MODE-
Mode 0110**

```
 15  14  13  12          7  6  5          0
| O |INT|+/-|  6 BITS ΔX  |+/-| 6 BITS ΔY |
```

O INDICATES A DATA WORD ──
INTENSIFY POINT IF A 1 ──
O INDICATES X COMPONENT
MOVES TO THE RIGHT; 1
INDICATES X COMPONENT
MOVES TO THE LEFT
6 BIT MAGNITUDE X COMPONENT ──
O INDICATES Y COMPONENT MOVES UP;
1 INDICATES Y COMPONENT MOVES DOWN
6 BIT MAGNITUDE Y COMPONENT ──
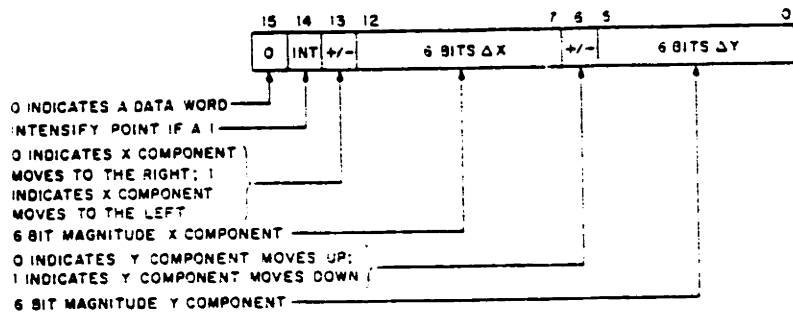
CP-0546

Figure 2-18   Data Word Formats (Sheet 2 of 2)

VT-11 Display Processor Graphic Data

The subroutine library VTLIB allows a Fortran program to call various subroutines and add elements to the display list. The only complication is that short vectors (shorter than 63 plotting units in either direction) can be single word instructions, and long vectors (up to the full 768 by 1024) are written as 2 word graphic instructions. VTLIB as it stands always determines if an instruction can be a short vector, and uses that mode if possible to save display list space, and to speed the refresh rate. The mixed length instructions becomes a complication if dynamics are to be used, and a short vector may grow too long for a single word. In this case, all instructions that may be modified should be explicitly written as long vectors. Since all of the graphics for the current level of function for TRACK III display are static (cartesion plots of one result variable against another) the option for dynamics was not included in the code.

One other oddity of the VT-11 display processor was that it had to be halted (the display processor, not the central processor) when ever the display list was modified. If a 2 word instruction was in the middle of being modified while the display pointer passed over that instruction, frequently the satelite computer would crash. To avoid crashes, a synchronous QIO was always issued to stop the DPU and an asynchronous QIO issued to start it up after the modifications. This start and stop proceedure even applies to simple extensions of the display list.

Note that all arguements to VTLIB are integer, machine plotting units. There is no user scaling. This was done to minimize the amount of calculation done on the satelite machine owing to its substantially slower speed.

The original VTLIB was purchased from Andrew Rubel and Associates Incorporated, and modified to operate with a VT-11 by the author.

To use the satelite with the local network, a plot instruction receiver task was written to "listen" to the appropriate channel and send the argument lists it received to the appropriate VTLIB subroutine. To maximize the efficiency of plotting and network data transfer, two 256-word long data buffers were "double buffered." This meant that while one buffer was being filled with plot commands and argument lists from the local network, the other buffer was being interrogated to produce plot

commands.   Network   plot   commands   are   distinguished from arguments by the commands being less   than   negative   30,000 (decimal).   No   arguments   for any of the VTLIB subroutines could be smaller than 30,000.   The   task   VTPLOT   receives buffers   of data, finds network plot commands and then sends the   subsequent   argument   list   to   the   proper   plotting subroutine, all with 2 buffers in constant use.

        The display list area is a fixed data partition   called DFILE.

        The Host plot   package   formulates   plot   commands   and argument   lists   for   the   soft-copy network plots as well as building plot commands to produce   identical   plots   on   the Hewlett-Packard 9872A   4 color pen plotter.   The basic plot library for both these devices is VHLIB.FTN   and   VHLIB.MAC. A   very   useful autoscaling plot routine used throughout the TRACK III software package is MGRAPH.FTN (and   an   identical subroutine but with a "Virtual" plotting array: VGRAPH).

Table 4-1. Interface Language Instructions

Instruction	Definition	Paragraph Reference
VECTOR GROUP		
PA x,y(,x,y(,...))	Plot absolute [i]	4-29
PR x,y (,x,y(....))	Plot relative [i]	4-31
CHARACTER GROUP		
CA n	Designate alternate set n [i]	4-10
CP spaces, lines	Character plot [d]	4-11
CS m	Designate standard set m [i]	4-12
DI run. rise	Absolute direction [d]	4-15
DR run, rise	Relative direction [d]	4-17
LB c....c	Label ASCII string [c]	4-22
SA	Select Alternate	4-33
SI wide. high	Absolute character size [d]	4-34
SL θ	Absolute character slant (from vertical) [d]	4-35
SR wide. high	Relative character size [d]	4-38
SS	Select standard	4-39
UC x,y,pen(,...)	User defined character [i]	4-41
LINE TYPE GROUP		
LT t(,l)	Designate line type t and length l [d]	4-23
PD	Pen down	4-30
PU	Pen up	4-32
SM c	Symbol mode [c]	4-36
SP n	Select pen [i]	4-37
VA	Adaptive velocity	4-47
VN	Normal velocity	4-48
VS v(,n)	Select velocity v for pen n [i]	4-49
DIGITIZE GROUP		
DC	Digitize clear	4-13
DP	Digitize point	4-16
OC	Output current position & pen status [i]	4-24
OD	Output digitized point & pen status [i]	4-25
AXES		
TL tp(,tn)	Tick length [d]	4-40
XT	X axis tick	4-50
YT	Y axis tick	4-51
SET-UP GROUP		
IP p1x,p1y,p2x,p2y	Input p1 and p2 [i]	4-20
IW xlo,ylo,xhi,yhi	Input window [i]	4-21
OP	Output p1 and p2 [i]	4-27
CONFIGURATION STATUS		
AP (θ)	Automatic pen pickup [i]	4-9
DF	Set default values	4-14
IM e(,s(p))	Input e,s, and p masks [i]	4-18
IN	Initialize	4-19
OE	Output error [i]	4-26
OS	Output status [i]	4-28

Formats: [i] = integer format between – 32767 to +32767. No decimal. [c] = ASCII character.
        [d] = decimal format between ± 127.999. Decimal is optional.

Hewlett-Packard 9872A Pen Plotter Graphics Instructions

```
C     VHLIB
C     VHLIB
C
C     VHLIB
C
C     -----------------------------------------------------------------
C
C         Copyright (C) 1980  Massachusetts Institute of Technology
C                        Cambridge, Massachusetts
C
C     This software is furnished under an agreement for use only on a
C     single computer system and may be copied only with the inclusion
C     of the above copyright notice.  This software, or any other copies
C     thereof, may not be provided or otherwise made available to any
C     other person except for use on such system and to one who subscribes
C     to these agreement terms.  Title to and ownership of the software
C     shall at all times remain in MIT.
C
C     The information in this document is subject to change without notice
C     and should not be construed as a commitment by MIT.
C
C     MIT assumes no responsibility for the use or reliability of its
C     software.
C
C     -----------------------------------------------------------------
C
C     This is the VT-11 plotting package emulator for the Hewlett Packard
C     9872 A plotter on the PDP 11/60 interfaced through an MDB IB-11
C     interface, and to a VT-11 through DECnet running a highly modified
C     version of Andy Rubel's VT-11 software and task to task communication.
C     Every effort has been made to make the subroutine
C     calls exactly like the VT-11 calls for direct plotting software
C     compatability.  However this was not always possible.  The most
C     notable departure is that there are NO DEFAULTS.  Every argument
C     for all of these subroutines MUST be included.
C
```

```
C     To access these subroutines the library must be included in the
C     Taskbuilder list:
C
C         LB:[2Ø1,1]VHLIB.OLB/LB
C
C     And the interface common area must be included under OPTIONS:
C
C         COMMON=IOPAGE:RW
C
C     For more specific information see the RSX-11M Task Builder Manual,
C     RSX-11M V3.2 Volume 3A.
C     For further background on the plotting subroutines see the FORTRAN
C     RT-11 Extensions Manual, Chapter 1 - VT11 Support.
C     N.B. There are some routines in the RT-11 VT11 Support not
C     emulated here.  They provide function not available on the H/P.
C     Several (EXTRA) routines have been added here to access extra
C     capability that the H/P does have.
C
C     N.B.   INIT and ENDP use system wide event flag #56 to determine
C     if the plotter is currently in use.  INIT checks the flag, and
C     once clear it sets is and begins the plot.  ENDP clears the flag
C     when plotting is done.  All plotting programmes MUST be aborted
C     by typing:
C         >HPC
C
C     (Hewlett Packard Clear) from the monitor.  This aborts the currently
C     plotting programme and clears the lockout flag and cleans up the plotter.
C     N.E. HPC will only abort an unprivileged users plot job if it is
C     issued from the terminal the job was issued from.
C     If a plotting programme is ABOrted, it
C     will leave the event flag set, and the plotter locked out.
C     In this event type
C         >HPC
C
C     from the monitor.  This programme will
C     clear event flag #56 and unlock the plotter.  This may produce an
C     FCS error (usually #-26, No such file).  Ignore it.
C
```

```fortran
C --------------------------------------------------------
C
C  APNT -- VT-11 EMULATION SUBROUTINE
C  APNT
C
C
C  13-NOV-80        by Erik Antonsson
C
C  FORTRAN CALL:
C     CALL APNT(X,Y,L,I,F,T)
C
C  WHERE:
C
C     X,Y = The absolute pen position to move to in user
C           coordinates.
C
C     L = A dummy argument for compatability
C     I = Negative for PEN UP
C       = Zero to leave the pen in the current up or down
C         position.  However if the last call to APNT
C         (or VECT) specified I=negative, then the next
C         call with I=0 will put the pen down.
C       = Positive for PEN DOWN
C     F = A dummy argument for compatability
C     T = Line type.  This is the same as in the H/P documentation
C       except: Zero = leave the line type alone
C              Negative = solid line
C              7 = H/P line type 0 (dots at the vector endpoints)
C
C
      SUBROUTINE APNT(X,Y,L,I,F,T)
      COMMON/SCALE/FX,FY,XO,YO
      COMMON/IPENST/IP
      COMMON/LINEST/IT
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *ILEV,VTBHP,VTOFSX,VTOFSY
      INTEGER F,T
      BYTE ID(15),IX(6),IY(6),ILTYP(7),ILT(3)
C
```

```
      DATA ID/'P','A','0','0','0','0','0','0','0',
     *'0','0','0','0','0','0','0','0','0','0',
      DATA ILTYP/'1','2','3','4','5','6','0'/
      DATA ILT/'L','T','0'/
      DATA IF/0/
      DATA IT/0/
      IF(IDEV.EQ.1)GOTO 9

C
C     VT-11 Routine
C
      IF(INDX.GT.INDXMX-5)CALL DFBDMP
      IDAT(INDX,IBUF)=-30004
      IF(T.EQ.0)GOTO 1
      IF(T.EQ.7)IDAT(INDX,IBUF)=-30006
      IT=T
      GOTO 2
1     IF(IT.EQ.7)IDAT(INDX,IBUF)=-30006
2     IDAT(INDX+1,IBUF)=IFIX(((X-XO)*FX*VTBHP)+VTOFSX)
      IDAT(INDX+2,IBUF)=IFIX(((Y-YO)*FY*VTBHP)+VTOFSY)
      IF(I)3,5,5
3     IDAT(INDX+3,IBUF)=0
      IP=-1
      GOTO 8
5     IDAT(INDX+3,IBUF)=1
      IP=1
8     INDX=INDX+4
      RETURN
```

```
C     H/P Routine
C     Determine the machine coordinates from the user coordinates
C
      5     MX=IFIX((X-XO)*FX)
            MY=IFIX((Y-YO)*FY)
            ENCODE(6,1000,IX)MX
      1000  FORMAT(I6)
            ENCODE(6,1000,IY)MY
            DO 10 J=1,6
            JX=J+2
            JY=J+9
            ID(JX)=IX(J)
            ID(JY)=IY(J)
      10    CONTINUE
C
C     Change PEN UP/DOWN status as needed
C
      12    IF(I)12,13,14
            CALL HPDCMD('PU',2)
            IP=-1
            GOTO 16
      13    IF(IP.GE.0)GOTO 16
      14    CALL HPDCMD('PD',2)
            IP=1
      16    IF(T)18,30,20
C
C     Change Line Type as needed
C
      18    CALL HPDCMD('LT',2)
            GOTO 30
      20    ILT(3)=ILTYP(T)
            CALL HPDCMD(ILT,3)
C
C     Plot Absolute pen move
C
      30    CALL HPDCMD(ID,15)
            RETURN
            END
```

```
C -----------------------------------------------------------------
C
C   CHAR -- VT-11 EMULATION (EXTRA)
C   CHAR
C
C         14-NOV-80        by Erik Antonsson
C
C   Changes the character size,slant and slope
C
C       FORTRAN CALL:
C         CALL CHAR(DW,DH,DT,DRN,DRS)
C       WHERE:  (all arguments must be real (floating point)
C       DW=Character width in cm
C       DH=Character height in cm
C       DT=Character slope (tan of slope angle, vertical=0)
C       DRN,DRS=run and rise respectively of slope the characters
C               should be written along.
C
C
      SUBROUTINE CHAR(DW,DH,DT,DRN,DRS)
      COMMON/CHAR/PDW,PDH
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTBHP,VTOFSX,VTOFSY
      BYTE LI(31),SI(31),SL(16),ICMD(14)
      DATA DI/'D','I',14*''0,',',14*''0/
      DATA SI/'S','I',14*''0,',',14*''0/
      DATA SL/'S','L',14*''0,',',14*''0/
```

```fortran
C
C     Change size (and aspect ratio)
C
      IF(IDEV.NE.1)RETURN             !Return if not for H/P
      IF(DW.EQ.0.0.AND.DH.EQ.0.0)GOTO 100
      IF(DW.EQ.0.0)DW=PDW
      ENCODE(14,1000,ICMD)DW
1000  FORMAT(F14.7)
      DO 10 J=1,14
      SI(J+2)=ICMD(J)
10    CONTINUE
      IF(DH.EQ.0.0)DH=PDH
      ENCODE(14,1000,ICMD)DH
      DO 40 J=1,14
      SI(J+17)=ICMD(J)
40    CONTINUE
      CALL HPDCMD(SI,31)
      PDW=DW
      PDH=DH
C
C     Change slant
C
100   IF(DT.EQ.0.0)GOTO 200
      ENCODE(14,1000,ICMD)DT
      DO 110 J=1,14
      SL(J+2)=ICMD(J)
110   CONTINUE
      CALL HPDCMD(SL,16)
```

```
C
C   Change slope
C
200      IF(DRN.EQ.0.0.AND.DRS.EQ.0.0)GOTO 300
         ENCODE(14,1000,ICMD)DRN
         DO 210 J=1,14
         DI(J+2)=ICMD(J)
210      CONTINUE
         ENCODE(14,1000,ICMD)DRS
         DO 240 J=1,14
         DI(J+17)=ICMD(J)
240      CONTINUE
         CALL HPDCMD(DI,31)
300      RETURN
         END
```

```fortran
C
C
C     CHDF -- VT-11 EMULATION (EXTRA)
C     CHDF
C
C         14-NOV-80        by Erik Antonsson
C
C     Returns the default characters in size,slant, and slope
C     This also sets the default for this plotting libe to be
C     a character size of .19 cm wide by .31 cm high.
C
C     FORTRAN CALL:
C           CALL CHDF
C
C
      SUBROUTINE CHDF
      COMMON/CHAR/PDW,PDH
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR.
*IDEV,VTBHP,VTOFSX,VTOFSY     !Return if not for H/P
      IF(IDEV.NE.1)RETURN
      CALL HPDCMD('DI',2)
      CALL HPDCMD('SI0.19,0.31',11)
      CALL HPDCMD('SL',2)
      PDW=0.19
      PDH=0.31
      RETURN
      END
C
```

```
C-------------------------------------------------------------------
C
C  ENDP -- VT-11 EMULATION (EXTRA)
C  ENDP
C
C       14-NOV-80       by Erik Antonsson
C
C  Terminates plotting, and moves pen out of the way, and clears the
C  other user lockout flag: system wide event flag #56.
C
C
C  FORTRAN CALL:
C       CALL ENDP
C
C
        SUBROUTINE ENDP(IPRMPT)
        COMMON/SCALE/FX,FY,XO,YO
        COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
       *IDEV,VTBHP,VTOFSX,VTOFSY
        BYTE BELL
        DATA BELL/"7/
        CALL NARG(N)
        IQR=0
        IF(N.LE.0)GOTO 10
        IQR=IPRMPT
10      IF(IDEV.EQ.1)GOTO 50
        CALL LFBDMP
        CALL WAITNT(,IOST(1,IBUFR))
        IF(IQR.NE.0)GOTO 20
        TYPE 900,BELL
900     FORMAT(' ',$Type <CR> to erase the VT-11.',A1)
        ACCEPT 905,IBCR
905     FORMAT(A2)
```

```
20    IDAT(1,IBUFR)=-30000
      CALL SNDNTW(2,IOST(1,IBUFR),IBYTE,IDAT(1,IBUFR))
      IF(IOST(1,IBUFR).LT.0)CALL NETERR('SNDNTW','ENDP   ',IBUFR)
      CALL DSCNTW(2,IOST(1,1))
      IF(IOST(1,1).LT.0)CALL NETERR('DSCNTW','ENDP   ',1)
      CALL CLSNTW
      RETURN
50    CALL HPDCMD('DF',2)
      CALL HPDCMD('IW',2)
      CALL HPDCMD('PU',2)
      CALL SPEN(0)
      CALL HPDCMD('PA15500,11000',13)
      CALL CLREF(56)
      TYPE 1000
1000  FORMAT(' Plot complete')
      RETURN
      END
C
```

```
C
C      INIT -- VT-11 EMULATION SUBROUTINE
C      INIT
C
C      13-NOV-80       by Erik Antonsson
C
C      Initializes the HP-9872A plotter through the IB-11 interface
C      This subroutine MUST be called before ANY plotting is performed.
C      Either using the plot package subroutines, QGRAPH, or RST .  However
C      it should be called ONLY once per plotted page.  N.B. System wide
C      event flag #56 must be clear before this routine will continue.
C      If set, it indicates that another plotting programme has control
C      of the plotter.  Once this programme gains control, it too sets
C      event flag #56.  ENDP clears the flag.  All plotting programmes
C      MUST be aborted by typing:
C          >HPC
C
C      from the monitor to properly clear the lockout flag.
C      If a plotting programme
C      is APORTED, the flag will not be clear until the user types:
C          >HPC
C
C      This is a Hewlett Packard Clear programme which clears the event flag
C
C
C      FORTRAN CALL:
C          CALL INIT(IDEV)
C
C      WHERE:
C
C          IDEV=0 for the VT-11 across the NET
C          IDEV=1 for the H/P 9872A
C
C
```

```
C
      SUBROUTINE INIT(IDEVA)
      EXTERNAL ENDP
      COMMON/SCALE/FX,FY,XO,YO
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTEHP,VTOFSX,VTOFSY
      INTEGER IBS(3),BUF(16),MSTAT(3)
      BYTE CONBLK(72)
      BYTE IDUM,BELL
      BYTE IDS,IDSH
      DATA BELL/."7/
      DATA IBS/."16,."41,."44/
      DATA IBD/."45/
      IDEV=IDEVA
      IF(IDEV.EQ.1)GOTO 8
      CALL OFNNTW(3,IOST(1,1),MSTAT,1,1000)
      IF(IOST(1,1).LT.0)CALL NETERR('OPNNTW','INIT   ',1)
      IF(IOST(1,1).LT.0)RETURN
      CALL BFMT1(ISTAT,CONBLK,5,'FORTY ',.6,'VTPLOT')
      CALL CONNTW(2,IOST(1,2),CONBLK)
      IF(IOST(1,2).LT.0)CALL NETERR('CONNTW','INIT   ',2)
      IBYTE=512
      INDXMX=255
      IBUF=1
      IBUFR=2
      IDAT(1,IBUF)=-30001
      INDX=2
      CALL NOSC
      CALL USEREX(ENDP)
      RETURN
C
```

```
C
C       Determine if the plotter is currently in use.
C
      H       CALL SETEF(56,IDS)
              IF(IDS.EQ.0)GOTO 10
              TYPE 997,BELL,BELL
997     FORMAT(/,' The H/P 9872A pen plotter is currently in use.',/,
      *,' If no other user is currently plotting type: G ',/,
      *,'$Do you wish to wait for it to become free? [Y/N/G]: ',A1,A1)
              ACCEPT 998,IDUM
998     FORMAT(A1)
              IF(IDUM.EQ.'G'.OR.IDUM.EQ.'g')GOTO 10
              IF(IDUM.NE.'Y'.AND.IDUM.NE.'y')STOP
      9       CALL SETEF(56,IDS)
              IF(ILS.EQ.2)GOTO 9
C
C       Wait until paper is loaded
C
      10      TYPE 1000,BELL
1000    FORMAT('$Type a carriage return when the paper is properly',
      *,' loaded in the plotter.',A1)
              ACCEPT 998,IDUM
C
C       Initialize the IB-11 interface and the plotter
C
              IEEE=IBUP(9,1,"0105,"045,0,6,"012,0)
              IF(IEEE.LT.0)CALL IEEER(IEEE)
C
C       Clear the interface
C
              IEEE=IBUP(2,1)
              IF(IEEE.LT.0)CALL IEEER(IEEE)
C
```

```fortran
C     Set timeout to 100 seconds
C
      IEEE=GPIB(12,100)
      IF(IEEE.LT.0)CALL IEEER(IEEE)
14    CALL HPDCMD('IW',2)
      CALL HPDCMD('PU',2)
      CALL HPDCMD('PA15000,0',9)
      CALL HPDCMD('PA15000,0',9)
      DO 15 IQ=1,10000           !wait loop for PA to finish before init
15    CONTINUE
      CALL HPDCMD('IN',2)
C
C     Set default scaling to match VT-11
C
      FX=10.
      FY=10.
      XO=0.
      YO=0.
C
C     Set all plotter values to default
C
      CALL HPDCMD('PA10500,0',9)
      CALL HPDCMD('DF',2)
      CALL CHDF
C
C     Set the window to 8.5x11.0 with 1/4 inch margins
C
      CALL HPDCMD('IW0,0,10500,8000',16)
C
C     Set P1 and P2 to the diagonal corners of this window
C
      CALL HPDCMD('IP0,0,10500,8000',16)
```

```
C
C    Set all pen velocities to 10 cm/sec
C
     CALL HPDCMD('VS10',4)
     CALL USEREX(ENDP)
     RETURN
     END
C
```

```
C---------------------------------------------------------------------
C
C    NMBR -- VT-11 EMULATION
C    NMBR
C
C        14-NOV-84      by Erik Antonsson
C
C    Prints out a number at the current pen location in the
C    current character format
C
C    FORTRAN CALL:
C        CALL NMBR(N,VAR,'FORMAT')
C
C    WHERE:
C        N=dummy argument for compatability
C        VAR= the integer or real number to be printed
C        'FORMAT'= the format desired for printing
C        i.e.: 'F6.2'.  The format must be enclosed in single quotes
C
C    EXAMPLE:
C        CALL NMBR(0,3.5,'F8.3')
C
C
      SUBROUTINE NMBR(N,VAR,IFMT)
      COMMON/FORMAT/IFMTP
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTBHP,VTOFSX,VTOFSY
      BYTE IFMT(1),ISP,IF(2),IFMTP(16),IPAR(2),INMBR(16)
      EQUIVALENCE (INT,REAL)
      DATA IPAR/'(',',',')'/
      DATA ISP/' '/
      DATA IF/'I','.','0'/
      REAL=VAR
      IFFL=0
      IF(IFMT(1).EQ.0)GOTO 100
```

```
      DO 10 J=1,18
      IFMTP(J)=ISP
10    CONTINUE
      DO 15 J=1,16
      INMBR(J)=0
15    CONTINUE
C
C     Determine if the number is to be a REAL or INTEGER
C     (by checking for 0 or I format specifiers)
C
      DO 20 J=1,16
      IF(IFMI(J).EQ.0)GOTO 25
      IF(IFMI(J).EQ.IF(1).OR.IFMT(J).EQ.IF(2))IFFL=1
20    CONTINUE
C
C     Add syntatical parentheses
C
25    IFMTP(1)=IPAR(1)
      IFMTP(J+1)=IPAR(2)
C
C     Place input string into FORMAT array
C
      DO 30 I=1,J-1
      IFMTP(I+1)=IFMT(I)
30    CONTINUE
100   IF(IFFL.EQ.0)ENCODE(16,IFMTP,INMBR)REAL
      IF(IFFL.EQ.1)ENCODE(6,IFMTP,INMBR)INT
200   CALL TEXT(INMBR,0)
      RETURN
      END
C
```

```
C      -------------------------------------------------------------
C
C      NOSC -- VT-11 EMULATION
C      NOSC
C
C      14-NOV-82      by Erik Antonsson
C
C      Resets the user scaled coordinates to VT-11 coordinates
C
       SUBROUTINE NOSC
       COMMON/SCALE/FX,FY,XO,YO
       COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
      *IDEV,VTBHP,VTOFSX,VTOFSY
       FX=16.
       FY=10.
       XO=0.
       YO=0.
       IF(IDEV.NE.0)RETURN
       CALL SCAL(0.,0.,1050.,800.,FX,FY)
       VTBHP=0.1
       VTOFSX=0.0
       VTOFSY=0.0
       RETURN
       END
C
```

```fortran
C
C
C      SCAL -- VT-11 EMULATION
C      SCAL
C
C      13-NOV-80      by Erik Antonsson
C
C      Sets user coordinate scaling and origin for subsequent calls
C      to VECT or APNT.
C
C      FORTRAN CALL:
C           CALL SCAL(XO,YO,XL,YL,FX,FY)
C
C      WHERE:
C           XO,YO= user coordinates of the machine origin
C           XL,YL= user coordinates of the upper right corner
C                  of a 10.5 inch (x) by 8.0 inch (y) rectangle with its
C                  lower left corner at the machine origin.
C           FX,FY= the use  coordinate scale factors:
C                  X(machine)=(XL-XO)*FX
C
C
C
C
      SUBROUTINE SCAL(XO,YO,XL,YL,FX,FY)
      COMMON/SCALE/RFX,RFY,RXO,RYO
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTBHP,VTOFSX,VTOFSY
      BYTE BELL
      DATA BELL/'''/
      RXO=XO
      RYO=YO
      IF(XL.EQ.XO)GOTO 100
      IF(YL.EG.YO)GOTO 200
      FX=1050./(XL-XO)
      FY=800./(YL-YO)
```

```
C
C   FX and FY are VT-11 compatable scale factors, RFX and RFY are the
C   real scale factors needed to plot correctly on the H/P9872A
C
        RFX=FX*10.
        RFY=FY*10.
        RETURN
100     TYPE 1000,BELL,BELL,BELL,BELL
1000    FORMAT(/,' *** ERROR XO=XL in Subroutine "SCAL" ***',4A1,/)
        CALL EXST(2)
200     TYPE 2000,BELL,BELL,BELL,BELL
2000    FORMAT(/,' *** ERROR YO=YI in Subroutine "SCAL" ***',4A1,/)
        CALL EXST(2)
        END
C
```

```
C-----------------------------------------------------------------
C     SPED -- VT-11 EMULATION (EXTRA)
C     SPED
C
C     15-NOV-80        by Erik Antonsson
C
C     Changes max pen speed.
C
C     FORTRAN CALL:
C
C     WHERE:     CALL SPED(ISPEED,IPEN)
C
C                ISPEED= pen speed in cm/sec (1 to 36)
C                IPEN= pen to travel at ISPEED (1 to 4)
C
C-----------------------------------------------------------------
      SUBROUTINE SPED(ISPEED,IPEN)
      COMMCN/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTBHP,VTOFSX,VTOFSY
      BYTE VS(15),ICMD(6)
      DATA VS/'V',';','S',',6* 0,',',6*"6/
      IF(IDEV.NE.1)RETURN
      IDUM=1
      IF(IPEN.LE.0)IPEN=1
      IF(IPEN.GT.4)IPEN=4
      IF(ISPEED.LE.0)ISPEED=1
      IF(ISPEED.GT.36)ISPEED=36
      ENCODE(6,1000,ICMD)ISPEED
1000  FORMAT(I6)
      DO 10 J=1,6
      VS(J+2)=ICMD(J)
10    CONTINUE
      ENCODE(6,1000,ICMD)IPEN
      DO 20 J=1,6
      VS(J+9)=ICMD(J)
20    CONTINUE
      CALL HPDCMD(VS,15)
      RETURN
      END
C
```

```
C-------------------------------------------------------------
C   SPEN -- VT-11 EMULATION (EXTRA)
C   SPEN
C
C        14-NOV-80        by Erik Antonsson
C
C   Selects a pen to write with, and leaves it at machine
C   location 10500,0 in the UP position, unless the pen desired
C   has already been picked up, then no action is taken.  Thus
C   an APNT should always be called after a call to SPEN, because
C   in general the pen position is unpredictable.
C
C   FORTRAN CALL:
C        CALL SPEN(IPEN)
C
C   WHERE:
C        IPEN= pen to be selected (1 to 4)
C
C
      SUBROUTINE SPEN(IPEN)
      COMMON/SPENCM/PIPEN
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTBHP,VTOFSX,VTOFSY
      INTEGER PIPEN
      BYTE SP(3),IP(5)
      DATA SP/'S','P',"60/
      DATA IP/'1','2','3','4',"60/
      IF(IDEV.NE.1)RETURN
      IF(IPEN.GT.4)IPEN=5
      IF(IPEN.LE.0)IPEN=5
      IF(IPEN.EQ.PIPEN)RETURN
      SP(3)=IP(IPEN)
      CALL HPDCMD('PU',2)
      CALL HPDCMD('PA10500,0',9)
      CALL HPDCMD(SP,3)
      PIPEN=IPEN
      RETURN
      END
```

```
C -------------------------------------------------------------------
C
C   SYMB -- VT-11 EMULATION (EXTRA)
C   SYMB
C
C       15-NOV-80      by Erik Antonsson
C
C   Prints a selectable symbol at the end of dach vector.
C
C   FORTRAN CALL:
C       CALL SYMB(IC)
C
C   WHERE:     IC= the symbol to be printed.
C
C       SUBROUTINE SYMB(IC)
        COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
       *IDEV,VTEHP,VIOFSX,VIOFSY
        BYTE IC,SM(3)
        DATA SM/'S','M',"40/
        IF(IDEV.NE.1)RETURN
        SM(3)=IC
        CALL HPDCMD(SM,3)
        RETURN
        END
C
```

```
C
C
C     SYOF -- VT-11 EMULATION (EXTRA)
C     SYOF
C
C            18-NOV-80      by Erik Antonsson
C
C     Turns off the symbol printing.
C
C     FORTRAN CALL:
C            CALL SYOF
C
C
      SUBROUTINE SYOF
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IEUF,IBUFR,
     *IDEV,VTBHP,VTOFSX,VTOFSY
      IF(IDEV.NE.1)RETURN
      CALL HPDCMD('SM',2)
      RETURN
      END
C
```

```fortran
C
C
C      TEXT -- VT-11 EMULATION
C      TEXT
C
C      14-NOV-80      by Erik Artonsson
C
C   Prints out ASCII character string using the current character
C   mode and characteristics.
C
C      FORTRAN CALL:
C         CALL TEXT('string',ICR)
C
C      OR
C         CALL TEXT( TRARY,ICR )
C
C      WHERE:
C         'string' = an ASCII literal string
C         STRARY = an array full of ASCII
C         ICR = the number of carriage returns after the string
C
C   This will return the pen to the position it began writing the
C   text from incremented down by the number of carriage returns.
C   Thus if ICR=0 the pen is returned to where it started.
C
C
      SUBROUTINE TEXT(ITEXT,ICR)
      COMMON/IPENST/IP
      COMMON/NFT/ICST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTBHP,VTOFSX,VTOFSY
      BYTE ITEXT(1),ILB(82),CP(10),ICRA(6),AS(3),SP,ITEXTE(80)
      INTEGER ITEXTI(40)
      EQUIVALENCE (ITEXTE(1),ITEXTI(1))
      DATA ILB/'L',' ','E',80*' '/
      DATA CP/'C',' ','P',' ',' ',' ',6*' '/
      DATA SP/"40/
      DATA AS/"10,"40,"176/
      ISHFT=0
```

```
      DO 10 J1=1,80
      IF(ITEXT(J1).EQ.AS(1))GOTO 10
      IF(ITEXT(J1).GT.AS(3))GOTO 20
      IF(ISHFT.EQ.0.AND.ITEXT(J1).GE.AS(2))GOTO 10
      IF(ITEXT(J1).EQ."16.AND.IDEV.EQ.0)ISHFT=1
      IF(ISHFT.NE.1)GOTO 20
      IF(ITEXT(J1).GT."37)GOTO 20
      IF(ITEXT(J1).EQ."17.AND.IDEV.EQ.0)ISHFT=0
10    CONTINUE
20    DO 22 J=J1-1,1,-1
      IF(ITEXT(J).NE.AS(2))GOTO 23
22    CONTINUE
23    IF(IDEV.EQ.1)GOTO 40
C
C     VT-11 Text routine
C
      IF(INDX.GT.INDXMX-((J/2)+5))CALL DFBDMP
      IDAT(INDX,IBUF)=-30003
      DO 30 IQ=1,J
      ITEXTB(IQ)=ITEXT(IQ)
30    CONTINUE
      IF(ISHFT.EQ.0)GOTO 33
      J=J+1
      ITEXTB(J)="17
33    K=J
      IF((J.AND.1).EQ.1)K=J+1
      IF(K.GT.J)ITEXTB(K)=0
      DO 35 IQ=1,K/2
      IDAT(INDX+IQ+2,IBUF)=ITEXTI(IQ)
35    CONTINUE
      IDAT(INDX+1,IBUF)=ICH
      IDAT(INDX+2,IBUF)=K/2
      INDX=INDX+(K/2)+3
      RETURN
```

```
C
C     H/P Text routine.
C
40         DO 50 I=1,J
           ILB(I+2)=ITEXT(I)
50         CONTINUE
           CALL HPDCMD('PU',2)
           CALL HPDCMD(ILB,J+2)
C
C     Perform carriage return(s)
C
           CALL HPDCMD('CP',2)
           ICRN=1-ICR
           ENCODE(6,1000,ICRA)ICRN
1000       FORMAT(I6)
           DO 70 J=1,6
           CP(J+4)=ICRA(J)
70         CONTINUE
           CALL HPDCMD(CP,10)
C
C     With these comment line "C"s the pen is always up after TEXT
C     Without the "C"s the pen will return to its previous state
C
           IF(IP.EQ.1)CALL HPDCMD('PD',2)
           IF(IP.EQ.-1)CALL HPDCMD('PU',2)
           RETURN
           END
C
```

```
C
C
C      VECT -- VT-11 EMULATION SUBROUTINE
C      VECT
C
C      13-NOV-80      by Erik Antonsson
C
C      Moves the pen in user coordinates REALTIVE to the current location
C      See APNT for further information.
C
       SUBROUTINE VECT(X,Y,L,I,F,T)
       COMMON/SCALE/FX,FY,XO,YO
       COMMON/IPENST/IP
       COMMON/LINEST/IT
       COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR.
*IDEV,VTBHP,VTOFSX,VTOFSY
       BYTE ID(15),IX(6),IY(6),ILTYP(7),ILT(3)
       INTEGER F,T
       DATA ID/'P','0','R','0','0','0','0','0','0',',',
*'0','0','0','0','0','0'/
       DATA ILTYP/'1','2','3','4','5','6','0'/
       DATA ILT/'L','T','0'/
       DATA IF/0/
       IF(IDEV.EQ.1)GOTO 9
       IF(INDX.GT.INDXMX-5)CALL DFBDMP
       IDAT(INDX,IBUF)=-30005
       IF(T.EQ.0)GOTO 1
       IF(T.EQ.7)IDAT(INDX,IBUF)=-30007
       IT=T
       GOTO 2
```

```fortran
1       IF(IT.EQ.7)IDAT(INDX,IBUF)=-30007
2       IDAT(INDX+1,IBUF)=IFIX(X*FX*VTRHP)
        IDAT(INDX+2,IBUF)=IFIX(Y*FY*VTRHP)
        IF(I)4,6,6
4       IDAT(INDX+3,IBUF)=0
        IP=-1
        GOTO 8
6       IDAT(INDX+3,IBUF)=1
        IP=1
7       INDX=INDX+4
        RETURN
5       MX=IFIX((X)*FX)
        MY=IFIX((Y)*FY)
        ENCODE(6,1000,IX)MX
1000    FORMAT(I6)
        ENCODE(6,1000,IY)MY
        DO 10 J=1,6
        JX=J+2
        JY=J+9
        ID(JX)=IX(J)
        ID(JY)=IY(J)
10      CONTINUE
C
C pen up
C
12      IF(I)12,13,14
        CALL HPDCMD('PU',2)
        IP=-1
        GOTO 16
13      IF(IP.GE.0)GOTO 16
C
C pen down
C
14      CALL HPDCMD('PD',2)
        IP=1
```

```
C
C    adjust line type
C
10      IF(T.EQ.0.)GOTO 30
        IF(T.GT.0.)GOTO 20
        CALL HPDCMD('LT',2)
        GOTO 30
20      ILT(3)=ILTYP(T)
        CALL HPDCMD(ILT,3)
C
C    plot vector
C
30      CALL HPDCMD(ID,15)
        RETURN
        END
C
```

```
C --------------------------------------------------------------
C
C HPDCMD -- H/P DATA COMMAND
C HPDCMD
C
C      15-NOV-80        by Erik Antonsson
C
C A subroutine to pass an ASCII command character string
C to the IB-11
C interface and (usually) on the the Hewlett-Packard 9872A
C pen plotter, and then wait for the command to complete.
C Each command string is terminated with a line feed (:12),
C and each LBstring command is terminated with :3,:12.
C
C FORTRAN CALL:
C      CALL HPDCMD(IBDA,N)
C
C WHERE:
C
C      IBDA = THE ASCII CHARACTER STRING TO BE SENT THROUGH THE
C             IB-11 IN THE LOW BYTE OF THE DATA REGISTER
C      N= THE NUMBER OF CHARACTERS TO BE SENT
C
C
      SUBROUTINE HPDCMD(IBDA,N)
      BYTE IBDA(1)
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTBHP,VTCFSX,VTOFSY
      IF(IDEV.NE.1)RETURN
      IEEE=IBUP(0,1,IBDA,N)
      IF(IEEE.LT.0)CALL IEEER(IEEE)
      IF((IBDA(1).NE.'I'.AND.IBDA(1).NE.'1').OR.(IBDA(2).NE.'B'.AND.
     &IBDA(2).NE.'b'))GOTO 20
      IEEE=IBUP(0,1,"03,1)
      IF(IEEE.LT.0)CALL IEEER(IEEE)
   20 IEEE=IBUP(0,1,"012,1)
      IF(IEEE.LT.0)CALL IEEER(IEEE)
      RETURN
      END
```

```
C   ------------------------------------------------------------------
C
C   DFBDMP -- DISPLAY FILE BUFFER DUMP
C   DFBDMP
C
C       21-APR-81       by Erik Antonsson
C
C   Dumps the next buffer of plot data across the network to the plot
C   task (VTPLOT) driving the VT-11 on the 11/40.
C
        SUBROUTINE DFBDMP
        COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IEYTE,IBUF,IBUFR,
     *IDEV,VTBHP
        IF(IDEV.NE.0)RETURN
        IF(INDX.LE.1)GCTO 20
        IDAT(INDX,IBUF)=-30008
        CALL WAITNT(,IOST(1,IBUFR))
        IF(IOST(1,IBUFR).LT.0)CALL NETERR('WAITNT','DFBDMP',IBUFR)
        IBUF=IBUF+1
        IF(IBUF.GT.2)IBUF=1
        IBUFR=IBUFR+1
        IF(IBUFR.GT.2)IBUFR=1
        CALL SNDNT(2,IOST(1,IBUFR),IBYTE,IDAT(1,IBUFR))
        IF(IOST(1,IBUFR).LT.0)CALL NETERR('SNDNT ','DFBDMP',IBUFR)
20      INDX=1
        RETURN
        END
C
```

```fortran
C ------------------------------------------------------------
C
C      NETERR
C
C      23-APR-81        by Erik Antonsson
C
C      Prints an error message if the NETwork produces an error.
C      FORTRAN CALL:
C          CALL NETERR(ISUB1,ISUB2,IA)
C
C      WHERE:
C             ISUB1= 6 ASCII character name of the NETWORK subroutine
C             ISUB2= 6 ASCII name of the user routine that called ISUB1
C             IA = The value of the second subscript on IOST
C
       SUBROUTINE NETERR(ISUB1,ISUB2,IA)
       COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
      *IDEV,VTBHP
       BYTE ISUB1(6),ISUB2(6),BELL
       DATA BELL/"7/
       TYPE 1000,ISUB1,ISUB2,IOST(1,IA)
 1000  FORMAT(/,' *** NETWORK ERROR ***',/,' ',6A1,'''','.',/,
      *'   FROM SUBROUTINE: ',6A1,'   IN: ',/,
      *'   ERROR/COMPLETEION CODE= ',I6)
       TYPE 1002
 1002  FORMAT(' See Appendix F or Chapter 8 of Volume 2 of the RSX',
      & ' DECnet Users Guide.')
       IF(IOST(1,IA).EQ.-40)GOTO 30
       IF(IOST(1,IA).EQ.-7)GOTO 50
       TYPE 1005,BELL,BELL,BELL,BELL
 1005  FORMAT(1X,4A1,/)
       RETURN
```

```fortran
C
   30     IERR=IOST(2,IA)
          IF(IERR.EQ.-6)TYPE 1030
 1030     FORMAT(' NET NOT INSTALLED ON LOCAL NODE.')
          IFD=2
          LUN=6
          CALL ERRMSG(LUN,IERR,IFD)
          RETURN
C
   50     TYPE 1050,IOST(2,IA)
 1050     FORMAT(' CONNECTION REJECTED BY THE NETWORK',/,
         *'   NETWORK REJECT CODE=',I6)
          IF(IOST(2,IA).EQ.39)TYPE 1055
 1055     FORMAT(' REMOTE NODE UNREACHABLE.')
          TYPE 1060,BELL,BELL,BELL,BELL
 1060     FORMAT(' See appendix A of Volume 2 of the RSX',
         *'   DECnet Users Guide.',4A1,/)
          RETURN
          END
C
```

```
C
C       ERRMSG -- SUBROUTINE TO PRINT OUT FCS AND DSW ERRORS
C       ERRMSG
C
C           5-OCT-81        by Erik Antonsson
C
C       FORTRAN CALL:
C           CALL ERRMSG(IUN,IERR,IFD)
C
C       WHERE:
C           IUN = An available Logical Unit Number for temporary use.
C           IERR = The value of the error code
C           IFD = 1 for an FCS error (File Control System)
C               = 2 for a  DSW error (Directive Status Word)
C
C
        SUBROUTINE ERRMSG(LUN,IERR,IFD)
        DIMENSION FDM(2)
        BYTE MSG(43),IBYTE,BELL
        DATA BELL/"7/
        DATA FDM/'FCS#','DSW#'/
        IF(IERR.GE.0)RETURN
        IN=ABS(IERR)-1
        IF(IFD.EQ.2)IN=IN+128
        OPEN(UNIT=LUN,NAME='LB:[1,2]QIOSYM.MSG;1',READONLY,
     &TYPE='OLD')
        IF(IN.LT.1)GOTO 110
        DO 100 IQ=1,IN
100     READ(LUN,1040)IBYTE
        CONTINUE
1040    FORMAT(A1)
110     READ(LUN,1050)MSG
1050    FORMAT(43A1)
        TYPE 1060,FDM(IFD),IERR,MSG,BELL,BELL,BELL,BELL
1060    FORMAT(1X,'*** ERROR: ',A4,I4,2X,43A1,'***',4A1)
        CLOSE(UNIT=LUN)
        RETURN
        END
```

```
C
C
C     IEEER -- ROUTINE TO PRINT IEEE ERRORS
C     IEEER
C
C
      SUBROUTINE IEEER(IEEE)
      COMMON/NET/IOST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTEHP
      IF(IDEV.NE.1)RETURN
      IF(IEEE.GE.0)RETURN
      TYPE 1000,IEEE,7,7,7,7
1000  FORMAT(' *** IEEE INTERFACE ERROR#',I6,' ***',4A1)
      CALL EXST(2)
      END
```

```
; VHMAC.MAC -- IB-11 IEEE BUS INTERFACE REGISTER COMMUNICATION
; VHLIB
;
;       15-NOV-80        by Erik Antonsson
;
;-------------------------------------------------------------
;
;       Copyright (C) 1982  Massachusetts Institute of Technology
;                           Cambridge, Massachusetts
;
; This  software is  furnished  under an  agreement  for use  only on a
; single computer  system and may  be copied  only with  the inclusion
; of the  above copyright  notice.    This software,  or any other copies
; thereof, may  not be  provided  or otherwise  made available  to any
; other person except for  use on such system and to one who subscribes
; to these  agreement terms.   Title to and  ownership of the software
; shall at all times remain in MIT.
;
; The information in this document is subject to change without notice
; and should not be construed as a commitment by MIT.
;
; MIT assumes  no responsibility  for the use  or reliability  of its
; software.
;-------------------------------------------------------------
;
; NARG -- RETURNS THE NUMBER OF ARGUMENTS IN THE PREVIOUS SUBROUTINE CALL
;
;       CALL NARG(N)
;
; This must be the first executable statement in the subroutine.
;
        .TITLE  NARG
        .IDENT  /V01A/
        .PSECT
NARG::  MOV     2(SP),R4
        MOV     (R4),@2(R5)
        RETURN
        .END
```

```
C ----------------------------------------------------------------
C
C     MGRAPH -- Multiple line graphing routine
C     MGRAPH
C
C     24-APR-81        by Erik Antonsson
C
C     FORTRAN CALL:
C        CALL MGRAPH(A,NROWS,NCOLS,[IROWS],[ITIMPL],[STIME],[FTIME],
C       &[BAD],[ICL],[ILT],[ILABEL],[DLABEL],[TITLE],[IAUTSC],[BNDRY],
C       &[IPLDEV],[ISPEED])
C     WHERE:
C        A        The array of data to be plotted
C        NROWS    The number of rows in A.   Each row can be pl tted
C                 against one chosen row or time.
C        NCOLS    The number of columns in A (The number of points
C                 on each curve).
C        (N.B. NONE of the first three arguments can be defaulted.)
C        IROWS    An array NROWS long containing information of which
C                 row to plot against which row:
C
C     (if this is a cross plot [ITIMPL=0])
C        IROW(1) Contains the row number of the Independent valrable
C        IROWS(2)..IROWS(NROWS)  Contains the numbers of the rows to be
C                 plotted against IROW(1).
C        [Default]   IROW(J) is set to J, for J=1,NROWS
C                 If IROW(2) (or higher) =0: No row will be plotted for that
C                 IROW.
C
C     (if this is a "TIME" plot [ITIMPL=1])
C        IROWS(1)..IROWS(NROWS)  Contairs the numbers of the rows to be
C                 plotted against  "TIME".
C        [Default]   IROW(J) is set to J, for J=1,NROWS
C                 If IROW(2) (or higher) =0: No row will be plotted for that
C                 IROW.
C        IROW must be DIMENSION IROW(NROWS)
C
```

```
C     ITIMPL  =0 for a cross plot A(IROWS(1),J) vs A(IROWS(2,...,NROWS),J)
C             =1 for a TIME plot.  If a TIME plot,
C             the independent variable begins at STIME and proceeds
C             in even increments up to FTIME.  All rows of A specified
C             in IROWS are plotted against TIME.
C     [Default] =1
C     STIME   The start time if ITIMPL=1 (REAL Floating Point Argument)
C     [Default] =0.0
C     FTIME   The final time if ITIMPL=1 (REAL Floating Point Argument)
C     [Default] =1.0
C     BAD     An array NROWS long containing the value of BAD points
C             not to be plotted for the corresponding row in A.
C             REAL Floating Point Argument.
C     [Default] =-9999.0 for: BAD(J),J=1,NROWS
C     ICL     An array NROWS long containing the pen number to use
C             for the corresponding row in A if the H/P 9872A is used.
C     [Default] =1 for: ICL(J),J=1,NROWS
C     ILT     An array NROWS long containing the line type to use for
C             the corresponding row in A.  If the H/P is used, see
C             the argument list for Subroutine APNT in the VTLIB
C             document.  If the VT-11 is used:
C             =-1  Solid line
C             =0   No change in line type
C             =7   Dots at the vector endpoints (at the poi ts in A)
C             =ILT+10 For periods at the vector endpoints
C     [Default] =-1 for: ILT(J),J=1,NROWS
C     ILABEL  ASCII string for the independent axis label
C     DLABEL  ASCII string for the dependent axis label
C     TITLE   ASCII string for the graph title
C             N.B. All ASCII strings must be 60 characters or less in
C             length.  The arguments can be specified as a quoted literal
C             or an array.
C     [Default] All text strings default to a space.
C     IAUTSC  =1 for auto scaling
C             =2 for user selected scaling
C     [Default] =1
```

```
C   BNDRY   A 4 element REAL array containing: XMIN,XMAX,YMIN,YMAX for
C           this plot if IAUTSC=0.  If IAUTSC=1 BNDRY is returned with
C           the values used in the current plot.
C        [Default] No default available, if defaulted, Autoscaling is invoked
C   IPLDEV  =1 for the VT-11
C           =2 for the H/P 9872A
C           =3 for the VT-11 then a hardcopy on the H/P if desired.
C        [Default] =3
C   ISPEED  = The Hardcopy pen speed in cm/sec, 1 to 36 cm/sec.
C           = 0 MGRAPH will prompt the user to select the pen speed.
C        [Default] =10
C   ITMDT  = 0 No Time/Date will be written in the upper left of HP
C          = 1 Time/Date will be written
C        [Default] =0 No Time/Date
```

```
      SUBROUTINE MGRAPH(A,NROWS,NCOLS,IROWS,ITIMPD,STIMED,FTIMED,
     &BAD,ICL,ILT,ILABEL,DLABEL,TITLE,IAUTSD,BNDRYD,IPLDVD,
     &ISPED,ITMDT)
      COMMON/NET/ICST(2,2),IDAT(256,2),INDX,INDXMX,IBYTE,IBUF,IBUFR,
     *IDEV,VTBHP,VTOFSX,VTOFSY
      DIMENSION A(NROWS,NCOLS)
      DIMENSION IROWS(NROWS),BAD(NROWS),ICL(NROWS),ILT(NROWS)
      DIMENSION BNDRY(4),BNDRYD(4)
      DIMENSION IPM(19)
      BYTE ILABEL(1),TITLE(1),DLABEL(1)
      BYTE IHCPY,ARROW(4),ITIME(20)
      DATA ARROW/"16, 30, 17, 0/
      CALL NARG(N)
      TYPE 996
996   FORMAT(' MGRAPH plot calculations....')
      BADD=-9999.0
      DO 2500 IQ=1,19
      IPM(IQ)=-1
2500  CONTINUE
      CALL GETADR(IPM,A,NROWS,NCOLS,IROWS,ITIMPD,STIMED,FTIMED,
     &BAD,ICL,ILT,ILABEL,DLABEL,TITLE,IAUTSD,BNDRYD,IPLDVD,
     &ISPED,ITMDT)
      DO 2530 IQ=(N+1),19
      IPM(IQ)=-1
2530  CONTINUE
      IF(IPM(1).NE.-1.AND.IPM(2).NE.-1.AND.IPM(3).NE.-1)GOTO 2650
      TYPE 997,7,7,7,7
997   FORMAT(' *** ERROR: ILLEGALLY DEFAULTED ARGUMENTS ***',4A1)
      RETURN
```

```
2650    IF(IPM(5).EQ.-1)ITIMPL=1
        IF(IPM(5).NE.-1)ITIMPL=ITIMPD
        IF(IPM(6).EQ.-1)STIME=0.0
        IF(IPM(6).NE.-1)STIME=STIMED
        IF(IPM(7).EQ.-1)FTIME=1.0
        IF(IPM(7).NE.-1)FTIME=FTIMED
        IF(IPM(14).EQ.-1)IAUTSC=1
        IF(IPM(14).NE.-1)IAUTSC=IAUTSD
        IF(IPM(15).NE.-1)GOTO 2755
        IAUTSC=1
        GOTO 2760
2755    DO 2756 IQ=1,4
2756    BNDRY(IQ)=BNDRYD(IQ)
2760    IF(IPM(16).EQ.-1)IPLDEV=3
        IF(IPM(16).NE.-1)IPLDEV=IPLDVD
        IF(IPM(17).EQ.-1)ISPEED=10
        IF(IPM(17).NE.-1)ISPEED=ISPED
        IF(IPM(4).EQ.-1)INDP=1
        IF(IPM(4).NE.-1)INDP=IHOWS(1)
C
C   Determine if a TIME plot is desired.
C
3       IF (ITIMPL.NE.1)GO TO 510
500     TINC=(FTIME-STIME)/FLOAT(NCOLS)
        XMIN=STIME
        XMAX=FTIME
```

```
C
C   Find the Maxima and Minima of the data.
C
510     IF(IAUTSC.EQ.1) GO TO 418
        XMIN=BNDRY(1)
        XMAX=BNDRY(2)
        YMIN=BNDRY(3)
        YMAX=BNDRY(4)
        GO TO 650
418     IF(ITIMPL.EQ.1) GO TO 419
        IJUMP=0
        IF(IPM(8).NE.-1)BADI=BAD(INDP)
        IF(IPM(8).EQ.-1)BADI=BADD
        DO 7 J=1,NCOLS
        IF(A(INDP,J).EQ.BADI)GO TO 7
        IF(IJUMP.EQ.1)GO TO 6
        XMAX=A(INDP,J)
        XMIN=A(INDP,J)
6       XMAX=AMAX1(XMAX,A(INDP,J))
        XMIN=AMIN1(XMIN,A(INDP,J))
        IJUMP=1
7       CONTINUE
419     IJUMP=0
        IF(ITIMPL.EQ.1)ISL=1
        IF(ITIMPL.EQ.0)ISL=2
        DO 12 K=ISL,NROWS
        IF(IPM(4).NE.-1)IDEP=IROWS(K)
        IF(IPM(4).EQ.-1)IDEP=K
        IF(IDEP.LT.1.OR.IDEP.GT.NROWS)GOTO 12
        IF(IPM(8).NE.-1)BADP=BAD(IDEP)
        IF(IPM(8).EQ.-1)BADP=BADD
```

```
      DO 9 J=1,NCOLS
      IF(A(IDEP,J).EQ.BADP)GO TO 9
      IF(IJUMP.EQ.1)GO TO 8
      YMIN=A(IDEP,J)
      YMAX=A(IDEP,J)
    8 YMAX=AMAX1(YMAX,A(IDEP,J))
      YMIN=AMIN1(YMIN,A(IDEP,J))
      IJUMP=1
    9 CONTINUE
   12 CONTINUE
C
C Check bounds on Max and Mins
C
  650 IF(XMAX.EQ.0.0.AND.XMIN.EQ.0.0)GOTO 651
      IF(XMAX.EQ.XMIN)GOTO 649
      IF(ABS(XMAX/(XMAX-XMIN)).LT.3200.0)GOTO 652
      XMAX=XMAX+(0.5/3200.0)*XMAX
      XMIN=XMIN-(0.5/3200.0)*XMIN
      GOTO 652
  651 XMAX=1.0
      XMIN=-1.0
  652 IF(YMAX.EQ.0.0.AND.YMIN.EQ.0.0)GOTO 653
      IF(YMAX.EQ.YMIN)GOTO 648
      IF(ABS(YMAX/(YMAX-YMIN)).LT.3200.0)GOTO 654
      YMAX=YMAX+(0.5/3200.0)*YMAX
      YMIN=YMIN-(0.5/3200.0)*YMIN
      GOTO 654
  653 YMAX=1.0
      YMIN=-1.0
  654 CONTINUE
```

```fortran
        IF(IAUTSC.EQ.0)GOTO 655
        IF(IPM(15).EQ.-1)GOTO 655
        BNDRYD(1)=XMIN
        BNDRYD(2)=XMAX
        BNDRYD(3)=YMIN
        BNDRYD(4)=YMAX
655     CONTINUE
        IF((XMAX-XMIN).GE.0.0)GOTO 656
        XTMP=XMIN
        XMIN=XMAX
        XMAX=XTMP
656     IF((YMAX-YMIN).GE.0.0)GOTO 657
        YTMP=YMIN
        YMIN=YMAX
        YMAX=YTMP
C
C Find Scaled Maximum and Minimum for plotting on the X axis.
C
657     XRANGE=XMAX-XMIN
        DO 10 I=-3,3
        DX=10.**I
        XTIK=XRANGE/DX
        IF(XTIK.LE.10.) GO TO 11
10      CONTINUE
11      MAXRND=0
        MINRND=0
        IRND=0
        REMAX=AMOD(XMAX,DX)
        REMIN=AMOD(XMIN,DX)
        IF(XMAX.GT.0..AND.REMAX.NE.0.)MAXRND=1
        IF(XMIN.LT.0..AND.REMIN.NE.0.)MINRND=-1
        IF(REMAX.EQ.0..AND.REMIN.EQ.0.)IRND=1
        ITIK=2*IFIX(.5*(XRANGE+DX)/DX)+IRND+MAXRND-MINRND+1
        IF(ITIK.GE.6.AND.ITIK.LE.12)GO TO 20
        IF(ITIK.NE.4.AND.ITIK.NE.5)GO TO 14
        ITIK=2*ITIK-1
        DX=DX/2.
        GO TO 15
```

```
14    IF(ITIK.NE.3)GO TO 15
      ITIK=4*ITIK-3
      DX=DX/4
15    CONTINUE
20    ISCMAX=IFIX(XMAX/DX+FLOAT(MAXRND))
      ISCMIN=IFIX(XMIN/DX+FLOAT(MINRNI))
      XSCMAX=FLOAT(ISCMAX)*DX
      XSCMIN=FLOAT(ISCMIN)*DX
      ITK=ISCMAX-ISCMIN+1
C
C  Find scaled maximum and minimum for plotting on the Y axis.
C
      YRANGE=YMAX-YMIN
      DO 25 J=-3,3
      DY=10.**J
      YTIK=YRANGE/DY
      IF(YTIK.LE.10.) GO TO 30
25    CONTINUE
30    MAXRND=0
      MINRND=0
      JRND=0
      REMAX=AMOD(YMAX,DY)
      REMIN=AMOD(YMIN,DY)
      IF(YMAX.GT.0..AND.REMAX.NE.0.)MAXRND=1
      IF(YMIN.LT.0..AND.REMIN.NE.0.)MINRND=-1
      IF(REMAX.EQ.0..AND.REMIN.EQ.0.)JRND=1
      JTIK=2*IFIX(.5*(YRANGE+DY)/DY)+JRND+MAXRND-MINRND+1
      IF(JTIK.GE.6.AND.JTIK.LE.12)GO TO 40
      IF(JTIK.NE.4.ANL.JTIK.NF.5)GO TO 33
      JTIK=2*JTIK-1
      DY=DY/2.
      GO TO 35
33    IF(JTIK.NE.3)GO TO 35
      JTIK=4*JTIK-3
      DY=DY/4
```

```
35      CONTINUE
40      JSCMAX=IFIX(YMAX/DY+FLOAT(MAXRND))
        JSCMIN=IFIX(YMIN/DY+FLOAT(MINRND))
        YSCMAX=FLOAT(JSCMAX)*DY
        YSCMIN=FLOAT(JSCMIN)*DY
        JTK=JSCMAX-JSCMIN+1

C
C       Scale plot
C
        XSRANJ=XSCMAX-XSCMIN
        YSRANJ=YSCMAX-YSCMIN
        XLSHFT=XSCMIN-.30*XSRANJ
        YBSHFT=YSCMIN-.36*YSRANJ
        XRSHFT=.16*XSRANJ
        YTSHFT=.23*YSRANJ

C
C       Begin the plot
C
        IF((IPLDEV.AND.1).EQ.1)IDEV=0       !If any Soft plot desired do first
        IF(IPLDEV.EQ.2)IDEV=1               !If Hardcopy only
        NETOFF=0
        IF((ISPEED.LT.0.OR.ISPEED.GT.36).AND.IPLDEV.EQ.2)GOTO 772
50      CALL INIT(IDEV)
        IF(IDEV.NE.0)GOTO 51

C
C       If plot to VT-11, readjust scaling so plot will fill screen
C
        VTBHP=0.115
        VTOFSX=-100.
        VTOFSY=-100.
```

```
C     If VT-11 softcopy plot requested, check if the NETwork is operative
C
C
      IF(IOST(1,1).LT.0.OR.IOST(1,2).LT.0)NETOFF=1
      IF(NETOFF.EQ.0)GOTO 51
      GOTO 770
   51 TYPE 1000
 1000 FORMAT(' Plotting....')
      CALL SPEN(1)
C
   52 CALL SPED(ISPEED,1)
      CALL SCAL(XLSHFT,YESHFT,XSCMAX+XRSHFT,YSCMAX+YTSHFT,FX,FY)
C
C     Draw box
C
      CALL APNT(XSCMIN,YSCMIN,-1,-5,0,-1)
      CALL VECT(XSRANJ,0.,0,0,0)
      CALL VECT(0.,YSRANJ,0,0,0,0)
      CALL VECT(-XSRANJ,0.,0,0,0,0)
      CALL VECT(0.,-YSRANJ,0,0,0,0)
      CALL VECT(0.,0.,-1,0,0)
      XINC=XSRANJ/FLOAT(NCOLS)
C
C     Draw data curves
C
      IF(ITIMPL.EQ.1)ISL=1
      IF(ITIMPL.EQ.0)ISL=2
      DO 57 K=ISL,NROWS
      IF(IPM(4).NE.-1)IDEP=IROWS(K)
      IF(IPM(4).EQ.-1)IDEP=K
      IF(IPM(8).NE.-1)BADI=BAD(INDP)
      IF(IPM(8).EQ.-1)BADI=BADI
      IF(IPM(8).NE.-1)BADP=BAD(IEEP)
      IF(IPM(8).EQ.-1)BADP=BADI
      IF(IDEP.LE.0.OR.IDEP.GT.NROWS)GOTO 57
```

```
      IBAD=0
      IFIRST=-5
      IF(IPM(10).NE.-1)ILINE=ILT(IDEP)
      IF(IPM(10).EQ.-1)ILINE=-1
      IF(ILINE.EQ.8)ILINE=-1
      IF(ILINE.LT.-1)ILINE=-1
      IF(ILINE.GT.17)ILINE=9
      IF(ILINE.GT.8) CALL SYMB('.')
      IF(ILINE.GT.8)ILINE=ILINE-10
      DO 55 J=1,NCOLS
      IF(ITIMPL.EQ.0)GOTO 558
      X=STIME+FLOAT(J-1)*TINC
      GOTO 563
 558  X=A(INDEP,J)
      IF(X.EQ.EADI)GOTO 54
 563  IF(X.LT.XMIN.OR.X.GT.XMAX)GO TO 54
      IF(A(IDEP,J).EQ.BADP)GO TO 54
      IF(A(IDEP,J).LT.YMIN.OR.A(IDEP,J).GT.YMAX)GO TO 54
      IF(IBAD.EQ.1)GOTO 53
      IF(IPM(9).NE.-1)ICOLOR=ICL(IDEP)
      IF(IPM(9).EQ.-1)ICOLOR=1
      IF(IFIRST.NE.0)CALL SPEN(ICOLOR)
      IF(IFIRST.NE.0)CALL SPED(ISPEED,ICOLOR)
      CALL APNT(X,A(IDEP,J),0,IFIRST,0,ILINE)
      ILINE=0
      IFIRST=0
      GOTO 55
 53   CALL APNT(X,A(IDEP,J),0,-5,0,0)
      IFIRST=0
      IBAD=0
      GOTO 55
 54   IBAD=1
 55   CONTINUE
      IBAD=0
      CALL SYOF
 57   CONTINUE
      CALL SPEN(1)
      CALL SPED(ISPEED,1)
```

```
C
C      Draw tickmarks and scale on the Y axis.
C
60     NTAG=1
       YTKLNG=15./FX
       YLOCX=110./FX
       YLOCY=9./FY
       DO 70 J=JSCMIN,JSCMAX
       NTAG=NTAG+1
       YSCNUM=FLOAT(J)*DY
       CALL APNT(XSCMIN,YSCNUM,0,-5,0,-1)
       CALL VECT(-YTKLNG,0.,0,0,0,0)
       CALL VECT(-YLOCX,-YLOCY,0,-5,0,0)
       IF(DY.GE.10.)GOTO 68
       IF(DY.EQ.5.0.OR.DY.EQ.1.0)GOTO 67
       IF(DY.EQ.2.5.OR.DY.EQ.0.5.OR.DY.EQ.0.1)GOTO 66
       IF(DY.EQ.0.25.OR.DY.EQ.0.05.OR.DY.EQ.0.01)GOTO 65
       CALL NMBR(NTAG,YSCNUM,'F9.4')
       GO TO 70
65     CALL NMBR(NTAG,YSCNUM,'F9.3')
       GO TO 70
66     CALL NMBR(NTAG,YSCNUM,'F9.2')
       GOTO 70
67     CALL NMBR(NTAG,YSCNUM,'F9.1')
       GOTO 70
68     CALL NMBR(NTAG,YSCNUM,'F9.0')
70     CONTINUE
```

```
C
C  Draw tickmarks and scale on the X axis.
C
      XTKLNG=15./FY
      XLOCX=75./FX
      XLOCY=25./FY
      DO 80 I=ISCMIN,ISCMAX
      NTAG=NTAG+1
      XSCNUM=FLOAT(I)*DX
      CALL APNT(XSCNUM,YSCMIN,0,-5,0,0)
      CALL VECT(0.,-XTKLNG,0,0,0,0)
      CALL VECT(-XLOCX,-XLOCY,0,-5,0,0)
      IF(DX.GE.10.)GOTO 78
      IF(DX.EQ.5.0.OR.DX.EQ.1.0)GOTO 77
      IF(DX.EQ.2.5.OR.DX.EQ.0.5.OR.DX.EQ.0.1)GOTO 76
      IF(DX.EQ.0.25.OR.DX.EQ.0.05.OR.DX.EQ.0.01)GOTO 75
      CALL NMBR(NTAG,XSCNUM,'F9.4')
      GO TO 80
   75 CALL NMBR(NTAG,XSCNUM,'F9.3')
      GO TO 80
   76 CALL NMBR(NTAG,XSCNUM,'F9.2')
      GO TO 80
   77 CALL NMBR(NTAG,XSCNUM,'F9.1')
      GOTO 80
   78 CALL NMBR(NTAG,XSCNUM,'F9.0')
   80 CONTINUE
C
C  Draw X axis title
C
      XTLOCY=65./FY
      IF(IPM(11).EQ.-1)GOTO 749
      CALL APNT(XSCMIN,YSCMIN-XTLOCY,0,-5,0,0)
      CALL TEXT(ILABEL,0)
```

```fortran
C
C   Draw Y axis title
C
749    YTLOC=20./FY
       IF(IPM(12).EQ.-1)GOTO 760
       IF(IDEV.EQ.1)GOTO 750
       CALL VECT((-90./FX),(-24./FY),0,-5,0,0)
       CALL TEXT(DLABEL,-1)
       CALL TEXT(ARROW,0)
       GOTO 760
750    CALL APNT(XLSHFT+(72./FX),YSCMIN,0,-5,0,0)
       CALL CHAR(0.,0.,0.,1.)
       CALL TEXT(DLABEL,0)
       CALL CHDF
C
C   Draw Plot Title
C
760    IF(IFM(13).EQ.-1)GOTO 762
       IF(IDEV.EQ.1)CALL APNT(XSCMIN,YSCMAX+(15./FY),0,-5,0,0)
       IF(IDEV.EQ.0)CALL APNT(XSCMIN-(45./FX),YSCMAX+(15./FY),
      *0,-5,0,0)
       CALL TEXT(TITLE,0)
762    IF(IDEV.NE.1)GOTO 765
C
C   Plot Time and Date in upper right corner if on H/P 9872A
C
       IF(IPM(19).EQ.-1)GOTO 765
       IF(ITMDT.EQ.0)GOTO 765
       CALL APNT(XSCMIN+XSRANJ-(120./FX),YSCMIN+YSRANJ+(90./FY),
      *0,-5,0,0)
       CALL TIME(ITIME(1))
       CALL DATE(ITIME(11))
       ITIME(9)=40
       ITIME(10)=40
       ITIME(20)=0
       CALL CHAR(0.13,0.22,0.,0.,0.)
       CALL TEXT(ITIME,0)
       CALL CHDF
```

```
C
C     Test for a hardcopy if choice was requested
C
765   IF(IDEV.EQ.1)GOTO 800       !already plotted a hardcopy
      IF(IPLDEV.EQ.1)GOTO 810     !VT-11 only
      CALL DFBDMP
770   TYPE 771,?
771   FORMAT('$Do you wish to get a hard copy of this plot?
     &  [Y/N]:  ',A1)
      ACCEPT 2011,IHCPY
2011  FORMAT(A1)
      IDEV=0
      IF(IHCPY.EQ."131.OR.IHCPY.EQ."171)IDEV=1
      IF(IDEV.NE.1)GOTO 775
C
772   IF(ISPEED.GT.0.AND.ISPEED.LE.36)GOTO 774
      TYPE 1068,?
1068  FORMAT('$Enter the Hardcopy pen speed in cm/sec.
     &  [Integer, Range:1-36, Default:10]: ',A1)
      ACCEPT 1069,ISPEED
1069  FORMAT(I6)
      IF(ISPEED.LT.1.OR.ISPEED.GT.36)ISPEED=10
C
774   IF(IDEV.EQ.1)GOTO 50        !Plot a hardcopy
775   IF(NETOFF.EQ.1)RETURN       !If the NETwork is OFF and no hardcopy
      CALL ENDP                   !ENDP for VT-11
      RETURN
800   IDEV=1
      CALL ENDP                   !ENDP for H/P 9872A
      IF(NETOFF.EQ.1)RETURN       !If NET is OFF, no VT-11 ENDP
      IF(IPLDEV.EQ.2)RETURN       !If no Softcopy plotted, no VT-11 ENDP
810   IDEV=0
      CALL ENLP                   !ENDP for VT-11
      RETURN
      END
```

VTLIB.FTN

Modified Andrew Rubel and Associates Plotting package
for a VT-11 running under RSX11-M V3.2

```
C------------------------------------------------
C         SUBROUTINE VTINIT(IUN,NWORDS,IPAUSD)
C------------------------------------------------
C
C THIS SUBROUTINE ATTACHES AND INITIALIZES THE GRAPHICS DISPLAY.
C IT MUST BE CALLED BEFORE ANY OTHER CALLS TO THE DISPLAY SUBROUTINES.
C
C INPUT ARGS:
C      IUN......LOGICAL UNIT NUMBER TO BE USED FOR THE GRAPHICS DISPLAY.
C      NWORDS...NUMBER OF WORDS IN THE USER'S DISPLAY FILE.
C      IPAUSE...= 1 FOR A PAUSE IN VTEND
C
C NOTES:
C      1.  THE USER MUST DEFINE A LABELED COMMON BLOCK NAMED DFILE
C          FOR THE DISPLAY FILE.
C
C      2.  DEFAULTS SET BY THIS ROUTINE ARE:
C          GRAPHIC MODE = LONG VECTOR.
C          LIGHT PEN NOT SENSITIZED TO HITS.
C          INTENSITY LEVEL 4.
C          FLASH DISABLED.
C          LINE TYPE SOLID.
C
C      3.  THE BEAM IS LOCATED AT THE LOWER LEFT OF THE SCREEN (0,0)
C          AFTER THIS CALL.
C
C      4.  EVENT FLAG NUMBER 1 IS SPECIFIED FOR LIGHT PEN HITS.
C
C>
      EXTERNAL VTEND
      DIMENSION JMODE(10)
      DIMENSION ISB(2),IPARAM(4),IOSB(2)
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,INTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTLIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IRPMAX,IBIT6,IAVVMX,ILVVMX,ICAWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISBFMX
      DATA JMODE/0, 4000, 10000, 14000, 20000, 24000,
     *"30000,"34000,"40000,"44000/
```

```
C
C   Initialize common.
C
4       MAXWD=NWORDS        ! MAX NUMBER OF WORDS IN DISPLAY FILE
        NEXTWD=1            ! NEXT AVAILABLE WORD IN DISPLAY FILE
        IXPOS=0             ! X BEAM POSITION
        IYPOS=0             ! Y BEAM POSITION
        ILPEN=2             ! LIGHT PEN DISABLED
        IINTEN=4            ! INTENSITY LEVEL 4
        IFLASH=2            ! FLASH DISABLED
        ILTYPE=1            ! SOLID LINE TYPE
        IMODE=3             ! LONG VECTOR MODE
        LUNGR=LUN           ! LUN FOR SCOPE
        INDEX=1             ! Index on temporary plot command buffer
        ISBFMX=255          ! Temporary plot buffer maximum size
C
C   BITS AND BYTES, BITES, NIBBLES, AND WHAT HAVE YOU:
C
        IDHALT=".173400     ! GRAPHIC DISPLAY HALT INSTRUCTION
        INTENA=".40000      ! INTENSITY ENABLE BIT
        INTDIS=0            ! INTENSITY NO ENABLE
        MODIFY=0            ! MODIFY FLAG
        IMINUS=".20000      ! SIGN BIT FOR LONG VECTOR, ABS VECT, OFFSET
        ICNTRL=".100000     ! GRAPHIC INSTRUCTION CONTROL BIT
        ILVMAX=1024         ! MAX LONG VECTCR X OR Y DISPLACEMENT
        IAVMAX=1024         ! ABSOLUTE VECTOR MAX COORDS
        IRPMAX=63           ! RELATIVE POINT MAX X OR Y DISPLACEMENT
        IBIT6=".100         ! BIT NUMBER 6
        IAVVMX=0.75*IAVMAX  ! Absolute vertical maximum
        ILVVMX=0.75*ILVMAX  ! Long vector vertical maximum
        ICHWD=14            ! Character width (14 physical units)
        ICHHT=24            ! Character height (24 physical units)
        MAXCHR=80           ! Maximum number of characters per ETEXT call
        IPAUSE=IPAUSD
        DO 5 J=1,10
5       MODE(J)=JMODE(J)    ! 10 POSSIBLE GRAPHIC MODES
```

```
C
C  Initialize the display buffer header.
C  The first 30 words must be inplace for the GR driver.  They are
C  the TRACKING CROSS used by DECgraphic-11 and hence also by the
C  GR driver.  This Tracking Cross subpicture is not used, but
C  the driver expects to see valid graphic data to be used
C  immediately beginning at word 31.
C
      IDBUFF(1) = "177777     ! Load register B: 63 Graphplot increment
      IDBUFF(2) = "170240     ! Load Reg A: NoStopInt,EngLPInt,EnbItalSel
      IDBUFF(3) = "117524     ! PtMode,EnbInten:5,EnbBlnk,EnbLineSel:0
      IDBUFF(4) = "0          ! move to 0,0
      IDBUFF(5) = "0
      IDBUFF(6) = "104140     !Short Vect Mode, No Inten change, EnbLPInt
      IDBUFF(7) = "50416      !Inten,+42,+16
      IDBUFF(8) = "40134      !Inten, +0,-34
      IDBUFF(9) = "65124      !Inten,-24,-24
      IDBUFF(10)= "67000      !Inten,-34, +0
      IDBUFF(11)= "65024      !Inten,-24,+24
      IDBUFF(12)= "40034      !Inten, +0,+34
      IDBUFF(13)= "45024      !Inten,+24,+24
      IDBUFF(14)= "47000      !Inten,+34,  0
      IDBUFF(15)= "45124      !Inten,+24,-24
      IDBUFF(16)= "30516      !NoInten,-42,-16
      IDBUFF(17)= "70416      !Inten,-42,+16
      IDBUFF(18)= "10516      !NoInten,+42,-16
      IDBUFF(19)= "70516      !Inten,-42,-16
      IDBUFF(20)= "10416      !NoInten,+42,+16
      IDBUFF(21)= "50516      !Inten,+42,-16
      IDBUFF(22)= "116724     !PtMode,EnbInten:3,EnbLPInt
      IDBUFF(23)= "0          !Move to 0,0
      IDBUFF(24)= "0
      IDBUFF(25)= "117224     !
      IDBUFF(26)= "0
      IDBUFF(27)= "0
```

```
        IDBUFF(28)="110000        !Long Vector Mode, no changes
        IDBUFF(29)=":1000         !Move relative 512,0
        IDBUFF(30)="0
        IDBUFF(31)=":170060        !Load Reg A, Enable Italics
        IDBUFF(32)=":102030        !Char Mode,Intensity 0,Blink
        IDBUFF(33)=":53040         ! V
        IDBUFF(34)=":50124         !TP
        IDBUFF(35)=":47514         !LO
        IDBUFF(36)=":20124         !T
        IDBUFF(37)=":74542         !by
        IDBUFF(38)=":42440         ! E
        IDBUFF(39)=":40513         !KA
        IDBUFF(40)=":102620        !Char Mode, Intensity:3,NoBlink
        IDBUFF(41)=":170040        !Load Register A, Disable Italics
        IDBUFF(42)=":117124        !PointMode,Inten:4,EnblPint,EnbBlinkReg,Line:0
        IDBUFF(43)=":0             !Move to 0,0
        IDBUFF(44)=":0
        IDBUFF(45)=":110000        !Long Vector Mode
        IDBUFF(46)=":0             !Move relative 0,0
        IDBUFF(47)=":0
        IDBUFF(48)=":173400        !Graphic Halt Instruction
        IDBUFF(49)=":0             !!!MUST ALWAYS BE FOLLOWED BY A ZERO!!!!!
        NEXTWD=49
        INIWDS=NEXTWD-1            ! NO. OF WORDS IN HEADER
C
C NOW USE THE LUN FOR THE GRAPHICS DISPLAY
C
        CALL ASNIUN(LUN,'GR',0,IDS)
        IF(IDS .NE. 1) GO TO 902
C
C ESTABLISH ADDRESS AND SIZE OF DISPLAY FILE
C
        CALL GETADR(IPARAM,IDBUFF)
        IPARAM(2)=NWORDS
        IPARAM(3)=1               ! LIGHT PEN EVENT FLAG
```

```
C
C AND START THE DISPLAY PROCESSOR, PASSING IT THE DISPLAY FILE
C
      CALL WTQIO("015400,LUN,2,250,IOSB,IPARAM,IDS)
      IF(IOSB(1).NE.1)GOTO 903
      IF(IDS .NE. 1) GO TO 903
C
C MAKE SURE THE THING GETS SHUT OFF BEFORE EXITING
C
      CALL USEREX(VTEND)
      RETURN
C
C ERROR SECTION
C
902   CALL VTERR(2)           ! COULDN'T ASSIGN LUN TO GR0:
      STOP
903   CALL VTERR(3)           ! COULDN'T FIRE UP DPU
      STOP
      END
```

```fortran
C------------------------------------------
C        SUBROUTINE VTEND
C------------------------------------------
C
C THIS SUBROUTINE STOPS AND DETACHES THE DPU.
C IT SHOULD BE CALLED AFTER ALL GRAPHICS CALLS ARE FINISHED
C BEFORE EXITING TO MONITOR LEVEL.
C
C>
      DIMENSION IOSB(2)
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IHPMAX,IBIT6,IAVVMX,ILVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISEFMX
      IF(IPAUSE.NE.1)GOTO 10
      TYPE *,' Type <CR> to exit.'
      ACCEPT 1000,IB
1000  FORMAT(A2)
C
C ISSUE A QIO TO SHUT THE THING DOWN.
C
10    IF(INDEX.GT.1)CALL DFBDMP
      CALL WTQIO(:016400,LUNGR,,,IOSB,,IDS)
      IF(IDS.EQ.1)GOTO 20
      CALL VTERR(5)      ! ERROR DETACHING SCOPE
      STOP
20    CALL WTQIC("016000,LUNGR,,,IOSB,,IDS)
      IF(IDS.EQ.1)RETURN
      CALL VTERR(5)      ! ERROR DETACHING SCOPE
      STOP
      END
```

```
C----------------------------------------------------
C       SUBROUTINE SETMOD(JMODE)
C----------------------------------------------------
C
C  THIS SUBROUTINE SETS THE GRAPHIC MODE FOR SUBSEQUENT GRAPHICS.
C
C  INPUT ARGS:
C     JMODE....GRAPHICS MODE FOR SUBSEQUENT GRAPHICS:
C              = 1 CHARACTER MODE
C              = 2 SHORT VECTOR
C              = 3 LONG VECTOR
C              = 4 POINT/OFFSET
C              = 5 GRAPHLOT X / BASIC LONG VECTOR
C              = 6 GRAPHLOT Y / BASIC LONG VECTOR
C              = 7 RELATIVE POINT
C              = 8 BASIC SHORT VECTOR
C              = 9 NOT VALID (RESERVED FOR CIRCLE MODE)
C              = 10 ABSOLUTE VECTOR
C              = ANYTHING ELSE...NO CHANGE FROM CURRENT MODE
C
C     NOTE THAT THE MODE ARGUMENT IS ONE GREATER THAN USED
C     BY THE DISPLAY INSTRUCTION 'SET GRAPHIC MODE'.  THIS
C     IS SO THAT THE VALUE OF ZERO COULD BE USED FOR 'NO
C     CHANGE'.
C
C
      COMMON /DFILE/IDBUFF(1)
      COMMON /GREAT/ MAXWF,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IRPMAX,IBITE,IAVVMX,ILVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISBFMX
C
C  DON'T ALLOW MODE SET WHEN MODIFY FLAG IS SET
C
      IF(MODIFY .NE. 0)RETURN
```

```
C  RESET CURRENT MODE IF IN VALID RANGE
C
      IF(JMODE .GT. 0 .AND. JMODE .LE. 10) IMODE=JMODE
      IWORD=ICNTRL .OR. MODE(IMODE)
C
C  FINALLY, PUT THE INSTRUCTICN INTO THE DISPLAY BUFFER
C
      IF(INDEX.GT.ISBFMX)CALL DFBDMP
      IBUF(INDEX)=IWORD
      INDEX=INDEX+1
      RETURN
C
C  ERROR
C
507   CALL VTERR(7)                   ! ILLEGAL DURING MODIFY
L     TYPE *,'ILLEGAL MODIFY IN SETMOD'
D     TYPE *,'MAXWD,NEXTWD,INIWDS,IMODE,MODIFY,MODPTR=',
D    1       MAXWD,NEXTWD,INIWDS,IMODE,MODIFY,MODPTR
      STOP
      END
```

```
C -----------------------------------------------------------------------
C            SUBROUTINE SETPAR(JMODE,JLPEN,JINTEN,JFLASH,JLTYPE,JTALIC)
C -----------------------------------------------------------------------
C
C  THIS SUBROUTINE SETS THE GRAPHIC MODE, LIGHT
C  PEN SENSITIVITY, INTENSITY, FLASH, AND LINE TYPE FOR
C  SUBSEQUENT GRAPHICS.
C
C  INPUT ARGS:
C      JMODE...GRAPHICS MODE FOR SUBSEQUENT GRAPHICS:
C             = 1 CHARACTER MODE
C             = 2 SHORT VECTOR
C             = 3 LONG VECTOR
C             = 4 POINT/OFFSET
C             = 5 GRAPHLOT X / BASIC LONG VECTOR
C             = 6 GRAPHLOT Y / BASIC LONG VECTOR
C             = 7 RELATIVE POINT
C             = 8 BASIC SHORT VECTOR
C             = 9 NOT VALID (RESERVED FOR CIRCLE MODE)
C             = 10 ABSOLUTE VECTOR
C             = ANYTHING ELSE...NO CHANGE FROM CURRENT MODE
C
C             NOTE THAT THE MODE ARGUMENT IS ONE GREATER THAN USED
C             BY THE DISPLAY INSTRUCTION 'SET GRAPHIC MODE'.  THIS
C             IS SO THAT THE VALUE OF ZERO COULD BE USED FOR 'NO
C             CHANGE'.
C
C      JLPEN...LIGHT PEN SENSITIVITY:
C             = POSITIVE...LIGHT PEN INTERACTION ENABLED.
C             = 0 ........NO CHANGE.
C             = NEGATIVE...LIGHT PEN INTERACTION DISABLED.
C
C      JINTEN...INTENSITY:
C             = 1 THROUGH 8.....SET THE INTENSITY LEVEL TO 1-8.
C             = ANYTHING ELSE...NO CHANGE FROM CURRENT INTENSITY.
C
```

```fortran
C  JFLASH...FLASH:
C     = POSITIVE...SUBSEQUENT GRAPHICS WILL FLASH.
C     = 0 .......NO CHANGE.
C     = NEGATIVE...SUBSEQUENT GRAPHICS WILL NOT FLASH.
C
C  JLTYPE...LINE TYPE:
C     = 1 SOLID
C     = 2 LONG DASH
C     = 3 SHORT DASH
C     = 4 DOT-DASH
C     = ANYTHING ELSE....NO CHANGE IN CURRENT LINE TYPE.
C
C  JTALIC...ITALIC MODE:
C     = POSITIVE...SUBSEQUENT TEXT WILL BE ITALICS FONT.
C     = 0 .......NO CHANGE.
C     = NEGATIVE...SUBSEQUENT TEXT WILL BE NORMAL FONT.
C
C>
      COMMON /DFILE/IDBUFF(1)
      COMMON /GREAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IRPMAX,IEIT6,IAVVMX,ILVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISBFMX
      DIMENSION LPEN(2),INTEN(8),IFL(2),ITYPE(4)
      DATA LPEN/"140,"100/
      DATA INTEN/"2000,"2200,"2400,"2600,
     *"3000,"3200,"3400,"3600/
      DATA IFL/30,20/
      DATA ITYPE/4,5,6,7/
      DATA ITALIC/"170260/
      DATA ITALOF/"170240/
```

```
C
C     DON'T ALLOW PARAMETERS TO BE SET IF MODIFY FLAG IS ON
C
      IF(MODIFY .NE. 0)RETURN
C
C     RESET CURRENT MODE IF IN VALID RANGE
C
      IF(JMODE .GT. 0 .AND. JMODE .LE. 10) IMODE=JMODE
      IWORD=ICNTRL .OR. MODE(IMODE)
C
C     LIGHT PEN ENABLE/DISABLE
C
      IF(JLPEN .GT. 0) ILPEN=1          ! LIGHT PEN ENABLE
      IF(JLPEN .LT. 0) ILPEN=2          ! LIGHT PEN DISABLE
      IWORD=IWORD .OR. LPEN(ILPEN)
C
C     INTENSITY
C
      IF(JINTEN.GT.0.AND.JINTEN.LE.8)IINTEN=JINTEN
      IWORD=IWORD .OR. INTEN(IINTEN)
C
C     FLASH ENABLE/DISABLE
C
      IF(JFLASH .GT. 0) IFLASH=1        ! FLASH ENABLE
      IF(JFLASH .LT. 0) IFLASH=2        ! FLASH DISABLE
      IWORD=IWORD .OR. IFL(IFLASH)
C
C     LINE TYPE
C
      IF(JLTYPE .GT. 0 .AND. JLTYPE .LE. 4)
     1      ILTYPE=JLTYPE
      IWORD=IWORD .OR. ITYPE(ILTYPE)
```

```
C
C   FINALLY, PUT THE INSTRUCTION INTO THE DISPLAY BUFFER
C
        IF(INDEX.GT.(ISBFMX-1))CALL DFBDMP
        IBUF(INDEX)=IWORD
        INDEX=INDEX+1
        IF(JTALIC.GT.0)IBUF(INDEX)=ITALIC
        IF(JTALIC.LT.0)IBUF(INDEX)=ITALOF
        INDEX=INDEX+1
        RETURN
C
C   ERROR
C
507     CALL VTERR(7)                    ! ILLEGAL DURING MODIFY
        TYPE *,'ILLEGAL MODIFY IN SETPAR'
        TYPE *,'MAXWD,NEXTWD,INIWDS,IMODE,MODIFY,MODPTR=',
     1        MAXWD,NEXTWD,INIWDS,IMODE,MODIFY,MODPTR
        STOP
        END
```

```
C-----------------------------------------------------------------------
C            SUBROUTINE VTEXT(ICR,NPCHRS,ITEXT)
C-----------------------------------------------------------------------
C
C THIS SUBROUTINE PUTS A TEXT RTRING INTO THE DISPLAY FILE
C
C INPUT ARG:
C    ICR ....Number of carriage returns after the text.   ICR=0 will
C            return the beam to its starting position before the text.
C            ICR=-1024.: No carriage returns.  Beam is left at end of text.
C    NPCHRS..The number of pairs of characters printed.
C    ITEXT...ARRAY OF PACKED ASCII CHARACTERS, ENDING IN AN ASCII NULL
C
C NOTE:
C    ONLY AN EVEN NUMBER OF CHARACTERS CAN BE PUT INTO THE DISPLAY
C    FILE.  IF ITEXT CONTAINS AN ODD NUMBER OF CHARACTERS, THEN
C    THE ASCII NULL IS USED AS THE LAST CHARACTER.
C)
      LOGICAL*1 ITEXT(1),IBLANK,NULL
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IHPMAX,IBITG,IAVVMX,ILVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISBFMX
      DATA IBLANK,NULL /' ',0/
      IXCI=0
      IYCI=0
C
C SET GRAPHICS MODE TO TEXT IF NOT ALREADY THERE
C
      IF(IMODE .NE. 1) CALL SETMOD(1)
```

```fortran
C
C FIND THE NUMBER OF CHARACTERS IN THE TEXT BY SEARCHING FOR A NULL
C
      IXMX=IAVMAX-14
      ISHIFT=0
      DO 10 NCHARS=1,NPCHRS*2
      IF (ITEXT(NCHARS).EQ. NULL) GO TO 20
      IF(ITEXT(NCHARS).GT."177)GOTO 15
      IF(ISHIFT.EQ.0.AND.ITEXT(NCHARS).GE."40)GOTO 9
      IF(ITEXT(NCHARS).EQ."16)IS IFT=1
      IF(ISHIFT.NE.1)GOTO 15
      IF(ITEXT(NCHARS).GT."37)GOTO 15
      IF(ITEXT(NCHARS).EQ."17)IS IFT=0
  5   IXCI=IXCI+ICHWD  ! Increment X pos by Char width
      IXPOS=IXPOS+ICHWD
      IF(IXPCS.GT.IXMX)GOTO 20          ! Limit X pos to stay on screen
 10   CONTINUE
C
C NO NULL FOUND, USE MAX NUMBER OF CHARACTERS
C
      NCHARS=NPCHRS*2
      GOTO 20
C
C PUT THE CHARACTERS INTO THE DISPLAY BUFFER
C
 15   IF((NCHARS.AND.1).EQ.0)ITEXT(NCEARS)=0
 20   IF((NCHARS.AND.1).EQ.0.AND.IS IFT.EQ.1)ITEXT(NCHARS)=" 17
      IF((NCHARS.AND.1).EQ.1.AND.IS IFT.EQ.1)ITEXT(NCHARS-1)=" 17
      IF(INDEX.GT.1)CALL DFBUMP
      CALL DFADD(NCHARS/2,ITEXT)          ! 2 CHARACTERS PER WORD
      CALL RVECTI(-IXCI,(ICR*-ICRHT),0)   ! perform carriage returns
      RETURN
      END
```

```
C-----------------------------------------------------------------
C         SUBROUTINE AVECTI(IX,IY,IVIS)
C-----------------------------------------------------------------
C
C THIS SUBROUTINE DRAWS AN ABSOLUTE VECTOR IN SCREEN COORDINATES.
C
C INPUT ARGS:
C     IX....X POSITION TO MOVE BEAM TO.
C     IY....Y POSITION TO MOVE BEAM TO.
C     IVIS...VISIBILITY
C            = 0 INVISIBLE (DARK) VECTOR
C            = 40000 VISIBLE VECTOR
C            = 1 Visible Vector
C
C NOTES:
C     1.  THE X OR Y COORDINATE MUST NOT EXCEED IAVMAX OR IAVVMX
C RASTER UNITS.  VECTORS ARE REDUCED TO THIS MAXIMUM IF THEY EXCEED IT.
C
C>
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IRPMAX,IPIT6,IAVVMX,ILVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMCN/BUFFER/IBUF(256),INDEX,ISEFMX
```

```
C
C     FORMULATE THE TWO WORD GRAPHIC INSTRUCTION, LIMITING
C     POSITION TO IAVMAX RASTER UNITS.
C
         IF(IX.GT.IAVMAX)IX=IAVMAX
         IF(IX.LT.0)IX=0
         IF(IY.GT.IAVVMX)IY=IAVVMX
         IF(IY.LT.0)IY=0
         IDX=IX-IXPOS
         IDY=IY-IYPOS
         IXDAT=IABS(IDX)
         IYDAT=IABS(IDY)
         MOD=3
         IF(IXDAT.LT.IRPMAX.AND.IYDAT.LT.IRPMAX)MOD=2
C
C     SET THE GRAPHICS MODE TO CORRECT MODE IF IT'S NOT ALREADY
C
         IF(IMODE .NE. MOD) CALL SETMOD(MOD)
C
C     UPDATE THE BEAM POSITION IN COMMON
C
         IXPOS=IX
         IYPOS=IY
C
C     Adjust VISIBILITY
C
         IF(IVIS.EQ.1)IVIS=INTENA
         IF(IVIS.NE.INTDIS.AND.IVIS.NE.INTENA)IVIS=INTDIS
```

```
C
C   Choose long or short vector mode data word format
C
      IF(MOD.EQ.3)GOTO 100
      IDAT=IYDAT .OR. ISHFT(IXDAT,7)
      IF(IDX .LT. 0) IDAT=IDAT .OR. IMINUS
      IF(IDY .LT. 0) ICAT=IDAT .OR. IBITE
      IF(INDEX.GT.ISBFMX)CALL DFEDMP
      IBUF(INDEX)=IDAT.OR.IVIS
      INDEX=INDEX+1
      RETURN
C
C   DO IT RIGHT IF DATA IS NEGATIVE
C
  100 IF(IDX.LT.0)IXDAT=IXDAT.OR.IMINUS
      IF(IDY.LT.0)IYDAT=IYDAT.OR.IMINUS
      IF(INDEX.GT.(ISBFMX-1))CALL DFEDMP
      IBUF(INDEX)=IXDAT.OR.IVIS
      INDEX=INDEX+1
      IBUF(INDEX)=IYDAT
      INDEX=INDEX+1
      RETURN
      END
```

```
C------------------------------------------------
C          SUBROUTINE RVECTI(IDX,IDY,IVIS)
C------------------------------------------------
C
C THIS SUBROUTINE DRAWS A RELATIVE VECTOR IN SCREEN COORDINATES.
C
C INPUT ARGS:
C     IDX...RELATIVE DISPLACEMENT IN THE X DIRECTION.
C     IDY...RELATIVE DISPLACEMENT IN THE Y DIRECTION.
C     IVIS...VISIBILITY:
C           = 0 INVISIBLE (DARK) VECTOR
C           = 40000 VISIBLE VECTOR
C           = 1 Visible vector
C
C NOTES:
C   1.  THE DISPLACEMENT MUST NOT EXCEED ILVMAX RASTER UNITS
C       IN EITHER THE X OR Y DIRECTION.  VECTORS ARE REDUCED TO
C       THIS MAXIMUM IF THEY EXCEED IT.
C
C>
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IIHALT,ILVMAX,
     *IAVMAX,IRPMAX,IBIT6,IAVVMX,ILVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISBFMX
```

```
C
C   FORMULATE THE TWO WORD GRAPHIC INSTRUCTION, LIMITING
C   DISPLACEMENT TO IIVMAX RASTER UNITS.
C
        IXP=IXPOS+IDX
        IYP=IYPOS+IDY
        IF(IXP.GT.IAVMAX)IDX=IAVMAX-IXPOS
        IF(IXP.LT.0)IDX=0-IXPOS
        IF(IYP.GT.IAVVMX)IDY=IAVVMX-IYPOS
        IF(IYP.LT.0)IDY=0-IYPOS
        IXDAT=IABS(IDX)
        IYLAT=IABS(IDY)
        MOD=3
        IF(IXDAT.LT.IRPMAX.AND.IYDAT.LT.IRPMAX)MOD=2
        IF(IMODE .NE. MOD) CALL SETMOD(MOD)
C
C   UPDATE THE BEAM POSITION IN COMMON
C
        IXPOS=IXPOS+IDX
        IYPOS=IYPOS+IDY
C
C   STUFF THE DATA INTO THE DISPLAY FILE WITH APPROPRIATE VISIBILITY
C
        IF(IVIS.EQ.1)IVIS=INTENA
        IF(IVIS.NE.INTDIS.AND.IVIS.NE.INTENA)IVIS=INTDIS
        IF(MOD.EQ.3)GOTO 100
        IDAT=IYDAT .OR. ISHFT(IXDAT,7)
        IF(IDX .LT. 0) IDAT=IDAT .OR. IMINUS
        IF(IDY .LT. 0) IDAT=IDAT .OR. IBIT6
        IF(INDEX.GT.ISBFMX)CALL DFBDMP
        IBUF(INDEX)=IDAT.OR.IVIS
        INDEX=INDEX+1
        RETURN
```

```
C
C  DO IT RIGHT IF DATA IS NEGATIVE
C
100     IF(IDX .LT. 0) IXDAT=IXDAT .OR. IMINUS
        IF(IDY .LT. 0) IYDAT=IYDAT .OR. IMINUS
        IF(INDEX.GT.(ISBFMX-1))CALL DFBDMP
        IBUF(INDEX)=IXDAT.OR.IVIS
        INDEX=INDEX+1
        IBUF(INDEX)=IYDAT
        INDEX=INDEX+1
        RETURN
        END
```

```
C --------------------------------------------------------
C        SUBROUTINE APNTI(IX,IY,IVIS)
C --------------------------------------------------------
C
C THIS SUBROUTINE PUTS AN ABSOLUTE POINT AT SCREEN COORDINATES.
C
C INPUT ARGS:
C     IX.....X POSITION TO MOVE BEAM TO.
C     IY.....Y POSITION TO MOVE BEAM TO.
C     IVIS...VISIBILITY
C            = 0 INVISIBLE (DARK) POINT
C            = 40000 VISIBLE POINT
C            = 1 Visible point
C
C NOTES:
C     1.   THE X OR Y COORDINATE MUST NOT EXCEED IAVMAX OR IAVVMX
C RASTER UNITS.  VECTORS ARE REDUCED TO THIS MAXIMUM IF THEY EXCEED IT.
C
C>
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,INTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IRPMAX,IBIT6,IAVVMX,IIVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISEFMX
```

```
C
C   FORMULATE THE TWO WORD GRAPHIC INSTRUCTION, LIMITING
C   POSITION TO IAVMAX RASTER UNITS.
C
      IF(IX.GT.IAVMAX)IX=IAVMAX
      IF(IX.LT.0)IX=0
      IF(IY.GT.IAVVMX)IY=IAVVMX
      IF(IY.LT.0)IY=0
      IDX=IX-IXPCS
      IDY=IY-IYPOS
      IXDAT=IABS(IDX)
      IYDAT=IABS(IDY)
      MOD=4
      IF(IXDAT.LT.IRPMAX.AND.IYDAT.LT.IRPMAX)MOD=7
C
C   SET THE GRAPHICS MODE TO CORRECT MODE IF IT'S NOT ALREADY
C
      IF(IMODE .NE. MOD) CALL SETMOD(MOD)
C
C   UPDATE THE BEAM POSITION IN COMMON
C
      IXPOS=IX
      IYPOS=IY
C
C   Adjust VISIBILITY
C
      IF(IVIS.EQ.1)IVIS=INTENA
      IF(IVIS.NE.INTDIS.AND.IVIS.NE.INTENA)IVIS=INTDIS
      IF(MOD.EQ.4)GOTO 100
      IDAT=IYDAT .OR. ISHFT(IXDAT,7)
      IF(IDX .LT. 0) IDAT=IDAT .OR. IMINUS
      IF(IDY .LT. 0) IDAT=IDAT .OR. IBIT6
      IF(INDEX.GT.ISBFMX)CALL DFBEMP
      IBUF(INDEX)=IDAT.OR.IVIS
      INDEX=INDEX+1
      RETURN
```

```
100    IF(INDEX.GT.(ISEFMX-1))CALL DEBDMP
       IBUF(INDEX)=IX.OR.IVIS
       INDEX=INDEX+1
       IBUF(INDEX)=IY
       INDEX=INDEX+1
       RETURN
       END
```

```
C-------------------------------------------------
C        SUBROUTINE RPNTI(IDX,IDY,IVIS)
C-------------------------------------------------
C
C THIS SUBROUTINE PUTS A RELATIVE POINT IN SCREEN COORDINATES.
C
C INPUT ARGS:
C    IDX...RELATIVE DISPLACEMENT IN THE X DIRECTION.
C    IDY...RELATIVE DISPLACEMENT IN THE Y DIRECTION.
C    IVIS...VISIBILITY:
C            = 0 INVISIBLE (DARK) POINT
C            = 40000 VISIBLE POINT
C            = 1 Visible point
C
C NOTES:
C    1.   THE DISPLACEMENT MUST NOT EXCEED ILVMAX RASTER UNITS
C         IN EITHER THE X OR Y DIRECTION.   VECTORS ARE REDUCED TO
C         THIS MAXIMUM IF THEY EXCEED IT.
C
C
       COMMON /PFILE/IDBUFF(1)
       COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
      *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
      *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
      *IAVMAX,IRPMAX,IBIT6,IAVVMX,IIVVMX,ICPRD,ICHHT,MAXCHR,IPAUSE
       COMMON/BUFFER/IBUF(256),INDEX,ISFFMX
C>
```

```
C
C     FORMULATE THE TWO WORD GRAPHIC INSTRUCTION, LIMITING
C     DISPLACEMENT TO ILVMAX RASTER UNITS.
C
      IXP=IXPOS+IDX
      IYP=IYPOS+IDY
      IF(IXP.GT.IAVMAX)IDX=IAVMAX-IXPOS
      IF(IXP.LT.0)IDX=0-IXPOS
      IF(IYP.GT.IAVVMX)IDY=IAVVMX-IYPOS
      IF(IYP.LT.0)IDY=0-IYPOS
      IXDAT=IABS(IDX)
      IYDAT=IABS(IDY)
      MOD=4
      IF(IXDAT.LT.IRPMAX.AND.IYDAT.LT.IRPMAX)MOD=7
      IF(IMODE.NE.MOD) CALL SETMOD(MOD)
C
C     UPDATE THE BEAM POSITION IN COMMON
C
      IXPOS=IXPOS+IDX
      IYPOS=IYPOS+IDY
C
C     STUFF THE DATA INTO THE DISPLAY FILE WITH APPROPRIATE VISIBILITY
C
      IF(IVIS.EQ.1)IVIS=INTENA
      IF(IVIS.NE.INTDIS.AND.IVIS.NE.INTENA)IVIS=INTDIS
      IF(MOD.EQ.4)GOTO 100
      IDAT=IYDAT .OR. ISHFT(IXDAT,7)
      IF(IDX .LT. 0) IDAT=IDAT .OR. IMINUS
      IF(IDY .LT. 0) IDAT=IDAT .OR. IBIT6
      IF(INDEX.GT.ISBFMX)CALL DFBFMF
      IBUF(INDEX)=IDAT.OR.IVIS
      INDEX=INDEX+1
      RETURN
```

```
100     IX=IXPOS
        IY=IYPOS
        IF(INDEX.GT.(ISBFMX-1))CALL DFBDMP
        IBUF(INDEX)=IS.OR.IVIS
        INDEX=INDEX+1
        IBUF(INDEX)=IY
        INDEX=INDEX+1
        RETURN
        END
```

```fortran
C-------------------------------------
C        SUBROUTINE DFADD1(IWORD)
C-------------------------------------
C
C THIS SUBROUTINE ADDS A SINGLE WORD TO THE END OF THE DISPLAY FILE.
C
C INPUT ARG:
C   IWORD...DATA TO BE ADDED TO THE DISPLAY FILE.
C
C NOTES:
C    1. THE 'MODIFY' FLAG IS CHECKED BEFORE ADDING TO THE DISPLAY FILE.
C         IF SET, DATA IS INSERTED INTO THE DISPLAY FILE AT THE
C         MODIFY POINTER ADDRESS INSTEAD OF BEING ADDED TO THE END.
C    2. A CHECK IS MADE TO SEE THAT THERE IS ENOUGH ROOM
C         IN THE DISPLAY FILE.  IF NOT, AN ERROR IS PRINTED
C         AND NO ADDITION IS MADE.
C
C>
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,INTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IRPMAX,IBITG,IAVVMX,ILVVMX,ICRWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISFPMX
      DIMENSION IOSE(2),IPARAM(4)
C
C IS THE MODIFY FLAG SET?
C
      IF(MODIFY .EQ. 0) GO TO 10
C
C YES, MODIFY THE DISPLAY FILE INSTEAD OF ADDING
C
      CALL DFPUT1(MODPTR,IWORD)
      MODIFY=0
      GO TO 99
```

```
C
C     SEE IF ENOUGH ROOM IN DISPLAY FILE
C
      10    IF(INDEX.GT.1)CALL DFEDMP
            IF(NEXTWD+1 .GE. MAXWD) GO TO 904
C
C     YES, SO PUT THE STUFF IN
C
            CALL WTQIO("016400,IUNGR,2,250,IOSB,IPARAM,IDS)
            IF(IDS.NE.1)GOTO 903
            IF(ICSB(1).NE.1)GOTO 903
            IDBUFF(NEXTWD+1)=0                  ! DHALT MUST BE FOLLOWED BY A ZERO
            IDBUFF(NEXTWD)=IDHALT               ! NEW DHALT
            IDBUFF(NEXTWD-1)=IWORD              ! GRAPHIC DATA WORD
            CALL QIO("017000,IUNGR,2,250,IOSB,IPARAM,IDS)
            IF(IDS.NE.1)GOTO 905
            IF(ICSB(1).NE.1)GOTO 905
            NEXTWD=NEXTWD+1
            RETURN
C
      903   CALL VTERR(20)              ! DISPLAY HALT FOR BUFFER INSERTION ERROR
            STOP
      904   CALL VTERR(4)               ! DISPLAY FILE FULL
            STOP
      905   CALL VTERR(21)              ! DISPLAY CONTINUE ERROR
            STOP
            END
```

```fortran
C------------------------------------
C     SUBROUTINE DFADD2(IWORD1,IWORD2)
C------------------------------------
C
C THIS SUBROUTINE ADDS TWO WORDS TO THE DISPLAY FILE.
C
C INPUT ARGS:
C     IWORD1...FIRST WORD TO BE ADDED
C     IWORD2...SECOND WORD TO BE ADDED
C
C NOTES:
C     1. THE 'MODIFY' FIRST CHECKED. IF SET, THE DATA IS PUT
C          INTO THE DISPLAY FILE AT THE MODIFY POINTER ADDRESS
C          INSTEAD OF BEING ADDED TO THE END.
C     2. A CHECK IS MADE TO SEE THAT THERE IS ENOUGH ROOM
C          IN THE DISPLAY FILE.  IF NOT, AN ERROR IS PRINTED
C          AND NO ADDITION IS MADE.
C>
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IRPMAX,IBIT6,IAVVMX,ILVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISBFMY
      DIMENSION IOSB(2),IPARAM(4)
C
C IS THE MODIFY FLAG SET?
C
      IF(MODIFY .EQ. 0) GO TO 10
C
C YES, IT'S SET, SO MODIFY THE DISPLAY FILE INSTEAD OF ADDING
C
      CALL DFPUT2(MODPTR,IWORD1,IWORD2)
      MODIFY=0
      GO TO 99
```

```
C  SEE IF ENOUGH ROOM IN DISPLAY FILE
C
   10    IF(INDEX.GT.1)CALL EFBDMP
         IF(NEXTWD+2 .GE. MAXWD) GO TO 904
C
C  YES, SO PUT THE STUFF IN
C
         CALL WTQIO("016400,IUNGR,2,250,IOSB,IPARAM,IDS)
         IF(IDS.NE.1)GOTO 903
         IF(IOSB(1).NE.1)GOTO 903
         IDBUFF(NEXTWD+2)=0              ! DHALT MUST BE FOLLOWED BY A ZERO
         IDBUFF(NEXTWD+1)=ILHALT         ! NEW DHALT
         IDBUFF(NEXTWD)  =IWORD2         ! 2ND WORD OF DATA
         IDBUFF(NEXTWD-1)=IWORD1         ! 1ST WORD OF DATA
         CALL QIO("017000,LUNGR,2,250,IOSB,IPARAM,IDS)
         IF(IDS.NE.1)GOTO 905
         IF(IOSB(1).NE.1)GOTO 905
         NEXTWD=NEXTWD+2
         RETURN
   55
C
C  ERROR, NOT ENOUGH ROOM
C
  903    CALL VTERR(20)           ! DISPLAY HALT FOR BUFFER INSERTION ERROR
         STOP
  904    CALL VTERR(4)            ! DISPLAY FILE FULL
         STOP
  905    CALL VTERR(21)           ! DISPLAY CONTINUE ERROR
         STOP
         END
```

```
C-------------------------------------------------------
C         SUBROUTINE DFADD(NWORDS,IARRAY)
C-------------------------------------------------------
C
C THIS SUBROUTINE ADDS DATA TO THE END OF THE DISPLAY FILE.
C
C INPUT ARGS:
C    NWORDS...NUMBER OF WORDS TO BE ADDED
C    IARRAY...INTEGER ARRAY OF DATA TO BE ADDED.
C
C NOTES:
C
C    SEE NOTES FOR SUBROUTINE DFALD1
C
C>
      DIMENSION IARRAY(NWORDS)
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IHPMAX,IBIT6,IAVVMX,ILVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISEFMX
      DIMENSION IOSB(2),IPARAM(4)
C
C IS THE MODIFY FLAG SET?
C
      IF(MODIFY .EQ. 0) GO TO 10
C
C YES, MODIFY THE DISPLAY FILE INSTEAD OF ADDING
C
      CALL DFPUT(MODPTR,NWORDS,IARRAY)
      MODIFY=0
      GO TO 99
C
C SEE IF ENOUGH ROOM IN THE DISPLAY FILE
C
10    NXTNEW=NEXTWD+NWORDS
      IF(NXTNEW .GE. MAXWD) GO TO 504
```

```fortran
C YES, ENOUGH ROOM, PUT THE STUFF IN
C
        CALL WTQIO("016400,LUNGR,2,250,IOSB,IPARAM,IDS)
        IF(IDS.NE.1)GOTO 903
        IF(IOSB(1).NE.1)GOTO 903
        IDBUFF(NXTNEW)=0             ! DHALT MUST BE FOLLOWED BY A ZERO
        IDBUFF(NXTNEW-1)=IDHALT      ! NEW DHALT
        IWORD=NEXTWD-2
        DO 20 J=NWORDS,1,-1
20      IDBUFF(IWORD+J)=IARRAY(J)    ! DATA WORD
        CALL QIO("017000,LUNGR,2,250,IOSB,IPARAM,IDS)
        IF(IDS.NE.1)GOTO 905
        IF(IOSB(1).NE.1)GOTO 905
        NEXTWD=NXTNEW               ! NEW 'NEXT WORD' POINTER
55      RETURN
C
C ERROR, DISPLAY FILE FULL
C
903     CALL VTERR(20)             ! DISPLAY HALT FOR BUFFER INSERTION ERROR
        STOP
904     CALL VTERR(4)              ! DISPLAY FILE FULL
        STOP
905     CALL VTERR(21)             ! DISPLAY CONTINUE ERROR
        STOP
        END
```

```
C---------------------------------
C        SUBROUTINE DFBDMP
C---------------------------------
C
C  THIS SUBROUTINE ADDS A SUB-BUFFER OF DATA TO THE END OF THE DISPLAY FILE.
C
C
C  NOTES:
C     SEE NOTES FOR SUBROUTINE DFADD1
C
C>
      COMMON /DFILE/IDBUFF(1)
      COMMON /GRDAT/ MAXWD,NEXTWD,INIWDS,IXPOS,IYPOS,ILPEN,IINTEN,
     *IFLASH,ILTYPE,IMODE,LUNGR,MODIFY,MODPTR,
     *INTENA,INTDIS,IMINUS,ICNTRL,MODE(10),IDHALT,ILVMAX,
     *IAVMAX,IRPMAX,IEIT6,IAVVMX,ILVVMX,ILVVMX,ICHWD,ICHHT,MAXCHR,IPAUSE
      COMMON/BUFFER/IBUF(256),INDEX,ISBFMX
      IF(INDEX.LE.1)GOTO 10
      CALL DFAED(INDEX-1,IBUF)
10    INDEX=1
      RETURN
      END
```

```
C---------------------------------------
      SUBROUTINE VTERR(IER)
C---------------------------------------
C
C
C>
      IF(IER.EQ.2)TYPE *,'    *** VTLIB COULDNT ASSIGN LUN TO GR0: '
      IF(IER.EQ.3)TYPE *,'    *** VTLIB COULDNT START DPU ***'
      IF(IER.EQ.4)TYPE *,'    *** VTLIB DISPLAY BUFFER FULL ***'
      IF(IER.EQ.5)TYPE *,'    *** VTLIB ERROR DETACHING GR0: ***'
      IF(IER.EQ.7)TYPE *,'    *** VTLIB ILLEGAL DISPBUFFER MODIFY ***'
      IF(IER.EQ.20)TYPE *,'   *** VTLIB BUFFER INSERTION HALT ERROR'
      IF(IER.EQ.21)TYPE *,'   *** VTLIB DISPLAY CONTINUE ERROR ***'
      RETURN
      END
```

```fortran
C   VTPLOT -- RECIEVER TASK FOR NETWORK PLOT COMMANDS
C   VTPLOT
C
C       16-FEB-81        by Erik Antonsson
C
        DIMENSION IDAT(256,2)
        INTEGER*2 IOST(2,2),MSTAT(3)
        BYTE CONBLK(72),MLBX(98)
        INTEGER STAT
        CALL OPNNTW(3,IOST,MSTAT)
        CALL GNDNTW(IOST,MLTYP,98,MLBX)
        IF(MLTYP.NE.1)GOTO 5
        CALL ACCNTW(2,IOST,MLBX)
        IBUF=2
        IBUFR=1
        IBYTE=512
        CALL RECNT(2,IOST(1,IBUFR),IBYTE,IDAT(1,IBUFR))
5       CALL WAITNT(,IOST(1,IBUFR))
        IBUF=IBUF+1
        IF(IBUF.GT.2)IBUF=1
        IBUFR=IBUFR+1
        IF(IBUFR.GT.2)IBUFR=1
        CALL RECNT(2,IOST(1,IBUFR),IBYTE,IDAT(1,IBUFR))
        INDX=1
7       IF(IDAT(INDX,IBUF).LE.-30000)GOTO 9
        IF(INDX.GE.IBYTE/2)GOTO 5
        INDX=INDX+1
        GOTO 7
9       GOTO (10,20,30,40,50,60,70,80),IABS(IDAT(INDX,IBUF)+30000)
        GOTO 200
10      CALL VTINIT(4,12000,0)
        INDX=INDX+1
        GOTO 7
```

```
20    CALL SETPAR(IDAT(INDX+1,IBUF),IDAT(INDX+2,IBUF),
     *IDAT(INDX+3,IBUF),IDAT(INDX+4,IBUF),
     *IDAT(INDX+5,IBUF),IDAT(INDX+6,IBUF))
      INDX=INDX+7
      GOTO 7
30    CALL VTEXTI(IDAT(INDX+1,IBUF),IDAT(INDX+2,IBUF),IDAT(INDX+3,IBUF))
      INDX=INDX+3
      GOTO 7
40    CALL AVECTI(IDAT(INDX+1,IBUF),IDAT(INDX+2,IBUF),
     *IDAT(INDX+3,IBUF))
      INDX=INDX+4
      GOTO 7
50    CALL RVECTI(IDAT(INDX+1,IBUF),IDAT(INDX+2,IBUF),
     *IDAT(INDX+3,IBUF))
      INDX=INDX+4
      GOTO 7
60    CALL APNTI(IDAT(INDX+1,IBUF),IDAT(INDX+2,IBUF),
     *IDAT(INDX+3,IBUF))
      INDX=INDX+4
      GOTO 7
70    CALL RPNTI(IDAT(INDX+1,IBUF),IDAT(INDX+2,IBUF),
     *IDAT(INDX+3,IBUF))
      INDX=INDX+4
      GOTO 7
80    CALL DFEDMP
      GOTO 5
200   CALL CLSNTW
      CALL EXIT
      END
```

```
DFILE.MAC
DFILE.MAC

by Erik K. Antonsson     15-APR-81
Modified by Erik Antonsson     16-AUG-81

>MAC DFILE=DFILE
>TKB
TKB>LB:[1,1]DFILE/-HD/PI,LB:[1,1]DFILE,LB:[1,1]DFILE=DFILE
TKB>/
ENTER OPTIONS:
TKB>STACK=0
TKB>PAR=DFILE:%:100000
TKB>//

IDBUFF:: .PSECT  DFILE,RW,D,GBL,REL,OVR
         .BLKW  16000.        ;160000. words, 100000(8) bytes
         .END
```

APPENDIX J

OTHER REFERENCES, SORTED

These are other references that I thought would be interesting, informative, and relevant, but I was unable to obtain or locate. They are sorted chronologically by journal.

JOURNAL OF BIOMECHANICS

1. Becket R. and Chang, K., "An Evaluation of the Kinematics of Gait by Minimum Energy," Journal of Biomechanics, 1:147-159, 1968.

2. Harris, E. H., Haslum, E. T., and Moffatt, C. A., "An Experimental and Analytic study of the Dynamic Properties of the Human Leg," Journal of Biomechanics, vol 2, p373, 1969.

3. Chaffin, D. B., "A Computerized Model - Development of and Use in Studying Gross Body Actions," Journal of Biomechanics, vol 2, pp 429, 1969.

4. Morrison, J. B., "The Mechanics of the Knee Joint in Relation to Normal Walking," Journal of Biomechanics, vol. 3, pp 51, 1970.

5. Chao, E. Y., Rim, K., Smidt, g. L., and Johnston, R. C., "The Application of a 4x4 Matrix Method to the Correction of the Measurements of Hip Joint Rotation," Journal of Biomechanics, 3, 459-471, 1970.

6. Jacobs, Shorecki, and Charnley, "Analysis of the Vertical Component of Force in Normal and Pathological Gait," Journal of Biomechanics, vol 5, pp 11, 1972.

7. Smidt, G. L., "A Biomechanical Analysis of Knee Flexion and Extension," Journal of Biomechanics, vol 6, pp79.

8. Waters, R. L., Morris, J., and Perry, J., "Translational Motion of the Head and Trunk During Normal Walking," Journal of Biomechanics, 6, 2, March 1973, p 167.

9. Morris, J. R. W., "Accelerometry - A Technique for the Measurement of Human Body Movement," Journal of Biomechanics, 6,729-736, 1973.

10. Winter, D. A., Sidwal, and Hobson, D. A., "Measurement and Reduction of Noise in Kinematics and Locomotion," Journal of Biomechanics, 1974, vol 7, 2.

11. Panjabi, M. M., White, and Brand, "A Note on Defining Body Parts Configurations," Journal of Biomechanics, 7(4):385-7, Aug. 1974.

12. Blacharski, P., Somerset, J., and Murray, D., "A 3-D study of the Kinematics of the Human Knee," Journal of Biomechanics, vol 8, pp 375.

13. Cappozzo, A., Figura, Marchetti, Pedotti, A., "The Interplay of Muscular and External Forces in Human Ambulation," Journal of Biomechanics, vol 9, pp 35.

14. Yamashita, T., and Kutoh, R., "Moving Pattern of Point of Application of Vertical Resultant Force During Level Walking," Journal of Biomechanics, vol 9, pp 93.

15. Hatze, H., "A Method for Describing the Motion of Biological Systems," Journal of Biomechanics, vol 9, pp 101.

16. Shoup, T. E., "Optical Measurement of the Center of Rotation for Human Joints," Journal of Biomechanics, vol. 9, pp 241.

17. Muksian, R., and Nash, C. D., "On Frequency Dependent Damping Coefficients in Lumped Parameter Models of Human Beings," Journal of Biomechanics, vol 9, pp 339.

18. Yeo, B. P., "Investigations Concerning the Principle of Minimal Total Muscular Force," Journal of Biomechanics, vol 9, pp 413.

19. Arcan, M., and Brull, M. A., "A Fundamental Characterisitc of the Human Body and Foot, The Foot-Ground Pressure Pattern," Journal of Biomechanics, vol 9, pp 453.

20. Andriachhi, T. P., Ogle, J. A., and Galante, G. O., "Walking Speed as a Basis for Normal and Abnormal Gait Measurements," Journal of Biomechanics, 10, 261,1977.

21. Lewis, J. L., and Lew, W. D., (1977) "A Note on the Description of Articulating Joint Motion," Journal of Biomechanics, 10, 675-678.

22. Hatze, H., "A Complete Set of Control Equations for the Human Musculo-Skeletal System," Journal of Biomechanics, 10, 799-805, 1977.

JOURNAL OF BONE AND JOINT SURGERY

1. Schwartz, R. P., and Heath, A. C., "The Pneumographic Method of Recording Gait," Journal of Bone and Joint Surgery, 14:783, 1932.

2. Schwartz, R. P., Heath, A. L., Misiek, W., and Wright, J. N., "Kinetics of Human Gait," Journal of Bone and Joint Surgery, 16:343-350, 1934.

3. Levens, A. S., Inman, V. T., and Blosser, J. A., "Transverse Rotation of the Segments of the Lower Extremity in Locomotion," J. Bone and Joint Surgery, 30-A:859-872, Oct. 1948.

4. Steindler, A., "A Historical Review of the Studies and Investigations Made in Relation to Human Gait," Journal of Bone and Joint Surgery, 35-A, 542, 1953.

5. Saunders, J. B. deC. M., Inman, V. T., and Eberhart, H. D., "The Major Determinants in Normal and Pathological Gait," Journal of Bone and Joint Surgery, 35-A:543-558, July 1953.

6. Murray, M. P., Dought, A. B., and Kory, R. C., "Walking Patterns of Normal Men," Journal of Bone and Joint Surgery, 46-A:335-360, Mar. 1964.

7. Wright, D. G., Desai, S. M., and Henderson, W. H., "Action of the Subtalar and Ankle-Joint Complex During the Stance Phase of Walking," Journal of Bone and Joint Surgery, 46-A:361-383, Mar. 1964.

8. Stott, J. R. R., Hutton, W. C., and Stokes, I. A. F., (1973) "Forces Under the Foot," Journal of Bone and Joint Surgery, 55B, 335-344.

9. Johnston, R. C., et al., "Reconstruction of the Hip. A Mathematical Approach to Determine Optimum Geometric Relationships," Journal of Bone and Joint Surgery, 61A(S):639-652, July 1979.

10. Simon, S. R., Munro, et al., "Peak Dynamic Force in Human Gait and its Attenuation by the Soft Tissues," Journal of Bone and Joint Surgery, (in press?).

TRANSACTIONS OF THE ORTHOPAEDIC RESEARCH SOCIETY (ORS)

1. Andrews, J. G., Chao, E. Y., Johnston, R. C., and Stauffer, R. N., "A General Method for Accurate Kinematic Analysis of Biomechanical Systems in Motion," Journal of Bone and Joint Surgery, 54-A, 1118, 1972, Presented at: ORS Jan 27-29, 1972, Washington DC.

2. Tandon, S. and Marsalais, E. B., "Three-Dimensional Stroboscope Gait Analysis (The Use of the Selspot System)," in Proc. 21st Annu. Orthop. Res. Soc., Februrary, 1975.

3. Crowninshield, R. D., Johnston, R. C., Andrews, J. G., and Brand, R. A., "The Effect of Joint Center Location on Hip Kinetics," Trans. 23rd Ann. Meeting Orthopaedic Res. Soc., (ORS) 2, 49, 1977.

AMERICAN SOCIETY OF MECHANICAL ENGINEERS (ASME)

1. Bartz, J. A., and Gianotti, C. P., "Computer Programme to Generate Dimensional and Inertial Properties of the Human Body," ASME Paper: 73-WA/BIO-3.

2. Abelnour, T. A., Passerello, C. E., and Huston, R. L., "Analytical Analysis of Walking," ASME Paper 75-WA/BIO-4.

3. Chao, E. Y., "Functional Evaluation of Total Knee Replacement Patients Through Gait Analysis," ASME Paper 75-APMB-5, June 1975.

4. Suntay, W. J.,Grood, E. S., Noyes, F. R., and Butler, D. L., (1978) "A Coordinate System for Describing Joint Position," in: Advances in Bioengineering, Eberhart, R. C., and Burnstein, A. H. eds., ASME, New York.

5. Uicker, J. J., Denavit, T., and Hartenberg, R. S., "An Iterative Method for the Displacement Analysis of Spatial Mechanisms," Journal of Applied Mechanics, vol 31, Trans ASME, vol 86, Series E, 1964, p309-314.

6. Chase, M? A?, Bayazitoglu, Y. O., "Development and Application of a Generalized d'Alembert Force for Multifreedom Mechanical Systems," J. of Engineering for Industry, 1971, 93, 317-327.

BULLETIN OF PROSTHETICS RESEARCH (BPR)

1. Peizer, E., Wright, D. W., and Mason, C., "Human Locomotion," Bull. Prosthetics, Res., BPR 10-12:48-105, Fall 1969.

2. Perry, J., "Clinical Gait Analyzer," Bull. Prosthet. Res., 188-192, Fall 1974.

ANNALS OF THE NEW YORK ACADEMY OF SCIENCE (ANN. N.Y. ACAD. SCI.)

1. Eberhart, H. D., and Inman, V. T., "An Evaluation of Experimental Procedures Used in a Fundamental Study of Human Locomotion," Ann. N.Y. Acad. Sci., 51:1213-1228, Jan. 1951. (1967?)

2. Elftman, H., "Body Dynamics and Dynamic Anthropometry," Ann N.Y. Acad. Sci., 63:553-558, Nov. 1955.

3. Drillis, R., "Objective Recording and Biomechanics of Pathological Gait," Ann. N.Y. Acad. Sci., 17:86-109, 1958.

BIOMEDICAL ENGINEERING (BIO-MED. ENG.)

1. Elftman, H., "Basic Function of the Lower Limb," <u>Bio-med. Enging.</u>, 2:342-345, Aug. 1967.

2. Grieve, D. W., "Gait Patterns and the Speed of Walking," <u>Bio-med. Enging.</u>, 3:119-122, Mar. 1968.

3. Morrison, J. B., "Bioengineering Analysis of Force Actions Transmitted by the Knee Joint," <u>Bio-med Eng.</u>, 3(4), 164-170, 1968.

4. Stokes, I. A., Stott, and Hutton, "Force Distributins Under the Foot -- A Dynamic Measuring System," <u>Biomed. Eng.</u>, 9(4):140-143, April, 1974.

5. Harrington, I. J., "A Bioengineering Analysis of Force Actions at the Knee in Normal and Pathological Gait," <u>Biomed. Eng.</u>, 11(5):167-172, May 1976.

IEEE

1. Gubina, F., Hemami, H., and McGhee, R. B., "On the Stability of Biped Locomotion," <u>IEEE trans on BME</u>, vol BME-21, 2, March 1974, pp 102-108.

2. Cheng, I., Koozekanani, S. H., and fatehi, M. T., "A Simple Computer Television Interface System for Gait Analysis," <u>IEEE Transactions on BME</u>, BME 22, 3, 1975, p 259.

MATHEMATICAL BIOSCIENCES

1. Chow, C. K., and Jacobson, D. H., "Studies of Human Locomotion via Optimal Programming," _Mathematical Biosciences_, vol. 10, pp. 239-306, 1971. (QA/.M4243)

2. Chow, C. K., and Jacobson, D. H., "Further Studies of Human Locomotion; Postural Stability and Control," _Mathematical Biosciences_, vol. 15, pp. 93-108, 1972. (QA/.M4243)

EXERCISE AND SPORT SCIENCE REVIEW (EXERC. SPORT SCI. REV.)

1. Dillman, C. J., "Kinematic Analysis of Running," _Exerc. Sport Sci. Rev._, 3:193-218, 1975.

2. Ramey, M. R., "Force Plate Designs and Applications," _Exerc. Sport Sci. Rev._, 3:303-319, 1975.

AMERICAN JOURNAL OF PHYSIOLOGY (AM. J. PHYSIOL.)

1. Elftman, H., "Forces and Energy Changes in the Leg During Walking," _Amer. J. Physiol._, 125:339-356, Feb. 1939.

2. Elftman, H., "The Function of the Muscles in Locomotion," _Amer. J. Physiol._, 125:357-366, Feb. 1939.

JOURNAL OF PHYSIOLOGY

1. Light, L. H., and McLellan, G. E., (1977), "Skeletal Transients Associated with Heel Strike," J. Physiol., 272, 9-10P.

2. Light, L. H., (1979), "Potential Implications of Heel Strike Transients," J. Physiol., 292, 31-32P.

3. Hawes, D., Light, L. H., and Repond, E., (1979), "Modelling the Distortion Produced by Heel Strike Transients in Soft Tissue," J. Physiol., 296, 10-11P.

PHYSICAL THERAPY

1. Perry, J., "The Mechanics of Walking, A Clinical Interpretation," Physical Therapy, 47:778-801, 1967.

2. Smidt, G. L., Wadsworth, J. B., "Floor Reaction Forces During Gait; Comparison of Patients with Hip Disease and Normal Subjects," Physical Therapy, vol 53, 10, Oct., 1973, pp 1056.

ERGONOMETRICS

1. Grieve, D. W., and Gear, R. J., "The Relationships Between Length of Stride, Step Frequency, Time of Swing and Speed of Walking for children and Adults," Ergonometrics, 9:379-399, 1966.

2. Rao, B. K. N., and Jones, B., (1975), "Heel and Shoulder Vibration during Walking," Ergonometrics, 18, 555-566.

UNIVERSITIES

1. Andrews, J. G., Chao, E. Y., Johnston, R. C., and Stauffer, R. N., "A General Method for Accurate Analysis of Biomechanical Systems in Motion," U. of Iowa, Dept. of Orthopaedic Surgery, Iowa City.

2. Iowa, University of, Gait Measurement and Analysis, U. of Iowa State Walkway (Lower Extermity)

3. Waas, R., "A Digital Optical Scanning System for Kinematic Analysis," Thesis, Ph.D. Engineering, Case Western Reserve University, Sept. 1969.

BOOKS

1. Bernstein, N. A., "Co-ordination and Regulation of Movements." Oxford, Pergamon Press, 1967, 196 pp.

2. Braune, W. and Fischer, O., "Der Gang des Menschen." Bei S. Hirzel, Leipzig, 1895.

3. Carlsoo, S., "How Man Moves."

4. Delagi, E. F., and Perotto, A. (1980), "Anatomic Guide for the Electromyographer." Second Edition, Charles C. Thomas, Springfield, Illinois.

5. Dempster, W. T., "Free-body Diagrams as an Approach to the Mechanics of Human Posture and Locomotion," in: Evans, F. G., ed., Biomechanical Studies of the Musculoskeletal System. Springfield, Ill., Charles C. Thomas, Publisher, 1961.

6. Evans, F. G., "Biomechanical Studies of the Musculo Skeletal System." Springfield, Ill.:Charles C. Thomas, 1961.

7. Fischer, O., "Theoretische Grundlagen fur eine Mechanik der Lebenden Korper mit Speziellen Anwedungen auf den Menschen, sowie auf einige Bewegungs-vorgange an Maschinen." (Theoretical Fundamentals for Mechanics of Living Bodies with Special Applications to Man, as well as to Some Processes of Motion in Mechanics.) B. G. Teubner, Leipzig and Berlin.

8. Frankel, V. H., and Burstein, A. H., "Orthopaedic Biomechanics - The Applications of Engineering to the Musculo Skeletal System." Phila:Lea and Febiger, 1970.

9. Frost, H. M., "Orthopaedic Biomechanics." (RD732/.F939)

10. Kenedi, R. M., ed., "Biomechanics and Related Engineering Topics." Oxford, Pergamon Press, 1964. (R856/.S989/1964)

   1. Paul, J. P., "Bioengineering Studies of Forces Transmitted by Joints (II)," in: Biomech. and Related Bioengineering Topics. Kenedi, R., ed., Pergamon Press, 1965.

   2. Contini, R., Gage, H., and Drillis, R., "Human Gait Characteristics," in: Kenedi, ed., Biomechanics and Other Bioengineering Topics. 1964.

11. "Kinesiology III." Washington DC, American Association of Health, Physical Education and Recreation, 1973.

   1. Hay, J. G., "The Center of Gravity of the Human Body," in: Kinesiology III. Washington DC, American Association of Health, Physical Education and Recreation, 1973.

   2. Miller, D. I., and Petak, K. L., "Three-dimensional Cinematography," in: Kinesiology III. American Association for Health, Physical Education, and Recreation, Washington, D. C., 1973, 14.

12. Klopsteg, P. E., and Wilson, P. D., eds., "Human Limbs and Their Substitutes. NY:Hafner Publishing Co." 1968.

13. Marey, E. J., "Movement." Translated by E. Prichard, New York: Appelton, 1895, 323 pp.

14. Muybridge, E., "Animals in Motion", Brown, L. S., ed., Dover Publications, New York, 1957.

15. Muybridge, E., "Animal Locomotion: The Muybridge Work at the University of Pennsylvania." Lippincott, Philadelphia, 1888.

16. Plagenhoef, S. C., "Patterns of Human Motion: A Cinematographic Analysis." Englewood Cliffs, N. J., Prentice-Hall, Inc., 1971.

17. Steindler, A., "Kinesiology of the Human Body Under Normal and Pathological Conditions." Springfield, Charles C. Thomas, 1955.

18. Weber, W. and Weber, E., "Mechanik der Menschlichen Gehwerkzeuge." Gottinger, in der Dieterschen Buchhandlung, 1836.

19. Williams, M., and Lissner, H. R., "Biomechanics of Human Motion." Saunders Co., Philadelphia, 1962.

MISCELLANEOUS

1. Anon (?), "The Antiquity of Human Walking," Sci. Am., v 216, NY, April, 1967.

2. Ariel, G., "Computerized Biomechanical Analysis of Human Performance," ASME Symp. Mechanics and Sports, 1973.

3. Baon, L., Kenwright, J., and Hirsch, C., "Gait Analysis After Prosthetic Replacement of the Hip Joint," Phys. Med. Biol., 17(5), 1972, (recd 1973), 741.

4. Brooks, and Jacobs, "The Gamma Mass Scanning for Inertial Anthropometric Measurement," Med. Sci. Sports, Vol. 7, 290-294, 1975.

5. Cavagna, G. A., Saibene, F. P., Santi, and Margaria, "Analysis of Mechanics of Motion," Exp. Med. Surgery, 1963, 21, 117-126.

6. Chandler, R. F., Clasuer, C. E., McConville, J. T., Reynolds, H. M., and Young, J. M., "Investigation of the Inertial Properties of the Human Body," Report DOT-HS-801 430, 1975.

7. Chao, E. Y., "The Biomechanics of Total Joint Replacement Surgery," Geriatrics, 31(3):48-57, Mar 1976.

8. Contini, R., and Drillis, R., "Kinematic and Kinetic Techniques in Biomechanics," in: Advances in Bioengineering and Instrumentation, Plenum Press, 1966.

9. Dempster, W. T., and Gavphran, G. R. L., "Properties of Body Segments Based on Size and Weight," Am. J. Anat., 120, 33-54.

10. Dommasch, H. S., Brandell, B. R., and Murray, E. B., "Investigation into techniques of Gait Analysis," J. Biol. Photogr. Assoc., 40(3), 1972, 106-116.

11. Elftman, H., "The Orientation of the Joints of the Lower Extremity," Bull. Hosp. Joint Dis., 6:139-143, 1945.

12. Evans, F. G., "Bibliography on the Physical Properties of the Skeletal System," Art. Limbs, II, 2, pp 48-66, 1968.

13. Frankel, V. H., "Biomechanics of the Musculo Skeletal System, Introduction," Arch. Surg., 107(3):405, Sept 1973.

14. Glanville, A. D., and Kreezer, G., "The Characteristics of Gait of Normal Male Adults," J. Exp. Physiol., 21:277-301, 1937.

15. Hargreaves, P., and Scales, J. T., "Clinical Assesment of Gait Using Load Measuring Footwear," Acta Orthop Scand., 46(6):887-895, Dec 1975.

16. Hatze, H., "A New Method for Body Parameter Measurement"

17. Hennacy, R. C., and Gunther, R., "A Piezoelectric Crystal Method for Measuring Static and Dynamic Pressue Distributions in the Feet," J. Am. Podiatry Assoc., 65(5):444-449, May 1975.

18. Hooker, W. W., "A Set of r Dynamical Attitude Equations for an Arbitrary n Body Satelite having r Rotational Degrees of Freedom," AIAA Journal, 1970, 8, 1205-1207.

19. Huston, R. L., and Passerello, C. E., "On Lagrange's Form of d'Alembert's Principle," Matrix and Tensor Quarterly, 1973, 23, 109-112.

20. Inman, V. T., "Human Locomotion," The Canadian, Medical Association Journal, 94:1047-1054, May 14, 1966.

21. International Seminar on Biomechanics, Proceedings, 1st, 1967, (QP303/.I6)

22. James, S. L., and Brubaker, C. E., "Biomechanics of Running," Orthop, Clin North Am. 4(3):605-615, July 1973.

23. Jensen, R. K., and Metcaff, W. K., "A Systematic approach to Quantitative Description of Musculoskeletal Geometry," J. Anat., (to be published).

24. Kasvand, T., and Quanbury, A. O., "Computer Aided Study of Human Motion," Computer Graphics, 1972, 6, 53-60.

25. Liberson, W. T., "Biomechanics of Gait: A Method of Study," Arch. Phys. Med., 46:37-48, 1965.

26. Marey, E. J., "De la Locomotion Terrestre chez tes Bigedes et les Quidrupeds," J. Anat. Physiol. 9, 42, 1873.

27. Marey, E. J., and Demeny, G., "Etudes Experimentales de la Locomotion Humaine," Computes Rendus, Acad. Sci., Paris, 105:544-552, 1887.

28. Murray, M. P., "Gait as a Total Pattern of Movement," Amer. J. Phys. Med., 46:290-333, 1967.

29. Murray, M. P., Seireg, A., and Scholz, R. C., "Center of Gravity, Center of Pressure, and Supportive Forces during Human Activities," J. Appl. Physiol.,, 23(6), 1967.

30. Nubar, Y., and Contini, R., "A minimum Principle in Biomechanics," Bull. Matl. Biophys., 1961, 23, 377-391.

31. Paul, J. P., "Load Actions on the Human Femur in Walking and Some Resultant Stresses," Exp. Mech, II/3, 1971, p. 121.

32. Paul, J. P., "The Pattern of Hip Joint Force During Walking," Digest of the 7th International Conference on Medical and Biological Engineering, Stockholm, Sweden, 1967, p. 516.

33. Peizer, E., and Wright, D., eds., "Human Locomotion," Proc. Conf. on Human Locomotion Eng., Inst. Mech. Eng., London, 1971.

34. Plagenhoef, S. C., "Methods for Obtaining Data to Analyze Human Motion," Res. Q. Am. Assoc. Health Phys. Educ., 37:103-112, 1966.

35. Ralson, H. J., "Energy-speed Relation and Optimal Speed During Level Walking," Int. Z. Ang. Physiol., vol. 17., pp. 227-283, 1958.

36. Roberts, T. D. M., "Standing with a Bent Knee," Nature, vol. 230, April 23, 1971, p 499-501.

37. Rozin, Robin, Magora, Simkin, Gonen, "Investigation of Gait, Part 2: Gait Analysis in Normal Individuals," Electromyography, 11(2), 1971, 183-190.

38. Schmidt-Nielson, K., "Scaling in Biology: The Consequences of Size," J. Exp. Zool., 194(1):287-307, Oct., 1975.

39. Van Hemelen (1977)

40. Weinbach, A. P., "Contour Maps, C. G. Moments of Inertia and Surface Area of the Human Body," Human Biology, 10, Sept., 1938, pp 356-371.

41. Winter, D. A., Greenlow, R. R., and Hobson, D. A., "Television-computer Analysis of Kinematics of Human Gait," Comput. Biomed. Res., 5, 489-504, 1972.

END OF FILM

EASE REWIND