SINTON, DAVID JAMES

THE EFFECTS OF MULTIPARAMETER TARGET STIMULI ON HUMAN SACCADIC EYE MOVEMENTS STUDIED WITH AN AUTOMATED SACCADE ANALYSIS SYSTEM

The Ohio State University PH.D. 1979

University Microfilms International 300 N. Zeeb Road, Ann Arbor, MI 48106 18 Bedford Row, London WC1R 4EJ, England

THE EFFECTS OF MULTIPARAMETER TARGET STIMULI ON HUMAN SACCADIC EYE MOVEMENTS STUDIED WITH AN AUTOMATED SACCADE ANALYSIS SYSTEM

DISSERTATION

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the Graduate School of The Ohio State University

By

David James Sinton, B. S., M. S.

* * * * *

The Ohio State University

1979

Reading Committee:

Professor H. R. Weed

Professor R. M. Campbell

Professor H. R. Blackwell

Approved By

H. Richard Blackwell

Advisor

Institute for Research in Vision

VITA

July 24, 1950 . . . . Born - Denver, Colorado

1972 . . . . B. S. Engineering Physics, Colorado School of Mines, Golden, Colorado

1973 - 1975 . . . . Teaching Assistant, Colorado School of Mines, Golden, Colorado

1975 . . . . M. S. Physics, Colorado School of Mines, Golden, Colorado

1975 - 1977 . . . . Research Assistant, Department of Biophysics, The Ohio State University, Columbus, Ohio

1977 - 1979 . . . . Research Assistant, Institute for Research in Vision, The Ohio State University, Columbus, Ohio

FIELDS OF STUDY

Major Field: Biomedical Engineering

Studies in Instrumentation. Professor R. M. Campbell

Studies in Biological Modelling. Professor H. R. Weed

Studies in Physiology. Professor A. L. Kunz

TABLE OF CONTENTS

Appendix

LIST OF TABLES

LIST OF FIGURES

Figure

Figure

INTRODUCTION

The properties of human eye movements have interested researchers extensively for many years. The most common eye movements, saccades, naturally have drawn particular attention. In most situations it is not possible to redirect one's gaze without saccadic movements. Saccadic eye movements are quick step-like changes in eye position which rapidly reposition the eye to new fixation points. The eye movement control mechanism uses saccadic movements to minimize the time required to cancel positional errors of the fovea with respect to target retinal image. As such the properties of saccadic eye movements tend to be "preprogrammed" by the neural control centers. That is, the magnitude, velocity, and other properties are set before the movement begins. Then after the movement is initiated, no change in the properties of that particular saccade can be illicited. Investigators have measured the properties of saccades in the laboratory in an attempt to characterize the neurological control system and to relate changes in saccadic properties to certain physiological disorders. Some of the properties of saccades which are measured by these investigators are, latency of response, magnitude of response, intersaccadic duration, velocity and acceleration of the eye during the saccade, duration of the saccade, and eye muscle forces involved with saccadic movements.

In view of the large number of individual investigators and the complexity of the physiological processes involved, it is perhaps not surprising that many methods have been used to measure ocular saccades. To faithfully characterize eye movements, the measurement systems should have wide bandwidth, fast response time, low noise, high sensitivity and accuracy, insensitivity to artifacts such as eye blinks and head translation, and ease of use. The measurement systems commonly used by investigators illustrate the tradeoffs encountered between the properties mentioned above. Measurement systems tend to segregate into three classes based on their accuracy. There are the highly accurate methods, methods of medium accuracy, and methods of low accuracy. The highly accurate measurement methods are characterized by touching the subject's eye with some sort of "contact lens." However, these methods are extremely difficult and expensive to use. Contact lens methods are thus used only when high accuracy and sensitivity are demanded. The amount of data that can be gathered at any one time using contact lens methods is limited due to subject fatigue. The measurement methods of lowest accuracy are characterized by the measuring of the voltages changes associated with eye rotation (such as the electrooculogram or EOG). The EOG type methods are much easier to use and less expensive than any other methods. However, they have low bandwidth and are sensitive to muscle movements. The group of methods of medium accuracy are characterized by the measurement in changes of reflected light from the surfaces of the eye. These methods are not as sensitive or accurate as the contact lens methods and they are somewhat harder and more expensive to use than the EOG methods. These "reflection" methods are the most popular among investigators today. They have sufficient sensitivity and accuracy to measure all but the smallest of saccades (the so-called microsaccades). They do not require contact with the surface of the eye. However, some types of reflection methods encounter difficulties when measuring saccadic directions other than horizontal. Accordingly, most of the published data concerning saccadic movements has been limited to horizontal directions. Only recently have reports appeared concerning eye movement properties in directions other than horizontal.

The difficulties involved with eye movement research do not end once the method of measurement is chosen. Now the investigator must analyze the data that is collected. In the past eye movement data were recorded on strip charts, magnetic tapes, or other similar media and then analyzed later through hand calculations. This tedious process tended to limit the amount of data gathered and analyzed. With the advent of less expensive computer hardware the current trend is to utilize computer software to analyze eye movement data. However now the added complexity of computer interfacing and software development is included in the list of problems encountered by eye movement researchers. The investigator must now become an expert in computer hardware and software in addition to being well versed in eye movement research. This is a combination not often encountered.

All of the afore mentioned difficulties in measuring and analyzing eye movements (in particular saccadic eye movements) have combined to create deficiencies in the knowledge of the properties of saccades. Not enough is known about the properties of saccades at angles other than horizontal. Not much is known concerning the effects of target contrast on saccades at angles other than horizontal. Published data do not exist concerning saccadic settling time.

The objectives of the present research are to provide data concerning the relations between saccadic latency and settling time to target position and contrast and to modify an existing saccade analysis system to take these data. The original saccade analysis system was developed by James Brown for his Ph.D. dissertation at the Ohio State University (Brown, 1977). Modifications done here to Brown's system include a new and more elaborate target presentation system to permit control of target contrast and computer selection of target parameters. Additions and changes were also made to the software packages of Brown's system to include full control of target parameters, and additional response data analysis. As with Brown's system the factors of accuracy of stimulus control, ease of system use, and accuracy and speed of data acquisition and analysis were taken into account in the modifications performed here on Brown's system.

The capabilities of the modified saccade analysis system are as follows: (1) untrained subjects may be used, (2) subjects view the test stimuli with binocular vision and in most cases may wear eye glasses if required, (3) the right eye of the subject is monitored but not touched by the measuring apparatus, (4) the measuring apparatus has a sensitivity of 2.5 minutes of arc and a band width of about 500 Hz, (5) the computer controls the test stimulus parameters, records, and analyzes the subject's response, extracting saccadic latency and settling time from the response, (6) saccadic latency and settling time are calculated with an error of 5 milliseconds (depends on the sample time), (7) any one of 40 target positions and nine contrast levels may be automatically presented to the subject, (8) target contrast is controlled to within 0.5% accuracy, (9) rapid testing may be accomplished with total system throughput of about 3 minutes per saccade, yielding large amounts of data in a relatively short time.

The saccade analysis system was used to test the responses of subjects to target position and contrast. One subject was tested in detail resulting in the collection of over 4800 responses. The data from this subject are presented here. The analysis indicates a small dependence of saccade properties on target position for this subject. However, the saccade properties are strongly affected by the target contrast. Saccadic latency triples as the target contrast decreases below the subject's foveal contrast threshold. The system was also utilized with a group of about 50 untrained subjects to demonstrate its ease of use. The data from this group of subjects will not be presented here.

THE AUTOMATED SACCADE ANALYSIS SYSTEM

System Overview

The saccade analysis system described here is a computer based system that can be used to measure, record, analyze, and output subject responses to two dimensional step-like target displacements and step changes in target contrast. The system can be easily utilized with untrained subjects. The subjects are allowed to view the stimulus target binocularly while the system monitors the subject's right eye. Within limits subjects may wear their eye glasses, if required.

The system described here is based on the saccade analysis system developed by Brown for his doctoral dissertation. The reader will be referred to Brown's dissertation for certain system details (Brown, 1977). The modifications to Brown's system performed here include a more complex target presentation system. This target presentation system provides the user with several advantages: (1) target light levels can be carefully controlled so that target contrast is controlled, (2) the computer controls the position and contrast of the target test stimulus, (3) several types of operations can be performed with the target presentation system including foveal and off-axis contrast threshold determinations. Software changes to Brown's system include the addition of software to control the target presentation system, changes to existing plotting and analysis software to accommodate three target parameters (two dimensional position and contrast); and new averaging software so that subject responses could be averaged according to target parameters. This last software addition was accomplished so that the subject's saccadic response properties could be displayed as functions of the subject's relative visibility level.

The design requirements for the modifications performed here on the saccade analysis system of Brown are as follows: (1) the target contrast must be accurately controlled, (2) the target position must be accurately controlled, (3) the target presentation system must be flexible enough to allow for contrast threshold measurements and eyetracker calibration, (4) the experimenter must not be overburdened with tasks during the actual experimental sessions. To meet these requirements a complex target presentation system was developed and new software was added to the saccade analysis system program packages. The target presentation system has several features to help meet the above requirements: (1) projectors are used as light sources, this yields a flexible system to allow for the various threshold measurements and calibration procedures, (2) photodiodes are mounted on each projector to monitor their light outputs, (3) a line voltage regulator is used, this combined with the photodiodes yields accurate control of the target contrast, (4) a complex system of stepper motors, filters, and target spot masks is used to allow for computer control of the target parameters, this reduces the amount of work the experimenter has to do, (5) position sensors were included with these stepper motors for accurate control of target parameters, (6) new software (ADCOM, SSPCOM, and SACCOM) and modifications to existing software (SPROUT) were accomplished to achieve full computer control of the target parameters, plotting of analyzed response data, and averaging of the analyzed response data.

The major hardware components of the saccade analysis system as illustrated in Fig. 1 are (1) a complex target presentation system, to provide visual stimuli to the subject, (2) a PDP-8/E computer, to select target parameters and to record, analyze, and output subject saccadic responses, (3) a modified Cornsweet-Crane double Purkinje image eyetracker, to monitor the two dimensional rotations of the subject's eye, (4) a dual flexible disk system to provide mass storage for system software and data, (5) a digital plotter, to plot both raw and analyzed subject responses, and (6) a teletype to provide both user interaction with system software and output for averaged subject responses.

The software packages developed of the saccade analysis system are (1) ADCOM, a new program package used to select target parameters, control the target presentation system, manage the sampling of the eyetracker output, and store sampled data on mass storage, (2) APLOTØ, a plotting program package used to plot the raw subject responses as produced and stored by ADCOM, (3) SPROUT, a modified analysis program package used to analyze large numbers of ADCOM-produced subject

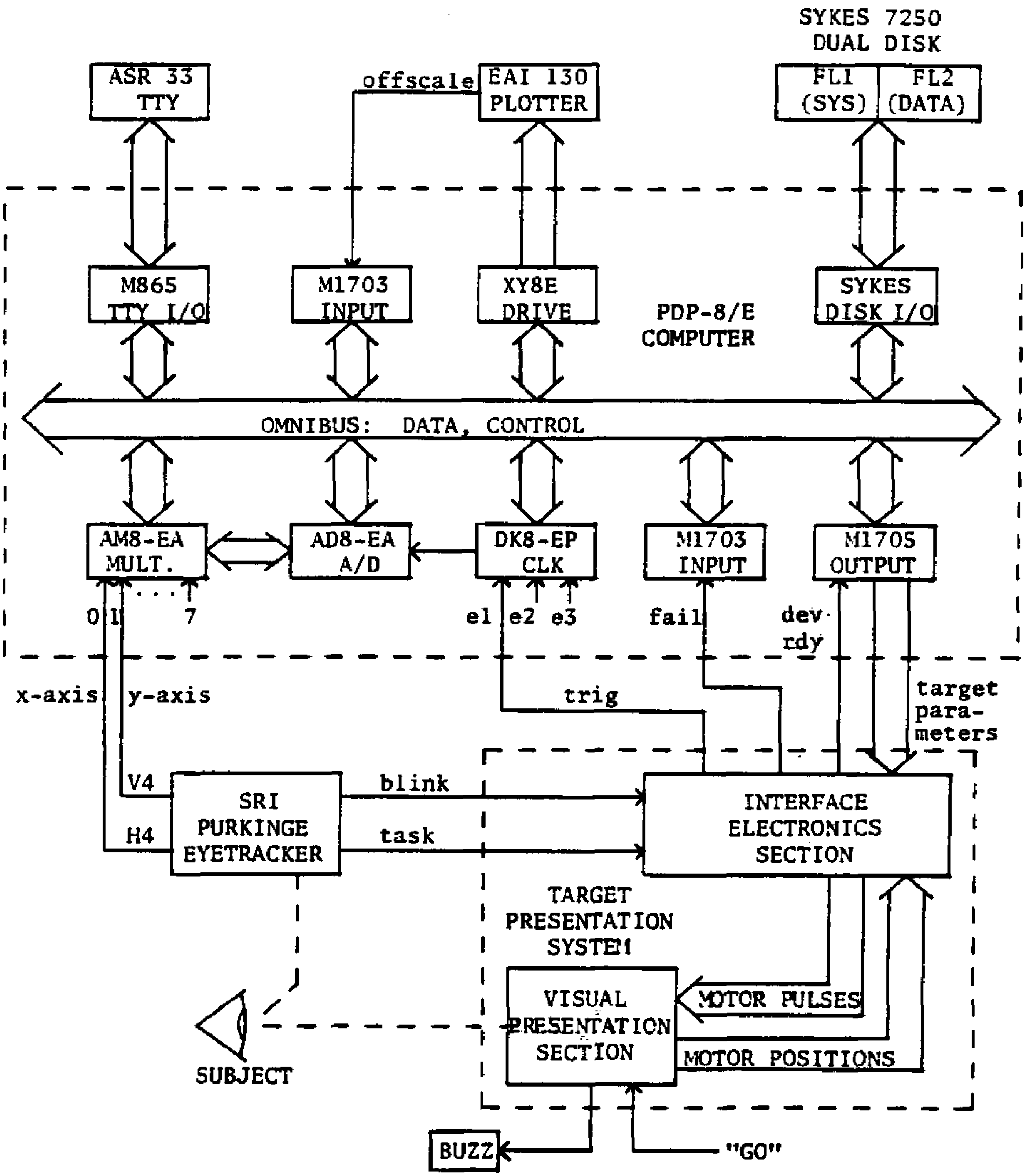


FIG. 1 - SACCADE ANALYSIS SYSTEM HARDWARE OVERVIEW

responses and store the results as a summary data file on mass storage, (4) SSPCOM, a new plotting program package used to plot the summarized data files as produced by SPROUT, and (5) SACCOM, a new averaging program package used to sort and average large numbers of SPROUT-produced saccade summary data files; averaged results are written onto the teletype. The system software data flow is illustrated in Fig. 2. All software programs are written onto the teletype. The system software data flow is illustrated in Fig. 2. All software programs are written in FORTRAN II high level language with some intermixing of SABR (Digital Equipment Corporation assembly language) statements. Saccade analysis system software packages are called and controlled using teletype keyboard commands.

A more detailed description of the hardware and software component operation with respect to the entire saccade analysis system is given in the following sections.

Target Presentation System

The target presentation system (TPS) is designed to provide a variety of visual stimuli to the test subject. The TPS is divided into two sections, the visual presentation section, as illustrated in Fig. 3 and Fig. 4, and the interface electronics section. The interface electronics section of the TPS provides the interfacing of the TPS to the system computer. The interface electronics section also interfaces some signals of the eyetracker to the computer and provides the user with manual control of the TPS target parameter assemblies.

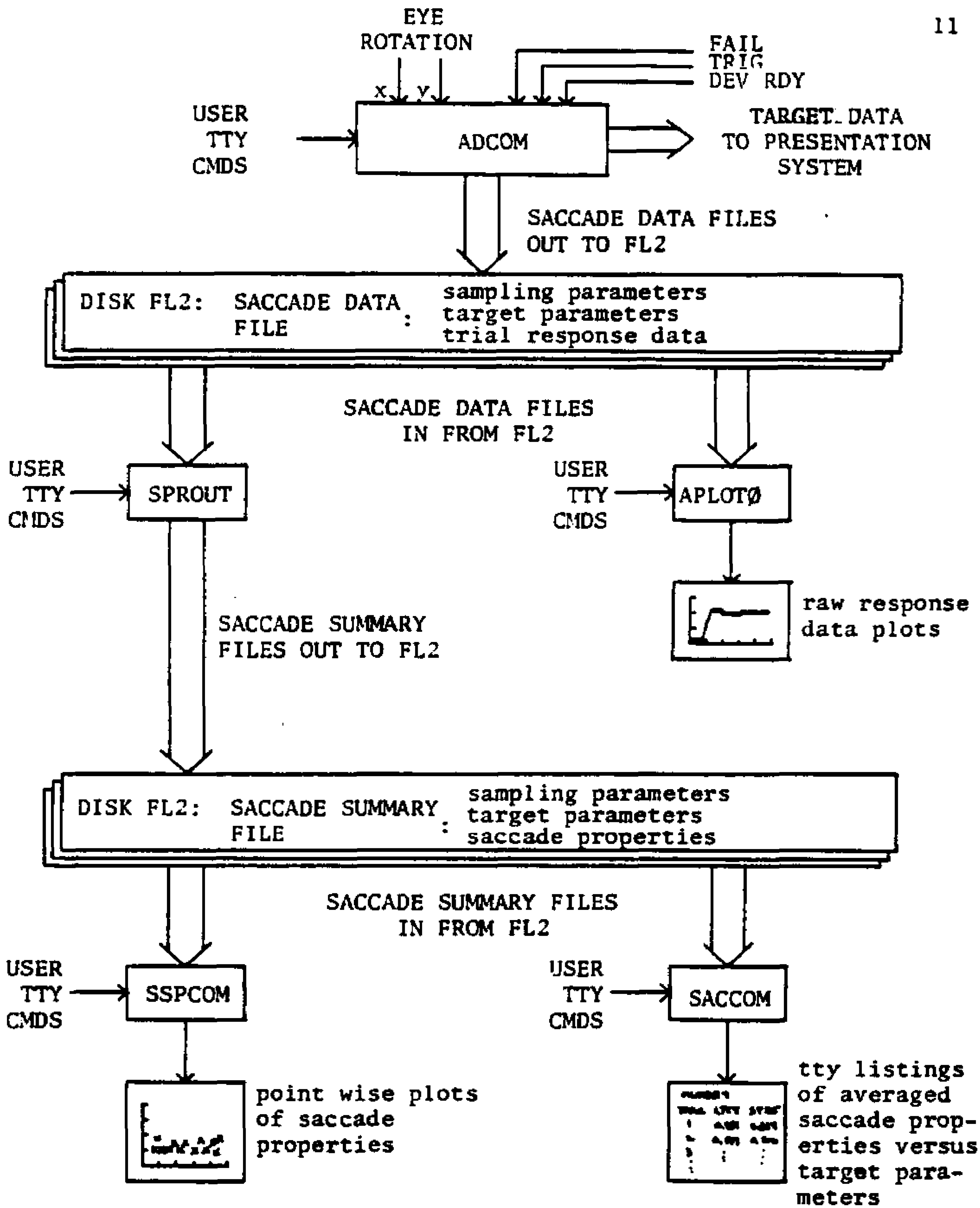


FIG. 2 - SACCADE ANALYSIS SYSTEM SOFTWARE OVERVIEW

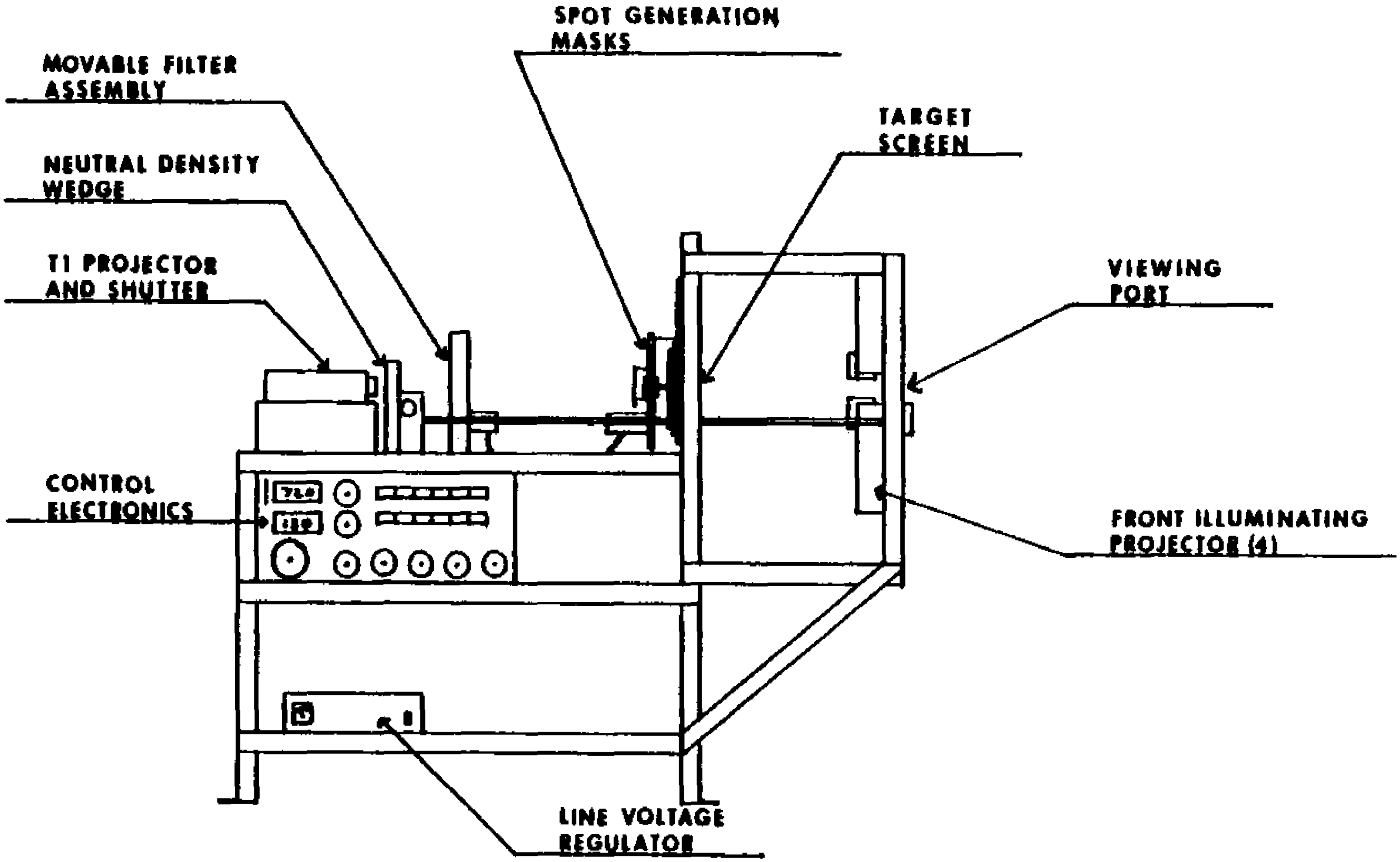


FIG. 3 - SCHEMATIC REPRESENTATION OF VISUAL PRESENTATION SECTION (SIDE VIEW) OF TARGET PRESENTATION SYSTEM

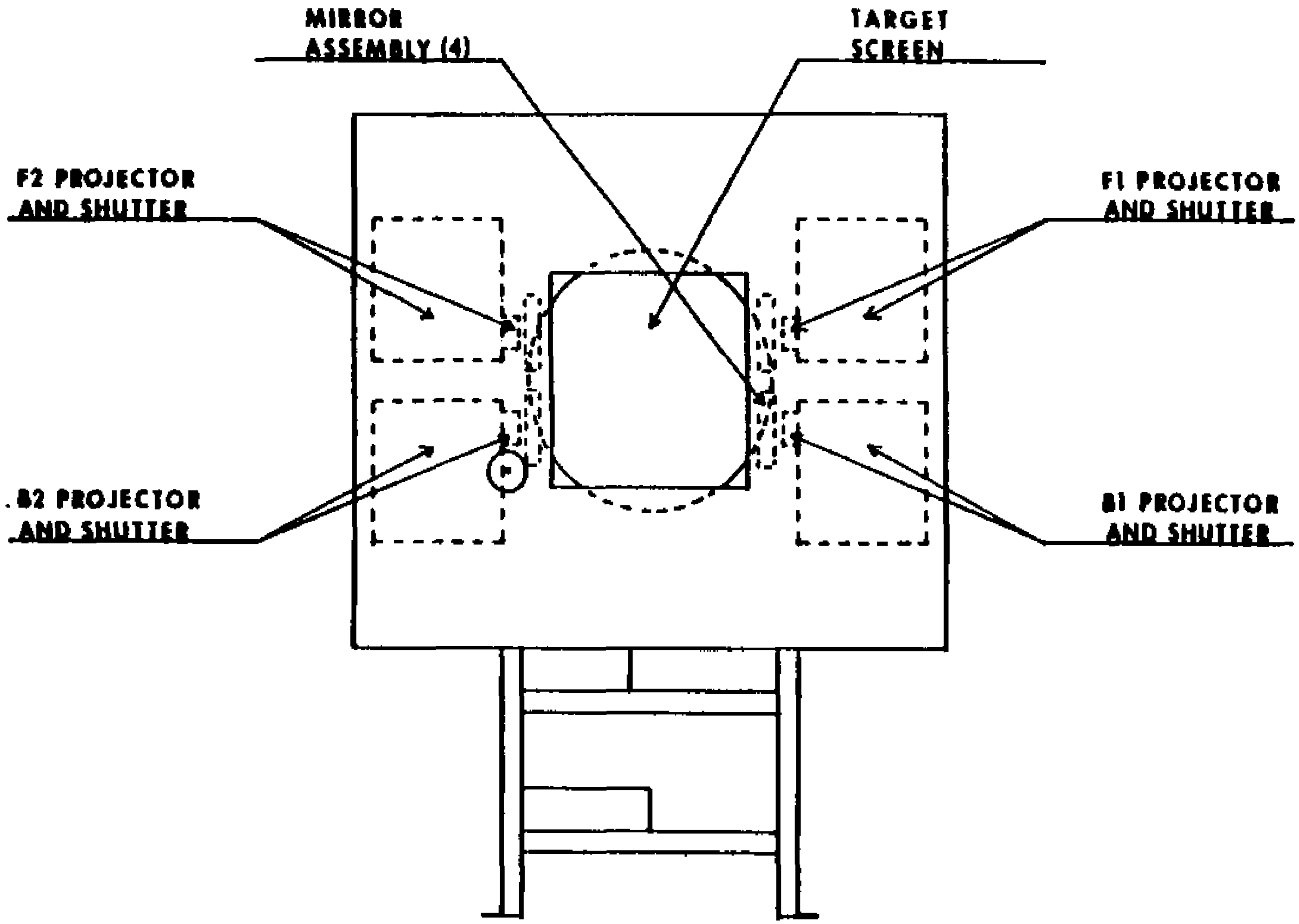


FIG. 4 - SCHEMATIC REPRESENTATION OF VISUAL PRESENTATION SECTION (FRONT VIEW) OF TARGET PRESENTATION SYSTEM

As illustrated in Fig.'s 3 and 4 the visual section of the TPS consists of five projectors (used as controllable light sources) and associated shutters, filters (used to set target contrast), a viewing screen and an electronics package for controlling and monitoring the light outputs and shutters of the projectors. The TPS visual section provides target spot locations and filters so that the user may select, under computer or manual control, any one of 40 possible target step deflections at any one of nine possible contrast levels. The target spot locations are arranged in eight azimuthal angles around a central fixation spot from 0 to 315 degrees in 45 degree increments. Zero degrees is referenced to horizontal and to the viewer's right of the central fixation spot while 90 degrees is vertical and directly up from the central fixation spot. Target spot step magnitudes may be any one of five radial distances from the central fixation spot the increments between radii being equal. For all work here the distance between the viewing screen and the subject is set at one meter so that the radial magnitudes of target steps as subtended at the subject's eye are 2.0, 2.5, 3.0, 3.5, and 4.0 degrees. Target "deflection" is provided by controlling the shutters associated with the TPS visual section projectors. One projector is used to provide the central fixation spot on the front surface of the viewing screen, while a second projector is used to back illuminate the spot generation masks and viewing screen. The shutters of the two projectors can be exchanged with one being opened while the other is closed simultaneously. Thus if used properly the shutter actions cause the central fixation spot to disappear at the same time that the selected target spot appears. The shutter actions are extremely rapid so that the time between the disappearance of the fixation spot and the appearance of the target spot is negligible. The appearance to the subject is that the central fixation spot has instantaneously jumped or "deflected" to a target location. The TPS visual section provides a warning tone signal to the subject just prior to target deflection. The warning tone time and target deflection time are adjustable over convenient ranges.

Step changes in the target spot contrast are provided by using the movable filter assembly of the TPS visual section and an associated neutral density wedge as indicated in Fig.'s 3 and 4. The neutral density wedge is adjusted to correspond to the subject's foveal contrast threshold. Then the nine neutral density filters of the movable assembly provide target contrasts of from foveal threshold to above foveal threshold in eight increments. The nine neutral density filters are arranged in a decreasing density order with each filter 0.115 log unit less dense than the previous filter for a total change of 0.920 log unit density over the range of the filters.

Computer positioning of the spot generation masks and movable filter assembly of the TPS visual section is accomplished by the use of the stepper motors attached to those assemblies. Computer software can then select the parameters of the target radius, angle, and contrast (or filter number) and direct the TPS visual section accordingly by driving the stepper motors.

The TPS visual section also provides the stimuli necessary for the evaluation of the test subject's foveal (or on-axis) and off-axis contrast thresholds, and for eyetracker gain calibration.

The TPS interface electronics sections provides all the signals necessary for computer control of the target parameters and subject eye movement data sampling. The TPS interface electronics section accepts "blink" and "track" signals from the eyetracker, movable assembly position signals from the TPS visual section, and shutter control signals from the TPS visual section. The TPS interface electronics section provides the computer with "failure" data, trigger signals, and TPS "ready" signals. Position signals from the TPS visual section are used to generate a "ready" signal which notifies the computer when movements of the TPS visual section target parameter assemblies have been completed. The shutter command signals are used by the TPS interface electronics section to generate a trigger pulse for the computer at the instant that target deflection occurs. This trigger pulse is used by the computer to initiate the sampling of the eyetracker x-axis (H4) and y-axis (V4) eye rotation signals. The subject's eye response to a target deflection is thus monitored and recorded beginning at the instant of target deflection. The TPS interface electronics section uses the eyetracker "blink" and "track" signals to generate a trial "failure" bit which is valid only during a target deflection. If either the subject blinks or the eyetracker loses track lock during the trial (target deflection time) the trial "failure" bit will be set. The ADCOM data input and control software package will test this "failure" bit and if set reject the data for that trial. No corresponding output data file to the disk will be produced.

For the saccade testing done here, a single trial sequence proceeds as follows: (1) After proper initialization of the TPS hardware under ADCOM direction, program ADCOM will select the target parameters for that trial. The TPS target parameters assemblies will respond by moving to their specified locations notifying both the computer and the experimenter when this procedure is completed. (2) The experimenter can then push a "GO" pushbutton to initiate a trial. (3) The warning tone sounds for about 1.5 seconds alerting the subject to fixate the central fixation spot on the viewing screen. (4) At the end of the warning tone the target deflects and the computer is triggered to begin recording the subject's eye response. (5) The target remains deflected for about 3 seconds. with the computer sampling the subject's eye response for the first 2.5 seconds. (6) At the end of sampling the computer tests the failure bit, and if no failure occurred proceeds to output the radius angle, and contrast (filter number) data of the target, and the subject's response data to disk storage as a saccade data file.

The reader is referenced to the appendix for a detailed description of the TPS visual presentation section and interface electronics section. Included are operating instructions, explanations of capabilities and operating modes, output specifications, and schematic diagrams of electronics.

Cornsweet - Crane Purkinje Image Eyetracker

The Purkinje image eyetracker was originally developed by T. N. Cornsweet and H. D. Crane. Current models of the machine are produced on a custom basis by SRI, Palo Alto, California. The eyetracker measures eye rotation by directing a collimated beam of infrared light into the eye and measuring the relative positions of two reflections, the first Purkinje image, produced by reflection from the front surface of the cornea, and the fourth Purkinje image, produced by reflection from the back surface of the lens. These two images move in concert when the eye translates but shift in opposite directions when the eye rotates. The positions of these two images are optically tracked using servo driven mirrors. The movement of the first Purkinje image can then be optically subtracted from the movement of the fourth Purkinje image to yield rotational signals that within limits are free from translational artifacts. The two dimensional fourth Purkinje image servo signals, after suitable processing, provide the electrical outputs H4 and V4 which are directly proportional to the horizontal and vertical angular rotations of the eye.

The eyetracker cannot distinguish certain translational artifacts from eye rotations that occur while the subject is viewing a fixed target. For example, if the subject is viewing a fixational target and translates his head to the left his eyes must rotate to the right to maintain foveal fixation. This will be interpreted as eye rotation by the eyetracker and shows up as essentially a baseline shift in the

H4 and V4 output records. To help reduce this sort of artifact, subjects used here are held stationary by means of a dental wax biting bar.

For all work done here the eyetracker was adjusted for each subject to yield H4 and V4 outputs of ± 0.2 volts per ± 1 degree of eye rotation from the central fixation position. Performance specifications for the eyetracker are listed below. The reader is referred to Dr. James Brown's dissertation for a more detailed description of the eyetracker operating principles, modifications, and methods used to measure the performance specifications.

RMS equivalent rotational noise, H4 and V4 outputs	2.5 minutes of arc
Equivalent rotational slew rate, H4 and V4 outputs	60 degrees/second
Response delay, H4 and V4 outputs	2 milliseconds
Rotational linearity within a 4-degree radius circle, H4 and V4 outputs	error less than noise

Computer Hardware and Software

Experimental data is input, processed, and output by the saccade analysis system's PDP-8/E computer under the control of software packages ADCOM, APLOTØ, SPROUT, SSPCOM, and SACCOM. These software packages were all developed specifically for the saccade analysis system. As mentioned earlier some of these software packages are extensions of ones developed by Dr. James Brown for his doctoral dissertation. The reader will be referred to his dissertation at times for a more detailed discussion.

As shown in Fig. 1 the computer outputs target parameter data (in the form of the number of stepper motor steps required) to the target presentation system (TPS) and accepts input ready, failure, and trigger signals from the TPS. Additionally the computer accepts analog eye rotation signals from the eyetracker. The H4 and V4 eyetracker output signals are digitally encoded by the computer's multiplexer and analog-to-digital converter. Sample timing is controlled by software and by the computer's real time clock. The trigger signal from the TPS interface electronics section is generated at the beginning of each target deflection and is sent to the computer to start the real time clock and data sampling. The TPS interface electronics section supplies the computer with a trial failure bit which is valid only during the target deflection time. The trial failure bit is set by the TPS interface electronics section if the subject blinks or the eyetracker loses track lock during the target deflection period. The TPS interface electronics section also supplies the computer with a ready signal which is set anytime stepper motor movements of the visual presentation section are completed.

As indicated in Fig. 2 the software package ADCOM controls the output of target parameter data to the TPS and the input of eyetracker data from the eyetracker and ready, fail, and trigger data from the TPS. ADCOM allows the user to control the target parameters, sampling parameters, and data storage file names through the use of teletype keyboard commands. ADCOM provides two sampling modes, a single trial or "one-shot" mode, and a multiple trial or "recycle" mode to automatically control target parameters and input and store trial data. Data for each trial is stored on disk FL2 as a data file whose name has been previously declared by the user. The sampling parameters are initialized to a convenient set of values when ADCOM is first called for use. In addition at ADCOM beginning the user is required to enter a beginning data file name and an initial "scrambling factor" for the target position array.

The one-shot mode sequence of data control and sampling by ADCOM is outlined below. Timing values given are those used for all experiments performed here. (1) The software begins the one-shot mode by requesting the user to initialize the TPS target parameter assemblies and then entering a flag wait loop. The software waits for the DEV RDY signal from the TPS which indicates that the spot generation masks and the movable filter assembly are positioned in their starting or reference positions. (2) The software then selects the target parameters in a pseudo random fashion. That is, one of the 40 possible target positions is chosen from a scrambled, randomized target position array and a number from 0 to 8 is generated at random to indicate filter number (target contrast). The target parameters are sent to the TPS interface electronics section which in turn directs the visual presentation section stepper motors accordingly. (3) The software enters another flag wait loop waiting for the DEV RDY flag of the TPS to indicate that the motor movements are completed. (4) The software now enters a clock trigger wait loop. (5) The experimenter initiates a trial (target deflection) by pushing the TPS "GO" pushbutton. The warning buzzer sounds for about 1.5 seconds after which the target deflects for about 3 seconds. (6) At the onset of target deflection the TPS interface electronics section TRIG trigger signal starts the computer's real time clock. (7) The clock controls the multiplexer/converter and the subject's eye response (H4 and V4 eyetracker signals) are sampled every 5 milliseconds for 2.5 seconds. (8) The 1000 sample values are stored temporarily in a data array in the computer's core memory. (9) After the last data sample the software checks the TPS interface electronics section FAIL bit. (10) If a failure has occurred (FAIL bit set) no further action is taken on the sampled data, a new set of target parameters is chosen and output to the TPS, after the TPS stepper motors have completed all movements control is transferred to the clock trigger wait loop. (11) If no failure occurs (a successful trial) the software stores the target's radius, angle, and filter number information along with the coded sampling parameters at the beginning of the sampled data array, then outputs the entire array to disk FL2 as a data file with the user defined file name. (12) Software control is transferred back to ADCOM command mode. The user may now enter new commands.

ADCOM operation in the recycle sample/control mode begins as in the one-shot mode. However, each time a data file is output to the FL2 disk the target position (radius and angle) just used is inhibited from further use and also the file number portion of the current file name is incremented by one. The next sequential location in the randomized target position array is chosen to yield a new target position. A new filter number is chosen and output along with the target position to the TPS interface electronics section. After the TPS stepper motors have completed their movements, ADCOM software control is passed to the clock trigger wait loop. In this manner, up to 40 consecutively numbered data files may be generated and stored on disk FL2, one for each target position of the randomized target position array. Each ADCOM-generated data file contains a header with the following information about the data: file name, multiplexer input channel numbers, sampling period, number of samples, target radius and angle, and filter number (target contrast). More detailed information about ADCOM, its operation and a program listing may be found in the appendix.

Once a disk data file (or a number of data files) has been generated by ADCOM, the system user may plot this raw subject response data by using the software package APLOTØ. Normally saccade data files are not plotted in raw form but are processed by the software package SPROUT. However, if desired, APLOTØ may be used to plot any sampled channel as a function of any other sampled channel or as a function of time. Full axis scaling capabilities and other features are available as discussed in detail in Dr. Brown's dissertation. To illustrate a typical subject response for a target deflection trial, APLOTØ was used to generate the plots shown in Fig. 5. Shown are X and Y axis eye rotations as functions of time and a plot of Y-axis eye rotation as a function of X-axis eye rotation.

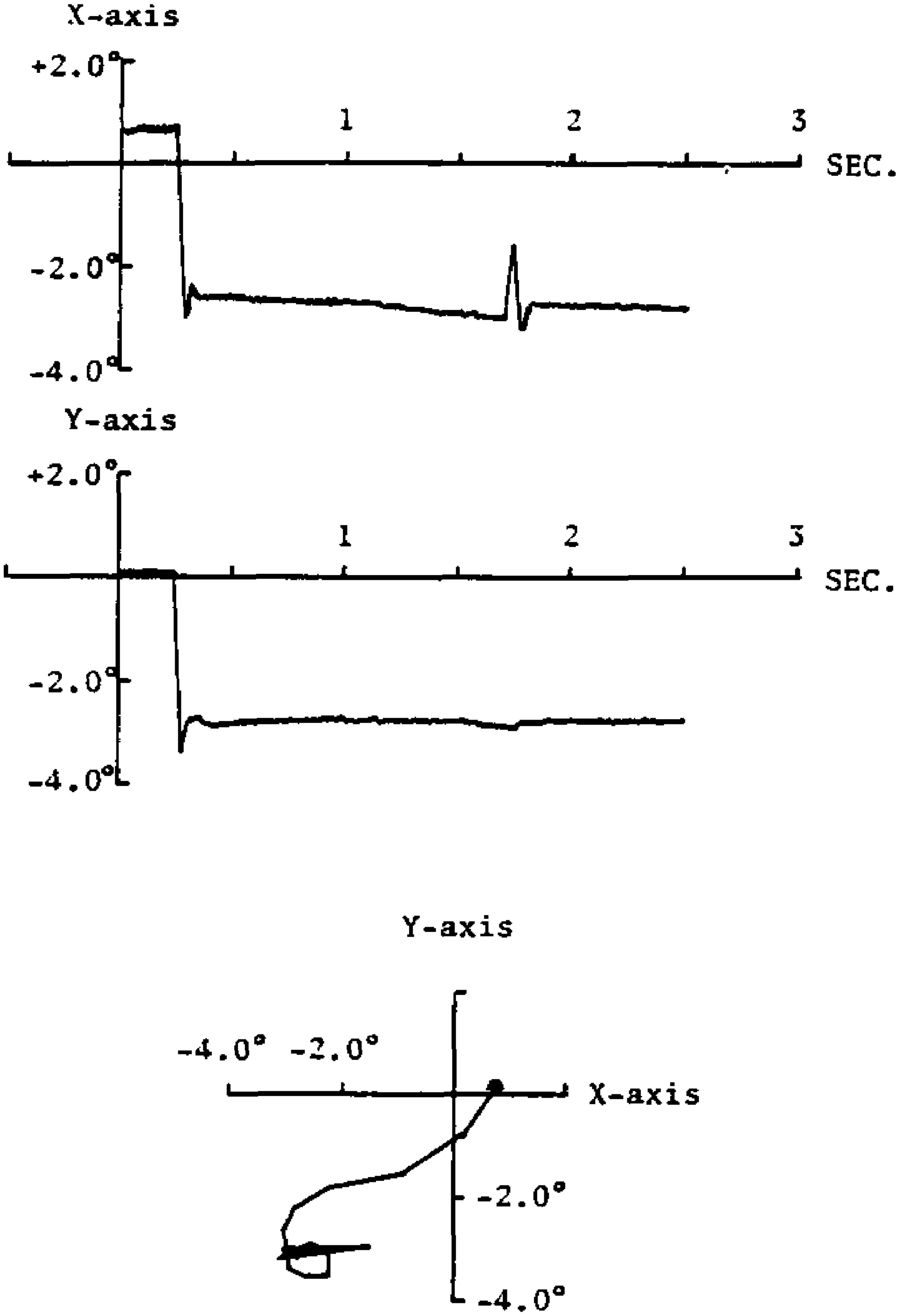


FIG. 5 - EXAMPLE OF APLOTØ OUTPUT. SUBJECT DEB RESPONSE TO TARGET STEP OF RADIUS 3.0°, ANGLE 225°, FILTER NUMBER 6.

The software package SPROUT is used to process ADCOM-produced data files of the form shown in Fig. 5. SPROUT operation is controlled by using teletype keyboard commands. These commands allow the user (1) to declare a file name for the SPROUT output array, (2) to load a previous SPROUT summary array into SPROUT, (3) to input single saccade data files or groups of consecutively numbered saccade data files to the summary array, (4) to delete entries from the summary array, (5) to print individual summary array entries, (6) to list summary array entries, and (7) to change processing threshold and scaling factors.

For each input saccade data file SPROUT calculates the latency of response and the settling times to within 25, 20, 15 and 10 minutes of arc radius about the final eye position. SPROUT also performs checks on the saccade data to insure that results from highly irregular data are not unknowingly included in the SPROUT output summary file.

SPROUT calculates saccadic latency by starting at the beginning of the saccade data record and calculating a velocity for each X-Y sample pair. The latency period ends at that sample just preceding the sample for which the velocity exceeds a threshold velocity (20 degrees per second for all work done here). To determine the settling time, SPROUT calculates the "final eye position" by averaging the X and Y data for the final 1 second of the 2.5 second data record. Then starting at the beginning of the data record, SPROUT calculates the distance from each X, Y sample pair to this final eye position. The settling time, then, is the time elapsing from the start of the record to the first sample pair of a period of 0.1 second or longer for which the calculated distance to the final eye position is within the settling radius.

SPROUT output is a summary data array which is stored on disk FL2 as a saccade summary data file. Each entry in the summary data file contains the following data about one saccade data file: the file name of the original saccade data file, the sampling parameters used to obtain the original data, the target radius, angle, and filter number used for that trial, and the calculated latency and settling times as discussed above. Additionally the very first entry of a SPROUT-generated summary data file contains the summary data file name and the number of summarized saccade files included in the summary data file. The processing results from up to 63 individual trials may be stored in each SPROUT output summary data file. These summary files are compact and space saving and they provide readily accessable data for further processing. Processing considerations and a more detailed discussion of SPROUT software are presented in the appendix.

After using SPROUT to process and summarize saccade data files, the summary files may be plotted using the SSPCOM software package. This software package is a specially designed plotting package used to accept the input of SPROUT-created summary files and generate a variety of pointwise plots. Latencies and settling times from a summary file may be plotted as functions of target radius, angle, or filter number with up to two parameters. For example the user may plot latency as a function of filter number with a target radius and/or a target angle declared as a data parameter. The user has full control of axis scaling and may choose any one of five symbols to represent the data points. An example of an SSPCOM-generated plot is shown in Fig. 6. A more detailed discussion of SSPCOM may be found in the appendix.

Further processing of SPROUT-created summary data files is accomplished by the software package SACCOM. SACCOM is specifically designed to input summary data files and sort and average the saccade properties (latency and settling time) according to target parameters (radius, angle, and filter number). SACCOM searches the summary data file for all summarized saccade trials which match their target radius, angle, and filter number, and averages the latency and settling time values for those trials. Control of SACCOM operation is by teletype keyboard commands. These commands allow the user to input summary files, zero the average data array, declare file names, declare a filter number, and print the averaged results on the teletype. SACCOM output is a printed teletype array of the averaged latencies and settling times (mean and standard deviation values are included) according to target position (radius and angle) and filter number. The user must declare a filter number before beginning processing. An example of SACCOM output is shown in Fig. 7. Further details of SACCOM operation and processing considerations may be found in the appendix.

Digital Plotter

The digital plotter used in this system is a model 130 Dataplotter, manufactured by Electronic Associates, Inc., West Long Beach, New Jersey. It is an X-Y plotter with a single pen. The pen-up,

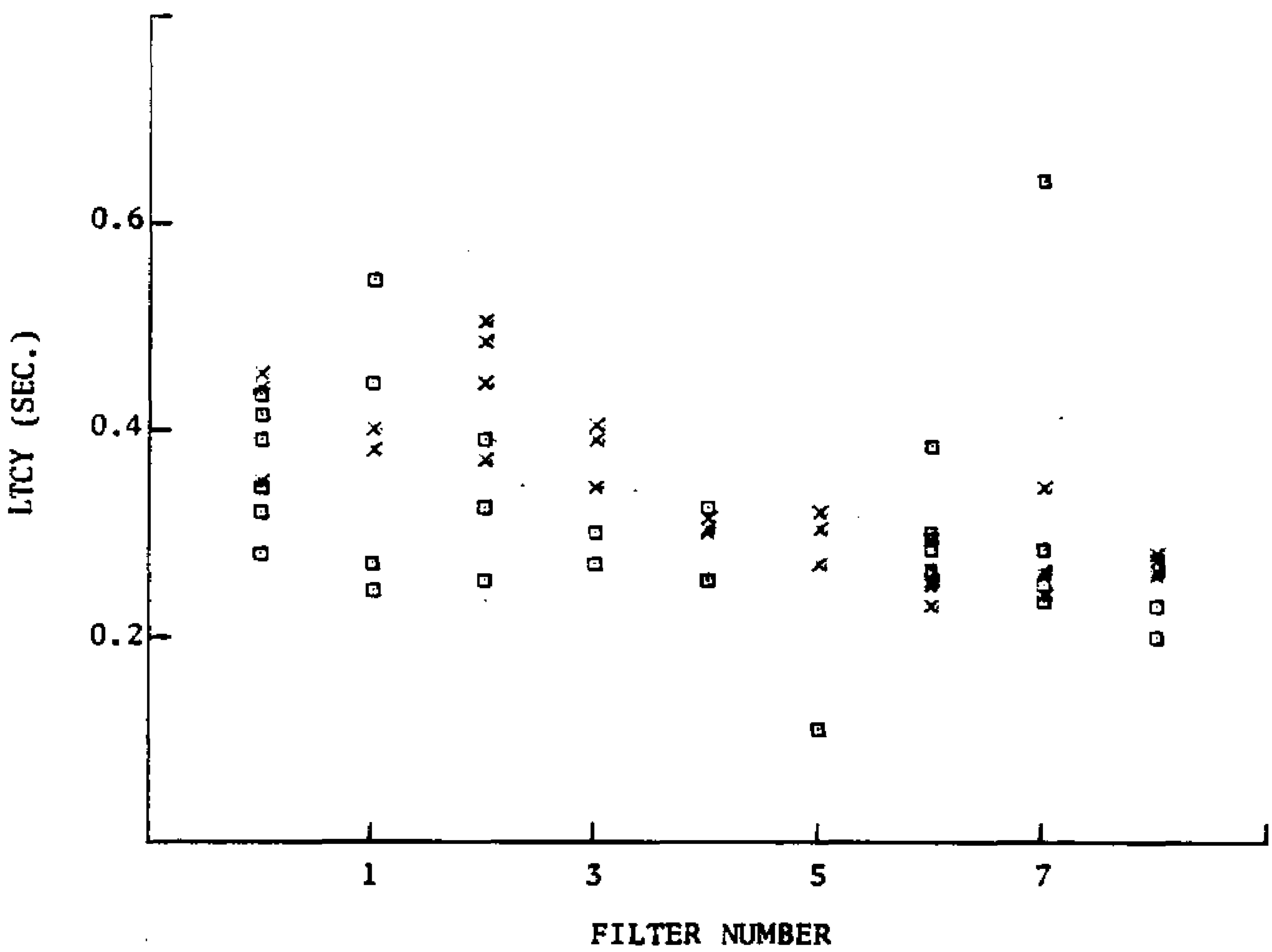


FIG. 6 - Example of SSPCOM output. Subject DEB responses to target deflections 2.0° (X) radius and 4.0° (□) radius. All target angles included. Y-axis is saccade latency (LTCY), X-axis is target filter number (contrast).

!P
FILTER NUMBER: 6

TARG POS	LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	NO. SAMP
1	0.278	0.302	0.305	0.308	0.535	3
STD DEV	0.047	0.033	0.038	0.038	0.196	
2	0.275	0.365	0.368	0.378	0.555	3
STD DEV	0.013	0.104	0.106	0.102	0.218	
3	0.263	0.313	0.345	0.391	0.510	13
STD DEV	0.021	0.071	0.080	0.124	0.239	
4	0.253	0.277	0.293	0.375	0.622	3
STD DEV	0.006	0.003	0.028	0.090	0.144	
5	0.000	0.000	0.000	0.000	0.000	0
STD DEV	0.000	0.000	0.000	0.000	0.000	
6	0.205	0.370	0.435	0.463	0.673	3
STD DEV	0.092	0.156	0.144	0.171	0.500	
7	0.291	0.362	0.430	0.505	0.556	7
STD DEV	0.044	0.092	0.095	0.137	0.109	
8	0.242	0.261	0.275	0.394	0.526	8
STD DEV	0.029	0.024	0.036	0.122	0.129	
9	0.262	0.301	0.327	0.379	0.540	5
STD DEV	0.024	0.052	0.104	0.097	0.169	
10	0.262	0.332	0.392	0.490	0.500	2
STD DEV	0.011	0.046	0.124	0.007	0.014	
11	0.205	0.225	0.225	0.445	0.490	1
STD DEV	0.042	0.051	0.051	0.198	0.240	
12	0.260	0.339	0.355	0.436	0.590	7
STD DEV	0.012	0.076	0.090	0.183	0.211	
13	0.269	0.344	0.347	0.371	0.607	5
STD DEV	0.048	0.112	0.113	0.101	0.051	
14	0.254	0.316	0.383	0.391	0.429	6
STD DEV	0.026	0.091	0.118	0.118	0.129	
15	0.245	0.360	0.401	0.413	0.526	5
STD DEV	0.011	0.060	0.066	0.070	0.090	
16	0.285	0.470	0.470	0.475	0.475	1
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.

FIG. 7 - Example of SACCOM output. Listing continues through the 40 target positions. LTCY is saccade latency, STXX is saccade settling time to XX minutes of arc. First number is mean second number is standard deviation.

pen-down, +X, -X, +Y, -Y inputs of the plotter are cabled to a computer plotter interface. Under software control, this interface provides pulses to raise and lower the pen and to step the pen along the axes. Step size along both axes is 0.01 inch per step and the maximum step speed is 200 steps per second. The plotter is also equipped with manual pen control pushbuttons for initial pen positioning.

EXPERIMENTAL RESULTS

Experimental Conditions

During the tests done here with the saccade analysis system the subject was asked to view the target presentation system (TPS) viewing screen with binocular vision. The eyetracker was set to monitor the movements of the subject's right eye. The right eye of the subject viewed the TPS screen through a glass plate located a few centimeters in front of the eye. This glass plate caused no difficulty to the subject yet at the same time allowed the eyetracker to monitor the movements of the eye. The subject was seated in a stenographer's chair and held in a fixed position with respect to the eyetracker by using a dental wax bite bar. The bite bar was clamped to a small three-axis machine tool bed to allow alignment of the subject with respect to the eyetracker.

The target viewed by the subject was a sharply focused spot of light presented on the TPS viewing screen. The viewing screen of the TPS was illuminated from the front to provide a constant background illumination with respect to the target spots. The central fixation spot brightness was adjusted to a comfortable viewing level. The peripheral target spot brightness (that seen during a trial) was controlled by the computer. The location of the peripheral target spot on the TPS viewing screen was also controlled by the computer. The positioning of the TPS and its central fixation spot was such that the central fixation spot was centered in the visual fields of the subject and the eyetracker. The viewing screen of the TPS was also positioned 1 meter from the cornea of the subject's right eye so that 17.5 mm of distance on the TPS viewing screen corresponded to 1 degree of eye rotation. The target spots of the TPS (including the central fixation spot) all subtended about 8.5 minutes of arc at the subject's eye. The surfaces surrounding the TPS viewing screen were covered in black material so as not to distract the subject during experimentation.

Tests were conducted in an almost dark room. The subject was permitted approximately 15 minutes to adapt to the ambient light conditons of the room before any testing began. The testing of the subject occurred over a 6-week time span with no more than 2 hours of testing during any one day. The subject was allowed at least 10 minutes of rest after every 40 trials. This assured that the subject was fully alert and well rested during the testing periods.

The eyetracker was calibrated and the subject's foveal contrast threshold was determined before each testing session began. The eyetracker's gains and offsets were adjusted by having the subject fixate on selected points of an array of spots presented on the TPS viewing screen especially for this purpose. Eyetracker gains were set to 0.2 volt per degree of eye rotation. The subject's foveal contrast threshold was evaluated by using the TPS's PT shutter control mode and the reference target spot parameters of 2.0 degrees radius, 0 degrees angle, and filter number 0 (the most dense). Also an array of fixation spots was adjusted to surround the 2.0 degree radius, 0 degree angle target spot. Under these conditions the target spot located in the middle of the fixation array flashed. The subject then adjusted the TPS neutral density wedge so that he saw the flashing target spot about one half of the time that it was on (or one twelfth of the total time). The subject was requested to make this foveal contrast threshold determination several times in succession and the average of all determinations was used as the final measure of the subject's foveal contrast threshold. This average value was set onto the TPS neutral density wedge to provide the range of target spot contrasts as described earlier.

Tests

The subject was instructed to fixate the central fixation target of the TPS viewing screen when the TPS warning buzzer sounded and then follow as best as possible the "deflection" of the central target spot to the peripheral target location during the "deflection" period. Each target deflection was initiated by the experimenter when he pushed the "GO" pushbutton of the TPS. Each trial consisted of about 1.5 seconds of warning buzzer tone followed by about 3 seconds of target deflection. The deflection was synchronous with the end of the warning tone. After the 3 seconds of target deflection the target returned to its central position. A short time interval then elapsed during which the data record was stored by the computer on the disk and new target parameters were set by the TPS. Trial repetition rate was about one trial every 30 seconds. All 40 of the target positions and all 9 of the target contrast filters were used in the tests done here. Trials were conducted in groups of 40 corresponding to the 40 possible target positions. For each group of 40 trials the computer software generated a pseudo random sequence of trials. That is, one trial at each position was performed but the positions were presented in a random sequence. Filter selection was always on a random basis. The subject could never obtain foreknowledge as to the target's position and contrast.

On each testing day no more than four groups of 40 trials (160 total trials were done. This number was the limit obtainable before subject fatigue and boredom set in. A resting period of at least 10 minutes followed each 40 trials.

Subject

One male subject, DEB, was tested in depth with the system. Subject DEB was in his early twenties and did not wear eyeglasses.

Results

Over the course of the testing period of six weeks, 4800 subject responses were collected and processed with the saccade analysis system. Processing checks performed by SPROUT caused the rejection of approximately 7% of the subject responses. The SPROUT analyzed response data was averaged using SACCOM. Tables of the SACCOM averaged data are presented in the appendix. Here 16 plots of some of the averaged data are presented as Fig.'s 8 through 23. The first eight of these plots (one for each target azimuthal angle) illustrate saccadic latency (LTCY) versus a quantity called relative visibility level (RVL). The last eight plots illustrate saccadic settling time to 15 minutes arc radius (ST15) versus RVL. The ST15 settling time data is felt to be the most useful of the four settling times calculated by SPROUT. It represents an eye position close to the target position while the ST25 and ST20 data represent eye positions somewhat farther away. The ST10 data is contaminated by the small repositioning movements of the eye once it reaches the target.

The quantity relative visibility level (RVL) was derived from the subject's threshold contrast data and the filter number used in a particular saccadic trial. Specifically RVL is defined as the ratio of the physical contrast of the target to the subject's threshold contrast of the target. These two quantities can be calculated in terms of the neutral density wedge and filter transmissions of the target presentation system. The physical contrast of the target can be calculated as the product of the neutral density wedge transmission for the subject's on-axis foveal threshold times the transmission of the filter number used in the particular saccade trial. The subject's threshold contrast of the target can be evaluated by knowing the transmission of the TPS neutral density wedge for the subject's off-axis threshold for the target location. This quantity was evaluated using a similar procedure to that used for evaluating the subject's foveal (or on-axis) threshold. The subject was requested to fixate the central fixation spot on the TPS viewing screen. The particular target spot location was "flashed" using the PT shutter control mode. The subject was then requested to adjust the setting of the TPS's neutral density wedge until he was able to see the flashing target spot only one-half of the time that it was on (just as for the foveal threshold determination). This procedure was repeated several times for each target location. The individual threshold trials for each target location were averaged together to yield an average off-axis threshold contrast corresponding to each target location. The calculated value of RVL provided a measure of how far above or below threshold contrast the target contrast was for a particular saccade trial. An RVL of 1 implies that the target contrast of the saccade trial was equal to the subject's threshold contrast for that target location. By using RVL as calculated here we gain the following advantages: (1) variations in the spatial illuminating patterns of the TPS T and B1 projectors onto the viewing screen can be dealt with, and (2) the absolute magnitudes of the TPS projector light levels can be set to levels convenient for experimental use. Particular care was exercised to ensure that the projector light levels were reset to the same values at the beginning of each experimental session. Attention was also given to these light levels throughout the course of an experimental session to ensure that the settings were stabilized at their proper values.

Discussion

The graphs of Fig.'s 8 to 23 illustrate the behavior of saccadic latency (LTCY) and settling time to 15 minutes arc (ST15) versus RVL. It is noted that both the LTCY and ST15 data asymptote quickly for RVL values greater than about 2. The asymptotic values of LTCY correspond with those reported by many other experimenters. Westheimer (1954), Saslow (1966a, b), and Carlow (1975) among others have all reported saccadic latencies in the 200 to 300 millisecond range. No consistent differences in latencies versus the target radius can be observed from the data shown here. Also there does not appear to be any significant differences in the latencies versus the target azimuthal angle.
Dr. James Brown in his work discovered some increases in latencies for

downward angles (225, 270 and 315 degrees). However not all of Dr. Brown's subjects exhibited this phenomena. It appears that subject DEB also does not exhibit this phenomena. The most significant observation concerning the LTCY data is the considerable effect that the RVL (and hence target contrast) has on the data. The LTCY increases markedly below an RVL of about 2. Similar effects of target contrast on saccade latency were also reported by Wheeles (1967). However his work reported only latency values for 6 degree horizontal saccades and two contrast values. Other workers have reported similar target contrast effects on smooth pursuit eye movements, Brown (1972a, b), and Haegerstrom-Portnoy (1979), and on fixational eye movements, Steinman (1965), and Boyce (1967). Haegerstrom-Portnoy (1979) measured the target contrast effects on the saccade latency associated with the beginning of the smooth pursuit eye movement. However, since saccadic movements were not studied directly no attempts were made to correlate the saccade latency for different saccade magniutdes and directions with target contrast. The previous data mentioned above and the latency data reported here indicate the marked effects of target contrast on eye movements (both saccadic and smooth pursuit). Future experimenters should realize these effects and take them into consideration when planning their experiments.

The saccade settling time to displaced targets is a saccade property that has not been reported in the literature to data. The settling time to 15 minutes arc (ST15) data shown here (Fig.'s 16 to 23) also indicate that target contrast has a marked effect on saccadic settling time. As expected from the LTCY data the ST15 data shows asymptotic behavior for RVL values higher than about 2. The ST15 data shows more scatter than the LTCY data indicating the variability in the number of saccades used by the subject's eye to reach the final target position. The asymptotic (or high RVL) values of the ST15 data tend to indicate that the subject requires two saccades to acquire the target much of the time. Asymptotic values of about 400 milliseconds are compatible with "two saccade" durations as reported by other experimenters, for example White and Eason (1962), and Robinson (1964). The ST15 data also shows a trend of longer settling times for more eccentric target radii. That is, the ST15 data for 2.0 degree radii targets tends to fall below that for 4.0 degree radii targets. The slight increase in ST15 data for the 4.0 degree radii targets is to be expected due to the longer distance traveled by the eye. The increased ST15 values for 4.0 degree radii are compatible with saccade duration data reported by White and Eason (1962), Robinson (1964) and Taümer (1976). Another point of interest concerning the ST15 data shown here is that the asymptotic (or high RVL) ST15 values tend to indicate the relative differences in saccade velocities versus target angle. For example, nasal saccades have been found to be slower than temporal saccades by Dodge and Cline (1901), Robinson (1964) and Fricker (1971). The data shown here indicate that the ST15 asymptotic value for 0 degrees (temporal) is perhaps 10 milliseconds higher than the value for 180 degrees (nasal). However the scatter of the 180 degree data makes interpretation somewhat difficult. Taümer (1976) plotted saccadic velocity, acceleration, and duration on polar coordinates to illustrate the changes in these saccade properties versus direction. Taümer's findings indicate that the shape of the polar graph was independent of target radius (i.e. the size of the saccade) and also showed a great deal of intersubject variability. The ST15 and LTCY data given here could be plotted in this manner to indicate differences in these properties versus direction. However, it is apparent from Fig.'s 8 to 23 that the direction effects on ST15 and LTCY are only minor for this particular subject.

Summary

The automated saccade analysis system was utilized to evaluate the effects of target contrast and two-dimensional target position on human saccadic parameters. Subject saccadic eye responses were studied for off-axis radial distances from the ocular position of rest of 2.0, 2.5, 3.0, 3.5, and 4.0 degrees, and azimuthal angles of 0, 45, 90, 135, 180, 225, 270, and 315 degrees. Nine steps in target luminance contrast ranging from the subject's foveal contrast threshold to various magnitudes above foveal contrast threshold (designated RVL) were provided to the subject. The results of these experiments indicate that saccadic latency and settling time increase two to three times as the target contrast falls below an RVL of 2. Only small effects of target position (radius and azimuth) were noted on the saccadic parameters. Further data analysis could perhaps provide insights as to the states of the extraocular muscles responsible for the saccades.

GENERAL CONCLUSIONS

Modifications were successfully performed on the saccade analysis system of Brown to allow for computer control of target parameters and accurate control of target contrast (to within 0.5%). The new and modified software of the saccade analysis system provide computer control of target parameters and additional response data analysis and plotting. The new saccade analysis system provides efficient, accurate, and rapid analysis of saccadic eye movements. The overall system throughput is three minutes per trial (an increase of one minute from Brown's version due almost entirely to the added data analysis program SACCOM). The saccade data presented here concerning the relations between saccadic latency and settling time to target contrast and position are accurate (standard errors less than 10% of mean for latency and settling time) and unique (settling time data unreported in literature to date).

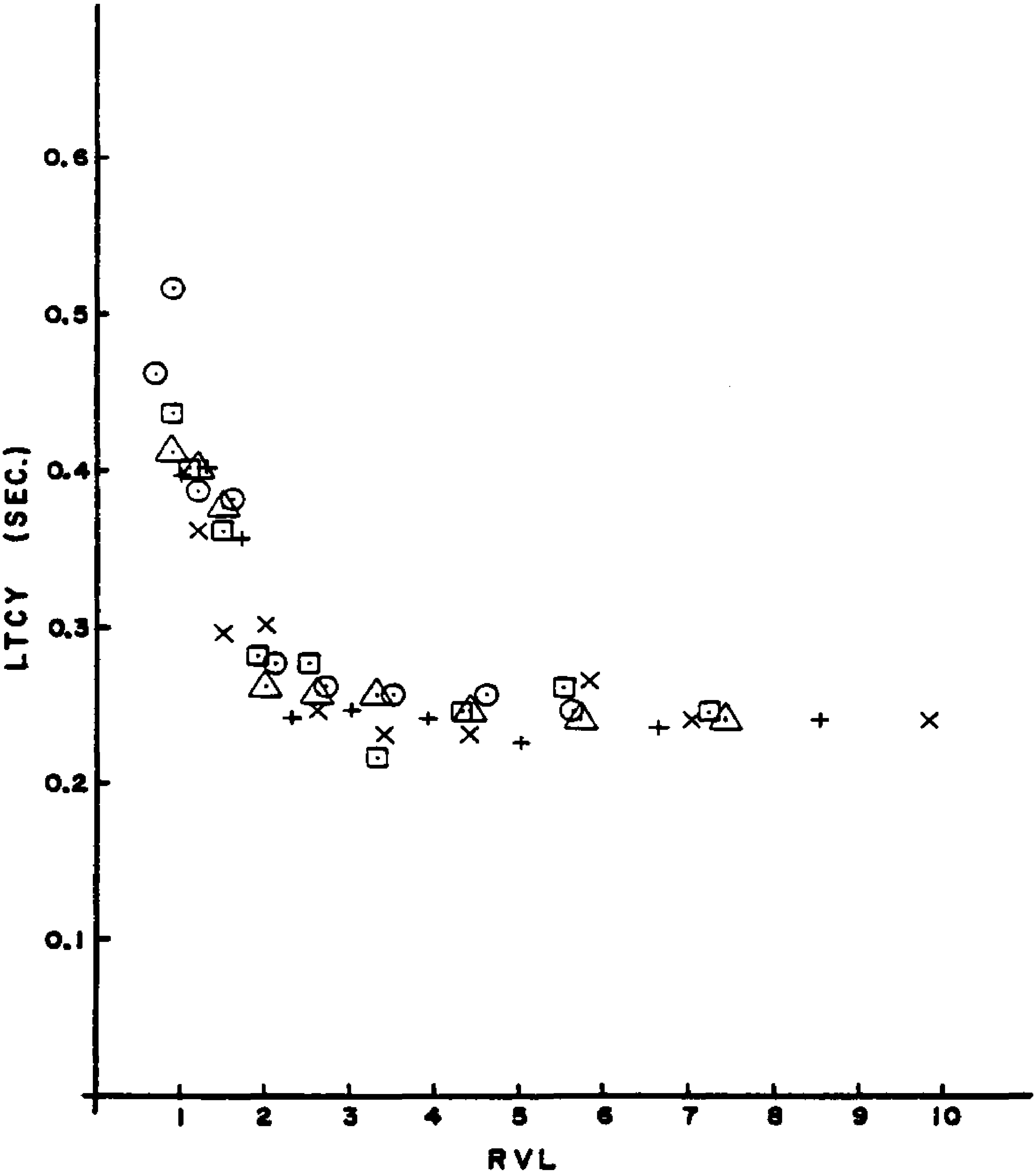


FIG. 8 - Subject DEB latency (LTCY) versus RVL for target angle 0 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

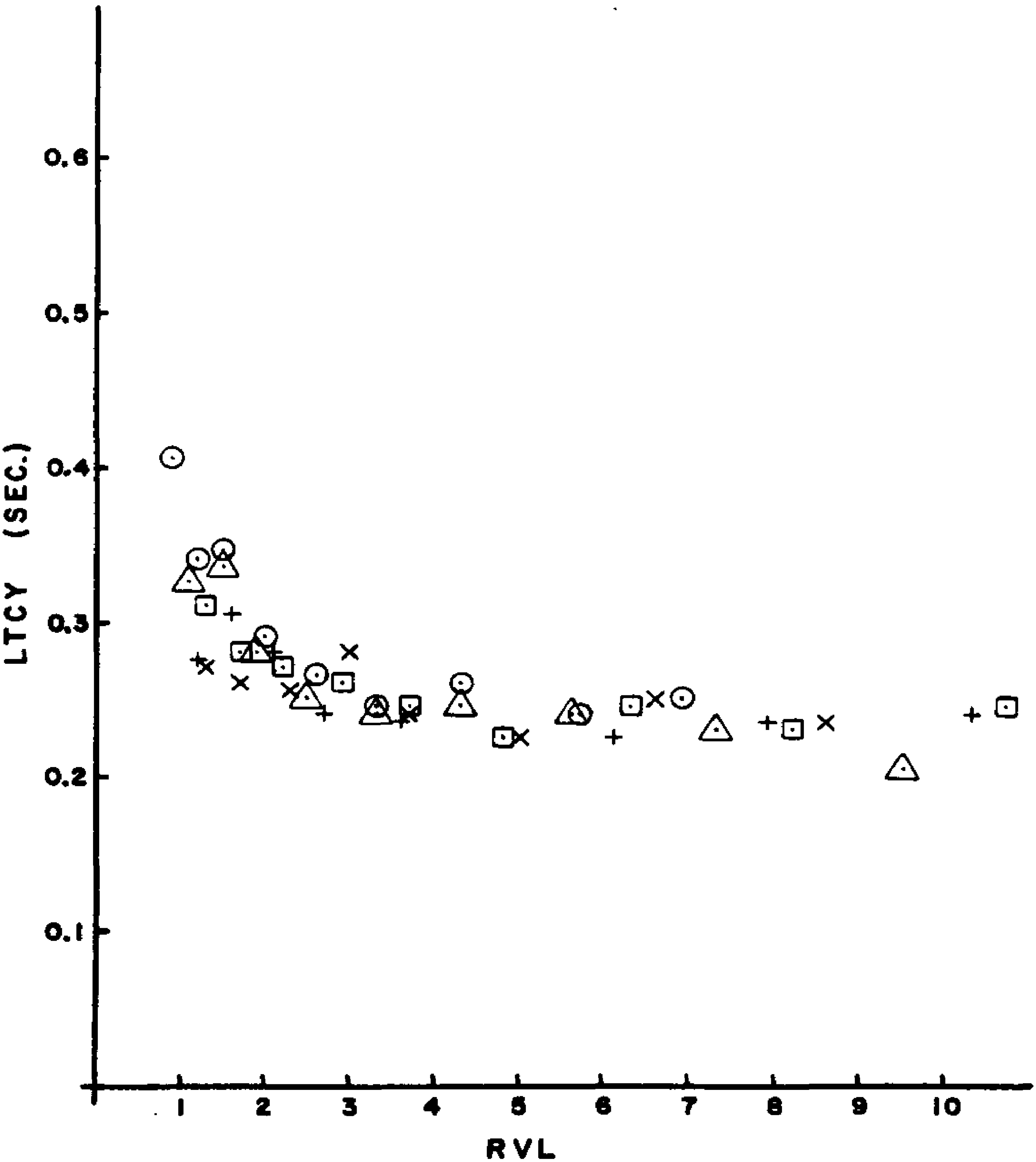


FIG. 9 - Subject DEB latency (LTCY) versus RVL for target angle 45 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

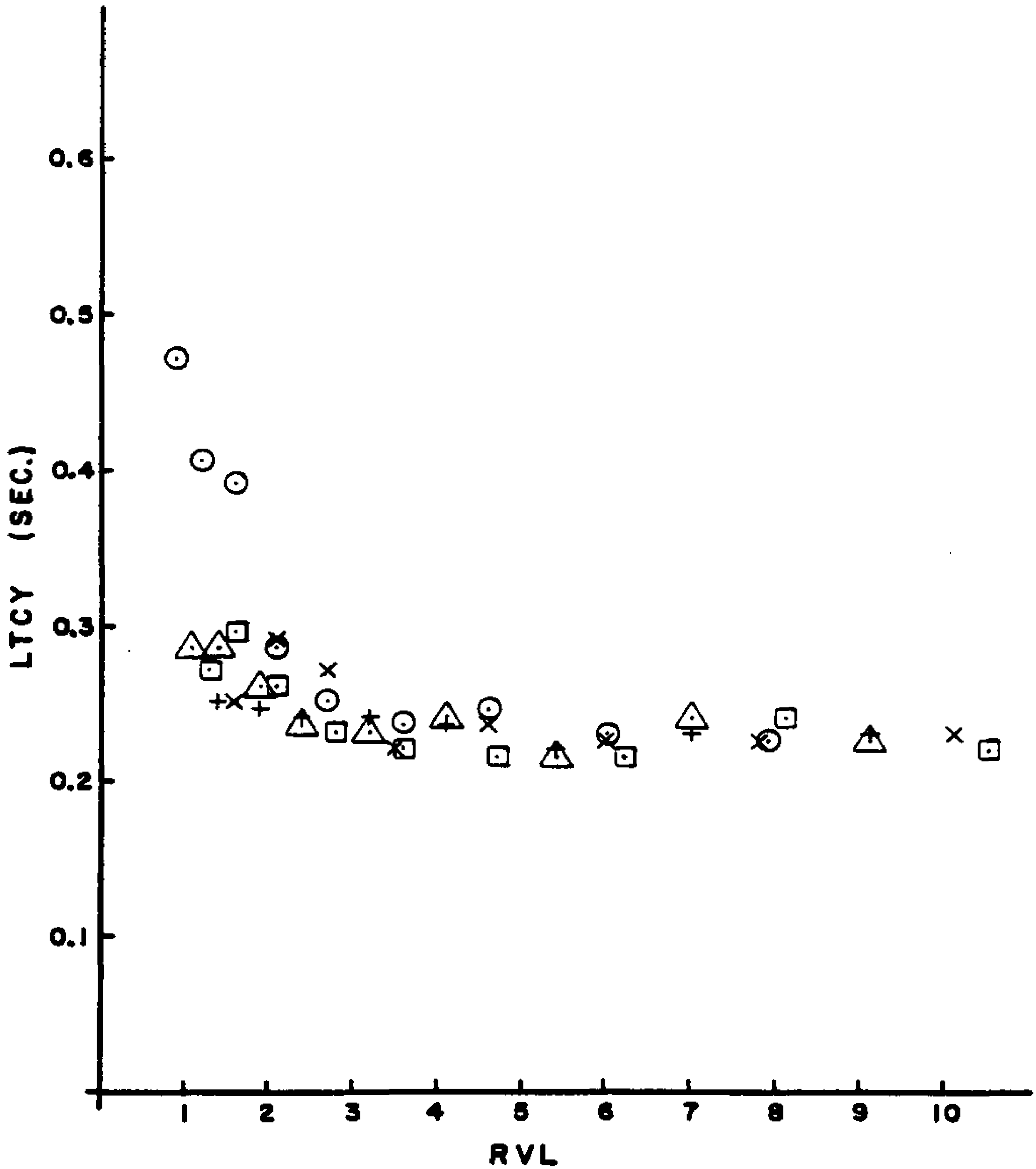


FIG. 10 - Subject DEB latency (LTCY) versus RVL for target angle 90 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

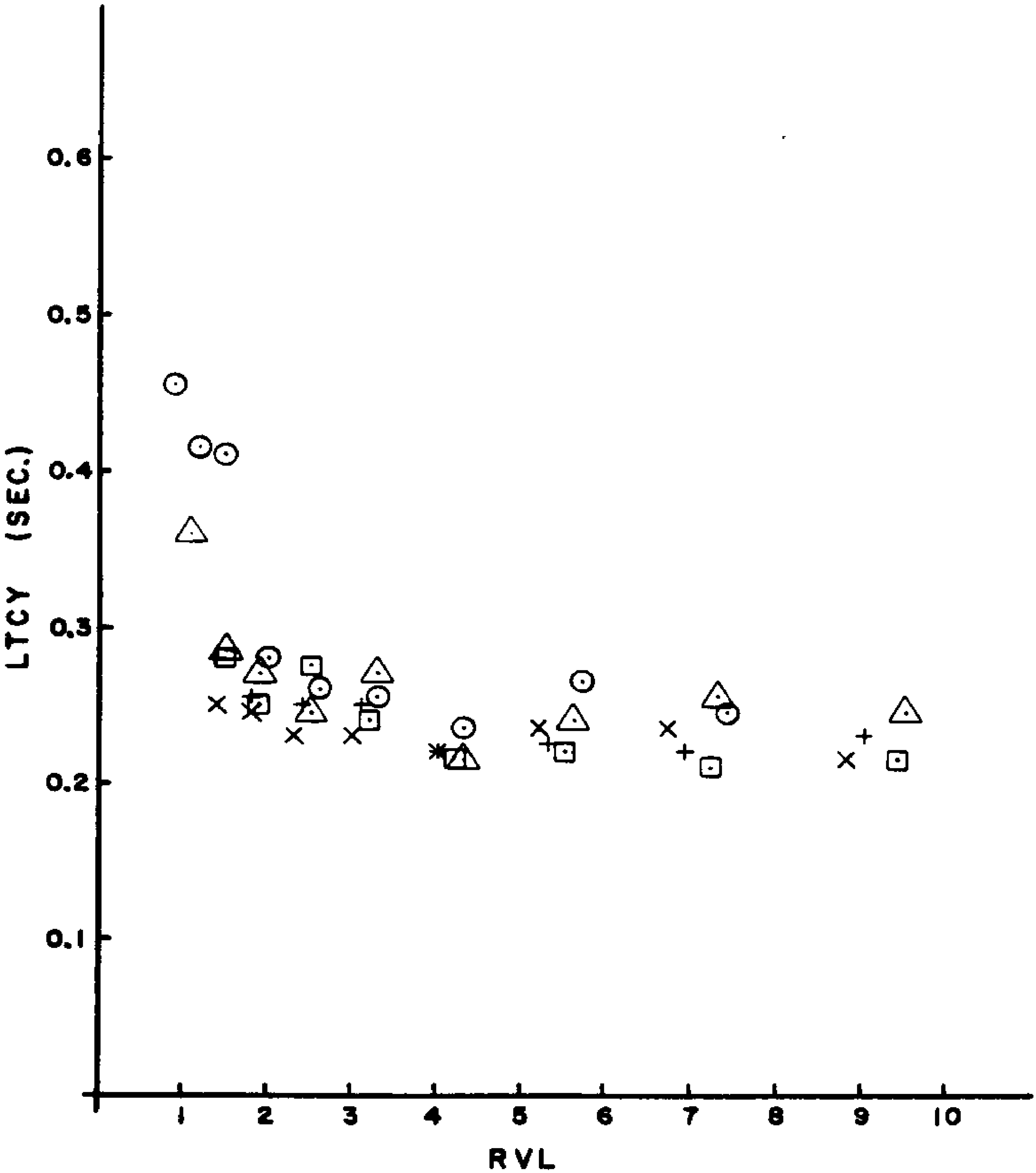


FIG. 11 - Subject DEB latency (LTCY) versus RVL for target angle 135 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

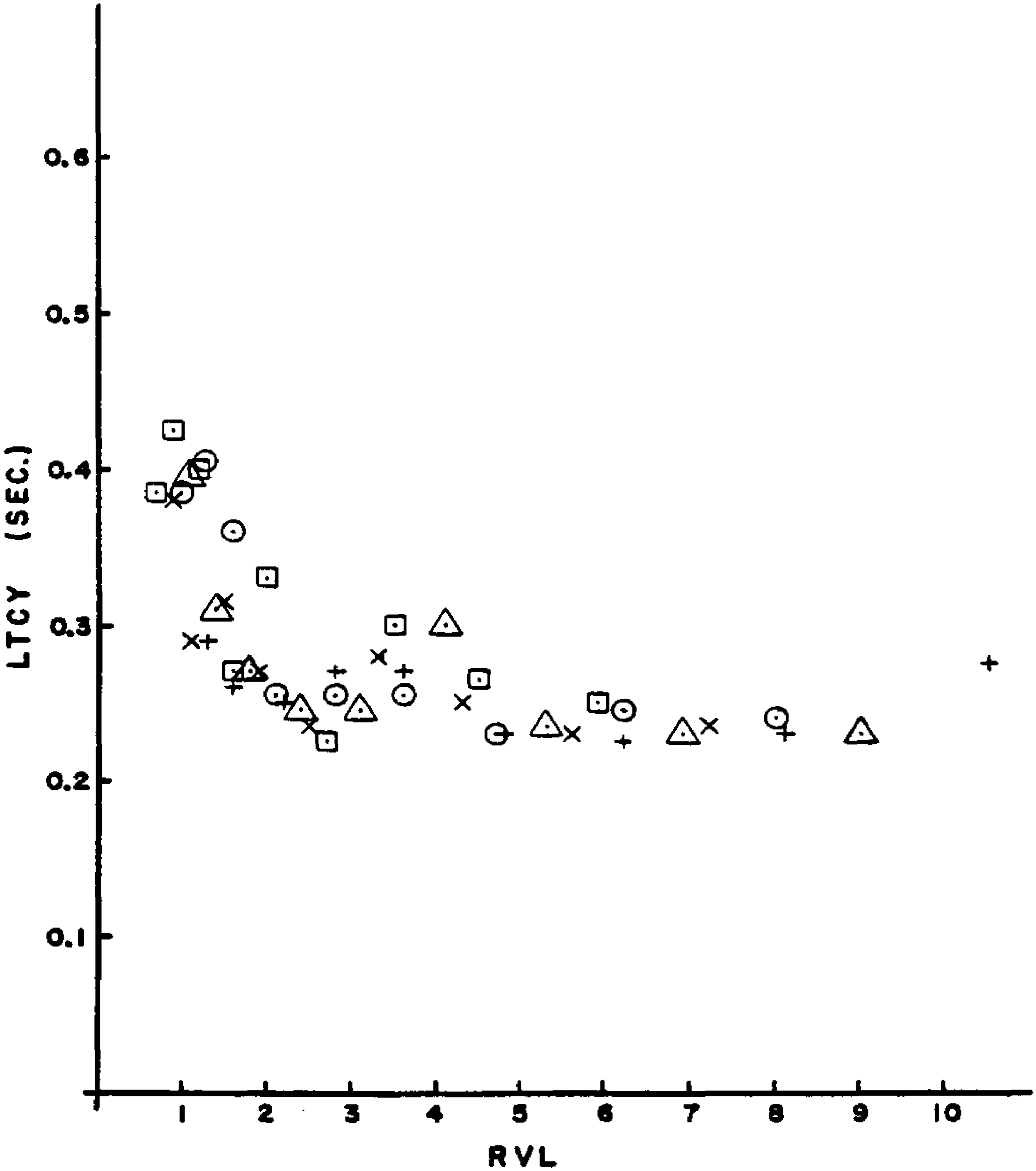


FIG. 12 - Subject DEB latency (LTCY) versus RVL for target angle 180 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

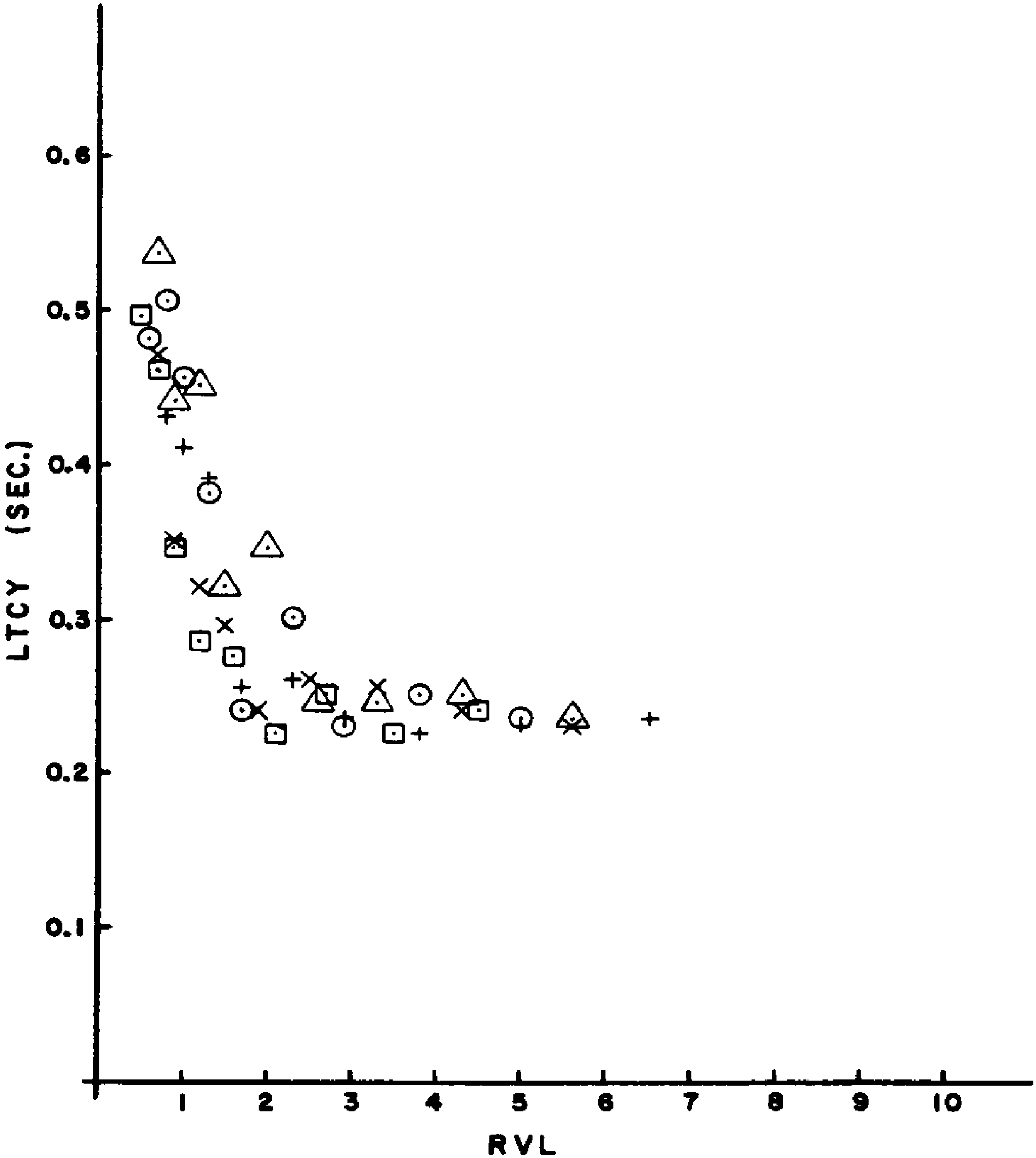


FIG. 13 - Subject DEB latency (LTCY) versus RVL for target angle 225 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

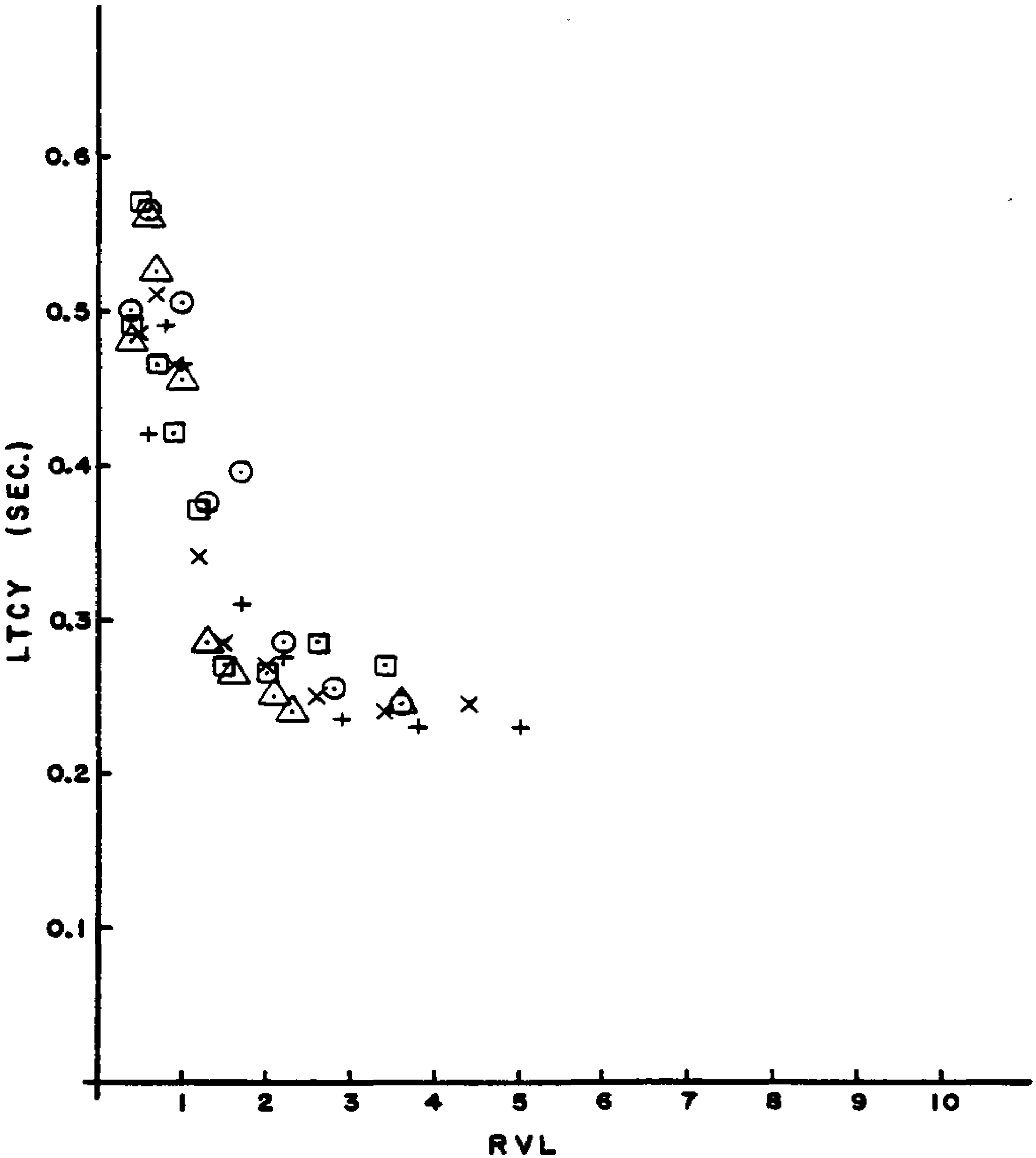


FIG. 14 - Subject DEB latency (LTCY) versus RVL for target angle 270 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

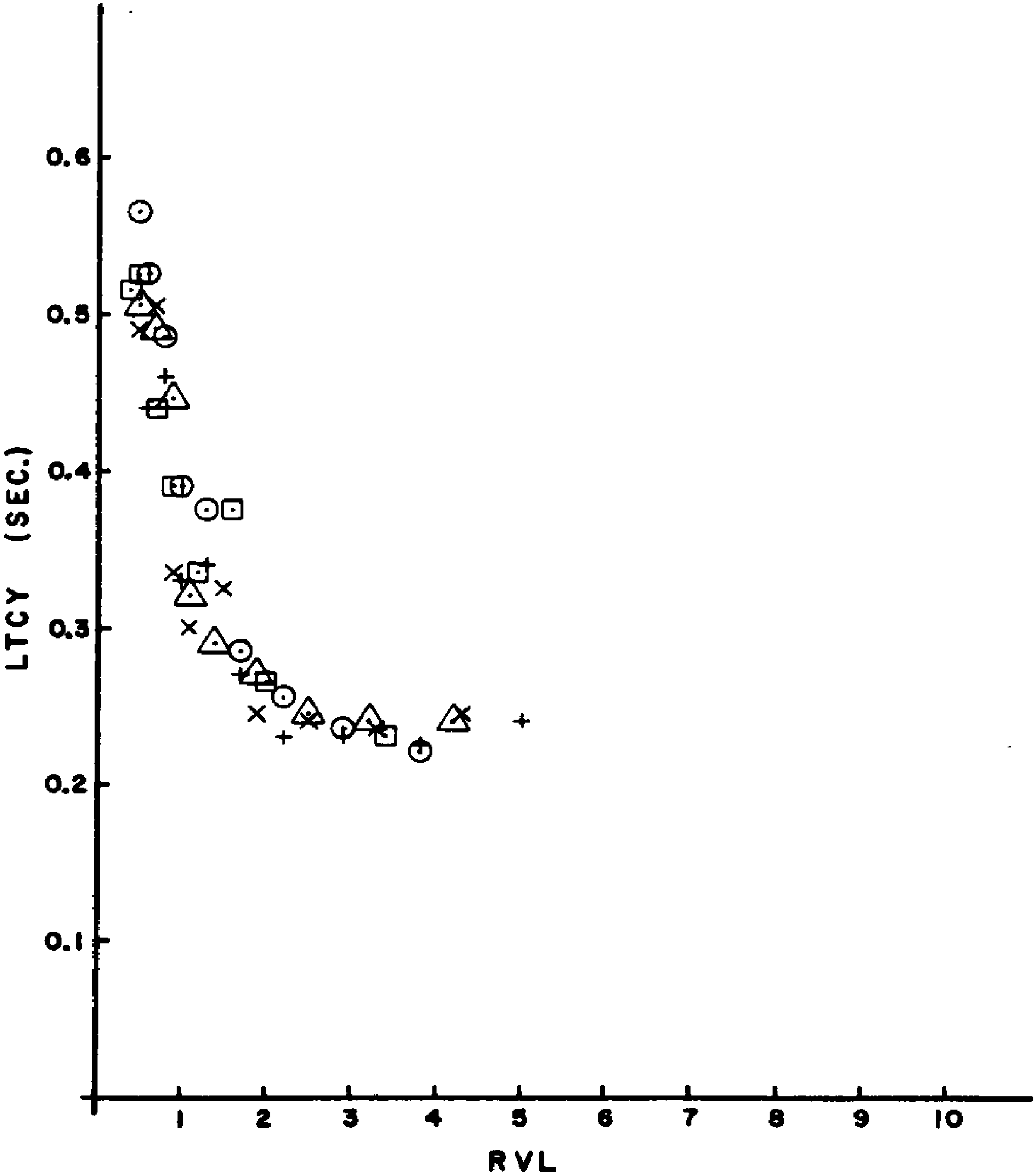


FIG. 15 - Subject DEB latency (LTCY) versus RVL for target angle 315 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

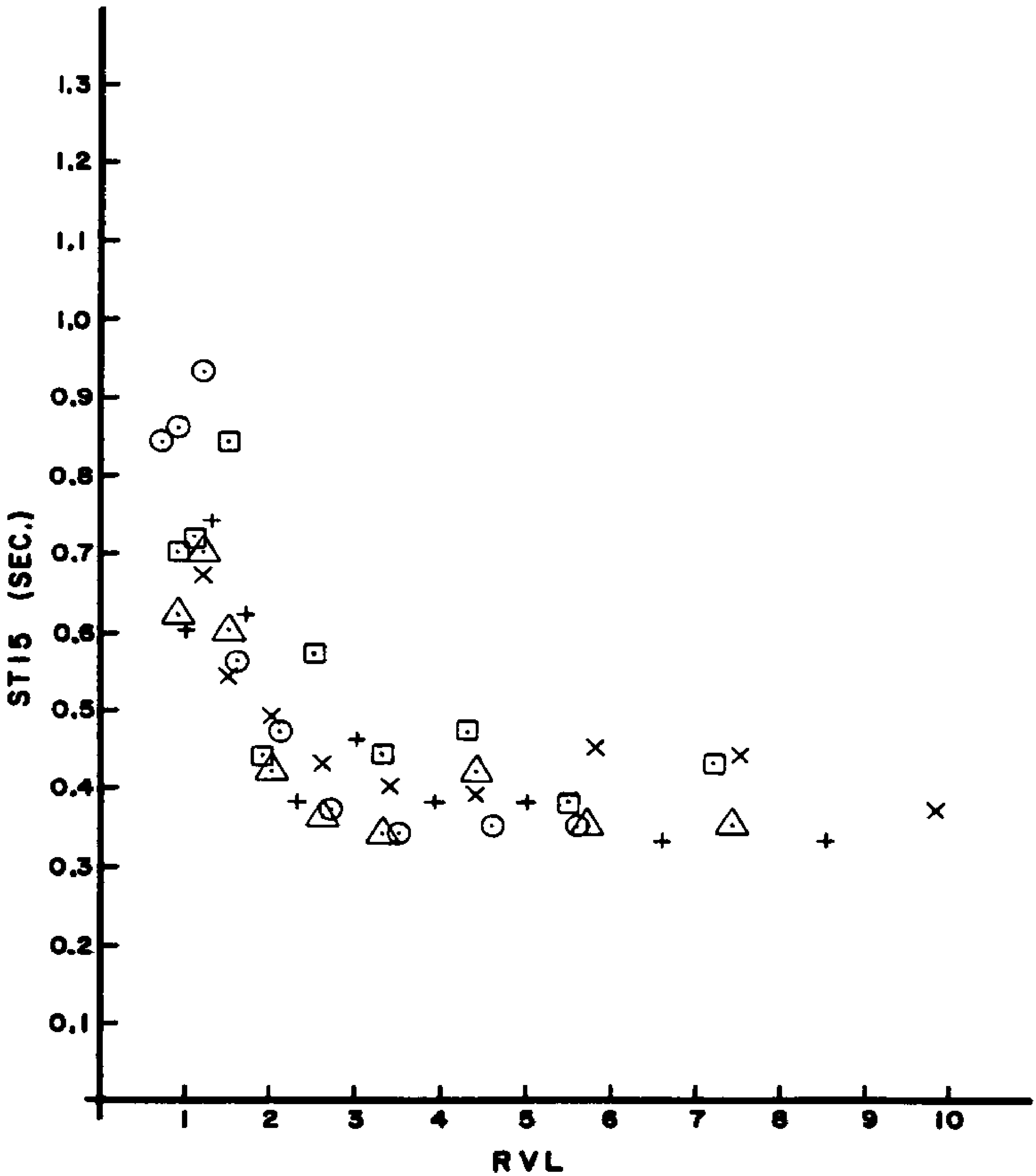


FIG. 16 - Subject DEB settling time (ST15) versus RVL for target angle 0 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

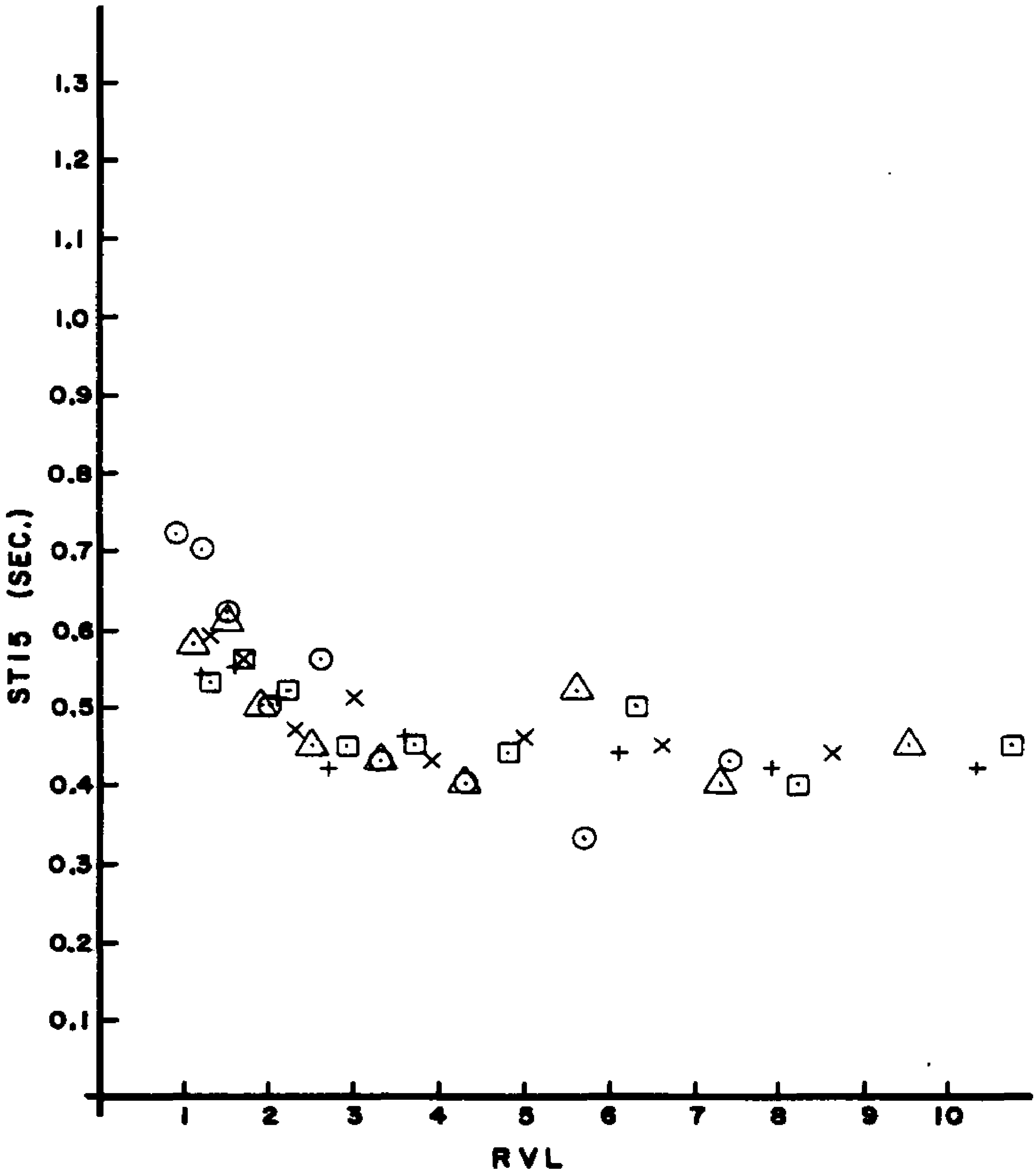


FIG. 17 - Subject DEB settling time (ST15) versus RVL for target angle 45 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

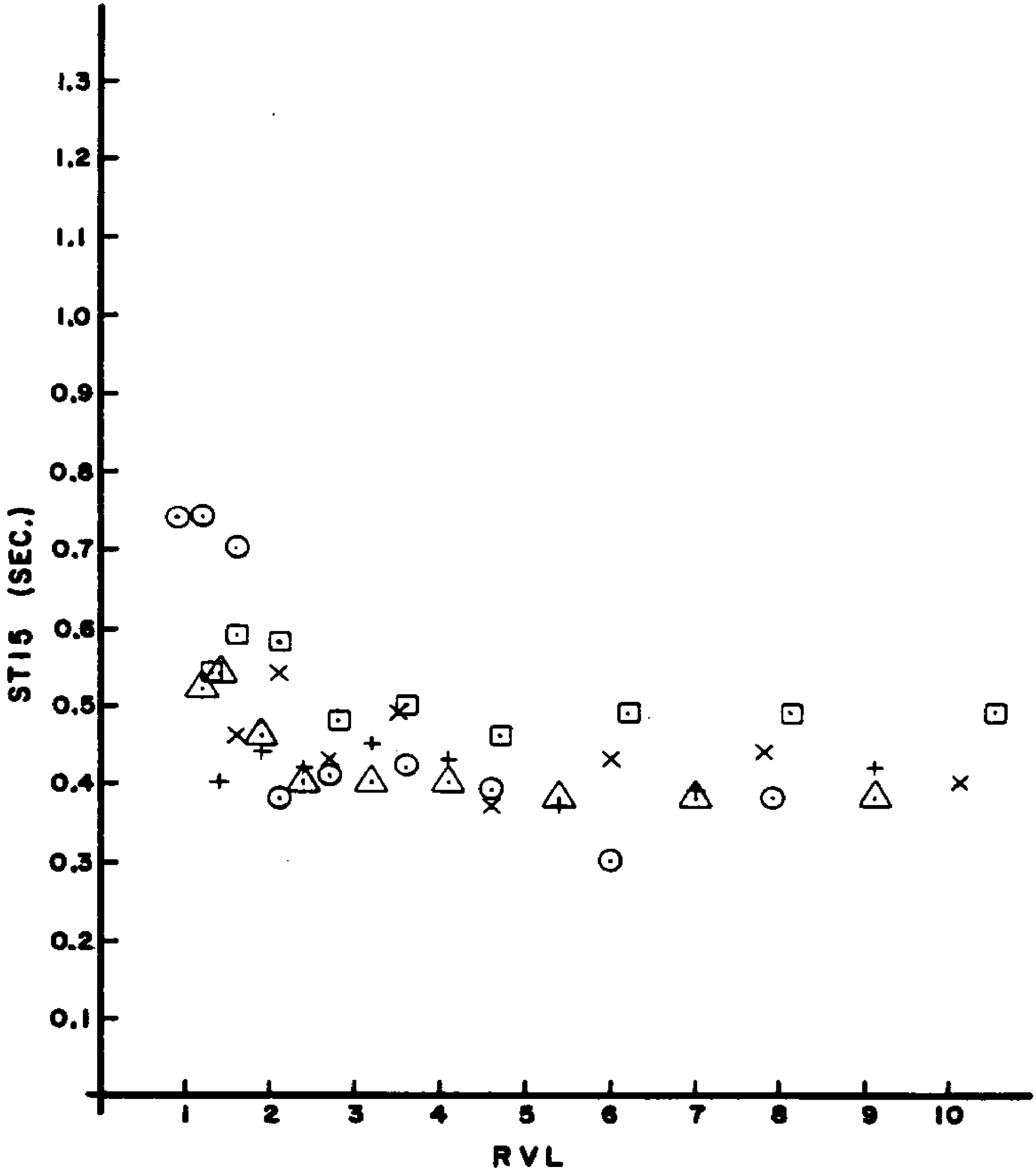


FIG. 18 - Subject DEB settling time (ST15) versus RVL for target angle 90 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

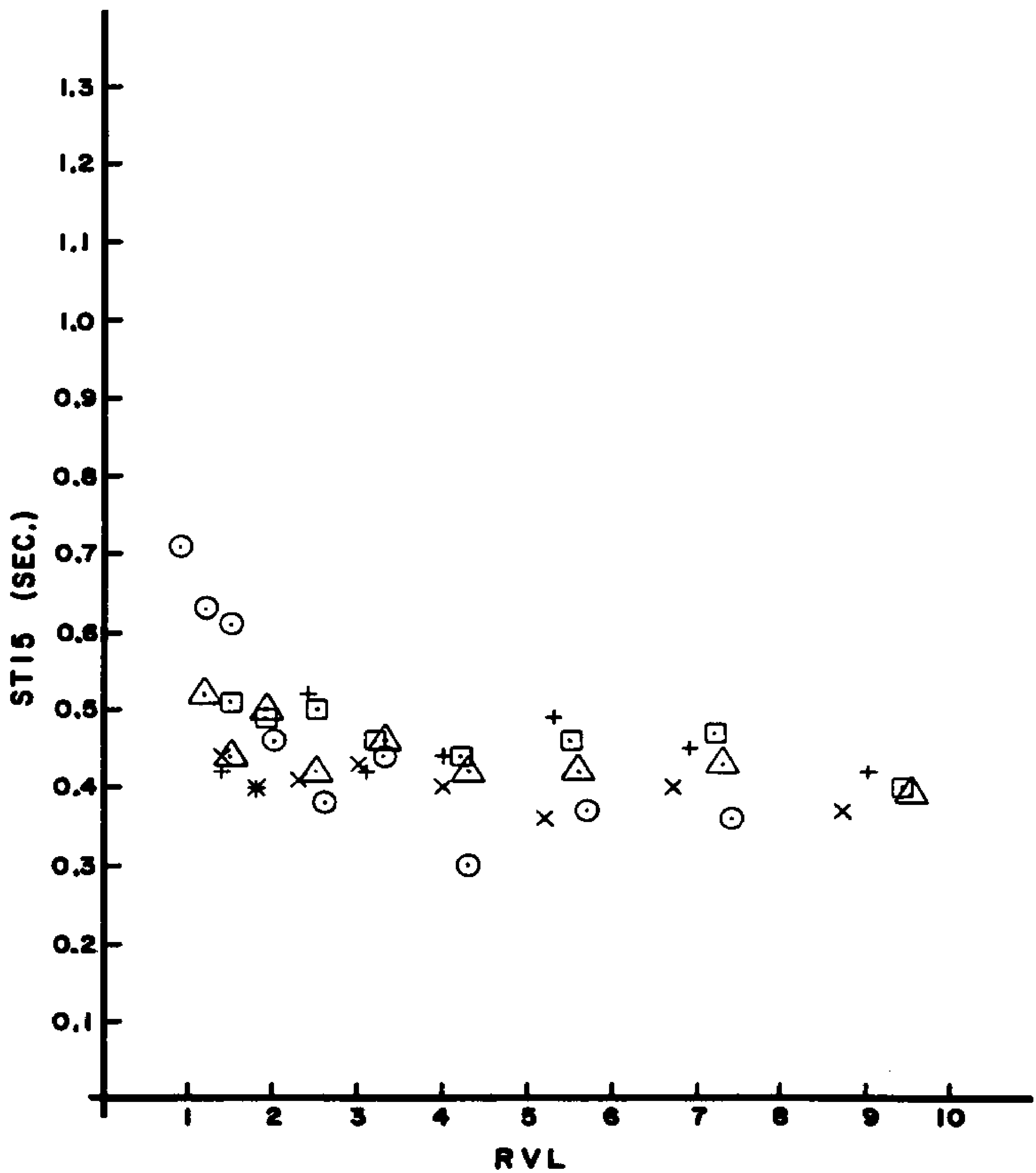


FIG. 19 - Subject DEB settling time (ST15) versus RVL for target angle 135 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

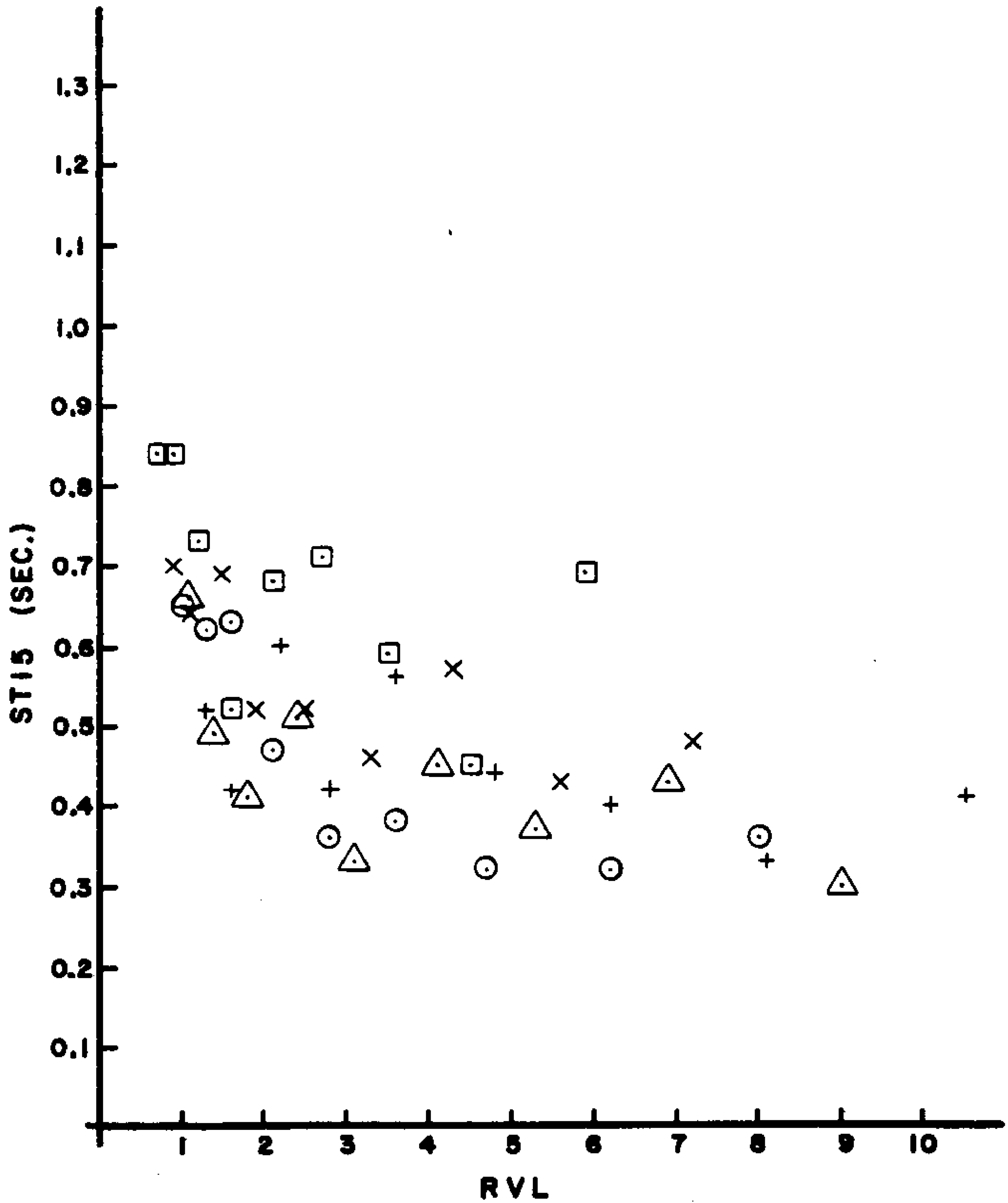


FIG. 20 - Subject DEB settling time (ST15) versus RVL for target angle 180 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

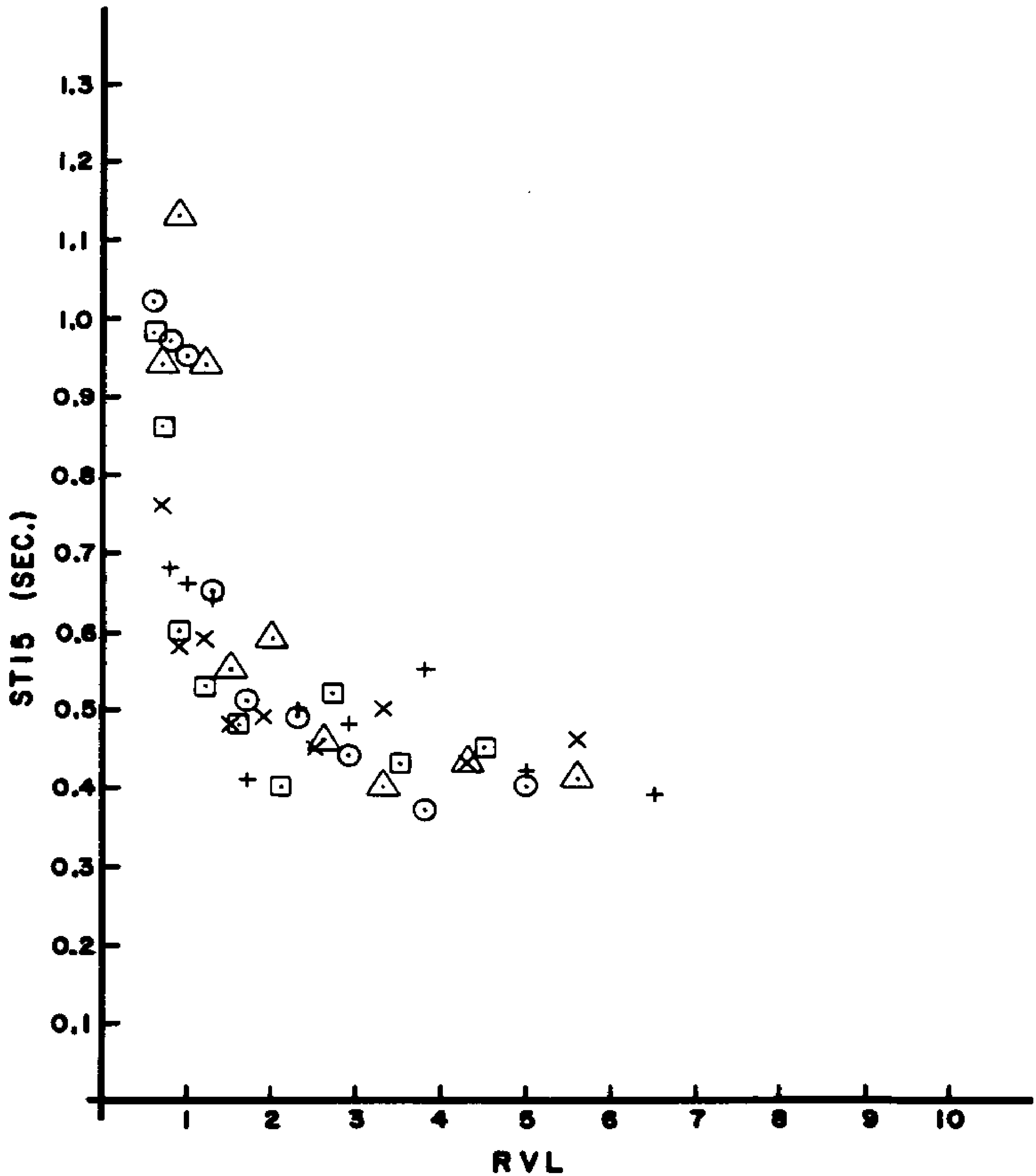


FIG. 21 - Subject DEB settling time (ST15) versus RVL for target angle 225 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

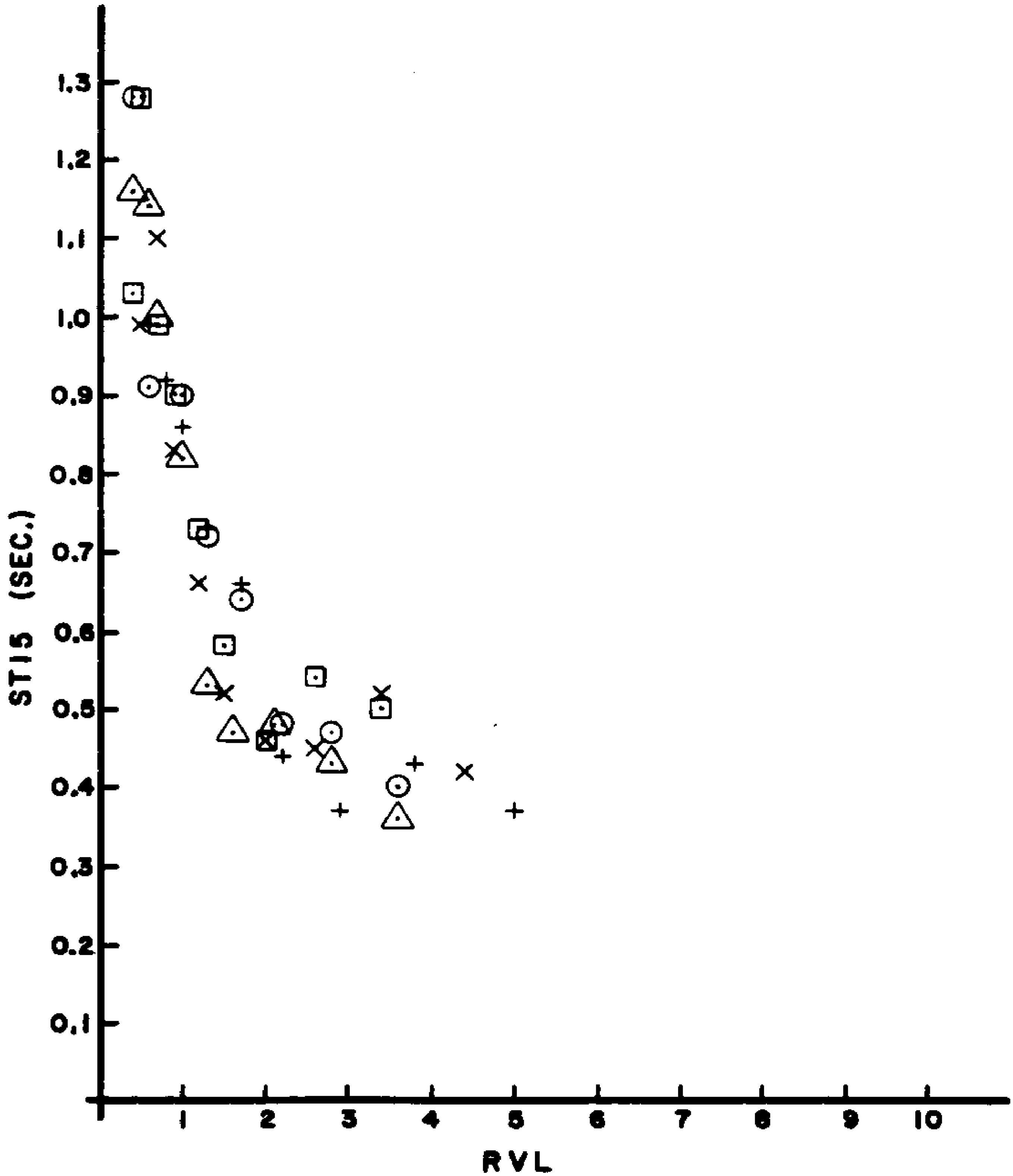


FIG. 22 - Subject DEB settling time (ST15) versus RVL for target angle 270 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

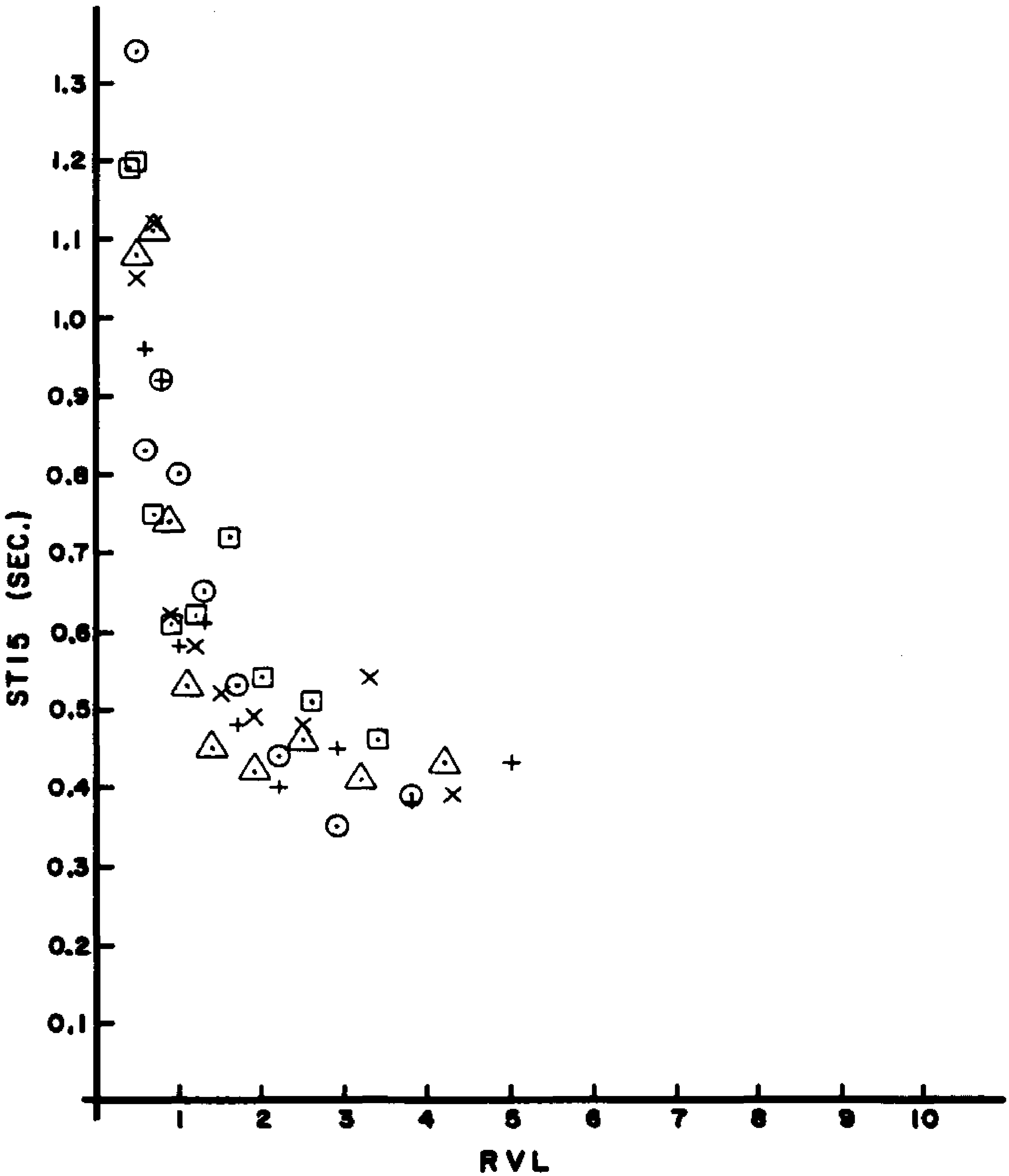


FIG. 23 - Subject DEB settling time (ST15) versus RVL for target angle 315 degrees, radius 2.0 (⊙), 2.5 (△), 3.0 (+), 3.5 (X), and 4.0 (⊡) degrees.

APPENDIX A: TARGET PRESENTATION SYSTEM

Introduction

The target presentation system (TPS) was designed to provide the controlled visual stimulus to subjects used in the study of saccadic eye movements. The TPS is divided into two sections; the visual presentation section (VPS), and the interface electronics section (IES). The VPS allows the user precise and complete control over the visual stimulus presented to the subject. The IES allows some VPS functions to be controlled by the saccade analysis system computer.

For the experiments performed here the complete TPS system was designed to allow for computer selection of one of 40 possible target spot step deflections and one of nine possible target spot contrast neutral density filters. The target spot deflections are divided into eight azimuthal angles from 0 to 315 degrees in 45 degree increments and five radial magnitudes from 2.0 to 4.0 degrees in 0.5 degree increments. The neutral density filters are set to range from the subject's foveal contrast threshold to above the subject's foveal threshold in increments of 0.115 log units density per filter (minimun density available 0.115, maximum density available 1.035, change 0.920 log units). Selection of the target parameters radius, angle, and filter number is via computer software when the TPS is used in its automatic mode. This software allows for the selection of a particular target position or a sequence of pseudo random target positions. This allows the user to present to the subject an unpredictable sequence of target deflections for rapid and efficient saccade response analysis. More details of the software may be found in the following appendix.

The interface electronics section (IES) is designed to provide signals to and accept signals from the PDP-8/E system computer, the VPS, and the Cornsweet-Crane Purkinje image eyetracker. The IES accepts target parameter data from the computer and translates it into the signals necessary to drive the stepper motors on the VPS. The IES provides the computer with: a DEV RDY signal to indicate when the VPS assemblies have been positioned correctly, clock trigger pulses to initiate eyetracker data sampling, and a FAIL data bit to indicate either a subject eyeblink or an eyetracker track loss. The IES accepts signals from the VPS concerning VPS stepper motor positions and VPS shutter signals (used to generate clock trigger pulses).

Visual Presentation Section

The visual presentation section (VPS) is shown schematically in Fig.'s 24 and 25. The VPS is made up of projectors, shutters, filters, a viewing screen, target spot generation masks, a target spot filter assembly, stepper motors, and a control electronics package. The five projectors T, B1, F1, B2, and F2 are used as controlled light sources. Each projector is equipped with photodiodes to monitor the light output of the projector's lamp. The control electronics of the VPS has controls and digital indicators to vary

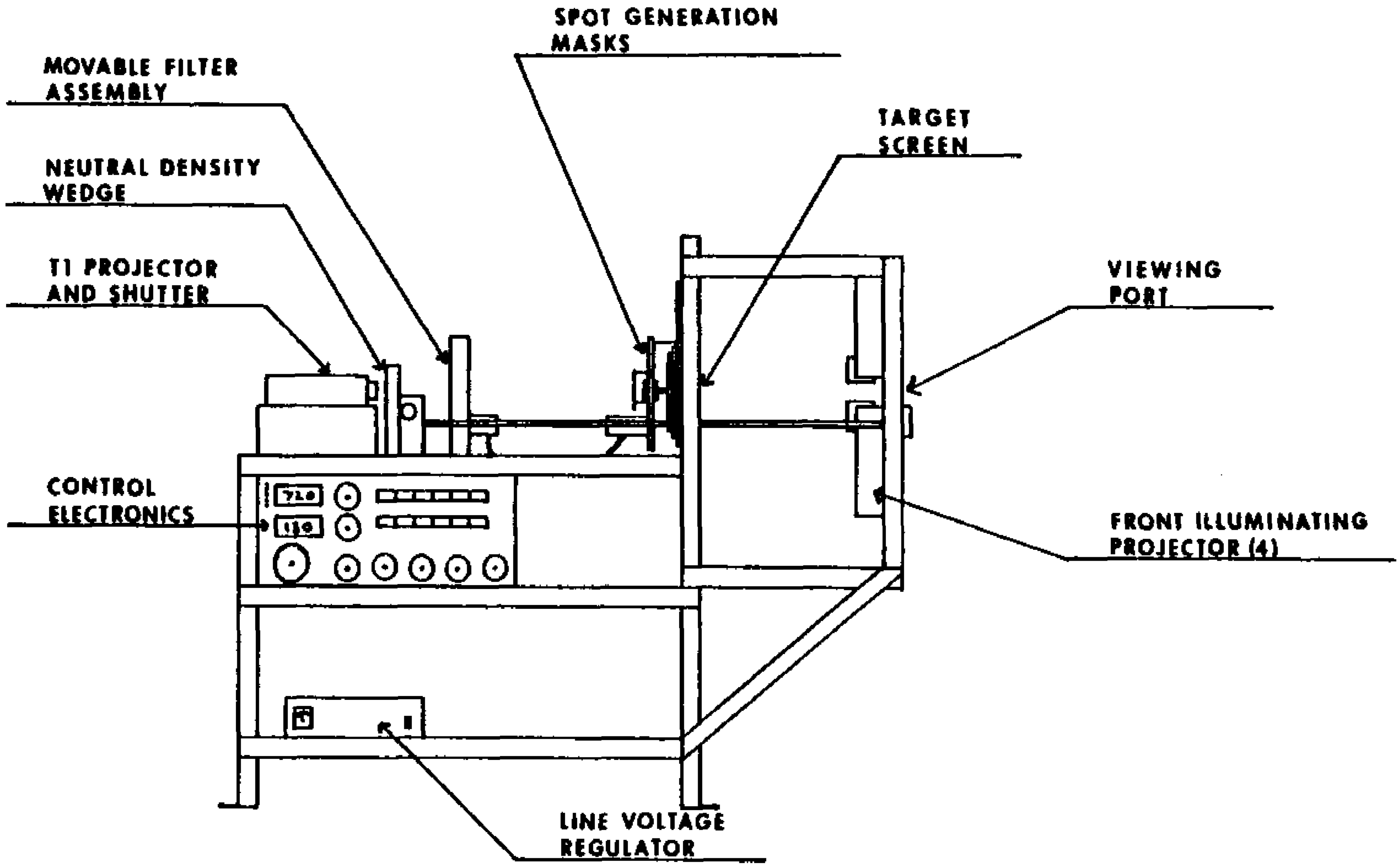


FIG. 24 - SCHEMATIC REPRESENTATION OF VISUAL PRESENTATION SECTION (SIDE VIEW) OF TARGET PRESENTATION SYSTEM

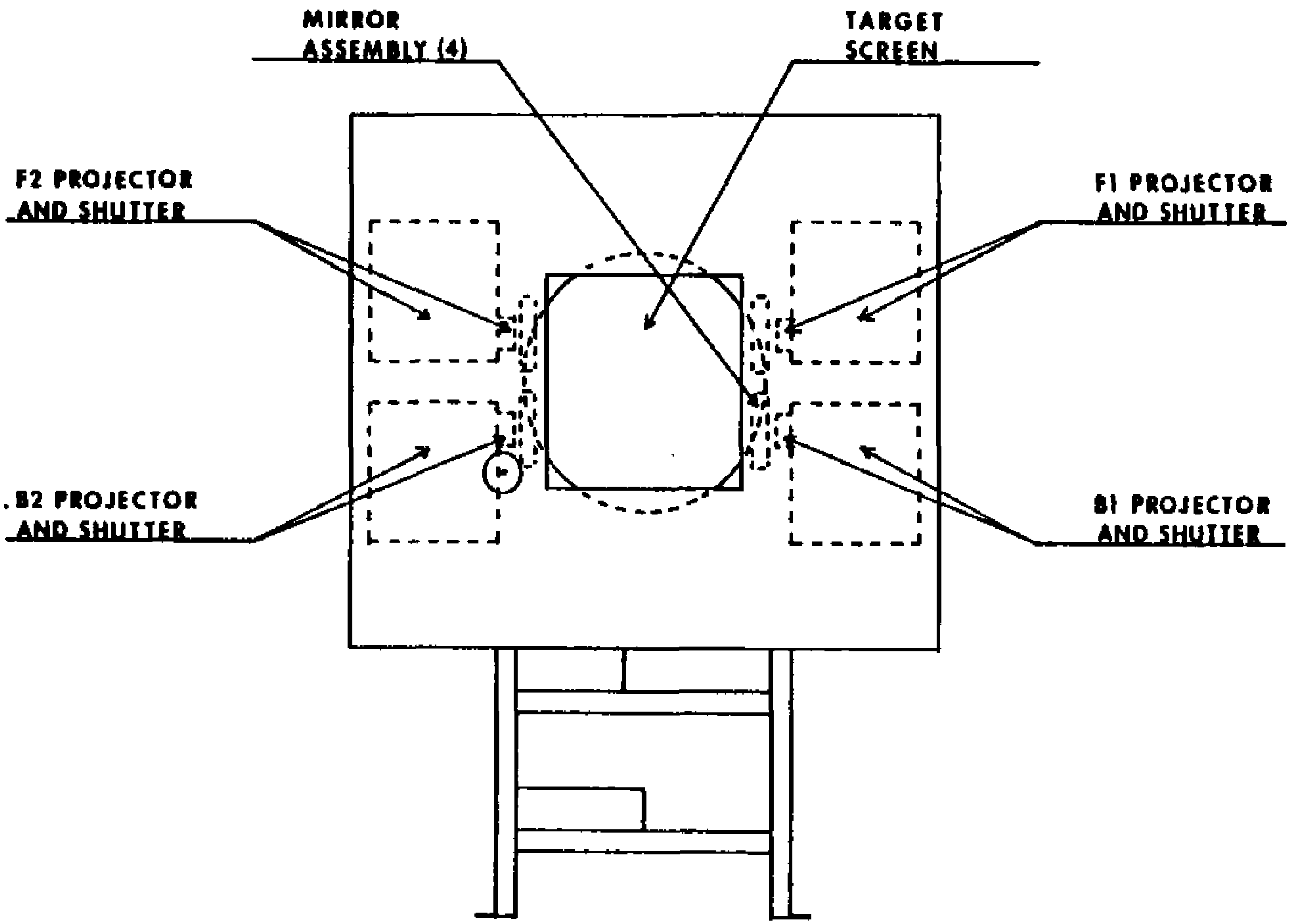


FIG. 25 - SCHEMATIC REPRESENTATION OF VISUAL PRESENTATION SECTION (FRONT VIEW) OF TARGET PRESENTATION SYSTEM

and monitor the light levels of the five projectors independently. In this manner the user can maintain a highly constant light output from each projector. This feature is most important for evaluating subjects' saccadic responses to varing target contrast levels. To aid in maintaining a constant light output from the projectors the VPS control electronics may be connected to a line voltage regulator (this was the case for all experiments performed here). By using projectors as light sources the user gains the advantage of being able to insert slides into the projectors. This allows the user a great flexibility in the visual stimuli presented to the subject.

As indicated in Fig.'s 24 and 25 projector T is used to back illuminate the target spot generation masks and filters while the other four projectors (F1, B1, F2, and B2) are used to front illuminate the viewing screen. For all experiments performed here VPS projector usage was as follows: (1) The T projector was used to back illuminate the spot generation masks through the movable filter assembly. (2) The F2 projector was used to provide a central fixation spot centered on the front of the viewing screen. (3) The B2 projector was used to provide a fixation array of four spots (at 0, 90, 180, and 270 degrees azimuth, 1.0 degrees radius) around the reference position target spot (radius 2.0 degrees, azimuth 0 degrees). This arrangement was used during the determination of the subject's foveal contrast threshold. (4) The B1 projector was used to provide a constant background illumination on the front of the viewing screen. (5) The F1 projector was used to provide a fixation array of four spots (as described for the B2 projector) around the central fixation spot produced by the F2 projector. This arrangement was used to calibrate the eyetracker H4 and V4 output signals at 0.2 volts per degree of eye rotation for all experiments done here.

Associated with each projector of the VPS is a shutter which is placed so as to chop the output light beam of the projector. The VPS control electronics has controls which allow the user to: (1) independently open or close each of the five shutters, or (2) have certain shutters (T and F2) undergo preprogrammed openings and closings. The two preprogrammed shutter activations of the T and F2 shutters used in the experiments performed here were PT and SP. The PT shutter control mode is one in which the T projector shutter is continuously opened for 0.2 second then closed for 1.0 second. This action coupled with the spot generation masks provides the subject with a flashing target spot on the viewing screen. The PT mode was used in conjunction with the B2 projector fixation array and the reference target spot location to evaluate the subject's foveal contrast threshold for the experiments done here. The SP shutter control mode is one which exchanges the T and F2 shutter positions. That is, in the SP mode the F2 shutter is initially open and the T shutter is initially closed. When the user activates the VPS "GO" pushbutton a warning tone sounds after which the F2 shutter closes and the T shutter opens. After a specified length of time the T shutter closes again and the F2 shutter opens again. The VPS control electronics has controls which allow the user to smoothly adjust the length of time that the warning tone sounds and the length of time that the T shutter stays open. For all experiments performed here the warning tone time was adjusted so that the warning tone sounded for approximately 1.5 seconds and the T shutter time was adjusted so that the T shutter stayed open for about 3 seconds. In subsequent discussions the warning tone time (or delay time) is given the name T1 and the T shutter time (or target deflection time) is given the name T2. The actions of the SP shutter mode coupled with the F2 projector central fixation spot and the target spot generation masks provides the subject with a visual stimulus in which the central fixation spot appears to move or be "deflected" to one of the 40 possible target spot locations. This action generates the target "deflections" referred to in subsequent discussions. This mode of operation was used in the experiments performed here to evaluate the subject's saccadic eye response to step-like target deflections.

The 40 target spots viewed by the subject are generated by using spot generation masks. These masks consist of three metal plates with holes drilled at specified locations, see Fig. 26. One plate, the master mask, has one hole drilled at each of the 40 possible target locations. The size of these holes is fixed at 2.5 millimeters diameter so that when the subject is placed one meter from the viewing screen the spot subtends an angle of 8.5 minutes of arc at the subject's eye. The radial locations of the holes are such that when the subject views the viewing screen from one meter distance

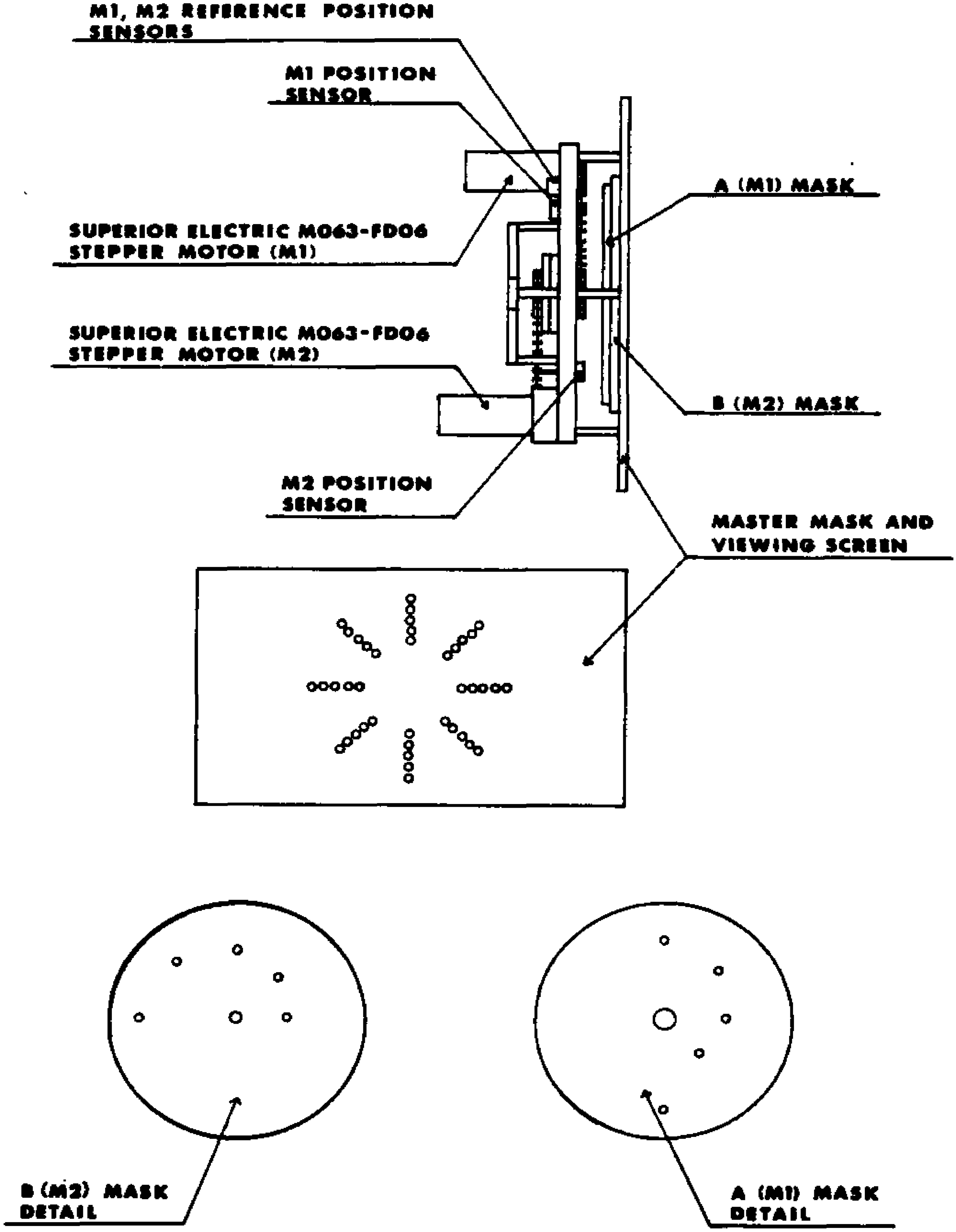


FIG. 26 - Detail of Visual Presentation Section target spot generation masks (M1 = MOTOR 1, M2 = MOTOR 2).

the target spots will subtend radial angles of 2.0, 2.5, 3.0, 3.5, and 4.0 degrees (at the subject's eye) from the central fixation spot. Along one side of this master mask is stretched a white plastic material which serves as the viewing screen as seen by the subject. The other two metal plates are pressed against the opposite surface of this master mask. These other two metal masks are disks which are mounted on concentric shafts and connected via gear and belt mechanisms to two stepper motors, one motor for each disk. These two disks are thus rotatable and each had five holes drilled in them as shown in Fig. 26. These two disks can be independently rotated. When the two disks are rotated in a prescribed relation to one another they will selectively uncover one of the 40 holes in the master mask. When the T projector shutter is opened the light from the projector will shine through the uncovered hole in the master mask and produce a target spot on the viewing screen as viewed by the subject. When the stepper motors are connected via the IES of the TPS to the computer, the computer can then control the positions of the two stepper motors and consequently which one of the 40 possible target spots is presented to the subject.

Discrete variations in target spot contrast are provided for by using nine neutral density filters mounted in a sliding assembly as shown in Fig. 27. This sliding assembly is connected to a stepper motor as shown in Fig. 27. The nine neutral density filters are arranged so as to provide equal step increments of 0.115 log units in density from the least dense to the most dense. The positioning

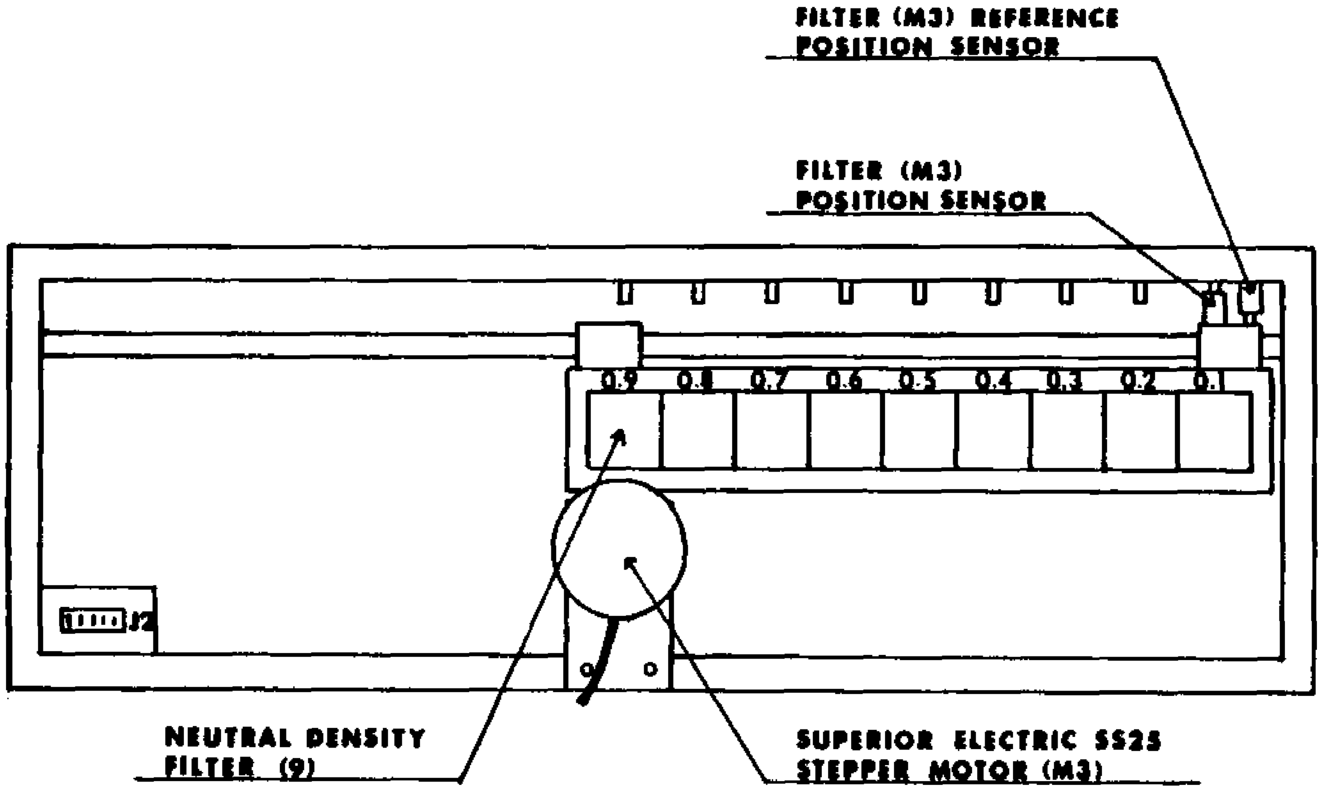


FIG. 27 - Detail of Visual Presentation Section movable filter assembly (M3 = MOTOR 3).

of the entire movable filter assembly is such that the T projector light beam shines through one of the filters before impinging on the rear surface of the target spot generation masks. By connecting the filter assembly stepper motor to the system computer via the IES of the TPS the computer can then control which of these nine filters intercepts the T projector light beam. This provides computer control of the target spot contrast as seen by the subject. This coupled with the computer control of the target spot location as described above provides the user with full computer control over the test stimuli presented to the subject during experimentation.

A neutral density wedge filter is also positioned between the T projector and the target movable filter assembly to intercept the T projector light beam; as shown in Fig. 24. This wedge filter is connected to a mechanical assembly so that the subject or the experimenter can adjust the setting of the wedge. The adjustable wedge was used in the experiments done here in conjunction with the PT shutter control mode and the B2 projector as described earlier to evaluate the subject's foveal contrast threshold. In the experiments done here the subject would adjust the wedge setting to correspond to his foveal contrast threshold. This setting would remain on the wedge throughout the experiment. This wedge setting in conjunction with the computer selectable filter described above provided the subject with discrete changes in target spot contrast that varied from his foveal threshold to above his foveal threshold in equal logrithmic increments.

Connected to the spot generation masks assembly and the movable filter assembly of the VPS are optoelectronic detectors. These detectors provide signals to the IES concerning the positions of the mechanical assemblies of the VPS. At the beginning of any experimental secession the mechanical target parameter assemblies described above can be positioned automatically to a reference position of target location 2.0 degrees radius, 0 degrees azimuth, filter number 0 (the most dense) by actuating the "REF POS" pushbutton on the IES front panel. The control/sampling software package ADCOM used in experiments done here specifically requested the user to actuate the REF POS pushbutton of the IES before beginning any target parameter selections. The software then could calculate the VPS target assembly motor movements required based on this reference position.

Certain electronics were added to the VPS control electronics to provide the IES with the T1 and T2 shutter actuation signals. Circuit diagrams for this circuitry and the diagrams of the position sensors for the VPS target parameter assemblies are presented following the sections listing VPS specifications, controls, and operating instructions.

Specifications, Inputs and Outputs

Deflection radii:	Five radial positions subtending angles of 2.0, 2.5, 3.0, 3.5 and 4.0 degrees from the central fixation spot when viewed from one meter.
Deflection angles:	Eight angles, 0 to 315 degrees in 45 degree increments. 0 degrees is referenced to viewer's right, 90 degrees referenced directly up from center fixation spot.
Target spot size:	Spot size is 2.5 millimeters diameter so as to subtend 8.5 minutes of arc at the viewer's eye when viewed from 1 meter.
Delay time T1 before target deflection:	Smoothly adjustable from 0 to 5 seconds.
Target deflection time T2:	Smoothly adjustable from 0 to 10 seconds.
Projector light levels:	Smoothly adjustable over convenient ranges.
Spot generation mask stepper motors:	Two 200 step per revolution, 1000 step per second stepper motors (Superior Electric Company type M063-FD06) one for each rotatable disk of the spot generation mask. A mask driven by MOTOR 1, B mask driven by MOTOR 2.
Filter assembly stepper motor:	One 200 step per revolution 200 step per second stepper motor (Superior Electric Company type SS25). Filter assembly motor designated MOTOR 3.
+5V:	+5 volt supply voltage, 1 ampere total supply capability supplied by IES and appearing at pin 1 of J1. Used to supply position sensors of the VPS target mechanical assemblies.

LG GND: Logic ground from IES, appears at pins 12, 13 and 14 of J1. This is the ground reference for all position sensor signals appearing on connector J1 and for the T1 and T2 signals of pins 2 and 3 on connector J1.

High level motor pulses: These are the 24 volt, high current pulses from the IES which drive VPS stepper motors MOTOR 1, MOTOR 2 and MOTOR 3. The J3 and J4 connections are as follows:

J3 pin	Signal Name	Data
1	M1M1	MOTOR 1 winding 1
2	M1M3	MOTOR 1 winding 3
3	M1M2	MOTOR 1 gnd. ret.
4	M1M5	MOTOR 1 winding 5
5	M1M4	MOTOR 1 winding 4
6	M1M6	MOTOR 1 gnd. ret.
7	M2M1	MOTOR 2 winding 1
8	M2M3	MOTOR 2 winding 3
9	M2M2	MOTOR 2 gnd. ret.
10	M2M5	MOTOR 2 winding 5
11	M2M4	MOTOR 2 winding 4
12	M2M6	MOTOR 2 gnd. ret.

J4 pin	Signal Name	Data
1	M3W1	MOTOR 3 winding 1
2	M3GND	MOTOR 3 gnd. ret.
3	M3W2	MOTOR 3 winding 2

Data outputs J1 connector: J1 is a 14 pin Cinch-Jones ribbon connector carrying data from the VPS to the IES. Connections are as follows:

J1 pin	Signal name	Data
2	T1	T1 delay time
3	T2	T2 deflection time
4	M1 REF POS SEN	MOTOR 1 ref. position sense
5	M2 REF POS SEN	MOTOR 2 ref. position sense
6	M3 REF POS SEN	MOTOR 3 ref. position sense
7	M3 POS SEN	MOTOR 3 position sense
10	M1 POS SEN	MOTOR 1 position sense
11	M2 POS SEN	MOTOR 2 position sense

All position sensor signals are the output of the collector of the phototransistor located in the sensor. T1 is a TTL level output, high during delay time T1. T2 is a TTL level output, high during target deflection time T2.

Controls and Indicators

LIGHT LEVEL ADJ:	Five front panel controls, one for each VPS projector, used to adjust the light output of the projector. Labeled T, F1, F2, B1 and B2.
T1 ADJ:	Front panel control. When a target deflection is initiated by GO pushbutton actuation a delay to T1 seconds occurs before target deflection occurs. A warning tone sounds during the T1 delay time. T1 ADJ is used to allow adjustment of T1 delay from 0 to 5 seconds.
T2 ADJ:	Front panel control used to adjust the target deflection time from 0 to 10 seconds. When target deflection is initiated by GO pushbutton actuation a delay of T1 seconds occurs after which the target deflects for T2 seconds before returning to its original position.
METER SELECT:	Six position front panel rotary switch used to select which projector light level is monitored by the front panel meters.
SHUTTER CONTROL:	Five front panel pushbuttons, one for each projector shutter, used to open or close the respective projector shutter. When a projector shutter is in its closed position it can be activated by one of the automatic control modes.
PT:	Front panel pushbutton used to select the PT shutter control mode. When activated PT mode causes the T shutter to open for 0.2 second then close for 1.0 second.
SP:	Front panel pushbutton used to select the SP shutter control mode. When activated SP mode causes the T and F2 shutters to exchange positions under control of the GO pushbutton.

When SP mode is activated the F2 shutter is initially open and the T shutter is initially closed. When the GO pushbutton is actuated the F2 shutter closes and the T shutter opens synchronously after the T1 delay time. The T shutter remains open for the deflection time T2 after which the T shutter closes synchronously with F2 shutter opening. The warning tone sounds throughout the delay time T1.

START: Front panel pushbutton which must be pushed after an automatic shutter control mode is selected (PT or SP) to enable the mode actions.

GO: A pushbutton located at the end of an extension cable. This pushbutton is used to initiate a target deflection cycle under SP shutter mode.

METERS: Two front panel 3-1/2 digit digital meters used to display the output signals of the projector photodiodes, one meter for each photodiode. When used in conjunction with the METER SELECT and LIGHT LEVEL ADJ controls the user can monitor and control the light levels of the five VPS projectors.

Operating Instructions

It is assumed that the reader has read and understood the previous sections concerning the VPS description, controls, inputs and outputs.

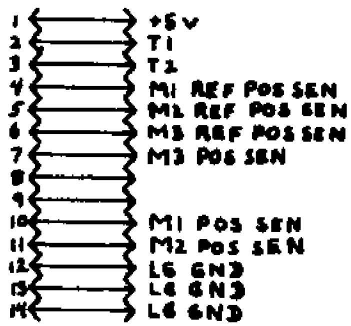
(1) Connect the VPS power cord to the output of a standard AC line voltage regulator (Sorenson ACR 2000 or equivalent) and adjust the regulator's output to provide 120 VAC, 15 A to the VPS. Connect the power cord of the line voltage regulator to a 120 VAC, 15 A, 50-60 Hz outlet. Make sure all connections between the VPS and IES are correctly made. Apply power to the line voltage regulator then to the VPS control electronics. Allow 5 to 10 minutes warm up before beginning adjustments of the projector light levels.

(2) After the initial warm up, begin adjusting the projector light levels to their desired levels by using the front panel controls and light level meters of the VPS control electronics. Allow a period of about 60 minutes warm up and fine adjustment of projector light levels before any experimental procedures are attempted. At the end of the warm up period the projector light levels as monitored using the VPS front panel meters should need only minor adjustments at infrequent intervals. The experimenter should continue to monitor the projector light levels during the experimental procedure.

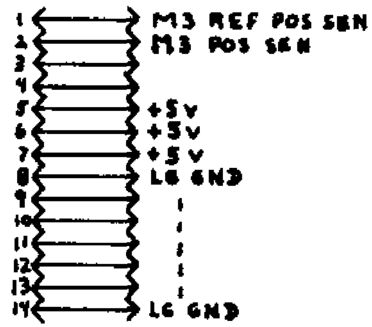
(3) During the period of warm up (about 1 hour) the user can make adjustments to the viewing screen patterns as projected by projectors F1, B1, F2, and B2. Make sure that all patterns are properly positioned on the viewing screen. The subject-to-viewing screen

distance should be 1 meter as measured from the center of the viewing screen to the subject's cornea. Test the actions of all projector shutters using the VPS front panel controls.

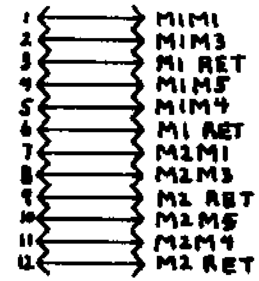
(4) After the VPS has warmed up the user may select any automatic shutter control mode (PT or SP) or manual shutter positions as he desires. The VPS is now ready for experimental use either as a stand alone piece of equipment or under computer control via the IES and PDP-8/E computer.



J1: 14-pin ribbon connector wired for VPS sensors

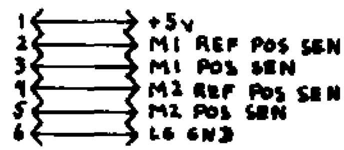


J2: 14-pin ribbon connector wired for movable filter assembly sensors



J3: 12-pin Cinch connector wired for MOTOR 1 and MOTOR 2

J4: 4-pin Cinch connector wired for MOTOR 3



J5: 6-pin Cinch connector wired for VPS target spot mask sensors

FIG. 28 - VPS connectors for connection with IES

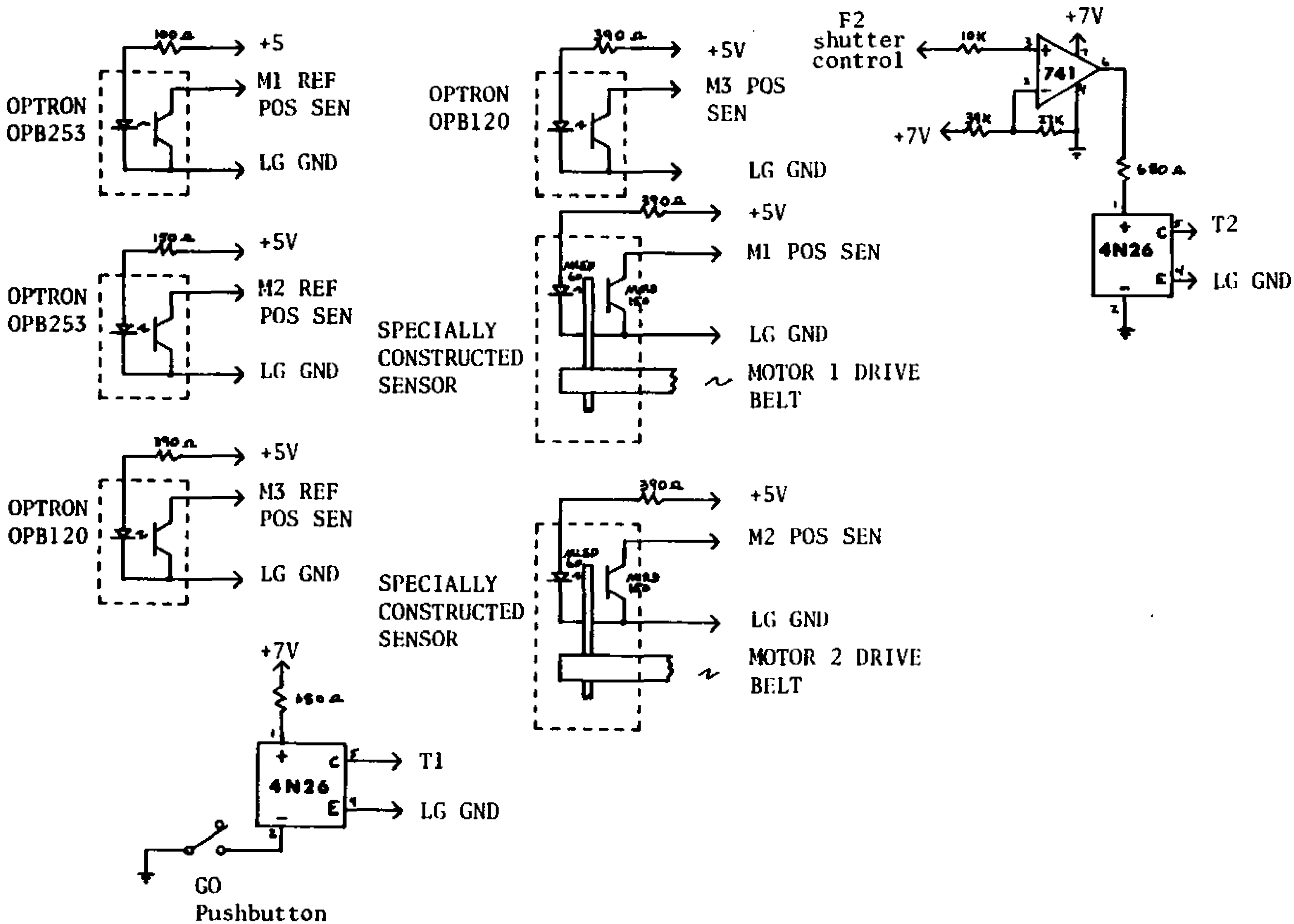


FIG. 29 - VPS mechanical assembly position sensors, T1, T2 sensors

Interface Electronics Section

The interface electronics section (IES) of the target presentation system (TPS) provides all the functions necessary to interconnect the saccade analysis system PDP-8/E computer with the visual presentation section (VPS) of the TPS. The IES generates four signals which are sent to the computer: (1) the TRIG signal which is used to initiate data sampling and storage by the computer, (2) the DEV RDY signal which is used to notify the computer when VPS stepper motor movements have been completed, (3) the FAIL signal which is used to notify the computer of a subject eyeblink and/or an eyetracker track loss and (4) the READ REQ signal which is used to request the computer to read the FAIL bit. The IES accepts signals from the VPS which convey information about: (1) whether or not the VPS target parameter mechanical assemblies have reached their respective reference positions, (2) indications of the positions of these mechanical assemblies relative to their reference positions, and (3) signals used to generate the TRIG signal. The IES accepts signals from the computer: (1) which convey information about the desired positions of the VPS target parameter mechanical assemblies relative to their reference positions, (2) a DSA(H) signal to strobe this position information into the IES registers, (3) a DSB(H) signal to indicate to the IES when no movement is required for MOTOR 3 and (4) a CFA signal to initiate VPS stepper motor movements. Lastly, the IES provides to the VPS stepper motors the pulses necessary to cause them to move.

Included in the IES are various controls and indicators for: (1) controlling the VPS stepper motor speeds, (2) allowing the user full manual and independent control of each VPS stepper motor, (3) failure indication to the user to notify him when the FAIL bit has been set, (4) IES reset to reset all internal flags of the IES and the DEV RDY and FAIL flags, (5) positioning the VPS target parameter mechanical assemblies automatically to their respective reference positions, and (6) VPS ready indication to indicate to the user when the IES DEV RDY flag is set.

The specifications, circuit diagrams, and circuit descriptions of the IES are presented in the following sections.

Specifications, Inputs and Outputs

Motor Speed:	Continuously variable from 0 to 1000 steps per second for MOTOR 1 and MOTOR 2. Continuously variable from 0 to 200 steps per second for MOTOR 3.
Motors:	Capable of driving two 200 step per revolution, 1000 step per second stepper motors, Superior Electric Company type M063-FD06 or equivalent, and one 200 step per revolution, 200 step per second stepper motor, Superior Electric Company type SS25 or equivalent.
BLINK:	A TTL level signal from the eyetracker indicating a subject eyeblink. This signal is active high and if it occurs during deflection time T2 will light the front panel FAIL indicator and set the IES FAIL bit. This signal appears at BNC1 on the IES back panel.
TRACK:	A TTL level signal from the eyetracker indicating an eyetracker track loss. This signal is active low and if it occurs during deflection time T2 will light the front panel FAIL indicator and set the IES FAIL bit. This signal appears at BNC2 on the IES back panel.
T1:	A TTL level input signal from the VPS control electronics that is high during delay time T1 the time before VPS target deflection. This signal appears at pin 2 of J4 on the IES back panel.
T2:	A TTL level input signal from the VPS control electronics that is high during target deflection time T2. This signal appears at pin 3 of J4 on the IES back panel.

M1, M2, M3, REF POS SEN: Signals from the optoelectronic reference position sensors on the VPS target parameter mechanical assemblies that indicate when and if the respective mechanical assemblies are in their reference positions. These signals are the output of the collector of a phototransistor and may be anywhere in the 0 to +5 volt range. They are sensed and converted to TTL level signals by the sensor circuitry of the IES on circuit card 4. M1 REF POS SEN appears at pin 4 of J4, M2 REF POS SEN appears at pin 5 of J4, and M3 REF POS SEN appears at pin 6 of J4 on the IES back panel.

M1, M2, M3, POS SEN: Signals from the optoelectronic position sensors on the VPS target parameter mechanical assebmlies indicating the position of the respective mechanical assemblies relative to their reference position. These signals are the output of the collector of a phototransistor and as such should be treated as an open collector TTL output. They are sensed and converted to TTL level signals by the sensor circuitry of the IES on circuit card 4. MI POS SEN appears at pin 10 of J4, M2 POS SEN appears at pin 11 of J4, and M3 POS SEN appears at pin 7 of J4 on the IES back panel.

M1M1, M1M3, M1M4, M1M5: MOTOR 1 high level drive pulses (24 volts, 3 amps) used to drive the windings of VPS stepper motor MOTOR 1. M1M1 appears at pin 1 of J5, M1M3 appears at pin 2 of J5, M1M4 appears at pin 5 of J5, and M1M5 appears at pin 4 of J5 on the IES back panel.

M1 RET: Two current return lines from VPS stepper motor MOTOR 1. They appear at pins 3 and 6 of J5 on the IES back panel.

M2M1, M2M3, M2M4, M2M5:	MOTOR 2 high level drive pulses (24 volts, 3 amps) used to drive the windings of VPS stepper motor MOTOR 2. M2M1 appears at pin 7 of J5, M2M3 appears at pin 8 of J5, M2M4 appears at pin 11 of J5, and M2M5 appears at pin 10 of J5 on the IES back panel.
M2 RET:	Two current return lines from VPS stepper motor MOTOR 2. They appear at pins 9 and 12 of J5 on the IES back panel.
M3W1, M3W2:	MOTOR 3 high level drive pulses (± 24 volts, 0.05 amps) for driving the windings of VPS stepper motor MOTOR 3. M3W1 appears at pin 14 of J5 and M3W2 appears at pin 15 of J5 on the IES back panel.
M3 RET:	A single current return line from VPS stepper motor MOTOR 3. It appears at pin 13 of J5 on the IES back panel.
TRIG:	A rear panel jack, BNC3, which provides a 2-3 microsecond TTL high level pulse at the onset of VPS target deflection time T2. The TTL fanout capability of this signal is 2. This pulse is compatible with and may be used to start a Digital Equipment Corporation PDP-8/E computer real time clock for subsequent data sampling by the computer's A/D converter.
+5V:	The +5 volt supply voltage used for powering the optoelectronic position sensors on the VPS target parameter mechanical assemblies. Its total current capability is 1 amp. It appears at pin 1 of J4 on the IES back panel.
LG GND:	Logic ground which appears at pins 12, 13 and 14 of J4, one row of pins each of J1, J2 and J3 and the shells

of BNC1, BNC2 and BNC3 on the IES back panel. This is the ground reference for T1, T2, M1, M2, M3 REF POS and POS SEN, +5V, BLINK, TRACK, TRIG signals, and the data and control signals of connectors J1, J2 and J3.

Data and control signals on J1, J2 and J3:

Connectors J1, J2 and J3 on the IES back panel are 40 pin 3M connectors providing connections for the data and control signals between the DEC PDP-8/E computer M1703 and M1705 omnibus interface modules and the IES. All signals are TTL level high true signals. Output signals from the IES have a fanout of 2. Input signals to the IES have a fan in of 1. The connections of J1, J2 and J3 are listed in the tables below.

Connector J1 on the IES back panel is connected via 40 conductor flat cable to the computer M1705 omnibus output interface channel A. Its connections are as follows:

J1 Pin	Signal Name	Data
D	DA1	MOTOR 1 stp dir
F	DA2	MOTOR 1 45°ct (MSB)
J	DA3	MOTOR 1 45° ct
L	DA4	MOTOR 1 45° ct
N	DA5	MOTOR 1 45°ct (LSB)
T	DA7	MOTOR 2 stp dir
V	DA8	MOTOR 2 45°ct (MSB)
X	DA9	MOTOR 2 45° ct
Z	DA10	MOTOR 2 45° ct
BB	DA11	MOTOR 2 45°ct (LSB)
FF	DEV RDY	Device ready flag
JJ	DSA(H)	Data strobe A
TT	CFA	Control flag A
A-UU	LG GND	Logic ground

DEV RDY is a high level signal which sets the M1705 interface channel A flag. This flag indicates to software that all VPS stepper motor movements have finished. That is, the selected target is available for subject viewing. The IES front panel indicator will light when DEV RDY is set high. DSA(H) is a 1 microsecond high level pulse used to strobe the data of M1705 channel A into the motor count latches of the IES. CFA is an active high level signal from the M1705 channel A which is used by computer software to initiate VPS motor movements.

Connector J2 on the IES back panel is connected via 40 conductor flat cable to the computer M1705 omnibus output interface channel B. Its connections are as follows:

J2 pin	Signal Name	Data
T	DB7	MOTOR 3 stp dir
V	DB8	MOTOR 3 fil ct (MSB)
X	DB9	MOTOR 3 fil ct
Z	DB10	MOTOR 3 fil ct
BB	DB11	MOTOR 3 fil ct (LSB)
JJ	DSB(H)	Data strobe B
A-UU	LG GND	Logic ground

DSB(H) is a 1 microsecond high level pulse used to indicate a zero motor movement to the IES for VPS stepper motor MOTOR 3.

Connector J3 on the IES back panel is connected via 40 conductor flat cable to the M1703 omnibus input interface. Its connections are as follows:

J3 pin	Signal name	Data
B	FAIL	Failure indic bit
DD	READ REQ	Read request
A-UU	LG GND	Logic ground

READ REQ is a 2-3 microsecond low pulse at the onset of VPS target deflection time T2. This signal is used to set the M1703 interface flag, allowing the computer to read the FAIL failure bit under software control. The FAIL bit is normally low. It will be set high by a subject eyeblink or an eyetracker track loss occurring during VPS target deflection time T2. This bit is always reset at the end of time T2.

Controls and Indicators

POWER:	Front panel switch used to apply line power to the IES circuitry. The power light will shine when power is applied.
RESET:	Front panel pushbutton used to apply a reset pulse to all IES registers and flags. When this pushbutton is actuated the IES electronics are initialized to their power-up conditions.
REF POS:	Front panel pushbutton used to apply pulses to the VPS stepper motors. This allows the stepper motors to move the VPS target parameter mechanical assemblies until they reach their respective reference positions at which time the motors will discontinue movements. When all assemblies have reached their reference positions the IES front panel READY indicator will light and the DEV RDY flag will be set. The mechanical assembly reference positions yield a target spot whose parameters are: 2.0 degrees radius, 0 degrees angle, filter number 0 (the most dense).
READY:	Front panel indicator which lights whenever the DEV RDY flag is set. This indicator notifies the user whenever all VPS stepper motor movements are completed. This indicator will thus come on when the VPS mechanical assemblies have reached their reference positions and whenever they have finished a computer directed target parameter selection.
FAIL:	Front panel indicator which will light if a failure occurs during VPS target deflection time T2. The eyetracker BLINK and TRACK signals are connected to their IES rear panel BNC connectors so that whenever a subject eyeblink or eyetracker

track loss occurs during VPS target deflection time T2 the FAIL indicator will light and the FAIL bit will be set.

MANUAL/AUTO: Front panel switch used to select the MANUAL (user) or AUTO (computer) control mode for the VPS stepper motors.

CLK/SINGLE STEP: Front panel switch used to apply continuous clock pulses (CLK) or single step pulses (SINGLE STEP) to the VPS stepper motors when the user selects the MANUAL mode of stepper motor control.

STEP: Front panel pushbutton used in conjunction with the CLK/SINGLE STEP and MANUAL/AUTO controls. When the user selects both the MANUAL stepper motor mode and the SINGLE STEP pulse mode he may cause any one of the three VPS stepper motors to advance one step by actuating the STEP pushbutton.

MANUAL SPEED: Front panel control used to adjust the VPS stepper motors step speed when they are used in the MANUAL mode of control. This control also adjusts the step speed of VPS motor MOTOR 3 (the filter assembly motor) in the AUTO mode of stepper motor control. Additionally this control controls the step speed of all three VPS stepper motors during the REF POS reference positioning automatic cycle.

M1 MAN: Front panel switch used to select the VPS stepper motor MOTOR 1 for movement when the switch is used in conjunction with the MANUAL operation mode. The motor will move when the switch is placed in the M position.

M2 MAN: Front panel switch used to select the VPS stepper motor MOTOR 2 for movement when the switch is used in conjunction with the MANUAL operation mode. The motor will move when the switch is placed in the M position.

M3 MAN: Front panel switch used to select the VPS stepper motor MOTOR 3 for movement when the switch is used in conjunction with the MANUAL operation mode. The motor will move when the switch is placed in the M position.

M1 MAN DIR: Front panel switch used to select the rotation direction of VPS MOTOR 1, clockwise (CW) or counter clockwise (CCW) in the MANUAL motor operation mode of the IES.

M2 MAN DIR: Front panel switch used to select the rotation direction of VPS MOTOR 2 clockwise (CW) or counter clockwise (CCW), in the MANUAL motor operation mode of the IES.

M3 MAN DIR: Front panel switch used to select the rotation direction of the VPS MOTOR 3, clockwise (CW), or counter clockwise (CCW), in the MANUAL motor operation mode of the IES.

AUTO SPEED: Front panel control used to adjust the step speed of VPS stepper motors MOTOR 1 and MOTOR 2 in the AUTO motor operation mode of the IES.

Operating Instructions

The IES package is designed specifically to operate in conjunction with the VPS and the PDP-8/E computer. It is assumed that the reader has read and understood all sections concerning the VPS, and the sections concerning the IES inputs, outputs, and controls.

(1) Before applying power to the IES, make sure all cables running between the IES, the PDP-8/E, the VPS, and the eyetracker are properly connected to their respective input and output jacks.

(2) Connect the IES power cord to a 120 VAC, 50-60 Hz, 20 AMP outlet. Apply power to the IES and allow the IES circuitry to warm up for a couple of minutes.

(3) The user may now choose to operate the VPS stepper motors under manual control. If this is desired he should select the MANUAL switch position of the IES front panel MANUAL/AUTO switch. The user is now free to move the VPS motors in either direction and at any rate of from 0 to 200 steps per second using the IES front panel controls.

(4) If the user chooses not to use the manual VPS motor control mode, but rather the automatic control mode, he must energize the PDP-8/E computer and call an appropriate software control package into computer memory.

(5) When using the automatic control mode of the IES, it is suggested that the user first set up the VPS for operation as described in the VPS operating instructions. After the VPS has been adjusted and warmed

up for its one hour warm up period the user should then energize the IES and PDP-8/E computer. After a few minutes of warm up the user can call his software control package. This software package should first request the user to initiate an automatic reference position cycle by actuating the REF POS pushbutton on the IES front panel. Then the software should wait for the DEV RDY signal from the IES before proceeding with any VPS motor control cycles.

(6) As an example of the complete utilization of the TPS (IES and VPS combined) with the PDP-8/E computer the following set up has been successfully used by the author:

All connections between the VPS, IES, PDP-8/E, and the Cornsweet-Crane eyetracker are checked to be sure that they are properly made. The VPS is energized and allowed to warm up for at least 60 minutes. During VPS warm up the VPS projector light levels are adjusted to their desired levels, the positioning of the F1, B1, F2, and B2 projector display patterns on the viewing screen are checked for correctness, the operation of all VPS projector shutters is checked for correctness, the PDP-8/E computer hardware is energized and allowed to warm up, and the eyetracker is energized and allowed to warm up.

After the warm up period the VPS F2 and F1 projector patterns are used with the subject to calibrate the eyetracker H4 and V4 output gains to 0.2 volts per degree of subject eye rotation. Then the VPS automatic shutter mode SP is chosen and checked for operation, T1 delay time is set for 1.5 seconds, T2 target deflection time is set for 3 seconds. The other VPS shutters are set as follows: F1 shutter closed, B1 shutter open, B2 shutter closed. This provides the subject with a viewing screen pattern of a central fixation target spot and a constant background illumination.

Program package ADCOM is called into computer core memory and the sampling parameters are set to two channels starting with channel 0, 5 milliseconds sampling period and 500 samples per channel. A beginning file name is chosen, all target positions are left enabled, and an initial target position scrambling factor is chosen. The IES is energized and allowed to warm up for a couple of minutes and sampling mode AR is chosen for data sampling and control.

The AR sampling mode of ADCOM first requests the user to push the IES REF POS pushbutton and then waits for the DEV RDY flag from the IES which indicates that the VPS target parameter mechanical assemblies have reached their reference positions. When the DEV RDY (and the IES READY indicator) indicate that the VPS is initialized the AR sampling mode selects target parameters and controls the IES to move the VPS target assemblies to new positions. The DEV RDY flag will be set again when the VPS assemblies have reached their new positions. The user can now initiate an experimental trial by activating the VPS GO pushbutton. When this is done the VPS warning buzzer sounds for the delay time T1 (1.5 seconds here) after which the central fixation spot on the VPS target viewing screen is "deflected" to the target position as set by the VPS spot generation masks. The contrast of the target will be determined by the filter set by the movable filter assembly of the VPS. The target spot remains on for the T2 target deflection time after which the spot "returns" to the central fixation spot location. The computer samples the eyetracker H4 and V4 eye rotation signals for 2.5 seconds beginning at the start of the T2 target deflection period. At the end of sampling the computer checks the IES FAIL bit. If no failure has occurred the data record just taken is stored on disk FL2 with the most current file name as the data record name, a new set of target parameters is chosen, and the VPS assemblies are moved to their appropriate locations as dictated by these new target parameters. The above process is repeated until all 40 target positions have been tested once; the experimental session ends. If a failure occurs during data sampling (T2 deflection period) the data for that record is rejected and not stored on disk, new target parameters are set into the VPS and the experimental session proceeds.

Circuit Operation

The circuits of the IES are divided into functional sections each of which is contained on a single circuit card. Circuit card 1 contains the automatic mode stepper motors step counters and registers. Circuit card 2 contains the start-up and automatic motor movement control logic and flag registers. Circuit card 3 contains the manual/auto mode control logic for the stepper motors. Circuit card 4 contains the sensor detection circuitry for the VPS target parameter mechanical assemblies' reference position sensors and position sensors. Circuit card 6 contains the +24 volt, -24 volt, and the +5 volt DC power supplies' regulation circuitry. Circuit cards 7, 8 and 9 are the power driver cards for the three VPS stepper motors. Card 7 was constructed by the author for driving the Superior Electric type SS25 stepper motor (VPS MOTOR 3). Cards 8 and 9 are Superior Electric Company type STM101 translator cards designed for use with the two Superior Electric type M063-FD06 stepper motors (VPS MOTOR 1 and MOTOR 2 respectively).

The operation of IES circuitry can best be understood by referring to the timing diagrams shown in Fig.'s 30, 31, 32, 33 and 34. When power is applied to the IES all circuits are initialized by the SUI and SUI1 signals of card 2 as shown in Fig. 30. When the IES is to be used under software control a reference positioning cycle as shown in timing diagram Fig. 30 must be completed. That is, computer software should request and wait for the user to actuate the REF POS pushbutton on the IES front panel. This action applies clock pulses to the three VPS stepper motors and drives them in their clockwise directions until the reference position sensors indicate that the motors have moved their respective VPS target parameter mechanical assemblies to their respective reference positions. The three flags M1SUF, M2SUF, and M3SUF of card 2 will be set as their respective VPS assemblies reach their reference positions. For example when MOTOR 1 has driven its mechanical assembly (the A target spot mask) to its reference position the M1SUF flag will be set and clock pulses will be inhibited from the MOTOR 1 drive card (card 8). After the last of the reference position flags (SUF's) is set the SUI1 and DEV RDY flags of card 2 are set. The READY indicator on the IES front panel will light to indicate to the user that the VPS assemblies are initialized. The IES is now ready to accept computer commands and target parameters. At any time during the automatic operation mode of the IES the user may actuate the RESET pushbutton on the IES front panel and perform a manual reset of all IES control logic, see Fig. 30. This returns all IES control logic to conditions present shortly after the power-on sequence. All flags are reset as indicated in Fig. 30. The user must initiate another reference positioning cycle before the software can continue its automatic IES control. It should be noted here that due to the relative positioning nature of the IES and computer software, if a reset is performed, the software should also be restarted at its beginning.

By referring to Fig.'s 31 and 32 the automatic (software) control mode sequence of the IES can be understood. The computer software must first calculate the number of "steps" (45° movements by MOTOR 1 and MOTOR 2 and 90° movements by MOTOR 3) required for each VPS motor and also its direction. This information is then placed at the outputs of the M1705 omnibus output interface. Under software control the DSA(H) strobe pulse is issued. This loads the count registers U9, U13 and U17 on card 1 with the M1705 data. It also initiates the $\overline{DS1}$ strobe signal of card 1. The $\overline{DS1}$ strobe signal loads the step counts for each motor from the U9, U13 and U17 registers into the down counters U5, U8 and U20 respectively on card 1. The $\overline{DS1}$ strobe also clears the DEV RDY flag register U2 on card 2. The computer software now sets the M1705 CFA flag to initiate VPS stepper motor movements. Clock pulses are applied to the motor driver cards 7, 8 and 9 through the gates U9A, U9B and U10A on card 2 and through the routing circuitry of card 3. When a down counter on card 1 reaches count zero the carry out signal of the counter will go low (CO of U5, U8 or U20 on card 1). This will inhibit clock pulses from reaching the particular motor associated with the card 1 counter that has reached zero count. Once the last counter on card 1 reaches zero all motor movements will stop and the DEV RDY flag of card 2 will be set indicating to the computer software that the VPS assemblies have reached their desired locations. The software and the IES circuitry now wait for the user to actuate the VPS GO pushbutton. This action initiates the T1 delay period at the beginning of which the READY indicator on the IES front panel is turned off. During the T1 delay period the VPS warning buzzer sounds. The T1 delay period is followed by the T2 target deflection period during which the subject is viewing the off-axis target spot and the computer is sampling the eyetracker output data. The READ REQ and COMP CLK TRIG pulse signals are generated on card 2 by gates U13F, U14E, and U14F at the beginning of the T2 target deflection period.

At any time during the operation of the IES the user may choose to select the manual operating mode for the VPS motors. This causes the multiplexers U8 and U15 on card 3 to select their B inputs and thus allow the user to control the VPS motor movements with IES front panel switches and pushbuttons. Fig. 33 indicates the manual/auto operation mode selection timing. Care should be taken when the manual mode of operation is chosen during an automatic control cycle. Position sensing of the VPS mechanical assemblies is always relative. Thus manual adjustments of the VPS mechanical assembly positions during automatic control may disrupt the automatic positioning of the VPS assemblies.

The operation of the failure detection circuitry and the FAIL flag can be understood by referring to the timing diagram of Fig. 34. On card 2 U14D and U20C with their associated resistors and diodes sense the levels of the eyetracker BLINK and TRACK signals. If either BLINK is high or TRACK is low during target deflection period T2 the flip-flop U7B on card 2 will be set by the clock pulses supplied by U20B on card 2. Thus the FAIL flag on card 2 will be set and the FAIL indicator on the IES front panel will light. Note that U7B on card 2 is held in reset until both SUI1 is set and T2 is present.

The position sensor detection circuitry on card 4 provides the reference position signals M1S, M2S and M3S to strobe the M1SUF, M2SUF and M3SUF flag registers U5 and U7A on card 2. The position sensor detection circuitry of card 4 also provides the clock pulses $\overline{M1C}$, $\overline{M2C}$ and $\overline{M3C}$ to the motor step down counters U5, U8 and U20 on card 1.

Notes for timing diagram Fig. 30

A. Power is applied to the IES by the user.

B. Power-on reset released from U16, U5, U7A, U12 and U2 on card 2 and U5 on card 3.

C. REF POS pushbutton on the IES front panel is actuated clocking U5A on card 3. This applies CLK pulses to motor driver cards 7, 8 and 9 as indicated by the M3B, M1CW and M2CW signals respectively. Pulses continue until reference position sensors indicate that the VPS assemblies have reached their reference positions. SUI1 selects the clockwise rotation direction for MOTOR 1, MOTOR 2 and MOTOR 3.

D. MIS indicates that MOTOR 1 has moved the VPS A target spot mask to its reference position. The M1S pulse sets the MISUF flag of card 2. Pulses are removed from M1CW and the MOTOR 1 driver card, card 8.

E. M2S indicates that MOTOR 2 has moved the VPS B target spot mask to its reference position. The M2S pulse sets the M2SUF flag of card 2. Pulses are removed from M2CW and the MOTOR 2 driver card, card 9.

F. M3S indicates that MOTOR 3 has moved the VPS movable filter assembly to its reference position. The M3S pulse sets the M3SUF flag of card 2. Pulses are removed from M3B and the MOTOR 3 driver card, card 7. Note the order of appearance of M1S, M2S and M3S is not restricted to that shown here. The three motors move independently. The last SUF to be set causes the SUI1 and DEV RDY flags of card 2 to be set. In this illustration the M3SUF appears last so that it sets SUI1 and DEV RDY. The IES is not available for automatic control mode usage until SUI1 is set, that is, until the VPS target parameter assemblies are initialized.

G. A manual, asynchronous reset is performed by user actuation of the RESET pushbutton on the IES front panel. All IES flags are reset.

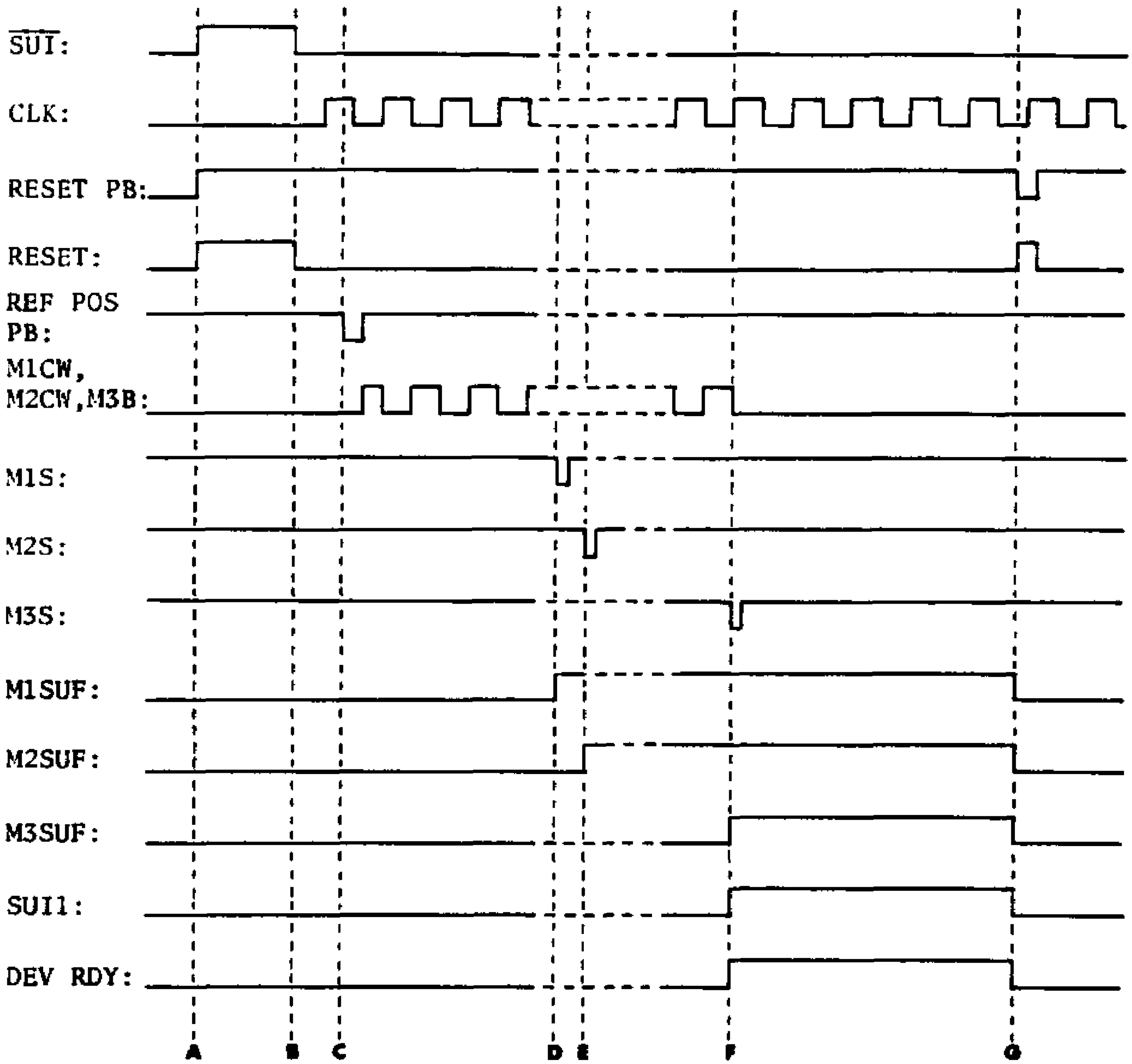


FIG. 30 - Start-up and reference positioning cycle timing

Notes for timing diagram Fig. 31

A. SUI1 indicates that all VPS mechanical assemblies are initialized (reached their reference positions). Releases the reset from count registers of card 1. Sets DEV RDY, and lights the READY IES front panel indicator.

B. DSA(H) strobe from the M1705 output interface of the computer loads the M1705 data into registers U9, U13 and U17 on card 1.

C. DSA(H) triggers the $\overline{DS1}$ strobe which loads the down counters U5, U8 and U20 on card 1 with the motor step counts of the U9, U13 and U17 on card 1. The $\overline{DS1}$ strobe resets the DEV RDY flag of card 2 and the U16B flip-flop on card 2.

D. Computer software sets the M1705 CFA flag to initiate VPS motor movements. CFA also produces the AC RUN signal. These two signals apply AUTO CLK pulses to MOTOR 1 and MOTOR 2 driver cards 8 and 9, and CLK pulses to MOTOR 3 driver card 7.

E. The MOTOR 1 count register U5 on card 1 has reached zero count. Pulses are inhibited from the MOTOR 1 drive card 8. MOTOR 1 movement is completed.

F. The MOTOR 2 count register U8 on card 1 has reached zero count. Pulses are inhibited from the MOTOR 2 drive card 9. MOTOR 2 movement is completed.

G. The MOTOR 3 count register U20 on card 1 has reached zero count. Pulses are inhibited from the MOTOR 3 drive card 7. MOTOR 3 movement is completed. DEV RDY flag is set. Note the order of completion of motor movements is entirely flexible. The last motor to complete its movement sets the DEV RDY flag. DEV RDY notifies computer that all movements are completed.

H. CFA (and thus AC RUN) is reset under software control after the computer senses the DEV RDY flag.

I. User actuates VPS GO pushbutton to initiate a target deflection trial. T1 delay time occurs which turns off the IES READY indicator. The VPS warning buzzer sounds throughout the T1 delay period.

J. At the end of T1 the target deflects and remains deflected for target deflection time T2. COMP CLK TRIG and READ REQ are generated at the high-to-low transition of T2.

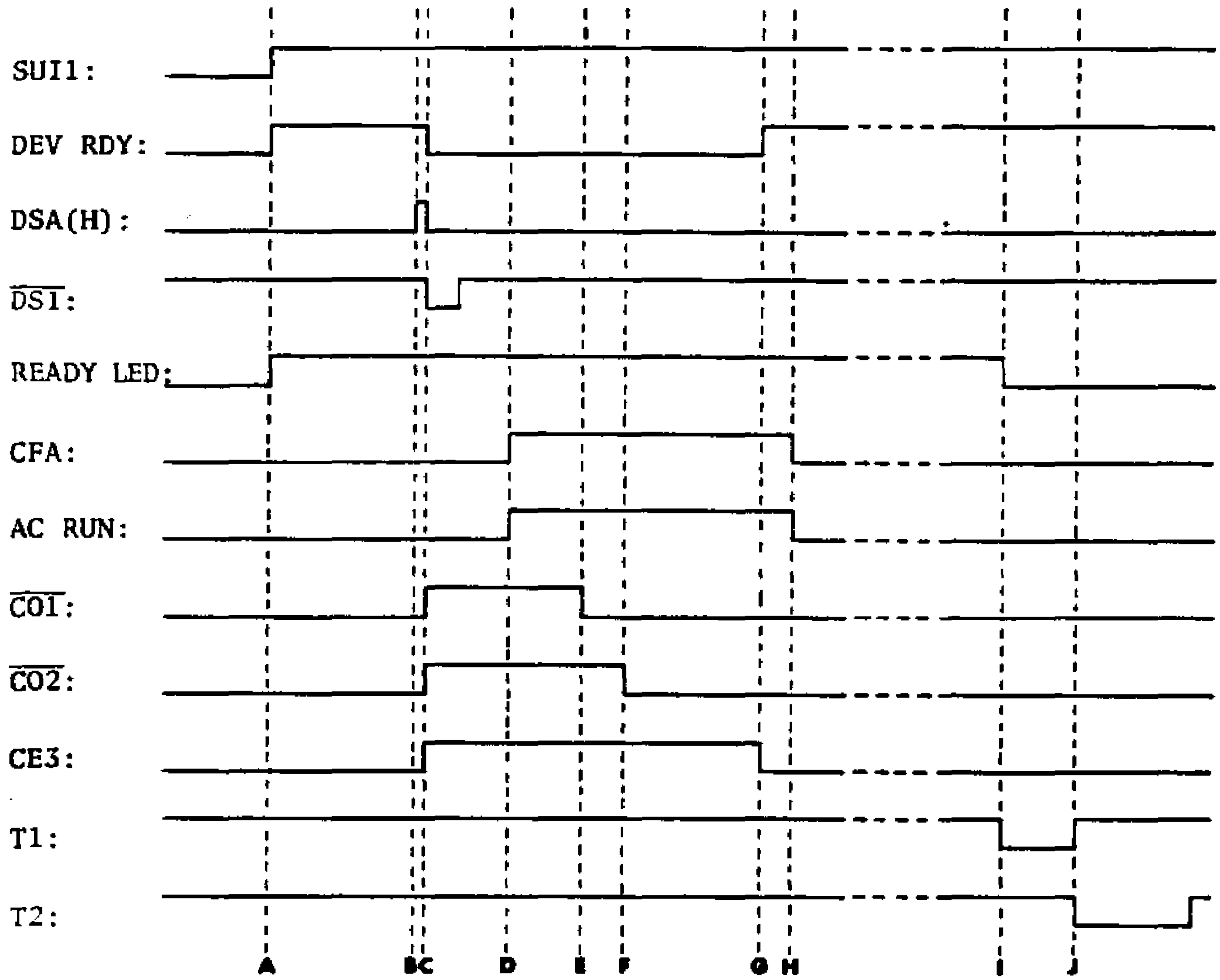


FIG. 31 - Automatic control mode timing cycle for first sequence after initialization

Notes for timing diagram Fig. 32

A. $\overline{\text{DS1}}$ loads count registers U5, U8 and U20 on card 1 with new motor step counts and resets DEV RDY flag.

B. Software sets CFA to initiate motor movements.

C. The last motor to complete movement, in this case MOTOR 3, sets DEV RDY, and READY LED (lights IES READY indicator).

D. User actuates VPS GO pushbutton to initiate a trial sequence. READY LED is reset. VPS warning buzzer sounds for duration of T1 delay period.

E. T1 time out initiates target deflection for duration of T2 period. T2 generates COMP CLK TRIG to initiate eyetracker data sampling.

F. Target deflection is completed.

G. $\overline{\text{DS1}}$ loads count registers U5, U8 and U20 on card 1 with new motor step counts and resets DEV RDY flag. DSB(H) strobe signal is sent from the M1705 interface by the software to indicate a zero motor movement required for MOTOR 3. DSB(H) causes DS2 to be issued from card 1. DS2 clocks U16B on card 2 to emulate the CE3 signal.

H. CFA is set by software to initiate motor movements. MOTOR 3 will not move this time because of DSB(H).

I. Last motor to finish movement, MOTOR 2 here, sets DEV RDY and READY LED.

J. Experimenter initiates another experimental trial as in note D above.

K. T2 generates COMP CLK TRIG as in note E above.

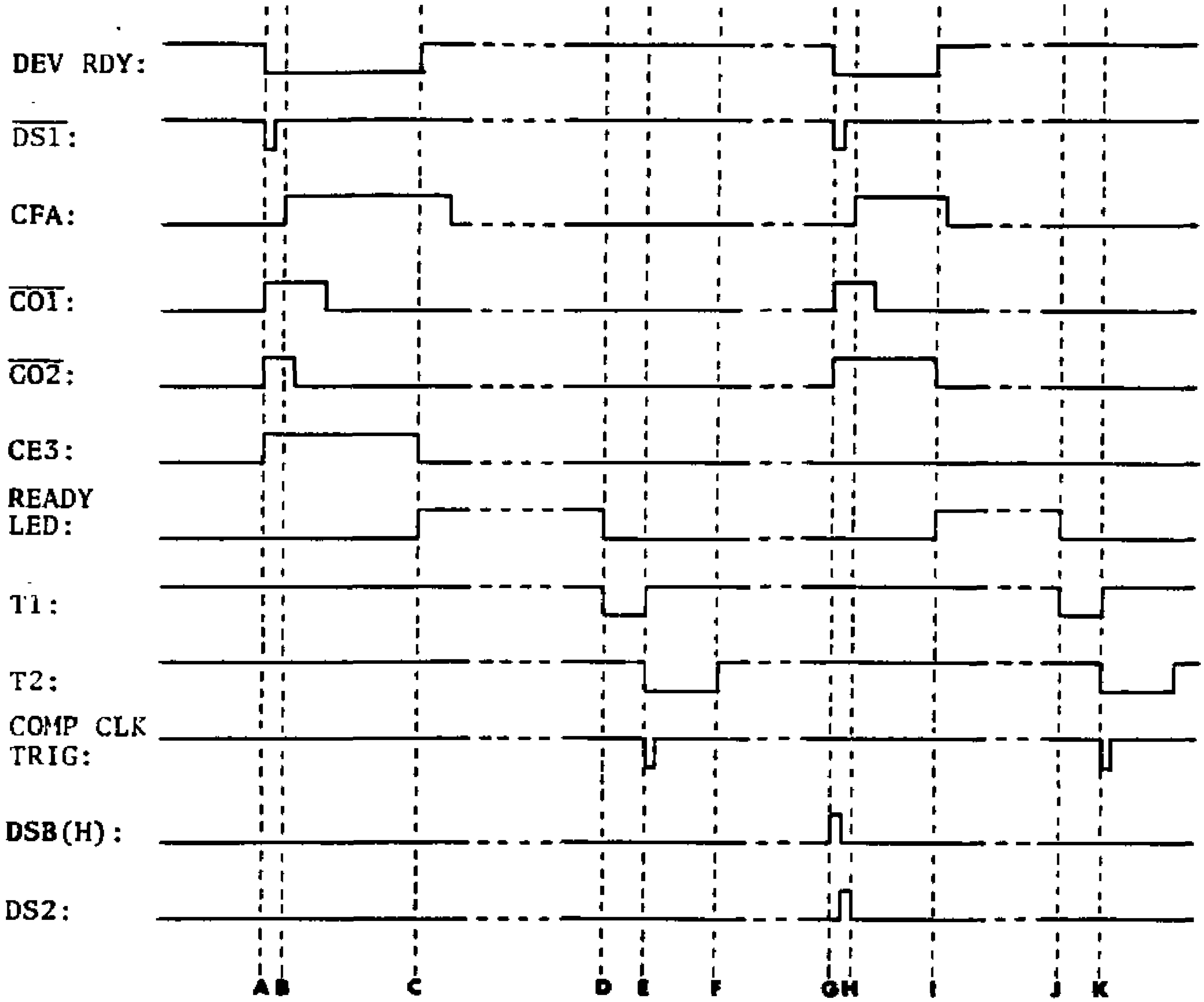


FIG. 32 - Automatic control mode timing cycle for second and following sequences after initialization.

<u>Notes for timing diagram Fig. 33</u>

A. The IES is in automatic control mode.

B. Since the IES is in automatic mode, actuating the STEP pushbutton has no effect on the motors.

C. Since the IES is in automatic mode, switching the M1, M2 or M3 MAN switches has no effect on the motors.

D. The user switches from automatic mode to manual mode. Since CLK/SS SW is low CLK pulses are applied to any VPS motor whose MAN switch is in the M position.

E. One of the MAN switches is placed in the M position. CLK pulses will be applied to that motor causing it to move in the direction selected by the user by that motor's MAN DIR switch.

F. The CLK/SINGLE STEP switch is switched to the SINGLE STEP position now single pulses will be applied to any motor whose MAN switch is in the M position each time the SINGLE STEP pushbutton is actuated by the user. Actuating the SINGLE STEP pushbutton will advance a motor one step in the selected direction.

G. A motor is selected by placing its MAN switch in the M position now that motor will advance one step each time the SINGLE STEP pushbutton is actuated. Direction of movement is controlled with the motor's MAN DIR switch.

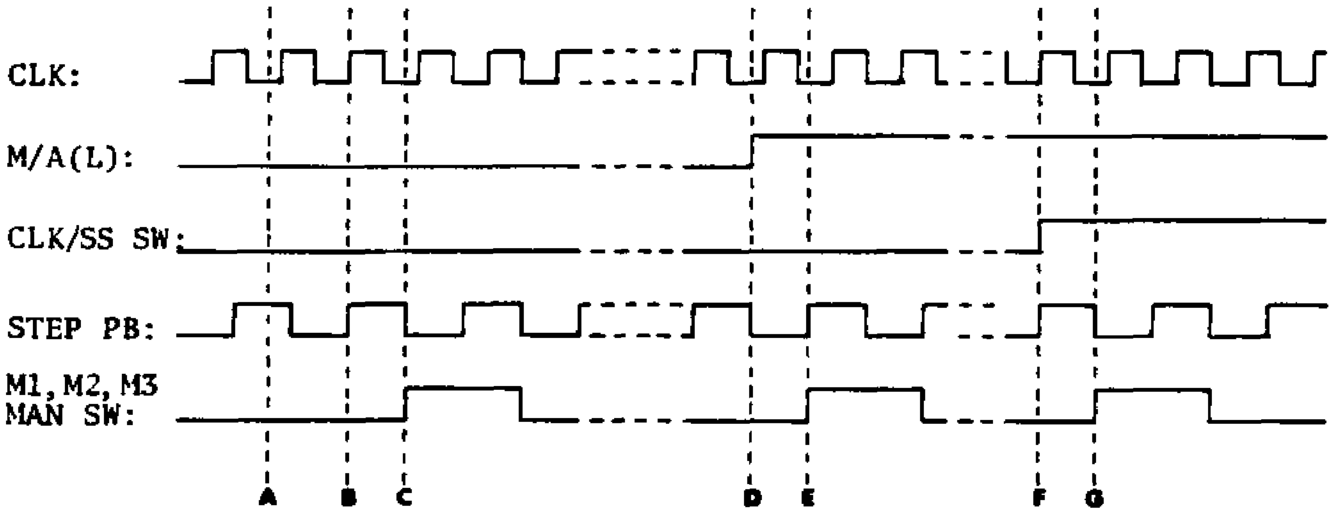


FIG. 33 - Manual/Auto mode select timing

Notes for timing diagram Fig. 34

A. Any BLINK(H) or TRACK(L) signal pulse that occurs before VPS initialization has no effect on the fail circuitry.

B. SUI1 indicates that the VPS assemblies have been initialized.

C. A TRACK(L) signal occurs to indicate a track loss that occurs just prior to the T1 delay time and continues into the T1 delay period but ends before T2 goes low. This track loss will not set FAIL flag.

D. User initiates a deflection trial by actuating the VPS GO push-button.

E. The T2 high-to-low transition generates the READ REQ signal to prompt the computer to read the FAIL flag at the end of data sampling.

F. A BLINK(H) has occurred, the FAIL flag will be set on the first low-to-high transition of CLK. BLINK going high indicates a subject eyeblink.

G. The end of the T2 period resets the FAIL flag. The computer must read the FAIL flag before T2 time-out to receive valid information. That is, FAIL flag is valid only during the T2 period.

H. Another deflection trial is initiated.

I. T2 period begins and READ REQ is generated.

J. A TRACK(L) has occurred and the first low-to-high transition of CLK sets FAIL. TRACK going low indicates an eyetracker track loss.

K. T2 period ends and the FAIL flag is reset.

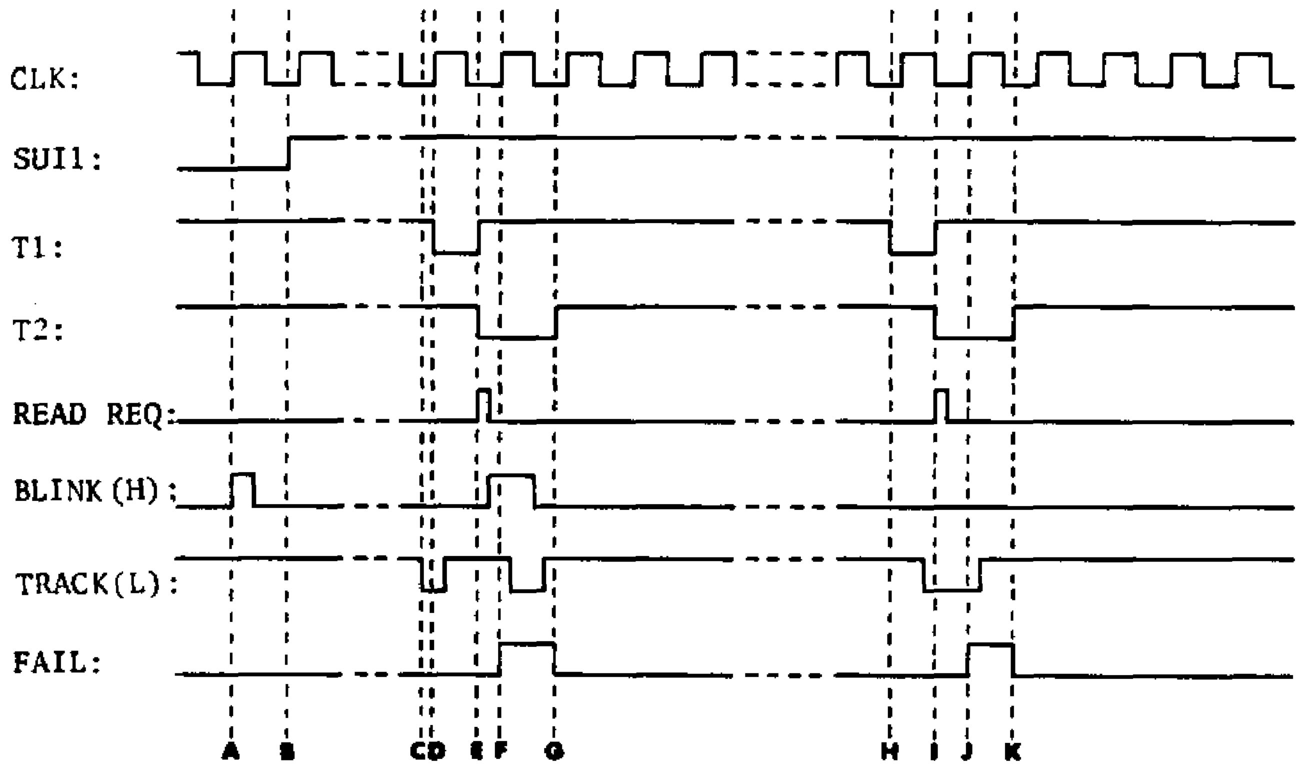


FIG. 34 - Failure timing

TABLE 1 - SIGNAL INTERCONNECTION LIST

SIGNAL NAME	FROM (CARD/PIN)	TO (CARD/PIN)
LG GND	6/12	1/2,B
		2/2,B
		3/2,B
		4/2,B
		5/2,B
		7/2,B
		8/k,L
		8/k,L
		BNC1, BNC2, BNC3
		L2, L3
		S2, S3, S5, S9, S10, S11
		J1, J2, J3, J4
+5V	6/12	1/3,C
		2/3,C
		3/3,C
		4/3,C
		5/3,C
		7/3,C
		S4, S6, S7, S8
		J4
AN GND	6/4,D	7/10,N,P
		8/k,L
		9/k,L
+24V	6/5,E	7/11
		8/P
		9/P
-24V	6/17, 18, V	7/12
CO1	1/4	2/19
CO2	1/E	2/W
ADIR	1/6	3/L
BDIR	1/H	3/10
CDIR	1/7	3/M
CE1	1/J	2/21
CE2	1/8	2/Y
CE3	1/k	2/20
$\overline{DS1}$	1/F	2/T
DS2	1/S	2/18

(Table 1 Cont'd.)

SIGNAL NAME	FROM (CARD/PIN)	TO (CARD/PIN)
DSA(H)	J1/JJ	1/S
DSB(H)	J2/JJ	1/10
DB7	J2/T	1/15
DB8	J2/V	1/T
DB9	J2/X	1/16
DB10	J2/Z	1/U
DB11	J2/BB	1/17
DA1	J1/D	1/R
DA2	J1/F	1/X
DA3	J1/J	1/20
DA4	J1/L	1/Y
DA5	J1/N	1/21
DA7	J1/T	1/P
DA8	J1/V	1/V
DA9	J1/X	1/18
DA10	J1/Z	1/W
DA11	J1/BB	1/19
M1A	2/10	3/J
M2A	2/L	3/8
M3A	2/H	3/K
FAIL	2/S	J3/B
FAIL LED	2/F	L2
AC RUN	2/12	8/U
CLK	2/E	3/9
READY LED	2/6	L3
DEV RDY	2/7	J1/FF
M1SUF	2/K	3/P
M2SUF	2/8	3/13
M3SUF	2/J	3/R
COMP CLK TRIG	2/11	BNC3
READ REQ	2/M	J3/DD
RESET	2/N	3/14
BLINK(H)	BNC1	2/13
TRACK(L)	BNC2	2/P

(Table 1 Cont'd.)

SIGNAL NAME	FROM (CARD/PIN)	TO (CARD/PIN)
T1	J4/2	2/R
T2	J4/3	2/16
MMS(L)	CRI	2/S
MMS(H)	CRI	2/14
RESET PB	S2	2/15
CFA	J1/TT	2/X
M1S	4/X	2/V
M2S	4/W	2/17
M3S	4/M	2/U
$\overline{\text{M1C}}$	4/17	1/13
$\overline{\text{M2C}}$	4/18	1/12
$\overline{\text{M3C}}$	4/L	1/11
M1CW	3/D	8/S
M1CCW	3/4	8/R
M2CW	3/E	9/S
M2CCW	3/5	9/R
M3D	3/6	7/H
M3B	3/F	7/E
REF POS PB	S3	3/S
CLK/SS SW	S4	3/15
STEP PB	S5	3/T
M1 MAN SW	S6	3/17
M2 MAN SW	S7	3/U
M3 MAN SW	S8	3/16
M/A(H)	S9	3/21
M/A(L)	S9	3/4
M1 MAN DIR SW(H)	S10	3/20
M1 MAN DIR SW(L)	S10	3/X
M2 MAN DIR SW(H)	S11	3/19
M2 MAN DIR SW(L)	S11	3/W
M3 MAN DIR SW(H)	S12	3/18
M3 MAN DIR SW(L)	S12	3/V

(Table 1 Cont'd.)

SIGNAL NAME	FROM (CARD/PIN)	TO (CARD/PIN)
AMS(X)	CR2	8/X
AMS(W)	CR2	8/W
AMS(T)	CR2	8/T
AUTO CLK	8/V	2/4
M1 REF POS SEN	J4/4	4/U
M2 REF POS SEN	J4/5	4/T
M3 REF POS SEN	J4/6	4/H
M1 POS SEN	J4/10	4/14
M2 POS SEN	J4/11	4/15
M3 POS SEN	J4/7	4/J
M1M1	8/A, B	J5/1
M1M3	8/C, D	J5/2
M1M4	8/E, F	J5/5
M1M5	8/H, J	J5/4
M2M1	9/A, B	J5/7
M2M3	9/C, D	J5/8
M2M4	9/E, F	J5/11
M2M5	9/H, J	J5/10
M3W1	7/20	J5/14
M3W2	7/18	J5/15
M3 RET	7/10	J5/13

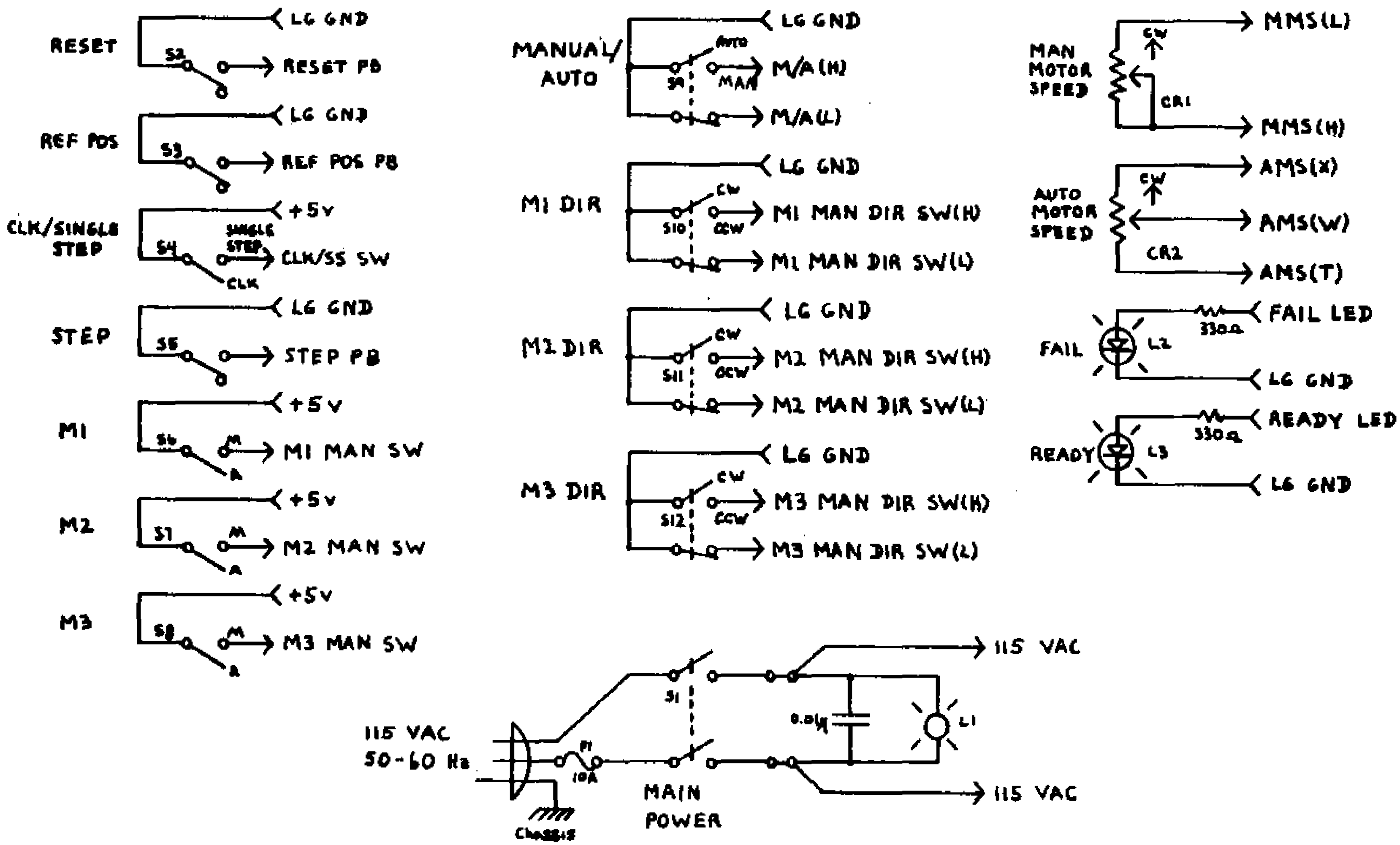
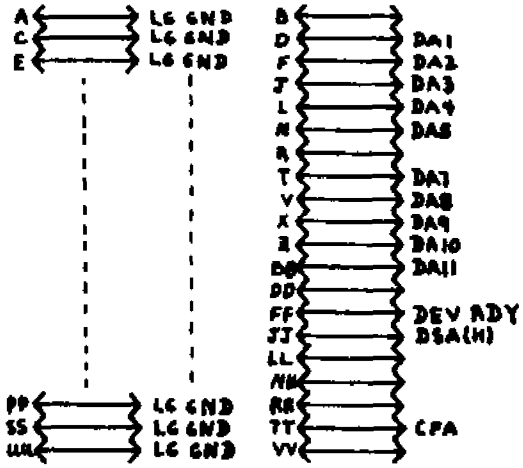
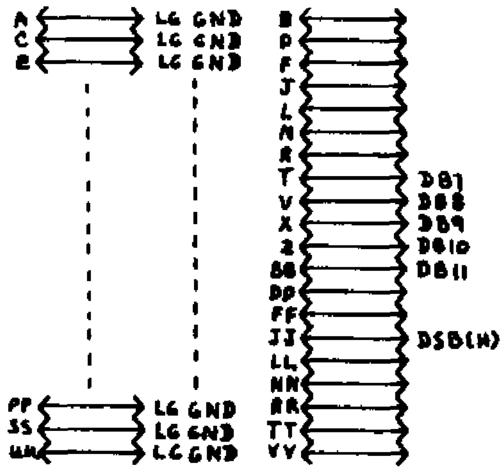


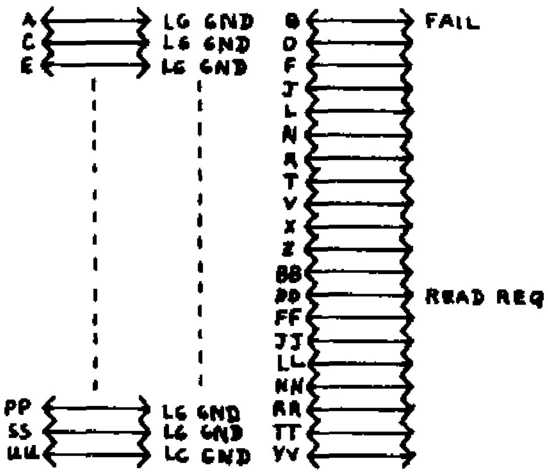
FIG. 35 - IES front panel connections



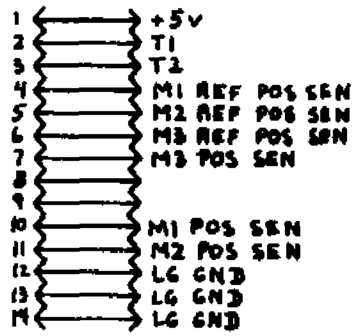
J1: 40 pin 3M connector wired for M1705A omnibus output cable



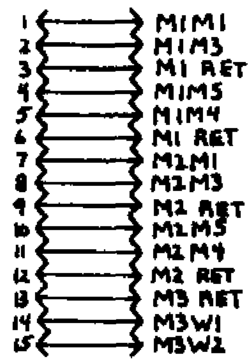
J2: 40 pin 3M connector wired for M1705B omnibus output cable



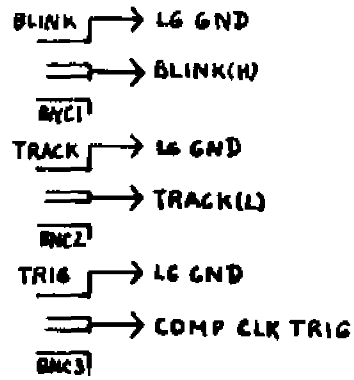
J3: 40 pin 3M connector wired for M1703 omnibus input cable



J4: 14 pin ribbon connector for VPS assembly position sensors



J5: 15 pin Cinch connector wired for VPS stepper motors



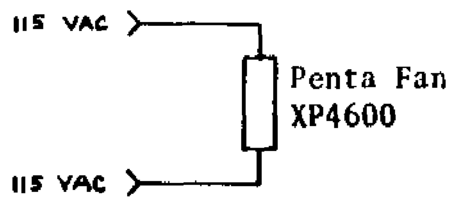


FIG. 36 - IES rear panel connections

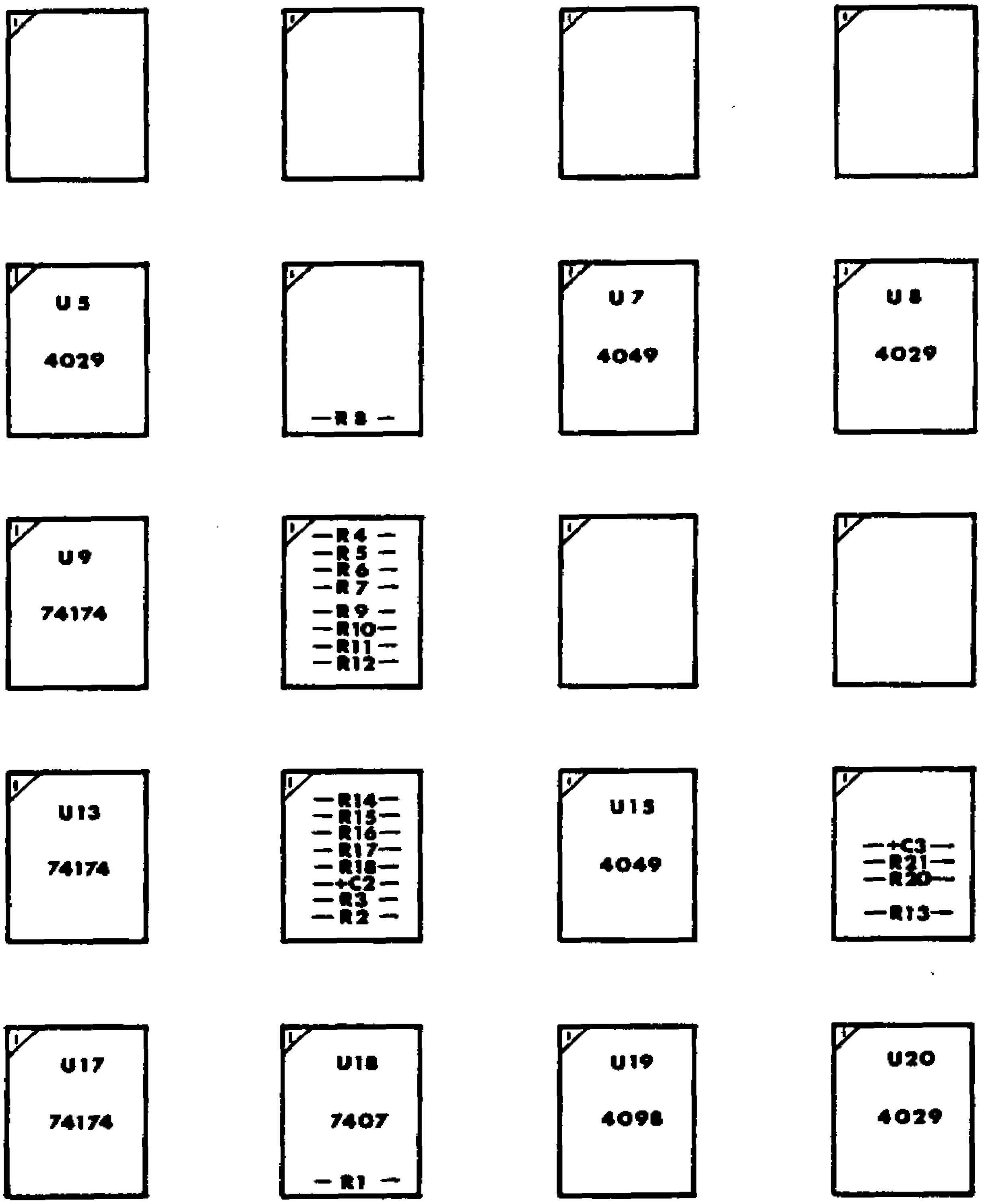


FIG. 37 - Card 1 layout, top view

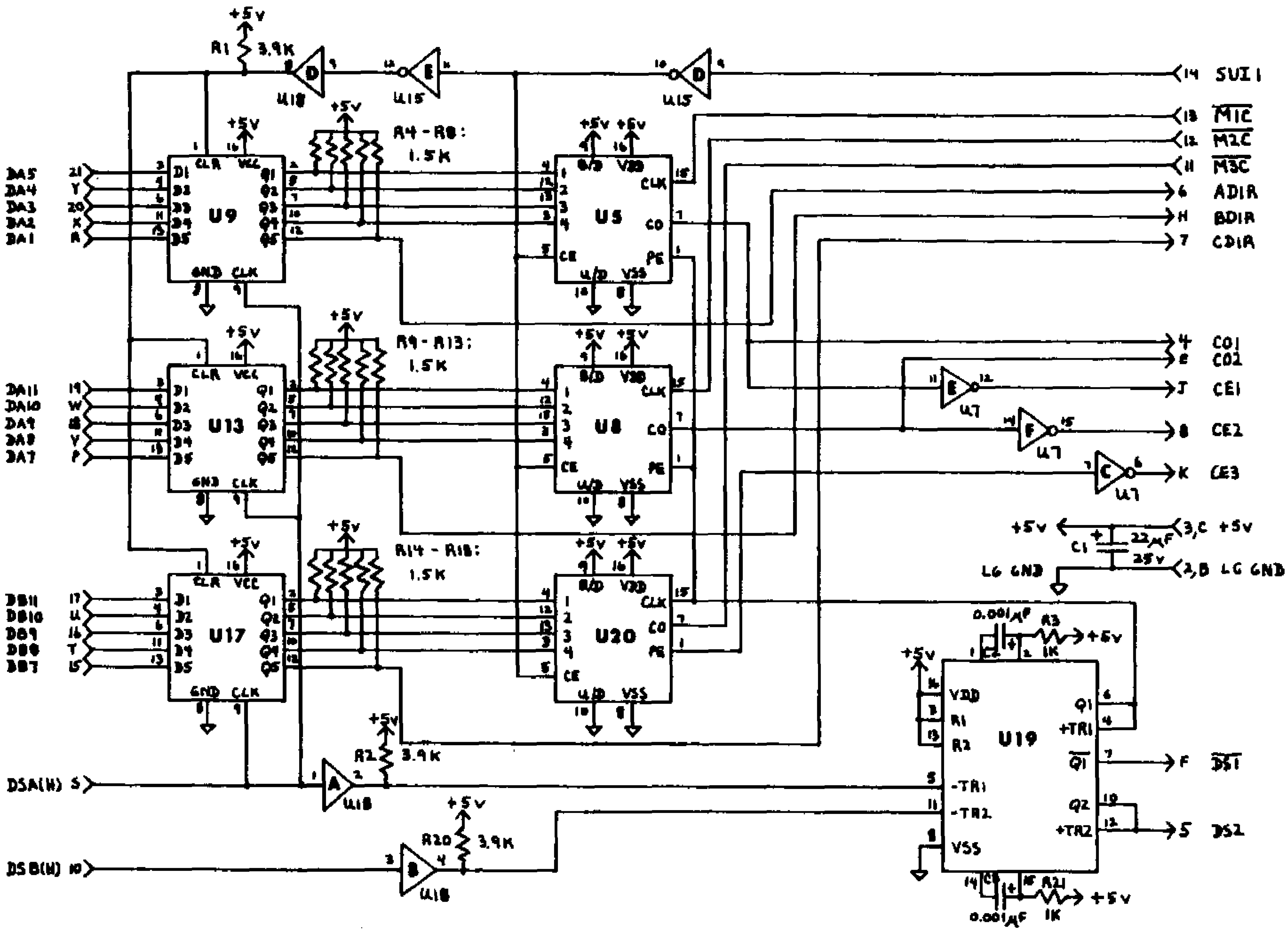


FIG. 38 - Motor step counters and registers, card 1

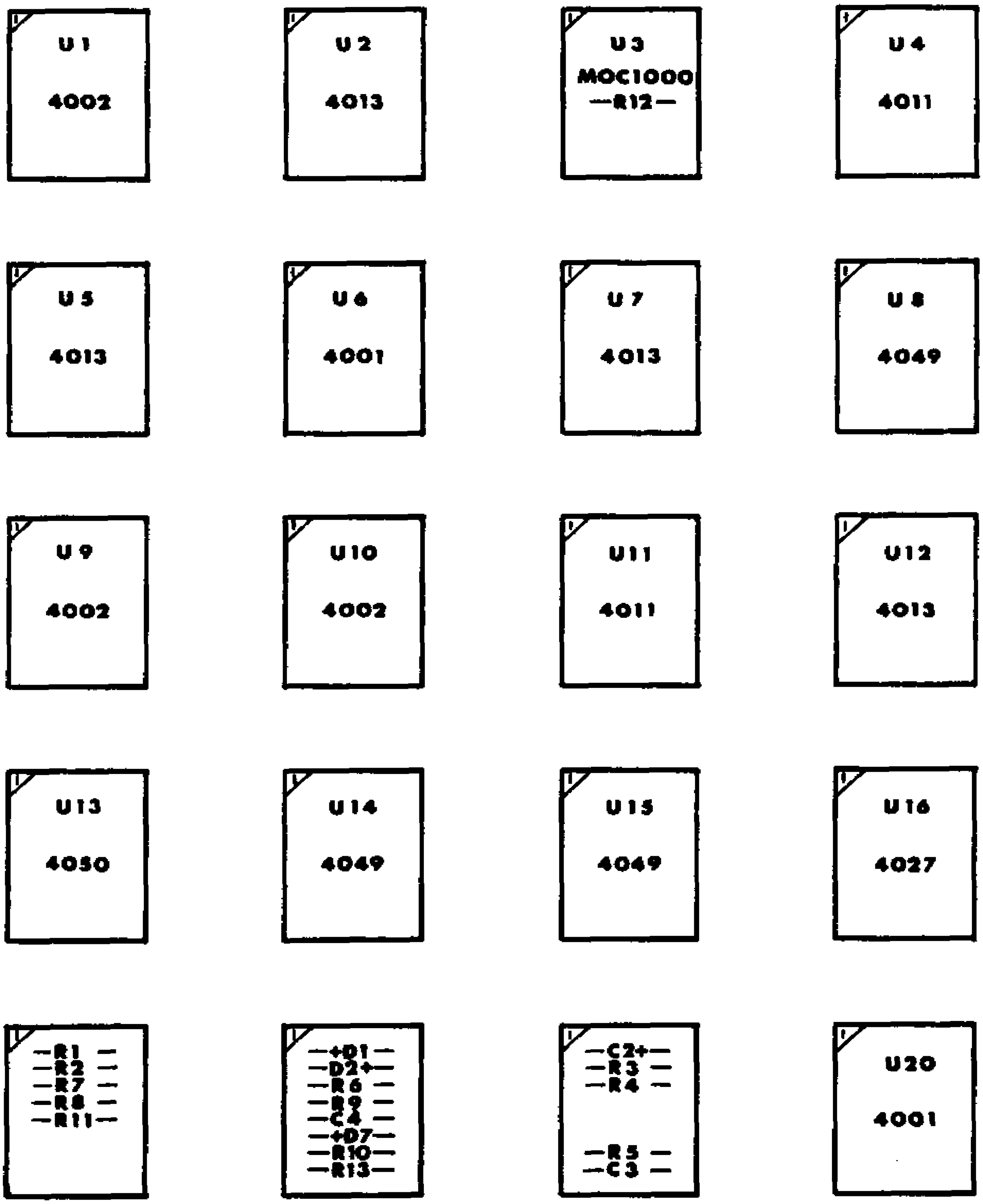


FIG. 39 - Card 2 layout, top view

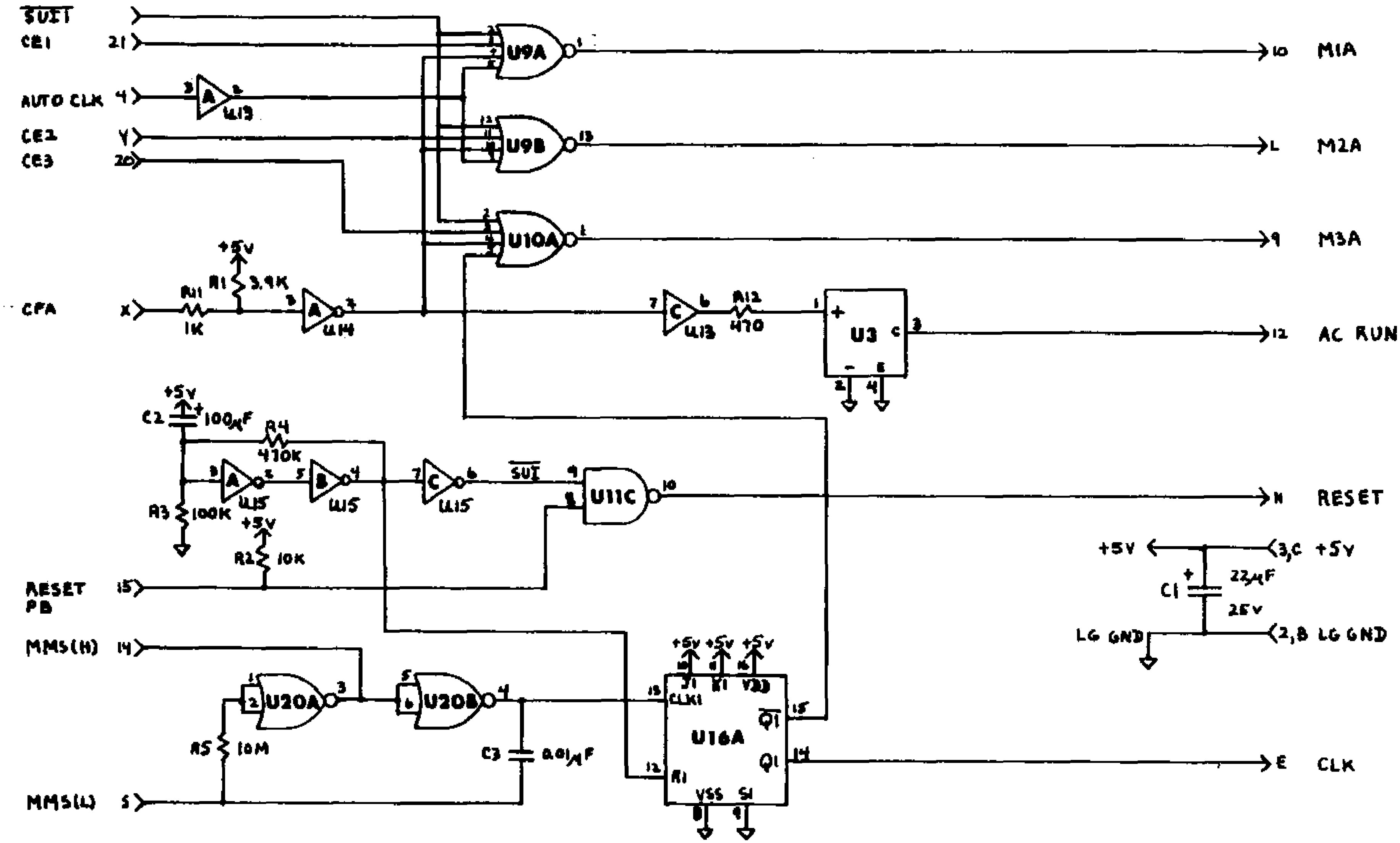


FIG. 40 - Start-up and motor flag control logic, card 2

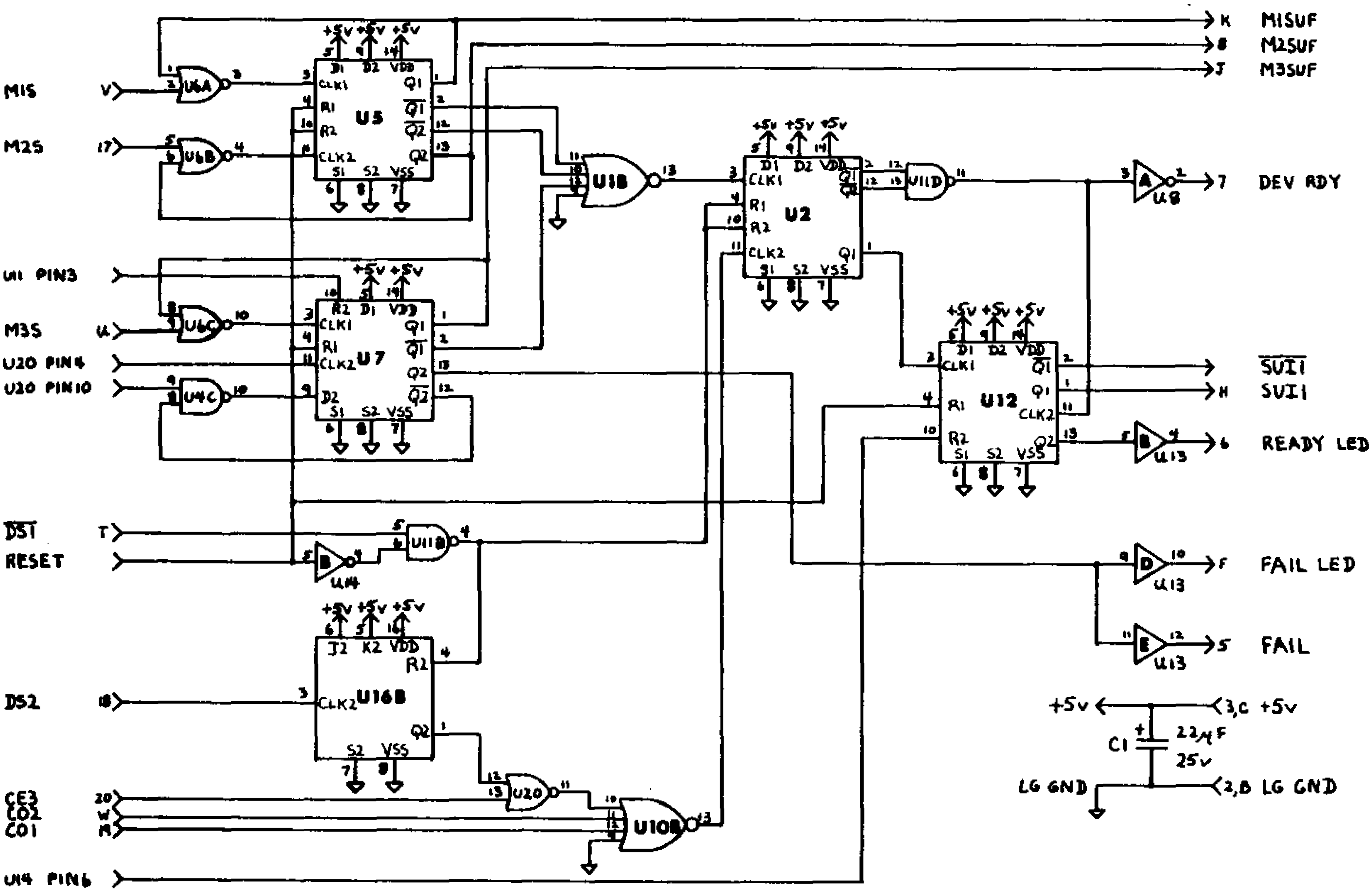


FIG. 41 - Start-up and motor flag control logic, card 2

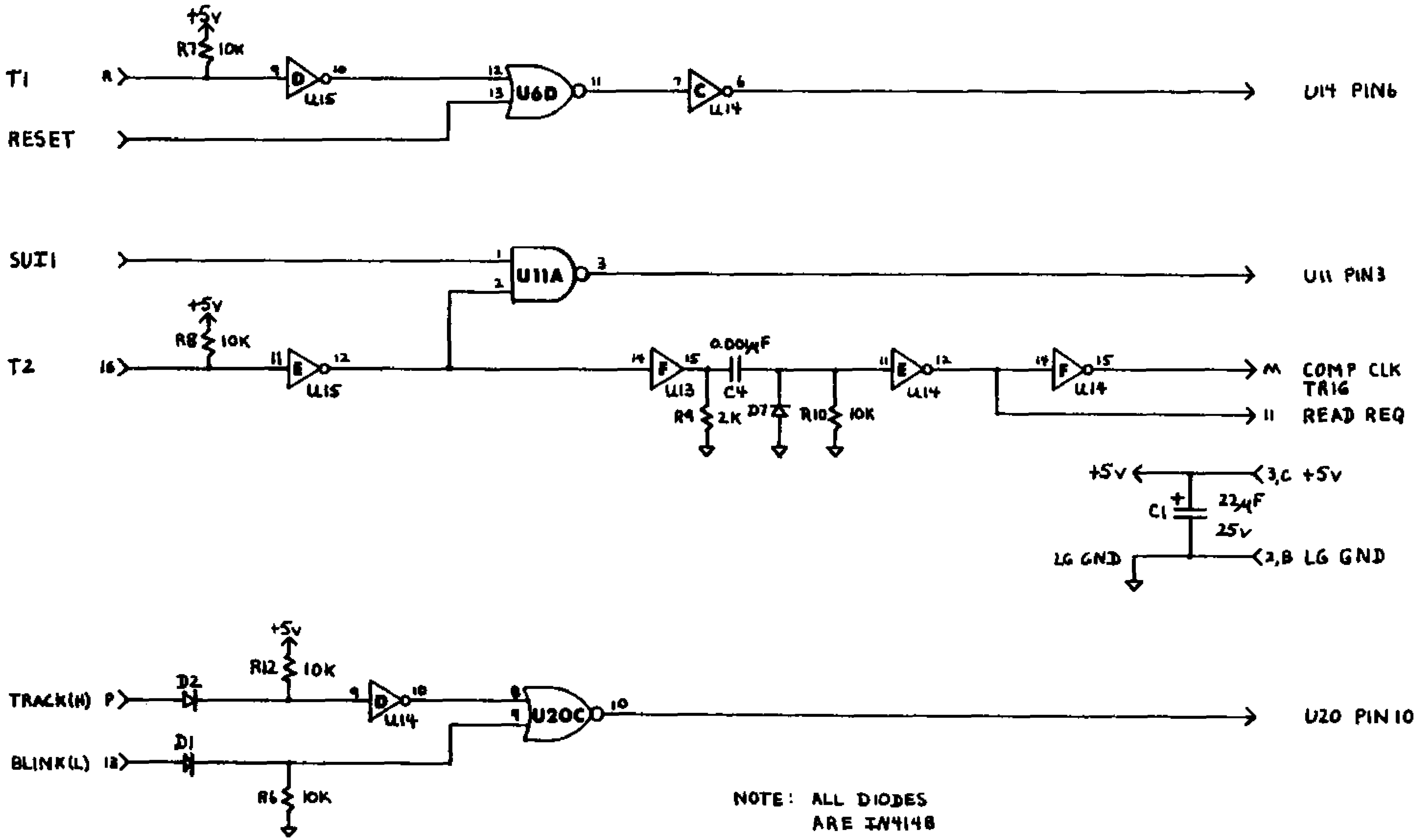


FIG. 42 - Start-up and motor flag control logic, card 2

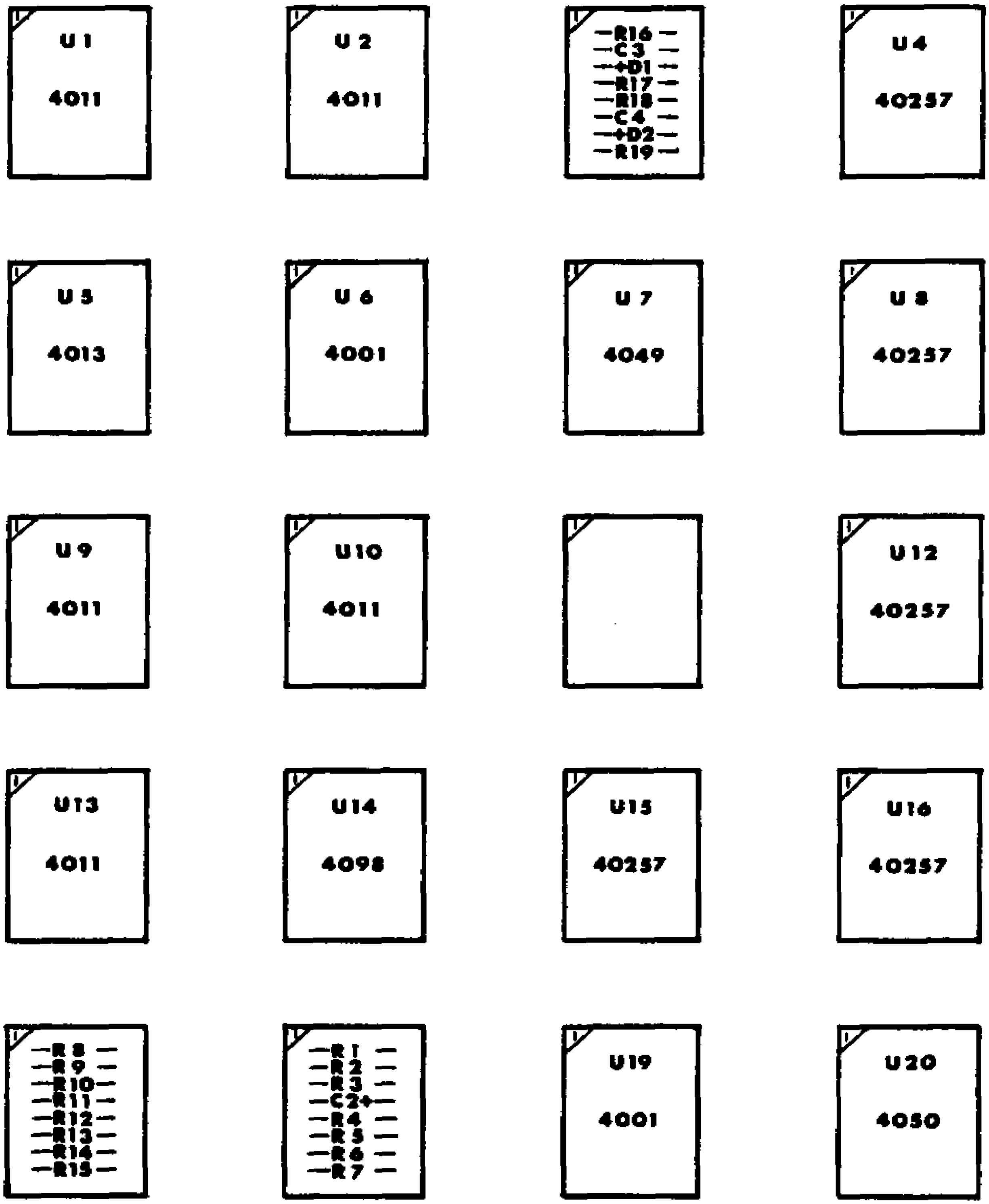


FIG. 43 - Card 3 layout, top view

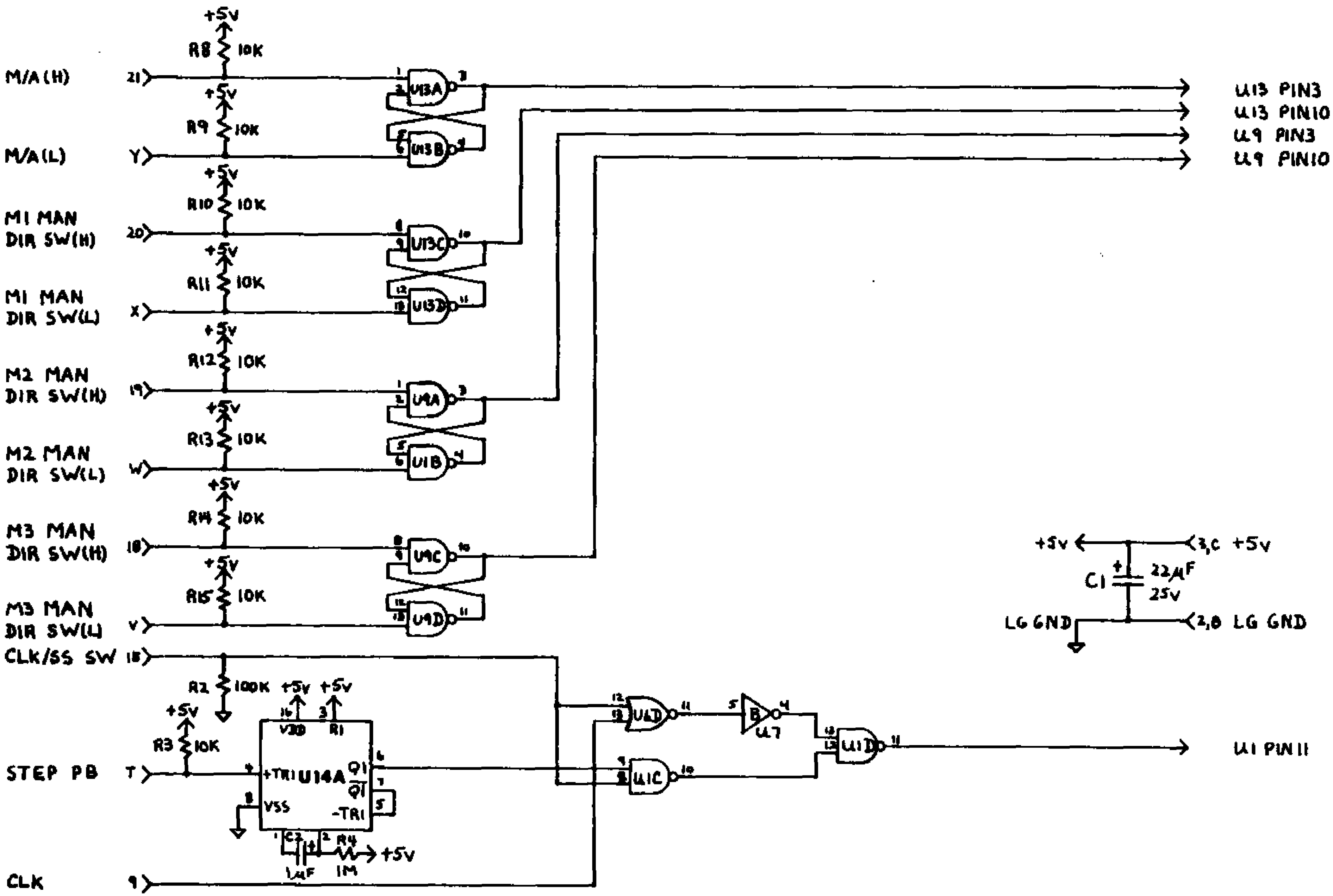


FIG. 44 - Manual/auto motor control mode logic, card 3

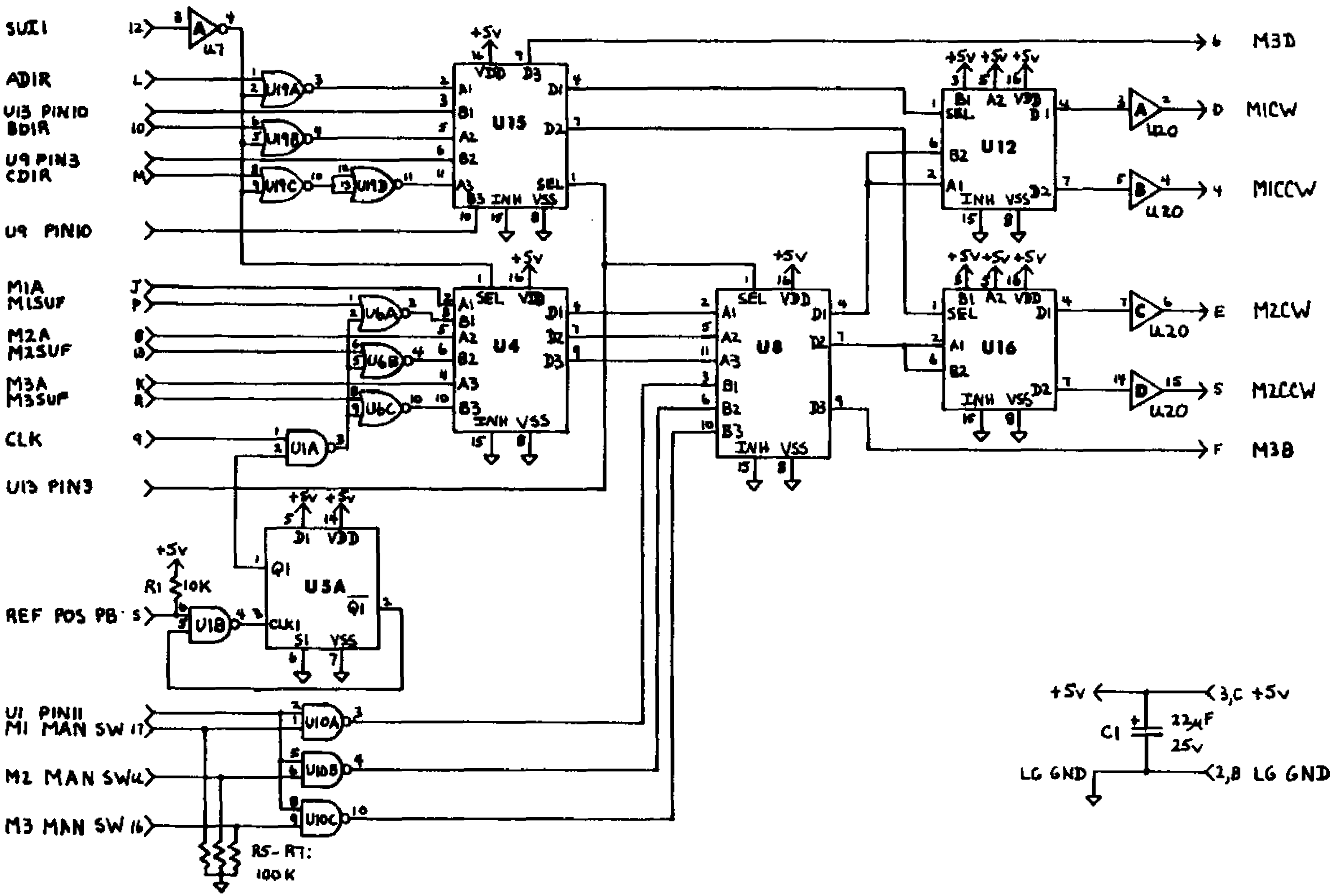


FIG. 45 - Manual/auto motor control mode logic, card 3

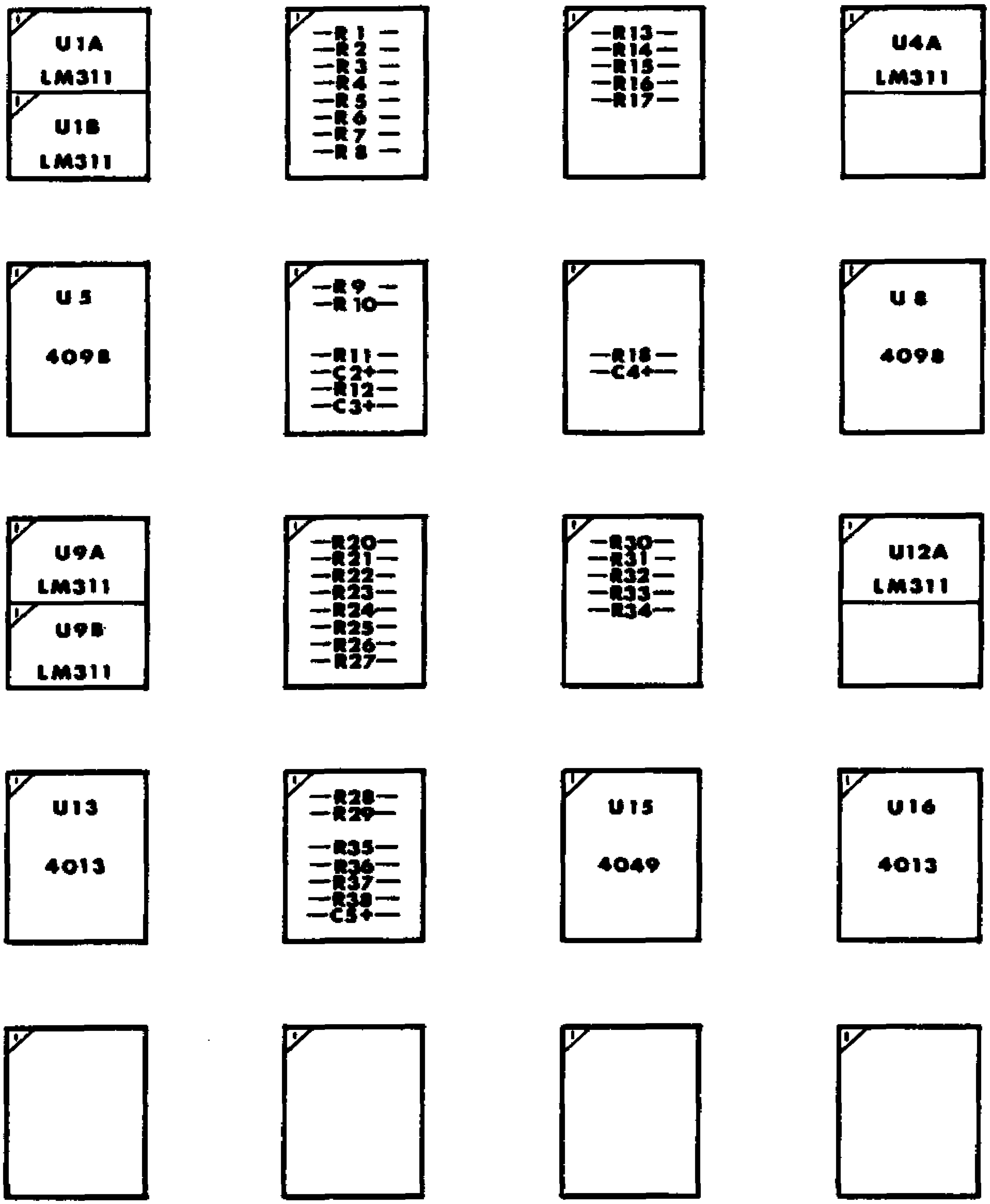


FIG. 46 - card 4 layout, top view

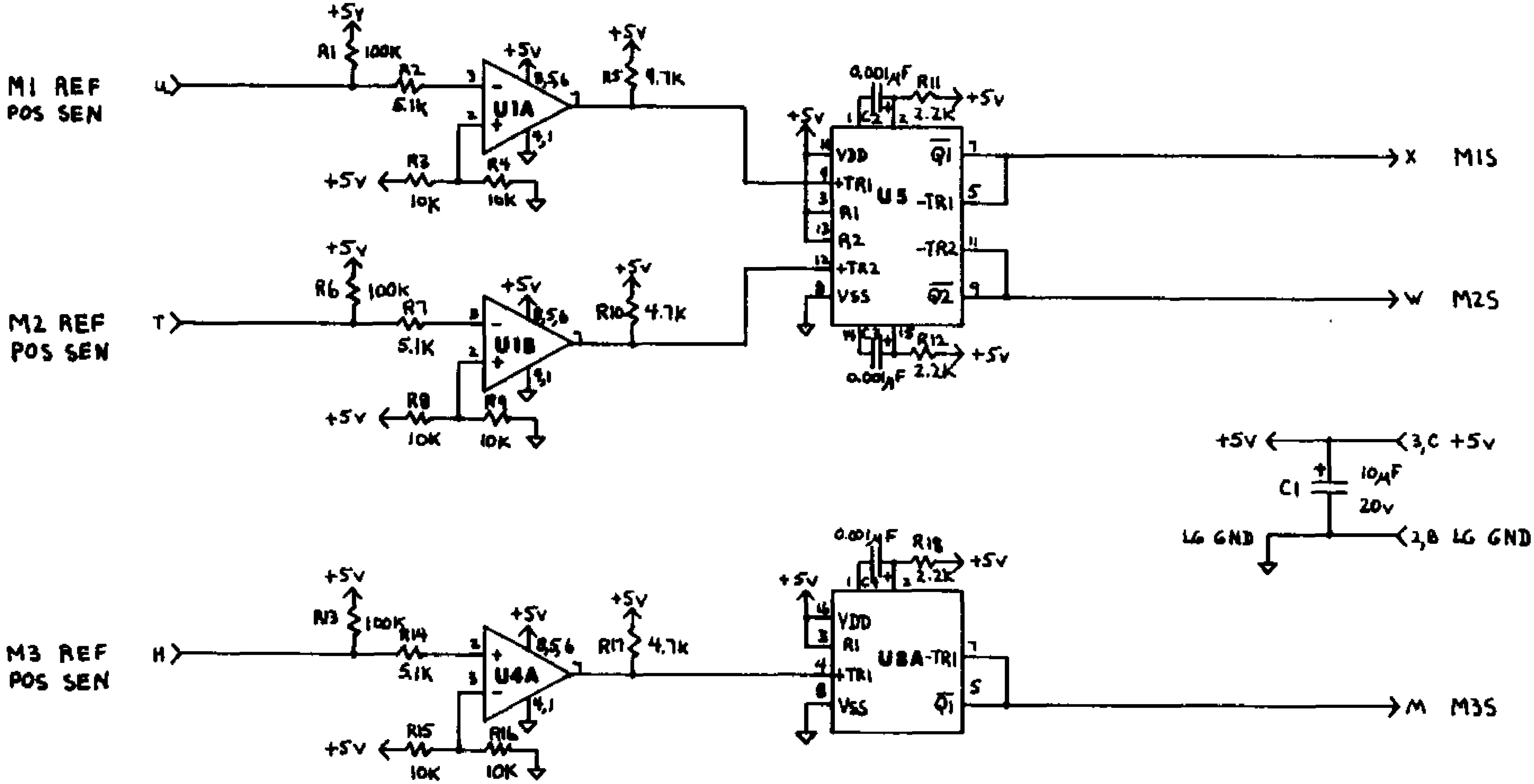


FIG. 47 - MOTOR 1, MOTOR 2, MOTOR 3 reference position sensor detection circuitry, card 4

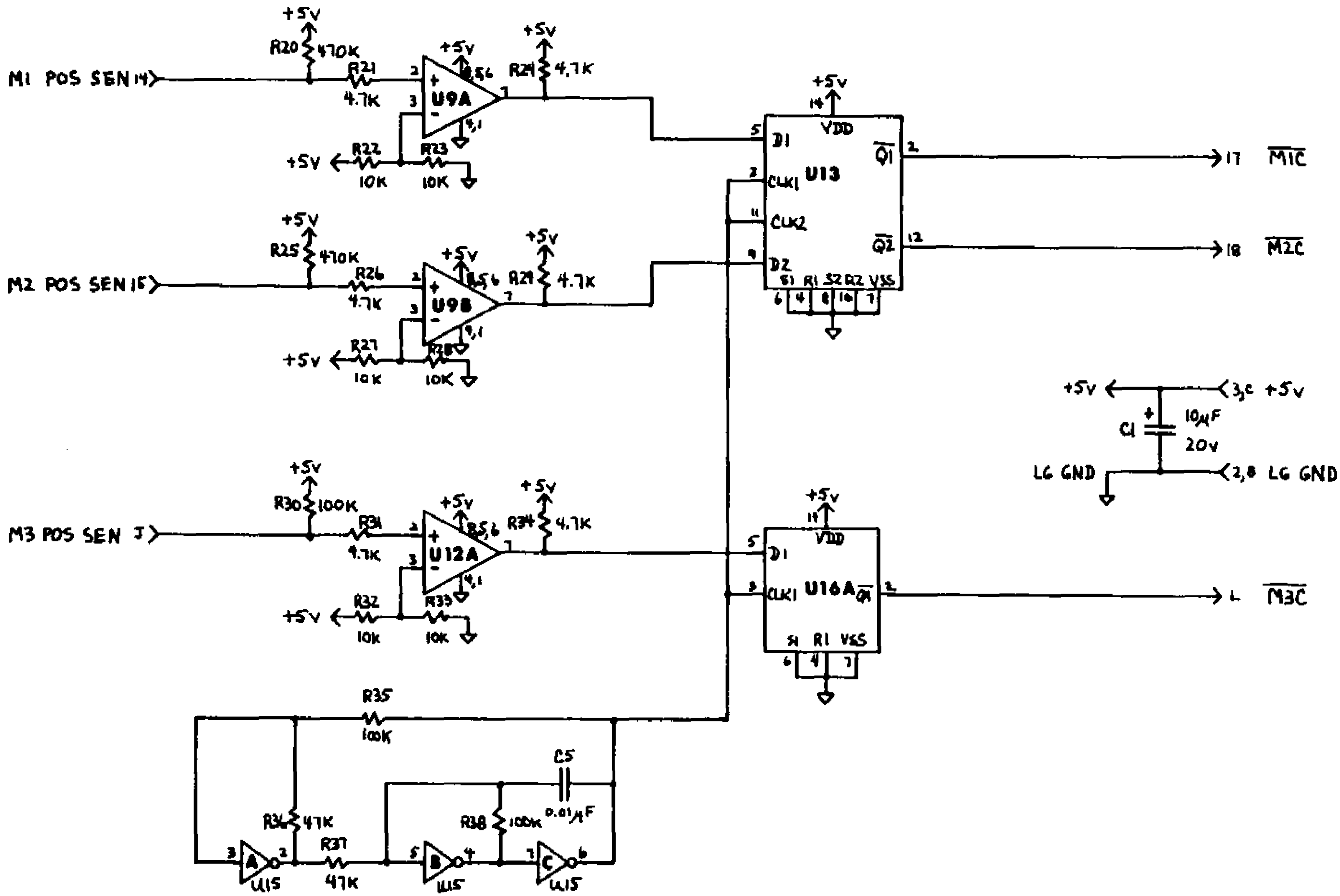


FIG. 48 - MOTOR 1, MOTOR 2, MOTOR 3 position sensor detection circuitry, card 4

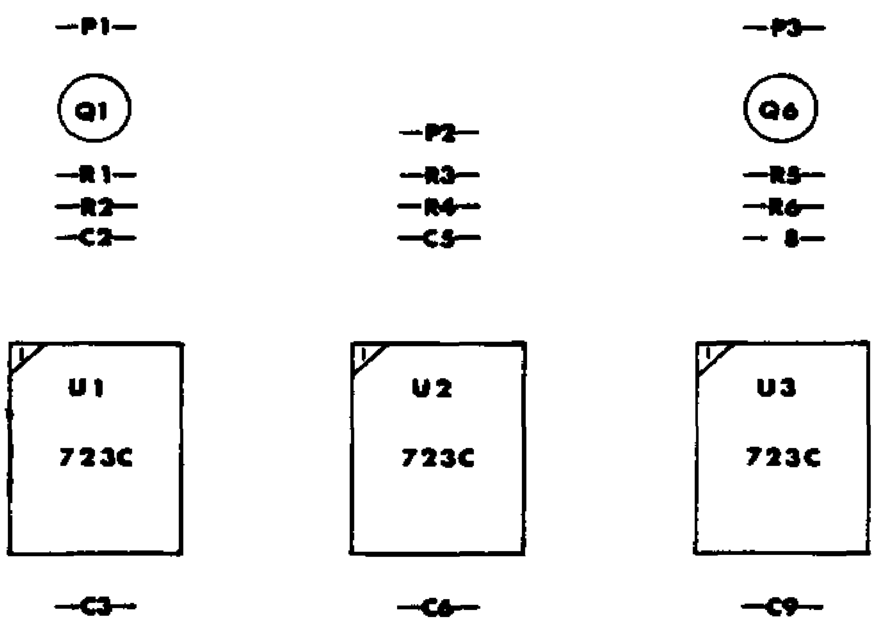


FIG. 49 - Card 6 layout, top view

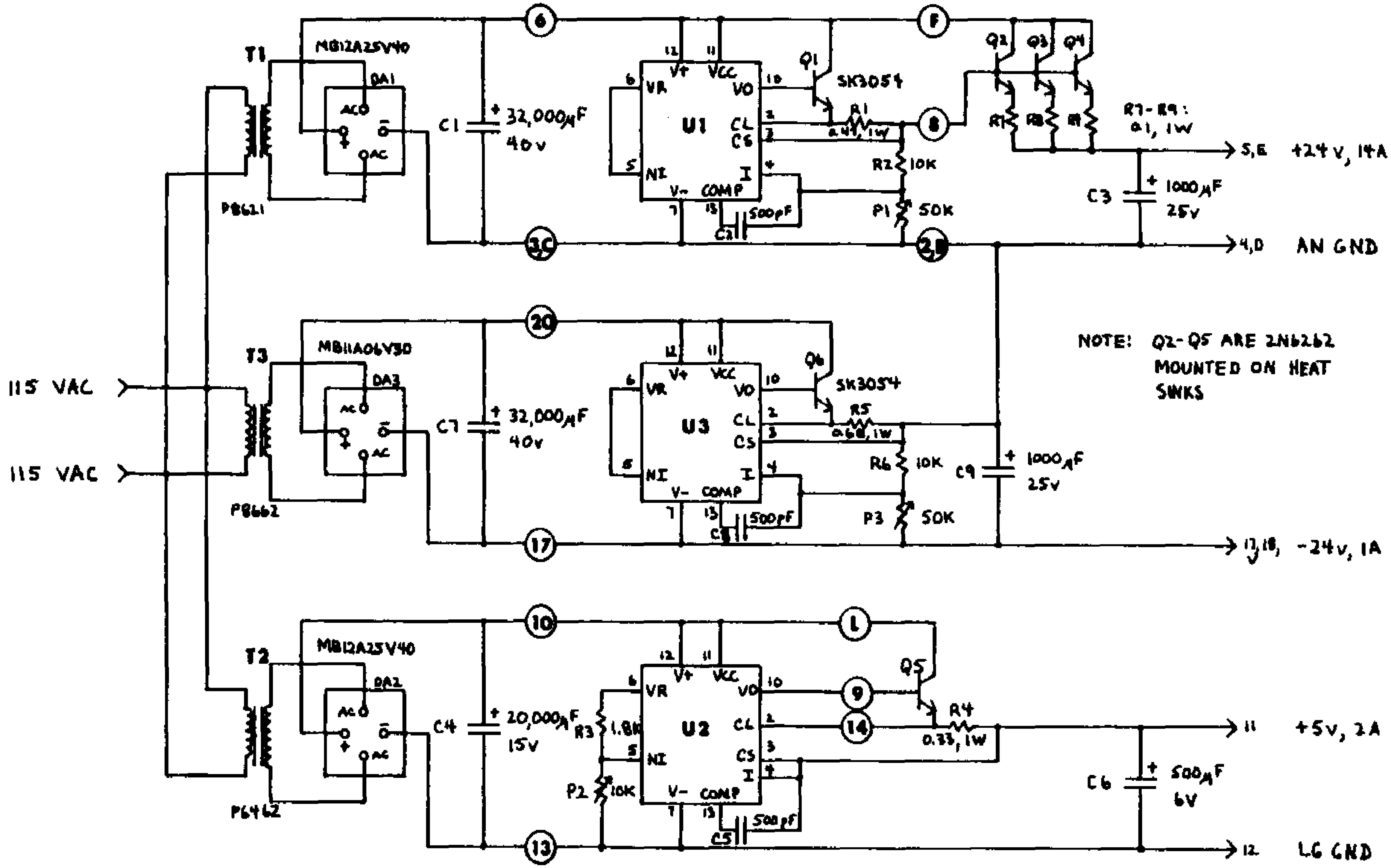


FIG. 50 - +24V, -24V, +5V power supply regulation circuitry, card 6

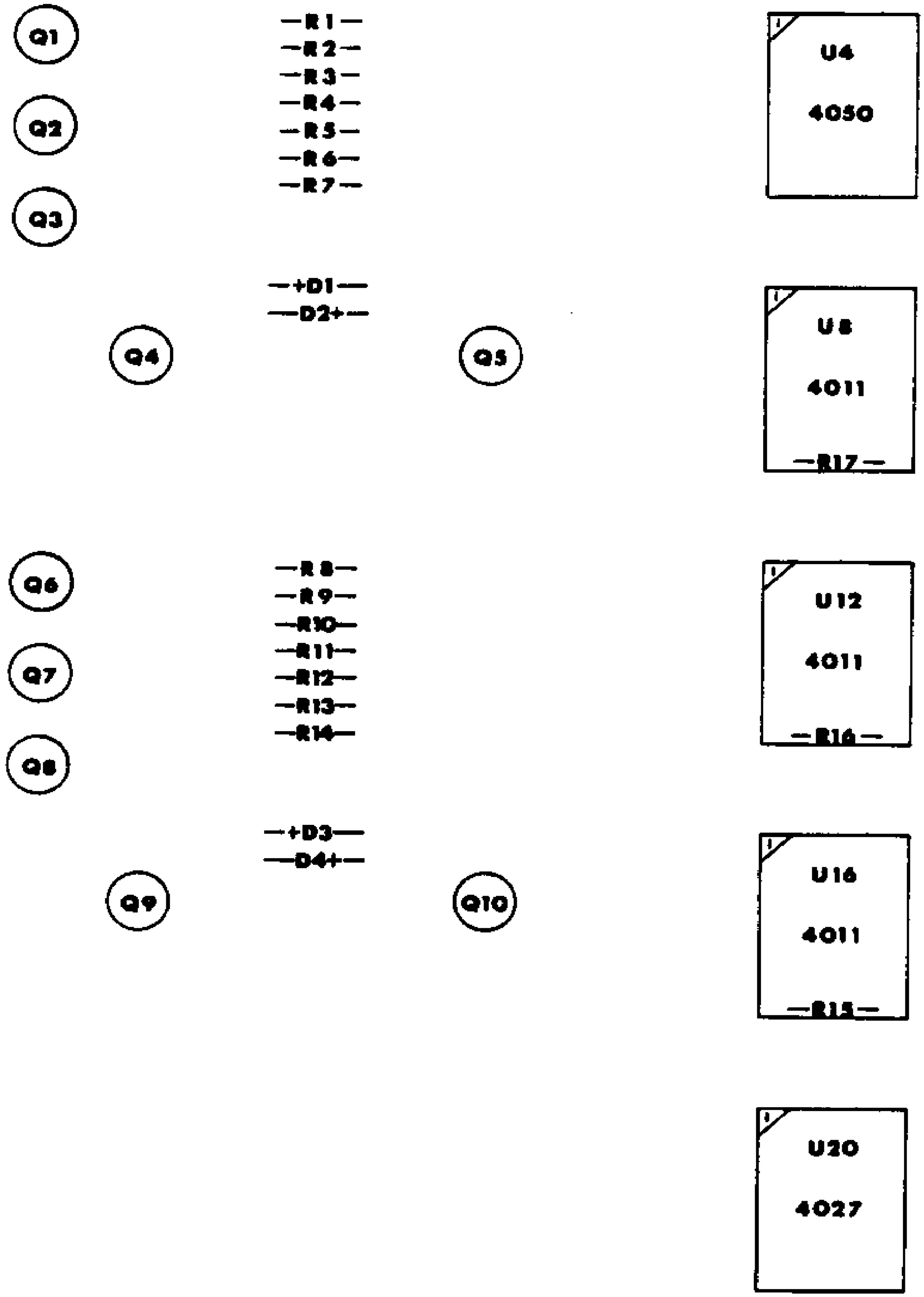


FIG. 51 - Card 7 layout, top view

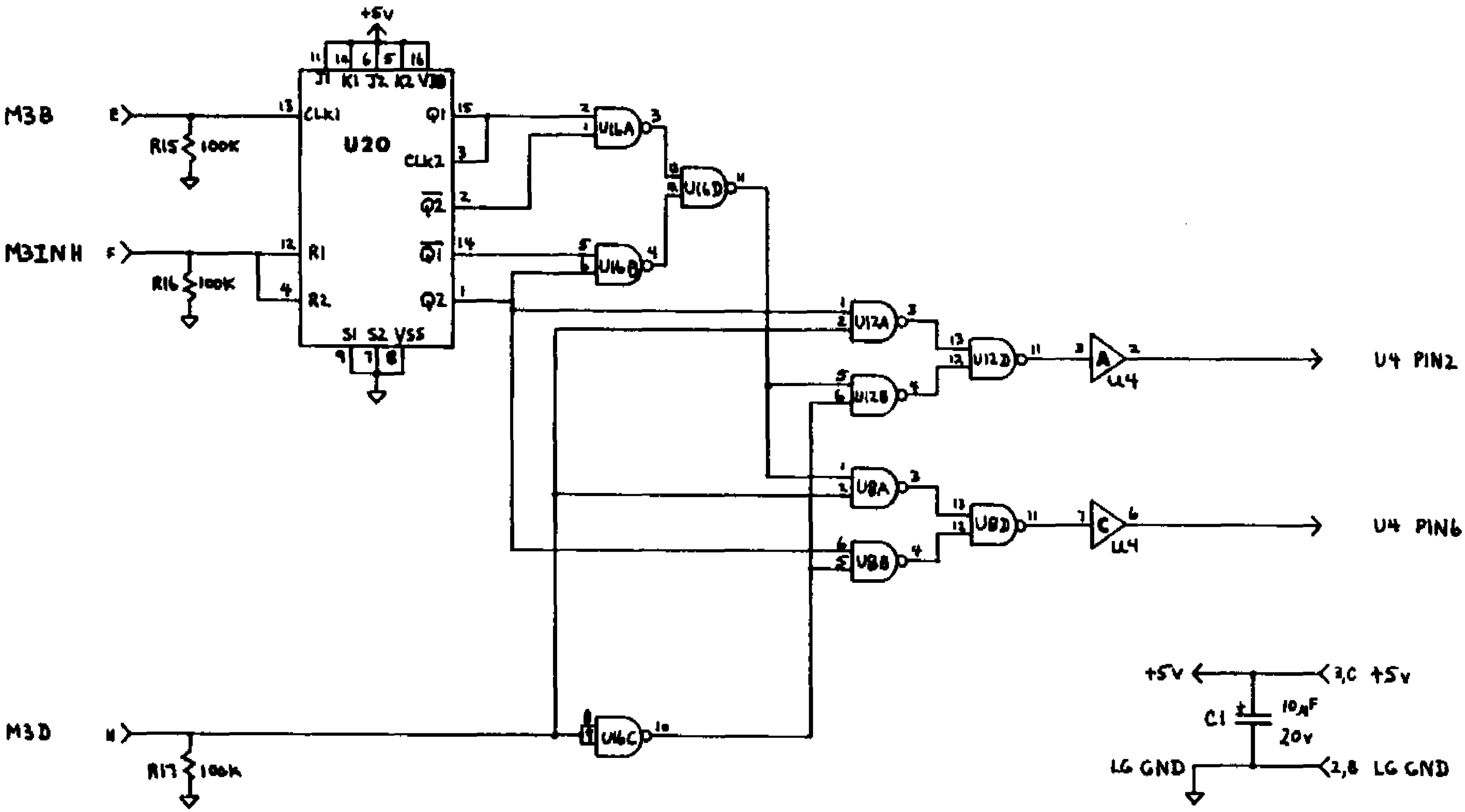


FIG. 52 - MOTOR 3 driver logic circuitry, card 7

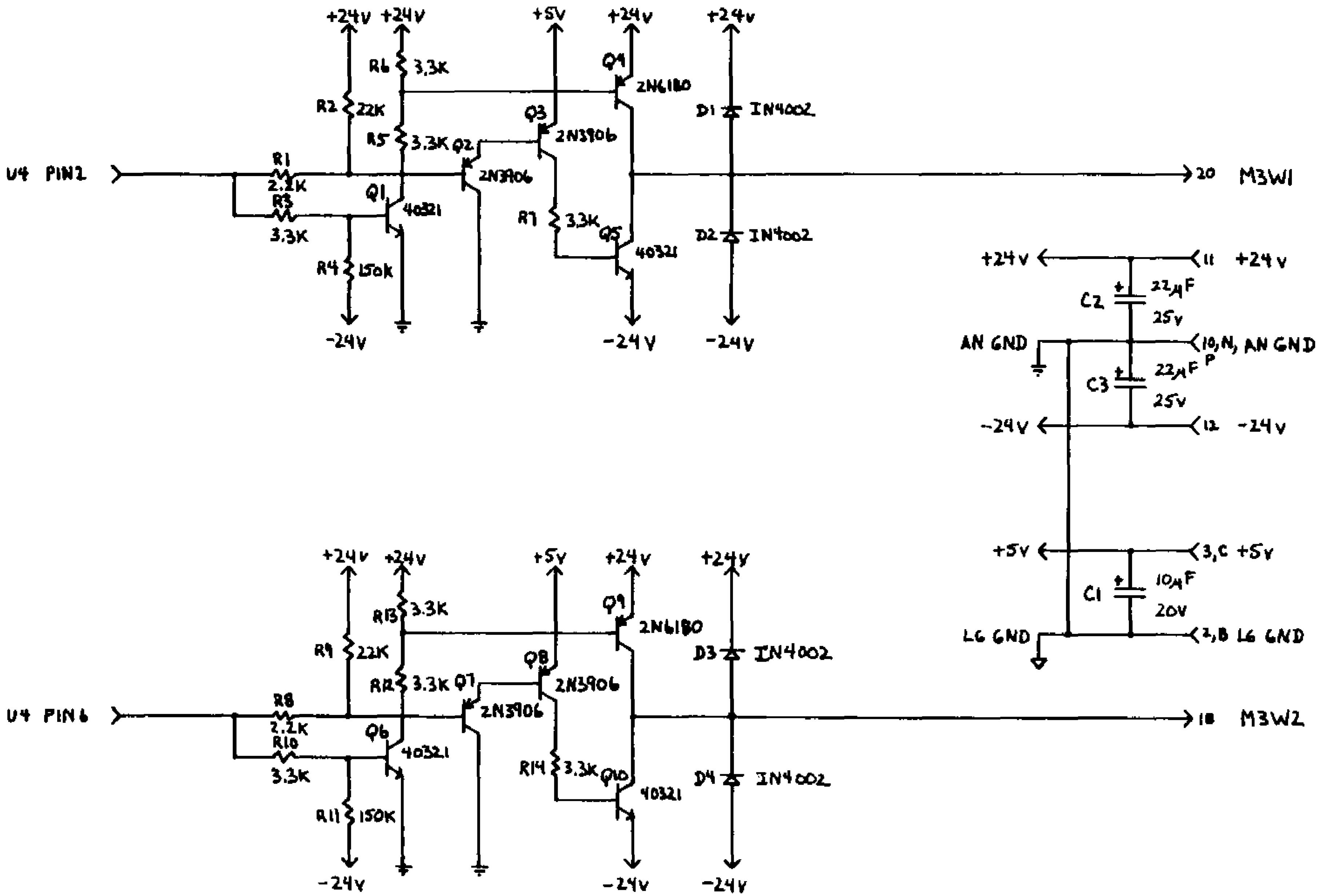


FIG. 53 - MOTOR 3 drive circuitry, card 7

APPENDIX B: SYSTEM SOFTWARE

Hardware

The saccade analysis system software was developed on and intended for use with a computer system consisting of the following hardware and software:

(1) A Digital Equipment Corporation (DEC) PDP-8/E computer equipped with the following DEC internal options:

- KM8-E memory extension and time share
- 8K total core memory
- DK8-EP programmable real time clock
- M865 teletype I/O control
- AD8-EA 10 bit analog-to-digital converter
- AM8-EA 8 channel analog preamplifier and multiplexer
- XY8-EA plotter control
- M1703 omnibus input interface (two required)
- M1705 dual channel omnibus output interface.

(2) DEC OS/8, version 3C, software operating system.

(3) Teletype Corporation model ASR-33 teletype.

(4) Sykes Datatronics, Inc. model 7250 dual buffered flexible disk system with PDP-8/E hardware I/O interface and OS/8 software handler package.

(5) Electronics Associates, Inc. model 130 digital X-Y dataplotter.

The interconnections of this computer system with the target presentation system (TPS) and the Cornsweet-Crane eyetracker are shown in Fig. 54 and outlined below:

(1) the M865 teletype control is the hardware interface between the computer and the ASR-33 teletype.

(2) The XY8-E plotter control is the hardware interface between the computer and model 130 data plotter.

(3) One M1703 input interface (internal device address is octal 14) is cabled to the plotter and used to transmit offscale signals to the computer for software management of a plotter offscale condition.

(4) One M1703 input interface (internal device address is octal 15) is cabled to the TPS interface electronics section so that the "failure" bit can be transmitted to the computer. The software checks this bit and rejects the trial data if the FAIL bit is set.

(5) The M1705 output interface (internal device address is octal 16 for channel A and octal 17 for channel B) is cabled to the TPS interface electronics section to allow for software control of the TPS target parameters.

(6) Eyetracker H4 and V4 (subject eyeball X and Y rotation) outputs are cabled to the input channels 0 and 1 respectively of the AM8-EA analog multiplexer.

(7) The TPS TRIG signal is cabled to one of the DK8-EP clock trigger inputs (EVENT 1) and is used to trigger the clock. The clock controls the A/D sampling of the eyetracker eye rotation signals.

(8) The PDP-8/E core memory, DK8-EP, M865, AD8-EA, AM8-EA, XY8-E,

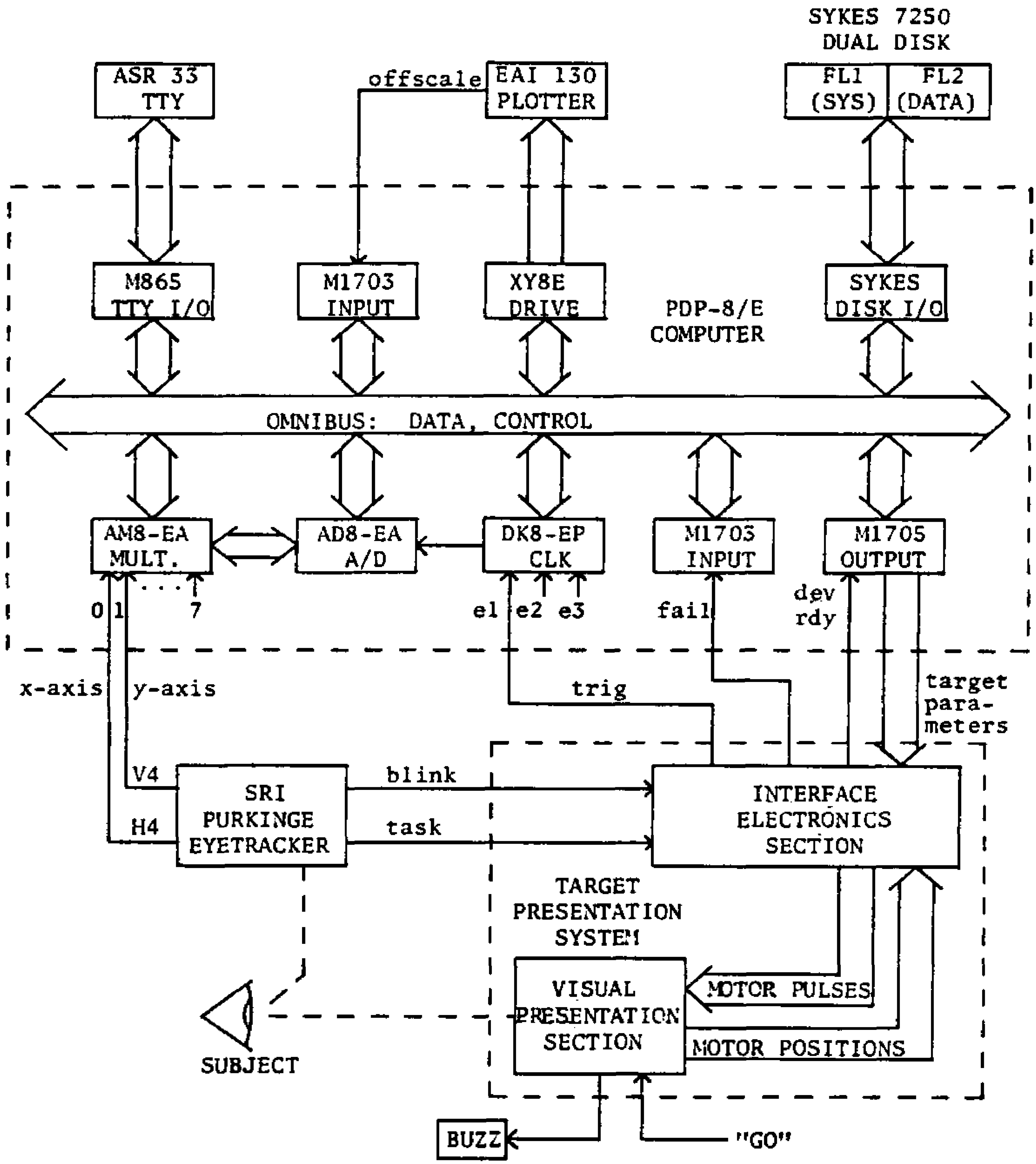


FIG. 54 - SACCADE ANALYSIS SYSTEM HARDWARE OVERVIEW

M1703's, M1705, and disk I/O interface all communicate internally by means of the PDP-8/E omnibus. Detailed descriptions of all computer hardware operation may be found in the manufacturer's publications as listed in the reference section.

The DEC OS/8 software system and Sykes dual disk drive allow for creation, manipulation and storage of a variety of data and program files. OS/8 supports standard FORTRAN II high level programming language which may be intermixed with DEC SABR assembly language instructions. This gives the user a highly versatile overall programming "language". The OS/8 system also includes a variety of text editing, file manipulation, debugging and general utility programs best referenced by the user in the DEC publications listed in the reference section.

In normal usage the OS/8 system software and saccade analysis system software most frequently used reside on disk drive FL1. This leaves disk drive FL2 free for data file storage and manipulation.

Software Overview

The software of the saccade analysis system can be broken down into three functional groups: input and control software, processing software, and output software as shown in Fig. 55. Program package ADCOM controls the TPS and controls sampling of analog data from the eyetracker, program package SPROUT processes saccade data files produced by ADCOM, and program packages APLOTØ, SSPCOM, and SACCOM output raw or processed saccade data to the plotter or teletype. All program packages are structured to have teletype command decoders to allow the user easy control and interaction with the software.

Program package ADCOM allows the user to control the TPS target parameters automatically and control the sampling of the analog output voltages of the eyetracker. All digitized data produced by ADCOM is stored in data files on disk drive FL2. Sampling parameters are user alterable.

Program package APLOTØ is a general purpose X-Y plotting program package designed to operate with ADCOM-produced data files as input. APLOTØ allows the user to plot previously sampled analog data in a variety of ways. The sampled data from any analog input channel may be plotted as a function of time or as a function of any other sampled channel. Scale factors are user selectable and a "slow motion" plotting feature has been included.

Program package SPROUT processes saccade data files created by ADCOM. SPROUT accepts data files from disk drive FL2 each of which consist of two channels (0 and 1) of sampled data representing a

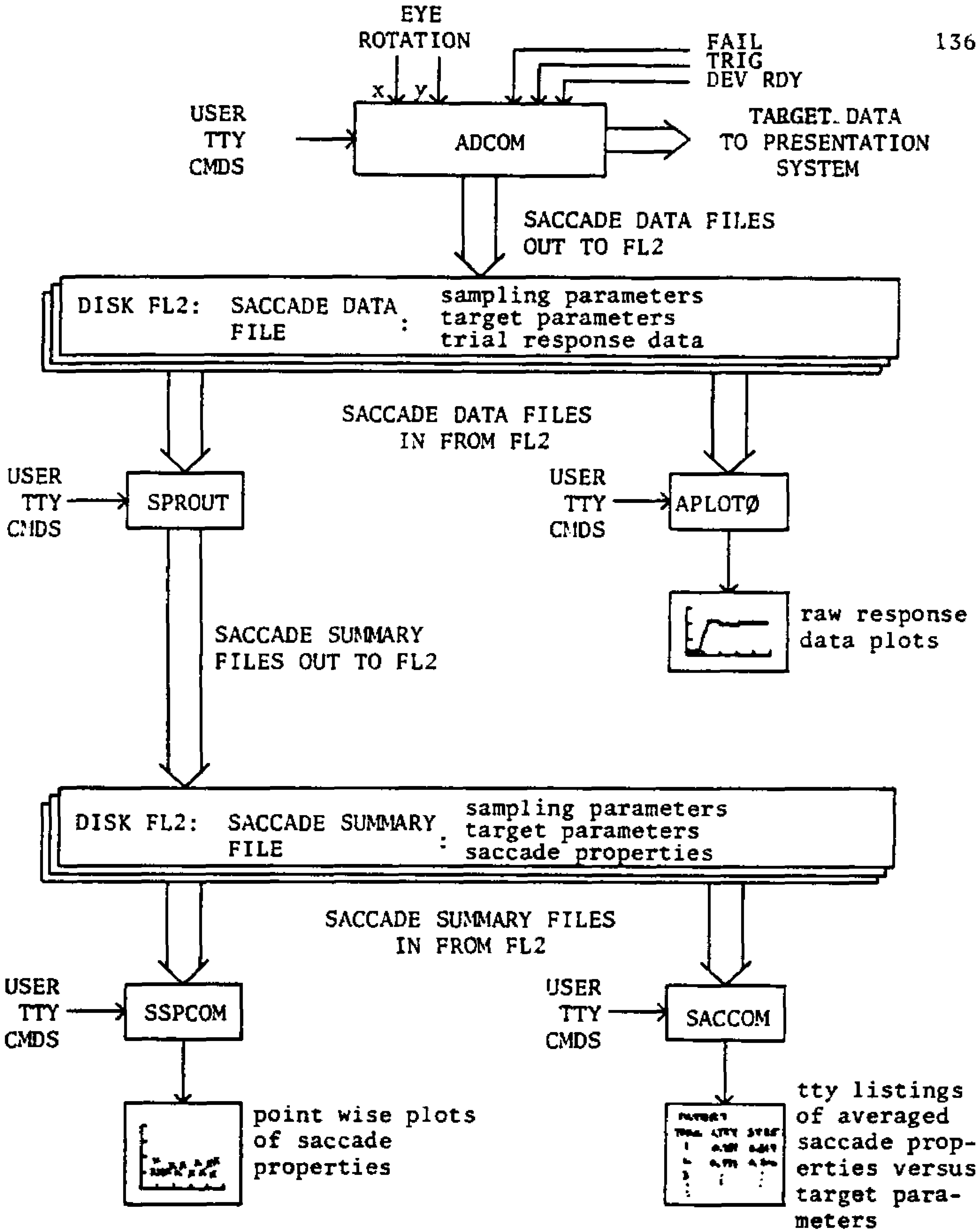


FIG. 55 - SACCADE ANALYSIS SYSTEM SOFTWARE OVERVIEW

subject's eye movement response to a step-like target deflection lasting about 3 seconds. SPROUT calculates saccade latency and settling times to four radii about the final target position for each input saccade data file. SPROUT will automatically input and process up to 63 saccade data files with no user interaction and store the results in a single "saccade summary file" on disk drive FL2. This provides an efficient and easily accessed method of storage for data about many subject saccade responses to target step-like movements.

Program package SSPCOM is a special purpose plotting program designed to accept SPROUT-created summary data files. SSPCOM allows the user to plot summary data in a variety of convenient ways with full control over plot parameters.

Program package SACCOM is a special purpose averaging program designed to accept SPROUT-created summary data files. SACCOM sorts and averages summary file data according to target parameters. Results are printed in an array on the teletype. SACCOM will automatically input and process up to 99 saccade summary files with no user interaction.

The design of all programs discussed here was based on earlier versions as written by Dr. James Brown. Program package APLOTØ is the only program package that has been left unchanged by this author. The reader is referred to Dr. Brown's doctoral dissertation for further details of APLOTØ. The reader will also be referred to Dr. Brown's dissertation for certain details at times in the following discussions. Program packages ADCOM, SPROUT, SSPCOM, and SACCOM are discussed in detail in the following sections. Program listings are included with each discussion.

General Design Considerations

All program packages of the saccade analysis system were designed with the following primary considerations in mind: (1) that the programs be flexible and allow the user easy interaction with the software, (2) that the parameters used in the programs be user selectable in a convenient manner, (3) that a minimum amount of user interaction with the software be required during actual experimental or processing intervals.

To meet the above design considerations all program packages were written in a similar format. This format is as follows: (1) User interaction with the software is provided for by including a list of teletype keyboard commands with each package. (2) All user selectable parameters are provided with convenient default values. (3) All program packages have an initial "header" which prints the available commands and initializes parameters to their default values. (4) All program packages enter a command input and interpretation mode after the "header" to allow the user to enter commands. (5) Automatic modes of program operation are available to minimize the amount of user interaction with the software during experimental or processing intervals. This allows the user to concentrate attention on other aspects of experimental procedures and thus reduces the amount of experimental time with the subject. (6) All commands cause a teletype request for specific information. A command is also included in some cases to list all pertinent parameters as they currently exist within the software. (7) Error checks are provided to prevent the user from entering meaningless commands or parameter specifications.

The various program packages are all discussed in detail in the following sections.

<u>ADCOM: Target Parameter Control, Analog Data Input and Storage Program Package</u>

Design Considerations:

Program package ADCOM is designed to control computer interaction with the target presentation system (TPS) and the Cornsweet-Crane eyetracker. ADCOM selects the target parameters to be presented by the TPS and calculates the number of movements needed for each of the TPS stepper motors. ADCOM provides this data and certain control signals to the TPS interface electronics section to move the stepper motors to their required positions. ADCOM then reverts to a sampling mode to sample and store the analog eye rotation data in digitally coded form on the flexible disk storage medium.

ADCOM has the following features: (1) All sampling parameters--number of channels, sampling time, and number of samples--have initial default values which are convenient for the work done here. (2) Once sampling has been triggered it may be stopped at any time with an immediate return to command mode. (3) A sampling recycle mode is provided to allow the user to trigger one sample set after another. Disk file storage file number incrementation, and target parameter selection occur automatically. A randomized array of the 40 possible target positions is provided so that each position will be utilized only once during a recycle mode experimental session. This means that a response for each target position can be gathered in a single experimental session and at the same time the subject cannot guess the target position to be presented in a particular trial. (4) The sampling and target parameters of the particular trial are stored along with the sampled response data. (5) At the end of sampling the FAIL bit of the TPS is checked. If set, the entire sample set is rejected and no disk storage of that set occurs. (6) ADCOM allows the user to design an experiment such that many trials may be accomplished quickly.

Specifications and features:

Analog multiplexer input voltage any channel:	± 1.0 volts
A/D converter step size:	1.95 millivolts (1/512 volts)
Multiplexer input channels:	8 (0 to 7)
Sampling intervals available:	1, 2, 5 milliseconds times 10^0, 10^1, 10^2, 10^3 or 10^4.
Maximum total samples per sample set (one disk data file):	1020 (samples per channel x channels per sample set).
Starting modes:	Sampling is started by any or all of three TTL level Schmitt trigger inputs (Event 1, 2, 3).
Target positions available:	40 positions divided into five possible radii (2.0, 2.5, 3.5 and 4.0 degrees) and eight possible azimuthal angles (0, 45, 90, 135, 180, 225, 270 and 315 degrees).
Filter positions available:	9 (0 to 8)
Control sequence:	Computer selects next sequential target position from target array, calculates a random number from 0 to 8 for filter position, and then calculates the TPS motor moves required from last position to this new position. This information is sent to the TPS and motor movements are initiated when CFA of M1705 is set. The computer then waits for the movements to finish (DEV RDY flag from TPS). After TPS signals DEV RDY computer advances to sample loop and waits for a Schmitt trigger (TRIG signal from TPS).
Sampling sequence:	When a Schmitt trigger is fired, the specified starting channel is immediately sampled. The multiplexer then increments to the next highest channel number, samples that

channel and so on until the specified number of channels has been sampled. After this the program waits for the specified sample time, then repeats the above sampling sequence. Samples are stored in sequential order in an array.

User stop: The sampling sequence may be stopped at any time by pressing the teletype "S" key.

Output data filenames: Must be six characters long, first four alphanumeric, last two numeric, i.e., DJST00 or HJ4$23. Data files are output to FL2.

Sampling modes: One shot: One set of samples is taken, the data is output to FL2 as a data file with the specified filename, the program returns to command mode.

Recycle mode: One sample set is taken, the data is output to FL2 as a data file with the specified filename. The last two digits of the filename are incremented, the target position just used is inhibited. The next target position of the array is chosen and set into the TPS. The program waits for the next sample set to be triggered. Up to 40 data sets may be taken and stored with no user interaction other than a starting trigger pulse for each sample set.

Automatic data rejection: Regardless of sample mode, after the last sample in the set is taken, a failure bit is checked. If set, the entire sample set is rejected and no disk storage occurs. No file number increment and target position inhibition is done.

Sampling parameter default values:

Number of channels: 2
Starting channel: 0
Number of samples per channel: 500
Sampling time: 5 milliseconds
Enabled Schmitt event: 1
All target position enabled

ADCOM Commands:

Commands to ADCOM may be entered on the teletype keyboard whenever ADCOM is in command mode. This is indicated when a colon is printed at the teletype left margin. Commands are: AC, AI, AS, AT, AM, AE, AF, AP, AD, AO and AR. Command functions are outlined below.

AC: Command AC causes a request for user entry of the number of analog channels to be sampled. Valid entry is an integer from 1 to 8. Control is returned to command mode after the computer reads the user entry.

AI: Command AI causes a request for user entry of the starting analog channel number for sampling. Valid entry is an integer from 0 to 7. Control is returned to command mode after the computer reads the user entry.

AS: Command AS causes a request for user entry of the number of samples per channel. Valid entry is an integer from 1 to 1020. The number of samples per channel times the number of channels must be less than or equal to 1020. Control is returned to command mode after the computer reads the user entry.

AT: Command AT causes a request for user entry of the sampling timebase 1, 2 or 5 milliseconds. Valid entry is the integer 1, 2 or 5. Control returns to command mode after the computer reads the user entry.

AM: Command AM causes a request for user entry of the sampling timebase multiplier. Valid entry is an integer from 0 to 4. The sampling time will be (timebase) x (10 to the timebase multiplier power). Control is returned to command mode after the computer reads the user entry.

AE: Command AE causes a request for user entry of "E" (for enable) or "D" (for disable) for each of the three Schmitt trigger inputs to the sampling clock. These inputs are used to start sampling. Control returns to command mode after the computer reads the user entry.

AF: Command AF causes a request for user entry of the output data filename. Valid entry is AAAANN where each A is any alphanumeric character and each N is an integer number. Control returns to command mode after the computer reads the user entry.

AP: Command AP causes a TTY print of the current values of all ADCOM sampling parameters and any inhibited target positions. Control returns to command mode at completion of listing.

AD: Command AD causes a request for user entry of a target position and "E" (for enable) or "D" (for disable) to enable or disable that target position. A disabled target position is not available for display to the subject by the TPS. Valid entry is a two digit integer code indicating target radius and angle as they are coded for members of the ADCOM target array ITARG. A list of the available codes is given in Table 2. Control returns to command mode after the computer reads and processes the user entry.

AO: Command AO causes a chain to program AOSMP which controls the one-shot sampling/control mode. After the specified sample set is taken AOSMP chains back to the command mode program.

AR: Command AR causes a chain to program ARSMP which controls the recycle sampling/control mode. After all enabled target positions in the ADCOM target array ITARG have been used once for a sample set trial program ARSMP chains back to the command mode program.

Table 2 - Command AD Codes

Target Radius	Target Angle	Command AD Code (decimal)
2.0°	0°	0
	45°	9
	90°	18
	135°	27
	180°	36
	225°	45
	270°	54
	315°	63
2.5°	0°	15
	45°	16
	90°	25
	135°	34
	180°	43
	225°	52
	270°	61
	315°	6
3.0°	0°	22
	45°	31
	90°	32
	135°	41
	180°	50
	225°	59
	270°	4
	315°	13
3.5°	0°	29
	45°	38
	90°	47
	135°	48
	180°	57
	225°	2
	270°	11
	315°	20
4.0°	0°	60
	45°	5
	90°	14
	135°	23
	180°	24
	225°	33
	270°	42
	315°	51

ADCOM data array format:

The sampled data of ADCOM is stored in sequentially ascending locations in a FORTRAN integer array of size 1024 words called IDATA(K) by the software. The first four words of the array are reserved for coded information about the sampling and target parameters. The maximum number of samples (samples per channel times number of channels) that can be taken is thus 1020. Each analog voltage sampled by the computer's multiplexer/converter is coded as a decimal integer in the range ± 512, corresponding to an analog input voltage range of ± 1 volt. The actual analog voltage corresponding to a digital sample may be later recovered for use in calculations by dividing the digitized form by 512. The first four IDATA(K) array words are coded as follows:

IDATA(1): Bits 0-5 contain the binary number of sequentially sampled analog channels, ICHNS. Bits 6-11 contain the binary number of the first analog channel used, ICHN1.

IDATA(2): Contains minus the number of samples per channel, ISPCT.

IDATA(3): Bits 0-5 contain the binary sample timebase multiplier ITBM, Bits 6-11 contain the binary sample timebase, IMSEC. Sample period for the data is (IMSEC)*(10**ITBM) milliseconds.

IDATA(4): Bits 0-3 contain the binary filter number used during the trial. Bits 4-8 contain the binary target deflection angle code. Bits 9-11 contain the binary target deflection radius code. IDATA(4) is internally calculated by ADCOM software.

Sample data storage in the IDATA array begins at IDATA(5) and continues sequentially, one group of samples followed by the next, in ascending order. For example, if two channels are sampled beginning with channel 0, and 500 samples per channel are taken the sample storage in the IDATA array is as such: IDATA(5) and IDATA(6) contain respectively, the channel 0 and 1 samples taken at the beginning of sampling, IDATA(7) and IDATA(8) contain the channel 0 and 1 samples taken one sampling period later, and so forth to the last sample pair contained in IDATA(1003) and IDATA(1004).

The entire IDATA array, formatted as above, is stored on disk FL2 by ADCOM using FORTRAN 12A2 format. The disk file thus created must be read back into core using this same format to recover the original data array.

ADCOM target array format:

The 40 possible target spot locations (radius and angle) of the TPS are stored in ADCOM in a FORTRAN integer array called ITARG(K). The radius and angle of the target spot location are coded according to positions of the TPS target spot generation masks relative to their reference positions (target spot radius 2.0 degrees, azimuthal angle 0 degrees). Bits 6-8 of a single ITARG entry contain the binary number of the TPS A mask position relative to its reference position. Bits 9-11 of a single ITARG entry contain the binary number of the TPS B mask position relative to its reference position. The 40 values of the ITARG array are shown in Table 3.

When a target spot location from the ITARG array is used in an experimental trial the code word from ITARG is converted into the code necessary for use in the subsequent processing program packages before it is stored in the data array IDATA(4) location. This conversion is accomplished by a subroutine, IPDCE, whose operation is discussed later. Also shown in Table 2 are the converted values of ITARG as they are stored in IDATA(4).

Table 3 - ITARG(K) Array Values

K	Target Radius	Target Angle	ITARG Value (decimal)	IDATA(4) Value (decimal)
1	3.0°	45°	31	26
2	3.5°	270°	11	147
3	4.0°	0°	60	4
4	2.0°	45°	9	24
5	2.5°	90°	25	49
6	2.0°	0°	0	0
7	3.5°	225°	2	123
8	2.0°	270°	54	144
9	4.0°	315°	51	172
10	2.5°	45°	16	25
11	2.0°	225°	45	120
12	3.5°	180°	57	99
13	3.5°	90°	47	51
14	4.0°	180°	24	100
15	2.5°	315°	6	169
16	2.5°	270°	61	145
17	2.0°	315°	63	168
18	4.0°	90°	14	52
19	2.5°	135°	34	73
20	2.5°	180°	43	97
21	2.0°	135°	27	72
22	3.0°	270°	4	146
23	3.0°	0°	22	2
24	3.0°	315°	13	170
25	3.5°	315°	20	171
26	3.5°	135°	48	75
27	2.5°	225°	52	121
28	4.0°	270°	42	148
29	2.5°	0°	15	1
30	3.0°	135°	41	74
31	3.5°	0°	29	3
32	4.0°	225°	33	124
33	3.5°	45°	38	27
34	4.0°	135°	23	76
35	3.0°	180°	50	98
36	3.0°	225°	59	122
37	2.0°	180°	36	96
38	3.0°	90°	32	50
39	4.0°	45°	5	28
40	2.0°	90°	18	48

ADCOM program operation:

ADCOM is actually a package of four main programs ADCOM, ADIS2, AOSMP, and ARSMP, and some associated subroutines TPOS, APCHK, RANDU, IPDCE, and SMPL2. ADCOM is a FORTRAN "header" program which prints a list of available sampling commands and initializes all sampling and target parameters to convenient values. ADIS2 is a FORTRAN/SABR program which interprets commands. Subroutines TPOS and APCHK are called by ADIS2. AOSMP and ARSMP are FORTRAN/SABR programs which control the target parameters and data sampling in one-shot and recycle mode respectively. Subroutines RANDU, IPDCE and SMPL2 are called by AOSMP and ARSMP.

Several important major variables are "passed" among main programs and subroutines via FORTRAN II common storage locations. These are:

ICHNS - the number of analog input channels sampled sequentially.

ICHN1 - the starting channel for sampling.

IMSEC - the sample period timebase.

ITBM - the sample period timebase multiplier.

ISAMP - the number of samples per channel.

IEN1, IEN2, IEN3 - the ASCII codes for "E" or "D" for clock Schmitt event enabling.

FNME - the output data filename stored in ASCII A6 format.

ICHCT - equals minus ICHNS.

ISPCT - equals minus ISAMP.

ICKEN - the clock enable word, instructs programmable clock.

ITMBS - the clock count word.

IW4 - IDATA(4) target radius and angle codeword.

ICST - the seed number for random number generator subroutine RANDU.

ITARG - the array of the 40 possible target spot locations coded for operation with the TPS.

IDATA - the 1024 word storage array for sampled trial data.

The operations of all ADCOM programs and subroutines are outlined below, followed by a complete ADCOM program listing.

ADCOM:

ADCOM is the name of the "header" program which also lends its name to the entire sampling/control software package. This program performs the initialization of all sampling parameters to convenient values as listed previously. ADCOM then prints a list of commands available to the user and requests the user to enter a beginning filename for use in sample set storage. This filename is restricted in format as previously stated. ADCOM then prints a list of the current sampling parameter values (the initialized values) and requests the user to enter an initial target array position. The purpose for this number is as an additional "scrambling factor" to the target spot position array ITARG. ADCOM serially shifts the ITARG array by the amount entered by the user (must be an integer number between 1 and 40). This successfully prevents the subject from guessing the sequence of target deflections which he will view. The subject is provided with a pseudo random sequence of target deflections. The procedure of presenting pseudo random target deflection sequences was found to be most efficient for use in the types of experiments performed here by the earlier work done by Dr. James Brown. After shifting ITARG, ADCOM chains to the command mode program ADIS2.

ADIS2:

ADIS2 interprets user entered keyboard commands. Error checks are included with each command to prevent the user from entering invalid sampling parameters. After the successful completion of any command sequence control always returns to the command entry mode. If an error is detected in the user entered data the command will print an error message and make its request again until a valid entry is made. The purpose of commands AC, A1, AS, AT, AM and AE is to allow the user to specify the direct sampling parameters. That is, which multiplexer channels will be sampled, what the sampling rate will be, and what the number of samples per channel will be. The data entered by these commands will be used to control the multiplexer and sampler in subsequent programs. Data entered by command AC is stored by the variables ICHNS and ICHCT. Data entered by command A1 is stored by the variable ICHN1. Data entered by command AS is stored by the variables ISAMP and ISPCT. Data entered by command AT is stored by the variable ITMBS. Data entered by commands AM and AE is stored by the variable ICKEN, the clock enable word, which is used later in clock control. Data entered by command AF is stored by the variable FNME. If the user is unsure of the current sampling parameter values he may request a listing by entering command AP on the teletype keyboard. Command AD requests the user to enter a target spot position code, as shown in Table 2, and an "E" for enable or a "D" for disable. Command AD then calls subroutine TPOS to enable or disable the specified target spot position in the ITARG array. If a TPS target spot location is disabled it will not be available for presentation to the subject under commands AO or AR. When the user is ready for data sampling he may do so in a one-shot mode or a recycle mode by entering command AO or AR respectively. These two commands operate similarly, subroutine APCHK is called to check the consistency of all sampling parameters. If APCHK returns no failures the target array ITARG is serially shifted by three places so that each time command AO or AR is envolked a new target spot deflection sequence is presented to the subject. After this is done a chain to the appropriate sample/control program, AOSMP or ARSMP, is performed.

APCHK(K):

APCHK(K) is a subroutine that is called by either command AO or AR of ADIS2. APCHK checks the validity and consistency of the sampling parameters specified by previous ADIS2 commands. The following checks are performed: (1) the number of analog channels ICHNS must be between 1 and 8, (2) the starting channel number ICHN1 must be between 0 and 7, (3) the number of channels plus the start channel must be 8 or less, (4) the sampling timebase IMSEC must be 1, 2, or 5, (5) the sampling timebase multiplier ITBM must be an integer between 0 and 4, (6) the total number of samples (samples per channel times number of channels) must be greater than zero but less than or equal to 1020, (7) the last two characters of the data sample set filename FNME must be integer numbers between 0 and 9, and (8) at least one target spot position of the ITARG array must be enabled. If any error check fails an error message is printed and APCHK returns with K = 1. If all checks pass correctly APCHK returns with K = 0.

TPOS(I, J, K):

Subroutine TPOS(I, J, K) is called by command AD of ADIS2 to search the target array ITARG for the target spot position I. Once I is found TPOS sets bit 0 of codeword I to the value of J. J is equal to 0 for enabling codeword I, J is equal to 1 for disabling codeword I. If the search of ITARG fails to find target spot position I, or if J is not equal to 0 or 1, TPOS sets K equal to 1 as a fail flag. Values of I permitted for TPOS (and command AD) are shown in Table 2. Subsequent programming checks bit 0 of the ITARG codewords to determine if the target spot position is available for use.

AOSMP:

Program AOSMP is chained to by ADIS2 when the user enters command AO. AOSMP provides TPS control and sampling for the one-shot mode, that is, one sample set is taken and output to FL2 followed by a chain back to ADIS2.

AOSMP begins operation by performing certain "housekeeping" functions on the M1705 interface and the initialization of certain internal constants. Then AOSMP types the message "INITIALIZE DRIVER" on the teletype. This prompts the user to set the TPS target parameter mechanical assemblies in their reference positions by his actuating the REF POS pushbutton on the TPS interface electronics section front panel. It should be noted here that all TPS circuitry should have been energized and TPS light sources set to their operating levels at some time prior to a command AO request by the user. The user should have also selected shutter control mode SP of the TPS visual presentation section. This ensures smooth operation of AOSMP software with the TPS. AOSMP now waits for the M1705 channel A device flag to be set by the TPS DEV RDY flag. When the TPS target assemblies have reached their reference positions the DEV RDY flag will be set and AOSMP recognizes this. AOSMP can then proceed to calculate the parameters of the target deflection. AOSMP chooses the first enabled target position codeword from the ITARG target position array. If AOSMP can find no enabled position an error message is printed and a return to command interpreter ADIS2 is performed. AOSMP makes sure that the enabled target position just chosen is different from the last (in this case the reference position). AOSMP then calls subroutine RANDU to calculate a random integer number between 0 and 8 to indicate the filter number to be used with the target position just chosen. AOSMP then calculates the number of motor "steps" required from the last target position (in this case the reference position) to the new target position and filter number for each of the TPS's three stepper motors. Note that for the target spot position masks a motor "step" in AOSMP notation is a rotation of 45 degrees by the TPS target spot maks motors (or 25 actual motor steps). Likewise a motor "step" in AOSMP notation for the TPS filter motor is a rotation of 90 degrees by the filter motor (or 50 actual motor steps). After AOSMP has calculated the number of "steps" required for each TPS motor, it calculates the rotation directions required for each motor based on the last target assembly positions relative to the new target assembly positions. Before proceeding further AOSMP updates the last target parameters to the new values currently being used. AOSMP now has all the information necessary to set up the two M1705 channel output words MØ5A and MØ5B according to the following format: (1) bit 1 of MØ5A is the TPS A spot mask motor direction, (2) bits 2-5 of MØ5A are the binary number of "steps" required for the A spot mask motor, (3) bit 7 of MØ5A is the TPS B spot mask motor direction, (4) bits 8-11 of MØ5A are the binary number of "steps" required for the B spot mask motor, (5) bit 7 of MØ5B is the TPS filter motor direction, (6) bits 8-11 of MØ5B are the binary number of "steps" required for the filter motor.

AOSMP then must set up the saccade data file codeword IDATA(4) which contains the target parameter information. This is done by first calling subroutine IPDCE to convert the target position being presently used, ITST, to a code which can be used in future processing and plotting programs (as described earlier). After IPDCE returns AOSMP includes the filter number IC with the converted target position ITST and stores the final result at IW4 for future storage by SMPL2 at IDATA(4). After all this is accomplished AOSMP is ready to control the TPS and move the TPS motors to their required positions.

The motor position words MØ5A and MØ5B are loaded into the TPS interface electronics section. The M1705A control flag A, CFA, is set to initiate motor movements. AOSMP waits for the TPS DEV RDY flag to be set at the completion of motor movements. When DEV RDY occurs AOSMP enters the sampling phase of operation by calling subroutine SMPL2. SMPL2 waits for a clock trigger pulse to occur and then samples and stores data in the IDATA array before returning to AOSMP. If neither flag KS1 nor KS2 has been set by SMPL2, AOSMP will store the complete IDATA array on disk FL2 with the filename FNME and then chain back to ADIS2. Flag KS1 will be set by SMPL2 if a user stop (teletype keyboard key "S" press) is initiated during data sampling. Flag KS2 will be set by SMPL2 when the TPS FAIL bit is set during a target deflection period T2 (a subject eyeblink or an eyetracker track loss occurred during T2). If KS1 is set by SMPL2 no data storage is performed by AOSMP and an immediate chain back to ADIS2 is performed. If KS2 is set by SMPL2 no data storage is performed by AOSMP and AOSMP is rerun starting with new target parameter selection.

ARSMP:

Program ARSMP is chained to by ADIS2 when the user enters command AR. ARSMP provides TPS control and data sampling for the recycle mode, that is, one sample set after another is taken and stored. ARSMP is basically a looping version of AOSMP so its operation is similar to AOSMP operation. ARSMP begins as does AOSMP with housekeeping operations on the M1705 and variable initializations.

Now however just before the TPS initialization request a temporary target position array ITEMP is set up whose values equal those of ITARG. This temporary array is used in the rest of the ARSMP programming. ARSMP operation now proceeds just as AOSMP operation up to and including data storage of IDATA on FL2. Now however after each successful trial the filenumber part of FNME (the last two characters of FNME) is incremented by one. Also the target position just used, ITST, is disabled by setting bit 0 of this codeword to 1. The corresponding ITEMP position is thus disabled by setting ITEMP(K) = ITST. ARSMP then loops back to the target parameter selection point. This looping continues until all positions of ITEMP are disabled and ARSMP can find no more enabled target positions. When this happens a message is printed on the teletype and ARSMP chains back to ADIS2. The procedure just described allows the experimenter to record a subject response for each of the 40 possible target spot deflections at least once during an experimental session. As with AOSMP if KS1 is set by SMPL2 (user stop) an immediate chain back to ADIS2 from ARSMP is initiated. No storage of the current sample set occurs. If KS2 is set by SMPL2 no storage of the current sample set occurs, no file number increment is done, and the target position currently used is not disabled. ARSMP then loops back to the target parameter choice point.

RANDU(K, M, A):

Subroutine RANDU(K, M, A) generates random numbers between 0.0 and 1.0 by the power residue method. This method generates overflows in the accumulator of the computer. Argument K must be an odd integer of 4 or less digits on the first call of RANDU. For subsequent calls of RANDU K should be set equal to the previous value of M computed by RANDU. Argument A is a random floating point number between 0.0 and 1.0. In the ADCOM programming here the initial value chosen for K was adjusted to yield a long random integer sequence with approximately equal numbers of integers of each value. The initial value of K used was 151.

IPDCE(I):

Subroutine IPDCE(I) is a code conversion subroutine. It converts the coded target position I into another code suitable for use in subsequent saccade data processing and plotting software. The result of the conversion of I (a one-for-one comparison of I with the possible values available for I) is stored in the common data variable IW4. Target position I must be a member of the target position array ITARG. Conversions of I are shown in Table 3.

SMPL2(I, J):

Subroutine SMPL2(I, J) is a general purpose sampling subroutine called by AOSMP or ARSMP to perform analog data sampling. SMPL2 stores the sampled data, sampling parameters, and target parameters in the common storage array FDATA which is equivalent to IDATA in the calling program AOSMP or ARSMP.

SMPL2 passes two arguments I and J. I is the flag that is set equal to 1 when a user stop (teletype keyboard key "S" press) is envolked. J is the flag that is set equal to 1 when the TPS FAIL bit is set (subject eyeblink or eyetracker track loss during data sampling). SMPL2 begins its operation by setting I = J = 0. SMPL2 then constructs and inserts the first four codewords into the FDATA data array. These codewords contain the sampling and target parameter information. Note that information necessary to construct these first four FDATA words is passed to SMPL2 via various common storage variables from the calling program. SMPL2 then instructs the programmable real time clock with the codeword ICKEN. An outer sample loop (samples per channel) is then entered, the user stop is checked, the A/D converter and multiplexer are instructed, and SMPL2 then idles in a clock trigger wait loop. Once triggered (by TPS TRIG pulse), the clock runs continuously and generates an overflow signal once each sample period as per its instructions. A clock trigger or a clock overflow signal causes SMPL2 to enter an inner sample loop (number of channels) wherein the specified number of multiplexer channels are sampled sequentially as quickly as possible, starting with the specified starting channel. The multiplexer has previously been given this information along with an instruction to increment channels each time the A/D takes a sample. When all channels have been sampled, the inner loop exits to the start of the outer loop. The user stop is checked, the A/D converter and multiplexer instructions are refreshed, and SMPL2 then waits for the clock to overflow. The clock overflow after a sample period timeout starts the inner loop again.

When the outer sample loop has cycled through the required number of samples per channel, SMPL2 then reads the M1703 12 bit input word and checks bit 0 for a failure. If a failure has occurred (TPS FAIL bit set) SMPL2 sets J = 1 before returning to the calling program. If the user stop feature is envolked at any time during data sampling SMPL2 immediately returns to the calling program with I = 1. A normal SMPL2 return to the calling program is with I = J = 0 indicating a successful experimental trial.

This completes the discussion of ADCOM programs. Complete program listings are included in the following pages.

```
*TTY:*.*<FL2:ADCOM.FT
C       PROGRAM ADCOM PRINTS COMMANDS FOR ADIS2
C       THEN CHAINS TO ADIS2. ADCOM INITIALIZES
C       SAMPLING PARAMETERS TO VALUES CONVENIENT
C       FOR EYETRACKER DATA SAMPLING.  INITIAL
C       VALUES ARE LISTED AND A FILENAME IS
C       REQUESTED BEFORE CHAINING TO ADIS2.


        COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IEN1
        COMMON IEN2,IEN3,FNME,ICHCT,ISPCT,ICKEN
        COMMON ITMBS,IW4,ICST,ITARG

        DIMENSION ITARG(40),ITTAR(40)


        ICHNS=2
        ICHN1=0
        IMSEC=5
        ITBM=0
        ISAMP=500
        IEN1=352
        IEN2=288
        IEN3=288
        ICHCT=-2
        ISPCT=-500
        ICKEN=-1167
        ITMBS=-500
        ICST=151
        IW4=0

        WRITE(1,5)
5       FORMAT('ADIS2 CMDS ARE: AC,AI,AS,AT,AM,AE'
     C  ',AF,AP,AD,AO,AR')
        READ(1,6)FNME
6       FORMAT('ENTER FILENAME: ',A6)

C       PRINT INITIAL VALUES

        WRITE(1,10)ICHNS,ICHN1
10      FORMAT(I2,' CHANNELS, STARTING WITH CHNL',I4)
        WRITE(1,11)ISAMP
11      FORMAT(I5,' SAMPLES PER CHANNEL')
        WRITE(1,12)IMSEC,ITBM
12      FORMAT('SAMPLE PERIOD: ',I2,' MSEC (TIMES 10'
     C  '**',I2,')')
        WRITE(1,13)IEN1,IEN2,IEN3
13      FORMAT('EVENT 1:',A2,' 2:',A2,' 3:',A2)
        WRITE(1,14)
14      FORMAT('ALL TARGET POSITIONS ENABLED')
        WRITE(1,15)FNME
15      FORMAT('NEXT FILE: ',A6)
```

```
C         GET TARGET POSITION ARRAY AND SHIFT IT AS
C         SPECIFIED

20        READ(1,21) I
21        FORMAT('ENTER INITIAL TARGET POSITION:',I2)
          IF(I)30,30,22
22        IF(I-40)35,25,30
25        ITARG(1)=ITTAR(40)
          J=1
          K=39
          GO TO 45
30        WRITE(1,31)
31        FORMAT('INVALID POS')
          GO TO 20
35        J=41-I
          K=I-1
          DO 40 L=1,J
          ITARG(L)=ITTAR(L+K)
40        CONTINUE
          IF(I-1)30,55,45
45        DO 50 M=1,K
          ITARG(J+M)=ITTAR(M)
50        CONTINUE

55        CALL CHAIN('ADIS2')
          STOP
          END
*
```

```
TTY:*.*<FL2:ADIS2.FT
C       PROGRAM ADIS2 INTERPERTS ADIS2 COMMANDS
C       AND ALLOWS CHANGING OF SAMPLING PARAMETERS.
C       WHEN SAMPLING BEGINS ADIS2 CHAINS TO THE
C       APPROPRIATE SAMPLING PROGRAM.

        COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IEN1
        COMMON IEN2,IEN3,FNME,ICHCT,ISPCT,ICKEN
        COMMON ITMBS,IW4,ICST,ITARG

        DIMENSION ITARG(40),ITEMP(40)

        IAC=67
        IA1=113
        IAS=83
        IAT=84
        IAM=77
        IAE=69
        IAF=70
        IAP=80
        IAD=68
        IAO=79
        IAR=82

5       READ(1,6)ICMD
6       FORMAT(':',A2)
        IF(ICMD-IAC)10,70,10
10      IF(ICMD-IA1)15,80,15
15      IF(ICMD-IAS)20,90,20
20      IF(ICMD-IAT)25,100,25
25      IF(ICMD-IAM)30,110,30
30      IF(ICMD-IAE)35,120,35
35      IF(ICMD-IAF)40,140,40
40      IF(ICMD-IAP)45,150,45
45      IF(ICMD-IAD)50,170,50
50      IF(ICMD-IAO)55,200,55
55      IF(ICMD-IAR)60,300,60
60      WRITE(1,65)
65      FORMAT('INVALID CMD')
        GO TO 5

C       CMD AC

70      READ(1,71)ICHNS
71      FORMAT('ENTER NUMBER OF CHANNELS: ',I5)
        ICHCT=-ICHNS
        IF(ICHNS)73,73,72
72      IF(8-ICHNS)73,5,5
73      WRITE(1,74)
74      FORMAT('INVALID')
        GO TO 70
```

```
C       CMD AI

80      READ(1,81)ICHN1
81      FORMAT('ENTER START CHANNEL: ',I5)
        IF(ICHN1)83,82,82
82      IF(7-ICHN1)83,5,5
83      WRITE(1,84)
84      FORMAT('INVALID')
        GO TO 80

C       CMD AS

90      READ(1,91)ISAMP
91      FORMAT('ENTER SAMPLES PER CHANNEL: ',I5)
        ISPCT=-ISAMP
        IF(ISAMP)93,93,92
92      IF(1020-ISAMP)93,5,5
93      WRITE(1,94)
94      FORMAT('INVALID')
        GO TO 90

C       CMD AT

100     READ(1,101)IMSEC
101     FORMAT('ENTER TIMEBASE:1,2,OR 5 (MSEC): ',I5)
        ITMBS=-(IMSEC*100)
        IF(IMSEC-1)104,5,102
102     IF(IMSEC-2)104,5,103
103     IF(IMSEC-5)104,5,104
104     WRITE(1,105)
105     FORMAT('INVALID')
        GO TO 100

C       CMD AM

110     READ(1,111)ITBM
111     FORMAT('ENTER TIMEBASE MULTIPLIER: ',I5)
        IF(ITBM)113,112,112
112     IF(4-ITBM)113,115,115
113     WRITE(1,114)
114     FORMAT('INVALID')
        GO TO 110
115     JTBM=5-ITBM
S       CLA CLL
S       TAD \JTBM
S       RTL;RTL;RTL
S       DCA \JTBM
S       TAD \ICKEN
S       AND (7077
S       TAD \JTBM
S       DCA \ICKEN
        GO TO 5
```

```
C       CMD AE

120     READ(1,121)IEN1,IEN2,IEN3
121     FORMAT('ENTER E OR D FOR EVENT 1:',A1,' 2:'
     C  ,A1,' 3:',A1)
        IF(IEN1-352)124,123,134
123     IED1=1
        GO TO 126
124     IF(IEN1-288)134,125,134
125     IED1=0
126     IF(IEN2-352)128,127,134
127     IED2=2
        GO TO 130
128     IF(IEN2-288)134,129,134
129     IED2=0
130     IF(IEN3-352)132,131,134
131     IED3=4
S       JMP CKF1
132     IF(IEN3-288)134,133,134
133     IED3=0
SCKF1,  CLA CLL
S       TAD \ICKEN
S       AND (7770
S       TAD \IED1
S       TAD \IED2
S       TAD \IED3
S       DCA \ICKEN
        GO TO 5
134     WRITE(1,135)
135     FORMAT('INVALID')
        GO TO 120

C       CMD AF

140     READ(1,141)FNME
141     FORMAT('ENTER FILENAME: ',A6)
        IACL=0
SCALL 0,CLEAR
SCALL 1,FAD
SARG    \FNME
S       CLA
S       TAD ACL
S       DCA \IACL
        IF(IACL+976)143,142,142
142     IF(IACL+391)145,145,143
143     WRITE(1,144)
144     FORMAT('INVALID FILENUMBER')
        GO TO 140
145     WRITE(1,146)FNME
146     FORMAT('FILES WILL START WITH: ',A6)
        GO TO 5
```

```
C       CMD AP

150     WRITE(1,151)ICHNS,ICHN1
151     FORMAT(I2,' CHANNELS, STARTING WITH CHNL',I4)
        WRITE(1,152)ISAMP
152     FORMAT(I5,' SAMPLES PER CHANNEL')
        WRITE(1,153)IMSEC,ITBM
153     FORMAT('SAMPLE PERIOD: ',I2,' MSEC(TIMES 10'
     C  '**',I2,')')
        WRITE(1,154)IEN1,IEN2,IEN3
154     FORMAT('EVENT 1:',A2,' 2:',A2,' 3:',A2)
        WRITE(1,155)
155     FORMAT('INHIBITED TARGET POSITIONS:')
        DO 160 I=1,40
        IF(ITARG(I))156,160,160
156     ITMP=ITARG(I)

S       CLA CLL
S       TAD \ITMP
S       AND (3777
S       DCA \ITMP

        WRITE(1,157)ITMP
157     FORMAT('POS:',I5)
160     CONTINUE
        WRITE(1,161)FNME
161     FORMAT('NEXT FILE: ',A6)
        GO TO 5

C       CMD AD

170     READ(1,171)IPOS,IADE
171     FORMAT('ENTER TARGET POS:',I2,' E OR D:',A1)
        IF(IADE-352)173,172,173
172     CALL TPOS(IPOS,0,IFAIL)
        IF(IFAIL)170,5,170
173     IF(IADE-288)190,174,190
174     CALL TPOS(IPOS,1,IFAIL)
        IF(IFAIL)170,5,170
190     WRITE(1,191)
191     FORMAT('INVALID')
        GO TO 170
```

```
C         CMD AO

200       CALL APCHK(IFAIL)
          IF(IFAIL)5,205,5

C         SHIFT ITARG BY 3 PLACES

205       DO 215 I=1,37
          ITEMP(I)=ITARG(I+3)
215       CONTINUE
          DO 220 J=1,3
          ITEMP(37+J)=ITARG(J)
220       CONTINUE
          DO 225 K=1,40
          ITARG(K)=ITEMP(K)
225       CONTINUE
          CALL CHAIN('AOSMP')

C         CMD AR

300       CALL APCHK(IFAIL)
          IF(IFAIL)5,305,5

C         SHIFT ITARG BY 3 PLACES

305       DO 315 I=1,37
          ITEMP(I)=ITARG(I+3)
315       CONTINUE
          DO 320 J=1,3
          ITEMP(37+J)=ITARG(J)
320       CONTINUE
          DO 325 K=1,40
          ITARG(K)=ITEMP(K)
325       CONTINUE
          CALL CHAIN('ARSMP')

          STOP
          END
*
```

```
 TTY:*.*<FL2:APCHK.FT
C       APCHK CHECKS SPEC'D SAMPLING PARAMETERS. IF
C       OK, RETURNS IFAIL=0; IF NOT, PRINTS ERROR
C       MESSAGE AND RETURNS IFAIL=1.

        SUBROUTINE APCHK(IFAIL)
        COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IEN1
        COMMON IEN2,IEN3,FNME,ICHCT,ISPCT,ICKEN
        COMMON ITMBS,IW4,ICST,ITARG

        DIMENSION ITARG(40)

S       OPDEF  BSW  7002

        IFAIL=0

C       NUMBER OF CHANNELS

        IF(ICHNS)30,30,20
20      IF(8-ICHNS)30,40,40
30      WRITE(1,31)
31      FORMAT('BAD CHANNELS')
        IFAIL=1

C       START CHANNEL

40      IF(ICHN1)60,50,50
50      IF(7-ICHN1)60,70,70
60      WRITE(1,61)
61      FORMAT('BAD START CHANNEL')
        IFAIL=1

C       CHANNELS PLUS START CHANNEL

70      IF(ICHNS+ICHN1-8)100,100,90
90      WRITE(1,91)
91      FORMAT('BAD CHANNEL SUM')
        IFAIL=1

C       TIMEBASE

100     IF(IMSEC-1)110,140,110
110     IF(IMSEC-2)120,140,120
120     IF(IMSEC-5)130,140,130
130     WRITE(1,131)
131     FORMAT('BAD TIMEBASE')
        IFAIL=1
```

```
C       TIMEBASE MULTIPLIER

140     IF(ITBM)160,150,150
150     IF(4-ITBM)160,170,170
160     WRITE(1,161)
161     FORMAT('BAD TIMEBASE MULT')
        IFAIL=1

C       SAMPLES CHECKS

170     IF(ISAMP)190,190,180
180     IF(ISAMP*ICHNS-1020)200,200,190
190     WRITE(1,191)
191     FORMAT('BAD SAMPLES')
        IFAIL=1

C       FILENUMBER

200     KACL=0
        MACL=0
SCALL 0,CLEAR
SCALL 1,FAD
SARG    \FNME
S       CLA
S       TAD ACL
S       AND (0077
S       DCA \KACL
S       TAD ACL
S       BSW
S       AND (0077
S       DCA \MACL
        IF(KACL-48)220,210,205
205     IF(KACL-57)210,210,220
210     IF(MACL-48)220,300,215
215     IF(MACL-57)300,300,220
220     WRITE(1,221)
221     FORMAT('BAD FILENUMBER')
        IFAIL=1

C       INHIBITED TARGET ARRAY CHECK

300     DO 305 I=1,40
        IF(ITARG(I))305,310,310
305     CONTINUE
        WRITE(1,306)
306     FORMAT('ALL TARGET POSITIONS INHIBITED')
        IFAIL=1

310     RETURN
        END
*
```

```
*TTY:*.*<FL2:TPOS.FT
C       SUBROUTINE TPOS(IPOS,IZ,IFAIL) SEARCHES
C       TARGET ARRAY, ITARG, FOR SPEC'D POSITION,
C       IPOS, AND SETS BIT 0 TO VALUE OF IZ (1 FOR
C       INHIBITED POSITION OR 0 FOR ENABLED
C       POSITION). IF IPOS IS NOT FOUND IN ITARG
C       OR IZ NOT= 0 OR 1 THEN TPOS RETURNS WITH
C       IFAIL=1.

        SUBROUTINE TPOS(IPOS,IZ,IFAIL)

        COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IEN1
        COMMON IEN2,IEN3,FNME,ICHCT,ISPCT,ICKEN
        COMMON ITMBS,IW4,ICST,ITARG

        DIMENSION ITARG(40)

        IFAIL=0
        ITMP=0

C       SEARCH ITARG FOR POSITION: IPOS

        DO 100 I=1,40
        ITMP=ITARG(I)

S       CLA CLL
S       TAD \ITMP
S       AND (3777           /MASK OFF INHIBIT BIT (0)
S       DCA \ITMP           /TEMP STORAGE

        IF(IPOS-ITMP)100,50,100

C       SET BIT 0 AS PER IZ (1 OR 0)

50      IF(IZ)70,51,60
51      ITMP=ITARG(I)

S       CLA CLL             /IZ=0
S       TAD \ITMP
S       AND (3777           /SET BIT 0 TO ZERO
S       DCA \ITMP           /TEMP STORAGE

        ITARG(I)=ITMP
        GO TO 200

60      IF(IZ-1)70,61,70
61      ITMP=ITARG(I)

S       CLA CLL             /IZ=1
S       TAD \ITMP
S       AND (3777           /SET BIT 0 TO ZERO
S       TAD (4000           /SET BIT 0 TO ONE
S       DCA \ITMP           /TEMP STORAGE
```

```
      ITARG(I)=ITMP
      GO TO 200

70    WRITE(1,71)
71    FORMAT('INVALID')
      IFAIL=1
      GO TO 200

100   CONTINUE
      IFAIL=1
      WRITE(1,150)
150   FORMAT('BAD TARGET POSITION')
200   RETURN
      END
*
```

```
 TTY:*.*<FL2:AOSMP.FT
C       PROGRAM AOSMP HANDLES SAMPLING IN ONE-SHOT
C       MODE.  A USER STOP (KEY "S" PRESS) WILL CAUSE
C       A CHAIN BACK TO ADIS2.  A FAIL CHECK IS DONE
C       AT END OF SAMPLING.  IF FAILURE OCCURS THEN
C       THE DATA IS REJECTED AND A NEW TARGET MOTION
C       IS INITIATED.

        COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IEN1
        COMMON IEN2,IEN3,FNME,ICHCT,ISPCT,ICKEN
        COMMON ITMBS,IW4,ICST,ITARG
        COMMON IDATA

        DIMENSION IDATA(1024),ITARG(40)

C       SPECIAL COMMANDS

S       OPDEF   MDIA  6160      /DISABLE M1705 INT A
S       OPDEF   MWRA  6163      /AC TO M1705 REG A
S       OPDEF   MTDA  6164      /TRIG M1705 STROBE A
S       OPDEF   MSFA  6166      /SET M1705 CTL FLG A
S       OPDEF   MRFA  6167      /CLR M1705 CTL FLG A
S       SKPDF   MSDA  6162      /SKP ON M1705 FLG A

S       OPDEF   MDIB  6170      /DISABLE M1705 INT B
S       OPDEF   MWRB  6173      /AC TO M1705 REG B
S       OPDEF   MTDB  6174      /TRIG M1705 STROBE B
S       OPDEF   MRFB  6177      /CLR M1705 CTL FLG B

S       OPDEF   CLAF  6152      /CLR M1703 FLGS
S       OPDEF   BSW   7002      /BYTE SWAP

S       MDIA              /TURN OFF M1705 INT'S
S       MDIB
S       CLAF              /CLR M1703 FLGS
S       MRFA              /CLR M1705 CTL FLGS
S       MRFB
S       CLA CLL
S       MWRA              /CLR DEVICE FLGS
S       MWRB
```

```
C       INITIALIZATIONS

        MØ5A=Ø
        MØ5B=Ø
        ITST=Ø
        IAL=Ø
        IBL=Ø
        ICL=Ø
        IA=Ø
        IB=Ø
        IC=Ø
        ITSTL=Ø
        WRITE(1,3)
3       FORMAT('INITIALIZE DRIVER')
SM5F1,  M5DA                /WAIT FOR DRIVER FLAG
S       JMP M5F1
S       CLA CLL
S       MWRA                /CLR DEVICE FLG

C       GET TARGET POSITION FROM TARGET ARRAY

5       DO 1Ø I=1,4Ø
        IF(ITARG(I))1Ø,6,6
6       IF(ITARG(I)-ITSTL)7,1Ø,7
7       ITST=ITARG(I)
        GO TO 15
1Ø      CONTINUE
        WRITE(1,11)
11      FORMAT('ARRAY INHIBITED')
        GO TO 3ØØ

C       CALCULATE FILTER MOTOR MOVEMENT

15      CALL RANDU(ICST,IY,YFL)
        IC=IFIX(9.Ø*YFL)
        ICST=IY

C       DECODE TARGET POSITION AND CALCULATE # OF
C       STEPS REQUIRED FROM LAST POSITION

S       CLA CLL
S       TAD \ITST          /GET TARGET POSITION
S       AND (ØØØ7          /MASK OFF A POSITION
S       DCA \IB            /PUT REMAINDER AT IB
S       TAD \ITST
S       AND (ØØ7Ø          /MASK OFF B POSITION
S       RTR;RAR
S       DCA \IA            /PUT REMAINDER AT IA

        IAC=IABS(IAL-IA)
        IBC=IABS(IBL-IB)
        ICC=IABS(ICL-IC)
```

```
C         TEST FOR ZERO MOTOR MOVEMENT OF TARGET
C         MOTORS. IF 0 SET=8.

          IF(IAC)25,20,25
20        IAC=8
25        IF(IBC)100,30,100
30        IBC=8

C         CALCULATE MOTOR DIRECTIONS: CCW=1,CW=0

100       IF(IAL-IA)110,110,115
110       IMA=1
          GO TO 120
115       IMA=0
120       IF(IBL-IB)125,125,130
125       IMB=1
          GO TO 135
130       IMB=0
135       IF(ICL-IC)140,140,145
140       IMC=0
          GO TO 150
145       IMC=1
150       IAL=IA
          IBL=IB
          ICL=IC
          ITSTL=ITST

C         SET UP M1705 WORDS

S         CLA CLL
S         TAD \IMA            /GET A DIRECTION
S         RTL;RTL
S         TAD \IAC            /ADD # OF A STEPS
S         BSW
S         DCA \M05A           /M1705A CONTRL WORD
S         TAD \IMB            /GET B DIRECTION
S         RTL;RTL
S         TAD \IBC            /ADD # OF B STEPS
S         TAD \M05A           /ADD A CTL WRD
S         DCA \M05A           /COMPLETE M1705A CTL WRD
S         TAD \IMC            /GET FILTER DIRECTION
S         RTL;RTL
S         TAD \ICC            /ADD # OF FIL STEPS
S         DCA \M05B           /COMPLETE M1705B CTL WRD
```

```
C       SET UP DATA FILE WORD #4

        CALL IPDCE(ITST)
S       CLA CLL
S       TAD \IC            /GET FILTER POSITION
S       BSW
S       RTL                /MOVE TO CORRECT POSITION
S       AND (7400          /MASK
S       TAD \IW4           /ADD RAD AND ANG DATA
S       DCA \IW4           /WORD #4 STORAGE

C       CONTROL M1705

S       CLA CLL
S       TAD \M05A          /GET M1705A CTL WRD''
S       MWRA               /LOAD IT
S       NOP                /WAIT FOR DATA TO SETTLE
S       CLA CLL
S       TAD \M05B          /GET M1705B CTL WRD
S       MWRB               /LOAD IT
S       NOP                /WAIT FOR DATA TO SETTLE
S       MTDA               /LOAD DRIVER REGISTER
S       CLA
S       TAD \ICC           /CHECK FOR ZERO FILT MOVE
S       SNA                /IF ZERO TRIG STROBE B
S       MTDB
S       MSFA               /SET M1705A CTL FLG
SM5F2,  MSDA               /WAIT FOR DRIVER TO FINISH
S       JMP M5F2
S       MRFA               /RESET M1705A CTL FLG
S       CLA CLL
S       MWRA               /CLR DEVICE FLG

C       ENTER SAMPLE ROUTINE

        CALL SMPL2(KS1,KS2)
        IF(KS1)300,200,300
200     IF(KS2)5,205,5

C       STORE DATA ON DISK

205     IWRDS=ISAMP*ICHNS+4
        CALL OOPEN('FL2',FNME)
        WRITE(4,210)(IDATA(M),M=1,IWRDS)
210     FORMAT(12A2)
        CALL OCLOSE
        WRITE(1,215)FNME
215     FORMAT('FL2:',A6,' DONE')

300     CALL CHAIN('ADIS2')

        STOP
        END
*
```

```
 TTY:*.*<FL2:ARSMP.FT
C        PROGRAM ARSMP HANDLES SAMPLING IN RECYCLE
C        MODE.  A USER STOP (KEY "S" PRESS) WILL CAUSE
C        A CHAIN BACK TO ADIS2.  A FAIL CHECK IS DONE
C        AT END OF SAMPLING.  IF FAILURE OCCURS THEN
C        THE DATA IS REJECTED AND A NEW TARGET MOTION
C        IS INITIATED.

         COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IEN1
         COMMON IEN2,IEN3,FNME,ICHCT,ISPCT,ICKEN
         COMMON ITMBS,IW4,ICST,ITARG
         COMMON IDATA

         DIMENSION IDATA(1024),ITARG(40),ITEMP(40)

C        SPECIAL COMMANDS

S        OPDEF    MDIA   6160        /DISABLE M1705 INT A
S        OPDEF    MWRA   6163        /AC TO M1705 REG A
S        OPDEF    MTDA   6164        /TRIG M1705 STROBE A
S        OPDEF    MSFA   6166        /SET M1705 CTL FLG A
S        OPDEF    MRFA   6167        /CLR M1705 CTL FLG A
S        SKPDF    MSDA   6162        /SKP ON M1705 FLG A

S        OPDEF    MDIB   6170        /DISABLE M1705 INT B
S        OPDEF    MWRB   6173        /AC TO M1705 REG B
S        OPDEF    MTDB   6174        /TRIG M1705 STROBE B
S        OPDEF    MRFB   6177        /CLR M1705 CTL FLG B

S        OPDEF    CLAF   6152        /CLR M1703 FLGS
S        OPDEF    BSW    7002        /BYTE SWAP

S        MDIA               /TURN OFF M1705 INT'S
S        MDIB
S        CLAF               /CLR M1703 FLGS
S        MRFA               /CLR M1705 CTL FLGS
S        MRFB
S        CLA CLL
S        MWRA               /CLR DEVICE FLGS
S        MWRB
```

```
C       INITIALIZATIONS

        M05A=0
        M05B=0
        ITST=0
        IA=0
        IB=0
        IAL=0
        IBL=0
        ICL=0
        K=0
        DO 2 I=1,40
        ITEMP(I)=ITARG(I)
2       CONTINUE
        WRITE(1,3)
3       FORMAT('INITIALIZE DRIVER')
SM5F1,  MSDA                /WAIT FOR DRIVER FLG
S       JMP M5F1
S       CLA CLL
S       MWRA                /CLR DEVICE FLG

C       GET TARGET POSITION FROM ITEMP

5       DO 10 J=1,40
        IF(ITEMP(J))10,6,6
6       ITST=ITEMP(J)
        K=J
        GO TO 15
10      CONTINUE
        WRITE(1,11)
11      FORMAT('TARGET ARRAY OVERFLOW')
        GO TO 400

C       CALCULATE FILTER MOTOR MOVEMENT

15      CALL RANDU(ICST,IY,YFL)
        IC=IFIX(9.0*YFL)
        ICST=IY

C       DECODE TARGET POSITION AND CALCULATE # OF
C       STEPS REQUIRED FROM LAST POSITION.

S       CLA CLL
S       TAD \ITST           /GET TARGET POSITION
S       AND (0007           /MASK OFF A POSITION
S       DCA \IB             /PUT REMAINDER AT IB
S       TAD \ITST
S       AND (0070           /MASK OFF B POSITION
S       RTR;RAR
S       DCA \IA             /PUT REMAINDER AT IA

        IAC=IABS(IAL-IA)
        IBC=IABS(IBL-IB)
        ICC=IABS(ICL-IC)
```

```
C       TEST FOR ZERO MOTOR MOVEMENT OF TARGET
C       MOTORS. IF Ø SET=8.

        IF(IAC)21,2Ø,21
2Ø      IAC=8
21      IF(IBC)25,22,25
22      IBC=8

C       CALCULATE MOTOR DIRECTIONS: CCW=1,CW=Ø

25      IF(IAL-IA)11Ø,11Ø,115
11Ø     IMA=1
        GO TO 12Ø
115     IMA=Ø
12Ø     IF(IBL-IB)125,125,13Ø
125     IMB=1
        GO TO 135
13Ø     IMB=Ø
135     IF(ICL-IC)14Ø,14Ø,145
14Ø     IMC=Ø
        GO TO 15Ø
145     IMC=1
15Ø     IAL=IA
        IBL=IB
        ICL=IC

C       SET UP M17Ø5 WORDS

S       CLA CLL
S       TAD \IMA        /GET A DIRECTION
S       RTL;RTL
S       TAD \IAC        /ADD # OF A STEPS
S       BSW
S       DCA \MØ5A       /M17Ø5A CONTROL WORD
S       TAD \IMB        /GET B DIRECTION
S       RTL;RTL
S       TAD \IBC        /ADD # OF B STEPS
S       TAD \MØ5A       /ADD A CTL WORD
S       DCA \MØ5A       /COMPLETE M17Ø5A CTL WRD
S       TAD \IMC        /GET FILTER DIRECTION
S       RTL;RTL
S       TAD \ICC        /ADD # OF FILTER STEPS
S       DCA \MØ5B       /COMPLETE M17Ø5B CTL WRD

C       SET UP DATA FILE WORD #4

        CALL IPDCE(ITST)
S       CLA CLL
S       TAD \IC         /GET FILTER POSITION
S       BSW
S       RTL             /MOVE IT TO CORRECT POSITION
S       AND (74ØØ       /MASK
S       TAD \IW4        /ADD RAD AND ANG DATA
S       DCA \IW4        /WORD #4 STORAGE
```

```
C       CONTROL M1705

S       TAD \M05A          /GET M1705A CTL WRD
S       MWRA               /LOAD IT
S       NOP                /WAIT FOR DATA TO SETTLE
S       CLA CLL
S       TAD \M05B          /GET M1705B CTL WRD
S       MWRB               /LOAD IT
S       NOP                /WAIT FOR DATA TO SETTLE
S       MTDA               /LOAD DRIVER REGISTER
S       CLA
S       TAD \ICC           /CHECK FOR ZERO FILT MOVE
S       SNA                /IF ZERO TRIG STROBE B
S       MTDB
S       MSFA               /SET M1705A CTL FLG
SM5F2,  MSDA               /WAIT FOR DRIVER TO FINISH
S       JMP M5F2
S       MRFA               /RESET M1705A CTL FLG
S       CLA CLL
S       MWRA               /CLR DEVICE FLG

C       ENTER SAMPLE ROUTINE

        CALL SMPL2(KS1,KS2)
        IF(KS1)400,200,400
200     IF(KS2)5,205,5

C       STORE DATA ON DISK

205     IWRDS=ISAMP*ICHNS+4
        CALL OOPEN('FL2',FNME)
        WRITE(4,210)(IDATA(M),M=1,IWRDS)
210     FORMAT(12A2)
        CALL OCLOSE
        WRITE(1,215)FNME
215     FORMAT('FL2:',A6,' DONE')
```

```
C       CHECK FOR FILENUMBER OVERFLOW. IF NO OVERFLOW
C       INCREMENT FILENUMBER.

SCALL 0,CLEAR           /PUT FILENUMBER AT IL
SCALL 1,FAD
SARG    \FNME
S       CLA
S       TAD ACL
S       DCA \IL
        IF(IL+391)300,250,300
250     WRITE(1,255)
255     FORMAT('FILENUMBER OVERFLOW')
        GO TO 400

300     ILA=0
S       TAD \IL         /INCREMENT FNME FILENUMBER
S       TAD (7          /(UNITS+7),INCR TENS IF REQ'D
S       DCA \IL         /SAVE NEW TENS
S       TAD \IL
S       AND (77         /UNITS+7
S       SNA
S       TAD (66         /FIXUP FOR ZERO
S       TAD (-6         /(UNITS+7)-6=UNITS+1
S       DCA \ILA        /SAVE NEW UNITS
S       TAD \IL         /PUT NEW UNITS WITH TENS
S       AND (7700
S       TAD \ILA
S       DCA \IL
SCALL 0,CLEAR           /INSERT NEW NUMBER IN FNME
SCALL 1,FAD
SARG    \FNME
S       TAD \IL
S       DCA ACL
SCALL 1,STO
SARG    \FNME
S       CLA CLL

C       INHIBIT TARGET POSITION JUST USED

S       CLA CLL
S       TAD \ITST       /GET TARGET POSITION
S       TAD (4000       /SET BIT 0 TO ONE
S       DCA \ITST       /PUT BACK
        ITEMP(K)=ITST

        GO TO 5

400     CALL CHAIN('ADIS2')

        STOP
        END
*
```

```
 TTY:*.*<FL2:RANDU.FT
C       SUBROUTINE RANDU GENERATES RANDOM NUMBERS
C       BETWEEN 0.0 AND 1.0 USING THE POWER RESIDUE
C       METHOD.
C       IX=FOR FIRST ENTRY MUST BE AN ODD INTERGER
C          WITH 4 OR LESS DIGITS. SUBSEQUENT REENTRY
C          IX=PREVIOUS VALUE OF IY COMPUTED BY RANDU.
C       IY=RESULTANT INTERGER REQUIRED FOR NEXT ENTRY
C          INTO SUBROUTINE.
C       YFL=RESULTANT RANDOM NUMBER BETWEEN 0.0 AND 1.0.

        SUBROUTINE RANDU(IX,IY,YFL)

        IY=IX*67
        IF(IY)5,6,6
5       IY=IY+2047+1
6       YFL=FLOAT(IY)
        YFL=YFL*0.4882812E-3
        RETURN
        END
*
```

```
C       SUBROUTINE IPDCE(IPOS) CONVERTS PLATE
C       POSITION DATA, IPOS, TO RADIUS AND ANGLE
C       DATA SUITABLE FOR USE IN SACCADE ANALYSIS
C       PROGRAMS.

        SUBROUTINE IPDCE(IPOS)

        COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IEN1
        COMMON IEN2,IEN3,FNME,ICHCT,ISPCT,ICKEN
        COMMON ITMBS,IW4,ICST,ITARG
        COMMON IDATA

        DIMENSION IDATA(1024),ITARG(40)

C       TEST FOR VALUE OF IPOS AND CONVERT TO REQUIRED
C       RADIUS AND ANGLE DATA. PUT VALUES AT IW4.

        DO 10 I=1,8
        J=I-1
        IF(IPOS-9*J)10,5,10
5       IW4=24*J
        GO TO 500
10      CONTINUE
        IF(IPOS-6)16,15,16
15      IW4=169
        GO TO 500
16      IF(IPOS-15)18,17,18
17      IW4=1
        GO TO 500
18      IF(IPOS-16)20,19,20
19      IW4=25
        GO TO 500
20      IF(IPOS-25)22,21,22
21      IW4=49
        GO TO 500
22      IF(IPOS-34)24,23,24
23      IW4=73
        GO TO 500
24      IF(IPOS-43)26,25,26
25      IW4=97
        GO TO 500
26      IF(IPOS-52)28,27,28
27      IW4=121
        GO TO 500
28      IF(IPOS-61)30,29,30
29      IW4=145
        GO TO 500
30      IF(IPOS-4)36,35,36
35      IW4=146
        GO TO 500
```

```
36      IF(IPOS-13)38,37,38
37      IW4=170
        GO TO 500
38      IF(IPOS-22)40,39,40
39      IW4=2
        GO TO 500
40      IF(IPOS-31)42,41,42
41      IW4=26
        GO TO 500
42      IF(IPOS-32)44,43,44
43      IW4=50
        GO TO 500
44      IF(IPOS-41)46,45,46
45      IW4=74
        GO TO 500
46      IF(IPOS-50)48,47,48
47      IW4=98
        GO TO 500
48      IF(IPOS-59)50,49,50
49      IW4=122
        GO TO 500
50      IF(IPOS-2)56,55,56
55      IW4=123
        GO TO 500
56      IF(IPOS-11)58,57,58
57      IW4=147
        GO TO 500
58      IF(IPOS-20)60,59,60
59      IW4=171
        GO TO 500
60      IF(IPOS-29)62,61,62
61      IW4=3
        GO TO 500
62      IF(IPOS-38)64,63,64
63      IW4=27
        GO TO 500
64      IF(IPOS-47)66,65,66
65      IW4=51
        GO TO 500
66      IF(IPOS-48)68,67,68
67      IW4=75
        GO TO 500
68      IF(IPOS-57)70,69,70
69      IW4=99
        GO TO 500
70      IF(IPOS-5)76,75,76
75      IW4=28
        GO TO 500
76      IF(IPOS-14)78,77,78
77      IW4=52
        GO TO 500
```

```
78      IF(IPOS-23)80,79,80
79      IW4=76
        GO TO 500
80      IF(IPOS-24)82,81,82
81      IW4=100
        GO TO 500
82      IF(IPOS-33)84,83,84
83      IW4=124
        GO TO 500
84      IF(IPOS-42)86,85,86
85      IW4=148
        GO TO 500
86      IF(IPOS-51)88,87,88
87      IW4=172
        GO TO 500
88      IF(IPOS-60)500,89,500
89      IW4=4
500     RETURN
        END
*
```

```
 TTY:*.*<FL2:SMPL2.FT
C       SUBROUTINE SMPL2 CONTROLS A/D SAMPLING AND
C       DATA STORAGE AS PER CLOCK AND A/D PARAMETERS
C       SPEC'D BY ADIS2. EITHER A USER STOP (KEY "S"
C       PRESS) OR AN EYETRACKER FAIL WILL CAUSE A
C       RETURN TO CALLING PROGRAM WITH KS1=1, KS2=1
C       RESPECTIVELY. SAMPLING STARTS WHEN AN ENABLED
C       SCHMITT TRIGGER FIRES. TARGET DATA IS STORED
C       IN WORD #4 OF DATA ARRAY.

        SUBROUTINE SMPL2(KS1,KS2)

        COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IEN1
        COMMON IEN2,IEN3,FNME,ICHCT,ISPCT,ICKEN
        COMMON ITMBS,IW4,ICST,ITARG

SFDATA, COMMN 2000                  /1024 WORD DATA ARRAY

        DIMENSION ITARG(40)

C       SPECIAL COMMANDS

S       OPDEF   CLZE   6130         /CLR CLK ENA REG
S       OPDEF   CLOE   6132         /AC TO CLK ENA REG
S       OPDEF   CLAB   6133         /AC TO CLK CNT REG
S       OPDEF   CLSA   6135         /CLK STATUS TO AC
S       SKPDF   CLSK   6131         /SKP ON CLK FLG

S       OPDEF   ADCL   6530         /CLR A/D
S       OPDEF   ADLM   6531         /AC TO A/D MUX REG
S       OPDEF   ADST   6532         /START A/D
S       OPDEF   ADRB   6533         /READ A/D
S       OPDEF   ADLE   6536         /AC TO A/D ENA REG
S       SKPDF   ADSK   6534         /SKP ON A/D DONE FLG

S       OPDEF   DCAI   3400
S       OPDEF   CDF0   6201         /CHNGE TO DATA FLD 0
S       OPDEF   CDF1   6211         /CHNGE TO DATA FLD 1
S       OPDEF   BSW    7002         /BYTE SWAP

S       OPDEF   CLAF   6152         /CLR M1703 FLGS
S       OPDEF   RDDA   6154         /READ M1703 REG
S       SKPDF   SDFS   6153         /SKP ON M1703 FLG

        KS1=0
        KS2=0
```

```
S       CLA CLL
S       TAD (7410          /SKP FOR START ROUTINE
S       DCA SKP1
S       TAD \ICKEN         /SET CLK ENA WORD
S       AND (0707
S       TAD (5060
S       DCA \ICKEN
S       TAD FDAAD          /SET DATA STRG PNTR
S       DCA FDPNT
S       TAD \ISPCT         /SET SMPLS CNTR
S       DCA FDCNT
S       TAD \ICHNS         /SET #CHNLS CODE
S       BSW                 .
S       TAD \ICHN1         /ADD START CHNL CODE
S       JMS FSTR           /PUT AT FDATA WRD#1
S       TAD \ISPCT         /PUT -(SMPLS PER CHNL)
S       JMS FSTR           /AT FDATA WRD#2
S       TAD \ITBM          /SET TMBASE MULT. CODE
S       BSW
S       TAD \IMSEC         /ADD TMBASE
S       JMS FSTR           /PUT AT FDATA WRD#3
S       TAD \IW4           /PUT TARGET DATA AT
S       JMS FSTR           /FDATA WRD#4

S       CLA CLL CMA        /CLR CLK
S       CLZE
S       CLA
S       TAD \ITMBS         /SET CLK TMBASE
S       CLAB
S       CLA
S       TAD \ICKEN         /INSTRUCT CLK
S       CLOE
S       CLSA

SSTRT1, JMS SPCHK          /CHECK FOR USER STOP
S       ADCL
S       TAD ADENA          /INSTRUCT A/D
S       ADLE
S       TAD \ICHN1         /SET FIRST A/D CHNL
S       ADLM
S       TAD \ICHCT         /SET CHNL CNTR
S       DCA CHCNT
```

```
SSKP1,  0                /START ROUTINE; SKP OR NOP
S       JMP A1
SCKF1,  CLSK             /WAIT FOR SCHMITT FIRE
S       JMP CKF1
S       ADST             /START A/D
S       TAD (7000        /CHNG SKP TO NOP
S       DCA SKP1
S       TAD (0007        /AC=7
S       CLZE             /DISABLE SCHMITT TRIGGERS
S       JMP A2
SA1,    CLSK             /WAIT FOR CLK OVERFLOW
S       JMP A1
SA2,    CLSA             /CLR CLK OVERFLOW
SADFG1, ADSK             /WAIT FOR A/D DONE FLG
S       JMP ADFG1
S       ADRB             /READ A/D
S       ADST             /RESTART A/D
S       JMS FSTR         /STORE DATA WORD
S       ISZ CHCNT        /ALL CHNLS DONE?
S       JMP ADFG1        /NO
S       ISZ FDCNT        /YES. ALL SMPLS DONE?
S       JMP STRT1        /NO. RECYCLE

SINSKP, SDFS             /WAIT FOR M1703 FLG
S       JMP INSKP
S       RDDA             /READ M1703 REG
S       CLL RAL          /FAIL CHECK
S       SZL
S       JMP FAIL
S       CLA CLL
        RETURN

SFAIL,  CLA CLL          /EYETRACKER FAIL
        KS2=1
        RETURN

S       PAGE
SFSTR,  0                /DATA STORAGE ROUTINE
S       CDF1
S       DCAI FDPNT
S       CDF0
S       INC FDPNT
S       JMP I FSTR
```

```
SSPCHK, Ø                     /USER STOP CHECK ROUTINE
S       CLA CLL
S       KSF                   /KEYBOARD FLG?
S       JMP I SPCHK           /NO
S       KRB                   /YES. READ KEYBOARD
S       TAD (-"S              /KEY="S"?
S       SZA CLA
S       JMP I SPCHK           /NO
        KS1=1
        RETURN

SFDPNT, Ø                     /DATA STORAGE PNTR
SFDCNT, Ø                     /SMPLS CNTR
SCHCNT, Ø                     /CHNL CNTR
SADENA, Ø3ØØ                  /A/D:CLK INH, AUTO INCR
SFDAAD, FDATA                 /STORAGE START ADDRESS

        END
*
```

SPROUT: Saccade Processing Program Package

Design considerations:

SPROUT programs were designed to input and process sets of ADCOM-created data files, each data file containing raw data for a single, two-dimensional eye movement response to a step-like target displacement. SPROUT analysis of each saccade data file yields saccadic response latency and settling times to four different radii about the target final position. The results of up to 63 individual saccadic responses can be processed and stored in a saccade summary array. The summary array itself may then be stored as a data file on disk FL2. This allows for efficient storage of saccade data. These summary data files are then readily available for future access by additional plotting or processing programs.

In addition to the general design features indicated earlier SPROUT programming includes various checks performed on the saccade data during processing to avoid unknowingly including irregular data in the summary data files.

The SPROUT program package described here is essentially the same as that developed by Dr. James Brown. Only a few modifications have been made so as to incorporate the change in target parameters from two (as used by Dr. Brown) to the three used here (radius, angle and filter number). Since no changes have been made in the methods of calculating saccadic latency and settling time the reader is directed to Dr. Brown's dissertation for a detailed discussion of these design considerations.

Regularity checks:

The regularity checks performed by SPROUT here are the same as those performed by Dr. Brown's version. However, a slight increase in the allowable standard deviations of the initial 0.1 second and final 1.0 second portions of the 2.5 second response record was made. This was found necessary due to the higher eyetracker gain settings used in the experiments performed here. The gain increase of the eyetracker caused a corresponding increase in the level of "subject noise" during periods of fixation. This increased noise level was compensated for by increasing the standard deviation threshold of the regularity check section from the 6 minutes of arc level used by Dr. Brown to 9 minutes of arc as used here.

SPROUT commands:

Commands to SPROUT may be entered on the teletype keyboard whenever SPROUT is in command mode. This condition is indicated by the printing of a "!" at the teletype left margin. Valid SPROUT commands are: F, G, L, P, I, D, A, E, S and V. No changes in SPROUT commands were made to those of Dr. Brown's version except for command P. The command functions are outlined below for convenience. The reader is referred to Dr. Brown's dissertation for detailed command explanations.

F: Command F causes a request for user entry of a saccade summary filename. This filename may be any six character alphanumeric string. Filenames entered by command F will be assigned to the summary array when it is output to disk FL2.

G: Command G causes a request for user entry of the saccade summary filename for a saccade summary file that currently exists on FL2. The summary array will then be read from FL2 into core memory. Note that command G updates any filename previously entered with command F to the filename of the G-obtained file.

L: Command L causes a teletype listing of all the original filenames of saccade data files summarized in the current saccade summary file.

P: Command P causes a request for user entry of the filename of an original saccade data file that is summarized in the current saccade summary file. If the original saccade file is found the processed results along with the sampling and target parameter information are listed on the teletype. If the original saccade file is not found an error message is printed.

I: Command I causes a request for user entry of the filename for a single saccade data file to be processed and inserted into the current summary array. Command I reads the filename and inputs the file. The file is processed and inserted into the current summary array. Control then returns back to command mode. Note that if the file named by command F does not exist on FL2 a return-to-OS/8- monitor error will occur with the subsequent loss of any core data.

D: Command D causes a request for user entry of the filename of an original saccade data file that is summarized in the current summary array. If the original saccade file is found, its entry will be deleted from the current summary array. If the original saccade file is not found, an error message will be printed.

A: Command A causes a request for user entry of the filenames of a group of saccade data files to be processed and inserted into the current summary array. The beginning and ending filenames of the group are requested. Thus these two filenames must have identical first four characters. The last two characters (which are numbers) of the beginning filename must represent a lower file-number than the last two characters (filenumber) of the ending filename. All files sequentially (in filenumber) between and including the beginning and ending files must exist on FL2. The specified saccade files are automatically input, processed, and inserted into the current summary array with no further user interaction. After all files have been processed control returns to command mode.

E: Command E causes a summary file output to FL2 to occur in the following sequence: (1) all spaces due to deletions are eliminated from the summary array, (2) that part of the array with data is output to FL2 with the existing summary filename, (3) the summary array is zeroed for further usage and control returns to command mode.

S: Command S causes a request for user entry of the scale factor, in volts/degree, to be used in saccade data file processing. This scale factor is normally initialized to 0.2 volts/degree, the scale used for all work done here.

V: Command V causes a request for user entry of the velocity threshold, in degrees/second, to be used in the latency determination. This threshold is normally initialized to 20 degrees/second, the value used for all work done here.

SPROUT saccade summary array format:

Saccade summary data is stored sequentially in ascending locations in a FORTRAN integer array of size 1024 words called IDATA(K). SPROUT software includes the following information in a single summary array "entry": (1) the original saccade data filename coded in A6 format and split into two characters per summary array word, (2) the sampling parameters used to collect the original saccade data, (3) the target parameters used for the saccade trial, (4) the saccade latency coded in units of sample periods, and (5) the four settling times also coded in units of sample periods. No changes to the saccade summary array format as used by Dr. Brown in his SPROUT version were made here with the exception of the target parameter codeword of a single summary array "entry". The target parameter information is stored in the sixth word of a saccade summary "entry". The format for this sixth word as used here is as follows: (1) bits 0-3 contain the binary number of the filter number used, (2) bits 4-8 contain the binary code for the target angle used, and (3) bits 9-11 contain the binary code for the target radius used. The actual target angle can be calculated by multiplying the binary target angle code by 15. The actual target radius used can be calculated as a fraction of the full scale radius by multiplying the binary radius code by 0.125 and adding 0.5 to the result. The resultant radial fraction is based upon the largest target radius being normalized to 1.0. The reader is referred to Dr. Brown's dissertation for further details of the saccade summary array format.

SPROUT program operation:

SPROUT is actually a package of three main programs SPROUT, SACUSR, and SACPRO with some associated subroutines SDCDE, FCOMP, and FNCHK. SPROUT is a FORTRAN "header" program which prints SPROUT commands and initializes parameters to convenient values. SACUSR is a FORTRAN/SABR program which interprets SPROUT commands. Subroutines SDCDE, FCOMP, and FNCHK are called by SACUSR. Program SACPRO is a FORTRAN/SABR program which inputs and processes saccade data files and appends the results to the summary array. The user begins by calling SPROUT which does its housekeeping functions and chains to SACUSR. When the user gives commands A or I, SACUSR chains to SACPRO for actual processing. When SACPRO is finished it chains back to SACUSR.

Major variables are "passed" among subroutines and chained programs by using FORTRAN common storage locations. These variables are:

ICHNS - the number of sampled channels of the saccade data file currently being processed.

ICHN1 - the number of the starting channel.

IMSEC - the sample period timebase.

ITBM - the sample period timebase multiplier.

ISAMP - the number of samples per channel.

IRAD - the decoded radius number for a target deflection. An integer from 0 to 4.

IANG - the decoded angle for a target deflection. IANG values are 0, 45, 90, 135, 180, 225, 270 or 315 degrees.

ICMD - integer equivalent of an ASCII keyboard command.

IFILT - the decoded filter number for the target deflection. An integer from 0 to 8.

SIFLO - the starting saccade data filename.

SIFHI - the ending saccade data filename.

SOFLE - the summary array filename.

TSFN - the temporary filename storage variable.

STIME - the sample period in seconds.

SLTCY - the calculated saccade latency.

ST25, ST20, ST15, ST10 - the calculated settling times to 25, 20, 15 and 10 minutes of arc.

VDEG - the scale factor in volts/degree.

SVEL - the velocity threshold in degrees/second.

The reader is directed to Dr. Brown's dissertation for further details concerning the common storage variables.

The operations of the programs SPROUT, SACUSR, and SACPRO, and the subroutines SDCDE, FCOMP, and FNCHK are essentially the same as Dr. Brown's versions. Consequently only the changes made by this author will be discussed here. The reader is directed to Dr. Brown's dissertation for further details of program operation. Complete program listings of all SPROUT programs are given following the discussion.

SPROUT:

SPROUT is a "header" program which also lends its name to the entire saccade analysis software package. This program performs the initialization of the parameters to convenient values as used here (0.2 volts/degree scale factor and 20 degrees/second threshold velocity). SPROUT prints a list of available commands then requests the user to enter a filename for the saccade summary file. SPROUT then chains to SACUSR for command interpretation and execution.

SACUSR:

The only change made to Dr. Brown's version of SACUSR is in the command P section. This section now prints out all three target parameters: radius, angle and filter number.

SDCDE(K):

Subroutine SDCDE(K) was modified from Dr. Brown's version to handle the difference in the target parameters. The target parameters are unpacked and decoded with the change previously noted to IRAD and the inclusion of IFILT.

FNCHK(I):

Subroutine FNCHK(I) as written by Dr. Brown was found to be faulty under certain conditions. That is, if the filename TSFN is of the form AAAAAN where A is any alphanumeric and N is an integer, FNCHK will return with I = 0 when it should return with I = 1. Changes were made in the examination of the last two characters of TSFN to insure that they are indeed both integers. FNCHK operates by unpacking TSFN, the ASCII A6 equivalent of a six character filename, the last two characters of which are equated to the variables ITNS and IONS respectively. Then ITNS and IONS are individually checked to be sure that they are the ASCII equivalent of an integer number from 0 to 9. FNCHK returns with I = 0 to indicate that TSFN has passed the test. If TSFN fails to meet the format requirement FNCHK returns with I = 1.

SACPRO:

Program SACPRO has only minor modifications from that version written by Dr. Brown. Some rearranging of the mathematics for calculating the standard deviations of the record's first 0.1 second and final 1.0 second reduced overall computational time slightly. The maximum allowable standard deviations of these sections of the data record was increased to allow for the increased gain setting of the eyetracker.

This completes the discussion of SPROUT program changes and operations. Complete program listings follow.

```
 TTY:*.*<FL2:SPROUT.FT
C        SPROUT ZEROES IDATA(1024), PRINTS OPENING
C        MESSAGE, REQUESTS SUMMARY FILENAME, CHAINS
C        TO SACUSR.

         COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IRAD
         COMMON IANG,ICMD,IFILT
         COMMON SIFLO,SIFHI,SOFLE,TSFN,STIME,SLTCY
         COMMON ST25,ST20,ST15,ST10,VDEG,SVEL
         COMMON IDATA

         DIMENSION IDATA(1024)
         VDEG=0.2
         SVEL=20.0
         DO 10 K=1,1024
10       IDATA(K)=0
         WRITE(1,21)
21       FORMAT('SPROUT CMDS: F,G,L,P,I,D,A,E,S,V'/)
         READ(1,31)SOFLE
31       FORMAT('SUMMARY FILENAME FL2:'A6)

         CALL CHAIN('SACUSR')
         STOP
         END
*
```

```
 TTY:*.*<FL2:SACUSR.FT
C       SACUSR PROCESSES INPUT COMMANDS FOR SACCADE
C       SUMMARY FILE MANIPULATION AND SACCADE FILE
C       PROCESSING.  WHEN CMD I OR A IS GIVEN,
C       SACUSR CHAINS TO SACPRO FOR SACCADE FILE
C       DATA PROCESSING AND INSERTION OF RESULTS
C       IN THE SUMMARY FILE.  IDATA(1024) CONTAINS
C       THE SACCADE SUMMARY FILE.


        COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IRAD
        COMMON IANG,ICMD,IFILT
        COMMON SIFLO,SIFHI,SOFLE,TSFN,STIME,SLTCY
        COMMON ST25,ST20,ST15,ST10,VDEG,SVEL
        COMMON IDATA

S       OPDEF BSW 7002

        DIMENSION IDATA(1024)

C       COMMAND VALUES.

        ICF=416
        ICG=480
        ICL=800
        ICP=1056
        ICI=608
        ICD=288
        ICA=96
        ICE=352
        ICS=1248
        ICV=1440

20      READ(1,21)ICMD
21      FORMAT('!',A1)

        IF(ICMD-ICF)51,100,51
51      IF(ICMD-ICG)52,150,52
52      IF(ICMD-ICL)53,200,53
53      IF(ICMD-ICP)54,250,54
54      IF(ICMD-ICI)55,300,55
55      IF(ICMD-ICD)56,350,56
56      IF(ICMD-ICA)57,400,57
57      IF(ICMD-ICE)58,450,58
58      IF(ICMD-ICS)59,550,59
59      IF(ICMD-ICV)70,575,70

70      WRITE(1,71)
71      FORMAT('BAD CMD'/)
        GO TO 20
```

```
C       CMD F: DECLARE NEW SUMMARY FILENAME.

100     READ(1,101)SOFLE
101     FORMAT('SUMMARY FILENAME FL2:'A6)
        GO TO 20

C       CMD G: GET AN EXISTING SUMMARY FILE FROM FL2.

150     READ(1,151)SOFLE
151     FORMAT('GET SUMMARY FILE FL2:'A6)
        CALL IOPEN('FL2',SOFLE)
        READ(4,161)IH,IM,IL,ISNUM
161     FORMAT(12A2)

C       CHECK FOR FILENAME AT START OF FILE.

S       CLA
SCALL 0,CLEAR              /CLEAR F.P.AC.
S       TAD \IH
S       DCA ACH            /F.P.AC HIGH WORD.
S       TAD \IM
S       DCA ACM            /F.P.AC MIDDLE WORD.
S       TAD \IL
S       DCA ACL            /F.P.AC LOW WORD.
SCALL 1,STO                /F.P.AC TO TSFN.
SARG    \TSFN

        CALL FCOMP(TSFN,SOFLE,IANS)
        IF(IANS)170,175,170
170     WRITE(1,171)
171     FORMAT('BAD SUMMARY FILE'/)
        GO TO 20

175     LW=(ISNUM+1)*16
        CALL IOPEN('FL2',SOFLE)
        READ(4,161)(IDATA(K),K=1,LW)
        GO TO 20

C       CMD L: LIST ALL SACCADE FILENAMES IN
C               SUMMARY ARRAY "SOFLE".

200     WRITE(1,201)SOFLE,IDATA(4)
201     FORMAT('SUMMARY FILENAME FL2:'A6,I5
     C  ' SACCADES'/)

        DO 230 K=17,1009,16
        IF(IDATA(K))205,230,205
205     CALL SDCDE(K)
        WRITE(1,211)TSFN
211     FORMAT(A6)
230     CONTINUE
        GO TO 20
```

```
C       CMD P: PRINT SUMMARIZED DATA FOR THE
C               SPECIFIED "ORIGINAL SACCADE FILE".

250     READ(1,251)SSFN
251     FORMAT('OSF:',A6)

        DO 270 K=17,1009,16
        CALL SDCDE(K)
        CALL FCOMP(TSFN,SSFN,IANS)
        IF(IANS)270,260,270
260     ISTME=IMSEC*10**ITBM
        RAD=FLOAT(IRAD)*0.125+0.500
        WRITE(1,261)TSFN,ISAMP,ISTME,RAD,IANG,IFILT,
     C  SLTCY,ST25,ST20,ST15,ST10
261     FORMAT('OSF:'A6,I6' S/CH'I6' MSEC   RAD='
     C  F5.3'  ANG='I4' DEG'/'FILTER #=',I2,//
     C  'LTCY='F7.3' SEC'/'ST25='F7.3' SEC'/'ST20=',
     C  F7.3' SEC'/'ST15='F7.3' SEC'/'ST10='F7.3,
     C  ' SEC'//)
        GO TO 20
270     CONTINUE
        GO TO 380

C       CMD I: PROCESS THE SPEC'D SACCADE FILE
C               AND INCLUDE IN SUMMARY FILE.

300     READ(1,301)SIFLO
301     FORMAT('FL2:'A6)
        SIFHI=SIFLO
        TSFN=SIFLO
        CALL FNCHK(IBAD)
        IF(IBAD)20,310,20
310     CALL CHAIN('SACPRO')

C       CMD D: DELETE THE SPEC'D SACCADE FROM
C               THE SUMMARY FILE.

350     READ(1,351)SSFN
351     FORMAT('OSF:'A6)

        DO 375 K=17,1009,16
        CALL SDCDE(K)
        CALL FCOMP(TSFN,SSFN,IANS)
        IF(IANS)375,360,375

360     K15=K+15
        DO 365 J=K,K15
365     IDATA(J)=0
```

```
        WRITE(1,371)TSFN
371     FORMAT(A6,' SACCADE SUMMARY DELETED'/)
        IDATA(4)=IDATA(4)-1
        GO TO 20
375     CONTINUE
380     WRITE(1,381)SSFN
381     FORMAT(A6,' SUMMARY NOT FOUND'/)
        GO TO 20

C       CMD A: PROCESS ALL SACCADE FILES FROM
C              "SIFLO" TO "SIFHI" INCLUSIVE AND
C              INSERT DATA INTO SUMMARY FILE.

400     READ(1,401)SIFLO
401     FORMAT('START FL2:'A6)
        TSFN=SIFLO
        CALL FNCHK(IBAD)
        IF(IBAD)20,405,20
405     READ(1,406)SIFHI
406     FORMAT('  END FL2:'A6)
        TSFN=SIFHI
        CALL FNCHK(IBAD)
        IF(IBAD)20,415,20
415     CALL CHAIN('SACPRO')

C       CMD E: COMPRESS SUMMARY DATA, INSERT
C              FILENAME "SOFLE", OUTPUT FILE TO
C              DISK AND ZERO SUMMARY ARRAY.

450     DO 480 K=17,1009,16
        IF(IDATA(K))480,455,480
455     DO 470 L=K,1009,16
        IF(IDATA(L))460,470,460
460     DO 465 M=0,15
        IDATA(K+M)=IDATA(L+M)
        IDATA(L+M)=0
465     CONTINUE
        GO TO 480
470     CONTINUE
480     CONTINUE

S       CLA
SCALL 0,CLEAR             /CLEAR F.P.AC.
SCALL 1,FAD               /SOFLE TO F.P.AC.
SARG    \SOFLE
S       TAD ACH           /SOFLE TO FIRST 3 FILEWORDS.
S       DCA \IH
S       TAD ACM
S       DCA \IM
S       TAD ACL
S       DCA \IL
        IDATA(1)=IH
        IDATA(2)=IM
        IDATA(3)=IL
```

```
        LW=(IDATA(4)+1)*16
        CALL OOPEN('FL2',SOFLE)
        WRITE(4,491)(IDATA(K),K=1,LW)
491     FORMAT(12A2)
        CALL OCLOSE
        WRITE(1,496)SOFLE,IDATA(4)
496     FORMAT('SACCADE SUMMARY FILE FL2:'A6','
     C  I5' ENTRIES,FILED.'//)
        DO 498 K=1,1024
498     IDATA(K)=0
        GO TO 20

C       CMD S: ENTER VOLTS/DEG TO BE USED FOR
C              SETTLING TIME COMPUTATIONS.

550     READ(1,551)VDEG
551     FORMAT('ENTER VOLTS/DEG: 'F6.4)
        GO TO 20

C       CMD V: ENTER DEG/SEC CUTOFF FOR LATENCY.

575     READ(1,576)SVEL
576     FORMAT('ENTER DEG/SEC LTCY CUTOFF: 'F4.1)
        GO TO 20


        STOP
        END
*
```

```
 TTY:*.*<FL2:FNCHK.FT
C       SUBROUTINE FNCHK(IBAD) CHECKS ENTERED
C       FILENAMES TO INSURE THAT THE LAST TWO
C       CHARACTERS ARE NUMBERS.  RETURNS IBAD=0
C       IF SO, IBAD=1 IF NOT.  CALLED BY SACUSR
C       AND PERFORMS THIS CHECK ON TSFN IN
C       COMMON.

        SUBROUTINE FNCHK(IBAD)

        COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IRAD
        COMMON IANG,ICMD,IFILT
        COMMON SIFLO,SIFHI,SOFLE,TSFN,STIME,SLTCY
        COMMON ST25,ST20,ST15,ST10,VDEG,SVEL
        COMMON IDATA

S       OPDEF  BSW  7002

        IBAD=0
SCALL 0,CLEAR             /CLEAR F.P.AC.
SCALL 1,FAD
SARG    \TSFN
S       CLA
S       TAD ACL
S       AND (0077         /MASK OFF ONES DIGIT
S       DCA \IONS         /PUT AT IONS
S       TAD ACL
S       BSW
S       AND (0077         /MASK OFF TENS DIGIT
S       DCA \ITNS         /PUT AT ITNS
        IF(IONS-48)20,10,5
5       IF(IONS-57)10,10,20
10      IF(ITNS-48)20,25,15
15      IF(ITNS-57)25,25,20
20      WRITE(1,21)
21      FORMAT('BAD FILENUMBER')
        IBAD=1
25      RETURN

        END
*
```

```
 TTY:*.*<FL2:FCOMP.FT
C        SUBROUTINE FCOMP(FNM1,FNM2,IANS) DOES AN
C        EFFECTIVE BIT BY BIT COMPARISON OF FNM1
C        AND FNM2.  IF THEY ARE THE SAME, FCOMP
C        RETURNS IANS=0;  IF DIFFERENT, FCOMP
C        RETURNS IANS=1.  THIS ALLOWS COMPARISONS
C        EXCEEDING THE ACCURACY OF A NORMAL FLOATING
C        POINT SUBTRACTION.

         SUBROUTINE FCOMP(FNM1,FNM2,IANS)

         F1=FNM1
         F2=FNM2
         IANS=0
         IH=0
         IM=0
         IL=0

SCALL 0,CLEAR                /CLEAR F.P.AC.
SCALL 1,FAD                  /SPLIT UP FNM1.
SARG     \F1
S        TAD ACH
S        DCA \IH
S        TAD ACM
S        DCA \IM
S        TAD ACL
S        DCA \IL

SCALL 0,CLEAR                /CLEAR F.P.AC.
SCALL 1,FAD                  /SPLIT UP FNM2,SUBTRACT
SARG     \F2                 /WORD BY WORD FROM FNM1
S        TAD ACH             /WORDS.
S        CIA
S        TAD \IH
S        SZA
S        JMP RET1
S        TAD ACM
S        CIA
S        TAD \IM
S        SZA
S        JMP RET1
S        TAD ACL
S        CIA
S        TAD \IL
S        SZA
S        JMP RET1
         RETURN
S RET1,  NOP
S        CLA CLL
         IANS=1
         RETURN
         END
*
```

```
 TTY:*.*<FL2:SDCDE.FT
C        SUBROUTINE SDCDE UNPACKS DATA FOR ONE
C        SACCADE IN THE SUMMARY ARRAY.  CALL SDCDE(K)
C        WHERE K IS AN IDATA ARRAY LOCATION ASSUMED
C        TO BE THE FIRST LOCATION OF A 16 WORD SACCADE
C        DATA BLOCK.  SDCDE PUTS THE UNPACKED DATA
C        IN COMMON STORAGE LOCATIONS.

         SUBROUTINE SDCDE(K)

         COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IRAD
         COMMON IANG,ICMD,IFILT
         COMMON SIFLO,SIFHI,SOFLE,TSFN,STIME,SLTCY
         COMMON ST25,ST20,ST15,ST10,VDEG,SVEL
         COMMON IDATA

S        OPDEF BSW 7002

         DIMENSION IDATA(1024)

         IH=IDATA(K)
         IM=IDATA(K+1)
         IL=IDATA(K+2)
         ISAMP=-IDATA(K+3)
         ITBM=IDATA(K+4)
         IMSEC=ITBM
         IANG=IDATA(K+5)
         IRAD=IANG
         IFILT=IANG

S        CLA                    /PUT FILENAME AT TSFN.
SCALL 0,CLEAR                   /CLEAR F.P.AC.
S        TAD \IH
S        DCA ACH                /F.P.AC HIGH WORD.
S        TAD \IM
S        DCA ACM                /F.P.AC MIDDLE WORD.
S        TAD \IL
S        DCA ACL                /F.P.AC LOW WORD.
SCALL 1,STO                     /F.P.AC TO TSFN.
SARG     \TSFN
```

```
S       CLA                 /GET TIMEBASE.
S       TAD \IMSEC
S       AND (77
S       DCA \IMSEC
S       TAD \ITBM           /GET TIMEBASE MULTIPLIER.
S       BSW
S       AND (77
S       DCA \ITBM
S       TAD \IRAD           /GET RADIUS NUMBER.
S       AND (0007           /MASK OFF ALL BUT IRAD
S       DCA \IRAD           /IRAD=0,1,2,3,OR 4
S       TAD \IANG           /GET ANGLE.
S       RTR;RAR
S       AND (0037           /MASK OFF ALL BUT IANG
S       DCA \IANG
S       TAD \IFILT          /GET FILTER NUMBER
S       BSW
S       RTR
S       AND (0017           /MASK OFF ALL BUT IFILT
S       DCA \IFILT          /IFILT=0,1,2,....,9
        IANG=15*IANG

        STIME=(FLOAT(IMSEC)*10.**ITBM)*.001
        SLTCY=STIME*FLOAT(IDATA(K+6))
        ST25=STIME*FLOAT(IDATA(K+7))
        ST20=STIME*FLOAT(IDATA(K+8))
        ST15=STIME*FLOAT(IDATA(K+9))
        ST10=STIME*FLOAT(IDATA(K+10))

        RETURN
        END
*
```

```
 TTY:*.*<FL2:SACPRO.FT
C        SACPRO IS CHAINED TO BY SACUSR. SACPRO
C        PROCESSES INDIVIDUAL SACCADE DATA FILES AND
C        INSERTS THE DATA IN THE SUMMARY FILE.
C        VARIOUS ERROR CHECKS PERFORMED ALONG THE
C        WAY; DATA IS PRINTED IF CHECKS FAIL
C        AND PROCESSING CONTINUES.  SACPRO CHAINS
C        BACK TO SACUSR WHEN FINISHED.


         COMMON ICHNS,ICHN1,IMSEC,ITBM,ISAMP,IRAD
         COMMON IANG,ICMD,IFILT
         COMMON SIFLO,SIFHI,SOFLE,TSFN,STIME,SLTCY
         COMMON ST25,ST20,ST15,ST10,VDEG,SVEL
         COMMON IDATA

S        OPDEF BSW 7002

         DIMENSION IDATA(1024)
         DIMENSION ITEMP(1024)


C        GET FILENUMBERS FROM FILENAMES.

         ILH=0
         IL=0
         IM=0
         IH=0
SCALL 0,CLEAR
SCALL 1,FAD
SARG     \SIFLO
S        TAD ACH
S        DCA \IH
S        TAD ACM
S        DCA \IM
S        TAD ACL
S        DCA \IL

SCALL 0,CLEAR
SCALL 1,FAD
SARG     \SIFHI
S        TAD ACL
S        DCA \ILH
```

```
C       GET SACCADE FILE AND CODEWORDS.

40      CALL IOPEN('FL2',SIFLO)
        READ(4,51)ITEMP
51      FORMAT(12A2)
        ICHNS=ITEMP(1)
        ICHN1=ICHNS
        ISAMP=-ITEMP(2)
        ITBM=ITEMP(3)
        IMSEC=ITBM

S       CLA       /DECODE SACCADE FILE CODEWORDS.
S       TAD \ICHNS        /# CHANNELS.
S       BSW
S       AND (77
S       DCA \ICHNS
S       TAD \ICHN1        /START CHANNEL.
S       AND (77
S       DCA \ICHN1
S       TAD \ITBM         /TIMEBASE MULTIPLIER.
S       BSW
S       AND (77
S       DCA \ITBM
S       TAD \IMSEC        /TIMEBASE.
S       AND (77
S       DCA \IMSEC


C       SACCADE DATA CALCULATIONS.


        NMSEC=IMSEC*10**ITBM
        STIME=FLOAT(NMSEC)*.001
        NSS=100/NMSEC
        NFS=1000/NMSEC
        SDIV=FLOAT(NSS+1)
        FDIV=FLOAT(NFS+1)

C       X,Y AVERAGES AND STANDARD DEVIATIONS
C       FOR FIRST 0.1 SEC AND FINAL 1.0 SEC.

        M1=6
        M2=2*NSS+6
        DIV=SDIV
S       JMS CALC
        XSAVG=XAVG
        YSAVG=YAVG
        XSDEV=XDEV
        YSDEV=YDEV
```

```
        M2=ISAMP*2+4
        M1=M2-2*NFS
        DIV=FDIV
S       JMS CALC
        XFAVG=XAVG
        YFAVG=YAVG
        XFDEV=XDEV
        YFDEV=YDEV
        GO TO 180

S CALC, 0

        XAVG=0.
        YAVG=0.
        XDEV=0.
        YDEV=0.

        DO 140 KY=M1,M2,2
        KX=KY-1
        XAVG=XAVG+FLOAT(ITEMP(KX))   .
        YAVG=YAVG+FLOAT(ITEMP(KY))
        XDEV=XDEV+(FLOAT(ITEMP(KX)))**2
140     YDEV=YDEV+(FLOAT(ITEMP(KY)))**2
        XAVG=XAVG/DIV
        YAVG=YAVG/DIV
        XDEV=SQRT((XDEV/DIV)-XAVG**2)
        YDEV=SQRT((YDEV/DIV)-YAVG**2)

S       JMP I CALC


C       LATENCY CALCULATION.

C       COUNTS/SAMPLE CUTOFF=DEG/SEC CUTOFF *
C         VOLTS/DEG * COUNTS/VOLT * SECS/SAMPLE:
180     CSCO=SVEL*VDEG*512.*STIME

        ILTCY=0

        DO 200 KY2=8,M2,2
        KY1=KY2-2
        KX2=KY2-1
        KX1=KX2-2
        TCNT=SQRT((FLOAT(ITEMP(KX2)-ITEMP(KX1)))**2+
     C  (FLOAT(ITEMP(KY2)-ITEMP(KY1)))**2)
        IF(CSCO-TCNT)220,220,200
200     ILTCY=(KX1-3)/2
```

```
C        SETTLING TIMES CALCULATIONS.

220      CMIN=VDEG*512./60.
         D10M=10.*CMIN
         D15M=15.*CMIN
         D20M=20.*CMIN
         D25M=25.*CMIN

C        NUMBER OF SAMPLES FOR 0.1 SEC SETTLING
C        TIME CRITERIA:
         IT=100/NMSEC

         IT10=0
         IT15=0
         IT20=0
         IT25=0

         DO 250 KY=6,M2,2
         KX=KY-1
         TD=SQRT((FLOAT(ITEMP(KX))-XFAVG)**2+
      C  (FLOAT(ITEMP(KY))-YFAVG)**2)
         IF(TD-D25M)264,264,262
262      IT25=(KY-4)/2
         GO TO 270
264      IF((KY-4)/2-IT25-IT)270,266,266
266      D25M=10000.

270      IF(TD-D20M)274,274,272
272      IT20=(KY-4)/2
         GO TO 280
274      IF((KY-4)/2-IT20-IT)280,276,276
276      D20M=10000.

280      IF(TD-D15M)284,284,282
282      IT15=(KY-4)/2
         GO TO 290
284      IF((KY-4)/2-IT15-IT)290,286,286
286      D15M=10000.

290      IF(TD-D10M)294,294,292
292      IT10=(KY-4)/2
         GO TO 250
294      IF((KY-4)/2-IT10-IT)250,296,296
296      D10M=10000.

250      CONTINUE
```

```
C        DONE WITH CALCULATIONS, CHECK FOR SMALL
C        STANDARD DEV'S, LTCY<0.1 SEC.

C        MAX ALLOWED SD=MAX SD(VOLTS)*COUNT/VOLT.
         SDMAX=.030*512.

         IF(ILTCY-NSS)310,301,301
301      IF(XSDEV-SDMAX)302,310,310
302      IF(YSDEV-SDMAX)303,310,310
303      IF(XFDEV-SDMAX)304,310,310
304      IF(YFDEV-SDMAX)320,310,310
310      WRITE(1,311)SIFLO,STIME,XSAVG,YSAVG,XSDEV,
      C  YSDEV,XFAVG,YFAVG,XFDEV,YFDEV,ILTCY,IT25,
      C  IT20,IT15,IT10
311      FORMAT('FILE FL2:'A6,F7.3' SEC'/
      C  2('('F7.2','F7.2')  <'F7.2','F7.2'>'/)5(I5/))


C        INSERT DATA IN FIRST EMPTY BLOCK OF
C        SUMMARY ARRAY.

320      DO 340 K=17,1009,16
         IF(IDATA(K))340,325,340
325      IDATA(K)=IH
         IDATA(K+1)=IM
         IDATA(K+2)=IL
         IDATA(K+3)=ITEMP(2)
         IDATA(K+4)=ITEMP(3)
         IDATA(K+5)=ITEMP(4)
         IDATA(K+6)=ILTCY
         IDATA(K+7)=IT25
         IDATA(K+8)=IT20
         IDATA(K+9)=IT15
         IDATA(K+10)=IT10
         IDATA(4)=IDATA(4)+1
         GO TO 350
340      CONTINUE
```

```
C       DONE WITH FILES OR FILENUMBER=99?

350     IF(IL-ILH)351,500,351
351     IF(IL+391)360,500,360
360     ILA=0
S       TAD \IL             /INCR SIFLO FILENUMBER.
S       TAD (7              /(UNITS+7),INCR TENS IF REQ'D.
S       DCA \IL             /SAVE NEW TENS.
S       TAD \IL
S       AND (77             /UNITS+7.
S       SNA
S       TAD (66             /FIXUP FOR ZERO.
S       TAD (-6             /(UNITS+7)-6=UNITS + 1.
S       DCA \ILA            /SAVE NEW UNITS.
S       TAD \IL             /PUT NEW UNITS WITH TENS.
S       AND (7700
S       TAD \ILA
S       DCA \IL

SCALL 0,CLEAR
SCALL 1,FAD
SARG    \SIFLO
S       TAD \IL
S       DCA ACL
SCALL 1,STO
SARG    \SIFLO

        GO TO 40


500     CALL CHAIN('SACUSR')
        STOP
        END
*
```

SSPCOM: Saccade Summary Plotting Program Package

Design considerations:

SSPCOM is a plotting program package designed to generate pointwise plots of saccade data as summarized in SPROUT-created saccade summary files. In addition to the general design considerations mentioned earlier, SSPCOM was designed to allow the user flexibility in plotting. Flexibility is achieved by allowing the user to select any one of the target parameters radius, angle, or filter number as the x-axis variable. The user may then select one or both of the other two target parameters (the ones not used as x-axis variable) as parameters of the plotted data. The y-axis is always time and the user may select any one of the five saccade data (latency or one of four settling times) as the y-axis variable. The user may select any one of five plotting symbols to represent the plotted data. This allows plotting different data on the same axes for comparisons.

The programming discussed here is based on program SSPLOT developed by Dr. Brown for a similar purpose. The reader is directed to his dissertation for SSPLOT details.

Commands and features:

Commands to SSPCOM may be entered on the teletype keyboard whenever SSPCOM is in command mode. This condition is indicated by the printing of a "!" at the teletype left margin. Valid SSPCOM commands are: FN, XS, YS, DA, PA, SY, PP and GO. These commands are detailed below:

FN: Command FN causes a request for user entry of the filename of a saccade summary data file. This file must be a SPROUT-generated summary file and must currently exist on disk FL2. SSPCOM then inputs the file specified and informs the user that it has done so by printing on the teletype the filename stored at the beginning of the file. SSPCOM also prints the number of saccade summaries contained in the summary file.

XS: Command XS causes a request for user entry of the x-axis variable and an appropriate scale factor. Command XS first requests an x-axis variable: target deflection radius, angle, or filter number. Then based on the x-axis variable selected command XS next requests an x-axis scale factor. If the x-axis is to be radius then XS requests the scale factor in "full scale inches for R = 1.00" (the largest target deflection is normalized to 1.0, which for experiments done here corresponds to 4.0°). Also command XS sets a parameterization variable so that only angle and/or filter number may be selected as data parameters. If the x-axis is to be angle command XS requests the x scale factor in degrees per inch. XS also sets the parameterization variable so that only radius and/or filter number may be selected as data parameters. If the x-axis is to be filter number then command XS requests the X scale factor in "full scale inches for F = 8," and sets the parameterization variable so that radius and/or angle may be chosen as data parameters.

YS: Command YS causes a request for user entry of the y-axis scale factor in milliseconds per inch.

DA: Command DA causes a request for user entry of the y-axis data to be plotted. The available choices are latency (LTCY) or settling time to 25, 20, 15 or 10 minutes of arc (ST25, ST20, ST15, ST10).

PA: Command PA causes a request for user entry of the data parameters. The user may choose either one or both of the two target parameters not used as the x-axis variable. For example if the x-axis is filter number then the user may select the target radius and/or angle as data parameters. When actual plotting is done only those entries in the summary file with matching parameters will be plotted. This feature is disabled by entering a negative number for the parameter.

SY: Command SY causes a request for user entry of the plotting symbol to be used for the data points. Available symbol choices are squares (S), crosses (C), diamonds (D), X's (X), or triangles (T). All symbols have maximum dimensions of 0.06 inches.

PP: Command PP causes all current plotting specifications to be printed on the teletype. The listing includes: summary filename, x-axis variable and scale, data parameters, y-axis variable and scale, and plotter symbol code.

GO: Command GO causes the plotter to generate a plot according to the previously specified parameters. When the plotter has finished plotting the individual data points, axes are drawn to extend just past the maximum data points. When completely finished the plotter pen returns to the plot origin and SSPCOM returns to command mode.

SSPCOM program operation:

SSPCOM is actually a package of two main FORTRAN/SABR programs SSPCOM and SPLOT1 with an associated subroutine PLTLN. SSPCOM is a "header" program which initiates all plot specifications to convenient values and sets up the symbol array ISYMB. SSPCOM then prints the list of available commands and chains to SPLOT1. SSPCOM is called and used only at the beginning of plot program utilization. The initial plot specifications are as follows: (1) x-axis variable is filter number with a scale of 4.0 inches for filter number = 8, (2) y-axis variable is latency with a scale of 200 milliseconds per inch, (3) no parameters are declared, (4) plotter symbol is the square. Program SPLOT1 performs all command interpretation and plotting of data.

Major variables are "passed" between SSPCOM and SPLOT1 via FORTRAN common storage locations. These variables are:

IXV - parameterization variable used to indicate the x-axis variable. If IXV = -1 the x-axis variable is target radius, if IXV = 0 the x-axis variable is target angle, if IXV = 1 the x-axis variable is filter number. IXV is initialized to 1.

XSFAC - the x-axis scale factor in units of plotter steps per x-axis variable. XSFAC is initialized so that the x-axis scale is 4.0 inches corresponding to filter number 8.

YSFAC - the y-axis scale factor in units of plotter steps per millisecond. YSFAC is initialized so that the y-axis scale is 200 milliseconds per inch.

ID1 - the ASCII equivalent of first two characters of y-axis variable specification. ID1 is initialized to the value of the letter combination LT.

ID2 - the ASCII equivalent of the last two characters of the y-axis variable specification. ID2 is initialized to the value of the letter combination CY. The combination of ID1 and ID2 indicates that initial y-axis variable is LTCY or latency.

IDEX - the saccade summary word index number used to select which data to plot on the y-axis. IDEX is intialized to 6 so that latency will be chosen for plotting.

IPAR - the radius parameter variable used to indicate which target radius is chosen as a data parameter. IPAR is initialized to a value of -1 to disable this feature.

IPAA - the angle parameter variable used to indicate which target angle is chosen as a data parameter. IPAA is initialized to a value of -1 to disable this feature.

INSYM - ASCII equivalent of the plotter symbol code. INSYM is initialized to the value of the letter S to indicate that squares are the plotting symbol.

ISY - the ISYMB array offset index used to select values of the ISYMB array corresponding to the symbol indicated by INSYM. ISY is initialized to a value of 1 so that square values are chosen from ISYMB.

ISYMB - the 40 word plotter symbol pen movement array.

The operation of programs SSPCOM, SPLOT1, and subroutine PLTLN are outlined on the following pages. Complete program listings follow the discussion.

SSPCOM:

SSPCOM is a "header" program used only once at the beginning of SSPCOM operation. SSPCOM initializes the plotter symbol array ISYMB and the plotting specifications to convenient values. After printing a list of available plotting commands, SSPCOM chains to SPLOT1 for command interpretation and plotting.

SPLOT1:

Program SPLOT1 performs all the functions of command interpretation and data plotting. SPLOT1 begins operation by doing some housekeeping and then prints a "!" at the teletype left margin to indicate that it is ready to accept commands. When the user enters a command SPLOT1 decodes the command and branches to the appropriate portion of its programming. Invalid commands cause an error message and a return to command input mode.

Command FN requests a filename to be entered in FORTRAN A6 format. The entered filename is stored as the variable SSFN. The file requested is input from disk FL2. The saccade summary file is stored in the core array IDATA(K). If SSFN does not exist on FL2 a return-to-OS/8-monitor error will occur. The user must recall SSPCOM programming.

Command XS requests the entry of an x-axis variable followed by an associated request for the x-axis scale factor. The x-axis variable will set the value of IXV which is used to indicate to subsequent programming what the x-axis variable is. The x-axis scale is computed in terms of plotter steps and stored as XSFAC.

Command YS requests the entry of a y-axis scale factor. The y-axis is always time so that an entry will be in milliseconds per inch. The y-axis scale is computed in plotter steps and stored as YSFAC.

Command DA will set the variable IDEX to indicate to the software which saccade data is to be plotted. IDEX is used as a subscript index to the IDATA array so that the proper IDATA value is chosen corresponding to the command DA entry.

Command PA allows the user to choose up to two target parameters as data parameters. The choice of parameters allowed is based upon which target parameter was previously specified as the x-axis variable and hence upon the current value of IXV. Only those summary file entries with matching target parameters to those specified with command PA are plotted when command GO is issued. To disable any parameter specification the user should enter a value of -1.

Command SY sets the value of ISY according to the user's symbol designation. ISY is used in subsequent programming as a subscript index to the ISYMB plotter symbol pen movement array so that values of ISYMB are chosen corresponding to the specified symbol.

If the user is at any time unsure of the plotting specifications he can request a listing of all current values of plotting specifications by entering command PP.

The GO command causes the SPLOT1 programming to execute the plot as specified by previous commands. The command GO programming first initializes the plotter (call PLTLN(-1 , 0, 0)) and zeros the variables IXMAX and IYMAX. These two variables are used to store the maximum plotter excursions for subsequent use in axes drawing. The programming then enters a summary array (IDATA) search/plot loop, wherein each summary array entry is examined, in sequence, as follows:
(1) the sampling parameter information, target deflection radius and angle, and filter number are extracted from the entry codewords,
(2) the X coordinate is calculated based upon the x-axis variable IXV,
(3) if any parameters are specified the entry is checked to see if its target parameters correspond to those specified by the parameterization command PA, (4) the Y coordinate is calculated based upon the

y-axis scale factor, YSFAC, and the requested data, IDATA(K + IDEX), (5) the maximum X and Y coordinates are retained, and (6) the plotter pen is moved to the calculated coordinates, IXNUM and IYNUM, and the specified plotter symbol is drawn using an eight-pass DO loop with the ISY and ISYMB symbol array coding.

The search/plot loop recycles until all summary array entries have been examined. These array entries which do not match the parameter requirements (if any exist) are skipped and the search continues to the next summary array entry. All plotter symbols are drawn as an eight-step sequence of short pen movements based upon the codes of the ISYMB array. The entries of this array contain the necessary X and Y coordinate moves and pen instructions to produce the plotter symbols.

PLTLN(KP, KX, KY):

This subroutine is a modified version of a DEC SABR program "Digital 8-12-U" copyrighted 1971. Changes were made in the DEC program by Dr. Brown to allow PLTLN to be used directly as a FORTRAN II subroutine and to monitor the M1703 input interface cabled to the plotter for detection and compensation of offscale plotter movements. The purpose of PLTLN is to drive a digital X-Y plotter in such a manner that the best fit straight line is plotted between coordinates previously and currently passed into the subroutine.

Arguments KX and KY are the integer X and Y coordinates as measured in plotter steps (100 steps per inch for the EAI model 130 used here). Each number must be in the range of -2047 to +2047.

Argument KP is used to control origin initialization and pen up or down. If KP = -1, KX and KY are ignored and PLTLN is initialized. This raises the plotter pen and zeros any old KX and KY values held inside the subroutine software. If KP = 0, the plotter pen is lowered and PLTLN examines the values of KX and KY currently passed and plots the best fit straight line from the old KX and KY coordinates, stored from the last PLTLN call, to the new KX and KY coordinates. If KP = 1, PLTLN operates as just described with the plotter pen raised.

This completes the discussion of SSPCOM programming. Program listings follow.

```
TTY:*.*<FL2:SSPCOM.FT
C       PROGRAM SSPCOM INITIALIZES SYMBOL ARRAY AND
C       PRINTS SSPCOM COMMANDS. THEN SSPCOM CHAINS
C       TO SPLOTI FOR ACTUAL PLOTING OF SACCADE
C       SUMMARY DATA.

        COMMON IXV,XSFAC,YSFAC,ID1,ID2,IDEX
        COMMON IPAR,IPAA,INSYM,ISY
        COMMON ISYMB

        DIMENSION ISYMB(40)

C       INITIALIZE ISYMB ARRAY.

        ISYMB(1)=-1594
        ISYMB(2)=390
        ISYMB(3)=391
        ISYMB(4)=455
        ISYMB(5)=454
        ISYMB(6)=-(2047+1)
        ISYMB(7)=0
        ISYMB(8)=0
        ISYMB(9)=-2042
        ISYMB(10)=7
        ISYMB(11)=-1600
        ISYMB(12)=384
        ISYMB(13)=-(2047+1)
        ISYMB(14)=0
        ISYMB(15)=0
        ISYMB(16)=0
        ISYMB(17)=-2042
        ISYMB(18)=384
        ISYMB(19)=7
        ISYMB(20)=448
        ISYMB(21)=6
        ISYMB(22)=-(2047+1)
        ISYMB(23)=0
        ISYMB(24)=0
        ISYMB(25)=-1594
        ISYMB(26)=391
        ISYMB(27)=-1658
        ISYMB(28)=455
        ISYMB(29)=-(2047+1)
        ISYMB(30)=0
        ISYMB(31)=0
        ISYMB(32)=0
        ISYMB(33)=-2044
        ISYMB(34)=517
        ISYMB(35)=581
        ISYMB(36)=4
        ISYMB(37)=-(2047+1)
        ISYMB(38)=0
        ISYMB(39)=0
        ISYMB(40)=0
```

```
C        CONVENIENT SSPCOM INITIALIZATIONS.

         IXV=1
         XSFAC=50.0
         YSFAC=0.5
         ID1=788
         ID2=217
         IDEX=6
         IPAR=-1
         IPAA=-1
         INSYM=1248
         ISY=1

C        PRINT SSPCOM COMMANDS AND CHAIN TO SPLOT1.

         WRITE(1,10)
10       FORMAT('SSPCOM CMDS ARE: FN,XS,YS,DA,PA,SY,'
     1   'PP,GO')

         CALL CHAIN('SPLOT1')
         STOP
         END
*
```

```
 TTY:*.*<FL2:SPLOT1.FT
C        PROGRAM SPLOT1 INTERPERTS SSPCOM COMMANDS
C        AND PLOTS DATA FROM SACCADE SUMMARY FILES
C        ACCORDINGLY. COMMAND ABILITIES INCLUDE
C        AXIS SCALING, DATA SELECTION, PARAMETERIZA-
C        TION, AND SYMBOL SELECTION.

S        OPDEF BSW  7002          /BYTE SWAP

         COMMON IXV,XSFAC,YSFAC,ID1,ID2,IDEX
         COMMON IPAR,IPAA,INSYM,ISY
         COMMON ISYMB

         DIMENSION ISYMB(40),IDATA(1024)

C        COMMAND VALUES.

         ICFN=398
         ICXS=1555
         ICYS=1619
         ICDA=257
         ICPA=1025
         ICSY=1241
         ICPP=1040
         ICGO=463

10       READ(1,15)ICMD
15       FORMAT('!',A2)

         IF(ICMD-ICFN)20,100,20
20       IF(ICMD-ICXS)25,200,25
25       IF(ICMD-ICYS)30,300,30
30       IF(ICMD-ICDA)35,400,35
35       IF(ICMD-ICPA)40,500,40
40       IF(ICMD-ICSY)45,600,45
45       IF(ICMD-ICPP)50,700,50
50       IF(ICMD-ICGO)55,800,55
55       WRITE(1,60)
60       FORMAT('BAD CMD')
         GO TO 10
```

```
C       CMD FN: DECLARE SUMMARY FILENAME, GET FILE
C               FROM FL2.

1ØØ     READ(1,1Ø5)SSFN
1Ø5     FORMAT('ENTER SACCADE SUMMARY FILENAME:',A6)
        CALL IOPEN('FL2',SSFN)
        READ(4,11Ø)(IDATA(K),K=1,4)
11Ø     FORMAT(12A2)
        LW=(IDATA(4)+1)*16
        WRITE(1,115)SSFN,IDATA(4)
115     FORMAT('SACCADE SUM FILE:',A6,I5,' SACCADES',/)
        CALL IOPEN('FL2',SSFN)
        READ(4,11Ø)(IDATA(K),K=1,LW)
        GO TO 1Ø

C       CMD XS: DECLARE X-AXIS SCALE IN RADIUS NUMBER,
C               DEG/INCH, OR FILTER NUMBER.

2ØØ     READ(1,2Ø1)IXD
2Ø1     FORMAT('WILL X-AXIS BE RAD, ANG OR FIL?:',A1)
        IF(IXD-1184)22Ø,215,22Ø

C       X-AXIS WILL BE RADIUS NUMBER.

215     READ(1,216)XSFAC
216     FORMAT('FULL SCALE (R=1.ØØ) INCHES:',F6.2)
        XSFAC=25.Ø*XSFAC
        IXV=-1
        GO TO 1Ø
22Ø     IF(IXD-96)23Ø,225,23Ø

C       X-AXIS WILL BE ANGLE.

225     READ(1,226)XSFAC
226     FORMAT('DEG/INCH:',F6.2)
        XSFAC=1ØØ.Ø/XSAFC
        IXV=Ø
        GO TO 1Ø
23Ø     IF(IXD-416)24Ø,235,24Ø

C       X-AXIS WILL BE FILTER NUMBER.

235     READ(1,236)XSFAC
236     FORMAT('FULL SCALE (F=8) INCHES:',F6.2)
        XSFAC=12.5*XSFAC
        IXV=1
        GO TO 1Ø
24Ø     WRITE(1,241)
241     FORMAT('NOT VALID')
        GO TO 2ØØ
```

```
C       CMD YS: DECLARE Y-AXIS SCALE IN MSEC/INCH.

300     READ(1,301)YSFAC
301     FORMAT('MSEC/INCH:',F7.2)
        YSFAC=100.0/YSFAC
        GO TO 10

C       CMD DA: DECLARE Y-AXIS DATA TO BE PLOTTED.

400     READ(1,405)ID1,ID2
405     FORMAT('LTCY,ST25,ST20,ST15,OR ST10?:',2(A2))
        IF(ID1-788)415,410,415
410     IDEX=6
        GO TO 10
415     IF(ID1-1236)460,420,460
420     IF(ID2+843)430,425,430
425     IDEX=7
        GO TO 10
430     IF(ID2+848)440,435,440
435     IDEX=8
        GO TO 10
440     IF(ID2+907)450,445,450
445     IDEX=9
        GO TO 10
450     IF(ID2+912)460,455,460
455     IDEX=10
        GO TO 10
460     WRITE(1,465)
465     FORMAT('DATA NOT AVAILABLE')
        GO TO 400

C       CMD PA: DECLARE PARAMETERS FOR DATA. RADIUS
C               NUMBER, ANGLE, OR FILTER NUMBER AS
C               PER CMD XS. ENTER NEGATIVE NUMBER TO
C               DISABLE.

500     IF(IXV)510,505,505
505     READ(1,506)PIPAR
506     FORMAT('WHICH RADIUS? 1.00,0.875,0.75,0.625,'
     1  '0.50:',F5.3)
        IPAR=IFIX((PIPAR-0.5)*8.0)
510     IF(IXV)515,520,515
515     READ(1,516)IPAA
516     FORMAT('WHICH ANGLE?:',I4)
520     IF(IXV)525,525,10
525     READ(1,526)IPAF
526     FORMAT('WHICH FILTER (0 TO 8)?:',I2)
        GO TO 10
```

```
C       CMD SY: DECLARE PLOTTER SYMBOL.

600     READ(1,605)INSYM
605     FORMAT('S,C,D,X,OR T?:',A1)
        IF(INSYM-1248)615,610,615
610     ISY=1
        GO TO 10
615     IF(INSYM-224)625,620,625
620     ISY=9
        GO TO 10
625     IF(INSYM-288)635,630,635
630     ISY=17
        GO TO 10
635     IF(INSYM-1568)645,640,645
640     ISY=25
        GO TO 10
645     IF(INSYM-1312)655,650,655
650     ISY=33
        GO TO 10
655     WRITE(1,660)
660     FORMAT('NOT VALID')
        GO TO 600

C       CMD PP: PRINT EXISTING PLOT SPECIFICATIONS.

700     WRITE(1,701)SSFN
701     FORMAT(/,'FILENAME: ',A6)
        IF(IXV)705,720,735
705     WRITE(1,706)XSFAC/25.0
706     FORMAT('XSCALE: ',F6.2,' INCHES FOR R=1.00')
        IF(IPAA)710,707,707
707     WRITE(1,708)IPAA
708     FORMAT('ANGLE PARAMETER: ',I5,' DEG')
710     IF(IPAF)750,715,715
715     WRITE(1,716)IPAF
716     FORMAT('FILTER PARAMETER: ',I3)
        GO TO 750
720     WRITE(1,721)100.0/XSFAC
721     FORMAT('XSCALE: ',F6.2,' DEG/INCH')
        IF(IPAR)725,722,722
722     WRITE(1,723)PIPAR
723     FORMAT('RADIUS PARAMETER: ',F5.3)
725     IF(IPAF)750,726,726
726     WRITE(1,727)IPAF
727     FORMAT('FILTER PARAMETER: ',I3)
        GO TO 750
```

```
735     WRITE(1,736)XSFAC/12.5
736     FORMAT('XSCALE: ',F6.2,' INCHES FOR F=8')
        IF(IPAR)740,737,737
737     WRITE(1,738)PIPAR
738     FORMAT('RADIUS PARAMETER: ',F5.3)
740     IF(IPAA)750,741,741
741     WRITE(1,742)IPAA
742     FORMAT('ANGLE PARAMETER: ',I5,' DEG')
750     WRITE(1,751)100.0/YSFAC,ID1,ID2,INSYM
751     FORMAT('YSCALE: ',F7.2,' MSEC/INCH',/,'YDATA:'
     1  ,2X,2(A2),'  SYMB: ',A1)
        GO TO 10

C       CMD GO: PLOT DATA AS SPECIFIED.

800     CALL PLTLN(-1,0,0)
        IXMAX=0
        IYMAX=0
        DO 900 K=17,LW,16

C       GET SAMPLE INFO, TARGET DATA, DECODE.

        IMSEC=IDATA(K+4)
        ITBM=IMSEC
        IANG=IDATA(K+5)
        IRAD=IANG
        IFILT=IANG

S       CLA CLL
S       TAD \IMSEC      /GET TIMEBASE
S       AND (0077
S       DCA \IMSEC
S       TAD \ITBM       /GET TIMEBASE MULT
S       BSW
S       AND (0077
S       DCA \ITBM
S       TAD \IRAD       /GET RADIUS NUMBER
S       AND (0007
S       DCA \IRAD
S       TAD \IANG       /GET ANGLE/15
S       RTR;RAR
S       AND (0037
S       DCA \IANG
S       TAD \IFILT      /GET FILTER NUMBER
S       BSW
S       RTR
S       AND (0017
S       DCA \IFILT
        IANG=15*IANG
```

```
C       PARAMETER CHECK, PLOTTER STEP CALCULATIONS.

        IF(IXV)810,820,830
810     IXNUM=IFIX(FLOAT(IRAD+1)*XSFAC)
        IF(IPAA)815,811,811
811     IF(IANG-IPAA)900,815,900
815     IF(IPAF)850,816,816
816     IF(IFILT-IPAF)900,850,900
820     IXNUM=IFIX(FLOAT(IANG)*XSFAC)
        IF(IPAR)825,821,821
821     IF(IRAD-IPAR)900,825,900
825     IF(IPAF)850,826,826
826     IF(IFILT-IPAF)900,850,900
830     IXNUM=IFIX(FLOAT(IFILT+1)*XSFAC)
        IF(IPAR)835,831,831
831     IF(IRAD-IPAR)900,835,900
835     IF(IPAA)850,836,836
836     IF(IANG-IPAA)900,850,900
850     IYNUM=IFIX(YSFAC*FLOAT(IMSEC)*10.0**ITBM
     1  *FLOAT(IDATA(K+IDEX)))

C       KEEP MAX EXCURSIONS.

        IF(IXNUM-IXMAX)865,865,861
861     IXMAX=IXNUM
865     IF(IYNUM-IYMAX)870,870,866
866     IYMAX=IYNUM

C       MOVE PEN, DRAW SYMBOL.

870     CALL PLTLN(1,IXNUM,IYNUM)
        DO 890 JS=0,7
        IPEN=ISYMB(ISY+JS)
        IXSYM=IPEN
        IYSYM=IPEN
```

```
S       CLA CLL
S       TAD \IPEN
S       RAL
S       CLA
S       RAL
S       DCA \IPEN
S       TAD \IYSYM
S       AND (0037
S       RAR
S       SZL
S       CIA
S       DCA \IYSYM
S       CLL
S       TAD \IXSYM
S       BSW
S       CLL
S       AND (0037
S       RAR
S       SZL
S       CIA
S       DCA \IXSYM

        IXSYM=IXSYM+IXNUM
        IYSYM=IYSYM+IYNUM
        CALL PLTLN(IPEN,IXSYM,IYSYM)
890     CONTINUE
900     CONTINUE

C       DRAW AXES AND RETURN TO ORIGIN.

        CALL PLTLN(1,0,0)
        DO 950 L=0,IXMAX,100
        I=L+100
        CALL PLTLN(0,I,0)
        CALL PLTLN(0,I,10)
        CALL PLTLN(0,I,0)
950     CONTINUE
        CALL PLTLN(1,0,0)
        DO 960 L=0,IYMAX,100
        I=L+100
        CALL PLTLN(0,0,I)
        CALL PLTLN(0,10,I)
        CALL PLTLN(0,0,I)
960     CONTINUE
        CALL PLTLN(1,0,0)
        GO TO 10

        STOP
        END
*
```

```
 TTY:*.*<FL2:PLTLN
C        PLTLN PLOTS THE BEST STRAIGHT LINE BETWEEN
C        THE OLD AND NEW CO-ORDINATES.  X CO-ORDINATE
C        CORRESPONDS TO "KX", Y TO "KY".  THESE ARE
C        STANDARD FORTRAN INTEGER VARIABLES AND MUST
C        BE >-2047 AND <+2047.  "KFG" CONTROLS
C        PLOTTING AS SUCH:  KFG=-1,INITIALIZE;
C        KFG=1,PLOT WITH PEN UP; KFG=0, PLOT WITH
C        PEN DOWN.

         SUBROUTINE PLTLN(KFG,KX,KY)

S        OPDEF PLCF 6502          /CLEAR PLTR FLG FF.
S        OPDEF PLPU 6503          /PEN UP.
S        OPDEF PLLR 6504          /LOAD DIR REG,CLR
S                                 /DIR REG,SET FLG.
S        OPDEF PLPD 6505          /PEN DOWN.
S        SKPDF PLSF 6501          /SKP ON PLTR FLG.

S        OPDEF CLAF 6142          /CLEAR M1703 FLAGS.
S        SKPDF SDFS 6143          /SKP ON M1703 FLAG.


         JFG=KFG
         JX=KX
         JY=KY

S        CLA CLL
S        TAD (4000
S        TAD \JX
S        DCA \JX
S        TAD (4000
S        TAD \JY
S        DCA \JY
S        CLA CLL
S        TAD \JFG
S        SPA              /MOVE PEN?
S        JMP PLOTA        /NO: INITIALIZE.
S        TAD PLOTPN       /ADD PEN STATUS.
S        CLL RTR
S        SPA CLA          /ANY CHANGE?
S        JMP PLOT1        /NO: CONTINUE.
S        SNL CLA
S        JMP A1           /GO LOWER THE PEN.
S        DCA PLOTPN       /RAISE THE PEN.
S        PLPU
S        JMP A2
S A1,    ISZ PLOTPN       /LOWER THE PEN.
S        PLPD
S A2,    JMS PLOTWT       /FLAGWAIT.
S        JMP PLOT1        /CONTINUE.
```

```
S PLOTA,CLA
S       PLPU              /RAISE PEN.
S       DCA PLOTPN        /INITIALIZATIONS.
S       TAD (4000         /"0" TO X COORD.
S       DCA PLOTNX
S       TAD (4000         /"0" TO Y COORD.
S       DCA PLOTNY
S       JMS PLOTWT
        RETURN

S PLOT1,TAD PLOTNX        /GET PREVIOUS X COORD.
S       CIA CLL
S       TAD \JX           /FORM NX-NPX.
S       SNL               /L=0: NPX<NX.
S       CIA
S       DCA PLOTDX        /ABS VAL OF DIFFERENCE.
S       RAL
S       DCA PLOTMV        /SAVE SIGN BIT.
S       TAD \JX           /UPDATE OLD X TO NEW X.
S       DCA PLOTNX
S       TAD PLOTNY        /GET PREVIOUS Y COORD.
S       CIA CLL
S       TAD \JY           /FORM NY-NPY.
S       SNL               /L=0: NPY<NY.
S       CIA
S       DCA PLOTDY        /ABS VAL OF DIFFERENCE.
S       TAD PLOTMV        /SAVE SIGN BIT.
S       RAL               /BIT 10=1: DRUM DOWN,+X.
S       DCA PLOTMV        /BIT 11=1: PEN LEFT,+Y.
S       TAD \JY           /UPDATE OLD Y TO NEW Y.
S       DCA PLOTNY
S       TAD PLOTDX
S       CIA CLL
S       TAD PLOTDY
S       SNL CLA           /L=0: DELTA X<DELTA Y.
S       JMP PLOT2
S       TAD PLOTDX        /REVERSE NUMBERS.
S       DCA PLOTNA
S       TAD PLOTDY
S       DCA PLOTDX
S       TAD PLOTNA
S       DCA PLOTDY
S       IAC               /SET MAJOR MOTION INSTR'N.
S       AND PLOTMV
S       TAD PLOTT1
S       JMP A3
```

```
S PLOT2,TAD PLOTMV
S       CLL RAR
S       TAD PLOTT2
S A3,   DCA PLOTNA
S       TAD I PLOTNA
S       DCA PLOT4A
S       TAD PLOTMV       /SET COMBINED MOTION.
S       TAD PLOTT3
S       DCA PLOTMV
S       TAD I PLOTMV
S       DCA PLTDBA
S       TAD PLOTDX
S       CLL RAR
S       DCA PLOTNA
S       TAD PLOTDX
S       CMA
S       DCA PLOTMV

S PLOT3,ISZ PLOTMV
S       JMP A4
C       ALL DONE WITH PLOT.
        RETURN

S A4,   TAD PLOTNA
S       TAD PLOTDY
S       DCA PLOTNA
S       TAD PLOTNA
S       CMA CLL
S       TAD PLOTDX
S       SZL CLA
S       JMP PLOT4        /SINGLE MOTION.

SPLOTDB,TAD PLTDBA       /COMBINED MOTION.
S       JMS OFFSCA       /OFFSCALE CHECK.
S       TAD PLOTDX
S       CIA
S       TAD PLOTNA
S       DCA PLOTNA
S A5,   JMP PLOT3

S PLOT4,TAD PLOT4A
S       JMS OFFSCA       /OFFSCALE CHECK.
S       JMP A5
```

```
SPLOTT1,A6
S A6,     0040            /PEN RIGHT(-Y).
S         0020            /PEN LEFT(+Y).
SPLOTT2,A7
S A7,     0004            /DRUM UP(-X).
S         0010            /DRUM DOWN(+X).
SPLOTT3,A8
S A8,     0044            /UP-RIGHT.
S         0024            /UP-LEFT.
S         0050            /DOWN-RIGHT.
S         0030            /DOWN-LEFT.

S         /CHECK FOR OFFSCALE AND EXECUTE MOVEMENT.

SOFFSCA,0
S         DCA CMND        /SAVE COMMAND.
S         CLAF            /CLR "OFFSCALE" FLG.
S         TAD CMND        /GET DIRECTION BIT.
S         AND (0060
S         SDFS            /FLAG?
S         SKP
S         JMP B1
S         DCA BITBUF      /NO,STORE BIT.
S         DCA COUNT
S         JMP EXEC
S B1,     SNA             /FLAG.RELEVANT MOVE?
S         JMP EXEC        /NO.
S         AND BITBUF
S         SNA CLA
S         JMP B2
S         ISZ COUNT       /YES,UPCOUNT.
S         JMP EXEC
S B2,     TAD COUNT
S         SNA CLA         /ALLOW ONSCALE MOVE?
S         JMP EXEC
S         CMA             /NO,DOWNCOUNT.
S         TAD COUNT
S         DCA COUNT
S         TAD (0060       /DELETE ONSCALE MOVEMENT.
S         CMA
S         AND CMND
S         DCA CMND
S   EXEC,TAD CMND         /GET ALLOWED CMD & DO IT.
S         PLCF
S         PLLR
S         JMS PLOTWT
S         JMP I OFFSCA
```

```
SPLOTWT,0                /PLOTTER FLAGWAIT.
S C1,    PLSF
S        JMP C1
S        PLCF
S        CLA CLL
S        JMP I PLOTWT

SPLOTPN,0
SPLOTNX,0
SPLOTNY,0
SPLOTDX,0
SPLOTDY,0
SPLOTNA,0
SPLOTMV,0
SPLTDBA,0
SPLOT4A,0

S  CMND,0
SBITBUF,0
S COUNT,0

         END
*
```

SACCOM: Saccade Summary Averaging Program Package

Design considerations:

In order to statistically evaluate the saccade data gathered here with respect to target parameters, it became necessary to average the processed results of many responses to the same target stimulus. SACCOM programming was developed to accomplish this goal. SACCOM programming was specifically designed to accept SPROUT-generated saccade summary files stored on disk FL2. SACCOM programming sorts the summary file and averages saccade latency and settling time data according to matching target parameters. SACCOM produces the means and standard deviations for the averaged latency and settling time data. Core memory size limitations necessitated compromises in SACCOM capabilities. SACCOM can only search for one filter number at a time and final results are printed on the teletype in tabular form.

All of the general design considerations mentioned earlier were followed in SACCOM development. SACCOM programming interacts with the user by teletype keyboard commands. Some commands generate requests for specific user responses. An automatic sort/average mode is used so as to minimize user effort during processing.

SACCOM commands:

Commands to SACCOM may be entered on the teletype keyboard whenever SACCOM is in command mode. This condition is signaled by the printing of a "!" at the teletype left margin. Valid SACCOM commands are: A, C, F, P and Z. Command functions are outlined below:

C: Command C causes a request for user entry of the filter number to be searched for. SACCOM programming can only search for one filter number at a time while searching all target deflection radii and angles. This is due to core memory size limitations.

F: Command F causes a request for user entry of starting and ending saccade summary filenames for the automatic sort/average programming. Both starting and ending filenames must be of the form AAAANN where A is any alphanumeric character and N is an integer. The ending filename number must be higher than the beginning filename number. All files between and including the starting and ending files must exist on disk FL2. If any file does not exist, an OS/8 monitor return error will occur when SACCOM tries to read the nonexistent file. This will cause an immediate return-to-OS/8-monitor with the subsequent loss of any core data. If the user wants to process only one file he should enter the same filename for the starting and ending files.

P: Command P will compute the final average and standard deviation values for the saccade data presently in the SACCOM data array DATA(I, J). The results are then printed on the teletype in a tabular format. Upon completion of the printing, the array DATA is zeroed (see command Z) for subsequent usage.

Z: Command Z will zero the sorted/averaged data array DATA(I, J). Command Z is automatically executed after command P has completed the print.

A: Command A instructs SACCOM programming to begin searching and sorting saccade summary files specified by command F for the filter number specified by command C and all target positions. The user may execute command A several times before printing the final results with command P. However, each time command A is given, new files should be used, but the filter number should be the same.

SACCOM data array formats

SACCOM programming contains three data arrays: DATA(40, 11), ITARG(40), and IDATA(1024). SPROUT-created saccade summary files are input into the data array IDATA(K) for sorting. The format of IDATA is therefore the same as that discussed in the SPROUT programming section. Sorted and partially processed saccade summary data are stored in the two dimensional array DATA(I, J). The 40 rows of DATA correspond to the 40 possible target positions of the target presentation system. The first five columns of DATA contain the sums of latency and settling times to 25, 20, 15 and 10 minutes arc, respectively. The sixth column of DATA contains a counter indicating the number of entries made in each row of DATA. The seventh through eleventh column of DATA contain the sums of squares of latency and settling times to 25, 20, 15 and 10 minutes arc, respectively. Fig. 56 illustrates DATA array formatting further. The 40 values of the target spot position radius and angle are contained in the array ITARG in a coded form which matches that as originally stored by ADCOM with each saccade response. The target array ITARG is used to compare with a saccade summary entry's target codeword until a match is found. The position of the match in ITARG corresponds to the same row position in DATA.

I \ J	1	2	3	4	5	6	7	8	9	10	11
1	$\sum^{N1} LTCY$	$\sum^{N1} ST25$	$\sum^{N1} ST20$	$\sum^{N1} ST15$	$\sum^{N1} ST10$	N1	$\sum^{N1} (LTCY)^2$	$\sum^{N1} (ST25)^2$	$\sum^{N1} (ST20)^2$	$\sum^{N1} (ST15)^2$	$\sum^{N1} (ST10)^2$
2	$\sum^{N2} LTCY$	$\sum^{N2} ST25$	$\sum^{N2} ST20$	$\sum^{N2} ST15$	$\sum^{N2} ST10$	N2	$\sum^{N2} (LTCY)^2$	$\sum^{N2} (ST25)^2$	$\sum^{N2} (ST20)^2$	$\sum^{N2} (ST15)^2$	$\sum^{N2} (ST10)^2$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
40	$\sum^{N40} LTCY$	$\sum^{N40} ST25$	$\sum^{N40} ST20$	$\sum^{N40} ST15$	$\sum^{N40} ST10$	N40	$\sum^{N40} (LTCY)^2$	$\sum^{N40} (ST25)^2$	$\sum^{N40} (ST20)^2$	$\sum^{N40} (ST15)^2$	$\sum^{N40} (ST10)^2$

FIG. 56 - SACCOM DATA(I, J) data array format

SACCOM program operation:

SACCOM is actually three main FORTRAN/SABR programs, SACCOM, SACCAV and SEARCH and one subroutine SFNCHK. SACCOM is a "header" program which initializes the target array ITARG, zeros the data array DATA, prints a list of commands, and then chains to SACCAV. SACCOM is called and executed only once at the beginning of averaging software usage. Program SACCAV interprets all commands. Subroutine SFNCHK is called by SACCAV. Program SEARCH performs the actual sorting of saccade summary data into the DATA array.

Major variables are "passed" among main programs by using FORTRAN common storage locations. These variables are:

DATA - the two dimensional array containing sums and sums of squares of sorted saccade summary data.

ITARG - the one dimensional array containing the codewords of the 40 possible target presentation system target positions.

SUFLI - the ASCII A6 format equivalent of the beginning saccade summary filename to be processed.

SUFLF - the ASCII A6 format equivalent of the final saccade summary filename to be processed.

IFILT - the filter position being searched for. An integer number from 0 to 8.

SACCOM program operations are outlined below followed by complete program listings.

SACCOM:

SACCOM is a FORTRAN "header" program which also lends its name to the entire package of sorting/averaging programs. SACCOM is called to start the operation of the averaging software package. SACCOM initializes the target array ITARG, zeros the DATA array, prints a list of commands available to the user, and chains to SACCAV.

SACCAV:

Program SACCAV is the command interpreter program. It begins operation with a little housekeeping and then prints a "!" at the teletype left margin to indicate that it is ready to accept user commands. When a command is entered by the user SACCAV decodes the command and branches to the appropriate portion of its programming for command execution. If an incorrect command is entered by the user, SACCAV prints an error message and returns to command input mode.

Command C requests the input of a filter number. The result is stored in the variable IFILT, IFILT is checked to be sure that it is an integer from 0 to 8. If IFILT does not pass this test an error message is printed and command C repeats its request. The programming returns to command input mode after a successful input of filter number.

Command F requests the entry of an initial summary filename. The entered result is stored in the variable SUFLI. Subroutine SFNCHK is called to be sure that SUFLI is of the proper format. That is, AAAANN where A is any alphanumeric and N is any integer number. The programming then requests the entry of a final summary filename, stored as the variable SUFLF. SFNCHK is again called to check the format of SUFLF. A further check is made on SUFLI and SUFLF to be sure that the filenumber part of SUFLF is a higher number than the filenumber part of SUFLI. Any error returned from SFNCHK or filenumber check will print an error message and produce a repeat of command F requests.

Command P programming first calculates the means and standard deviations of the latency and settling time data stored in the DATA array. This is accomplished by passing the DATA array through a DO loop which operates a row-at-a-time on DATA as follows: (1) column 6 is checked to be sure that two or more entries have been made to the row, (2) if less than two entries are made no further action is taken on that row and the loop advances to the next row, (3) if two or more entries have been made to the row the mean and standard deviations of the summary data are calculated with columns 1 to 5 of DATA containing the means and columns 7 to 11 containing the standard deviations of the latency and settling times to 25, 20, 15 and 10 minutes arc, respectively. After the means and standard deviations are calculated, the results are printed on the teletype in tabular form along with headings. After printing is completed, command Z is automatically executed and the DATA array is zeroed for subsequent use.

Command A causes an immediate chain to program SEARCH for sorting of the summary files specified by command F.

SFNCHK(IF, AF):

Subroutine SFNCHK is called to check the format of filename AF. AF must be of the format AAAANN where A is any alphanumeric and N is an integer number. SFNCHK checks the last two characters of AF to be sure that they are integers. If the last two characters of AF are integers SFNCHK returns with IF = 1 otherwise SFNCHK will return with IF = 0 indicating an error.

SEARCH:

Program SEARCH sequentially sorts all saccade summary files starting with filename SUFLI and ending with filename SUFLF into the DATA array. SEARCH begins by extracting the beginning and ending file numbers from SUFLI and SUFLF and storing them in the variables ILL and ILH, respectively. SEARCH then inputs the SUFLI saccade summary file from FL2 into the IDATA array. SEARCH then enters a sort/calculation loop wherein each summary entry of IDATA is examined, in sequence, as follows: (1) the sampling parameter information and target parameter information are extracted and decoded, (2) the filter number setting of the particular entry is compared to IFILT, if no match occurs, the sort/calculation loop proceeds to the next IDATA entry, (3) if the filter number of the entry matches IFILT the target position of the entry is compared to the ITARG array until a match is found, the index of the match being stored by the variable LL, (4) the saccade data of the entry is included with the data already existing in row LL of the DATA array as follows: (a) columns 1 to 5 of row LL are the running sums of latency and settling times to 25, 20, 15 and 10 minutes arc respectively, (b) columns 7 to 11 are the running sums of squares of latency and settling times to 25, 20, 15 and 10 minutes arc respectively, (5) the column 6 running entry counter of row LL is incremented by one, (6) the sort/calculation loop proceeds to the next entry of IDATA. When the IDATA array has been completely sorted, SEARCH checks to see if ILL = ILH. If this is so the program is complete and a chain back to SACCAV is done. If ILL ≠ ILH the ILL file number is incremented by one and SUFLI is updated to this new filename. SEARCH then loops back to the summary array data input from FL2 point.

This completes the discussion of SACCOM and its component programming, program listings follow.

```
 TTY:*.*<FL2:SACCOM.FT
C        PROGRAM SACCOM IS AN INTRODUCTORY PROGRAM
C        THAT INITIALIZES TARGET POSITIONS AND ZEROES
C        THE DATA ARRAY.  SACCOM THEN CHAINS TO SACCAV
C        TO CONTINUE THE SACCADE DATA ANALYSIS.

         COMMON DATA,ITARG,SUFLI,SUFLF,IFILT

         DIMENSION DATA(40,11),ITARG(40)

         DO 5 I=1,8
         DO 5 J=1,5
         ITARG(J*8-I+1)=J+191-(I*24)
5        CONTINUE

         DO 10 K=1,40
         DO 10 J=1,11
         DATA(K,J)=0.0
10       CONTINUE

         WRITE(1,15)
15       FORMAT('SACCAV COMMANDS ARE: A,C,F,P,Z',/)

         CALL CHAIN('SACCAV')

         STOP
         END
*
```

```
 TTY:*.*<FL2:SACCAV.FT
C       PROGRAM SACCAV INTERPRETS COMMANDS AND
C       COMPUTES MEANS AND STANDARD DEVIATIONS OF
C       SACCADE SUMMARIES CREATED BY SPROUT. SACCAV
C       CHAINS TO SEARCH FOR SORTING OF SACCADE
C       SUMMARY ARRAYS. FINAL VALUES ARE PRINTED
C       ON TTY.

        COMMON DATA,ITARG,SUFLI,SUFLF,IFILT

        DIMENSION DATA(40,11),ITARG(40)

C       COMMAND VALUES

        ICA=96
        ICC=224
        ICF=416
        ICP=1056
        ICZ=1696

10      READ(1,15) ICMD
15      FORMAT('!',A1)
        IF(ICMD-ICA)20,500,20
20      IF(ICMD-ICC)25,100,25
25      IF(ICMD-ICF)30,200,30
30      IF(ICMD-ICP)35,300,35
35      IF(ICMD-ICZ)40,400,40
40      WRITE(1,45)
45      FORMAT('BAD COMMAND',/)
        GO TO 10

C       CMD C: DECLARE FILTER POSITION FOR SEARCH.
C              CHECK FOR VALIDITY.

100     READ(1,105) IFILT
105     FORMAT('SELECT FILTER VALUE (0 TO 8):',I1)
        DO 110 J=1,9
        K=J-1
        IF(IFILT-K)110,10,110
110     CONTINUE
        WRITE(1,115)
115     FORMAT('INVALID',/)
        GO TO 100
```

```
C         CMD F: DECLARE INITIAL AND FINAL SACCADE
C                SUMMARY FILENAMES FOR SEARCH. CHECK
C                FOR VALIDITY.

200       READ(1,205) SUFLI
205       FORMAT('INITIAL SUMMARY FILENAME: ',A6)
          CALL SFNCHK(ICHK,SUFLI)
          IF(ICHK)250,250,220
220       READ(1,225) SUFLF
225       FORMAT('FINAL SUMMARY FILENAME: ',A6)
          CALL SFNCHK(ICHK,SUFLF)
          IF(ICHK)250,250,230
230       ILI=0
          ILF=0
S         CLA CLL
SCALL 0,CLEAR                 /CLEAR F.P.A.C.
SCALL 1,FAD                   /GET LOW FILENUMBER
SARG      \SUFLI
S         TAD ACL
S         DCA \ILI
SCALL 0,CLEAR                 /CLEAR F.P.A.C.
SCALL 1,FAD                   /GET HIGH FILENUMBER
SARG      \SUFLF
S         TAD ACL
S         DCA \ILF
          IF(ILI-ILF)10,10,250
250       WRITE(1,255)
255       FORMAT('BAD FILENAMES',/)
          GO TO 200

C         CMD P: COMPUTE MEANS AND STANDARD DEVIATIONS
C                AND PRINT RESULTS ON TTY.

300       DO 350 K=1,40
          IF(DATA(K,6)-1.0)350,350,305
305       DO 350 J=1,5
          DATA(K,J)=DATA(K,J)/DATA(K,6)
          DUMY=DATA(K,J+6)-DATA(K,6)*(DATA(K,J)**2)
          STUP=DUMY/(DATA(K,6)-1.0)
          DATA(K,J+6)=STUP**0.5
350       CONTINUE
```

```
      WRITE(1,355)IFILT
355   FORMAT('FILTER NUMBER:',I2)
      WRITE(1,360)
360   FORMAT('TARG',5X,'LTCY',5X,'ST25',5X,'ST20',5X,
     C 'ST15',5X,'ST10',3X,'NO.')
      WRITE(1,365)
365   FORMAT('  POS',4X,'(SEC)',4X,'(SEC)',4X,'(SEC)',
     C 4X,'(SEC)',4X,'(SEC)',2X,'SAMP')
      DO 380 K=1,40
      WRITE(1,370)K,(DATA(K,J),J=1,5),IFIX(DATA(K,6))
370   FORMAT(2X,I2,5F9.3,I6)
      WRITE(1,375)(DATA(K,J),J=7,11)
375   FORMAT('STD DEV',F6.3,4F9.3)
380   CONTINUE

C     CMD Z: ZERO MEAN-STANDARD DEVIATION DATA
C            ARRAY. AUTOMATICALLY DONE AFTER
C            CMD P.

400   DO 410 K=1,40
      DO 410 J=1,11
      DATA(K,J)=0.0
410   CONTINUE
      GO TO 10

C     CMD A: APPEND FILES SPECIFIED BY CMD F TO
C            MEAN-STD DEV DATA ARRAY.

500   CALL CHAIN('SEARCH')

      STOP
      END
*
```

```
 TTY:*.*<FL2:SFNCHK.FT
C        SUBROUTINE SFNCHK(ICHK,SUFF) CHECKS THE INPUT
C        FILENAMES TO BE SURE THE LAST TWO
C        CHARACTERS ARE NUMBERS. RETURNS ICHK=1 IF
C        SO,ICHK=0 IF NOT.  CALLED BY SACCAV, PERFORMS
C        CHECK ON SUFF.

         SUBROUTINE SFNCHK(ICHK,SUFF)

S        OPDEF BSW 7002

         TEMP=SUFF
         ICHK=1
SCALL 0,CLEAR
SCALL 1,FAD
SARG     \TEMP
S        CLA
S        TAD ACL
S        AND (0077
S        DCA \IONE
S        TAD ACL
S        BSW
S        AND (0077
S        DCA \ITEN

         IF(IONE-48)25,15,10
10       IF(IONE-57)15,15,25

15       IF(ITEN-48)25,30,20
20       IF(ITEN-57)30,30,25

25       ICHK=0
30       RETURN

         END
*
```

```
 TTY:*.*<FL2:SEARCH.FT
C       PROGRAM SEARCH SORTS SACCADE SUMMARY FILES
C       BY TARGET POSITION AND FILTER VALUE. SUMS
C       AND SUMS OF SQUARES OF SACCADE DATA ARE
C       COMPUTED SO THAT MEANS AND STANDARD DEVIATIONS
C       CAN BE COMPUTED BY SACCAV.

S       OPDEF BSW 7002  /BYTE SWAP

        COMMON DATA,ITARG,SUFLI,SUFLF,IFILT

        DIMENSION IDATA(1024),DATA(40,11),ITARG(40)

C       GET FILENUMBERS FROM FILENAMES

        ILH=0
        ILL=0
S       CLA CLL
SCALL 0,CLEAR           /CLEAR F.P.A.C.
SCALL 1,FAD             /GET LOW FILENUMBER
SARG    \SUFLI
S       TAD ACL
S       DCA \ILL
SCALL 0,CLEAR           /CLEAR F.P.A.C.
SCALL 1,FAD             /GET HIGH FILENUMBER
SARG    \SUFLF
S       TAD ACL
S       DCA \ILH

C       READ FROM FL2: SACCADE SUMMARY SUFLI

10      CALL IOPEN('FL2',SUFLI)
        READ(4,15)IH,IM,IL,ISNUM
15      FORMAT(12A2)
        LW=(ISNUM+1)*16
        CALL IOPEN('FL2',SUFLI)
        READ(4,20)(IDATA(K),K=1,LW)
20      FORMAT(12A2)

C       SORT SACCADE SUMMARY ARRAY ACCORDING TO
C       TARGET POSITION AND FILTER SETTING.

        DO 100 K=1,ISNUM
        ISAMP=-IDATA(K*16+4)
        ITBM=IDATA(K*16+5)
        IMSEC=ITBM
        IARF=IDATA(K*16+6)
        IFILM=IARF
```

```
C        DECODE SACCADE PARAMETERS

S        CLA CLL
S        TAD \IMSEC          /GET TIMEBASE
S        AND (0077           /MASK
S        DCA \IMSEC
S        TAD \ITBM           /GET TIMEBASE MULTIPLIER
S        BSW
S        AND (0077           /MASK
S        DCA \ITBM
S        TAD \IARF           /GET TARGET POSITION
S        AND (0377           /MASK
S        DCA \IARF
S        TAD \IFILM          /GET FILTER POSITION
S        AND (7400           /MASK
S        BSW
S        RTR
S        DCA \IFILM

C        DOES FILTER POSITION CORRESPOND TO THAT
C        SET BY SACCAV?

         IF(IFILM-IFILT)100,25,100
25       LL=0

C        YES! FIND TARGET POSITION SLOT IN
C        ITARG ARRAY.

         DO 35 J=1,40
         IF(IARF-ITARG(J))35,30,35
30       LL=J
         GO TO 40
35       CONTINUE
40       STIME=(FLOAT(IMSEC)*10.0**ITBM)*0.001

C        COMPUTE SUMS AND SUMS OF SQUARES OF SACCADE
C        DATA.

         DO 45 M=1,5
         DUMY=(FLOAT(IDATA(K*16+M+6)))*STIME
         DATA(LL,M)=DATA(LL,M)+DUMY
         STUPE=DUMY**2
         DATA(LL,M+6)=DATA(LL,M+6)+STUPE
45       CONTINUE
         DATA(LL,6)=DATA(LL,6)+1.0
100      CONTINUE
```

```
C       DONE WITH SUMMARY FILES?

        IF(ILL-ILH)150,200,150
150     ILA=0
S       TAD \ILL           /INCR SUFLI FILENUMBER
S       TAD (0007          /(UNITS+7), INCR TENS IF REQD
S       DCA \ILL           /SAVE NEW TENS
S       TAD \ILL
S       AND (0077          /UNITS+7
S       SNA
S       TAD (0066          /FIXUP FOR ZERO
S       TAD (-6            /(UNITS+7)-6=UNITS+1
S       DCA \ILA           /SAVE NEW UNITS
S       TAD \ILL           /PUT NEW UNITS WITH TENS
S       AND (7700
S       TAD \ILA
S       DCA \ILL
SCALL 0,CLEAR              /CLEAR F.P.A.C.
SCALL 1,FAD                /INSERT NEW FILENUMBER
SARG    \SUFLI             /INTO SUFLI
S       TAD \ILL
S       DCA ACL
SCALL 1,STO
SARG    \SUFLI
        GO TO 10

C       YES! GO BACK TO SACCAV FOR FINAL RESULTS

200     CALL CHAIN('SACCAV')

        STOP
        END
*
```

APPENDIX C: RAW DATA

The data presented in the tables given here is the average of 4342 saccade responses of subject DEB. The plots of Fig.'s 8 to 23 were generated using the RVL, LTCY, and ST15 data of these tables. Each table represents the averaged responses for a single target position (radius and angle). The responses for a target position were averaged according to the filter number used for that trial. The conventions for each table are as follows: (1) F represents the filter number, (2) LTCY is saccade latency, (3) ST25, ST20, ST15 and ST10 are the settling times to 25, 20, 15 and 10 minutes arc respectively, (4) RVL is the calculated value of relative visibility level corresponding to that target position and filter number, (5) N is the number of responses averaged for each filter number, (6) $\overline{X}$ is the mean of the responses, (7) S is the standard deviation of the responses.

TABLE 4

RADIUS: 2.0°   ANGLE: 0°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.717	$\bar{X}$ S	0.457 0.136	0.725 0.352	0.772 0.394	0.835 0.387	0.999 0.399	11
1	0.932	$\bar{X}$ S	0.513 0.186	0.688 0.254	0.817 0.272	0.864 0.232	1.116 0.502	9
2	1.219	$\bar{X}$ S	0.387 0.122	0.736 0.277	0.791 0.296	0.931 0.298	1.113 0.323	17
3	1.584	$\bar{X}$ S	0.381 0.074	0.475 0.117	0.546 0.163	0.559 0.155	0.936 0.331	7
4	2.065	$\bar{X}$ S	0.277 0.039	0.350 0.100	0.378 0.080	0.470 0.154	0.714 0.507	11
5	2.694	$\bar{X}$ S	0.261 0.041	0.314 0.068	0.350 0.116	0.374 0.118	0.495 0.256	10
6	3.509	$\bar{X}$ S	0.253 0.036	0.287 0.036	0.293 0.037	0.338 0.098	0.510 0.268	10
7	4.572	$\bar{X}$ S	0.255 0.020	0.281 0.024	0.324 0.074	0.352 0.082	0.461 0.242	12
8	5.595	$\bar{X}$ S	0.246 0.019	0.283 0.031	0.308 0.068	0.354 0.099	0.486 0.242	14

TABLE 5

RADIUS: 2.0°   ANGLE: 45°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.888	$\overline{X}$ S	0.406 0.118	0.555 0.174	0.622 0.172	0.724 0.275	0.994 0.408	17
1	1.154	$\overline{X}$ S	0.338 0.109	0.506 0.100	0.575 0.171	0.699 0.237	0.895 0.247	14
2	1.510	$\overline{X}$ S	0.344 0.074	0.488 0.196	0.543 0.225	0.618 0.195	0.803 0.270	16
3	1.962	$\overline{X}$ S	0.292 0.065	0.380 0.071	0.424 0.066	0.503 0.142	0.552 0.145	12
4	2.559	$\overline{X}$ S	0.266 0.043	0.364 0.107	0.413 0.112	0.557 0.180	0.745 0.399	11
5	3.338	$\overline{X}$ S	0.246 0.037	0.301 0.073	0.354 0.084	0.434 0.114	0.580 0.196	11
6	4.348	$\overline{X}$ S	0.260 0.036	0.335 0.066	0.377 0.089	0.402 0.093	0.514 0.124	12
7	5.666	$\overline{X}$ S	0.238 0.021	0.284 0.045	0.317 0.075	0.332 0.079	0.556 0.204	15
8	7.378	$\overline{X}$ S	0.249 0.031	0.329 0.057	0.377 0.090	0.430 0.072	0.555 0.208	10

TABLE 6

RADIUS: 2.0°   ANGLE: 90°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.946	$\bar{X}$ S	0.471 0.110	0.685 0.231	0.709 0.235	0.744 0.212	0.944 0.343	13
1	1.230	$\bar{X}$ S	0.405 0.130	0.639 0.166	0.674 0.145	0.745 0.180	0.914 0.285	13
2	1.610	$\bar{X}$ S	0.392 0.122	0.603 0.215	0.639 0.195	0.695 0.188	0.960 0.332	13
3	2.091	$\bar{X}$ S	0.285 0.060	0.325 0.086	0.349 0.091	0.381 0.092	0.487 0.212	13
4	2.727	$\bar{X}$ S	0.252 0.030	0.297 0.064	0.339 0.075	0.409 0.084	0.551 0.213	10
5	3.557	$\bar{X}$ S	0.235 0.033	0.294 0.074	0.339 0.126	0.421 0.279	0.553 0.360	11
6	4.634	$\bar{X}$ S	0.247 0.038	0.304 0.084	0.343 0.092	0.394 0.113	0.484 0.204	19
7	6.038	$\bar{X}$ S	0.232 0.026	0.251 0.023	0.268 0.018	0.303 0.052	0.413 0.170	10
8	7.863	$\bar{X}$ S	0.225 0.037	0.285 0.098	0.315 0.113	0.382 0.128	0.559 0.227	10

TABLE 7

RADIUS: 2.0° ANGLE: 135°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.886	$\bar{X}$	0.457	0.603	0.654	0.712	0.883	18
		S	0.146	0.216	0.251	0.260	0.369	
1	1.151	$\bar{X}$	0.414	0.458	0.559	0.628	0.854	9
		S	0.122	0.100	0.087	0.100	0.311	
2	1.506	$\bar{X}$	0.411	0.488	0.494	0.605	0.767	11
		S	0.186	0.178	0.178	0.262	0.386	
3	1.957	$\bar{X}$	0.281	0.346	0.391	0.465	0.733	5
		S	0.093	0.126	0.109	0.167	0.250	
4	2.552	$\bar{X}$	0.259	0.309	0.331	0.377	0.479	15
		S	0.029	0.056	0.062	0.112	0.214	
5	3.330	$\bar{X}$	0.257	0.332	0.386	0.440	0.611	13
		S	0.041	0.087	0.112	0.122	0.261	
6	4.337	$\bar{X}$	0.235	0.259	0.276	0.296	0.481	18
		S	0.018	0.020	0.053	0.072	0.259	
7	5.652	$\bar{X}$	0.264	0.296	0.326	0.373	0.509	17
		S	0.051	0.077	0.107	0.113	0.196	
8	7.359	$\bar{X}$	0.247	0.293	0.304	0.363	0.506	11
		S	0.020	0.067	0.079	0.096	0.284	

TABLE 8

RADIUS: 2.0° ANGLE: 180°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.964	$\bar{X}$ S	0.387 0.162	0.527 0.229	0.576 0.229	0.650 0.241	0.817 0.313	15
1	1.253	$\bar{X}$ S	0.403 0.138	0.450 0.130	0.492 0.179	0.622 0.214	0.943 0.370	12
2	1.640	$\bar{X}$ S	0.359 0.095	0.432 0.093	0.475 0.133	0.630 0.294	0.714 0.329	13
3	2.130	$\bar{X}$ S	0.257 0.045	0.408 0.277	0.423 0.299	0.473 0.327	0.626 0.395	10
4	2.778	$\bar{X}$ S	0.255 0.070	0.325 0.096	0.342 0.107	0.365 0.121	0.451 0.158	11
5	3.624	$\bar{X}$ S	0.254 0.030	0.295 0.065	0.350 0.107	0.378 0.120	0.425 0.185	13
6	4.720	$\bar{X}$ S	0.231 0.009	0.256 0.010	0.264 0.011	0.325 0.144	0.361 0.196	7
7	6.151	$\bar{X}$ S	0.244 0.021	0.291 0.058	0.294 0.058	0.315 0.075	0.447 0.176	11
8	8.010	$\bar{X}$ S	0.240 0.028	0.283 0.053	0.298 0.066	0.356 0.148	0.545 0.206	13

TABLE 9

RADIUS: 2.0° ANGLE: 225°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.602	$\bar{X}$ S	0.479 0.114	0.827 0.230	0.954 0.281	1.023 0.195	1.083 0.159	6
1	0.782	$\bar{X}$ S	0.506 0.078	0.682 0.275	0.796 0.308	0.967 0.257	1.077 0.146	6
2	1.023	$\bar{X}$ S	0.455 0.156	0.712 0.294	0.781 0.340	0.951 0.381	1.136 0.519	8
3	1.330	$\bar{X}$ S	0.381 0.127	0.519 0.157	0.602 0.203	0.646 0.279	0.767 0.297	13
4	1.734	$\bar{X}$ S	0.240 0.069	0.425 0.181	0.480 0.177	0.511 0.205	0.683 0.256	10
5	2.262	$\bar{X}$ S	0.298 0.061	0.394 0.130	0.415 0.130	0.491 0.092	0.753 0.336	10
6	2.946	$\bar{X}$ S	0.232 0.055	0.331 0.108	0.390 0.114	0.436 0.119	0.639 0.327	8
7	3.839	$\bar{X}$ S	0.252 0.018	0.294 0.059	0.311 0.084	0.372 0.102	0.587 0.313	24
8	4.999	$\bar{X}$ S	0.237 0.037	0.297 0.057	0.334 0.113	0.401 0.161	0.595 0.198	16

TABLE 10

RADIUS: 2.0°   ANGLE: 270°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.439	$\bar{X}$ S	0.502 0.105	1.032 0.188	1.142 0.200	1.278 0.272	1.438 0.330	6
1	0.571	$\bar{X}$ S	0.565 0.149	0.904 0.092	0.906 0.090	0.909 0.092	0.956 0.106	4
2	0.747	$\bar{X}$ S	0.245 --	0.565 --	0.780 --	0.785 --	0.795 --	1
3	0.970	$\bar{X}$ S	0.506 0.091	0.848 0.235	0.858 0.233	0.897 0.252	1.146 0.375	9
4	1.265	$\bar{X}$ S	0.374 0.083	0.539 0.174	0.558 0.158	0.722 0.229	0.986 0.225	9
5	1.651	$\bar{X}$ S	0.395 0.132	0.520 0.283	0.557 0.275	0.635 0.288	0.817 0.335	12
6	2.150	$\bar{X}$ S	0.287 0.040	0.398 0.101	0.438 0.100	0.485 0.117	0.647 0.270	19
7	2.802	$\bar{X}$ S	0.254 0.040	0.376 0.098	0.417 0.109	0.466 0.100	0.613 0.246	23
8	3.649	$\bar{X}$ S	0.247 0.032	0.323 0.087	0.357 0.085	0.396 0.087	0.533 0.216	14

TABLE 11

RADIUS: 2.0° ANGLE: 315°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.456	$\bar{X}$ S	0.564 0.098	1.079 0.447	1.193 0.394	1.340 0.481	1.679 0.708	5
1	0.592	$\bar{X}$ S	0.526 0.085	0.738 0.343	0.750 0.353	0.829 0.382	1.000 0.522	5
2	0.775	$\bar{X}$ S	0.484 0.099	0.824 0.449	0.848 0.434	0.918 0.442	1.211 0.507	11
3	1.007	$\bar{X}$ S	0.392 0.104	0.574 0.242	0.698 0.216	0.798 0.216	0.947 0.248	10
4	1.313	$\bar{X}$ S	0.377 0.088	0.471 0.185	0.561 0.295	0.649 0.282	0.859 0.409	8
5	1.713	$\bar{X}$ S	0.283 0.056	0.391 0.139	0.461 0.115	0.529 0.164	0.669 0.319	7
6	2.231	$\bar{X}$ S	0.255 0.042	0.286 0.061	0.311 0.068	0.436 0.178	0.531 0.181	18
7	2.907	$\bar{X}$ S	0.236 0.037	0.293 0.086	0.332 0.104	0.350 0.105	0.457 0.149	15
8	3.786	$\bar{X}$ S	0.222 0.039	0.270 0.066	0.302 0.093	0.393 0.090	0.533 0.169	13

TABLE 12

RADIUS: 2.5°   ANGLE: 0°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.888	$\bar{X}$ S	0.412 0.152	0.527 0.157	0.529 0.158	0.621 0.229	0.731 0.248	15
1	1.154	$\bar{X}$ S	0.401 0.131	0.508 0.168	0.527 0.184	0.697 0.278	0.845 0.356	15
2	1.510	$\bar{X}$ S	0.376 0.099	0.469 0.098	0.527 0.145	0.603 0.144	0.644 0.162	15
3	1.962	$\bar{X}$ S	0.262 0.044	0.306 0.070	0.317 0.068	0.417 0.200	0.560 0.165	9
4	2.559	$\bar{X}$ S	0.254 0.033	0.285 0.037	0.338 0.088	0.361 0.088	0.519 0.153	8
5	3.338	$\bar{X}$ S	0.256 0.031	0.289 0.030	0.302 0.044	0.343 0.081	0.567 0.233	11
6	4.358	$\bar{X}$ S	0.244 0.027	0.795 0.065	0.314 0.081	0.415 0.123	0.539 0.143	17
7	5.666	$\bar{X}$ S	0.242 0.023	0.272 0.019	0.274 0.019	0.346 0.118	0.523 0.373	10
8	7.379	$\bar{X}$ S	0.242 0.024	0.279 0.011	0.322 0.079	0.350 0.101	0.433 0.139	16

TABLE 13

RADIUS: 2.5°   ANGLE: 45°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.145	$\bar{X}$ S	0.326 0.080	0.400 0.107	0.482 0.152	0.575 0.163	0.761 0.367	18
1	1.489	$\bar{X}$ S	0.337 0.075	0.541 0.176	0.587 0.182	0.607 0.182	0.760 0.223	17
2	1.948	$\bar{X}$ S	0.278 0.049	0.404 0.154	0.443 0.153	0.504 0.175	0.589 0.214	16
3	2.531	$\bar{X}$ S	0.252 0.034	0.367 0.136	0.430 0.156	0.451 0.145	0.505 0.177	12
4	3.300	$\bar{X}$ S	0.238 0.024	0.366 0.078	0.407 0.108	0.431 0.105	0.494 0.108	10
5	4.305	$\bar{X}$ S	0.245 0.023	0.297 0.048	0.325 0.063	0.396 0.119	0.523 0.171	15
6	5.619	$\bar{X}$ S	0.238 0.024	0.318 0.104	0.366 0.141	0.523 0.248	0.580 0.305	14
7	7.307	$\bar{X}$ S	0.230 0.040	0.287 0.057	0.315 0.045	0.398 0.101	0.477 0.137	8
8	9.515	$\bar{X}$ S	0.205 0.027	0.310 0.086	0.354 0.110	0.452 0.044	0.656 0.267	6

TABLE 14

RADIUS: 2.5°　　　ANGLE: 90°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.095	$\bar{X}$ S	0.286 0.049	0.433 0.112	0.484 0.102	0.519 0.128	0.575 0.126	14
1	1.423	$\bar{X}$ S	0.283 0.055	0.410 0.109	0.447 0.119	0.541 0.142	0.638 0.252	12
2	1.862	$\bar{X}$ S	0.260 0.071	0.374 0.112	0.392 0.128	0.460 0.155	0.597 0.212	15
3	2.419	$\bar{X}$ S	0.237 0.033	0.328 0.108	0.369 0.116	0.401 0.120	0.544 0.234	16
4	3.154	$\bar{X}$ S	0.230 0.042	0.282 0.052	0.314 0.084	0.395 0.153	0.507 0.324	14
5	4.115	$\bar{X}$ S	0.239 0.034	0.309 0.075	0.356 0.085	0.401 0.111	0.587 0.346	9
6	5.372	$\bar{X}$ S	0.214 0.029	0.254 0.054	0.280 0.056	0.381 0.077	0.535 0.206	12
7	6.985	$\bar{X}$ S	0.239 0.035	0.285 0.050	0.349 0.080	0.384 0.097	0.442 0.113	14
8	9.096	$\bar{X}$ S	0.227 0.028	0.302 0.089	0.337 0.113	0.377 0.109	0.468 0.106	12

TABLE 15

RADIUS: 2.5°   ANGLE: 135°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.145	$\bar{X}$ S	0.359 0.065	0.431 0.108	0.465 0.126	0.519 0.178	0.731 0.213	18
1	1.489	$\bar{X}$ S	0.284 0.065	0.397 0.075	0.420 0.080	0.438 0.070	0.634 0.231	14
2	1.948	$\bar{X}$ S	0.268 0.045	0.402 0.155	0.447 0.149	0.503 0.186	0.654 0.304	12
3	2.531	$\bar{X}$ S	0.243 0.025	0.335 0.079	0.397 0.092	0.415 0.086	0.520 0.161	11
4	3.300	$\bar{X}$ S	0.270 0.076	0.365 0.107	0.425 0.099	0.465 0.104	0.517 0.158	10
5	4.305	$\bar{X}$ S	0.217 0.053	0.278 0.061	0.347 0.084	0.417 0.084	0.455 0.080	11
6	5.619	$\bar{X}$ S	0.238 0.032	0.311 0.087	0.350 0.108	0.415 0.169	0.608 0.237	22
7	7.307	$\bar{X}$ S	0.255 0.051	0.295 0.087	0.340 0.097	0.427 0.166	0.592 0.337	11
8	9.515	$\bar{X}$ S	0.243 0.024	0.299 0.048	0.347 0.095	0.392 0.097	0.528 0.209	12

TABLE 16

RADIUS: 2.5°   ANGLE: 180°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.080	$\bar{X}$ S	0.397 0.140	0.568 0.272	0.627 0.273	0.668 0.275	0.935 0.339	13
1	1.404	$\bar{X}$ S	0.311 0.092	0.430 0.134	0.465 0.133	0.492 0.133	0.663 0.263	19
2	1.837	$\bar{X}$ S	0.272 0.065	0.353 0.133	0.373 0.143	0.407 0.155	0.550 0.251	16
3	2.387	$\bar{X}$ S	0.244 0.055	0.378 0.085	0.384 0.086	0.514 0.223	0.592 0.289	8
4	3.113	$\bar{X}$ S	0.244 0.027	0.292 0.035	0.314 0.078	0.326 0.077	0.535 0.268	8
5	4.061	$\bar{X}$ S	0.298 0.102	0.355 0.121	0.395 0.176	0.450 0.255	0.692 0.360	11
6	5.301	$\bar{X}$ S	0.237 0.036	0.288 0.074	0.328 0.120	0.372 0.144	0.519 0.200	17
7	6.893	$\bar{X}$ S	0.232 0.021	0.296 0.080	0.322 0.120	0.428 0.219	0.566 0.264	18
8	8.976	$\bar{X}$ S	0.230 0.027	0.271 0.036	0.274 0.037	0.300 0.068	0.477 0.192	8

TABLE 17

RADIUS: 2.5°   ANGLE: 225°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.680	$\bar{X}$ S	0.535 0.115	0.772 0.320	0.795 0.324	0.940 0.347	1.038 0.301	10
1	0.884	$\bar{X}$ S	0.439 0.074	0.980 0.419	1.100 0.436	1.127 0.425	1.315 0.343	6
2	1.156	$\bar{X}$ S	0.453 0.110	0.742 0.404	0.863 0.385	0.944 0.340	1.101 0.539	11
3	1.502	$\bar{X}$ S	0.320 0.058	0.441 0.105	0.502 0.134	0.554 0.147	0.610 0.163	14
4	1.959	$\bar{X}$ S	0.344 0.111	0.471 0.146	0.542 0.156	0.593 0.170	0.731 0.190	8
5	2.555	$\bar{X}$ S	0.246 0.016	0.324 0.067	0.338 0.071	0.465 0.196	0.602 0.300	11
6	3.335	$\bar{X}$ S	0.245 0.023	0.331 0.078	0.378 0.093	0.398 0.093	0.470 0.160	13
7	4.337	$\bar{X}$ S	0.252 0.020	0.303 0.063	0.366 0.111	0.428 0.133	0.582 0.242	20
8	5.647	$\bar{X}$ S	0.234 0.027	0.293 0.043	0.324 0.083	0.410 0.127	0.593 0.217	17

TABLE 18

RADIUS: 2.5°   ANGLE: 270°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.434	$\bar{X}$	0.482	1.081	1.137	1.165	1.318	12
		S	0.151	0.299	0.294	0.278	0.191	
1	0.565	$\bar{X}$	0.562	1.136	1.137	1.145	1.435	8
		S	0.118	0.374	0.374	0.371	0.335	
2	0.739	$\bar{X}$	0.526	1.087	1.091	1.104	1.237	14
		S	0.152	0.293	0.293	0.289	0.291	
3	0.960	$\bar{X}$	0.453	0.757	0.777	0.819	0.957	8
		S	0.095	0.333	0.326	0.329	0.246	
4	1.252	$\bar{X}$	0.284	0.428	0.461	0.528	0.577	10
		S	0.031	0.105	0.091	0.112	0.115	
5	1.633	$\bar{X}$	0.267	0.394	0.433	0.468	0.590	13
		S	0.040	0.106	0.095	0.092	0.270	
6	2.132	$\bar{X}$	0.249	0.354	0.391	0.484	0.676	19
		S	0.024	0.086	0.078	0.154	0.386	
7	2.772	$\bar{X}$	0.242	0.331	0.404	0.427	0.634	13
		S	0.022	0.077	0.084	0.080	0.241	
8	3.610	$\bar{X}$	0.244	0.295	0.335	0.360	0.530	17
		S	0.024	0.058	0.091	0.101	0.271	

TABLE 19

RADIUS: 2.5°   ANGLE: 315°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.501	$\bar{X}$ S	0.505 0.134	0.878 0.270	0.982 0.295	1.077 0.352	1.173 0.298	15
1	0.652	$\bar{X}$ S	0.488 0.159	1.006 0.334	1.013 0.328	1.113 0.339	1.237 0.341	9
2	0.853	$\bar{X}$ S	0.447 0.100	0.695 0.231	0.701 0.228	0.743 0.190	0.800 0.192	11
3	1.108	$\bar{X}$ S	0.319 0.062	0.443 0.124	0.467 0.124	0.530 0.172	0.819 0.501	10
4	1.444	$\bar{X}$ S	0.290 0.039	0.402 0.110	0.432 0.101	0.453 0.097	0.622 0.330	12
5	1.884	$\bar{X}$ S	0.270 0.044	0.366 0.103	0.396 0.090	0.420 0.087	0.621 0.326	18
6	2.460	$\bar{X}$ S	0.244 0.039	0.303 0.082	0.343 0.100	0.462 0.105	0.615 0.243	10
7	3.198	$\bar{X}$ S	0.239 0.032	0.282 0.053	0.321 0.077	0.411 0.125	0.614 0.366	8
8	4.165	$\bar{X}$ S	0.240 0.033	0.287 0.047	0.334 0.085	0.429 0.160	0.589 0.198	20

TABLE 20

RADIUS: 3.0°   ANGLE: 0°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.027	$\bar{X}$ S	0.394 0.091	0.486 0.140	0.526 0.143	0.599 0.209	0.790 0.339	14
1	1.335	$\bar{X}$ S	0.400 0.124	0.635 0.364	0.646 0.362	0.737 0.356	0.902 0.349	12
2	1.747	$\bar{X}$ S	0.354 0.117	0.430 0.160	0.477 0.155	0.622 0.238	0.756 0.304	11
3	2.270	$\bar{X}$ S	0.242 0.020	0.298 0.049	0.341 0.077	0.375 0.084	0.423 0.127	6
4	2.960	$\bar{X}$ S	0.246 0.030	0.326 0.071	0.397 0.070	0.456 0.067	0.566 0.179	10
5	3.862	$\bar{X}$ S	0.241 0.032	0.310 0.079	0.358 0.104	0.382 0.117	0.533 0.264	20
6	5.030	$\bar{X}$ S	0.226 0.051	0.311 0.078	0.355 0.093	0.384 0.122	0.486 0.162	25
7	6.555	$\bar{X}$ S	0.234 0.018	0.282 0.052	0.309 0.083	0.329 0.083	0.457 0.195	9
8	8.536	$\bar{X}$ S	0.241 0.019	0.280 0.025	0.288 0.031	0.331 0.081	0.481 0.188	14

TABLE 21

RADIUS: 3.0° ANGLE: 45°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.243	$\overline{X}$ S	0.273 0.043	0.450 0.066	0.480 0.084	0.543 0.112	0.640 0.189	20
1	1.616	$\overline{X}$ S	0.307 0.051	0.426 0.099	0.444 0.107	0.554 0.167	0.639 0.233	12
2	2.114	$\overline{X}$ S	0.279 0.091	0.428 0.150	0.449 0.143	0.512 0.129	0.557 0.160	11
3	2.747	$\overline{X}$ S	0.239 0.016	0.282 0.015	0.317 0.069	0.415 0.092	0.506 0.079	6
4	3.582	$\overline{X}$ S	0.235 0.025	0.360 0.091	0.435 0.068	0.455 0.042	0.514 0.079	11
5	4.673	$\overline{X}$ S	0.346 0.172	0.576 0.372	0.636 0.370	0.706 0.395	0.800 0.410	7
6	6.087	$\overline{X}$ S	0.225 0.029	0.343 0.083	0.387 0.117	0.442 0.093	0.573 0.223	13
7	7.931	$\overline{X}$ S	0.235 0.025	0.332 0.082	0.385 0.079	0.422 0.063	0.608 0.314	15
8	10.328	$\overline{X}$ S	0.242 0.027	0.315 0.058	0.347 0.078	0.419 0.080	0.491 0.174	15

TABLE 22

RADIUS: 3.0° ANGLE: 90°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.430	$\bar{X}$ S	0.251 0.054	0.340 0.080	0.368 0.092	0.402 0.112	0.447 0.126	16
1	1.859	$\bar{X}$ S	0.243 0.047	0.375 0.093	0.411 0.096	0.445 0.083	0.565 0.177	20
2	2.433	$\bar{X}$ S	0.240 0.039	0.382 0.093	0.401 0.092	0.415 0.092	0.567 0.282	21
3	3.161	$\bar{X}$ S	0.239 0.034	0.394 0.300	0.440 0.309	0.447 0.311	0.577 0.395	8
4	4.122	$\bar{X}$ S	0.234 0.042	0.292 0.050	0.335 0.061	0.426 0.086	0.507 0.089	8
5	5.377	$\bar{X}$ S	0.221 0.028	0.281 0.088	0.321 0.103	0.368 0.110	0.511 0.204	7
6	7.004	$\bar{X}$ S	0.230 0.024	0.271 0.040	0.371 0.105	0.387 0.186	0.685 0.415	10
7	9.127	$\bar{X}$ S	0.229 0.026	0.347 0.099	0.377 0.087	0.422 0.077	0.487 0.095	11
8	11.885	$\bar{X}$ S	0.221 0.037	0.323 0.089	0.333 0.083	0.383 0.121	0.531 0.347	15

TABLE 23

RADIUS: 3.0° ANGLE: 135°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.404	$\bar{X}$ S	0.278 0.056	0.352 0.066	0.383 0.086	0.418 0.113	0.512 0.149	13
1	1.825	$\bar{X}$ S	0.255 0.069	0.363 0.085	0.395 0.103	0.404 0.097	0.580 0.248	12
2	2.388	$\bar{X}$ S	0.252 0.059	0.394 0.123	0.447 0.132	0.521 0.220	0.567 0.215	13
3	3.103	$\bar{X}$ S	0.251 0.051	0.337 0.114	0.361 0.114	0.418 0.129	0.495 0.139	11
4	4.046	$\bar{X}$ S	0.220 0.038	0.308 0.087	0.357 0.113	0.435 0.105	0.470 0.098	11
5	5.278	$\bar{X}$ S	0.227 0.031	0.363 0.088	0.391 0.108	0.486 0.206	0.741 0.457	9
6	6.875	$\bar{X}$ S	0.221 0.018	0.312 0.086	0.388 0.094	0.448 0.144	0.584 0.342	18
7	8.959	$\bar{X}$ S	0.232 0.022	0.330 0.132	0.360 0.133	0.421 0.157	0.535 0.210	15
8	11.667	$\bar{X}$ S	0.223 0.028	0.347 0.089	0.369 0.092	0.421 0.093	0.475 0.166	10

TABLE 24

RADIUS: 3.0° ANGLE: 180°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.266	$\bar{X}$ S	0.288 0.062	0.438 0.068	0.471 0.084	0.520 0.124	0.683 0.246	20
1	1.646	$\bar{X}$ S	0.262 0.067	0.363 0.113	0.390 0.118	0.425 0.103	0.479 0.128	9
2	2.154	$\bar{X}$ S	0.251 0.049	0.312 0.110	0.382 0.113	0.599 0.276	0.798 0.355	9
3	2.798	$\bar{X}$ S	0.271 0.081	0.374 0.128	0.421 0.128	0.425 0.128	0.471 0.128	15
4	3.649	$\bar{X}$ S	0.268 0.083	0.429 0.251	0.450 0.257	0.563 0.297	0.572 0.300	14
5	4.760	$\bar{X}$ S	0.231 0.037	0.313 0.090	0.319 0.092	0.441 0.139	0.502 0.193	8
6	6.201	$\bar{X}$ S	0.226 0.030	0.287 0.052	0.342 0.101	0.400 0.122	0.502 0.210	16
7	8.080	$\bar{X}$ S	0.231 0.035	0.270 0.043	0.273 0.043	0.329 0.150	0.445 0.205	6
8	10.522	$\bar{X}$ S	0.277 0.085	0.406 0.145	0.409 0.145	0.411 0.146	0.573 0.202	8

TABLE 25

RADIUS: 3.0° ANGLE: 225°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.784	$\bar{X}$ S	0.430 0.078	0.565 0.137	0.590 0.131	0.684 0.109	0.862 0.267	14
1	1.019	$\bar{X}$ S	0.408 0.092	0.594 0.165	0.596 0.164	0.655 0.172	0.754 0.230	14
2	1.334	$\bar{X}$ S	0.388 0.090	0.479 0.149	0.521 0.152	0.635 0.213	0.766 0.338	10
3	1.733	$\bar{X}$ S	0.255 0.042	0.361 0.120	0.381 0.136	0.408 0.131	0.547 0.266	12
4	2.260	$\bar{X}$ S	0.259 0.038	0.344 0.082	0.398 0.101	0.502 0.134	0.701 0.297	14
5	2.948	$\bar{X}$ S	0.236 0.028	0.351 0.067	0.398 0.062	0.477 0.122	0.588 0.286	13
6	3.840	$\bar{X}$ S	0.224 0.022	0.280 0.057	0.350 0.084	0.553 0.321	0.697 0.370	13
7	5.004	$\bar{X}$ S	0.229 0.037	0.363 0.080	0.390 0.064	0.423 0.072	0.470 0.075	13
8	6.516	$\bar{X}$ S	0.234 0.023	0.312 0.073	0.331 0.088	0.386 0.109	0.479 0.168	12

TABLE 26

RADIUS: 3.0°   ANGLE: 270°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.598	$\bar{X}$ S	0.418 0.154	0.876 0.321	0.934 0.320	0.993 0.309	1.144 0.217	13
1	0.778	$\bar{X}$ S	0.490 0.128	0.811 0.278	0.851 0.259	0.916 0.319	1.065 0.384	17
2	1.018	$\bar{X}$ S	0.466 0.131	0.732 0.273	0.779 0.271	0.862 0.282	0.920 0.305	16
3	1.323	$\bar{X}$ S	0.371 0.134	0.638 0.375	0.666 0.412	0.730 0.407	0.902 0.388	12
4	1.725	$\bar{X}$ S	0.312 0.100	0.490 0.290	0.534 0.327	0.664 0.385	0.933 0.322	9
5	2.250	$\bar{X}$ S	0.276 0.036	0.396 0.098	0.424 0.104	0.435 0.103	0.569 0.279	7
6	2.931	$\bar{X}$ S	0.234 0.028	0.309 0.078	0.352 0.085	0.374 0.096	0.555 0.235	12
7	3.819	$\bar{X}$ S	0.228 0.021	0.338 0.083	0.376 0.086	0.427 0.133	0.558 0.235	16
8	4.973	$\bar{X}$ S	0.231 0.021	0.308 0.063	0.327 0.068	0.367 0.107	0.491 0.259	13

TABLE 27

RADIUS: 3.0° ANGLE: 315°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.597	$\bar{X}$ S	0.438 0.105	0.826 0.259	0.870 0.269	0.962 0.354	1.136 0.374	23
1	0.776	$\bar{X}$ S	0.461 0.145	0.863 0.318	0.867 0.315	0.922 0.281	1.019 0.272	18
2	1.015	$\bar{X}$ S	0.329 0.083	0.550 0.197	0.556 0.197	0.581 0.231	0.702 0.223	9
3	1.319	$\bar{X}$ S	0.342 0.131	0.512 0.292	0.565 0.291	0.610 0.271	0.931 0.520	12
4	1.720	$\bar{X}$ S	0.268 0.043	0.390 0.094	0.442 0.146	0.475 0.136	0.692 0.316	13
5	2.244	$\bar{X}$ S	0.231 0.037	0.326 0.118	0.366 0.121	0.404 0.164	0.470 0.165	9
6	2.923	$\bar{X}$ S	0.232 0.027	0.311 0.073	0.379 0.184	0.447 0.278	0.584 0.308	16
7	3.809	$\bar{X}$ S	0.227 0.026	0.266 0.023	0.334 0.111	0.381 0.119	0.486 0.153	8
8	4.960	$\bar{X}$ S	0.240 0.026	0.343 0.081	0.365 0.082	0.431 0.128	0.545 0.162	12

TABLE 28

RADIUS: 3.5°   ANGLE: 0°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.176	$\bar{X}$ S	0.360 0.117	0.558 0.204	0.583 0.196	0.667 0.220	0.843 0.289	18
1	1.529	$\bar{X}$ S	0.295 0.058	0.386 0.094	0.472 0.114	0.540 0.209	0.728 0.373	11
2	2.000	$\bar{X}$ S	0.299 0.090	0.433 0.112	0.461 0.114	0.488 0.148	0.605 0.243	13
3	2.599	$\bar{X}$ S	0.245 0.040	0.352 0.068	0.353 0.067	0.426 0.076	0.570 0.156	5
4	3.388	$\bar{X}$ S	0.229 0.023	0.332 0.078	0.334 0.077	0.403 0.133	0.632 0.340	10
5	4.420	$\bar{X}$ S	0.230 0.034	0.316 0.067	0.344 0.084	0.386 0.105	0.510 0.169	14
6	5.758	$\bar{X}$ S	0.263 0.032	0.385 0.085	0.414 0.087	0.450 0.079	0.498 0.118	11
7	7.503	$\bar{X}$ S	0.239 0.027	0.334 0.089	0.356 0.101	0.439 0.134	0.502 0.162	17
8	9.770	$\bar{X}$ S	0.241 0.022	0.307 0.055	0.322 0.066	0.366 0.114	0.561 0.245	14

TABLE 29

RADIUS: 3.5°   ANGLE: 45°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.342	$\bar{X}$ S	0.268 0.033	0.425 0.072	0.495 0.105	0.592 0.167	0.645 0.183	11
1	1.745	$\bar{X}$ S	0.262 0.066	0.425 0.086	0.487 0.078	0.559 0.163	0.606 0.177	8
2	2.283	$\bar{X}$ S	0.254 0.029	0.402 0.076	0.444 0.072	0.471 0.075	0.510 0.075	14
3	2.966	$\bar{X}$ S	0.280 0.131	0.411 0.189	0.464 0.184	0.509 0.171	0.687 0.325	11
4	3.868	$\bar{X}$ S	0.238 0.031	0.351 0.093	0.392 0.091	0.430 0.094	0.495 0.106	17
5	5.046	$\bar{X}$ S	0.226 0.027	0.357 0.080	0.385 0.080	0.455 0.067	0.528 0.129	14
6	6.573	$\bar{X}$ S	0.251 0.050	0.386 0.109	0.432 0.089	0.450 0.083	0.499 0.094	16
7	8.565	$\bar{X}$ S	0.233 0.023	0.365 0.093	0.399 0.104	0.443 0.090	0.505 0.108	12
8	11.153	$\bar{X}$ S	0.235 0.029	0.347 0.085	0.379 0.100	0.385 0.094	0.453 0.103	12

TABLE 30

RADIUS: 3.5° ANGLE: 90°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.590	$\bar{X}$ S	0.252 0.046	0.395 0.078	0.429 0.085	0.464 0.137	0.600 0.254	12
1	2.067	$\bar{X}$ S	0.291 0.086	0.451 0.140	0.465 0.137	0.536 0.153	0.648 0.246	15
2	2.704	$\bar{X}$ S	0.271 0.040	0.385 0.109	0.414 0.101	0.429 0.088	0.492 0.192	5
3	3.514	$\bar{X}$ S	0.220 0.016	0.348 0.101	0.368 0.094	0.487 0.008	0.559 0.177	6
4	4.582	$\bar{X}$ S	0.236 0.029	0.311 0.086	0.346 0.100	0.369 0.117	0.474 0.161	9
5	5.977	$\bar{X}$ S	0.226 0.040	0.353 0.107	0.383 0.123	0.432 0.116	0.582 0.259	13
6	7.786	$\bar{X}$ S	0.226 0.038	0.358 0.082	0.384 0.083	0.435 0.103	0.530 0.215	20
7	10.146	$\bar{X}$ S	0.229 0.031	0.336 0.086	0.372 0.091	0.400 0.087	0.455 0.105	17
8	13.212	$\bar{X}$ S	0.229 0.023	0.333 0.088	0.392 0.092	0.442 0.141	0.517 0.173	13

TABLE 31

RADIUS: 3.5°   ANGLE: 135°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.375	$\bar{X}$ S	0.251 0.053	0.352 0.103	0.404 0.123	0.444 0.115	0.573 0.173	21
1	1.787	$\bar{X}$ S	0.246 0.039	0.355 0.094	0.374 0.095	0.402 0.083	0.457 0.085	10
2	2.338	$\bar{X}$ S	0.231 0.022	0.362 0.100	0.382 0.095	0.414 0.118	0.532 0.149	14
3	3.038	$\bar{X}$ S	0.232 0.028	0.308 0.074	0.371 0.084	0.428 0.075	0.622 0.395	9
4	3.961	$\bar{X}$ S	0.222 0.057	0.340 0.087	0.377 0.095	0.399 0.092	0.607 0.250	13
5	5.168	$\bar{X}$ S	0.234 0.012	0.324 0.102	0.352 0.106	0.360 0.107	0.421 0.103	7
6	6.731	$\bar{X}$ S	0.235 0.041	0.325 0.099	0.376 0.105	0.396 0.113	0.544 0.155	16
7	8.771	$\bar{X}$ S	0.213 0.017	0.297 0.074	0.332 0.086	0.366 0.095	0.451 0.147	16
8	11.422	$\bar{X}$ S	0.229 0.025	0.327 0.073	0.355 0.089	0.408 0.070	0.648 0.344	10

TABLE 32

RADIUS: 3.5° ANGLE: 180°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.872	$\bar{X}$ S	0.382 0.262	0.582 0.255	0.639 0.225	0.704 0.233	0.966 0.366	13
1	1.133	$\bar{X}$ S	0.288 0.164	0.577 0.206	0.591 0.200	0.636 0.207	0.734 0.290	8
2	1.483	$\bar{X}$ S	0.314 0.116	0.588 0.249	0.634 0.229	0.693 0.242	0.866 0.270	9
3	1.927	$\bar{X}$ S	0.269 0.099	0.508 0.199	0.520 0.209	0.524 0.208	0.620 0.272	10
4	2.512	$\bar{X}$ S	0.237 0.059	0.356 0.152	0.461 0.176	0.523 0.199	0.653 0.271	9
5	3.277	$\bar{X}$ S	0.281 0.103	0.406 0.176	0.409 0.174	0.459 0.151	0.711 0.410	4
6	4.269	$\bar{X}$ S	0.251 0.064	0.454 0.257	0.477 0.249	0.570 0.271	0.671 0.263	13
7	5.563	$\bar{X}$ S	0.228 0.039	0.369 0.163	0.427 0.142	0.434 0.142	0.441 0.141	9
8	7.244	$\bar{X}$ S	0.236 0.036	0.402 0.264	0.412 0.258	0.479 0.240	0.721 0.404	10

TABLE 33

RADIUS: 3.5° ANGLE: 225°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.676	$\bar{X}$ S	0.468 0.144	0.632 0.187	0.662 0.190	0.764 0.207	0.919 0.348	14
1	0.879	$\bar{X}$ S	0.350 0.071	0.480 0.091	0.533 0.109	0.577 0.116	0.874 0.313	10
2	1.150	$\bar{X}$ S	0.322 0.077	0.515 0.190	0.527 0.188	0.591 0.227	0.739 0.235	12
3	1.494	$\bar{X}$ S	0.296 0.110	0.387 0.095	0.412 0.149	0.477 0.238	0.530 0.253	7
4	1.948	$\bar{X}$ S	0.241 0.037	0.376 0.094	0.420 0.090	0.486 0.141	0.766 0.404	11
5	2.541	$\bar{X}$ S	0.262 0.086	0.413 0.180	0.447 0.187	0.451 0.186	0.530 0.240	12
6	3.310	$\bar{X}$ S	0.256 0.057	0.341 0.075	0.382 0.108	0.501 0.148	0.556 0.141	6
7	4.314	$\bar{X}$ S	0.242 0.039	0.296 0.041	0.337 0.069	0.427 0.140	0.580 0.261	14
8	5.618	$\bar{X}$ S	0.230 0.057	0.361 0.114	0.411 0.189	0.457 0.237	0.527 0.264	24

TABLE 34

RADIUS: 3.5° ANGLE: 270°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.528	$\bar{X}$ S	0.484 0.120	0.848 0.303	0.888 0.275	0.988 0.273	1.192 0.311	13
1	0.687	$\bar{X}$ S	0.512 0.103	0.955 0.222	1.041 0.278	1.103 0.320	1.195 0.290	15
2	0.899	$\bar{X}$ S	0.467 0.140	0.712 0.232	0.766 0.291	0.833 0.317	0.971 0.298	9
3	1.168	$\bar{X}$ S	0.338 0.072	0.593 0.326	0.603 0.326	0.656 0.302	0.790 0.339	10
4	1.523	$\bar{X}$ S	0.283 0.047	0.382 0.116	0.452 0.078	0.524 0.151	0.592 0.191	12
5	1.987	$\bar{X}$ S	0.268 0.066	0.397 0.131	0.413 0.146	0.460 0.184	0.595 0.238	12
6	2.588	$\bar{X}$ S	0.250 0.035	0.359 0.093	0.435 0.060	0.447 0.061	0.553 0.129	10
7	3.372	$\bar{X}$ S	0.238 0.019	0.327 0.081	0.407 0.151	0.525 0.224	0.617 0.276	14
8	4.391	$\bar{X}$ S	0.243 0.036	0.339 0.087	0.407 0.134	0.418 0.132	0.488 0.148	11

TABLE 35

RADIUS: 3.5°   ANGLE: 315°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.517	$\bar{X}$ S	0.492 0.119	0.905 0.354	0.979 0.370	1.046 0.365	1.249 0.287	13
1	0.673	$\bar{X}$ S	0.507 0.071	0.909 0.122	0.932 0.159	1.122 0.348	1.227 0.371	5
2	0.880	$\bar{X}$ S	0.336 0.057	0.489 0.102	0.501 0.102	0.618 0.141	0.786 0.355	8
3	1.143	$\bar{X}$ S	0.302 0.060	0.442 0.150	0.480 0.144	0.581 0.233	0.733 0.356	20
4	1.491	$\bar{X}$ S	0.323 0.153	0.458 0.209	0.489 0.204	0.525 0.199	0.777 0.323	13
5	1.945	$\bar{X}$ S	0.246 0.045	0.390 0.075	0.432 0.080	0.493 0.127	0.751 0.339	11
6	2.533	$\bar{X}$ S	0.237 0.028	0.359 0.078	0.385 0.073	0.483 0.131	0.733 0.398	15
7	3.301	$\bar{X}$ S	0.235 0.056	0.367 0.158	0.417 0.224	0.545 0.267	0.693 0.314	13
8	4.299	$\bar{X}$ S	0.219 0.032	0.317 0.089	0.356 0.093	0.390 0.080	0.513 0.162	15

TABLE 36

RADIUS: 4.0° ANGLE: 0°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.869	$\bar{X}$ S	0.437 0.139	0.620 0.281	0.645 0.272	0.696 0.376	0.805 0.448	11
1	1.130	$\bar{X}$ S	0.403 0.114	0.561 0.174	0.629 0.176	0.724 0.238	0.882 0.341	9
2	1.479	$\bar{X}$ S	0.363 0.114	0.761 0.319	0.804 0.314	0.841 0.332	0.907 0.353	9
3	1.921	$\bar{X}$ S	0.278 0.043	0.403 0.108	0.407 0.109	0.435 0.097	0.562 0.181	6
4	2.505	$\bar{X}$ S	0.277 0.078	0.406 0.147	0.448 0.147	0.569 0.144	0.679 0.199	16
5	3.268	$\bar{X}$ S	0.213 0.064	0.376 0.067	0.436 0.098	0.441 0.098	0.575 0.148	10
6	4.257	$\bar{X}$ S	0.244 0.057	0.358 0.063	0.444 0.095	0.470 0.091	0.566 0.099	10
7	5.548	$\bar{X}$ S	0.258 0.057	0.347 0.095	0.360 0.108	0.384 0.106	0.610 0.274	14
8	7.224	$\bar{X}$ S	0.246 0.039	0.341 0.074	0.376 0.125	0.426 0.116	0.483 0.143	18

TABLE 37

RADIUS: 4.0°   ANGLE: 45°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.290	$\bar{X}$ S	0.311 0.100	0.455 0.117	0.481 0.115	0.529 0.136	0.764 0.396	14
1	1.677	$\bar{X}$ S	0.279 0.070	0.456 0.240	0.468 0.235	0.558 0.231	0.684 0.293	17
2	2.194	$\bar{X}$ S	0.268 0.051	0.374 0.113	0.418 0.123	0.517 0.172	0.567 0.181	9
3	2.851	$\bar{X}$ S	0.258 0.075	0.369 0.138	0.373 0.138	0.449 0.139	0.496 0.103	8
4	3.717	$\bar{X}$ S	0.245 0.024	0.382 0.081	0.425 0.098	0.451 0.097	0.624 0.294	11
5	4.849	$\bar{X}$ S	0.223 0.078	0.390 0.105	0.397 0.108	0.440 0.102	0.576 0.177	5
6	6.317	$\bar{X}$ S	0.245 0.040	0.375 0.094	0.422 0.106	0.498 0.106	0.622 0.288	10
7	8.231	$\bar{X}$ S	0.230 0.029	0.316 0.070	0.343 0.075	0.395 0.099	0.554 0.237	13
8	10.719	$\bar{X}$ S	0.243 0.081	0.338 0.151	0.375 0.148	0.450 0.113	0.523 0.170	10

TABLE 38

RADIUS: 4.0° ANGLE: 90°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.263	$\bar{X}$ S	0.272 0.060	0.481 0.076	0.495 0.082	0.535 0.090	0.620 0.150	18
1	1.642	$\bar{X}$ S	0.294 0.082	0.479 0.068	0.516 0.123	0.589 0.157	0.724 0.276	10
2	2.148	$\bar{X}$ S	0.259 0.077	0.444 0.103	0.496 0.111	0.576 0.152	0.791 0.369	15
3	2.791	$\bar{X}$ S	0.228 0.021	0.425 0.055	0.432 0.059	0.484 0.072	0.579 0.247	9
4	3.639	$\bar{X}$ S	0.222 0.038	0.426 0.084	0.457 0.118	0.496 0.156	0.611 0.155	8
5	4.748	$\bar{X}$ S	0.216 0.034	0.362 0.076	0.390 0.088	0.455 0.116	0.553 0.106	7
6	6.184	$\bar{X}$ S	0.216 0.088	0.403 0.158	0.427 0.152	0.490 0.196	0.610 0.244	21
7	8.059	$\bar{X}$ S	0.242 0.044	0.419 0.076	0.467 0.143	0.490 0.168	0.721 0.376	12
8	10.494	$\bar{X}$ S	0.220 0.030	0.395 0.188	0.412 0.218	0.487 0.263	0.596 0.373	13

TABLE 39

RADIUS: 4.0°   ANGLE: 135°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	1.469	$\bar{X}$ S	0.282 0.075	0.436 0.095	0.457 0.096	0.509 0.085	0.542 0.082	23
1	1.909	$\bar{X}$ S	0.250 0.046	0.445 0.065	0.473 0.102	0.492 0.096	0.556 0.142	15
2	2.498	$\bar{X}$ S	0.273 0.113	0.455 0.149	0.487 0.153	0.497 0.148	0.605 0.215	15
3	3.246	$\bar{X}$ S	0.242 0.016	0.374 0.073	0.404 0.067	0.460 0.087	0.625 0.243	11
4	4.232	$\bar{X}$ S	0.214 0.029	0.319 0.088	0.373 0.091	0.443 0.055	0.655 0.349	14
5	5.521	$\bar{X}$ S	0.222 0.029	0.376 0.100	0.413 0.135	0.461 0.148	0.631 0.245	9
6	7.192	$\bar{X}$ S	0.212 0.022	0.346 0.075	0.362 0.067	0.469 0.140	0.569 0.266	14
7	9.372	$\bar{X}$ S	0.216 0.027	0.363 0.103	0.385 0.094	0.397 0.092	0.481 0.050	6
8	12.204	$\bar{X}$ S	0.230 0.050	0.327 0.103	0.382 0.093	0.417 0.085	0.421 0.085	6

TABLE 40

RADIUS: 4.0°   ANGLE: 180°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.711	$\bar{X}$ S	0.387 0.121	0.804 0.357	0.832 0.371	0.844 0.380	0.969 0.396	7
1	0.924	$\bar{X}$ S	0.427 0.162	0.791 0.142	0.794 0.139	0.837 0.169	0.996 0.302	8
2	1.209	$\bar{X}$ S	0.402 0.190	0.721 0.302	0.726 0.300	0.733 0.295	0.739 0.293	6
3	1.571	$\bar{X}$ S	0.270 0.091	0.491 0.168	0.513 0.149	0.520 0.146	0.695 0.296	11
4	2.049	$\bar{X}$ S	0.329 0.177	0.628 0.250	0.665 0.265	0.681 0.271	0.685 0.272	5
5	2.673	$\bar{X}$ S	0.224 0.032	0.572 0.271	0.574 0.271	0.707 0.320	0.859 0.265	6
6	3.481	$\bar{X}$ S	0.298 0.106	0.556 0.326	0.558 0.328	0.594 0.362	0.603 0.355	8
7	4.536	$\bar{X}$ S	0.265 0.166	0.364 0.259	0.391 0.251	0.452 0.267	0.646 0.424	7
8	5.907	$\bar{X}$ S	0.249 0.050	0.575 0.322	0.611 0.295	0.686 0.309	0.875 0.483	11

TABLE 41

RADIUS: 4.0°　　ANGLE: 225°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.547	$\bar{X}$ S	0.493 0.185	0.957 0.286	0.960 0.286	0.975 0.278	1.036 0.271	11
1	0.711	$\bar{X}$ S	0.461 0.123	0.667 0.207	0.721 0.177	0.855 0.241	0.972 0.290	10
2	0.930	$\bar{X}$ S	0.345 0.096	0.500 0.138	0.541 0.124	0.604 0.184	0.725 0.272	15
3	1.209	$\bar{X}$ S	0.285 0.074	0.472 0.054	0.500 0.098	0.531 0.104	0.616 0.179	9
4	1.576	$\bar{X}$ S	0.252 0.033	0.389 0.073	0.399 0.070	0.482 0.121	0.637 0.267	16
5	2.056	$\bar{X}$ S	0.227 0.021	0.383 0.059	0.389 0.060	0.401 0.056	0.479 0.186	9
6	2.679	$\bar{X}$ S	0.252 0.088	0.377 0.127	0.435 0.137	0.519 0.152	0.550 0.172	15
7	3.490	$\bar{X}$ S	0.225 0.026	0.367 0.074	0.389 0.112	0.428 0.153	0.546 0.222	9
8	4.545	$\bar{X}$ S	0.242 0.042	0.354 0.061	0.404 0.057	0.446 0.078	0.641 0.189	8

TABLE 42

RADIUS: 4.0°   ANGLE: 270°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.412	$\bar{X}$ S	0.489 0.107	0.880 0.271	0.982 0.225	1.034 0.268	1.207 0.270	10
1	0.536	$\bar{X}$ S	0.570 0.120	1.233 0.281	1.276 0.334	1.285 0.326	1.423 0.347	11
2	0.701	$\bar{X}$ S	0.465 0.101	0.846 0.303	0.905 0.324	0.988 0.358	1.110 0.334	14
3	0.911	$\bar{X}$ S	0.422 0.118	0.757 0.278	0.795 0.305	0.902 0.294	1.007 0.302	14
4	1.187	$\bar{X}$ S	0.368 0.079	0.631 0.279	0.658 0.282	0.729 0.257	0.825 0.298	13
5	1.549	$\bar{X}$ S	0.270 0.062	0.464 0.086	0.552 0.109	0.575 0.094	0.752 0.135	6
6	2.018	$\bar{X}$ S	0.265 0.086	0.416 0.148	0.433 0.140	0.455 0.140	0.588 0.286	15
7	2.629	$\bar{X}$ S	0.285 0.175	0.451 0.342	0.516 0.335	0.538 0.336	0.685 0.377	12
8	3.424	$\bar{X}$ S	0.268 0.076	0.430 0.119	0.435 0.120	0.497 0.108	0.716 0.266	10

TABLE 43

RADIUS: 4.0°   ANGLE: 315°

F	RVL		LTCY (SEC)	ST25 (SEC)	ST20 (SEC)	ST15 (SEC)	ST10 (SEC)	N
0	0.414	$\bar{X}$ S	0.516 0.112	1.035 0.305	1.073 0.267	1.191 0.394	1.266 0.452	9
1	0.538	$\bar{X}$ S	0.527 0.119	1.097 0.273	1.145 0.296	1.199 0.276	1.311 0.319	14
2	0.704	$\bar{X}$ S	0.442 0.059	0.675 0.152	0.677 0.152	0.751 0.127	0.874 0.341	4
3	0.915	$\bar{X}$ S	0.391 0.165	0.561 0.169	0.584 0.153	0.607 0.158	0.741 0.300	6
4	1.194	$\bar{X}$ S	0.337 0.076	0.531 0.210	0.575 0.194	0.633 0.182	0.686 0.215	11
5	1.557	$\bar{X}$ S	0.377 0.111	0.619 0.197	0.664 0.179	0.718 0.150	0.847 0.197	8
6	2.028	$\bar{X}$ S	0.263 0.070	0.442 0.150	0.455 0.142	0.540 0.205	0.700 0.221	19
7	2.643	$\bar{X}$ S	0.260 0.043	0.428 0.193	0.474 0.187	0.508 0.186	0.696 0.247	13
8	3.442	$\bar{X}$ S	0.230 0.024	0.352 0.068	0.398 0.076	0.461 0.125	0.734 0.327	10

LIST OF REFERENCES

References for eye movements:

Bahill T.A. and Stark L. (1977). Oblique Saccadic Eye Movements: Independence of Horizontal and Vertical Channels. Arch. Ophthalmol. 95, 1258-1261.

Boyce P. R. (1967). The Effect of Change of Target Field Luminance and Colour on Fixational Eye Movements. Optica Acta 14, 213-217.

Brown B. (1972a). Dynamic Visual Acuity, Eye Movements and Peripheral Acuity for Moving Targets. Vision Res. 12, 305-321

Brown B. (1972b). The Effect of Target Contrast Variation on Dynamic Visual Acuity and Eye Movements. Vision Res. 12, 1213-1224.

Carlow T., Dell'Osso L. F., Troost B. T., Daroff R. B., and Birkett J. E. (1975). Saccadic Eye Movement Latencies to Multimodal Stimuli. Vision Res. 15, 1257-1262.

Ditchburn R. W. (1973). Eye Movements and Visual Perception. Clarendon Press, Oxford.

Dodge R. and Cline T. S. (1901). The Angle Velocity of Eye Movements. Psychological Rev. 8, 145-157.

Fricker S. J. (1971). Dynamic Measurements of Horizontal Eye Movements. Invest. Ophthal. 10, 724-732.

Haegerstrom-Portnoy G., and Brown B. (1979). Contrast Effects on Smooth-Pursuit Eye Movement Velocity. Vision Res. 19, 169-174.

Robinson D. A. (1964). The Mechanics of Human Saccadic Eye Movement. J. Physiol. 174, 245-264.

Robinson D. A. (1968). The Oculomotor Control System: A Review. Proc. IEEE 56, 1032-1049.

Saslow M. G. (1966a). Effects of Components of Displacement-Step Stimuli Upon Latency for Saccadic Eye Movement. J. Opt. Soc. Am. 57, 1027-1029.

Saslow M. G. (1966b). Latency for Saccadic Eye Movement. J. Opt. Soc. Am. 57, 1030-1033.

Steinman R. M. (1965). Effect of Target Size, Luminance, and Color on Monocular Fixation. J. Opt. Soc. Am. 55, 1158-1165.

Taumer R., Lemb M., and Namisle M. (1976). Characteristics of Human Saccadic Eye Movements in Different Directions. Albrecht v. Graefes Arch. klin. exp. Ophthal. 200, 163-174.

Westheimer G. (1954). Mechanisms of Saccadic Eye Movements. A.M.A. Arch. Ophthal. 52, 710-724.

Wheeles L. L., Cohen G. H., and Boynton R. M. (1967). Luminance as a Parameter of the Eye-Movement Control System. J. Opt. Soc. Am. 57, 394-400.

White C. T. and Eason R. G. (1962). Latency and Duration of Eye Movements in the Horizontal Plane. J. Opt. Soc. Am. 52, 210-213.

Yarbus A. L. (1967). Eye Movements and Vision. Translation by Haigh B. and Riggs L.A.. Plenum Press, New York.

References for the computer system:

PDP8/E & PDP8/M Small Computer Handbook, 1972. Published by the Digital Equipment Corporation (D.E.C.), Maynard, Mass. 01754.

Logic Handbook 1973-1974. D.E.C., 1973.

Introduction to Programming. D.E.C., 1972, 1975

Programming Languages. D.E.C., 1972.

PDP8/E, PDP8/F & PDP8/M Maintenance Manuals, Vols. 1, 2, 3. D.E.C., 1972.

LAB-8/E Maintenance Manual. D.E.C., 1972.

OS/8 Handbook. D.E.C., 1974.

Program Description and Operating Instructions - OS/8 Handler for Sykes Disk 7000 System. Published by Sykes Datatronics, Inc., Rochester, N.Y. 14606, 1974.

User's Manual for Sykes Disk 7000 System with PDP-8 Family of Computers. Sykes Datatronics, 1974.

130 Dataplotter Maintenance Manual. Published by Electronic Associates, Inc., West Long Beach, N. J., 1971.

Technical Manual 33 Teletypewriter Sets, Vols. 1 and 2. Published by the Teletype Corporation, Skokie, Illinois, 1971.

References for Cornsweet-Crane eyetracker:

Cornsweet T. N. and Crane H. D. (1967). Double-Purkinje-Image Eyetracker. Quarterly report 3 prepared for the National Aeronautics and Space Administration, Ames Research Center, Moffett Field, California 94035. SRI project 6009. NASA contract NAS2-3517.

Cornsweet T. N. and Crane H. D. (1973). Accurate Two-Dimensional Eye Tracker Using First and Fourth Purkinje Images. J. Opt. Soc. Am. 63, 921-928.

Crane, H. D. and Steele, C. M. (1978). Accurate Three-Dimensional Eyetracker. Applied Optics 17, 691-705.

User's Manual, Double Purkinje Eyetracker. Prepared by Stanford Research International, Palo Alto, California and supplied with the eyetracker.

Reference for earlier work:

Brown, J. R. (1977). An Automated Saccade Analysis System. Ph.D. dissertation presented to The Ohio State University.