GEORGIA INSTITUTE OF TECHNOLOGY
OFFICE OF CONTRACT ADMINISTRATION
SPONSORED PROJECT INITIATION

Date: April 25, 1979

Gr Cd
AF

Project Title: RADC Post-Doctoral Program (C-E Systems)

Project No: A-2366 (Sub-project under E-21-626/Paris/EE)-- under task assignments being directed by Dr. W. M. Leach/EE)

Project Director: Mr. Hugh W. Denny

Sponsor: USAF, AFSC, Rome Air Development Center, Griffiss AFB, NY 13441

Agreement Period: From 4/1/79 Until 9/30/79
(Task Performance Period)

Type Agreement: Contract No. F30602-78-C-0120

Amount:
$ 48,575 RADC (A-2366)
5,397 GIT (E-222-005)
$ 53,972 TOTAL

Reports Required: See E-21-626 (Dr. W. M. Leach/EE)

Sponsor Contact Person (s):

Technical Matters

Contractual Matters
(thru OCA)

Dr. W. M. Leach
Associate Professor
School of Electrical Engineering
Campus
Extension 2963

Dr. D. T. Paris
Director
School of Electrical Engineering
Campus
Extension 2902

Defense Priority Rating: DO-A7 under DMS Reg. 1

Assigned to: ETL/EMCG (School/Laboratory)

COPIES TO:

Project Director
Division Chief (EES)
School/Laboratory Director
Dean/Director—EES
Accounting Office
Procurement Office
Security Coordinator (OCA)
✓ Reports Coordinator (OCA)

Library, Technical Reports Section
EES Information Office
EES Reports & Procedures
Project File (OCA)
Project Code (GTRI)
Other Dr. D. T. Paris/EE
Dr. W. M. Leach/EE

CA-3 (3/76)

GEORGIA INSTITUTE OF TECHNOLOGY
OFFICE OF CONTRACT ADMINISTRATION

SPONSORED PROJECT TERMINATION

Date: February 25, 1980

Project Title: RADC Post-Doctoral Program (C-E Systems)

Project No: A-2366

Project Director: Hugh W. Denny

Sponsor: USAF, AFSC, Rome Air Development Center, Griffiss AFB, NY 13441

Effective Termination Date: October 31, 1979

Clearance of Accounting Charges: ---

Grant/Contract Closeout Actions Remaining:

NONE

- _ Final Invoice and Closing Documents
- _ Final Fiscal Report
- _ Final Report of Inventions
- _ Govt. Property Inventory & Related Certificate
- _ Classified Material Certificate
- _ Other ______________________

TERMINATED

Assigned to: ETL/EC (School/Laboratory)

COPIES TO:

Project Director
Division Chief (EES)
School/Laboratory Director
Dean/Director—EES
Accounting Office
Procurement Office
Security Coordinator (OCA)
Reports Coordinator (OCA) ✓

Library, Technical Reports Section
EES Information Office
Project File (OCA)
Project Code (GTRI)
Other ______________________

CA—4 (1/79)

A-2366

FINAL REPORT
PROJECT NO. A-2366

ELECTRO-OPTICAL TARGET SIMULATOR REQUIREMENTS ANALYSIS

By

David E. Schmieder, A. Craig Kenton, and Gene R. Loefer

Prepared for

ROME AIR DEVELOPMENT CENTER
GRIFFISS AIR FORCE BASE
Rome, New York

September 1979

GEORGIA INSTITUTE OF TECHNOLOGY
Engineering Experiment Station
Atlanta, Georgia 30332

1979

FINAL TECHNICAL REPORT
Project No. A-2366

ELECTRO-OPTICAL TARGET SIMULATOR REQUIREMENTS ANALYSIS

David E. Schmieder
A. Craig Kenton
Gene R. Loefer

Contract No. F30602-78-C-0120

Prepared for

Rome Air Development Center
Griffiss Air Force Base
Rome, New York

September 1979

GEORGIA INSTITUTE OF TECHNOLOGY
Engineering Experiment Station
Atlanta, Georgia 30332

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (*and Subtitle*) ELECTRO-OPTICAL TARGET SIMULATOR REQUIREMENTS ANALYSIS		5. TYPE OF REPORT & PERIOD COVERED Final Technical Report May 1979 - October 1979
		6. PERFORMING ORG. REPORT NUMBER A-2366-001
7. AUTHOR(*s*) David E. Schmieder A. Craig Kenton Gene R. Loefer		8. CONTRACT OR GRANT NUMBER(*s*) F30602-78-C-0120
9. PERFORMING ORGANIZATION NAME AND ADDRESS GEORGIA INSTITUTE OF TECHNOLOGY Engineering Experiment Station Atlanta, Georgia 30332		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center, RBCA Building No. 3 Griffiss AFB, N.Y. 13441		12. REPORT DATE October 1979
		13. NUMBER OF PAGES 76
14. MONITORING AGENCY NAME & ADDRESS(*if different from Controlling Office*)		15. SECURITY CLASS. (*of this report*) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (*of this Report*)

17. DISTRIBUTION STATEMENT (*of the abstract entered in Block 20, if different from Report*)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*)

Target Source
Electro-Optical Simulation
Missile Seeker Testing
Electromagnetic Compatibility
Moving Target Simulation

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*)

Target source simulation requirements are examined for electromagnetic compatibility testing of various IR, E-O, and laser guided missiles. Missile seeker characteristics are analyzed. These characteristics are used to generate a baseline set of requirements for a target simulator. Various simulator configurations are studied and cost driving features are identified. Recommendations for breadboard testing are presented.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

PREFACE

This report was prepared by the Electromagnetics Laboratory of the Engineering Experiment Station, Georgia Institute of Technology. The effort was administered under the Post Doctoral Program directed by Jacob Scherer, Rome Air Development Center. Technical monitors were Charles M. Blank, John H. Edwards, Jr., and Jerard J. Genello.

The overall Post Doctoral Program principal investigator at Georgia Tech is Demetrius T. Paris. The lead investigator was W. Marshall Leach, Jr. Engineering Experiment Station participation was administered by Hugh W. Denny. The task Project Director was Glenn E. Riley of the Electro-Optics Division.

TABLE OF CONTENTS

1.0 INTRODUCTION

1.1 Objectives

An analysis study was performed to generate the requirements for a target source simulator. This target simulator is to be used to test infrared, laser, and electro-optical guided missiles in the Rome Air Development Center (RADC) anechoic test chamber. In this test a missile will be subjected to RF radiation to determine its electromagnetic interference (EMI) susceptibility. In particular, the tests will attempt to determine the conditions which cause loss of missile control as a function of target parameters. In these tests the simulator will provide a target upon which to lock the missile seeker and control gimbal pointing angle and track rate. The objective of this study was to determine the target source requirements.

In order to derive the target simulator requirements it is necessary to define the simulator's operational objectives. These are summarized in Table 1-1 and represent a priori objectives upon which this study was based. In general the operational objectives specify the target parameters which must be varied to determine missile susceptibility to EMI.

1.2 Approach

The analysis approach consists of a three part effort to:

1) define the relevant seeker parameters
2) derive simulator design requirements
3) configure a strawman design.

Seeker parameters are important design drivers. They effectively describe the systems to be tested and constrain every aspect of the simulator design. To keep this report unclassified only summary characteristics typical of the various seeker classes are discussed. Next, simulator design requirements are determined from an examination of the limitations that the various seeker parameters, operational requirements, and chamber constraints place on the requirement in question. Finally a strawman

Table 1-1. Simulator operational objectives

1) SIMULATE TARGET CHARACTERISTICS:
    A) Radiant
    B) Temporal
    C) Spatial
    D) Dynamic
2) PROVIDE ADJUSTABLE TARGET LOCK-ON CAPABILITY
    A) Signal Strength
    B) Target Rate
    C) Target Size
3) ALLOW EXPERIMENTAL CONTROL
    A) Remotely Vary Signal by Known Amounts
    B) Remotely Change Rate by Controlled Amounts
    C) Provide Position Feedback

design is postulated to show what impact the various requirements have on a representative simulator configuration.

Thus the approach to requirements generation in this report calls for an examination of both what is desired and what is reasonable. The requirements analysis first looks at what is desired by examining seeker parameters and operational usage requirements. The strawman design effort then places further limits by identifying those desired requirements which result in an unreasonable or impractical configuration. The final set of design requirements is thus a balance between desire and practicality.

Though the above three part effort constitutes the basic approach in this report, other efforts are required to add perspective, identify options, and make trades and recommendations. Once the desired requirements have been generated, practical simulator design configurations are examined and trade-off considerations are identified. The costs of the strawman design are estimated. Cost reduction options are identified and appraised. Finally, a list of preliminary development specifications is proposed and a breadboard configuration for evaluating the specifications is recommended.

2.0 DESIGN REQUIREMENTS

2.1 Seeker Parameters

Missile seeker design parameters are the key to the establishment of target simulator requirements. These parameters effectively characterize the limitations imposed on the simulator if that simulator is to provide an acceptable target. In addition to these parameters, a thorough description of each missile seeker's operating characteristics is desirable if the simulator is to operate efficiently with all candidate seekers. Such a thorough description is not included in this study due to the inability to obtain all the necessary seeker descriptive data in a timely period and due to the desire to keep this report unclassified.

Nevertheless, the various parameters at issue are typically confined to a limited range of values for each seeker type. Table 2-1 shows a representative range of such values. The simulator should be designed to be compatible with these values.

2.2 Chamber Constraints

The anechoic test chamber will impose certain physical constraints on the simulator design. A sketch of the chamber is shown in Figure 2-1 with an example simulator design. The important design drivers are the distance from the test platform to the missile pedestal, the height of the missile pedestal, and the height of the test platform. These distances determine the angular adjustment range requirements of the simulator and constrain its configuration to one which is compatible with the platform.

2.3 Requirements Analysis

In this section specific target simulator design requirements are provided together with their supporting rationale. These requirements are generated here with little consideration of their impact on the hardware design or overall configuration. The latter considerations are provided later in this report. Table 2-2 summarizes the preliminary requirements that result from this approach. Also shown is the driving consideration, or rationale, from which the requirement is generated. The following subsections describe these design requirements and rationale in more detail.

2.3.1 Minimum target angular size

The minimum target size need be no less than approximately 1.0 mr. Most reticle and laser type seekers resolve to no better than this resolution or at least function adequately with a one milliradian target subtense. While IR or E-O imaging seekers may resolve to much better than one milliradian, they typically require several resolution elements across the target to meet their lock-on criteria. Thus the 1.0 mr minimum

Table 2-1. Expected seeker parameters

Seeker Type	Track Rate (deg/s)	Aper. (in)	NEI (W/cm^2)	Resolution (mr)	IFOV (deg.)	Waveband (μm)	FOV (deg.)
E-O (TV)	30	2-3	10^{-6}	0.1	0.1-0.3	0.4-0.7	2-5
Imaging (IR)	11	2-3	$10^{-12}-10^{-11}$	0.5	0.1-0.5	3.5-5.5 8-13	2-3
Reticle (IR)	20	2-3	$10^{-10}-10^{-11}$	1.0	1.5-7	1.5-3.0 3.5-5.5	3-7
Laser	30	2-3	$10^{-6}-10^{-4}$	1.0	3-5	1.06	3-5

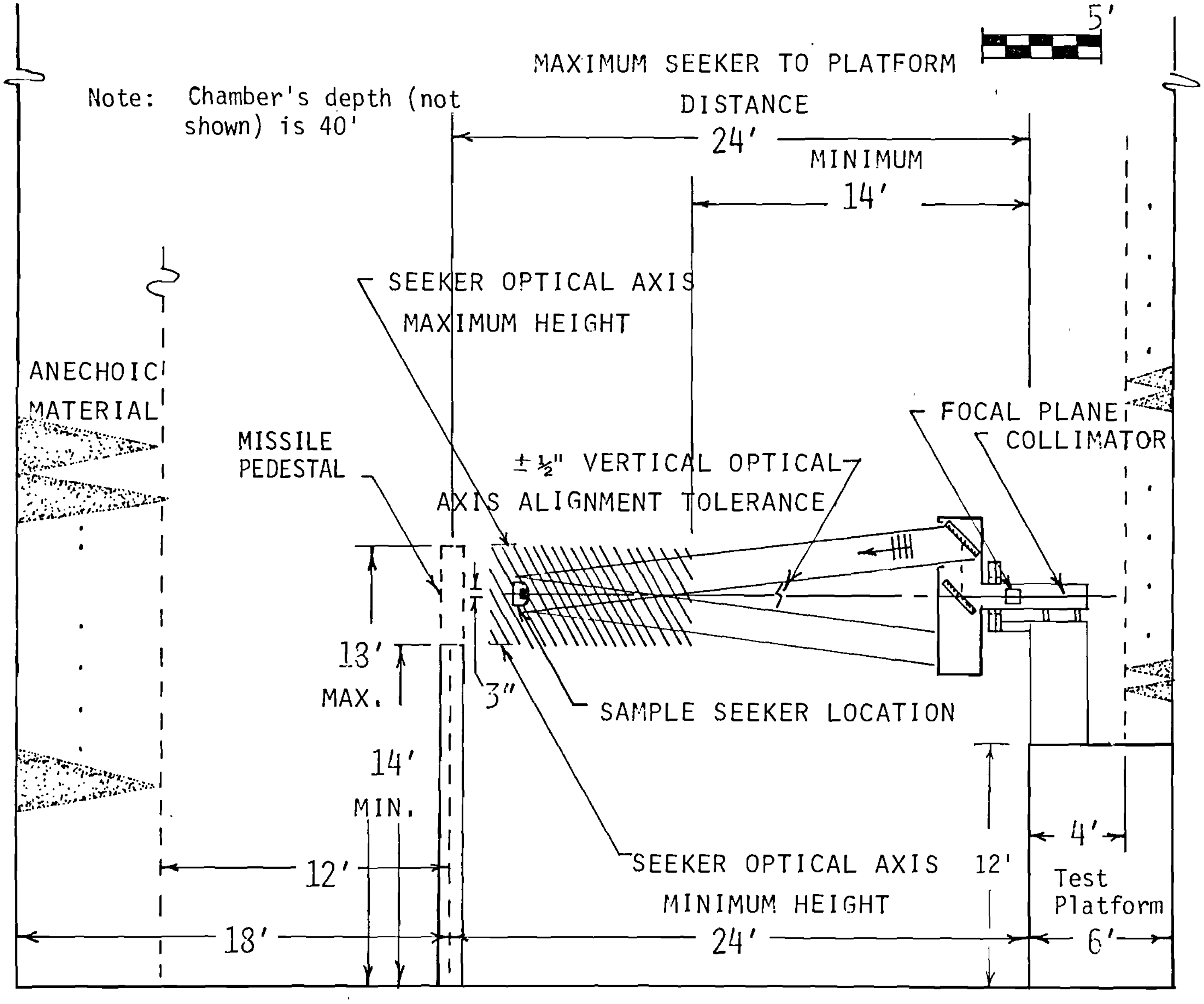


Figure 2-1. RADC anechoic chamber dimensions

Table 2-2 Preliminary design requirements

PARAMETER	REQUIREMENT	SOURCE DRIVER
min. target angular size	1.0 mr	Seeker resolution & gate size
max. target angular size	1 deg.	30°/s track rate & gate size
unvignetted aperture at seeker (spot size)	4 in required 8 in desired	3 in max. aperture, ± 1/2 position tolerance (growth to 7 in aperture desired)
min. irradiance at seeker		expected seeker NEI
1.5-3.0 μm	10^{-11}W/cm^2	--
3.5-5.5 μm	4×10^{-11}	--
8-13.5 μm	10^{-12}	--
0.6-0.9 μm	6×10^{-8}	50 fL
0.4-0.7 μm	6×10^{-9}	50 fL
1.06 μm	1.5×10^{-6}	--
Source-seeker dist.	14' - 24'	chamber/missile dimen.
max. angular motion (apparent at seeker)	± 2	discriminator noise level
max. target angular rate	30 /s	expected max. seeker rate
min. target angular rate	1.2 mr/s	noise equivalent angle & 1.5 sec. time const.
target angular accel.	60°/s^2	expected max. seeker rate
source ht. relative to seeker	6' to 2' (below)	16' ± 2' pedestal ht.
source motion type	circular	RADC
target rate variability	$\leq$ 0.6 mr/s	noise equivalent angle and time constant
target position	$\leq$ 0.8 mr	gimbal pick-off

Table 2-2 (Continued)

PARAMETER	REQUIREMENT	SOURCE DRIVER
target jitter	$\leq$ 0.8 mr	gimbal pick-off
collimation accuracy/ resolution	$\leq$ 1.0 mr $\leq$ 0.2 mr (desired)	seeker resolution/ min. expected target size
laser source char.		
prf	10-20 pps	seeker requirements
pulse width	10-25 ns	seeker requirements
center frequency	1.063 μm	seeker requirements
$\Delta\lambda$, 1/2 power	360 Å	seeker requirements
max. irrad.	1 W/cm^2	NEI, 10^4 dyn. rge.
spot shape	round	
intensity adj. rge.	Nd 2, continuous	
E-O source char.		
contrast range	± (20% - 100%)*	seeker requirements
background size	2° required 9° desired	2 x 1° tgt. size max. FOV & ± 2 motion
max. irrad		
0.6-0.9 μm	8 x 10^{-5} W/cm^2	sun @ zenith, 1 km range
0.4-0.7 μm	8 x 10^{-5} W/cm^2	from 1° target size
shape	round	track symmetry
uniformity	± 10%	track symmetry
intensity adj. range	Nd 2*, continuous	expected test range
IR source char.		
max. irrad.		
1.5-3.0 μm	10^{-7} W/cm^2	seeker dyn. rge. (10^3) and NEI
3.55-5.5 μm	4 X 10^{-7}	" "
8.0-13.5 μm	10^{-9}	
backgrd. size	2° min. req'd 11° desired	1° target size reticle seeker FOV & ± 2° amplitude
intensity adj. range	Nd 2**, continuous	expected test range
Controls		
intensity	remote	test flexibility
angle rate	remote	" "
source type	removable	" "

* variability from 2% to 100% desired, contrast = $\frac{\text{(target rad.-bkgd rad.)}}{\text{bkgd rad.}}$

** without color temperature change

Table 2-2 (Continued)

PARAMETER	REQUIREMENT	SOURCE DRIVER
alignment		
spot ht.	manually adj.	usage
spot dist.	manually adj.	"
set-up time	$\leq$ 15 min.	"
transportability	portable collimated source	usage
status display		
position	voltage vs time	usage
rate	voltage vs time	usage
intensity	voltage vs Nd	usage

target size is a reasonable minimum target size requirement with which to test all seeker types.

2.3.2 Maximum Target Angular Size

The maximum target size to be provided by the simulator can, in some cases, be estimated from expected track rate constraints. Figure 2-2 shows the effect of target size on the maximum seeker track rate that can be achieved with imaging and reticle generic seeker types; laser seekers generally achieve lower rates. Reticle trackers are typically limited by the reticle spin speed to approximately 100°/s regardless of target size. However, in reticle seekers the gimbal drive rate limits the maximum available track rate to only 20°/s.

Imaging seekers, however, achieve a track rate which depends upon target size. This is the case, at least, for nonpredictive (Type 0) servo loops in which the gimbal pointing angle is only a function of target position and not target rate or acceleration. The target size (Θ_{max}) relationship to track rate ($\dot{\phi}$) can be expressed as:

$$\theta_{max}/2 = \dot{\phi}\ f_t \qquad (2\text{-}1)$$

This relationship is merely a statement of the fact that the pull-in range of an imaging seeker adaptive gate centroid tracker is limited to half the gate (target) size in a field period (f_t). Thus, the plot of maximum imaging seeker track rate, in Figure 2-2, results from Equation (2-1). For the maximum, 30°/s, imaging (E-O) seeker track rate given in Table 2-1, it is seen that the maximum target size need be no greater than 1° for a frame time of 1/60 s.

The above conclusion is less valid in the presence of low signal to noise ratios. This can be seen from Figures 2-3 and 2-4. Figure 2-3 shows the representative track rate that can be expected of a reticle seeker as a function of NEI (noise equivalent irradiance) multiple. While a lower NEI multiple requires lower track rates, target size plays no role. Again, this is not the case for an imaging seeker. Figure 2-4 shows the track rate that can be achieved with an imaging (EO or IR) seeker as a function of signal to noise ratio in the video (SNRV) for various

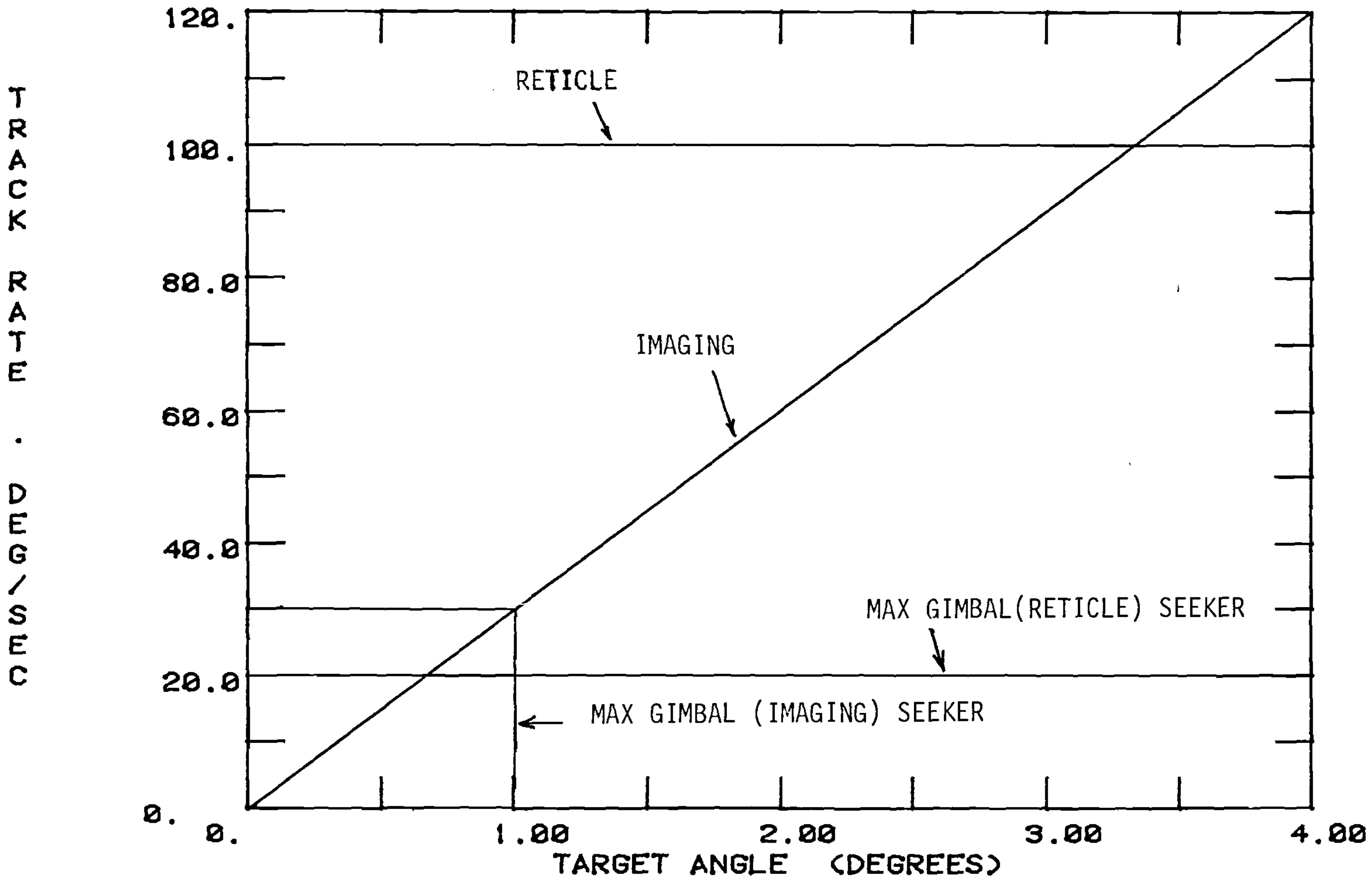


Figure 2-2. Maximum seeker track rates.

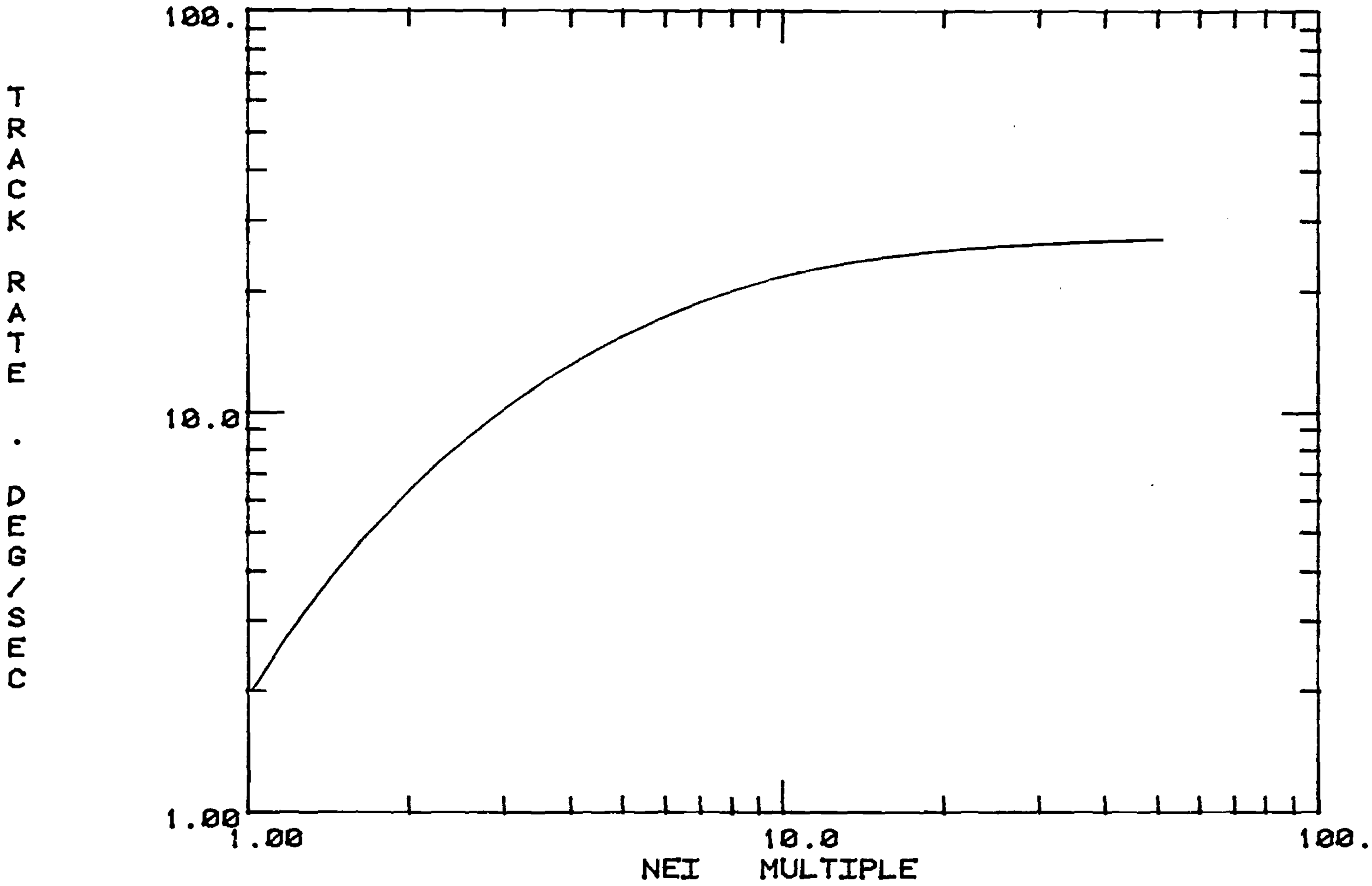


Figure 2-3. Reticle seeker track rate limit in noise.

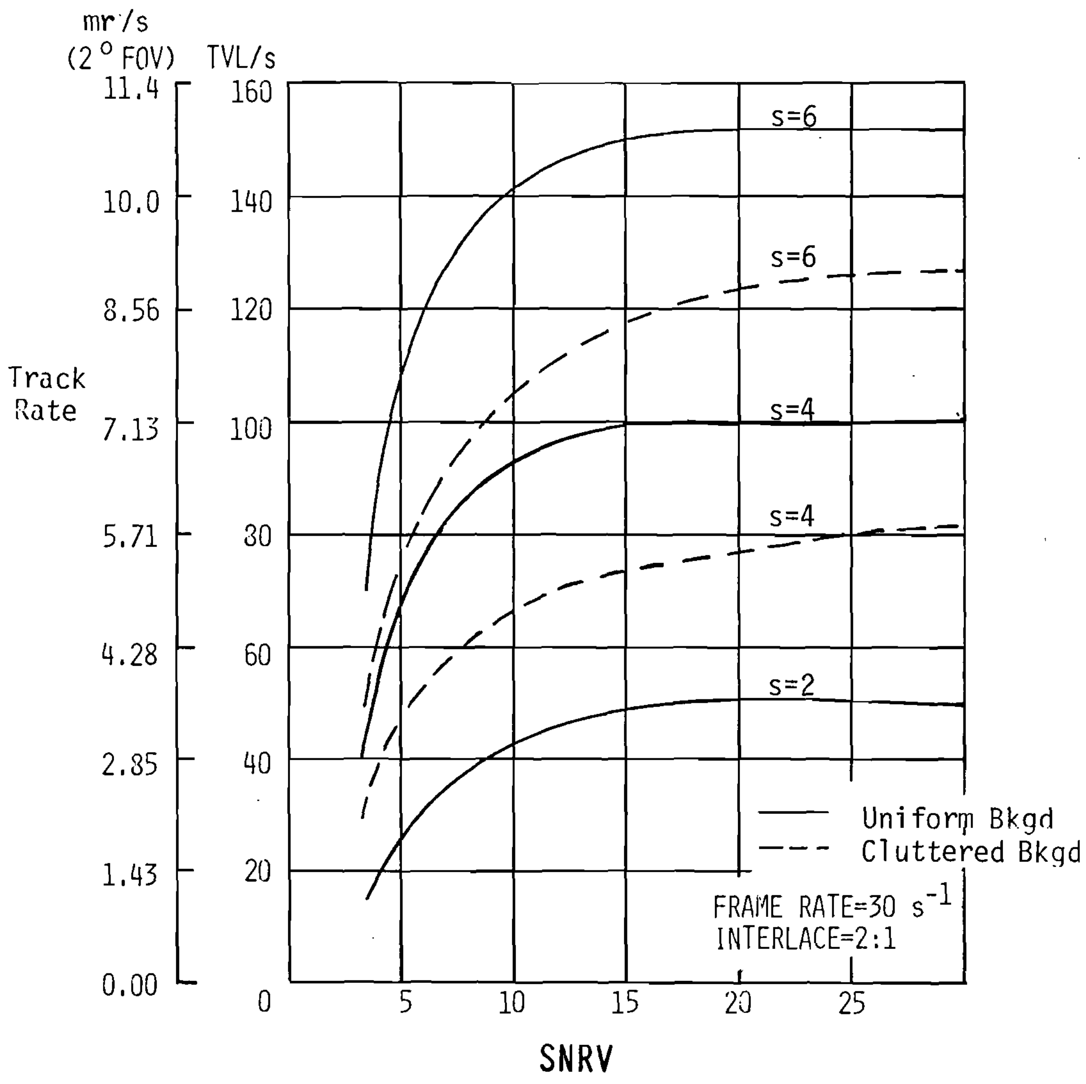


Figure 2-4. Imaging seeker track rate limit in noise

scan lines (S) across the target. It is seen that for the larger targets, i.e., larger S, higher rates can be tolerated for a given SNRV. Thus an even larger target than the previously specified 1° target would permit higher track rates under degraded SNRV levels. Nevertheless a 1° target size represents a large fraction of the typical seeker's field of view and is approaching the seeker blind range size criteria - the point where seeker controls are locked pending impact. This fact, combined with the realization that the typical test will simulate the lower rates encountered in mid-course guidance, indicates that no benefit could be achieved with the capability to simulate targets larger than 1°.

2.3.3 Unvignetted aperture size

The effects of target source vignetting are illustrated in Figure 2-5. Vignetting occurs when rays originating from points on the target extremities do not evenly illuminate the seeker aperture. This would normally cause the edges of an extended target to appear as shown in the center intensity plot in the figure. However, since the seeker aperture must translate in order to track the moving target, the intensity profile will distort as shown in the left and right intensity plots. This distortion and asymmetry of the target intensity profile will reduce the effective size of the target and could also produce an apparent tracking "error." The error might be mistaken for EMI effects during susceptibility testing. To avoid such effects it is necessary to require that the simulator be capable of unvignetted illumination of a minimum spot size at the seeker aperture. This spot size should be large enough to include the gimbal aperture, a missile placement tolerance, and gimbal motion. Here it is assumed that the minimum desired spot size is 4 inches. This will accommodate the largest (3 inch) aperture in Table 2-1 and still allow ± ½ in. for alignment. Gimbal motion is expected to be no more than a few degrees and will, therefore, not contribute significantly to the spot size requirement. Of course an even larger unvignetted spot size would be required if larger seeker apertures were expected. For this reason a spot diameter of up to 8 inches would be a desired, although not required, simulator feature.

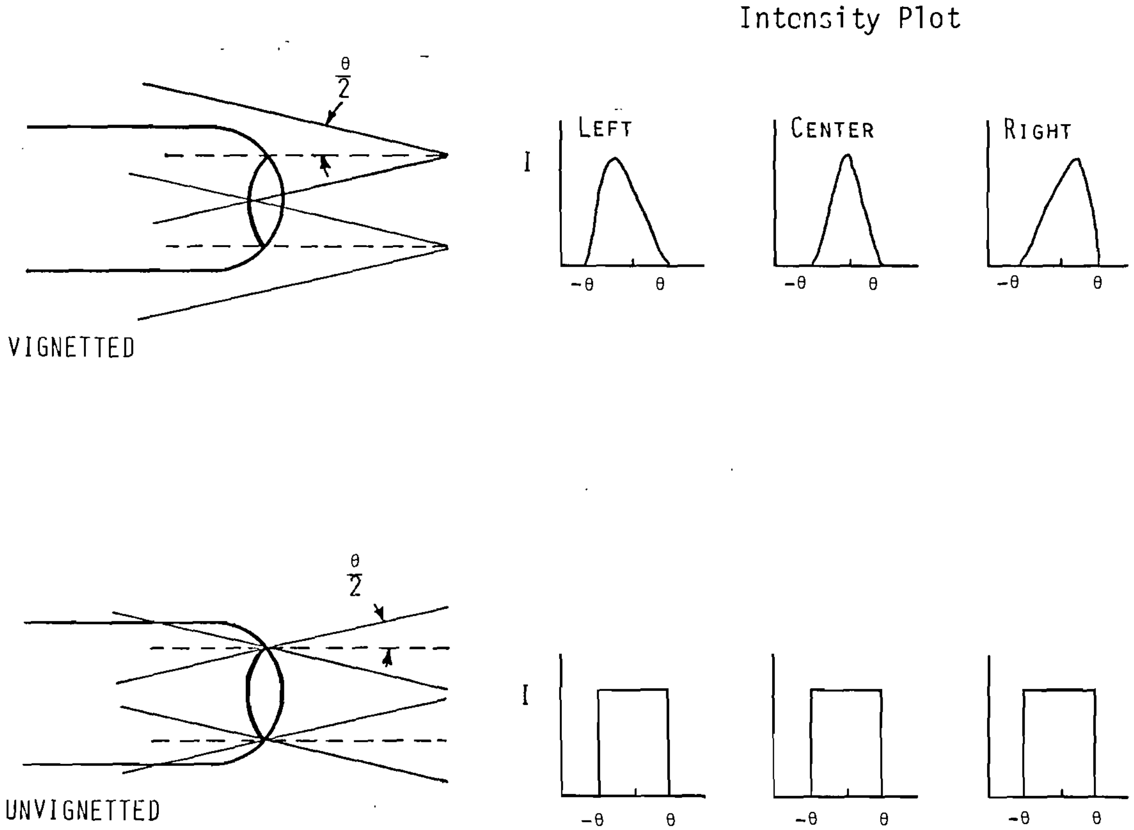


Figure 2-5. Effects of target source vignetting at the seeker aperture

2.3.4 Minimum/maximum irradiance at seeker

The target simulator should be designed to present average irradiances as low as the seeker NEI and as high as permitted by the seeker dynamic range. The minimum irradiances in Table 2-2 correspond to the minimum NEI's given in Table 2-1. The maximum irradiances correspond to the maximum NEI in the table times a nominal dynamic range, in most cases, of approximately 10^3.

2.3.5 Source-seeker distance, source height

Chamber and platform dimensions determine the required simulator adjustment ranges. It is seen from Figure 2-1, that the distance from the source platform to the missile pedestal is 24 ft. Therefore, the maximum seeker-simulator distance would be 24 ft if a seeker were removed from the missile body and placed directly on the pedestal. The shortest distance of 14 ft would be encountered for a seeker mounted on the longest expected missile body, or approximately 20 ft. Thus a 20 ft missile, mounted on the pedestal at the missile center of gravity, would shorten the seeker-simulator distance by approximately 10 ft.

The source (simulator) height relative to the seeker height will determine the required angular adjustment range. Since the stage, as shown in Figure 2-1, is 12 ft high, the missile pedestal height of 16 ± 2 ft would indicate that the stage could be as much as 6 ft, or as little as 2 ft, below the seeker. The final adjustment range required will, of course, depend upon the simulator support base height.

2.3.6 Maximum angular motion

A key parameter requirement is the maximum angular motion amplitude that the simulator is required to provide when viewed from the seeker. This motion amplitude must exceed the expected noise level in the seeker gimbal angle read out signal by some margin factor to permit ease of measurement. Thus the required amplitude can be expressed as:

$$\phi_m \geq N(\text{mv}) \times R\ (\text{deg/mr}) \times \text{margin factor} \qquad (2\text{-}2)$$

where R is either the slope of the tracker discriminator curve or the slope of the gimbal angle versus pick-off voltage curve, and N is the rms noise level in the signal measurement. In an RF susceptibility test, tracking performance may be measured at either point. Thus, with knowledge of N and R and with selection of a margin factor, the required amplitude may be estimated.

Pointing angle noise levels and angular responsivities may be estimated from typical seeker discriminator and pick-off curves. Figures 2-6 and 2-7 illustrate the characteristics that can be expected in such curves. Figure 2-6 shows a typical tracker discriminator curve. The discriminator angle responsivity is:

$$R_d = \frac{1 \text{ deg}}{100 \text{ mV}}$$

and the rms noise level is estimated to be:

$$N_d = 4 \text{ mV}.$$

Likewise, Figure 2-7 shows a representative gimbal angle pick-off curve. Here the angle responsivity is:

$$R_p = \frac{25 \text{ deg}}{3000 \text{ mV}}$$

and the rms noise level is:

$$N_p = 12 \text{ mV}.$$

The discrimination noise level and angle responsivity may be employed to determine the required target amplitude and other related requirements by introducing the concept of a noise equivalent angle (NEA). The NEA would be defined as:

$$\text{NEA} = N\ R \qquad (2\text{-}3)$$

and would simply be the angle (either discriminator or pick-off) equivalent to the noise level in the angle measurement. Thus Equation 2-2 could be rewritten as:

$$\phi_m \leq \text{NEA x margin factor.} \qquad (2\text{-}4)$$

It follows that the tracker discriminator noise equivalent angle is:

$$\text{NEA}_d = \frac{4 \text{ mV } 1 \text{ deg}}{100 \text{ mV}} = 0.04 \text{ deg}$$

and the pick-off noise equivalent angle is:

$$\text{NEA}_p = \frac{12 \text{ mV } 25 \text{ deg}}{3000 \text{ mV}} = 0.1 \text{ deg.}$$

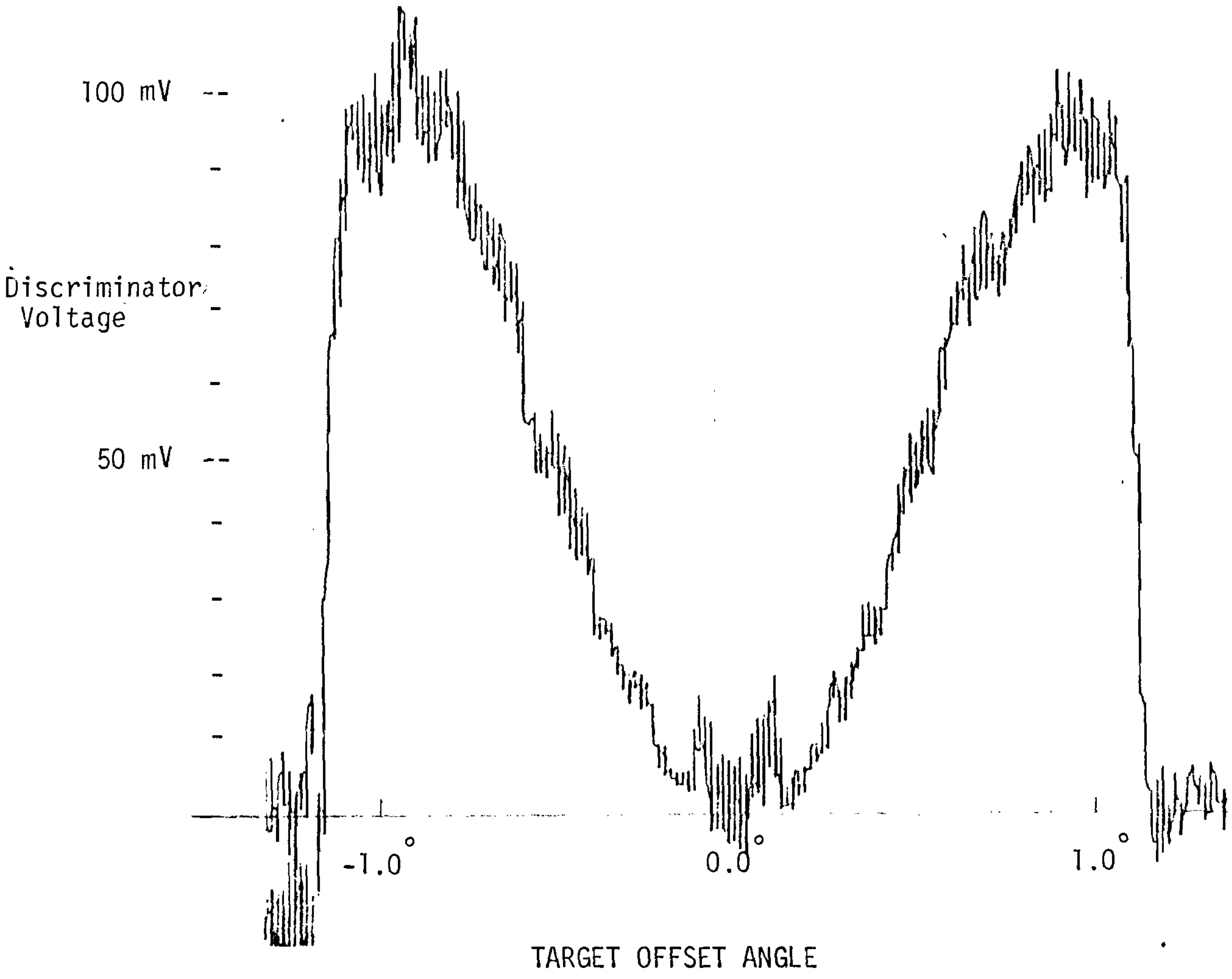


Figure 2-6. Tracker discriminator voltage versus target offset angle from line of sight

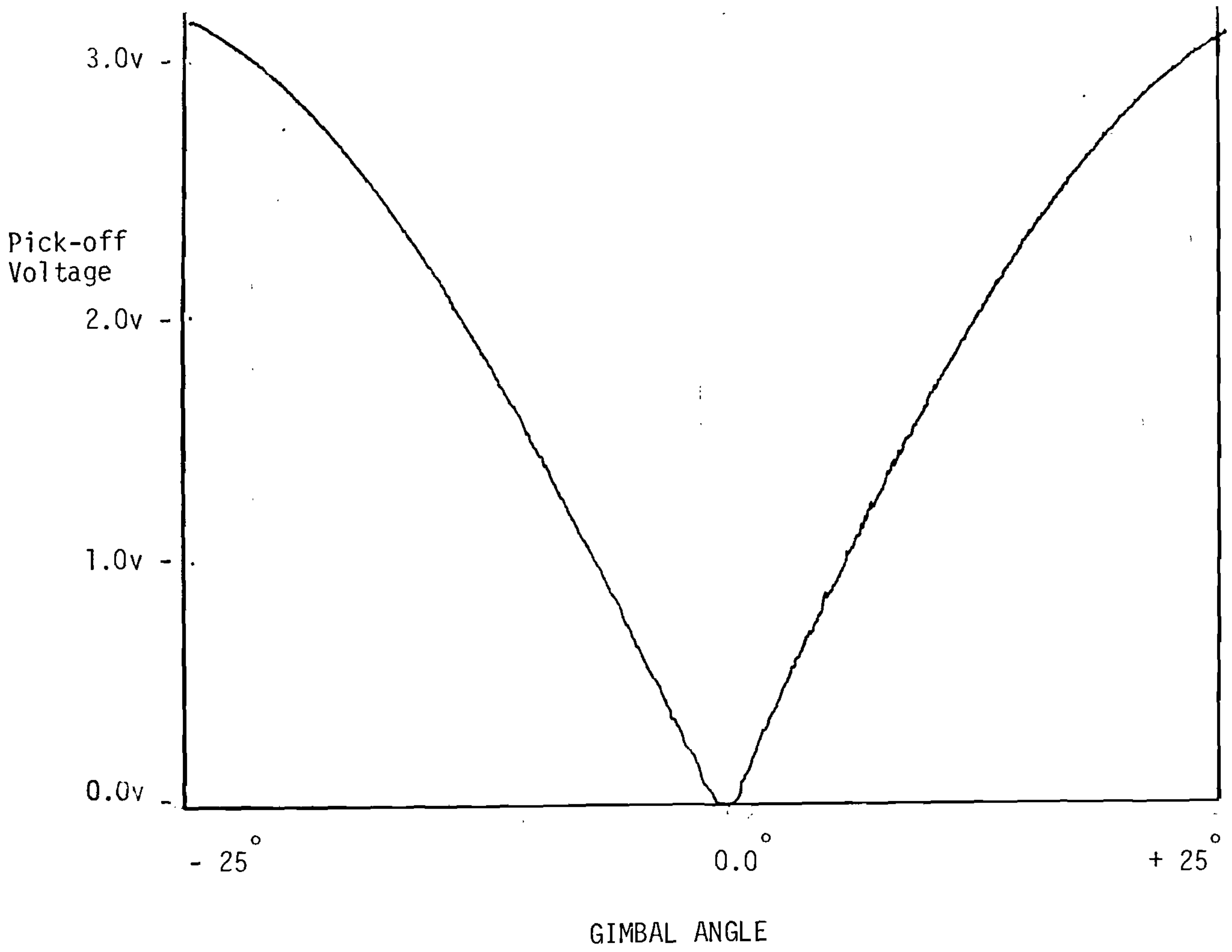


Figure 2-7. Pick-off voltage versus gimbal angle.

The required amplitude follows directly from Equation 2-4 for a given margin factor. Here a margin factor of 20 is selected. This choice is a result of allowing a factor of 2 to account for the possibility of encountering higher NEA's among the various seekers to be tested, and an additional factor of 10 to raise the angle measurement signal above the noise level for easy measurement. The required peak target motion amplitude is then the product of NEA_p, since it is higher than NEA_d, and the margin factor, or

$$\phi_m \text{ (req'd)} = (0.1 \text{ deg})\ (20) = 2 \text{ deg}$$

2.3.7 Target position repeatability, jitter

The target position repeatability and jitter requirements can also be related to the NEA. A reasonable position repeatability (cycle to cycle position change) and position jitter (intercycle position change) requirement is that such effects be held to approximately one-half of the NEA. Thus, if these errors are added in quadrature to legitimate EMI induced errors, they will result in less than a 15% error contribution. Therefore, the requirement shall be one-half the 0.1 deg (1.7 mr) NEA or approximately 0.8 mr.

2.3.8 Minimum target rate, rate variability

Again, the target rate and rate variability requirements can be related to the expected NEA. Here, however, the requirement is also related to the missile time constant. This time constant is the nominal time delay that occurs from the time a target position change is detected to the time the missile's velocity vector begins to change. Such time constants for tactical missiles seldom exceed approximately 1.5 seconds. The NEA, when divided by this time constant, produces an effective minimum rate which the target simulator should be capable of providing without unacceptable jitter, or

$$\frac{\text{NEA}}{\text{Time Const.}} = \frac{1.7 \text{ mr}}{1.5 \text{ s}} = 1.2 \text{ mr/s}$$

Likewise the variability in this rate should be acceptable if held to half as much or 0.6 mr/s.

2.3.9 Collimation accuracy/resolution

Rays emitted from a point source in the simulator should be collimated to a parallelism consistent with the expected seeker depth of field. Such a requirement could lead to a collimation accuracy of near 0.2 mr for 2 inch diameter seeker aperture operating in the shorter wavelength region. Therefore, a 0.2 mr collimation accuracy is a desired requirement. In reality, however, most of the seekers are not expected to possess resolution much higher than 1 mr. Those with better resolution will still function adequately with a 1 mr target blur size. Therefore, 1 mr will be the required collimation accuracy.

2.3.10 Laser source characteristics

Laser source characteristics, shown in Table 2-2, are selected to match the expected laser seeker characteristics. The maximum irradiance available from the simulator at the seeker front aperture is chosen from the product of a 10^4 laser seeker dynamic range and the highest laser seeker NEI in Table 2-1. The choice of a 10^4 seeker dynamic range is conservative since most laser seekers are expected to have a dynamic range of 10^6 or better. Nevertheless, the source operating range of 10^{-6} W/cm^2 to 1.0 W/cm^2 is believed to be adequate for EMI susceptibility testing. Furthermore, adjustability of the laser source intensity continuously over a 10^2 (Nd 2) range centered anywhere throughout the operating region is expected to allow adequate signal control for signal to noise ratio and range closure parameter variation studies.

2.3.11 E-O source characteristics

A visible or near IR source and background is required to provide a target for E-O missile seekers. These seekers are imaging devices which contain a video edge or centroid tracker. The primary source characteristics are contrast, target and background size, maximum irradiance, and irradiance adjustment range. The target would probably be a diffuser that is back illuminated by a tungsten projection lamp.

The signal to noise ratio can be varied by either varying the contrast or the signal intensity. Since provision for signal intensity control must be made for both the laser and the IR sources, it is assumed that the same mechanism would be used to control E-O signal strength rather than a separate contrast control.

While the contrast control would be a desirable feature, it would require an additional source to control the background radiance. Such a dual source arrangement should be capable of varying contrast from 2% to 100% and still be able to meet the specified maximum and minimum irradiance levels in Table 2-2.

If a single source is used, that source should be capable of presenting a constant contrast fixed at a value no less than 20%. It must remain constant as the source intensity is varied to provide the specified maximum and minimum irradiance levels. In either case a capability to present both positive and negative contrast is required.

The target background size presented to the seeker is an important consideration that affects the capability to test over the desired range of parameter variations. Sufficient uniform background extent must be provided to ensure that undesired background objects do not infringe on the track gates. The infringement could occur under low signal to noise ratio (SNR) conditions or high track rates when the gates are expected to drift about the target. Here, even vignetting of the background would reduce its effective extent in much the same way that vignetting reduces the target extent as illustrated in Figure 2-5. The potential problem with background vignetting is that the vignetted background pattern might present a more desirable "target" to the seeker than does the intended target. The gate transition from tracking the true target to tracking the vignetted background might not be detected. This phenomenon might also occur so frequently at the lower SNR's that it would restrict the desired range of parameter variations. Here it will be assumed that an unvignetted background size equivalent to 2° or twice the 1° maximum E-O target size is adequate.

A more desirable controlled background extent would be 9^{o}. This would completely fill the 5^{o} maximum E-O seeker field of view in Table 2-1 and allow for the $\pm 2^{o}$ of required target motion as well.

2.3.12 IR source characteristics

The target simulator IR source must be capable of meeting the requirements for both imaging and reticle IR seeker types. Maximum required irradiance values shown in Table 2-2 follow directly from the range of expected seeker NEI's and a 10^{3} dynamic range. Again the background size is an important consideration. As with the E-O seeker a 2^{o} background is a reasonable requirement for a maximum 1° target subtense. It should be noted, however, that when the lower rates of mid-course guidance are simulated, much smaller target sizes, and therefore background sizes, can be tolerated with either IR or E-O imaging seekers.

Here the effects of background vignetting may not be as severe as they would be with E-O seekers. Certain E-O seekers require negative contrast targets. In that case the background would be brighter than the target, but the background radiance must match the foreground radiance to present an unbounded background extent.

Infrared seekers, however, are generally designed to operate against positive rather than negative contrast targets. In the simulator, the focal plane background will tend to have approximately the same temperature as the foreground, or baffle, in front of the simulator. Since vignetting between surfaces of the same temperature will produce no modulation, provided the surfaces have equal emissivities, it is likely that the target background will appear unbounded. Of course, the advantage of an unbounded background is its lack of potential false track objects for either imaging or non-imaging trackers.

The problem with achieving an unbounded background with the above logic is the difficulty of matching focal plane and front baffle temperature and emissivities. Even if emissivities can be well matched, heat build-up at the focal plane from target sources could cause an apparent difference of 0.2^{o} to 0.3^{o} C - enough for some seekers to lock on and track. Therefore, it is still desirable to provide as large an unvignetted focal plane background as possible.

The background size requirement for reticle seekers can be estimated from the largest target size that they can track. This is nominally 5 mr. Thus if a larger 8-10 mr (½ deg) background were provided to account for seeker to seeker variability and possible seeker design changes, the simulator would allow reticle seekers to be tested with little risk of false track difficulties. Likewise, both E-O and IR imaging seekers could be tested with a smaller unvignetted background of ½° as well – provided rates were held to the lower levels encountered in mid-course guidance.

2.3.13 Controls

Control of target intensity and angle rate is required to be remote. This shall simply mean that some technique must be available to adjust lamp or blackbody intensity and servo motor speed from a remote control panel. The source type may be removable so that the appropriate source for a given seeker can be inserted before test activity begins.

2.3.14 Alignment/portability

It is required that the target simulator be portable and easily aligned with the missile seeker. The portability requirement could be met by modularizing the simulator so that the source/collimator could be removed. The objective is to allow the simulator to be used, without target dynamics, with the seeker outside the chamber. Alignment both inside and outside the anechoic chamber may be accomplished manually.

2.3.15 Status display

Sufficient information should be provided to permit derivation of target position, rate, and intensity. In general these requirements could be met with precalibrated position or rate transducers which furnish a voltage proportional to rate or position. In the case of intensity, signal change as a function of a control voltage would be adequate. However, the intensity change must preserve the apparent color temperature characteristics of the source.

3.0 CONFIGURATION TRADES

Three target simulator design concepts are addressed in this section. A comparison of the three configurations serves to identify design trade-offs and the major problems that would be encountered in a detailed design.

3.1 Design Considerations

The design of the target simulator must address four major areas in order to satisfy the seeker design requirements discussed in section 2.0. They are:

1) the scan mechanism (optical and mechanical),
2) the field-of-view (FOV),
3) the depth-of-field (DOF),
4) the spectral range.

The scan mechanism must meet the seeker simulation requirements, both optical and mechanical, for adequate performance. These requirements are scan type, rate, rate variability, position repeatability, acceleration, jitter, and position indication.

The target simulator must be able to provide an unvignetted image (containing the entire FOV) of an extended target with an acceptable background. Configuration drivers include object (target plus background) size, focal length, aperture, resolution, seeker distance, and the unvignetted spot diameter needed at the seeker.

The large depth-of-field (DOF) requirement for the seeker necessitates the use of collimating optics at some point in the optical path. Practical laboratory dimensions are relatively small and do not permit the simulator-to-seeker distances required to present an in-focus image.

To satisfy the spectral coverage requirements of the various seekers, the target simulator must be designed with optical and mechanical components which provide a wide spectral range. This requirement typically drives the design to reflecting optics.

Additional driving considerations include physical size limitations of the system, mechanical and optical design complexity, portability, weight, component availability and cost.

3.2 Candidate Configurations

The three configurations to be considered are the:

1) Rotating Aperture Configuration,
2) Stationary Aperture Configuration, and the
3) Projection Configuration.

A block diagram of the optical train comprising these configurations is shown in Figure 3-1. Figure 3-2 illustrates the scan pattern and image seen by the seeker with each of the three concepts.

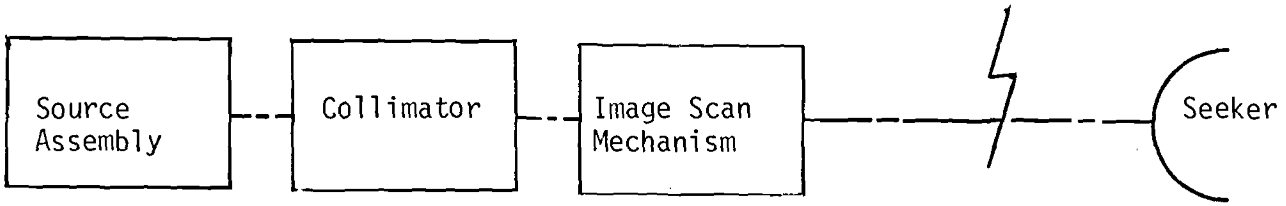
The rotating aperture configuration produces target motion with a rotating periscope assembly. The periscope exit mirror is tilted to produce beam cross-over at the seeker location. This concept has been implemented in the NSWC anechoic chamber at Dahlgren, Virginia. Figure 3-3 shows a photograph of that system. Figures 3-4 and 3-5 show the system with front and back panels removed. The rotating aperture approach has been selected for the strawman design and will be discussed in more detail in section 4.0.

The stationary aperture configuration generates the target motion in the collimator focal plane. A scan pattern can be generated with this concept by simply rotating a target aperture. The source must either be extended to cover the extremities of the motion or be made to follow the target with a reduced version of the periscope.

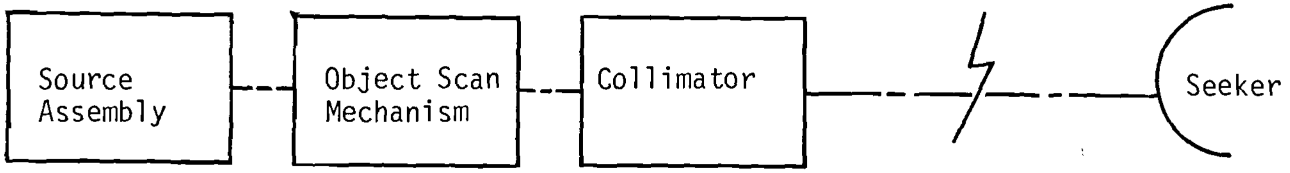
The projection configuration forms an image of a target either on a reflecting screen illuminated from the front or on a translucent screen illuminated from the rear. Scanning would typically be accomplished with mirrors. Since the projection concept produces an image on the screen, a collimating lens would have to be provided at the seeker to enable the seeker to see an in-focus image.

3.3 Configuration Comparison

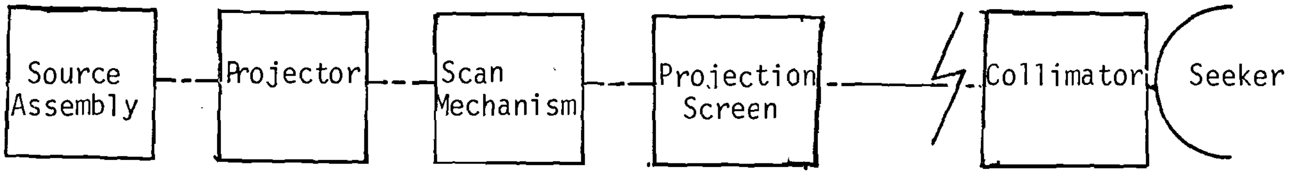
Table 3-1 lists the relative design trade-offs between the three configurations.



a) ROTATING APERTURE



b) STATIONARY APERTURE



c) PROJECTION

Figure 3-1. Target simulator configurations (block diagram)

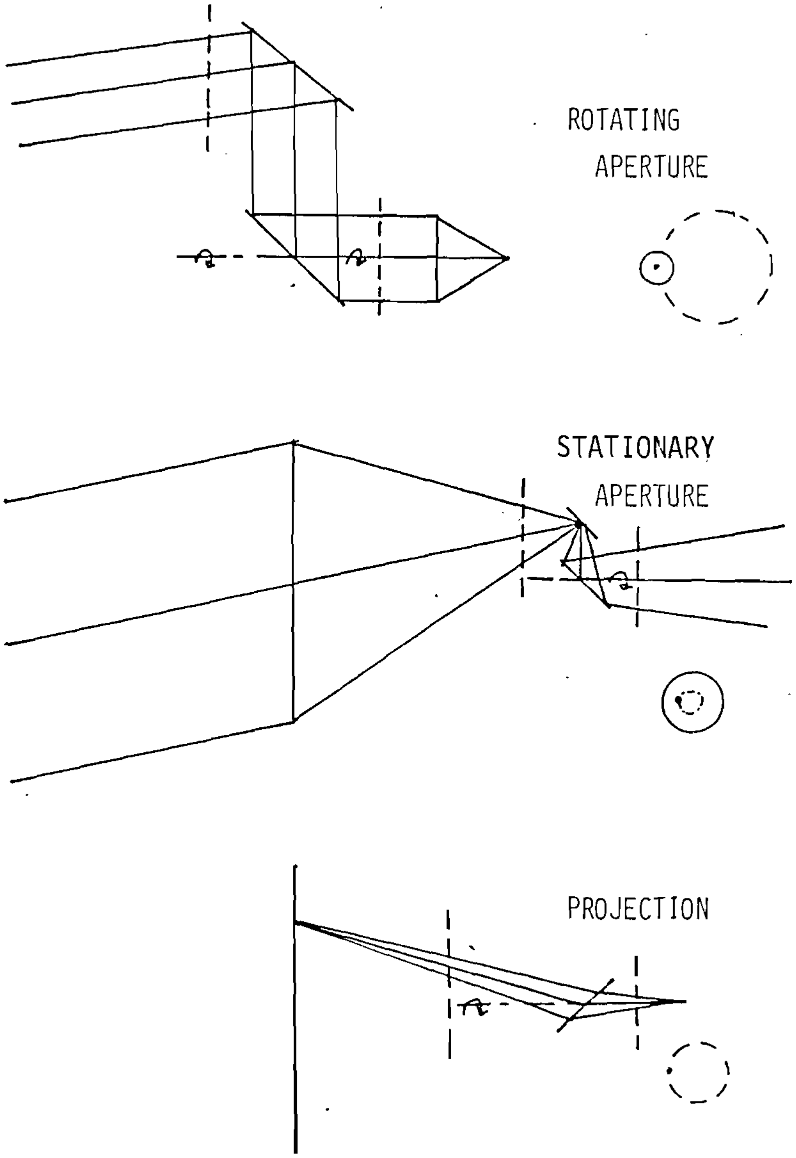


Figure 3-2. Target simulator configurations

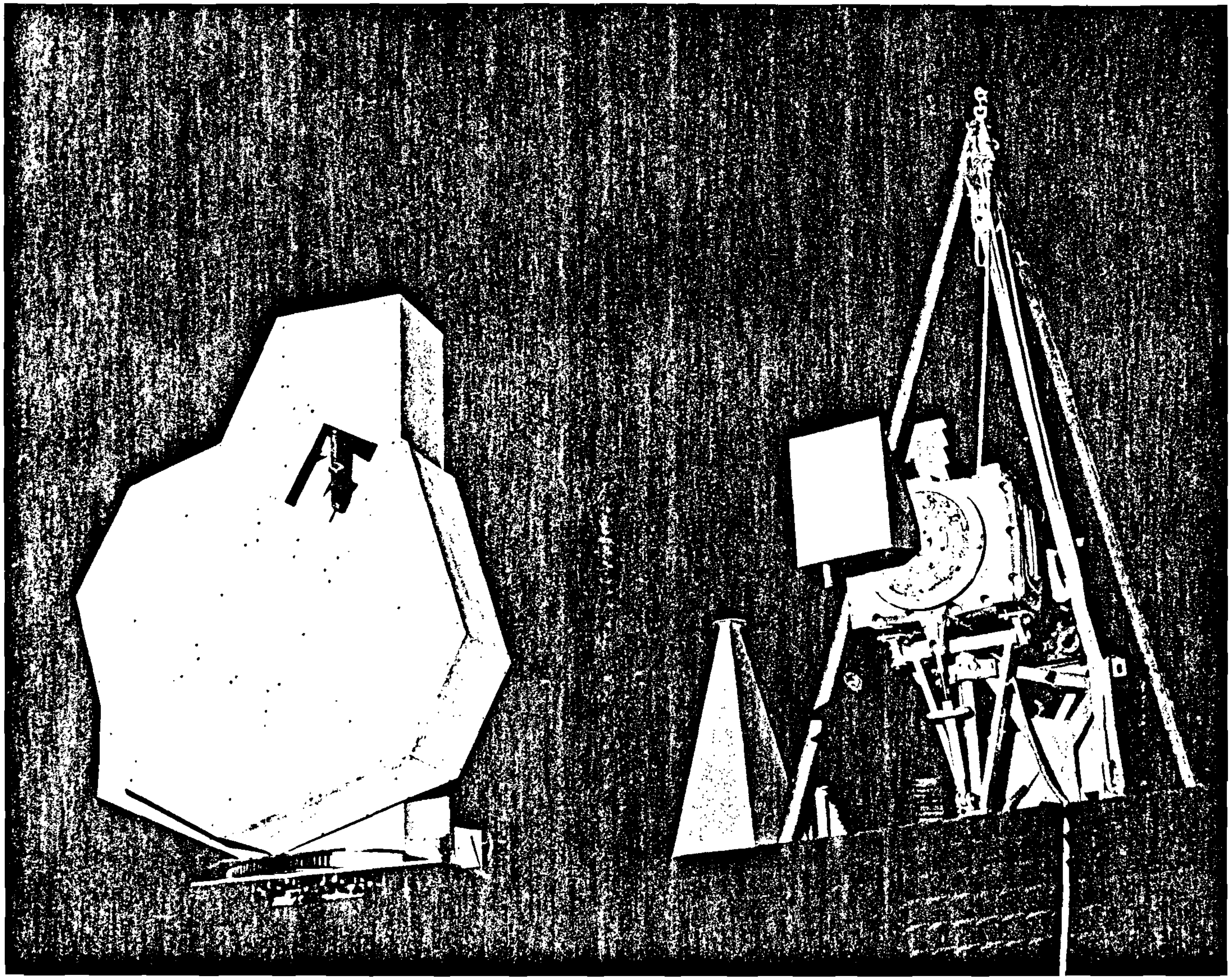


Figure 3-3. Rotating aperture concept employed in the NSWC anechoic chamber

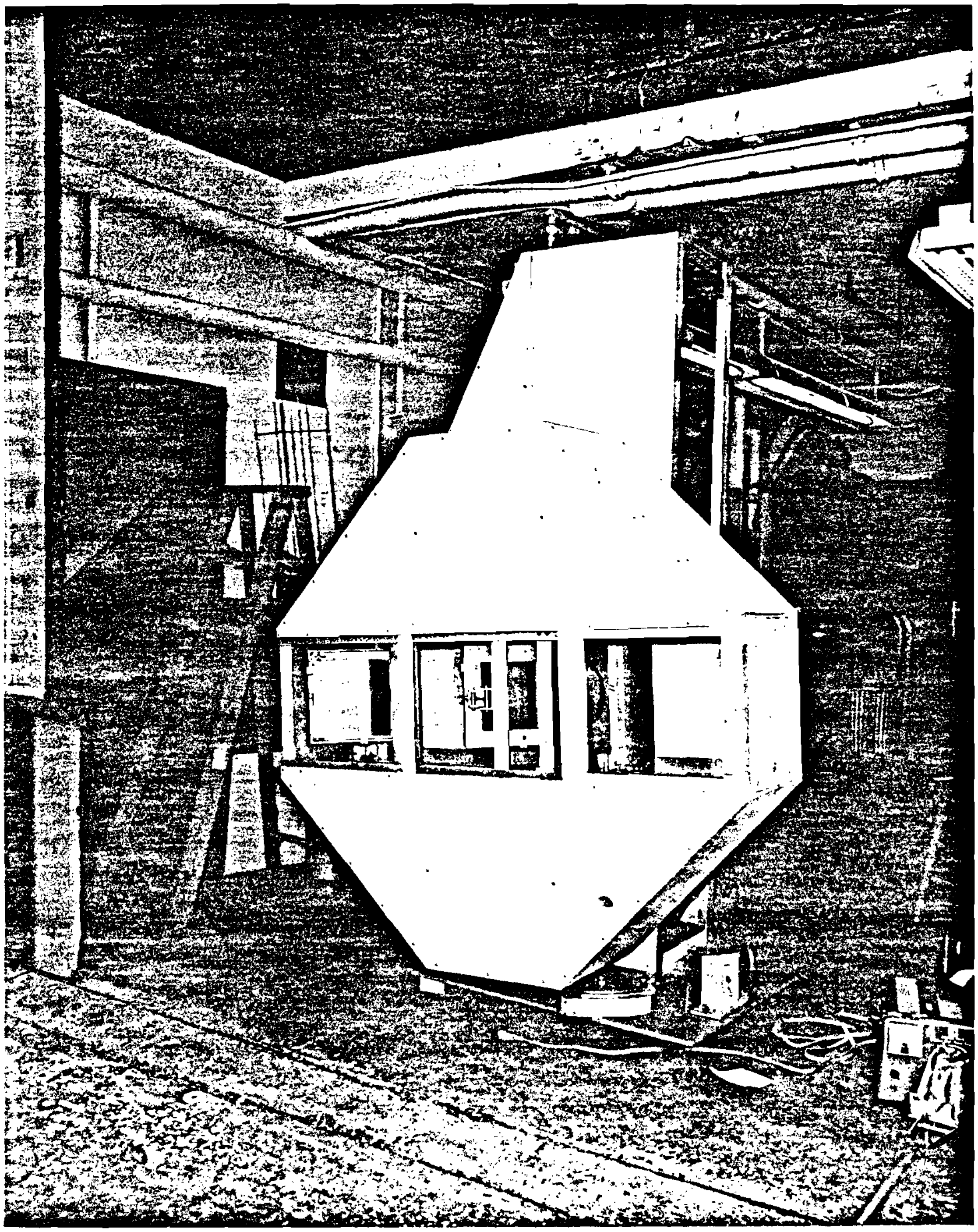


Figure 3-4. NSWC simulator with front panel removed

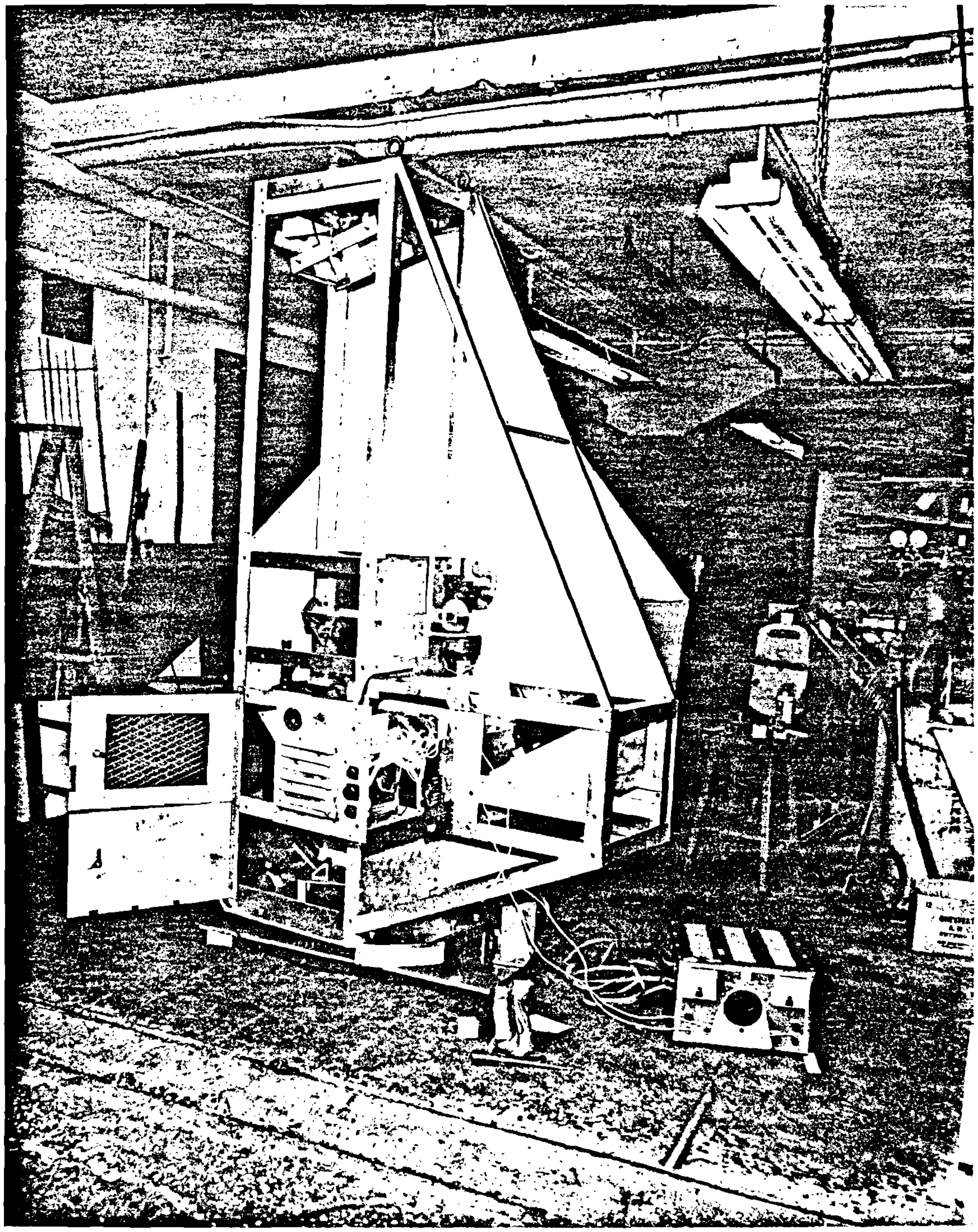


Figure 3-5. NSWC simulator with rear panels removed

Table 3-1. Configuration comparison

DESIGN PARAMETER	ROTATING APERTURE	FIXED APERTURE	PROJECTION
Aperture	Moderate	Large	Small
Mechanical Design	Complex	Moderate	Simple
Optical Design	Simple-near axis	Complex - high off-axis angles	Simple - near axis
Scan	Fixed: Low rates	Flexible: moderate rates	Flexible; high rates large angles
Depth of Field	Large	Large	Short without collimator lens at seeker
Power Control	Simple	Simple	May be limited; Projection materials questionable
Seeker Spot Size	Moderate	Limited	No limits

The rotating aperture configuration is simple in concept. It generally requires smaller optics than the stationary aperture system since the FOV requirement is considerably less, as illustrated in Figure 3-2. Furthermore, the optical design is relatively simple because no image plane relay optics are required. An extended object will require relatively small off-axis angles and produce high resolution. This configuration, however, is mechanically complex because of the large rotating periscope assembly required.

The stationary aperture configuration dictates a focal plan scanning mechanism that would be much less complex mechanically mainly due to its smaller size and weight. This system has the capability of a wider range of scan types and rates in addition to better positional and rate variation accuracy. Optically, however, the system is more complex since the target in the focal plane will be moving. The target movement will require large off-axis angles in order to simulate motion for the seeker. The collimator f-number (aperture and focal length) must be carefully considered in order to meet the target resolution requirement. Resolution decreases at large off-axis angles due to optical aberrations. Finally, the system will require large optics to provide an unvignetted FOV at the seeker.

The projection configuration has the scan flexibility and accuracy achievable in the stationary aperture system. Much larger scan angles would, however, be possible. The imaging system would simply project an arbitrary size target onto a reflective or transmissive screen and thus could be designed using small optics. Serious questions exist as to the availability of materials suitable for the projection screen, either by reflection or transmission. These materials must meet broad spectral coverage requirements. Another major problem with this concept is that that target is imaged onto the screen and does not provide collimated light to the seeker. This problem could be corrected by supplying a relatively small wide band lens at the seeker, but a lens could possibly have an undesirable effect on the RF field used for interference testing. In addition, a front projection approach could place the projector in a position to interfere with the RF pattern. Finally both front and rear projection approaches are likely to encounter difficulties in providing the required power levels in the infrared spectral regions.

3.4 Design Trades Procedure

For the rotating and stationary aperture configuration, the collimator is the major component that must be specified. The geometrical and analytical details of beam production by a collimator with an extended source are analyzed in Appendix A.

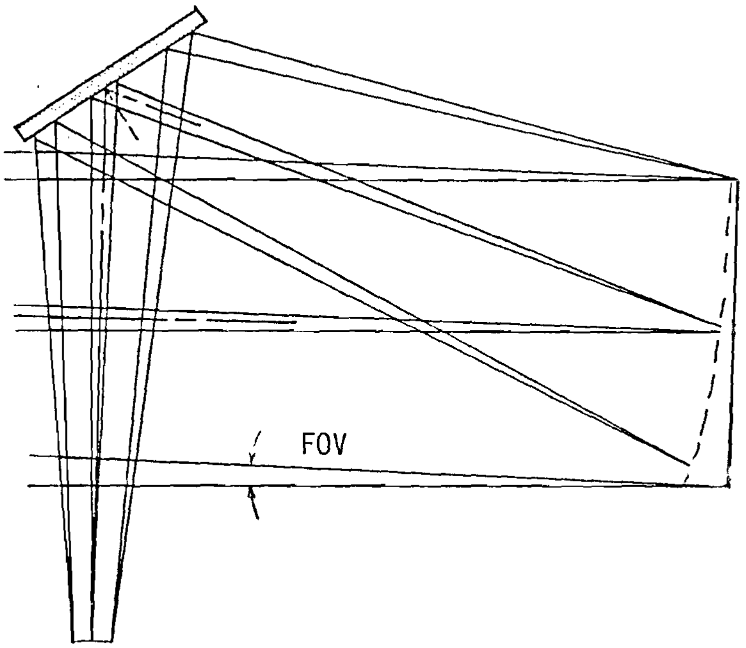
The aperture of the collimator may first be determined from the required FOV, the aperture-seeker distance, and the required unvignetted beam diameter via Equations A-10a and A-10b.

The selection of a collimator focal length and therefore the f-number (given the aperture) becomes an interactive process involving several considerations:

a) The off-axis angles for the stationary or rotating target directly affect the target resolution due to spherical aberration (if a spherical mirror is used), coma, and astigmatism.

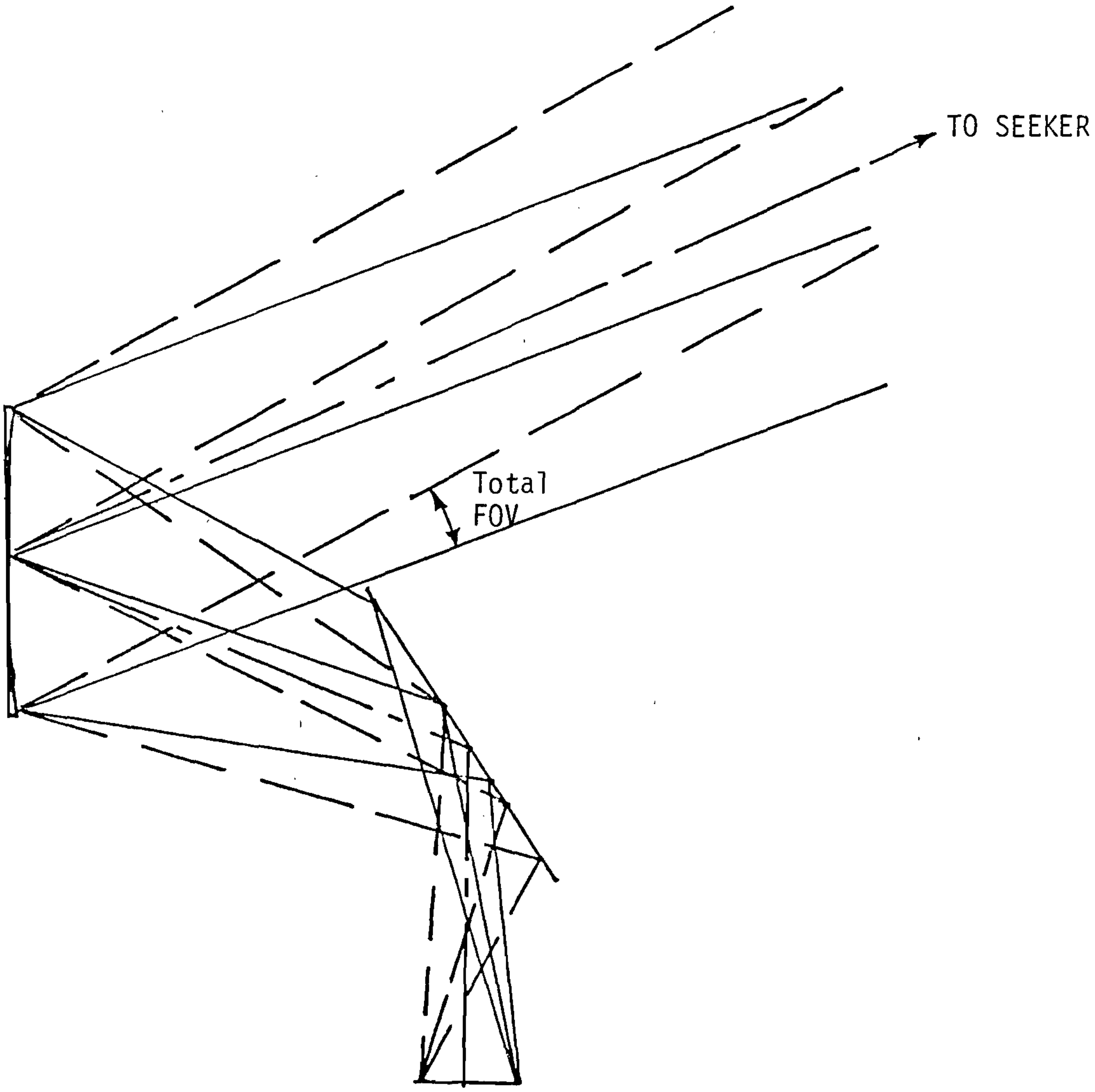
b) Physical restrictions on the size of the collimator will generally require folding the optics without vignetting the entire FOV that must be imaged. Examples of folding for the rotating and fixed configurations are illustrated in Figures 3-6 and 3-7. Any folding will increase the off-axis angles required. A paraboloid mirror has no spherical aberration, and hence can be used at large off-axis angles with little effect on resolution. Adequate working space in the focal plane is also a requirement which must be considered in the collimator configuration. Generating an adequately folded collimator to meet all requirements, including procurement, entails a major design effort.

c) Specification of the target subtense and collimator focal length determines the physical size of the target in the focal plane. This must be a practical dimension in order to allow adequate irradiation based on sizes and intensities of readily available radiation sources.



FIXED FOCAL PLANE

Figure 3-6. Optical folding for an f/3.75 collimator (rotating aperture configuration)



SCANNED FOCAL PLANE

Figure 3-7. Optical folding for a collimator (stationary aperture configuration)

4.0 STRAWMAN DESIGN

A strawman design configuration that meets the requirements in section 2.0 is developed in this section. The purpose of the strawman design effort is not to develop the optimum configuration for meeting the requirements but rather, to show the impact of these requirements on a representative example design.

The rotating aperture concept is chosen for this example design. It is selected because (1) it promises to meet the design requirements with the least complexity and, (2) all of the major optical and mechanical parts appear to be obtainable - although many require custom fabrication. Though these reasons are not sufficient to guarantee that the resulting design is optimal, it does indicate that the resulting design will be representative.

4.1 System Description

A sketch of the strawman design for the rotating aperture configuration showing the basic mechanical and optical layout is illustrated in Figure 4-1. Figure 4-2 shows a scaled optical design of the target simulator which will meet all the optical and physical requirements. Figure 4-3 shows the focal plane and source layout.

The target simulator designed using the rotating aperture concept is comprised of the following four optical subassemblies: (1) the rotating periscope; (2) the fixed collimator; (3) the focal plane configuration; and (4) the controlled source assembly. A description of each of these subassemblies follows.

4.1.1 Rotating periscope subassembly

The rotating aperture concept is achieved by reflecting the image beam through two large moving mirrors in a periscope arrangement as illustrated in Figures 4-1 and 4-2. The central periscope entrance mirror is set at 45° to the collimator's optical axis so that the beam is reflected normal to its original direction. The beam then intercepts the periscope's exit mirror which is tilted slightly off of the 45° reference direction. This is done in order to direct the image beam

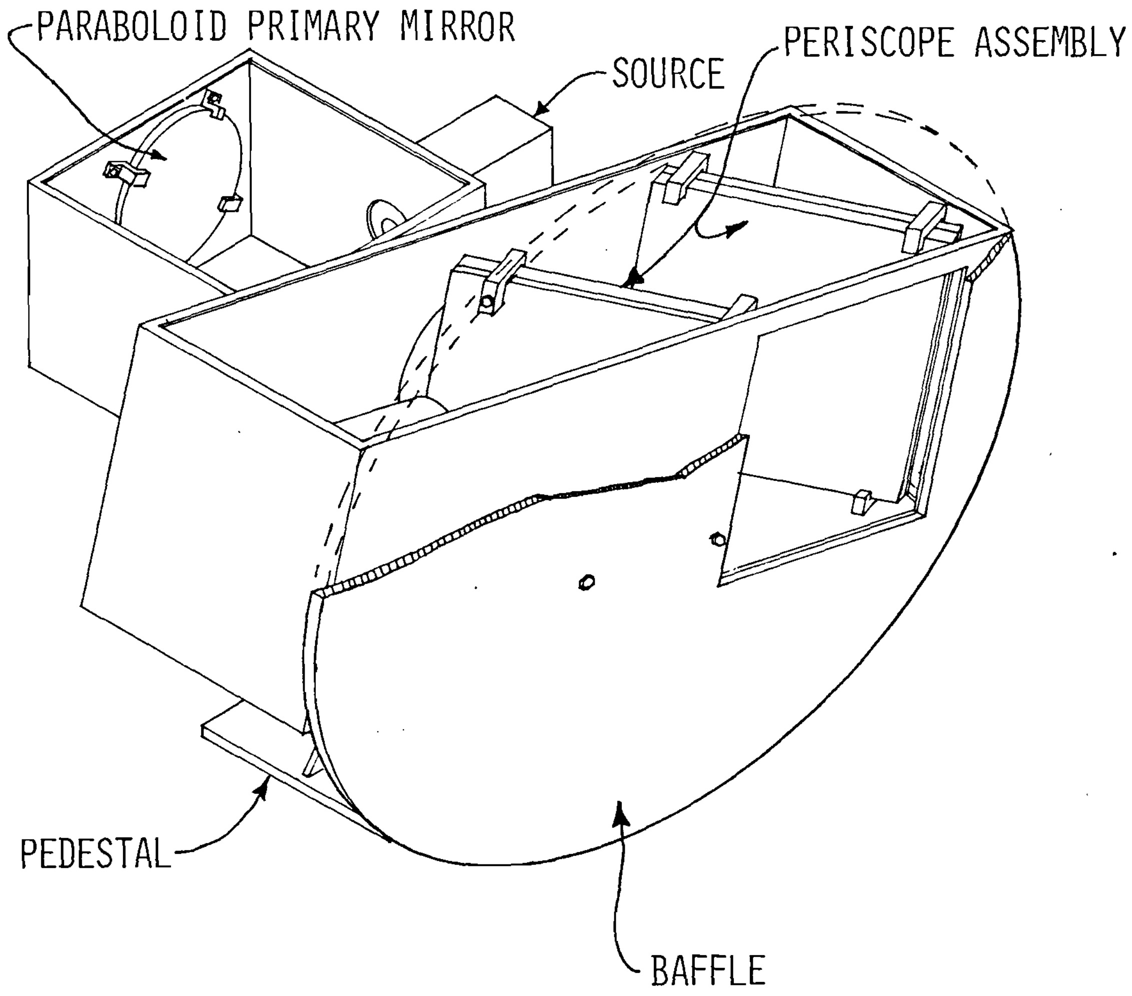


Figure 4-1. Strawman design configuration (perspective view)

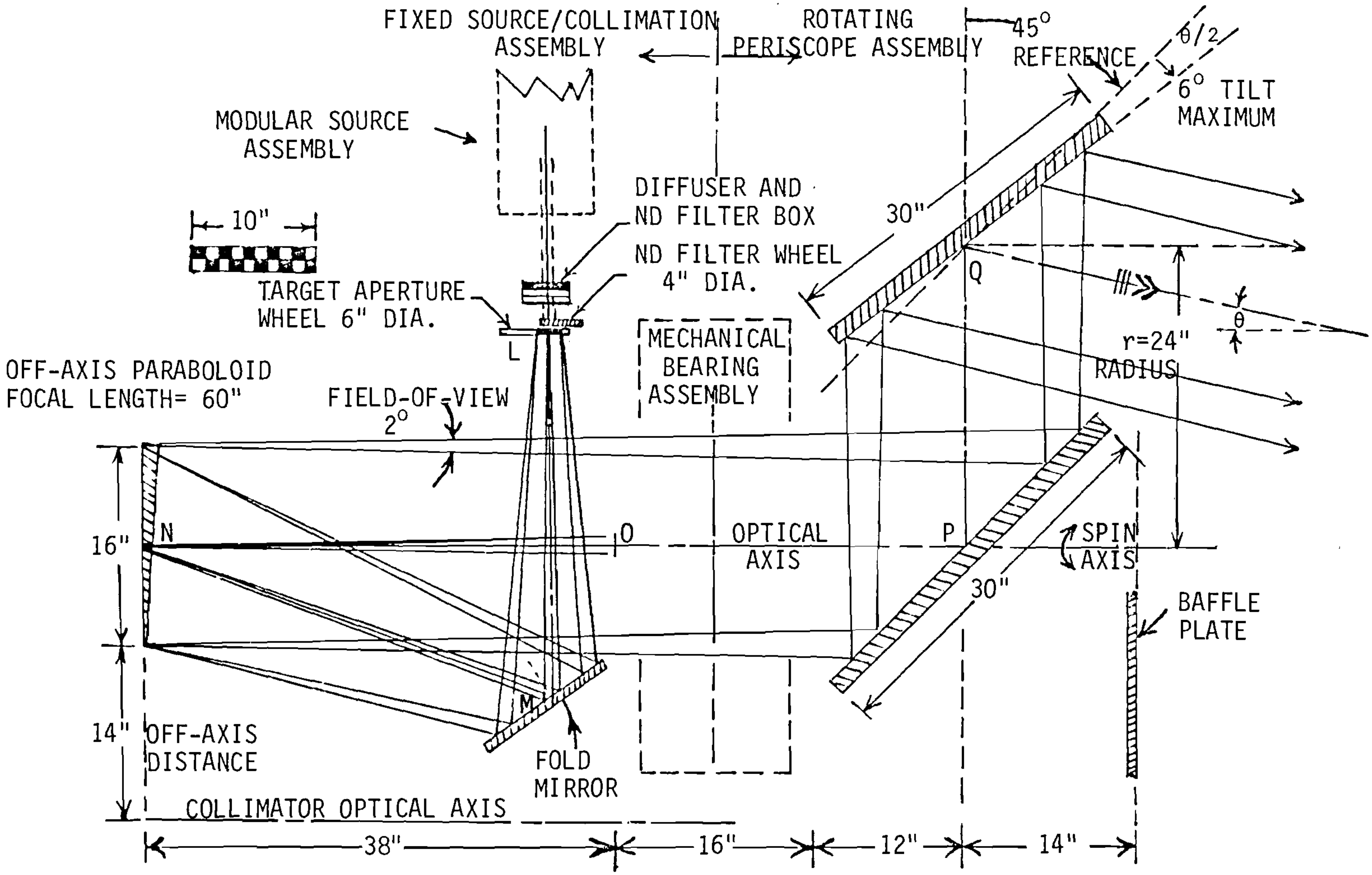


Figure 4-2. Strawman design configuration (top view)

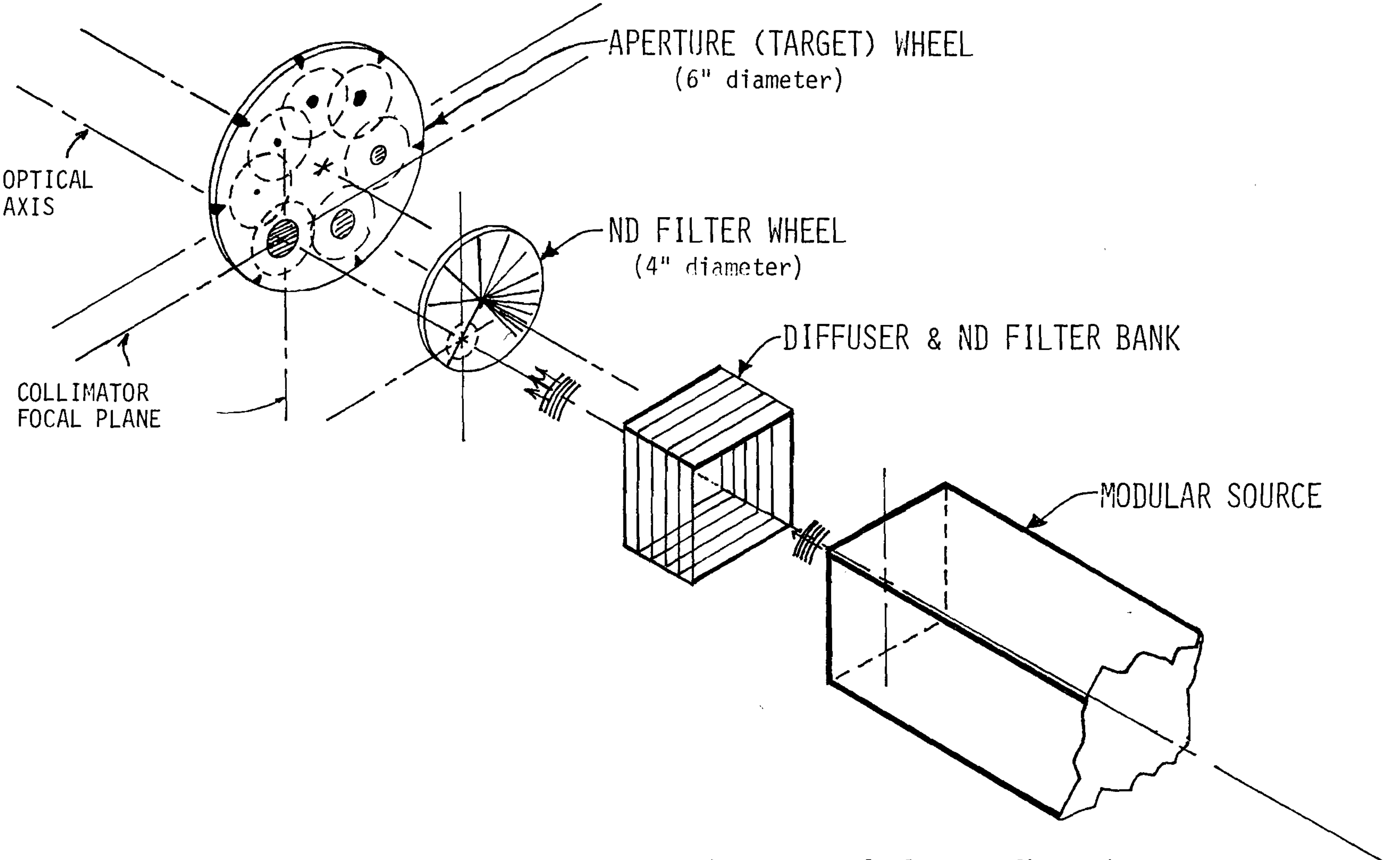


Figure 4-3. Source generation and focal plane configuration

to eventually intercept the spin axis (in line with the collimator axis) at an angle of twice the mirror tilt angle. The tilt angle is adjustable to allow the intercept along the spin axis to be variable over a large range. The entire subassembly is then rotated by a servo motor at a controlled rate about the spin axis. The periscope subsystem is coupled to the remainder of the simulator by a mechanical bearing. The bearing possesses a large enough bore for the image beam from the collimator to pass through unhindered. Attached to the periscope assembly is a large baffle plate, illustrated in Figure 4-1, for uniform background control. An analysis of the details of the beam crossing geometry is contained in Appendix A.

4.1.2 Fixed collimator subassembly

The function of this subassembly, as shown in Figure 4-2, is to provide an image beam of the fixed object located precisely in the focal plane. Since the object is extended and not a point source, the collimation process must not reduce the object's field of view. A reflecting collimator design is necessary due to the wide wavelength range requirement extending into the far IR. The geometrical and analytical details of the collimation process are also analyzed in Appendix A.

4.1.3 Focal plane subassembly

Figure 4-3 illustrates a notched target wheel containing various circular aperture sizes. Manually locking the wheel into a detent centers the desired aperture precisely in the focal plane and on the optical axis of the collimator. The design specifications in section 4.2 lead to a 1" diameter aperture for a 1° target angular subtense. The target simulator is required to image a 2° FOV object (target plus background) and thus will require a 2" diameter clear region around the aperture. A 6" diameter wheel would allow a minimum of six apertures each possessing a 2¼" diameter clear region with a maximum aperture of 1" diameter.

4.1.4 Source generation and radiation control

The broad spectral coverage requirement necessitates the use of three types of radiation sources. A blackbody source is required for the IR imaging and reticle seekers. A tungsten lamp, or the equivalent, is required for visible and near IR E-O seekers, and a laser source is required for laser seekers.

The minimum design effort and least costly method of incorporating the sources in the target simulator is to provide interchangeable source modules. Each of the source modules may be used with the dual intensity control scheme illustrated in Figure 4-3. This scheme offers course intensity control with a bank of Nd filters and continuous control, for smooth signal variation, with a variable Nd filter wheel. Precalibration of this wheel with each of the three source modules will allow the seeker signal to noise ratio to be varied during tests.

Continuously variable ND filter wheels are available from Optical Coatings Laboratory Incorporated. A stock 4" diameter wheel will allow a 1" usable aperture with neutral attenuation over the 0.38-3.0 μm wavelength range. A custom wheel must be fabricated to cover the 3.0-12 μm range. The attenuation ratio can be specified to be over the range from Nd 0 to around Nd 4 which will manifest itself as a linear function of rotation angle from 0-270°. Specifying an adjustment range from Nd 0 to Nd 2 will satisfy the intensity variation required for all of the sources. For a 4" diameter wheel, there will be about a 15% intensity variation, measured tangentially, over a 1" diameter usable area. This figure may be substantially reduced, if necessary, by using a larger 6" diameter wheel or the variation may be eliminated entirely by using multiple wheels.

4.2 Design Specifications

Several key design specifications drive the strawman configuration to the dimensions shown in Figure 4-2. These specifications are collimator aperture, focal length, field-of-view, and resolution. Field coverage and simulator-to-seeker distance determine the collimator aperture diameter according to Equations A-10a and A-10b in the Appendix. Figure 4-4 shows a plot of collimator diameter versus FOV (target angle) for several simulator-to-seeker distances. The maximum expected simulator-to-seeker distance is 24 ft per section 2.0. However the maximum distance from the seeker to the collimator aperture will be 4-5 ft longer or 28-29 ft. Thus it is seen from Figure 4-4 that, for the required 2 deg FOV, the collimator aperture must be approximately 16 inches.

The focal length is constrained both by a desire to keep it as short as practical and by the requirement to present a 1 deg (0.017 rad) maximum target angular subtense. Since most commonly available blackbody cavities present diameters of no greater than 1 inch, the focal length should be sized to produce a 1 deg subtense with a 1 inch cavity. The required focal length is therefore 1 in/0.017 rad or approximately 60 inches.

Finally, the 1 mr system resolution requirement determines the type mirror, spherical or paraboloid, to be specified. Collimator f-number and fold angle are important in this determination. The choice of a 60 inch collimator focal length with a 16 inch aperture results in an f-number of 3.75. Folding of an f/3.75 optical path requires an approximate 8°(14 mr) off-axis mirror angle. This off-axis angle results in an excessive resolution loss as indicated in Figure 4-5. Figure 4-5 shows a plot of the blur diameter as a function of off-axis angle for several f-numbers. The blur diameter with an f/3.75 spherical mirror will exceed 2 mr. It is therefore concluded that an off-axis parabolic mirror is required.

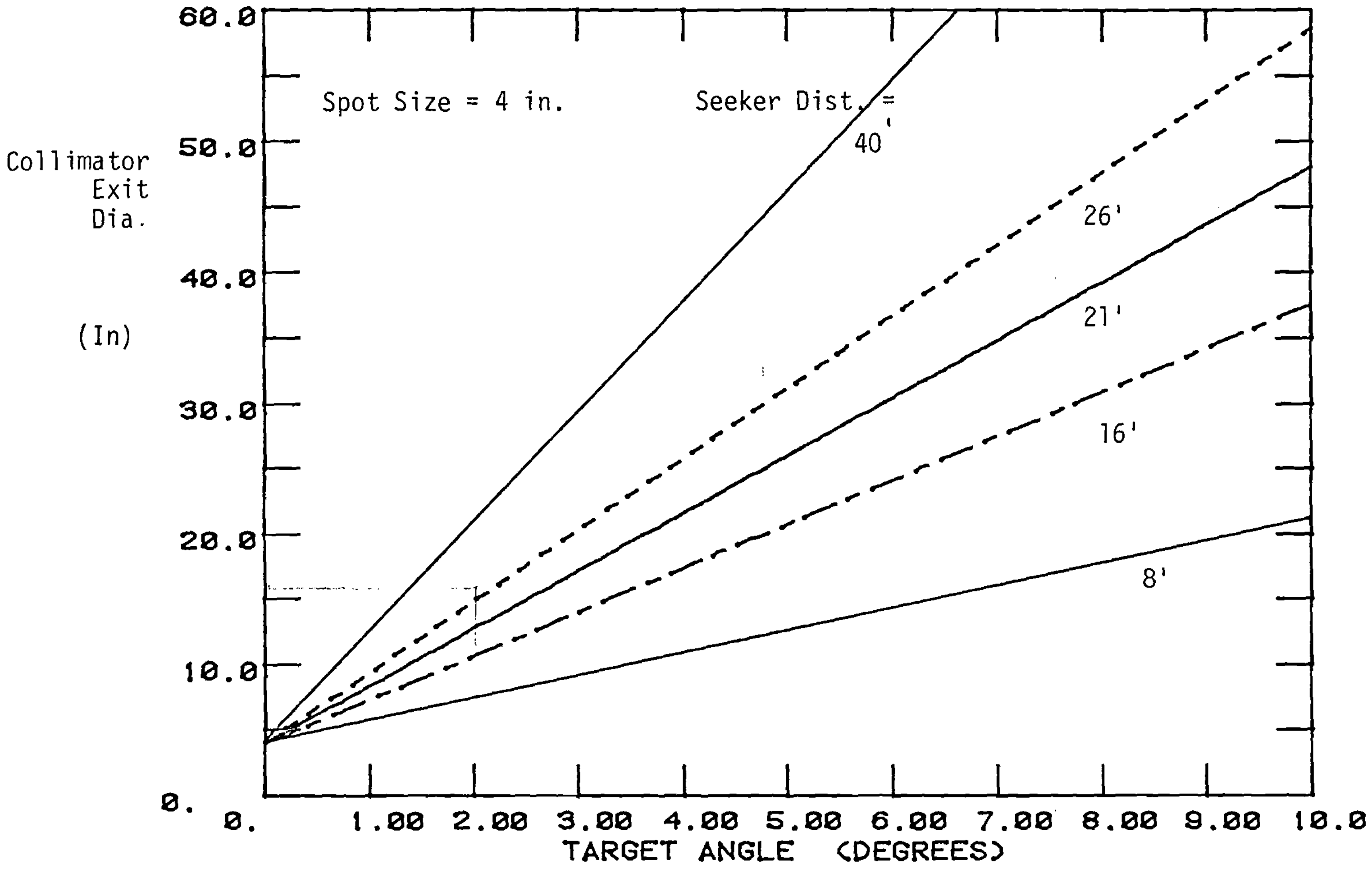


Figure 4-4. Collimator aperture size requirements

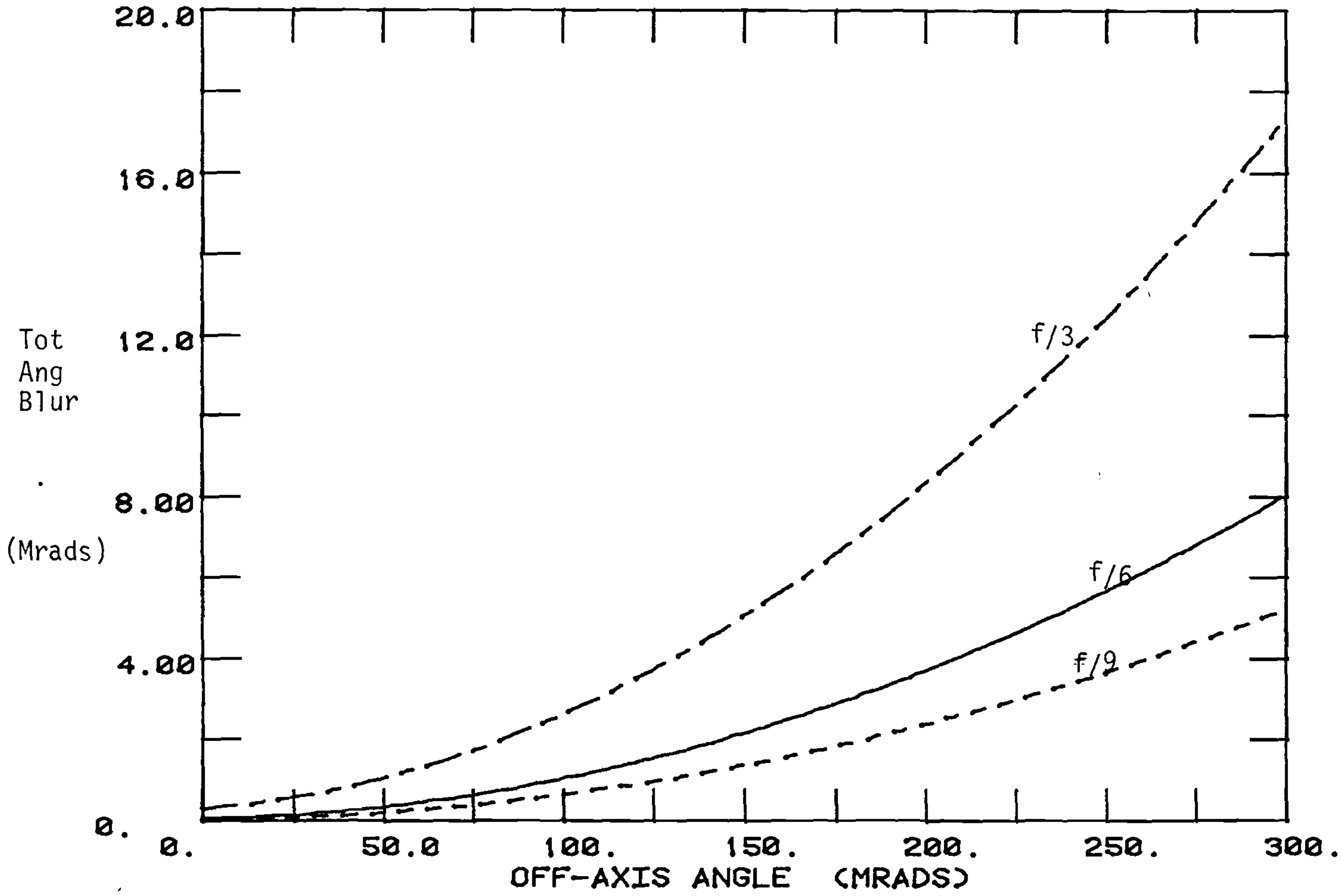


Figure 4-5. Resolution available with off-axis viewing

5.0 COST ESTIMATE AND EVALUATION

5.1 Strawman Cost Estimate

One of the major purposes of the strawman design exercise is to provide a representative configuration from which to estimate procurement costs. In this section the costs of the strawman design described previously are examined. This examination is expected to lead to an identification of the major configuration cost drivers. It is also expected to help focus attention on areas where cost savings can be made and where those efforts would be wasted.

The costs of the strawman configuration are summarized in Table 5-1. Here the major cost category is personal services. A breakdown of personal service costs by task, as shown in Table 5-2, indicates that mechanical design (including engineering liaison), drafting, and fabrication constitute the largest cost contributors. This is a result of the mechanical complexity of the rotating aperture approach. The large amount of weight (over 300 lbs.) and the precision of the motion associated with the periscope assembly necessitate a careful mechanical design analysis. For instance, the bearing and drive assemblies necessary to both support the cantilevered periscope weight and drive it without wobble or jitter will require custom design features. Good drawings of the various subassemblies will be required to allow the fabrication shop to build to the needed specifications. Finally, fabrication costs are high because of the system mechanical complexity. This complexity is increased by the multitude of mirror adjustment mechanisms required to align the system.

Total personal service (direct labor) costs are not only large because of the large amount of engineering and fabrication labor required for this system but are also large because of the burden rate applied to direct labor costs. The typical commercial burden rate, approximately a factor of 3, is multiplied times the direct labor charge to determine total labor costs. The labor burden rate is composed of overhead, G & A, and profit in approximately the proportions indicated in Table 5-1. Final labor cost to the simulator program is an estimated $274,800. At 80% of the total cost, this is by far the dominant cost contributor with the rotating aperture strawman design.

Table 5-1. Total cost summary

COST CATERGORY	AMOUNT	BURDEN RATE*	COST
Direct Labor	$ 91,600	3	274,800
Optical Parts	40,805	1.34	54,679
Mechanical Parts	10,650	1.34	14,271
		Total	$343,750

*Labor: Overhead (2.0) x G&A (1.34) x Profit (1.12) = 3.0

Parts: G&A (1.34)

Table 5-2. Direct labor cost breakdown

Task	Duration (mm)	Rate (mm)	Amount
Config. Trade Study	2	$2,100	$ 4,200
Optical Design	3	2,100	6,300
Mechanical Design	5	2,100	10,500
Servo Design	1	2,100	2,100
Drafting	10	1,600	16,000
Fabrication			
Eng Liaison	3	2,100	6,300
Machinist/Technician	18	1,600	28,800
Management	6	2,900	17,400
		Total	$ 91,600

The remaining 20% of the simulator cost is optical and mechanical parts cost. Table 5-3 shows the expected mechanical parts list and approximate parts cost. The items on this list are representative of the parts required and, due to their small percentage of the total cost, offer little opportunity for savings.

An optical parts list, including radiation sources, is given in Table 5-4. These costs were found to be representative from conversations with vendors. They are not necessarily the lowest costs available for that item. Therefore, some savings could be expected and the total optical parts cost shown is believed to be conservative. Again, however, the relatively low percentage of the total cost represented by parts precludes much savings by cost reduction efforts in this area. Nevertheless, some savings could accrue if RADC purchased the more expensive radiation sources since these parts are burdened by a factor of 1.34, as shown in Table 5-1. Thus direct RADC purchase would result in a theoretical 34% savings on the items affected. Such savings, however, might rapidly disappear if integration difficulties were later encountered. The potential savings obtained through separate parts purchase is not recommended.

5.2 Strawman Evaluation

The strawman design and, therefore cost, result from the goal to design and fabricate a simulator which meets all the requirements in section 2.0. As noted previously, the requirements in section 2.0 were not generated from a consideration of design difficulty or cost impact, but rather they were generated from an examination of test requirements. It is to be expected, then, that a strawman design configured to meet these requirements would not prove to be cost effective. A better design would offer the cheapest approach possible, compromise on the less important requirements, and still meet most of the essential requirements.

The strawman design is deficient in several areas. The prime deficiencies are high cost, high risk, and excessive size and weight. Cost is relative; for the strawman cost to be judged high implies that reasonable

Table 5-3. Mechanical parts cost

Quantity	Description	Approx. Cost
2/ea.	Kaydon KGZ00XP0 20" bore bearing	$ 3,000
2/ea.	Kaydon KD060XP0 6" bore bearing	500
1/ea.	SLO-SYN M061-FC08 stepping motor	250
1/ea.	SLO-SYN BA1800-3 preset indexer	600
75 ft.	1 1/2 in. x 1/12 in. x 1/4 in. 6061-T6 AL. angle	150
10 ft.	6 in. O.D. x 1 in. wall 6061-Y6 AL. tubing	450
10 ft.	8 in. O.D. x 1/2 in. wall 6061-T6 AL. tubing	350
1/ea.	.063 x 48 x 144 in. 6061-T6 AL. sheet	150
2/ea.	1/2 x 48 x 144 in. 6061-T6 AL. plate	1,500
	Miscellaneous hardware	600
1/ea.	Custom made gear for drum	3,600
1/ea.	Worm & worm gear	100
	Total	$10,650

Table 5-4. Optical parts cost

Description	Approx. Cost
Laser (ILS model NT-114, + flash lamp simmer)	$ 18,600
Nd filter wheel (6 in. dia., OCLI custom 3-13.5 μm)	5,500
Nd filter wheel (6 in. dia., OCLI stock 0.38-3.0 μm)	200
Off-axis parabolic mirror, 16 in. dia.	10,000
IR source (IR Industries, Inc. model 463 w/controller-rack mount)	3,365
Projection lamp & controller	850
9-inch mirror	50
30-inch mirror	900
30-inch mirror	900
6 ND filters	440
Total	$ 40,805

compromises in design requirements can be made to significantly reduce that cost. A number of promising compromises are discussed in the next section. Risk is high because of the large amount of weight cantilevered on the periscope rotation bearing. This weight increases the risk that design difficulties will be encountered. Such difficulties could delay delivery and add significantly to the cost. In addition the design is risky because the system is large and requires large optical elements. These elements generally require custom fabrication efforts and long lead times. Damage of any element in shipment or during system assembly could seriously delay the program. Finally, excessive weight is a deficiency simply because it requires a modular design approach to assure portability. Such portability, though never fully attainable, increases system cost and complexity.

6.0 ALTERNATE REQUIREMENTS

Relaxed design requirements can result in reduced simulator costs. Moreover, they can make the various simulator configurations look more attractive in terms of size and complexity. This section examines the impact of reduced requirements on system utility. It attempts to identify the changes which should be made to reduce cost without unnecessarily sacrificing system performance.

6.1 Reduced Field-of-View (FOV)

The most obvious and, on the surface, promising design requirement change to reduce cost is the relaxation of the required FOV. Figure 6-1 shown the reduction in required collimator aperture to be achieved by reducing the FOV or moving closer to the seeker. For instance the present 2° FOV requires an aperture of 16 inches when the collimator to seeker distance is approximately 29 ft. A reduction in the FOV to 1° and a reduction of the collimator-seeker distance to 25 ft would require only an 8 inch aperture instead of the 16 inch aperture in the strawman design. Thus these changes would cut the strawman design size in half and possibly the cost as well. From section 2.3.12 it is seen that the FOV could probably be reduced to as little as ½ degree if track rates were held to mid-course guidance levels. Furthermore, the seeker-collimator distance could be reduced 5 ft by cantilevering the simulator platform. The effect of such changes on cost is worth examining.

The cost savings to be obtained through a factor of two size reduction is summarized in Table 6-1. It is seen that no more than a 15% overall savings can be realized. Parts costs do not change significantly since only a few optical components benefit from a large savings. Direct labor charges are reduced by only 30%. This is because many of the design and fabrication tasks remain the same, even though the size of the structure is halved. For example, all the engineering tasks in Table 5-2 must still be accomplished with the smaller system. Likewise, while fabrication is made easier with a smaller structure, a technician must go through

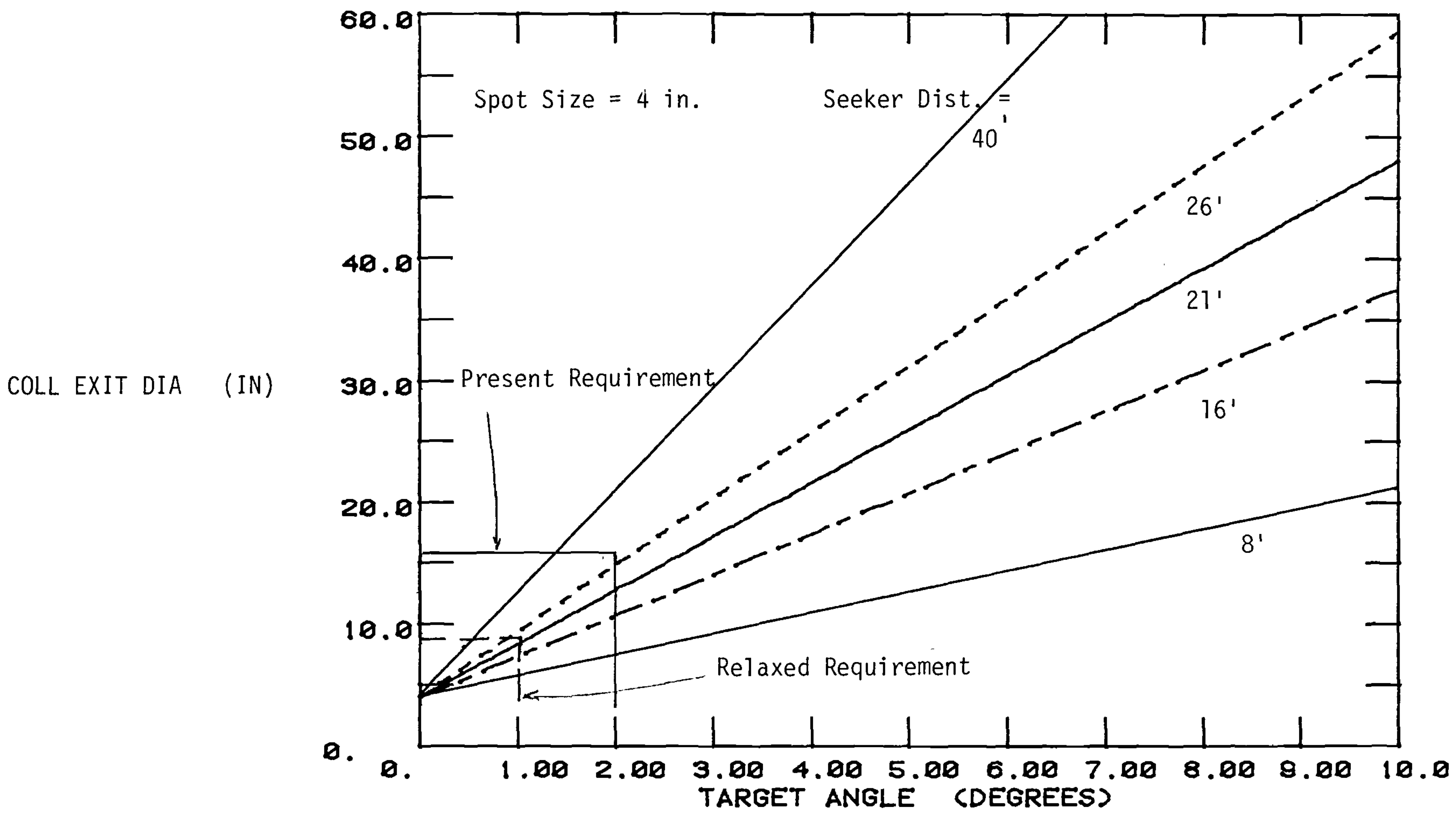


Figure 6-1. Required collimator aperture size.

Table 6-1. Cost savings with 8 in. system design

Cost Category	16 in. System	8 in. System	Cost Difference	Burden Rate	Savings
Optical Parts	40,805	33,555	7,250	1.34	$ 9,715
Mechanical Parts	10,650	8,520	2,130	1.34	2,854
Personal Services	91,600	82,340	9,260	3.0	27,780
				Total Savings	$40,349
			Total 8 in. sys. cost		$303,400

the same steps and spend largely the same amount of time to cut an 8 foot aluminum member as he would a 4 foot member. Perhaps the real cost savings with the smaller 8 inch aperture system is a hidden cost savings - the reduced potential for overrun offered by a lower risk design.

It is concluded that size reduction alone, achieved by reducing essentially only one design requirement, will not produce significant savings. All of the requirements which affect the configuration should be re-examined with the objective of balancing each requirement with its potential configuration impact.

6.2 Requirement Change Options

The various requirement change options are listed in Table 6-2 together with the present requirement, in parentheses, from Table 2-2. The changes indicated in the table could potentially reduce system cost. Table 6-3 shows the advantages and disadvantages of these options. As discussed above, reduction of the unvignetted FOV reduces the collimator aperture requirement and, therefore, reduces overall system size. A change to $\frac{1}{2}^{\circ}$ would confine utilization to mid-course guidance rates during susceptibility testing, but such confinement is still consistent with the EMI test scenario and is, therefore, recommended.

Lowering the motion amplitude requirement also offers the advantage of reducing the required collimator aperture, as does reducing the required spot diameter, and moving closer to the seeker. The motion amplitude could be reduced to $\pm 1^{\circ}$ at the risk of reducing the signal over noise measurement margin factor referred to in section 2.0. This change is recommended since a 1° amplitude still leaves a nominal margin factor of 10. In addition, the reduced motion amplitude requirement, together with the reduced FOV requirement, makes the stationary aperture design approach, discussed in section 3.0, appear feasible. The latter approach can become much simpler than the rotating aperture strawman design approach when these key requirements are relaxed.

Table 6-2. Cost reduction options

1) Reduce unvignetted background FOV requirement (2^o)

2) Lower motion amplitude requirement ($\underline{+}$ 2^o)

3) Reduce the unvignetted spot dia. requirement (4 in)

4) Compress spectral coverage interval (0.4-12 μm)

5) Relax collimation accuracy requirement (0.8 mr)

6) Move the target simulator closer (14-24 ft)

7) Remove the requirement for signal calibration (Nd 2)

Table 6-3. Option appraisal

	Option	Advantages	Disadvantages	Comment*
(1)	Reduce FOV	Reduces aper. req'mt.	Potentially reduces utilization	Recommend change from 2° to 1/2°
(2)	Lower motion amp.	Reduces aper.	Makes track signal more difficult to meas.	Recommend change from ± 2° to ± 1°
(3)	Compress spectrum	Not significant	Significantly reduces utility	Not recommended
(4)	Relax collim. acc.	Removes collim.	" "	Not recommended
(5)	Move closer	Reduces aper.	RF interference	Trade vs (2)
(6)	Reduce spot dia.	Reduces aper.	Significantly reduces utility	Not recommended
(7)	Remove calib.	Simplifies focal plane design	SNR cannot be easily determined during test	Not recommended

*These recommendations, together, are expected to reduce costs to < $200,000.

The options of moving closer to the seeker and of reducing the required spot diameter are not recommended because of the reduced system utility that could result. Moving closer would have the same effect on the configuration as lowering the motion amplitude requirement. It is not recommended as an initial design feature because of the RF interference that it might produce. Likewise, reduction of the unvignetted spot diameter at the seeker could cause much test difficulty with seekers locking onto the vignetted background. The expense of wasted test time that this relaxation might cause offsets the potential savings.

All the remaining requirement change options are not recommended either because they suffer from significantly reduced system utility or offer little opportunity for cost savings. For instance the simulator spectral bandpass requirement might be reduced from the 0.4 μm - 13 μm range to the visible and near infrared range, but the wider bandpass can be easily provided with reflective optical components and little cost savings would be found. The collimation accuracy requirement could be relaxed to the point of eliminating the need for collimation at all. This, however, would reduce utility with imaging and reticle seekers. The present 1 mr collimation accuracy requirement is not tight by typical optical component standards and, short of collimator elimination, does not represent a significant cost. Finally, the requirement for a capability to provide calibrated changes in signal level does add significantly to the cost but is believed to be a necessary test parameter.

In summary, many of the design requirements can be relaxed with some increase in the risk of decreased simulator utility. The requirements changes recommended above will reduce the cost significantly below the strawman design cost estimate. Though a revised strawman design is beyond the scope of this study, it is believed that the recommended requirements changes would reduce the simulator cost to well below $200K.

7.0 CONCLUSIONS AND RECOMMENDATIONS

The design requirements developed in section 2.0 together with the recommended changes discussed in section 6.0 will likely provide the most cost effective target simulator. This conclusion is the result of an analysis which considers both test requirements and hardware constraints. The final requirements, however, are based on the expected range of seeker parameters and general seeker operating principles. Yet it is possible that deviations from the nominal assumptions by present inventory or newly developed seekers will cause some difficulties. For this reason it would be desirable to examine these requirements with a working breadboard before a larger sum of money is committed to construction.

The revised requirements offer an opportunity to construct a breadboard simulator with less expense. The primary impact of the recommended design changes is the enhanced attractiveness of the stationary aperture design concept. This concept is now more promising because of the reduced aperture and simplified focal plane design permitted. The concept is particularly attractive because it can be easily breadboarded to provide a test bed upon which to both verify the design requirements and evaluate the stationary aperture design approach.

Figures 7-1 and 7-2 show a sketch of the stationary aperture concept breadboard. This breadboard potentially meets all the recommended requirements except for providing continuously variable neutral density signal control and the higher signal levels. Neutral density signal control and increased signal power could be potentially provided but would require more study. The only moving parts, with the exception of the adjustment mechanisms, is the target wheel. An aperture of selectable diameter can be mounted on the wheel to provide the desired target subtense and motion amplitude. For instance, the f/4 parabolic mirror presents a 64 inch focal length, thus if the aperture is displaced 1.1 inches from the center of the wheel rotation, the peak to peak motion

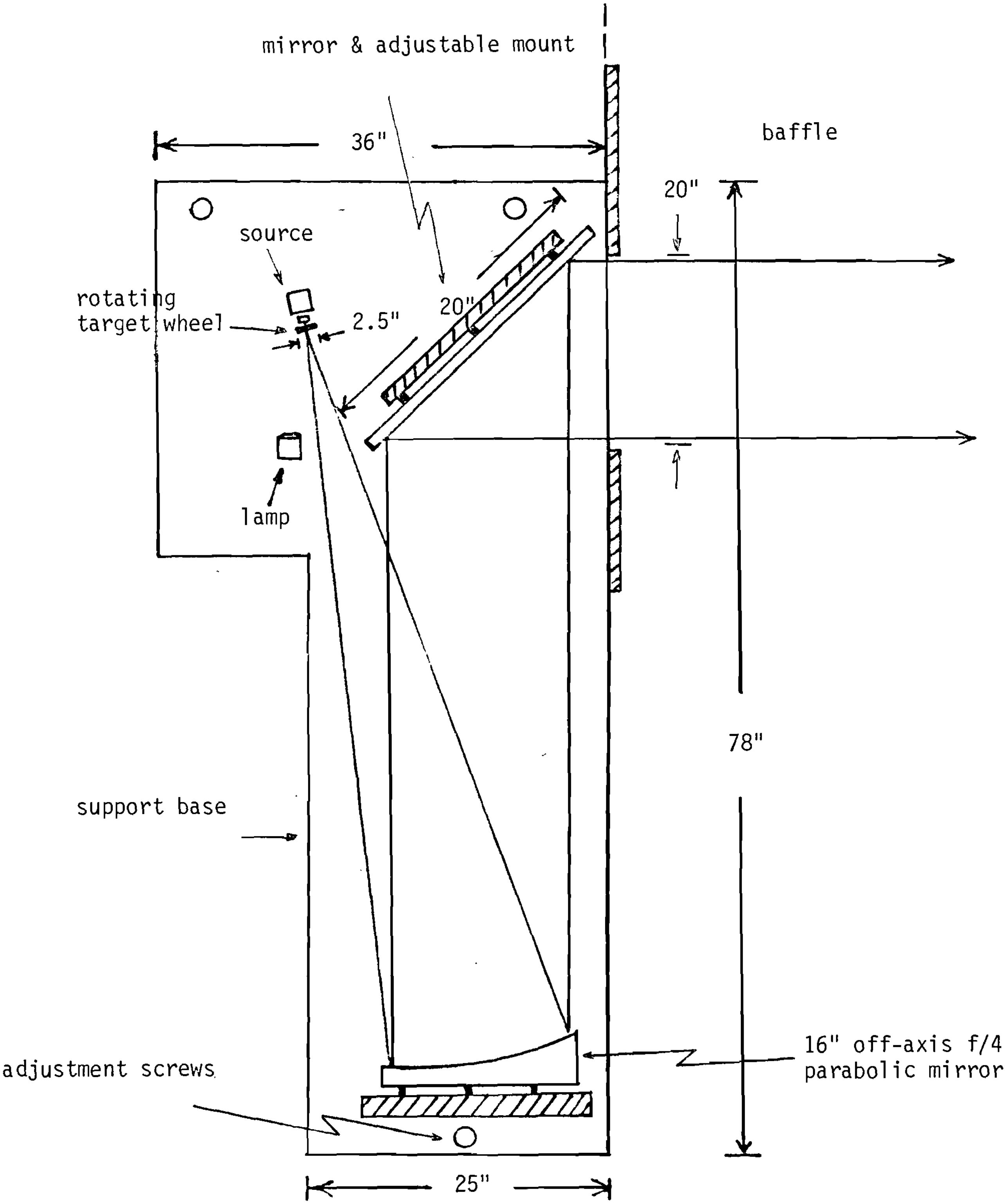


Figure 7-1. Sketch of stationary aperture breadboard simulator

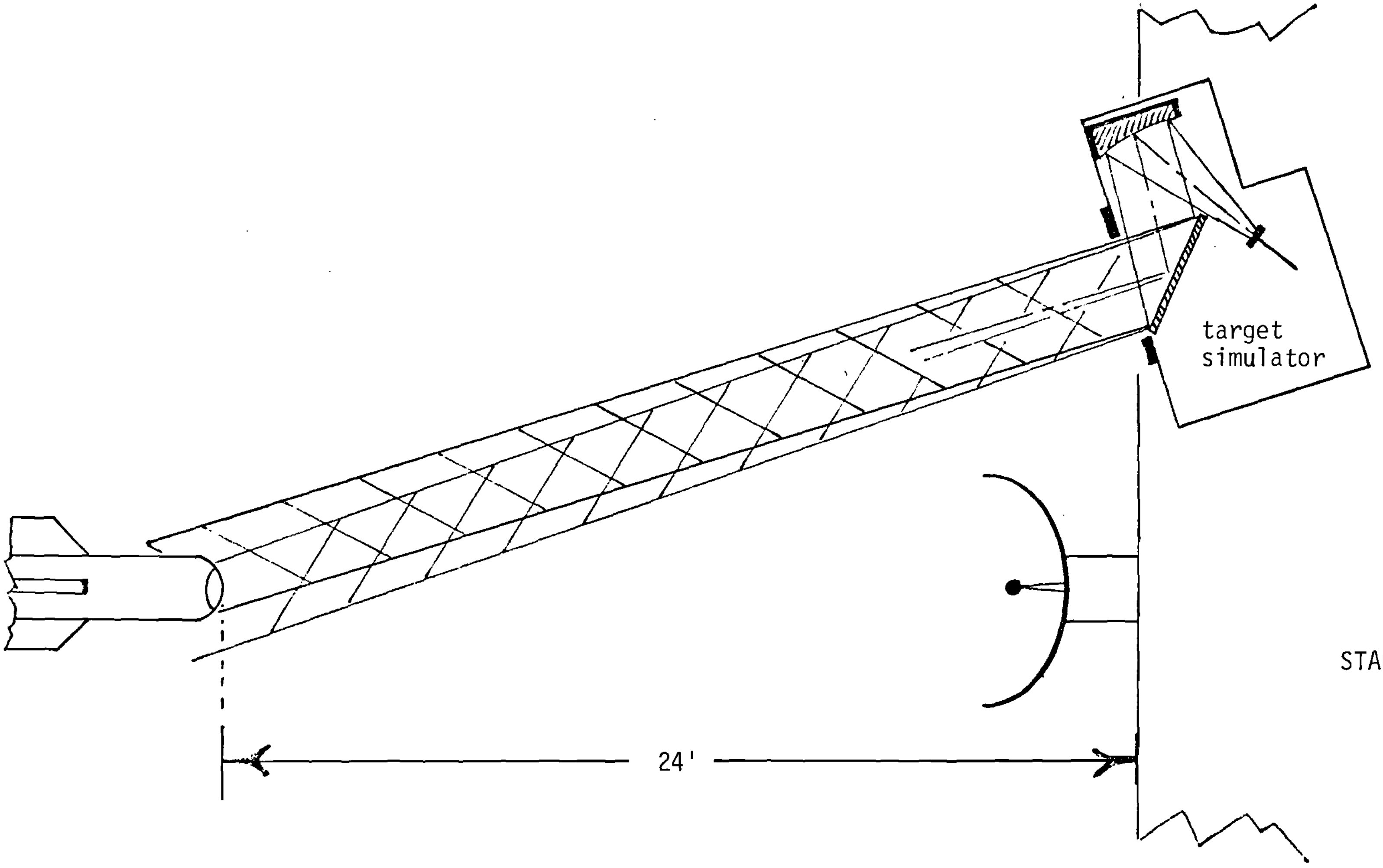


Figure 7-2. Illustration of breadboard simulator in anechoic test chamber

amplitude will be 2.2/64 radians or approximately 2^{o}. However the source extent for the laser, blackbody, and lamp must be large enough to fill the nominal 2.5 inches of target travel, including target extent.

Figures 7-3 and 7-4 show focal plane arrangements for extending the effective source dimensions. For IR target simulation, an extended blackbody cavity with a nominal 2.5 inch diameter would suffice. For a laser target simulation, a beam expander would be required. Thus if the source laser had a typical exit aperture of 0.25 inches, a 10x beam expansion telescope could be used to obtain the necessary 2.5 inch source extent. Beam expansion by 10x, however, reduces the power by a factor of 100 and therefore may prevent achieving the higher required irradiance levels.

A primary consideration in pursuing the breadboard approach is its expected cost. In an effort to estimate breadboard costs a preliminary labor and parts cost breakdown is provided in Tables 7-1 and 7-2, respectively. For estimation purposes the commercial labor rates in Table 5-2 are again assumed. Of course the final cost figure will depend upon the burden rate incurred as well as the labor rate. With the approximate 85% burden rate at Georgia Tech, labor costs would be approximately $50,000.* Materials are unburdened, so the total cost to RADC for both labor and materials would be approximately $70-80K. This cost, however, could be reduced if RADC should decide to perform some of the tasks in their facility.

In conclusion, a target simulator which meets all the test requirements would be very costly, but it is believed that a still effective simulator can be built for much less. Without a much more vigorous analysis it is difficult to determine the exact compromises to make. Here a breadboard approach offers several advantages. First, it provides an opportunity to examine both the requirements and the design in actual usage. Secondly, it is an interim simulator that can be used for full scale testing without delaying important test programs with a more elaborate procurement effort. Finally if, as expected, the breadboard simulator meets most of the testing requirements, it might be refined into a final configuration without the need for a follow-on procurement.

---

*does not include reports or trips

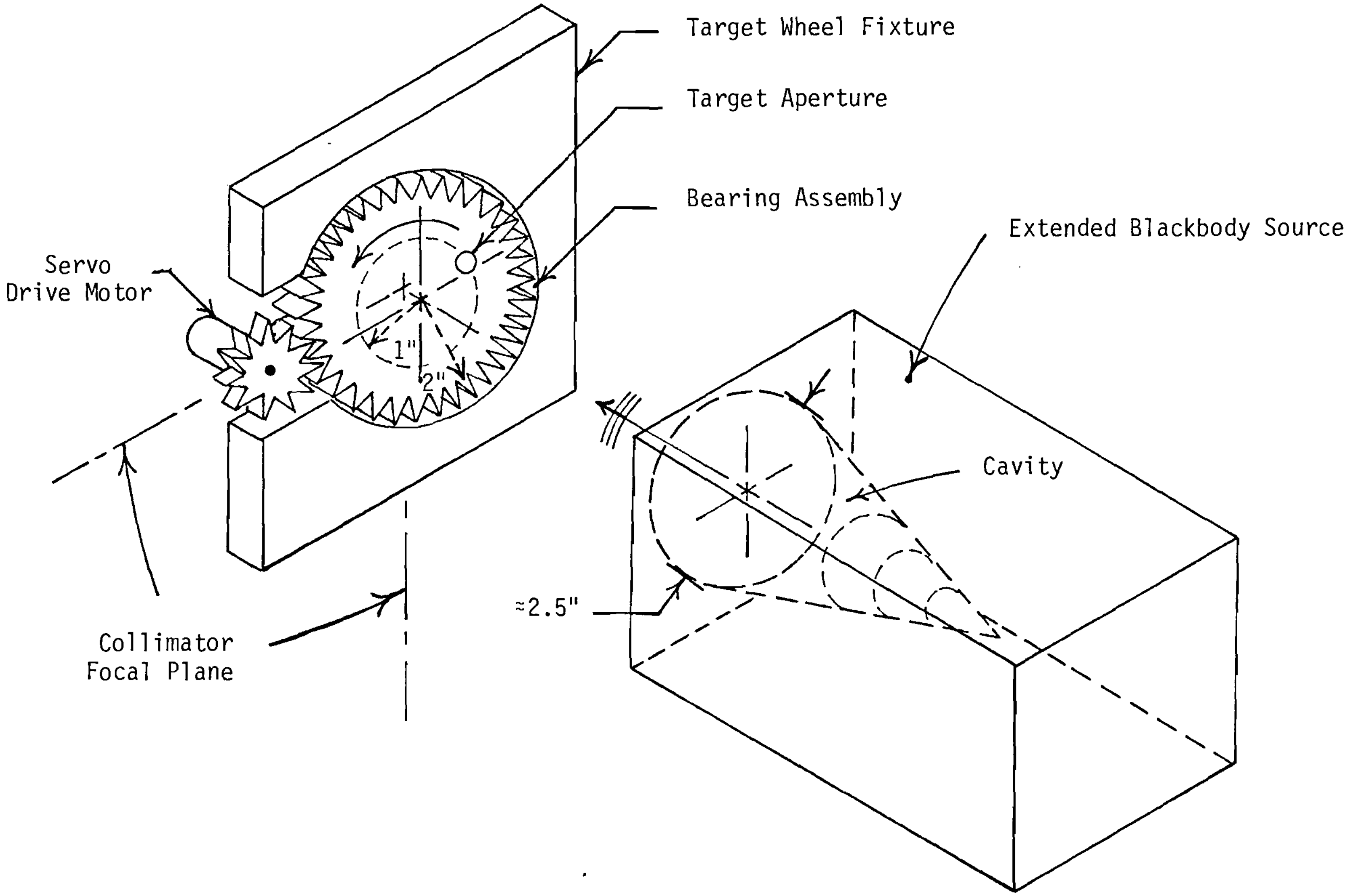


Figure 7-3. Sketch of target wheel with extended blackbody radiation source

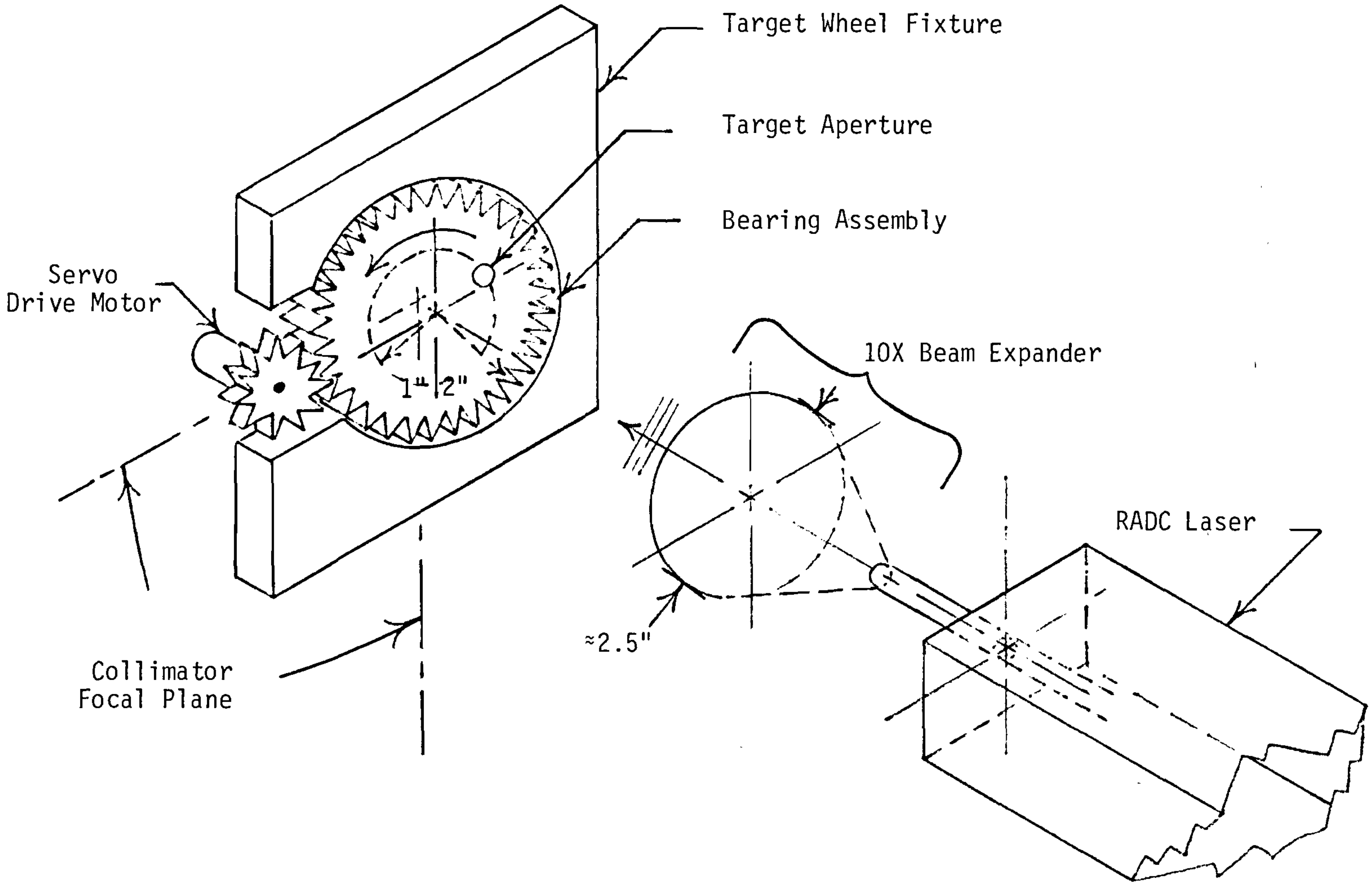


Figure 7-4. Sketch of target wheel with laser source and beam expander

Table 7-1. Preliminary estimate of unburdened direct labor costs

TASK	DURATION (MM)	RATE ($/MM)	AMOUNT ($)
Optical design	3.0	$ 2,500	$ 7,500
. analysis			
. ray trace			
. vendor search			
. parts order			
. fab liaison			
. alignment			
. test			
Mechanical design	1.5	2,100	$ 3,150
. target wheel layout			
. parts selection/order			
. liaison			
Drafting	2.0	1,600	$ 3,200
Servo Design			
. design approach	0.8	2,100	$ 1,680
. parts selection/order			
. liaison			
Fabrication	2.5	1,600	$ 4,000
Management	2.0	2,900	$ 5,800
		Total	$25,330

Table 7-2. Preliminary parts estimate

DESCRIPTION	APPROX. COST
16 in. off-axis parabolic mirror	$10,000
20 in. dia. flat mirror	900
stock metal	600
servo motor & controls	250
parabolic mirror mount	2,600
flat mirror mount	1,500
misc. hardware	600
projection lamp & pwr. supply	850
extended blackbody	3,500
laser beam expander	1,200
Total	$22,000

APPENDIX A - ROTATING APERTURE CONFIGURATION GEOMETRY

This appendix outlines the analytical relations and geometrical constructions relevant to the optical design of the circularly "rotating aperture" concept for the target simulator.

It will be assumed that the extended object (target source) is located perfectly in the focal plane of an ideal optical collimating system.

A.1 Source Imaging

The requirement of achieving an unvignetted target image for adequate seeker lock-on necessitates a detailed analysis of the trade-offs involved upon imaging an extended source by collimating optics.

Figure A-1 geometrically illustrates the basic optical layout required for a beam impinging on the seeker. In conjunction with this figure, Table A-1 defines the symbols used in the geometrical and analytical analysis.

By definition, a collimator will image a point object located on the optical axis exactly at the focal point into a perfectly collimated beam with a diameter of the collimating device. Figure A-1 illustrates the method by which an extended circular object is imaged. Ray tracing the extremities of the object shows a resulting diverging beam produced by the collimator. The intersection of the two collimated beams derived from two extreme points on the object defines a conical region in the image space where an unvignetted image may be formed. In Figure A-1 the cross-section of this conical region appears as the triangle of base d_c and height x_c.

The following analytical relations are easily derived from the geometry.

→(*i*) Relation between the Field of View (ϕ), Object Diameter (d), and Focal Length (f_c).

$$\tan (\phi/2) = \frac{d}{2f_c} \qquad \text{(A-1)}$$

Table A-1. Symbol Key for the Single Beam Geometry

POINTS ALONG THE OPTICAL AXIS

L: Collimator Focal Plane
M: Collimator Fold Mirror
N: Collimator Exit Mirror (Paraboloid)
O: Physical Collimator Exit
P: Periscope Entrance Mirror
Q: Periscope Exit Mirror
R: Point at Unvignetted Image of Diameter d_{st}
C: Unvignetted Imaging Limit

PARAMETERS

d: Collimator Source Aperture Diameter (in focal plane).

f_c: Collimator Focal Length ($\overline{LN}$)

d_c: Collimator Exit Aperture Diameter.

ϕ: Field-of-View

x_b: Distance from Collimater Exit to Point Q along the optical axis ($\overline{NQ}$).

b: Total Beam Diameter perpendicular to the optical axis at distance x_b.

x_s: Distance from Collimator Exit to Point R along the optical axis ($\overline{NR}$).

d_{st}: Unvignetted Image Beam Diameter perpendicular to the optical axis at distance x_s.

x_{sb}: Distance $x_s - x_b$ or ($\overline{QR}$). (Periscope Exit Mirror to Unvignetted Image Distance).

x_c: Distance from the Collimator Exit to Point C along the optical axis ($\overline{NC}$). (Maximum Unvignetted Image Distance)

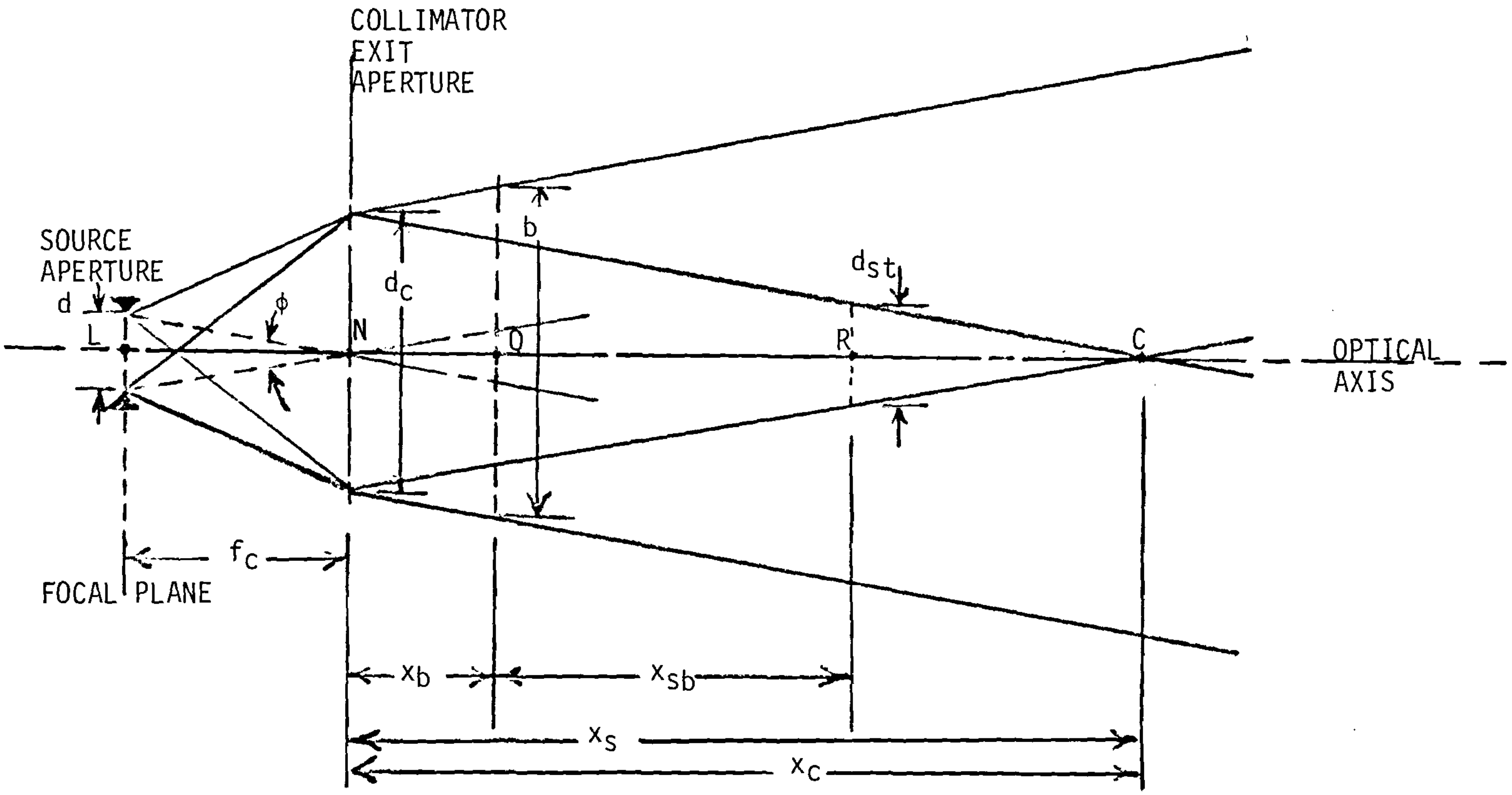


Figure A-1. Extended object imaging by a collimator

Upon rearrangement, equation (A-1) yields:

$$\phi = 2\tan^{-1}\left(\frac{d}{2f_c}\right) \quad \text{(A-2a)}$$

$$d = 2f_c\tan\ (\phi/2) \quad \text{(A-2b)}$$

$$f_c = \frac{d}{2\ \tan\ (\phi/2)} \quad \text{(A-2c)}$$

For a $\phi \leq 20^o$ (= 349 mr), one may use the approximation $\tan(\phi/2) \simeq \phi/2$ (in radians) which results in less than 1% error. Equation A-1 then yields the approximate relation:

$$\phi = \frac{d}{f_c}\ ;\ d = \phi f_c\ ;\ \text{or}\ f_c = \frac{d}{\phi} \quad \text{(A-3)}$$

→ (*ii*) Relation between the Field of View (ϕ), Collimator Exit Diameter (d_c), and Maximum Unvignetted Image Distance (x_c).

$$\tan(\phi/2) = \frac{d_c}{2x_c} \quad \text{(A-4)}$$

Rearranging equation (A-4) yields:

$$\phi = 2\tan^{-1}\left(\frac{d_c}{2x_c}\right) \quad \text{(A-5a)}$$

$$d_c = 2x_c\tan\ (\phi/2) \quad \text{(A-5b)}$$

$$x_c = \frac{d_c}{2\ \tan\ (\phi/2)} \quad \text{(A-5c)}$$

Again, for $\phi \leq 20^o$ (= 349 mr), equation (A-4) may be approximated as:

$$\phi = \frac{d_c}{x_c}\ ;\ d_c = \phi x_c;\ \text{or}\ x_c = \frac{d_c}{\phi} \quad \text{(A-6)}$$

→ (*iii*) Relation between d, d_c, f_c, and x_c.

Equating the RHS's of equations A-1 and A-4 yields the exact ratio:

$$\frac{d}{f_c} = \frac{d_c}{x_c} \quad \text{(A-7)}$$

→ (*iv*) Relation between the Field of View (ϕ), Unvignetted Image Diameter (d_{st}), Distance to the Unvignetted Image (x_s), and Maximum Unvignetted Image Distance (x_c).

$$\tan(\phi/2) = \frac{d_{st}}{2\ (x_c - x_s)} \quad \text{(A-8)}$$

→ (*v*) Unvignetted Image Diameter at x_s: $d_{st} = d_{st}(x_s, d_c, \phi)$

Equations A-5c and A-8 yield:

$$d_{st} = d_c - 2x_s \tan(\phi/2), \qquad \text{(A-9a)}$$

and the approximate expression:

$$d_{st} = d_c - x_s \phi. \qquad \text{(A-9b)}$$

→ (*vi*) Collimator Exit Aperture: $d_c = d_c(x_s, d_{st}, \phi)$

Equation A-9a yields:

$$d_c = d_{st} + 2x_s \tan(\phi/2) \qquad \text{(A-10a)}$$

along with the approximate expression $d_c = d_{st} + x_s\phi$ (A-10b)

→ (*vii*) Beam Diameter at x_b: $b = b(x_b, d_c, \phi)$

$$\tan(\phi/2) = \frac{b-d_c}{2x_b} \qquad \text{(A-11)}$$

which yields:

$$b = d_c + 2x_b \tan(\phi/2) \qquad \text{(A-12a)}$$

or the approximate expression:

$$b = d_c + x_b\phi. \qquad \text{(A-12b)}$$

A.2 Beam Crossover Geometry

The circularly rotating aperture configuration is based on rotation of the resulting beam in the object space of the collimator in order to simulate a moving object (target and background). The optical design of the simulator must be such that the seeker optical system is able to construct an unvignetted image of the original object regardless of any physical motion of the simulator and the seeker optical system within its gimbal assembly. To achieve this requirement within this configuration, an analysis of the image beam crossings must be undertaken.

Figure A-2 illustrates a cross-section of two extreme image beams in a plane of the spin axis, each of which is generated as the target simulator traverses its circular path in the plane normal to the spin axis. The periscope exit mirror, at Point Q, is adjusted so that the optical axis of the image beam intercepts the spin axis at Point Z forming the Beam Convergence Angle Θ. Table A-2 provides an additional symbol key, to be used

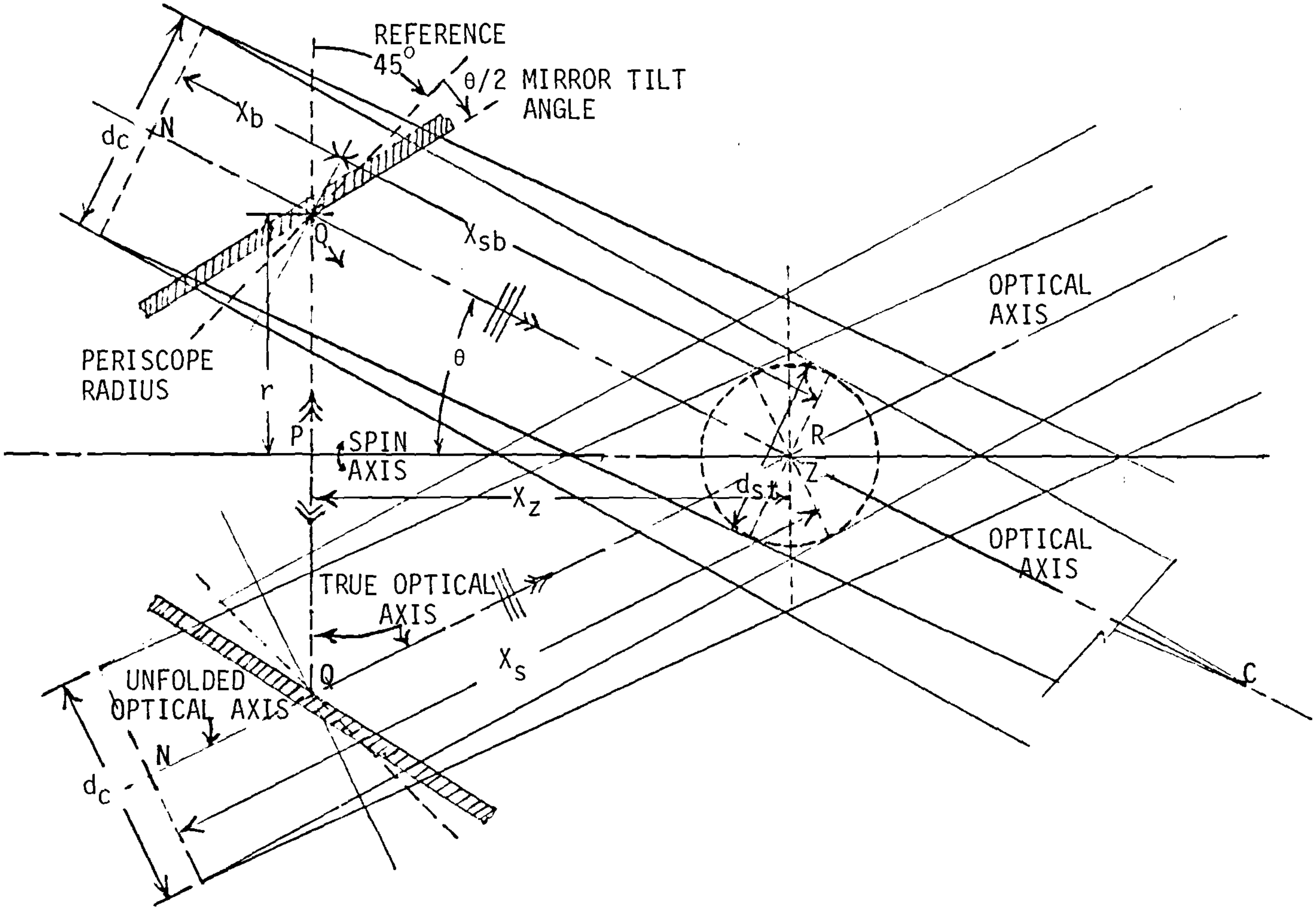


Figure A-2. Crossing beam geometry - unvignetted region (cross-section)

Table A-2. Additional Symbol Key for the Crossing Beam Geometry

POINTS ALONG THE OPTICAL AXIS

Z: Beam Crossover Point

PARAMETERS

x_Z: Distance to Beam Crossover Point Z along the Spin Axis ($\overline{PZ}$)
r: Periscope Optical Radius Arm Distance ($\overline{PQ}$)
θ: Beam Convergence Angle (QZP)
$\frac{\theta}{2}$: Mirror Tilt Angle

in conjunction with Table A-1 for this description and Figure A-2.

Provided that the unvignetted region of the beam, as described in section A.1, extends further than the spin axis crossover point (i.e. $\overline{NC} > \overline{NZ}$), a volume in space will be generated in which an unvignetted image may be formed at any time during the beam motion. The cross-section of this region is the innermost quadrilateral around Point Z in Figure A-2. Requiring the seeker optics at all times to be within this volume simplifies the optical and mechanical design of the simulator.

This analysis concerns itself with the special case, as illustrated in Figure A-2, requiring the center of the seeker body to be fixed at Point Z. Point R is considered coincident with Point Z so the geometry of section A.1 can be used in the analysis. The seeker diameter plus positional tolerance is considered to fill the diameter d_{st}. Therefore, the collimator exit-to-seeker distance is x_s ($\overline{NR}$), the periscope mirror-to-seeker distance is x_{sb} ($\overline{QR}$), and the periscope axis-to-seeker distance is x_z ($\overline{PZ}$).

It is observed that a precise analysis of the crossover region dimensions is not necessary when the design is carried out using the maximum seeker location distance, x_z, for a particular application, and when d_{st} is defined as above. This is because if x_z is decreased, θ is increased, and the unvignetted region can only increase in dimensions.

The analytical relations that are needed to specify the beam crossing geometry, inferred from Figure A-2 are:

→ (*i*) Relations between the Beam Convergence Angle (θ), Periscope Radius Arm (r), and Optical Seeker Distance (x_{sb}):

$$\sin\Theta = \frac{r}{x_{sb}} \qquad \text{(A-13)}$$

which yields:

$$\Theta = \sin^{-1}\frac{r}{x_{sb}} \qquad \text{(A-14a)}$$

$$x_{sb} = \frac{r}{\sin\Theta} \qquad \text{(A-14b)}$$

$$r = x_{sb}\sin\Theta \qquad \text{(A-14c)}$$

→ (*ii*) Relations between Θ, r, and the Spin Axis Seeker Distance (x_z):

$$\tan\Theta = \frac{r}{x_z} \quad \text{(A-15)}$$

Equation A-15 yields:

$$\Theta = \tan^{-1}\left(\frac{r}{x_{sb}}\right) \quad \text{(A-16a)}$$

$$x_z = \frac{r}{\tan\Theta} \quad \text{(A-16b)}$$

$$r = x_z \tan\Theta \quad \text{(A-16c)}$$

→ (*iii*) Relation between the Mirror Tilt Angle $(\frac{\Theta}{2})$, r, and x_{sb}.

Equation A-14a yields:

$$\left(\frac{\Theta}{2}\right) = \frac{1}{2}\sin^{-1}\left(\frac{r}{x_{sb}}\right) \quad \text{(A-17)}$$

It should be noted that the variation of the placement of a seeker along the spin axis (x_z) dictates a necessary change in the Beam Convergence Angle Θ (for a fixed r) and will result in a change of the radius of the moving circular image received by the seeker optics.

FINAL TECHNICAL REPORT

"INTEGRATED HARDWARE/SOFTWARE TECHNOLOGY PLAN"

Project A-2366

October 1979

Authors

H. W. Denny
J. E. Balsam
L. A. Jackson
W. M. Leach
G. H. Lunsford
J. D. Norgard
R. W. Rice

Prepared for

ROME AIR DEVELOPMENT CENTER
GRFIFFISS AIR FORCE BASE, NEW YORK

Prepared by

ELECTRONICS TECHNOLOGY LABORATORY
ENGINEERING EXPERIMENT STATION
GEORGIA INSTITUTE OF TECHNOLOGY
ATLANTA, GEORGIA

ABSTRACT

A software design philosophy is set forth which is considered to meet the needs of RADC's Automated Data Acquisition and Control System (ADACS). The recommended approach embodies a "Table-Driven Menu" concept which has been successfully applied to EMS/V testing. In the development of the software, it is recommended that strong emphasis be placed on careful definition and design of the software structure prior to the writing of code. In addition to overall software considerations, specific assessments are made of the instrumentation control system, the computer system, and the data analysis and display requirements associated with ADACS.

FOREWORD

This report was prepared by the Electronics Technology Laboratory of the Engineering Experiment Station and by the School of Electrical Engineering, both of the Georgia Institute of Technology. This work was conducted in response to Task N-9-5220 under Contract No. F30602-78-C-0120 of the RADC Post-Doctoral program. The described work was performed under the general supervision of Mr. D. W. Robertson, Director, Electronics Technology Laboratory, and Mr. H. W. Denny, Head, Electromagnetic Compatibility Branch. This report was authored by Mr. Denny, by Drs. W. M. Leach and J. D. Norgard of the School of Electrical Engineering, and by Dr. R. W. Rice, Dr. G. H. Lunsford, Mr. J. E. Balsam, and Mr. L. A. Jackson of the Engineering Experiment Station.

TABLE OF CONTENTS

LIST OF FIGURES

INTEGRATED HARDWARE/SOFTWARE TECHNOLOGY PLAN

1. Introduction

The Electromagnetic Compatibility Analysis Facility (EMCAF) at Rome Air Development Center (RADC) is presently being upgraded through the implementation of an Automated Data Acquisition and Control System (ADACS). A major role projected for ADACS is the support of tests and evaluations of the electromagnetic susceptibility/vulnerability (EMS/V) properties of guided munitions systems.

EMS/V tests and evaluations of munitions systems involve the accumulation and analysis of large quantities of data. Much of this data is obtained through multiple runs of particular test sequences with only small changes imposed in certain important parameters of the exposure environment between runs. While the results of the tests may indicate the general level of susceptibility, they do not, in themselves, necessarily define the level of vulnerability of the weapon to a particular (or actual) EM environment. The test results must be further evaluated against a threat scenario to assess the likelihood of mission success of that weapon in that environment. This vulnerability assessment process requires extensive handling and manipulation of data--a very time consuming job when done manually.

An extensive complement of equipments--RF sources, antennas, transmission line components, monitoring devices, and recording instruments--are being integrated into ADACS. Wherever possible, equipments compatible with IEEE Standard 488* have been obtained to facilitate automation of the testing. To control these equipments and to support the data analysis needs of EMS/V assessments, a Hewlett-Packard Model 1000/45 (2117A Computer (F-Series)) System with several ancillary equipments (plotters, terminals, printers, etc.) is being purchased. The next planned step in the development of ADACS is the preparation of the software for this computer system that is necessary to support the acquisition and processing of the EMS/V data. In particular, RADC is seeking software that will perform at least the following set of functions:

---

*IEEE-488-1975, "Digital Interface for Programmable Instrumentation."

(1) Coordinate the definition, preparation, and conduct of EMS/V activities in connection with data acquisition, equipment control, and testing of munitions systems in an interactive manner.

(2) Supervise the actual taking of measurement data and provide the means for appropriate data reduction, analysis, graphic presentation, and information display using both interactive dialogue between test engineer and computer and stored program procedures.

(3) Support the continued growth and evolution of EMS/V testing procedures in the direction of greater automation, increased measurement capability, and enhanced archival techniques for both saving and utilizing information obtained from current and previous munitions-system testing.

As a precursor to this software development step, a study effort was undertaken by the Georgia Institute of Technology, under Contract F30602-78-C-0120, to develop a comprehensive technology plan for a totally integrated hardware/software system for weapons system testing in the EMCAF. The object of this effort has been to examine the EMS/V test procedures performed on munitions systems at RADC in an effort to define the software system requirments for ADACS.

Participating in this study were Dr. W. M. Leach and Dr. J. D. Norgard of the School of Electrical Engineering and Mr. H. W. Denny, Dr. R. W. Rice, Mr. L. A. Jackson, Dr. G. H. Lunsford, and Mr. J. E. Balsam of the Engineering Experiment Station. Also participating in early discussions on the program were Mr. D. E. Clark and Mr. S. P. Zehner of the Engineering Experiment Station. (A companion study effort on the development of specifications for a targeting system for weapons system testing was conducted in parallel to this software study. That effort was performed in a separate document.)

During the course of the study, several meetings were held with Mr. G. J. Genello and Mr. C. M. Blank of RADC and Mr. Charles Sykes and Mr. Richard Robeson of Atlantic Research Corporation (ARC). During these meetings, RADC set forth those functional features required (and desired) in the ADACS software and ARC defined the test methodology, the test procedures, and the post-test data analysis requirements.

The experiences and opinions of other military organizations involved in weapons system EM evaluations were obtained through visits to the Naval Surface Weapons System's Dahlgren (VA) Laboratory, the Missile Research and Development Command in Huntsville, AL and the Electronics Warfare Vulnerability Analysis Facility (EWVAF) at White Sands Missile Range, NM. (Trip reports on these respective visits are provided in Appendix A.) In addition to these discussions, a visit was made to Scientific Atlanta (SA) where project personnel were briefed on the organizational aspects of the software for the SA Series 2020 Antenna Analyzer. Several contacts were also made with Hewlett-Packard to obtain specific information on the 2117A computer, its RTE-IV Executive Software, and related data manipulation and output hardware/software.

From these discussions and with support from independent investigations, an assessment of the software needs to support and achieve the objectives of ADACS was made. An approach to the realization of the software was formulated. The results of this assessment and the attendant software-development approach are set forth herein. Section 2 of this report discusses general attributes deemed desirable in the ADACS software; Section 3 sets forth some procurement aspects that RADC should consider; and Section 4 defines the general program schedule that is recommended. Section 5 presents the overall conclusions and recommendations developed during this study.

The body of the report is supplemented with seven (7) appendices. Appendix A contains written reports on four visits to other agencies involved in weapons vulnerability assessments. Appendix B provides a general overview of EMS/V testing in RADC's EMCAF. Pertinent features of the HP 1000/45 system are assessed in Appendix C; generic data analysis and display requirements/capabilities are discussed in Appendix D. Appendix E describes the approach being taken by WSMR toward automating their facility. Current DoD thinking on the development of software is summarized in Appendix F. An illustrative Statement-of-Work is contained in Appendix G.

2. General Software Features

Air-launched ordnance systems, i.e., air-to-air and air-to-ground guided missiles and guided bombs, can experience perturbations in performance from high level electromagnetic (EM) fields. Under certain circumstances, such perturbations can seriously jeopardize the chances for successful completion of the system's mission. A primary purpose of ADACS is to help identify and evaluate those sets of conditions threatening mission success so that appropriate corrective measures can be taken prior to deployment. This identification and evaluation process incorporates an extensive testing program to identify those EM conditions likely to cause system perturbations and an assessment program to determine the impact of the perturbations on missile performance in an expected (or typical) threat environment.

A total EMS/V assessment of a system involves an ensemble of evaluations against the threat environments most likely to be encountered by the system when operational. The evaluations (measurements and analysis of effects on mission) may utilize either an open-loop or a closed-loop approach.* (The approach represented by ADACS is modified open-loop.) The open-loop process is one in which the results or outputs of the process do not exert any direct, immediate influence on the inputs to the process. For example, Figure 1 illustrates that no direct feedback path exists between the output performance data and the inputs provided by the sources and targets. However, an <u>indirect</u> path does exist between the output and the input in that upon evaluation of the output data changes can be made in source/target parameters and test reruns made. In particular, note the dashed line in Figure 1 from Performance Data to Sources/Targets. It is the presence of this feedback loop between the resultant output data and the target/source that transforms the open-loop method of testing to a modified closed-loop procedure with adaptive, interactive (albeit delayed) operations. (Depending upon the sophistication of the control system software and equipment connections, there may be additional feedback loops that can be introduced into this general testing system configuration.) A major feature desired in the ADACS software is that it provide capabilities for rapid post-test analysis of the results so that the output-input feedback path is made easier and that the time lapse for feedback is lessened.

---

*For a detailed discussion of the advantages and disadvantages of the open-loop and closed-loop approaches to EMS/V assessments, see Reference 1.

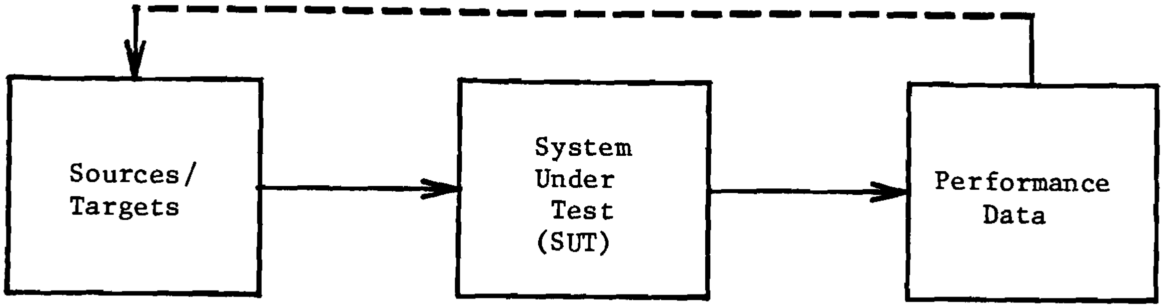


Figure 1. Open-Loop Testing.

Figure 2 depicts the overall vulnerability assessment process which ADACS is to support and into which it is to be integrated. The major elements of this process are System Study, Preliminary Measurements, EM Measurements, Measurement Data Analysis, Flight Simulation, Simulation Analysis, EM Environment, Analysis of Vulnerability, and Final Reporting. ADACS is expected to play a role in each of these elements with the possible exception of Flight Simulation. (Flight Simulation is expected to be performed primarily by the weapon's system developer who will need inputs from the EMI tests and will provide outputs to Simulation Analysis or to Analysis of Vulnerability, depending upon whether the system developer or RADC performs Simulation Analysis.)

The System Study phase is directed to a basic operational analysis of the particular weapon system to be tested. The activities that occur in this phase consist of making the initial examination of the munitions system, studying diagrams and blue prints relating to how it is built, determining its functions, and attempting to identify

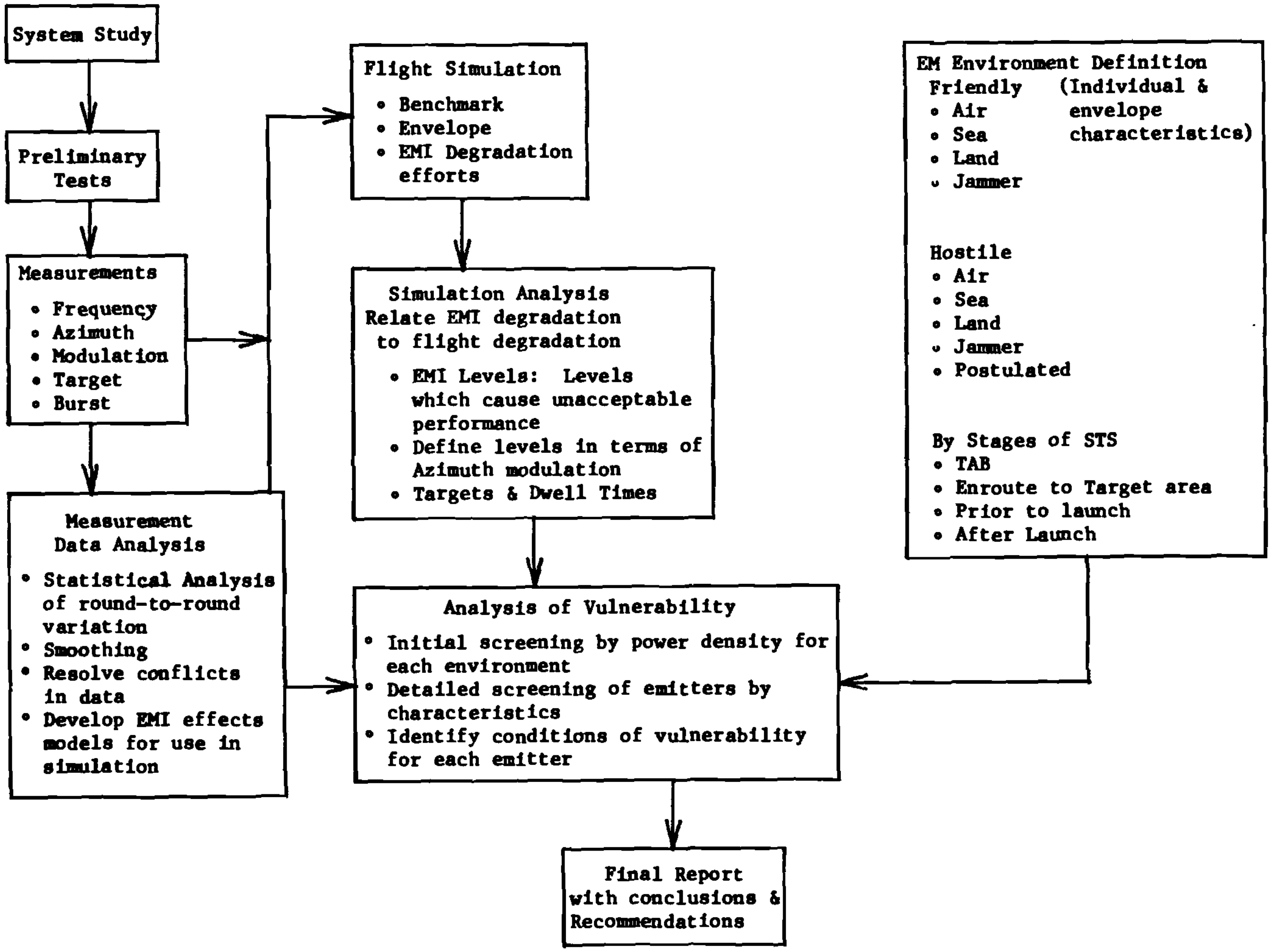


Figure 2. Vulnerability Assessment Process.

those points/areas that may be most susceptible to anticipated hostile threats. A specific investigation of the guidance and control circuitry is made in order to identify potentially susceptible components and devices and to detect logical entry points where unwanted energy may couple into the system. Sensitive indicators (i.e., Standard Responses - see Appendix B) of potentially disruptive effects, e.g., warhead detonations, erroneous fin commmands, saturation of guidance control circuits, etc., are identified and arrangements are made to appropriately monitor these indicators during the testing. Also during this initial precursory phase of evaluation, emphasis is placed upon defining those unique aspects of the particular missile that require special hardware/ software test script modifications and establishing priorities among the testable set of parameters.

Following the System Study, a series of Preliminary Tests are run to see if indeed EM energy does penetrate the system and produce Standard Responses at the monitoring points. If Standard Responses appear at the selected points, the Preliminary Tests help to bracket the ranges over which the environment must be varied in order to quantify the effects noted. If effects from the EM environment (or from directly injected signals) are not evident at the preselected monitoring points, efforts are made to see if Spurious Responses are detected at other locations. If so, the monitor points are appropriately altered and Standard Responses defined or selected for the new locations.

It is during and following the Preliminary Test phase that the test procedures are altered and refined to be compatible with the unique characteristics of the particular weapon to be tested. Appropriate changes are made in the EM environment exposure sequence and in the data acquisition steps and then the test routine is finalized.

It is likely that the System Study and Preliminary Test phases will identify certain procedures that need to be changed and will suggest the data manipulation and output methods appropriate for characterizing the vulnerability properties of the particular type of system. (Operational personnel at the three testing facilities visited emphasized that "every system is different" and that "there is no such thing as a comprehensive standardized test procedure" that will accommodate the needs and uniqueness of each type of weapons system.) The organization and structuring of the ADACS software must allow for flexibility because it is likely that the testing program (at least the ordering of the tests) and the data analysis procedures will need to be adjusted following the Study and Preliminary Test phases. (Note that the question of manual vs. automation is latent in the performance of the preliminary tests. It is doubtful that a significant degree of automation of preliminary tests is worth the developmental effort in view of the marked individuality of each separate munitions system.)

A primary role of the ADACS software will be to allow for the automation of the Measurements through providing the proper commands to set/adjust the radiating sources in a prescribed order and by reading the appropriate measurement instruments at the proper time. The computer software system can control such activities as setting the Electromagnetic Interference (EMI) source frequency, field intensity, relative antenna/System-Under-Test (SUT) orientation, and target position. With the HP-1000/45, the software can control the testing apparatus in a correct time sequence and provide the graphics display capability necessary to support the adaptive, interactive procedure that is needed and desired in ADACS (see Appendices B, C, and D). (It is not anticipated that automation of the testing procedures will appreciably reduce the time spent in munitions system testing, but rather will significantly reduce the time devoted to data reduction/analysis and graphics/display output. Moreover, the flexibility to perform additional types of testing and analysis in a more interacive, "real-time" environment will be a significant benefit because of the increased degree of automation provided by the system software.)

As discussed in Appendix D, the raw data produced by the measurment process must undergo a variety of analyses, reductions, or alterations such as culling, units conversion, scaling, averaging, curve fitting, combining, etc. The associated manipulations typically tend to be very time consuming when performed manually. Thus, an additional major objective of the software design for ADACS needs to be to provide a machine capability for performing the myriad types of post-test data processing that is integral to the Measurement Data Analysis activity.

Before the Analysis of Vulnerability can be performed, information about future "real world" EM environmental threat sources must be available.

It is expected that the DoD Electromagnetic Compatibility Analysis Center (ECAC) at Annapolis, MD will supply a coarse listing, perhaps containing as many as 1,000 entrees, of possible threat sources upon definition of the operational theater by either RADC or the weapons system developer. The results of the EMI tests usually indicate that many of these possible threats will not in fact be likely to cause any problems. Therefore, initially the ECAC-supplied list will need to be culled and the obvious non-offenders deleted. Typically, the remaining list of possible "true" threats is relatively small. It is this reduced list which is then cross matched with the results of the EM tests/analyses and the Flight Simulation to provide the actual Analysis of Vulnerability. The ability of the ADACS software to search and cull the emitter list and to support the vulnerability analysis is thus highly important.

In addition to these technically-related features of the software, the software system should be sufficiently flexible and accessible so as to allow adaptation to varying situations and to allow immediate access to results. Flexibility is necessary because there is a tendency for every munitions system that is evaluated to be different in its method of operation, in the nature and location of the points-of-entry for EM energy, in the number and location of the monitor points, and in the manner in which a given perturbation affects its operational characteristics. Differences of these types mean that, for greatest efficiency in testing and assessment, the test sequences and data analysis routines should be tailored to fit the given system being evaluated. A single large "canned" program is not likely to exhibit the flexibility needed. This is not to say that standardized routines have no place in ADACS. There are, in fact, many of the individual testing sequences that will not need to change, whatever the system being tested. The software routines controlling such test procedures (and analogous data analysis procedures) could, and should, be standardized. It would be as much a mistake to have no standardized routines as it would be to have a single "canned" program. Nor is it desirable to have the situation wherein the software has to be completely rewritten for each system.

It is felt that flexibility can be achieved while maximizing standardization through the use of a "multi-level" software hierarchy.[*] As the name implies, there would be multiple levels of software in such an approach. First, at the lower levels, there are the canned software subroutines (modules) that perform those basic, standardized, and frequently used, operations such as controlling measurement sequences, data logging, file management, and particular data analyses. Each basic module is highly independent, has a single purpose, and has a specific function to perform. At intermediate levels are those combinations of modules that are assembled into those particular sequences appropriate for the given system currently being evaluated. Examples of intermediate-level routines are those which perform the Frequency Scans, Modulation Scans, etc., that are discussed in Appendix B. Collectively, all of the modules form a system library which contains all of the routines that the system is capable of performing. Then, acting as an overall manager to control and direct operations between the various modules containing the standardized routines would be the Program Executive (not to be confused with the machine-related System Executive (Supervisor) software, which for the HP 2117A is the RTE-IV).

The Program Executive performs the function of tying together (sequencing) the standardized routines (modules) into a specialized program appropriate for the system currently being tested. The Program Executive should be structured to assist and allow the test engineer to select the test sequences needed, appropriately order them according to the needs of a particular system, and then to perform the various data manipulation and analysis chores appropriate for that system.

In concept, the Program Executive would be used as follows: Based upon the needs defined during the System Study and Preliminary Test phases, the test engineer would define in a Test Plan the particular series of tests appropriate for the system. Then, selecting the desired routines from a library, the Program Executive would assemble the routines (modules) into the desired order. The final assemblage would then

---

[*] An appropriate multi-level software is being implemented in the EWVAF at WSMR. A detailed discussion of the EWVAF software approach is presented in Appendices A and E.

become the software program for the testing of that particular system. The Program Execuive would be structured and written initially with the particular software modules, which control tests and perform data manipulation and analysis exercises, occupied by dummy or blank programs, i.e., stubs. By initially leaving all such "active" modules unoccupied, the Program Executive could be constructed to be as comprehensive and flexible as needed yet would remain relatively fixed or "canned." The particular testing needs would dictate which "dummy", or stub, modules would be replaced with active controller/analysis subroutines. For this approach to work, considerable attention must be placed on defining and implementing (and documenting) the interface protocol (linkage) between the Program Executive and the operating modules. Each programmed algorithm needs to conform to a basic skeletal structure that will meld easily into the overall ordering of the software development. The software system should be organized so that the treelike structure, which depicts transfers of control in program flow among various routines, is readily reducible to flow charts and to natural function logic paths.

This suggested concept of a multi-level software hierarchy for ADACS implies the need for an interactive capability which allows the user to make changes in the selection and/or ordering of the operational subroutines. (The Program Executive and the software for the operational subroutines (modules) would not normally be altered as part of a regular test program. In special circumstances, modifications and refinements of the programming steps would be done only as needed by appropriate software specialists, not by the test engineer or the test operator.) Note that the ability to alter test sequences is a step beyond those routine measures of simply entering the end points (the Start and Stop of a frequency scan, for example), the stepping increments (the dB steps in the power output levels, for instance), or selecting the ranges desired for an output plot.

To assist the test planner in the use of the Program Executive, a listing of the available operational subroutines should be available in the form of a displayed menu. Appropriate items from this menu are assembled by the test planner with the aid of the Program Executive. Parameters, sequencing information, and instructions relating to data types and processing requirements are placed in a table (array) which is stored on a disk file. This table is also prepared by the Test Engineers based on information gathered from the System Study and Preliminary Testing phases. (This design concept is called a "Table-Driven Menu." It has been implemented at the EMVAF facility at White Sands Missile Range and has been found to offer the orderliness and flexibility that is needed in weapon system EMS/V testing.)

Additional general features desired in the ADACS software are that:

(1) It must be written for the HP 1000/45. The computer code should be written in FORTRAN and needs to be constructed in such a way as to minimize processing time, I/O time, and storage requirements.

(2) It must be capable of supporting the initial facility-related testing activities, such as obtaining antenna patterns and gains, making field calibrations, and developing field mapping data.

In summary, the ADACS software is expected to facilitate the feedback of output information derived from the test data to the input of the testing process and, thus, to improve the quality of EMS/V assessments and to permit the assessments to be accomplished in less time. To realize these benefits, the design and development of the ADACS software must seek to realize the following attributes:

(a) It should have considerable built-in flexibility and modularity to promote easy writing and interchange of various modules to reflect the unique demands of each different munitions system tested.

(b) It must control equipments and instruments conforming to IEEE 488-1975.

(c) The software must be capable of accommodating a significant number of data reduction and analysis routines for post-test processing. Many of the routines needed are available from selected sources; they will need to be properly formatted and made compatible with the ADACS software design. Some procedures will require developmental efforts.

(d) An ability to cull certain designated emitters from an ECAC-supplied listing must be achieved.

(e) A hierarchical software, involving three or more levels, appears to offer greatest flexibility, adaptability, and applicability to vulnerability assessments. It will likely incorporate a number of fixed submodules that can be combined (through the aid of an appropriate operating software sytem) in various combinations to perform paticular tasks.

(f) At the lowest hierarchical level, the programs would be relatively static, changing only when a new procedure is changed or modified, etc. At higher levels, however, a capability should exist which assists and permits program and test engineers to adapt the testing sequence to reflect the unique requirements of a particular munitions system. In essence, a new "program" is created for each new system although the elements of the software do not change except in the order of call-up and utilization.

3. Procurement Options

To develop the type of ADACS software having the attributes described in the previous section, RADC has the options of (1) developing in-house; (2) contracting with an organization having the appropriate mix of weapons system evaluation understanding, machine control and IEEE 488 experience, and programming expertise; or (3) contracting out only detailed software programming of specific submodules whose functions have previously been caefully defined and limited. Of these options, the first will likely require the greatest commitment of in-house personnel time (estimates provided upon request by the organizations visited of the amount of effort necessary to develop the software to support testing of this type ranged up to 10 person-years). However, by doing the total definition, development, and implementation in-house, RADC would retain the greatest degree of control over the details of the program and would assure that it met their specific goals. Success by the contractor under Option 2 in meeting RADC's functional performance goals with the software will require that a lot of emphasis is placed upon the goals being <u>clearly</u> and <u>carefully</u> defined by RADC. Considerable interaction should take place between RADC and the contractor during the early phases of the effort to assure that mutual understanding exists as to what the program should be designed to do, what can't be done, what will be very expensive to do, what tradeoffs must be made, where the limits of responsibilities are drawn, etc. (A suggested method for achieving this understanding is set forth in the next section.) It is considered important that the contractor chosen to develop the ADACS program (if this indeed is the option chosen) be thoroughly familiar with the nature and details of weapons systems EM vulnerability assessments and the intricacies of control of instruments via the IEEE 488 bus as well as being qualified to write software statements in an accepted programming language.

The third option represents a level of effort (on RADC's part) somewhere in between that required by the first two options. With this third option, RADC would define all aspects of the problem down to the point where the programming statements are ready to be written by programmers. Here RADC retains more direct influence over the structuring and details of the program itself with an attendant greater involvement (i.e., time) in the development of the program.

It is expected that the final decision will be strongly influenced by the relative availability of funds to pursue each particular option. In view of present workload associated with current systems evaluation activities, Option 2 appears to be the preferred one because it will allow RADC to obtain an operating ADACS software program, with supporting documentation, within a realistic time frame without undue demands of in-house personnel time.

4. Program Schedulihg

In the procurement and development of software[*], Cave and Salisbury [2] emphasize that the problem to be solved must be defined in general terms by the user. The functional requirements must be analyzed and detailed functional specifications developed for the system in a manner which is easily understood by the end user. McHenry and Walston [3]

---

[*] For additional information, a summary of the points and concepts contained in References 2, 3, and 4 is provided in Appendix F.

also stress that the definition phase of a software development effort is the most critical. DeRoze and Nyman [4] state flatly that the lack of means to produce clear, concise, and unambiguous statements of user requirements is one of the biggest contributors to the high cost of software. A distinction is to be made between the definition phase and the design phase. The definition phase should stress "what" the software product must accomplish. "How" the software will satisfy the defined requirements is the subject of the design phase. (The design of the software topology or structure is not to be confused with the writing of the code to implement that topology.) A clear understanding between the user (RADC) of the ADACS software and the developer of the needed and desired performance objective must be achieved before any coding of statements is begun. One way of assuring maximum communication between the user and developer/programmer is to insist that a thorough and comprehensive software design be prepared and submitted to RADC for review and acceptance before any appreciable coding is done.

Figure 3 depicts a program schedule which emphasizes definition and design prior to the actual writing of code. Note that the first half of the program focuses on defining the functional objectives of the software and on developing a software design plan to meet these objectives. It is recommended that RADC require that the design plan be submitted for review and that the developer present his design recommendations to RADC via a formal briefing. RADC should exercise this opportunity to very carefully scrutinize the software design, obtain clarification on details of accomplishment of the stated goals, and in general try to find out what the software will and won't do at this point in the development cycle rather than waiting until after the program is on the machine and debugged to find out exactly what the program will do. Proper review before actual software preparation will allow realistic tradeoffs to be made.

As noted, actual coding does not begin until after the design plan is approved. (The Design Plan should include detailed descriptions of the necessary interfacing rules which will allow control and analysis modules to be written independent of each other and of the main program (the Program Executor).) With exact programming rules laid down, the software development effort can proceed along several parallel paths.

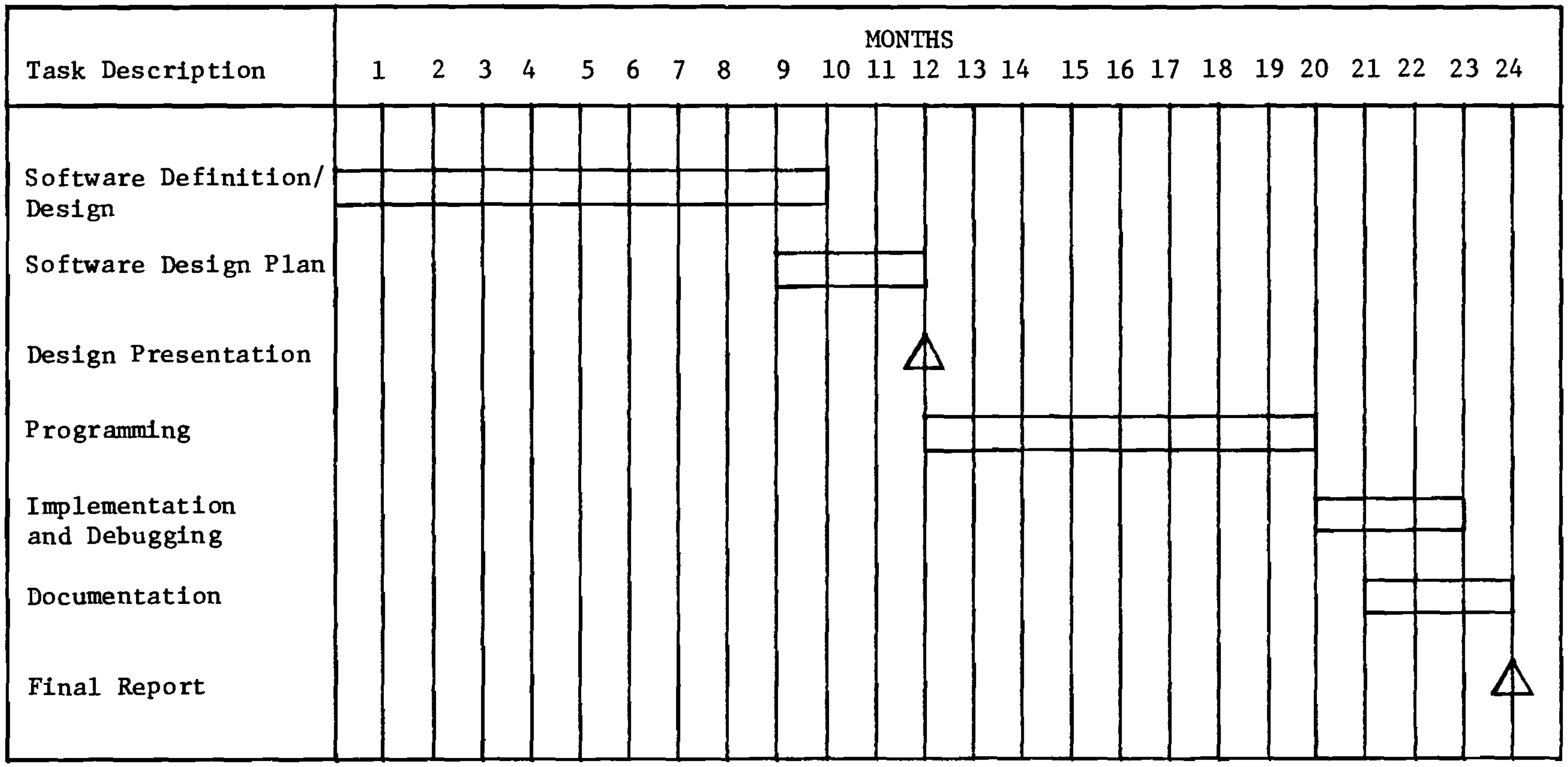


Figure 3. Suggested Schedule for ADACS Software Development Program.

The schedule of Figure 3 also anticipates an Implementation Period during which the written software is installed on the HP computer at RADC and the necessary debugging accomplished.

Documentation is a major and critical (for RADC's subsequent updating and modifying of the program to accommodate future needs) part of the program. Clear and complete documentation of the software to emphasize User Instructions is a very important part of the total software package.

The program schedule of Figure 3 (or one similar) should achieve the goal of obtaining a clear understanding between RADC and the software developer of the objective and capabilities of the ADACS software and allots representative time periods for the writing of code, implementation of the written program on the HP 1000/45 system at RADC, and the preparation of needed documentation.

5. Summary and Recommendations

A possible software design philosophy is set forth which, it is felt, will meet the needs of RADC while providing the flexibility to test new and diverse munition systems without the need to write software code as part of a test plan. The recommended design philosophy is based on a workable system that has been designed by expert software programmers and hardware engineers for automated EMS/V testing that is similar to that performed by RADC.

Discussions with engineers at various DoD agencies which perform EMS/V tests on munitions system revealed differing views on what can be accomplished by state-of-the-art software in an automated test facility. There was one common view, however--that view was that a general test program from which the operator can configure a complete test sequence by dialogue with the program is neither desirable nor possible. It is not desirable because it places the decision burden on a test operator who may be a technician rather than the test engineer who writes the test plan. Because of the sophistication of the tests and the large costs involved, it is desirable to have a software system that allows the test engineer to configure the test sequence as part of his test plan and to have that test sequence entered as coded data at the program run time. The concept of a generalized program with dialogue (prompts) is not considered to be the optimum approach for munitions system testing because of the great diversity of the systems and because the software requirements might exceed the state-of-the-art. In paticular, it is felt that such an approach would likely require considerable software development each time that a new munition system is tested, which is very undesirable because the development and maintenance time for new software could easily exceed the munition system test time.

The White Sands Missile Range is the only DoD agency visited during this effort where a specialized menu test program is being developed. The general software design features of the White Sands system should be implementable on the ADACS at RADC and this system should meet the needs of RADC's EMCAF. Although not a program with dialogue, this system has the desirable feature that any test sequence can be configured <u>without the writing of new software code</u>. The tests are configured by the test engineer during the Preliminary Test phase by writing coded data tables which are entered as data into the program at run time. Because these data tables reside on disk files in the computer, they can easily be edited to modify a test sequence at any time. In addition, these data files can be saved for future tests. The White Sands system is a well planned one which requires a minimum of time to set up a test and minimizes the risk of errors during the tests. A similar system should be seriously considered by RADC.

The software data manipulation requirements for the ADACS have been examined and defined. These requirements can be met with relatively simple and disjoint software routines which will require only minor design efforts. Some of the routines are available through the Hewlett-Packard program users group and some will require development.

The Hewlett-Packard 1000/45 computer system has been evaluated in an effort to identify the hardware constraints that can be anticipated by RADC. These are summarized in an appendix.

Finally, a skeleton model of a Statement of Work for procurement of the ADACS software is included. Specific task descriptions have been omitted. In their place, a summary of what should be included in these descriptions is given.

In conclusion, it is recommended that RADC obtain the services of a qualified contractor to design and write an ADACS software program reflecting the philosophy of approach presently being implemented by

WSMR. A minimum of 24 months should be anticipated fo this developmental effort.

6. References

1. H. W. Denny, "Open-Loop Versus Closed-Loop Vulnerability Assessments of Airborne Guided Munitions Systems," Final Technical Report, Contract DAAK40-76-C-1323, Engineering Experiment Station, Georgia Institute of Technology, Atlanta, GA 30332, April 1977.

2. W. C. Cave and A. B. Salisbury, "Controlling the Software Life Cycle--The Project Managment Task," IEEE Trans. on Software Engineering, Vol. SE-4, No. 4, July 1978.

3. R. C. McHenry and C. E. Walston, "Software Life Cycle Management: Weapons Process Developer," IEEE Trans. on Software Engineering, Vol. SE-4, No. 4, July 1978.

4. B. C. DeRoze and T. H. Hyman, "The Software Life Cycle--A Management and Technological Challenge in the Department of Defense," IEEE Trans. on Software Engineering, Vol. SE-4, No. 4, July 1978.

APPENDIX A

TRIP REPORTS

Agency	Date	Visitor(s)
NSWC - Dahlgren	May 11, 1979	H. W. Denny
MIRADCOM - Huntsville	June 26, 1979	W. M. Leach/ J. D. Norgard
EWVAF - WSMR	July 11, 1979	J. D. Norgard
EWVAF - WSMR	August 8, 1979	J. D. Norgard

ENGINEERING EXPERIMENT STATION

GEORGIA INSTITUTE OF TECHNOLOGY

Visit With: Harold Parks, Mead Corder

Firm or Agency: Naval Surface Weapons Center

Address: Dahlgren, VA **Telephone No.:** 703/663-8481

Date: May 11, 1979 **By:** H. W. Denny **Project:** A-2366

Distribution: M. W. Leach, EE; R. W. Moss; R. W. Rice; S. Zehner; G. Lunsford; D. E. Clark

(The significance of Parks' and Corder's comments lies in their experience of some 6 plus years with the original system and with the design of the latest system. However, their perspectives must be weighed in light of their own unique testing needs. There are probably differences in the needs of the Air Force and the Navy and in the ways that the respective organizations do business. Thus the needs for Dahlgren may not be exactly those for RADC. Another factor has to do with experiences and capabilities of the two organizations and in the ways in which they are set up to conduct tests and analyze the findings.)

Dahlgren has a large anechoic chamber comparable in size and power handling capability to the one being completed at RADC. It has been in operation for several years. Their present setup and capabilities are the result of a gradual buildup over the years. An impression is that it is in a state of continuing evolution. For example, presently they are upgrading their computer capabilities to do more comprehensive data reduction. They are presently completing a second generation testing system called EMETEP (for Electromagnetic Environment Transportable Evaluation (?) Platform). EMETEP, equipment wise, is quite similar to RADC's compliment. It basically utilizes a computer-controlled signal synthesizer operating between 50 MHz and 2.5 GHz. Independent control over modulation characterisics (pulse rate, height, width, AM, FM, etc.) are available. Amplitude adjustments are made at low power levels. Output drive to the antennas is provided by broadband RF power amplifiers. This new system uses two HP 21MX computers--one for a process controller and one for data analysis. The 21MX is an admitted overkill for the controller function but the extra cost was justified on the basis of providing backup capability.

All their equipments since 1973 have been computer controllable. At the present time, they are able to monitor (read) the power output of their signal sources and read frequency, PRF, pulse width. They can access and record all pertinent parameters of their test sourcs. On about one half of their sources, they can remotely control modulation parameters such as pulse width and PRF. With some sources, they can even remotely (ergo, with computer) adjust frequency. (These comments apply to the older, higher power level test facility and not to EMETEP.)

They are presently set up to handle 10 or 12 channels of information simultaneously. This information comes from the System Under Test (S.U.T.) via fiber optic links. It is Parks' opinion that during a given type of test the determination of interference basically boils down to the effect at one ot two primary critical points or on one or two primary parameters. On a very large system, such as the Phoenix, as many as 121 parameters may be brought out for monitoring. Of these, perhaps 10-12 may need to be recorded. Typical parameters usually of concern include the warhead detonation signal, wing control signals (perhaps 3 to 4), seeker head behavior (2 to 4 parameters), self-destruct circuits where they exist, and other specialized items such as altimeter readouts, etc. Of all parameters which could be monitored, the very important ones range in number from 6 to perhaps 12. Their present mode of operation calls for up to 10 channels of S.U.T. performance parameters to be simultaneously viewed on small oscilloscopes and recorded on analog tape. They can also record 10 channels on a light-pen recorder (a Bell and Howell Model 5-133 Datagraph). Digitized voltage and frequency readings (and others, I surmise) are temporarily stored on diskettes. The results of any test can be printed out directly.

While a test run is in progress, an operator views and evaluates the results obtained and makes the decision as to whether or not the results are to be stored. Mr. Parks commented that to store all data produced by every test run on a system would overwhelm storage capabilities. An instrument found very useful in analyzing preliminary data (as well as for post-test analyses) is a Real Time Spectrum Analyzer (made by Spectral Dynamics). This device (instrument) is basically a low frequency digital processor. Parks says that this instrument has been found to be the most usable/useful readout device that they have. With proper setup they can obtain three dimensional displays of events. This device is highly prized by the engineers who are concerned with evaluating the behavior/response properties of the system while under test.

Some limited discussions took place as to how rapidly a test could be run. The question was posed as to what element of the test system limits the testing rate (speed of gathering data). The **test** itself limits the rate of testing. A certain minimum "dwell time" is necessary to effect a response in a system. In other words, a certain RF environment parameter combination, i.e., intensity, PW, PRF, etc., must be maintained for a length of time of 0.5 to 1.0 sec in order for a response to be induced in the S.U.T., if it responds at all. Thus the limiting factors on the testing are not computer I/O time, instrument response time, nor data accumulation capability. This RF exposure dwell time (burst) is very important and requires (permits) everything to be slowed down.

(The following comments/observations were recorded while talking to Mead Corder, who is basically responsible for the softwre and programming aspects of the test facility. These comments are assumed to reflect his particular experience and perspectives and may not totally apply to RADC's situation.)

Mr. Corder began his discussion by commenting that when it comes to engineering tests what you wind up doing amounts to a tradeoff between what you want to do and what you can realistically achieve. He is not sure that it is possible to become totally automated because "there is no such thing as a standard test" or (standard series of tests). He says that the final product will evolve through experience. He stated that if he had to verbalize a "Philosophy of Approach" to automated testing, it would be:

> "Start off attempting to do something less than that of ultimate complexity. Begin by implementing less than the desired end. Do the less ambitious and build up. As the learning curve builds up, you will probably decide to stop along the way because of practical difficulties, excessive complexity, expense, hassle, etc."

Pursuing further in the philosophical vein, Mead asserts that it is generally not possible to "RUN A TEST." Certain pieces (subsets) will be common to all (or many) tests. Obviously, these common denominators are the most likely candidates for automation and as candidates as part of a so-called "menu" of tests or software routines. Such elements should be appropriately considered for computer control. It has been Dahlgren's experience that the most effective role for the computer is **Data Reduction.** The computer can take a big load off the analyst through the reduction of data. He says that this is a more important function for the computer than the role of instrument control. (It should perhaps be emphasized that currently many of their highest powered sources are manual.)

He further observes that the interfacing of devices of computer reading/control is a one time effort. The timing (ordering) sequence will change from test-to-test (possibly even day-to-day). He is skeptical about the cost effectiveness of having just automatic control over the test sequence and test procedure. The remote operation of some events, instrument readings for example, is desirable. Even if deemed (proven) cost effective, that is not the total answer. Hardware reliability is a concern - you do not want your total test program to grind to a halt because of a problem with your test controller. The personnel availability/capability mix also enters in. These various considerations lead to the conclusion that regardless of what the computer can do, you should retain the capability to do everything manually if need be.

Relative to software considerations, Mead notes that for a program of given complexity, two programming options exist: FORTRAN and assembly language. FORTRAN usually leads to easier programming and debugging of the software. The penalty paid for this benefit is loss of usable memory in the machine. They have found their needs better served by programming equipment interfaces in assembly language because (1) it provides better control and it is easier to program the interface, and (2) it is more efficient in terms of memory utilization. They have found that conservation of memory is highly critical to their operation.

For their computer (an EIA 680?), they have found that permanent memory requirements very quickly fill up when using FORTRAN. They assert that real time (time sharing) executive software can be expected to occupy $30K^+$ memory. (This number may be unique to their machine.)

At the present time, they employ two front end processors in their system. Each processor has 16K of memory. One processor interfaces with all the RF equipment. The other one looks at the input data.

Mr. Corder seems to feel that the computer/interface/language cycle time will be what limits what data and how much can be read in a given period of time. The time required to gather a given set of data equals the parameters to be monitored times equipments to be read. When an equipment is ready to be read, it must be read. Thus, in commutating through several equipments at a given test point (frequency, for instance), when an equipment must be read, the computer must be able to get there in the available time window. If a large number of equipments are involved, this requirement can rapidly reach time limits or impose a stretch out in the testing process. Mead asserts that language can easily bog down the system where a large number of equipments are involved.

Dahlgren is apparently negative about trying to interlace test execution and data reduction and analysis or program development. They recommend total dedication of the system to the testing process when gathering data. They feel that there is too much risk posed to the test process through an operator inadvertently entering a command which crashes the computer, wipes out data just gathered, etc. Their policy has been to minimize "outside interference" while a test is actually underway. Another factor contributing to their reluctance is that they have found that the second operator frequently needs much of the same hardware that is needed by the first (test) operator.

In addition, I detected that they felt that it was not really necessary to do data reduction and analysis truly simultaneously with the testing. The reason is that there is so much "dead time" between particular test runs that there is ample opportunity to take both quick looks and detailed examinations of the result of previous runs. These dead times result from the need to reconfigure the setup (in their case, to retune some of their sources, change antennas, change the test setup, etc.).

Between evaluations of different systems, they have found that much software reprogramming is necessary. They have found it necesary to tailor their software for each different system tested.

Post Script:

The discussions with Mr. Parks and Mr. Corder were very informative. It is clear that they have some very definite opinions as to what should or could be done in the general area of weapons systems automated RF evaluations. Their perspectives need to be given credibility because of their 6 to 8 years of direct experience doing this kind of effort. As noted in the beginning, however, the equipment capabilities and their computer are different from those at RADC. Therefore, some of their opinions may not apply, or apply strongly, to RADC's situation.

SCHOOL OF ELECTRICAL ENGINEERING

GEORGIA INSTITUTE OF TECHNOLOGY

Visit With: Jim Hill

Firm: Redstone Arsenal

Address: Huntsville, AL

Date: June 26, 1979 **By:** W. M. Leach, Jr., and J. D. Norgard

Project: A-2366

On June 26, 1979, we visited Redstone Arsenal in Huntsville, Alabama, for the purpose of technical discussions with respect to the RADC Software Definition Task funded through the Post Doctoral Program. Our contact at Huntsville was Glenn Brown, a contractor with an office at Redstone who is actively involved in weapon system susceptibility and vulnerability testing and assessment. His sponsor at Redstone is Chalmers Riley who is responsible for a large RF anechoic chamber facility built about 12 years ago which he plans to automate with a small computer in the near future. Glenn Brown and Ben Lowe, an engineer with BDM, accompanied us during our visit.

The purpose of our trip was to visit Jim Hill who works at Redstone. Mr. Charlie Ponds is head of this section and is Mr. Hill's immediate supervisor. Mr. Ponds' group is primarily involved in electromagnetic hazard and susceptibility testing. Their facility is presently being upgraded with a computer-controlled RF measurement and instrumentation system and an array of large high-gain horn antennas. The antennas are scaled up versions of the standard gain horns sold by Scientific Atlanta. They were constructed by local contractors and the mouth of the largest one is big enough for the average person to walk into. The RF system was procured from Watkins Johnson for approximately one million dollars. It is capable of either computer or manual control and consists of a WJ frequency synthesizer which drives a bank of Hughes TWT amplifiers. This system consists of approximately 15 feet of rack mounted equipment.

Jim Hill is a hardware engineer whose area of interest is computer controlled measurement systems. He has developed the measurement system at Redstone using an IBM computer as the controller. The initial software development for the computer was done by Hill, but now a contractor is employed for this purpose. Hill emphasized the concept of a hardware engineer and software programmer team to program and run the system. He and the software contractor comprise that team at their facility. Their test and display programs consist of a collection of about 400 subroutines plus a menu program for electromagntic hazard testing. Hill stated that they had found it impossible to write a menu program for susceptibility testing because they had found tests on each system to be so different that a menu program could not be developed. For susceptibility tests, they write a main program which calls on the library subroutines that have already been developed. Some of these may require modification for the specific tests to be performed. In some cases, new subroutines may have to be written. He said that the test programs for each system are saved after development for possible re-use, after appropriate modifications, on future systems.

The following are some additional points made by Hill during our conversation:

1. The software programmer must know process control.
2. A computer controlled measurement facility needs a full time software programmer and maintenance person.
3. Their software is developed in modular or "building block form."
4. It is not possible to write a single test program that will "do everything."
5. For EMS tests, they start with the equipment control subroutines and write a main program to perform the desired tests.
6. Each new system tested requires both hardware and software development.
7. It is easy to over burden the computer.
8. Do not put classified material into a computer used in a laboratory environment.
9. The HP 1000 computer is designed for instrument control and not data processing.
10. All programs must be written "in house." Otherwise, the programmer cannot see the instruments being controlled.
11. Below 350 MHz, Hill performs his field strength calibration with two B-dot probes, one at the weapon position and one near the transmitting antenna. The output from the one near the transmitting antenna is used during the tests to calculate the field strength at the weapon. Above 350 MHz, Hill's calibration is similar to the method used at RADC.
12. Hill recommends monitoring both forward and reflected power in the antenna feed line during tests. An increase in reflected power can mean a defective cable or antenna.

(We asked to see any electro-optical target systems used in EMS tests. The most sophisticated system that they had was the standard light bulb on a rotating boom.)

SCHOOL OF ELECTRICAL ENGINEERING

GEORGIA INSTITUTE OF TECHNOLOGY

Visit With: Bill Collins

Firm: White Sands Missile Range

Address: Electronics Warfare Vulnerability Analysis Facility

Date: 11-12 July 1979 **By:** J. D. Norgard **Project:** A-2366

On Wednesday, July 11, I visited the Electronics Warfare Vulnerability Analysis Facility (EWVAF) at the Discrimination Radar (DR) site at White Sands Missile Range (WSMR). I met with Bill Collins, the director of the facility, and with Bill Cooper and Pat White, both contractors from PSL/NMSU who write their computer system software. The purpose of this meeting was to learn how they have structured their software system for their semi-automated weapon susceptibility tests. They operate a "distributed computer system" composed of Interdata and SEL main computers and a Motorola 6800 minicomputer. The Interdata and SEL operate in parallel for backup capabilities, while the 6800 operates the basic functions of instrument control and serves as an "intelligent" IO device. The 6800 buffers its data and the main computer reads the 6800 as a terminal. The facility has a small rectangular anechoic chamber and a large tapered anechoic chamber. Vulnerability tests can be performed simultaneously in either chamber. They intend to purchase another minicomputer so that they will have one for each chamber. The 6800 is programmed in Basic; whereas, the main computer is programmed in FORTRAN. The typical test operator re-programs the minicomputer in Basic (the only language simple enough for their people to learn). The minicomputer has a "stand alone" capacity and can operate in a degraded state when the main computer is down. All of their equipment have 100% manual backups.

They are now working on a General Purpose Test Program (GPTP) for data acquisition and have not begun to write the data reduction program. The GPTP software pre-processes the raw data with numerous statistical routines to reduce it to a form suitable for storage. Their approach is to use a modular structure called by an "executive" program with several "levels." The executive program consists of an operating program, a table driven menu, and computed GO TO statements. The table driven menu itself is a fixed program which can edit variable table stored on the disk. The table contains a sequencing array which controls a relay bank (relay masks) connected to the equipment available at the facility (facility dependent). The table also contains a patching array which selects the TM channels for various operations (weapon dependent). The table can be edited and changed to accommodate new weapon tests without reprogramming the software. This system seems to provide the needed flexibility in a menu program. They use this scheme for all of their tests.

They recommend top-down design and bottom-up coding. The hardware limitations are so demanding that it is impossible to code top-down. They estimate a 5 man-year effort for their data acquisition program for software at the executive level and another 5 man-year effort for the development of the hardware drivers and interfaces.

Their weapon tests are limited to no more than 1 minute of missile run time. This is due to possible overheating and wear. They keep an automatic record of all tests including test number, purpose, data, start/stop times, total lapsed time, spin-up time, etc. In order to perform a test quickly, all computer interupts from other users are suspended. Also, the operating system is locked-out. Extensive program overhead is required to integrate the terminal for any abort instruction. Each test has a definite start-up and shut-down sequence. If a test is aborted, the shut-down sequence is initiated. Before a test can be resumed or re-started after an abort, a definite cool-down period must elapse. Timing programs must be written to prevent an early re-start.

They use air line relays and fiber optic TM packages to isolate their equipment from the chamber. They have had considerable problems with RF leakage due to poor shielding during high power/low frequency tests. (Several ROM's have arced over.) Equipment must be connected to the IEEE bus with cables less than 20 feet in length (or else there are timing problems). This requires that the equipment be located close to the chamber.

Their last recommendation was to keep each part of the software simple and don't try to automate everything. This is especially true with regards to the data reduction program, which is very weapon oriented.

SCHOOL OF ELECTRICAL ENGINEERING

GEORGIA INSTITUTE OF TECHNOLOGY

Visit With: Bill Cooper

Firm: White Sands Missile Range

Address: New Mexico

Date: 8-9 Aug 1979 **By:** J. D. Norgard **Project:** A-2366

On 8 August 1979, I met wih Bill Cooper at the DR building at WSMR to discuss the details of designing a software system using a "table driven" menu concept. We first discussed the role of the operating system (OS), which manages the system resources in a way to perform a sequence of operations. The OS consists of a "supervisor," an "executive/ manager," and a large number of instructions divided into "subprograms" (system tasks) which can perform the various operations. The "supervisor" directs the overall operation of the system. The "executive" is the interface between the operator and the system. It interprets (decodes) the operator commands, searches a privileged menu for the command (which returns a number), does a computed GOTO on the number (which establishes the proper subprogram operation), and executes the subprogram instructions. The "manager" acts as the "executive" for the users but has a restricted (non-priviledged) menu. The non-privileged menu for the "manager" is a subset of the privileged menu for the "executive." In all other respects, the "manager" acts just like the "executive."

A "direct" menu is located in memory and contains all of the direct operations that the system can perform; an "indirect" menu (an extended menu) is stored on disk and contains user/operator written operations. The user/operator written items in the menu are "procedural" programs consisting of a list of the basic operations, i.e., chains/links of the basic operations. These procedural programs are called Command Substitution System (CSS) programs. The system looks first into the menu in memory, then, if it cannot find the called item, it searches the menu on disk. The commands entered by the operator are called Console Commands; the commands entered by a user are called User Commands. Either command has the form

mnemonic, operation (list)

Examples of console commands are given in Table I. Examples of user commands are given in Table II. A list of the extended "indirect" menu items is given in Table III. Only the mnemonic is given, the list of operations has been omitted.

The "executive" or the "manager," if it has the privilege, executes the instructions corresponding to a command found in either menu according to the options/instructions present in the operand list. The various options are implemented through a command processor which generates a supervisory call (SVC) which handles the subprogram linkage. When an SVC is generated, the supervisor is interrupted, critical parameters

TABLE I

Allocate
Assign
Attn
Bias
Bfile
Brecord
Build
Cancel
Close
Continue
Delete
Display
End B
Examine
Ffile
Frecord
Load
Mark
Memory
Modify
Options
Roll, No Roll
Pause
Remove
Rename
Reprotect
Reset
Rewind
Send
Set
Start
Task
Team
Tempfile
Volume
Wfile
$Build
$Clear
$Copy
$End
$Exit
$IF
$Job
$No Copy
$Skip
$Termjob

TABLE II

Allocate
Assign
Bias
Bfile
Brecord
Build
Cancel
Close
Continuous
Delete
Display
EndB
Examine
Ffile
Frecord
Inquire
Load
Log
Message
Modify
Pause
Print
Remove
Reprotect
Rewind
Send

Set Code
Sign Off
Sign On
Start
Submit
Tempfile
Volume
Wfile
XDelete
$Build
$Clear
$Copy
$Continue
$Else
$EndB
$EndC
$Exit
$GOTO
$IF
$Job
$Label
$No Copy
$Pause
$Skip
$Termjob
$Write

TABLE III

HLOC
F7C
BASIC
F7HLOC
ALLONEW
ASSEMBLE
CAL
CALCLG
COMPILE
COPY A
COPY T
ESTAB
F70
EXIST
EXPAND
FORT
FORTCLG
FORT6C
FORT6CA
FORT6CAE
FORT6CLG
HELP
MAC
MACCLG
RUN
SYSGEN1
SYSGEN2
EDIT
F7CLG
F7D
TASK
SYSGEN2X
STATUS
MTM
SPL
SHUTDOWN
SPLCA
SIZE
STARTUP
PATCH04
BCDIC
EBCDIC
TIMER
F70L
QQ
OLD
F7CE
ESTABF
COPY

are stored, a set of instructions are executed (depending on the directions contained in the data stored in a "parameter block"), and then an interrupt return is issued returning control back to the supervisor. The "parameter block" contains (a) a function code to direct its operation, (b) logical unit number(s) of the device(s) necessary to perform the function, and (c) an error status code that can be checked by the supervisor. Examples of SVC functional directives are:

1. IO: read/write (and proceed, pause, wait)
2. System utility - pause
   get/release storage
   set status
   fetch positioner
   unpack binary number
   log message
   interrogate clock (timing)
   fetch data
   time of day wait
   timmer management (set clocks)
   interval wait
   pack numerical data
   pack file descriptor
   scan mnemonic table
   move ASCII characters
   peek
   expand/contract allocation
3. End of task
4. Overlay call
5. Intertask coordination
6. File handling services
7. User SVC

All of the subprograms that are available in the operating system through the use of a console or user command can be incorporated in a FORTRAN program through the use of the SVC. This is accomplished in the program by calling a subroutine which contains a direct call to a SVC. The "parameter table" must now be written in Assembly Language.

On August 9, we discussed how the OS controls the "table driven" menu. We used the "state diagram" for the Stinger Test Program as an example of the use of this design concept, see Figure 1. Referring to the figure, each square block represents a basic module, i.e., a particular function which can be performed during the test of the weapon system. Each circle represents a local "executive" containing a menu. The mnemonic for each module is the letter placed near the line leaving the "executive" to the module. Each module is a FORTRAN subroutine stored on disk in source, assembly, and object codes. For each module, a CSS procedural file is also stored on disk which will link the various object codes into an executable machine code and load and execute the program. Since each module is an independent entity, it can be compiled

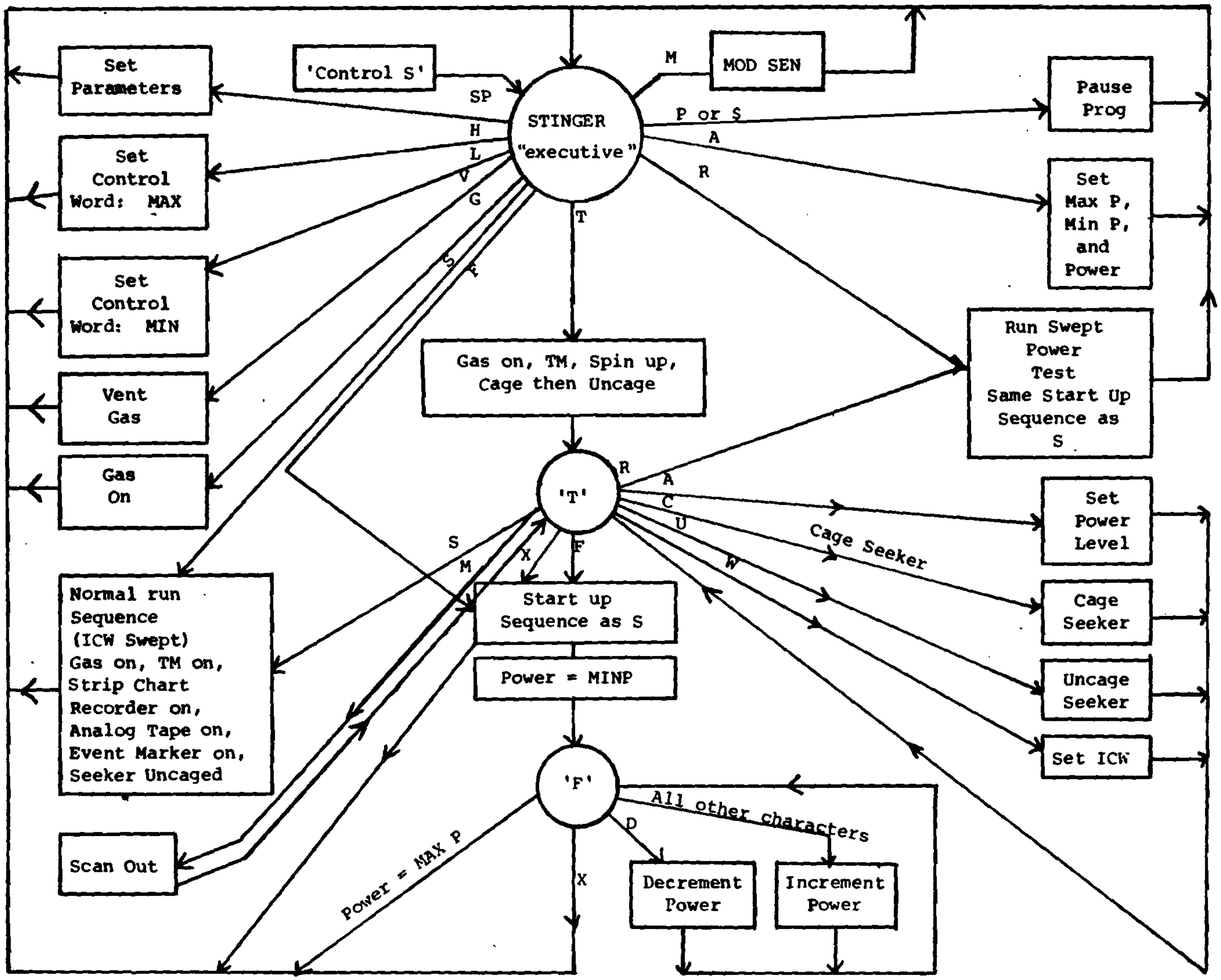


Figure 1. State Diagram for Stinger Test Program.

and saved. If a module is changed, it, of course, must be recompiled and stored in the updated form. The only change then required to update a CSS procedural file is to use the linkage editor to relink the entire code. No further recompilation is necessary. In this sense, each module is thought of as an element in a library file. For this reason, all subroutines must be reentrant and all subroutines common to all simultaneous users should be placed in a run-time library if multiple task programming (time sharing) is to be used. In a reentrant subroutine, any impure code (data addresses assigned inside the subroutine) must be designated and kept separate from any pure code (data addresses assigned outside the subroutine). The code, therefore, must not contain direct memory references on storage, and the calling routines must pass to the subroutine through the call list all storage information. This is important in the writing of any Assembly Language subroutines to be called by a FORTRAN program.

To enter the highest level executive state in the Stinger code, a console or user command is entered to begin the Stinger CSS file. A prompt is issued to the calling console to indicate a menu item is requested. Some of the menu items are single purpose manual operations; whereas, others form a normal test run sequence, such as item "S." Item "S" requires that the parameters contained in several "mechanical" relay tables and an "electrical" data table be set before it is called. This is accomplished by menu item "SP" which contains a lower-level "executive" which can edit the tables and change the table entries to values other than their present (default) initial values. The two "mechanical" tables are used in a "Start Normal Run Sequence" (SNRS) subroutine and a "Normal Shut Down Sequence" (NSDS) subroutine. The SNRS subroutine locks out all other tasks, assigns exclusive control of the relay banks to the calling executive, logs all comments, conditions, parameters, etc., and turns on all devices per a "Start Up Delay Table" (SUDT), cf. Table IV. The NSDS subroutine turns off all devices per a "Shut Down Delay Table" (SDDT), cf Table IV. The "electrical" table is stored in a common block and contains a "Data Type Table" (DTT) for TM channels and probes, cf. Table V. The data type code is:

0: not used
1: use one data point only
2: read peak value
3: read minimum value
4: use average value
5: read RMS value

The SUDT table contains the time delay (in seconds), from the start of the test, before the relay is energized. A "Start Up Time Table" (SUTT) is generated by calling the clock at the beginning of the test to convert the delay times after the start of the test to actual real times which can be compared to other calls to the clock throughout the remainder of the test. The SDDT contains the time delay, in seconds, from the end of the test (normal stop, or abort) before the relay is de-energized. A "Shut Down Time Table" (SDTT) is generated by calling the clock at the end of the test to convert the delay time after the end of the test to actual real time, which can be compared

TABLE IV. "Mechanical" Arrays

Relay Mask 1111 1111 1111 1111 = X'FFFF' = 65535 (protection key for relays)

		relay #	start up	shut down		
SUDT/SDDT	(1)	0	34	1	IR Source	
	2	1	15	3	Strip Chart (EEC1)	
	3	2	39	0	Event Marker (EEC1)	
	4	3	1000	1000	Mag Tape Start	(Tapered Chamber)
	5	4	1000	1000	Mag Tape Stop	
Automatic relays	6	5	4	1	IR Receiver	
	7	6	4	1	G & C Power	
	8	7	39	0	RF Power	
	9	8	0	3	Magnetic Tape (2200 Ampex)	
	10	9	X	X	Spare	
	11	10	15	3	Strip Chart (XMT/EEC2)	
	12	11	39	0	RF (ON/OFF)	(Anechoic Chamber)
	13	12	4	1	G & C Power	
	14	13	34	1	IR Source	
	15	14	4	1	IR Receiver	
	16	15	39	0	Event Marker (XMT/EEC2)	
	17		X	X		
Manual relays	18		X	X		
	19		X	X		(Not in use)
	20		X	X		
	21		29	3		
	22	(21)	0	3	Honeywell Fiber Optics	
	23	(22)	4	1	Analog Tape	
	24	(23)	39	3	Maximum Test Start/Stop Time	

UPTIME	180	Maximum Time Up
CDR	0.5	Cool Down Ratio
MDV	1024	Maximum Data Value (used to set scale)

TABLE V. "Electrical" Array

CWS 1 Control Word Scalefactor (# of data points/run = 1024/CWS)
NODPPF 1024 # of Data Points/Frame

		Channel #	Date Type		
DDT	(1)	0	0	Spin Speed	
	2	1	2	Noise	
	3	2	0	30V Squib Wing	
	4	3	0	Detector	
	5	4	0	-30V Thermal Battery	
	6	5	0	Target ACQ	
	7	6	0	Vane 1 POS	
	8	7	0	Vane 2 POS	
	9	8	0	Vane 4/2 POS	(TM Channels)
	10	9	1	Yaw	
	11	10	0	Pitch	
	12	11	0	Pitch Error	
	13	12	4	Yaw Error	
	14	13	3	Video AGC	
	15	14	0	Yaw Vane	
	16	15	3	1st State AGC	
	17	16	0	Frequency	
	18	17	0	Phase	
	19	18	0	Radiometer	
	20	19	0	Step	
	21	20	0	Scanner	(Probes)
	22	21	0	Polarization	
	23	22	0	Mast	
	24	23	1	Power Meter	
	25	24	0		
	26	25	0		
	27	26	0		(Not in use)
	28	27	0		
	29	28	0		
	30	29	0		
	31	30	0		

PLOT 0 (Channel to be plotted 0-31)

to other calls to the clock. A "State Diagram" for the above described procedures is contained in Figure 2.

The table also contains a "relay protect mask" which gives the calling executive exclusive control of 4 relay blanks (each blank contains 4 relays) if a one (as opposed to a zero) is present in the appropriate position of the relay mask hexadecimal word. Also, the range of the "control word" and the number of increments it is allowed to have are contained in the table.

Bill Cooper believes that the dialogue between the user and the "supervisor" should be kept to a minimum, since most users are <u>not</u> interested in programming. He uses a single character prompt in his menus and stores all of the default values of the parameters to be used in a file. The user, therefore, changes only those parameters that are necessary, eliminating the need for the user to retype those numbers each time the program is executed.

His philosophy is: the operator guides the program, the program doesn't guide the operator. The program is only a tool to support the user and the program is originally created from a test plan.

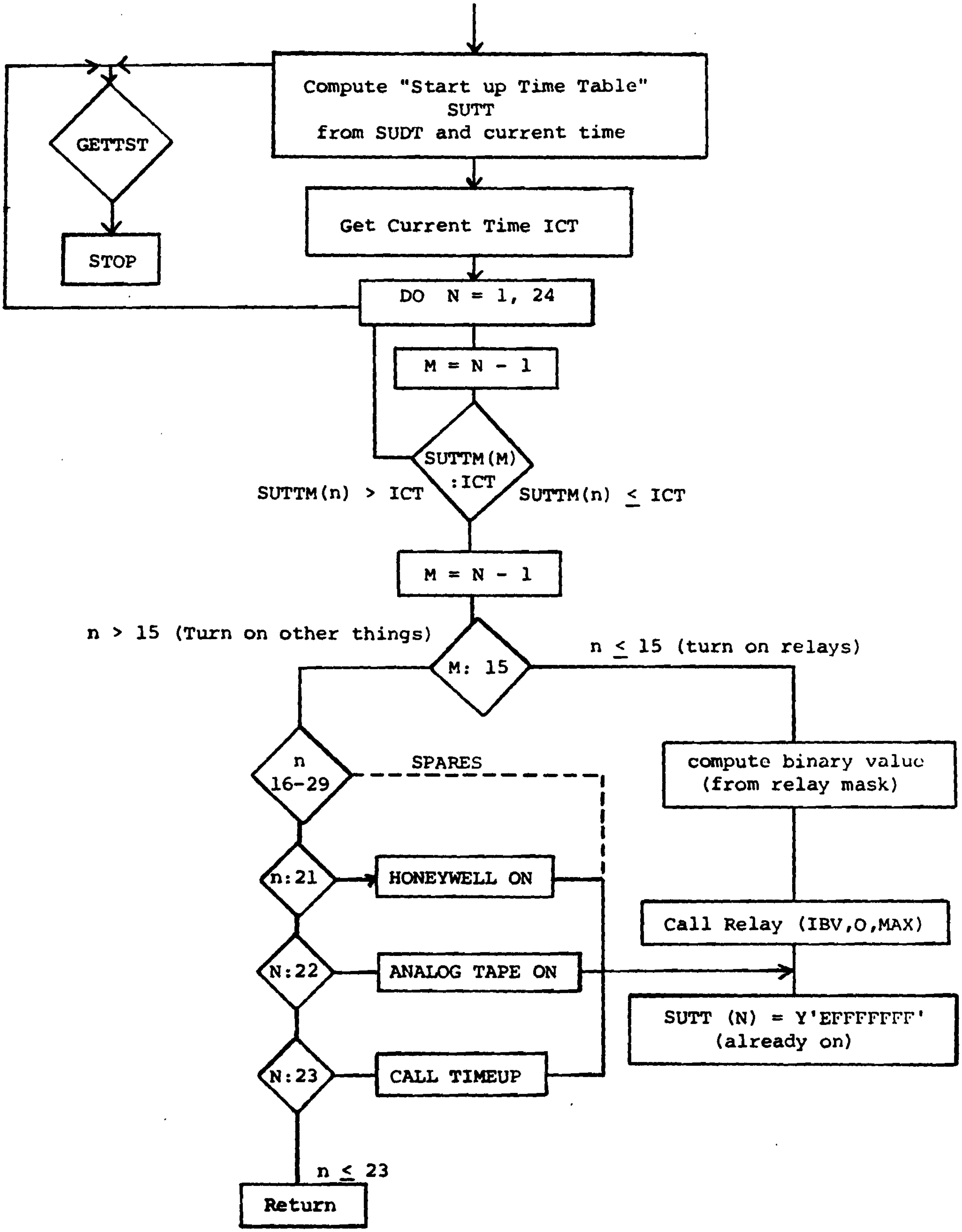


Figure 2. State Diagram for Start Up Sequency.

APPENDIX B[*]

EMS/V TESTING IN EMCAF[**]

A basic objective of open-loop weapons system testing is to establish the transfer relationship existing between a defined EM field and its produced effect at a critical point within the system-under-test (S.U.T.). (Only non-adaptive behavior is observable during open-loop testing, i.e., the incident EM field is not altered to account for the likelihood that the S.U.T. may encounter a new environment as a result of an EM pertubation of its guidance and control system.) A wide variety of tests are required to characterize the EM coupling into the weapon's electronic systems under all of the likely environmental conditions.

EM coupling is typically identified or characterized by the effects noted on some critical element of the weapon or by the presence of a particular voltage level, signal-to-noise ratio, unwanted bit, or other manifestation of potential disruption of the performance of the system. An example of a common manifestation of disruption is the loss of fin control. Loss of fin control can be the result of target breaklock, an incorrect command to the fins, or the presence of false guidance signals. Typically perturbations such as loss of fin control can be related to a measurable quantity at some critical location within the guidance and control circuitry. These measurable quantities usually are defined in terms of "Standard Responses," The most commonly employed Standard Responses are:

- Voltage level at a test point;
- Minimum discernible signal (MDS);

---

[*] Prepared by H. W. Denny, G. H. Lunsford, and J. E. Balsam

[**] The first step in the development of the ADACS software should be to define the functions which the software is to perform (see Appendix F). Without this definition, there is a high risk that the software contractor will not produce a system that meets the needs of RADC. It is felt that this (or similar) background material in this appendix should be supplied to a prospective or winning software contractor so that he will be more likely to understand the overall goal of the Air Force EMS/V test program.

- Fixed dB amount above MDS;
- Logic level (false acquisition); and
- Pulse count of command pulse train.

The particular standard responses tend to be unique to a particular system. The specific ones to be used to indicate possible interference are determined during the System Study and Preliminary Test Phases of the weapons evaluation cycle. A very large number of interference-related parameters can be responsible for these types of effects. Typical parameters of importance include frequency, modulation, power level, polarization, and incidence angle (azimuth) of the interference source in addition to target size, target intensity, etc. Table B-1* presents a listing of EM exposure field parameters along with the normal ranges of these parameters that are used at RADC for EMS/V assessments. This list of parameters is somewhat self-explanatory except perhaps for entry 5.d.6, "Burst Time." Burst time refers to the duration of exposure of the system to a string of pulses of RF energy. (See related discussions reported in the Dahlgren trip report in Appendix A.) A burst can consist of as few as 10 pulses or as many as 10^6 pulses depending upon burst length and the duration and PRF of the individual pulses. Burst time is an important parameter because it reflects (1) the physical event of a flyby past a static source or the effects of a sweep from a scanning source and (2) the recognition that typically a certain minimum "build up" time is required after onset of exposure for an effect to be realized from the exposure.

In addition to these exposure field parameters, certain target parameters are examined. Table B-2 lists the more significant target parameters and indicates the type of seeker to which the parameters apply.

The test methodology presently employed is diagrammed in Figure B-1. (It should be noted that the testing efforts reflected by this diagram represent only the first portions of the total vulnerability assessment process--See Figure 2, page 6.) Initially, the system to be tested is studied to ascertain its principles of operation and

---

*This table along with Tables B-2 and B-3 were prepared by Mr. Sikes and Mr. Robeson of ARC.

TABLE B-1

EXPOSURE FIELD PARAMETERS ASSOCIATED WITH WEAPONS SYSTEMS EMS/V ASSESSMENTS

Field Parameters	Desired Range
1. Frequency (F)	50 MHz - 18 GHz
2. Power density (P)	Up to at least 10 dBm/cm^2
3. Polarization (POL)	Horizontal, Vertical
4. Aspect angle (θ)	Azimuth: 0° - 360° Roll: 0° - 180°
5. Modulation	
a. CW	N/A
b. AM	Sine wave, Square wave Mod. index: 0 - 90% F_m: 10 Hz - 20 kHz
c. FM	Sine wave, Square wave, NB, WB F_{dev}: up to 100 kHz F_m: 10 Hz - 20 kHz
d. Pulse	
1. Repetition rate (PRF)	10 Hz - 1 MHz
2. Duty cycle (DC)	10^{-4} - 1
3. Width (PW)	0.1 μs - 1 ms
4. Rise time (T_r)	50 ns - 2 μs
5. Fall time (T_f)	50 ns - 2 μs
6. Burst time	Width - 10 ms - 1 s PRF: 0.1 Hz - 10 Hz or Single burst
e. Complex	
1. Noise	NB: 1 - 4 MHz WB: 50 MHz - 70 MHz Pseudorandom
2. Chirp	(Not Identified)

TABLE B-2

TARGET PARAMETERS (T)

Parameter		Type Guidance		
	LG	TV	IR	RF
1. Illumination (TI)	x	x	x	x
2. Contrast ratio		x		
3. Size		x		
4. Track Rate (TR)		x	x	x

Legend: LG - Laser Guided
TV - Television Guided
IR - Infrared Guided
RF - Radio Frequency Guided

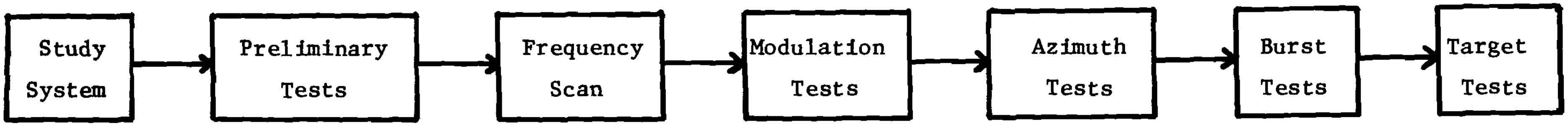


Figure B-1. General EMS/V Testing Methodology for Guided Munitions Systems To Be Implemented at RADC.

to define means for assessing its performance, or peturbations thereto. Physical and analytical models of the system are examined; from this examination, the necessary tests are defined. Likely test points are selected and appropriate test procedures are tentatively defined.

During the preliminary testing phase, the subject system is setup in one of the chambers of the EMCAF. Primary power is applied. Test points are accessed and are monitored for Standard Responses while the system is subjected to frequency scans, representative modulation tests, and target tests. For these preliminary tests, the frequency scans and modulation tests are conducted nose-on and target response tests are conducted with no RF applied. These preliminary tests are aimed at establishing degradation criteria and identifying the modulation parameters producing the most noticeable effects at the test points.

As an adjunct to the Preliminary Tests, various support tests are conducted. These support tests include the making of antenna patterns and the measuring of the gains of any probe antennas not previously measured, calibrating the test chamber, and the gathering of needed calibration data.

The "Frequency Scan" block of Figure B-1 represents the beginning of activities undertaken during the formal RF testing procedure. (Although the order of conducting these RF tests is not necessarily fixed, past experience has shown that the indicated progression is suitable in terms of maximizing the information gained and minimizing the degrading effects upon the missile itself).

The primary purpose of "Frequency Scan" is to determine the Power Density (P) in dBm/cm^2 required to obtain a given Standard Response level at various frequencies to which the S.U.T. is found to be susceptible. These susceptible frequencies were found during the Preliminary Tests by sweeping the transmitted frequency through the band(s) of interest at a constant power level and observing the behaviors of selected degradation outputs. The resulting frequencies that indicate susceptibility of the S.U.T. then become those selected for the Frequency Scan Test. The typical frequency scanning range is from 50 MHz to 18 GHz with frequency increments being no greater than 5% of the lowest frequency in the band. The frequency scan testing is generally performed for various combinations of Pulse Width (PW), Pulse Repetition Frequency

(PRF), Exposure Dwell Time (EDT) per sweep, and Horizontal/Vertical Polarization (H/V POL). The principal output consists of plots of Power Density vs. Frequency for various values of the modulation quantities and polarization. This test and other RF tests are generally conducted remotely and in a rote-type mode and thus lend themselves to automation with the expanded computing/controlling capability of the HP1000/45 system.*

"Frequency Scan" is followed by "Modulation Tests" wherein a few selected frequencies (indicated by the "Frequency Scan" results) are used to generate a parametric family of curves with Power (P) in dBm/cm^2 as the ordinate and modulation quantities such as PRF and PW as abscissae. Four to five frequencies at which the munitions system showed the greatest sensitivity are selected for parametric variation. The basic outputs of these modulation tests involving PRF and PW consist of plots showing the Power Density (P) for each frequency (F) that is required to evoke a Standard Response from the S.U.T. During the PRF testing, the Pulse Width (PW) is held constant and vice-versa. Special degradation effects in the S.U.T. can be monitored simultaneously with the measurements that are being taken for specific Standard Response purposes. As in the case of the Frequency Scan, a basic menu-driven type interactive program is desirable, since the dialogue mechanism would permit the test operator to choose the F for modulation parameterization, etc.

The next block in Figure B-1 represents the "Azimuth Tests." The mounting pedestal can accomplish azimuthal movement under automatic control, but roll maneuvers must be manually performed. (The pedestal is not equipped to provide any pitching relocation.) In addition to the usual plot depicting Power Density (P) vs. Frequency required to achieve the Standard Response, a plot showing degradation level as a function of aspect angle can also be generated. (RADC currently has this Azimuth Test co-programmed with other modulation tests.)

The last RF test block shown in Figure B-1 is entitled "Burst Tests" and represents those testing efforts that seek to measure the response time of the weapon to the number and duration of pulse bursts

---

*A more indepth examination of the software-related aspects of automating "Frequency Scan" is included as an addendum to this appendix.

incident upon it. It is important to establish the relationship between dwell time (a measure of absorbed incident energy) and weapon degradation in order to assess missile performance in an intentionally hostile enviroment.

At the conclusion of the series of RF tests, the next phase of testing activity begins with the introduction of the target itself in the "Target Tests" activities. With the advent of the target entering the testing regime, the implications of vulnerability arise; i.e., the RF testing reveals missile susceptibility, whereas the degradation in missile capability to achieve the intended target intercept is a measure of missile vulnerability to the given threat system. The information provided by the target tests is combined with the results of the RF tests as part of the vulnerability assessment process.

In summary, the formal data acquisition process typically involves a series of frequency scans, modulation tests, azimuth tests, burst tests, and target tests. The data matrix produced by this process is shown in Table B-3. The size of this matrix indicates that a considerable volume of data is produced by the typical test program, implying the need for extensive storage, retrieval, and manipulation capabilities. The relative capabilities of the HP1000/45 system for meeting this need are reviewed in Appendix C. The types of data manipulation and processing routines needed to provide the types of outputs desired from the testing efforts are discussed in Appendix D.

TABLE B-3

EMS/V DATA ACQUISITION MATRIX

TEST TYPE	DEPENDENT VARIABLE	INDEPENDENT VARIABLE	ADJUSTED PARAMETERS	FIXED PARAMETERS
Preliminary tests	TP(volts)*	P(mw/cm^2)	F(MHz)	PRF,PW,POL,θ
	TR(BL)	TI(dBm/cm^2)	TR(deg/s)	T,no EMI**
Frequency scan	P(mw/cm^2)	F(MHz)	P(mw/cm^2)	PRF,PW,POL,θ
Modulation tests	SR	PRF(Hz)	P(mw/cm^2)	F,PW,POL,θ
	SR	PW(µs)	P(mw/cm^2)	F,PRF,POL,θ
	SR	T_r(µs)	P(mw/cm^2)	F,PRF,PW,T_f,POL,θ
	SR	T_f(µs)	P(mw/cm^2)	F,PRF,PW,T_r,POL,θ
Azimuth tests	SR	θ(deg)	P(mw/cm^2)	F,PRF,PW,POL
Target tests	P at BL	PRF(spot)	P(mw/cm^2)	F,PW,POL,θ,T
	P at BL	F(spot)	P(mw/cm^2)	PRF,PW,POL,θ,T
	TR at BL	TI(dBm/cm^2)	TR(deg/s)	F,P,PRF,PW,POL,θ,T
(AM)	P at BL	F_m(Hz)	P(mw/cm^2)	PRF,PW,POL,θ,T
(FM)	P at BL	F_m(Hz)	P(mw/cm^2)	PRF,PW,POL,θ,T
(Chirp)	P at BL	F_m(Hz)	P(mw/cm^2)	PRF,PW,POL,θ,T
(Noise)	Noise pwr at BL	F(MHz)	P(mw/cm^2)	POL,θ,T

*Legend:
TP - Test Point
P - RF Power Density
F - Frequency of Source
PRF - Pulse Repetition Frequency
PW - Pulse Width
POL - Polarization
θ - Azimuth Angle
F_m - Modulation Frequency
TR - Track Rate
TR(BL) - Track Rate at Breaklock
BL - Breaklock
TI - Target Illumination
T - Target
SR - Standard Response
T_f - Pulse Fall Time
T_r - Pulse Rise Time

**Provides data on system base line performance against which EMI-performance is evaluated.

ADDENDUM TO APPENDIX B

AUTOMATION ASPECTS OF THE FREQUENCY SCAN TEST PROCEDURE

Figure B-2 illustrates a detailed flowchart of the Frequency Scan Testing Procedure as is presently implemented in a partially automated mode. The program basically consists of user input and initialization sections and two nested loops which control the hardware outputs and measure the monitoring inputs. These two loops may be subdivided into sections according to the function each part serves. With the advent of a more modular structure like that described in Appendix E, these implicit loops may be transformed into more versatile procedures which afford a greater readability.

The statements in Section 1 of Figure B-2 provide the user with the capability to specify parameters involved in the Frequency Scan (FS) Test, such as frequency, frequency step, PRF, PW, azimuth position, types of standard responses (SR), and SR monitoring points. Also input at this time are types of display overlays and maximum weapon operating time. The software control program allows the test operator to review and modify these input data prior to beginnning the active testing procedure.

Section 2 initializes the hardware by turning on the weapon, reading the clock, controlling the various synthesizers and attenuators for the proper frequency and power, and setting the desired modulation. The frequency meter is read and a check is made to verify that it agrees with the frequency setting. If not, the program is stopped. Otherwise, the synthesizer modulation and the azimuth position are set, and the displays are initialized with the proper axes, labels and headers.

Section 3 in Figure B-2 provides an opportunity to verify transmitted power levels and to check power meter readings each time a new frequency is set during the test. These activities are accomplished by reading the power meter, increasing the power level, waiting sufficient time for meter resonse, and rereading the meter again. If the power meter does not register the increase, the power is increasd again and rechecked. If maximum power is attained without an increase in the power meter reading, the program is halted. If the power meter

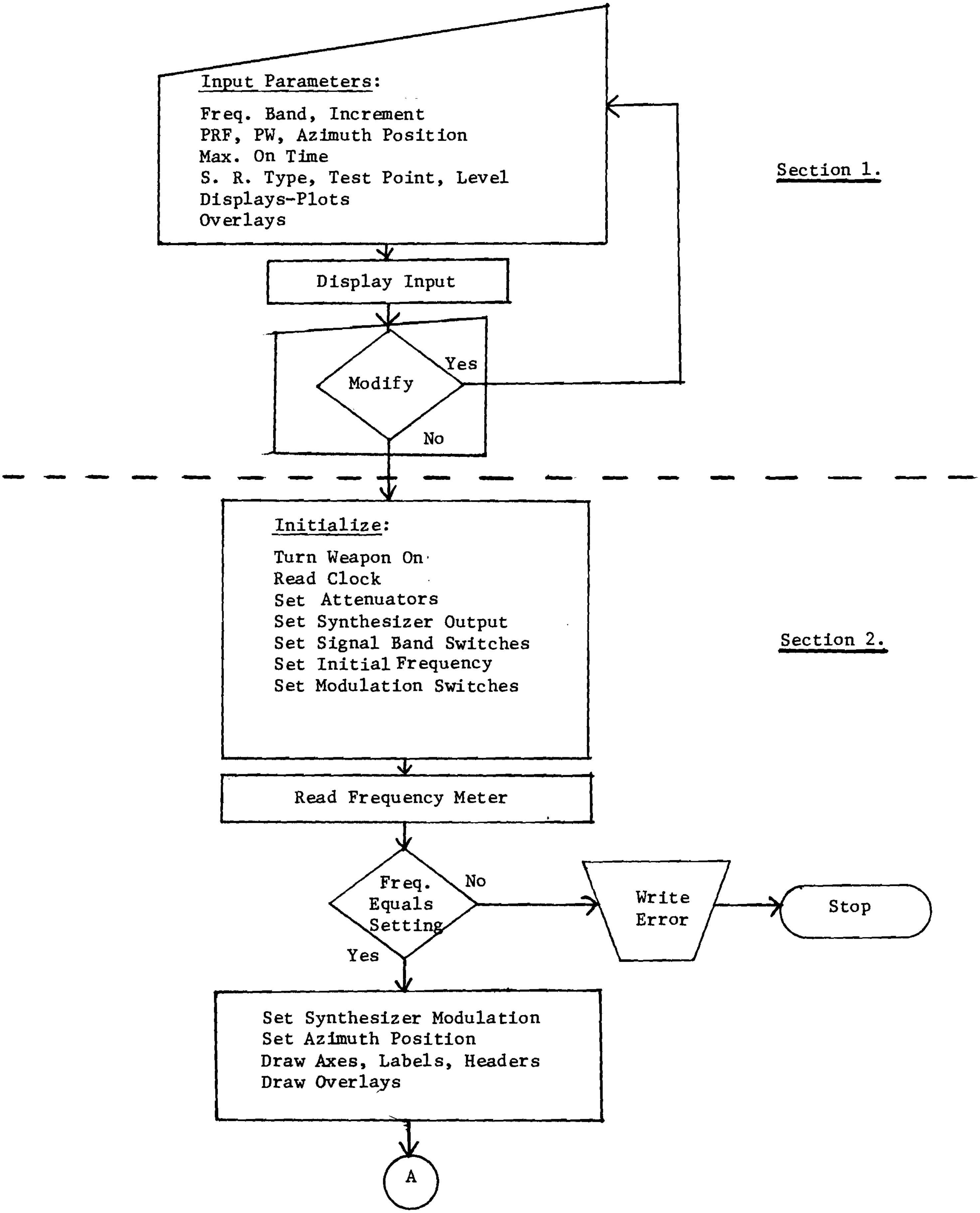


Figure B-2. Logical Implementation of Frequency Scan Test Procedure.

Section 3.

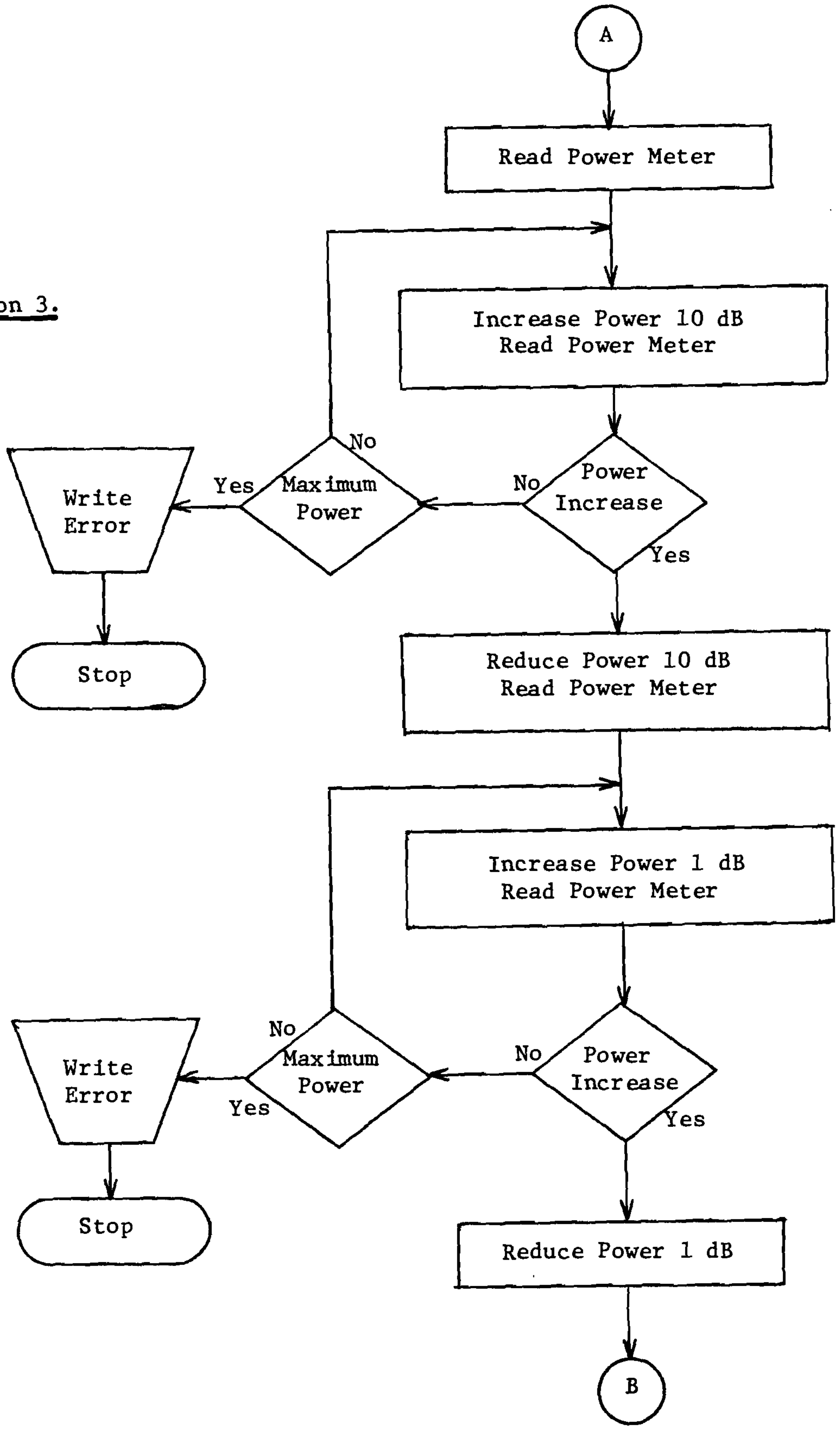


Figure B-2. (Continued)

Section 4.

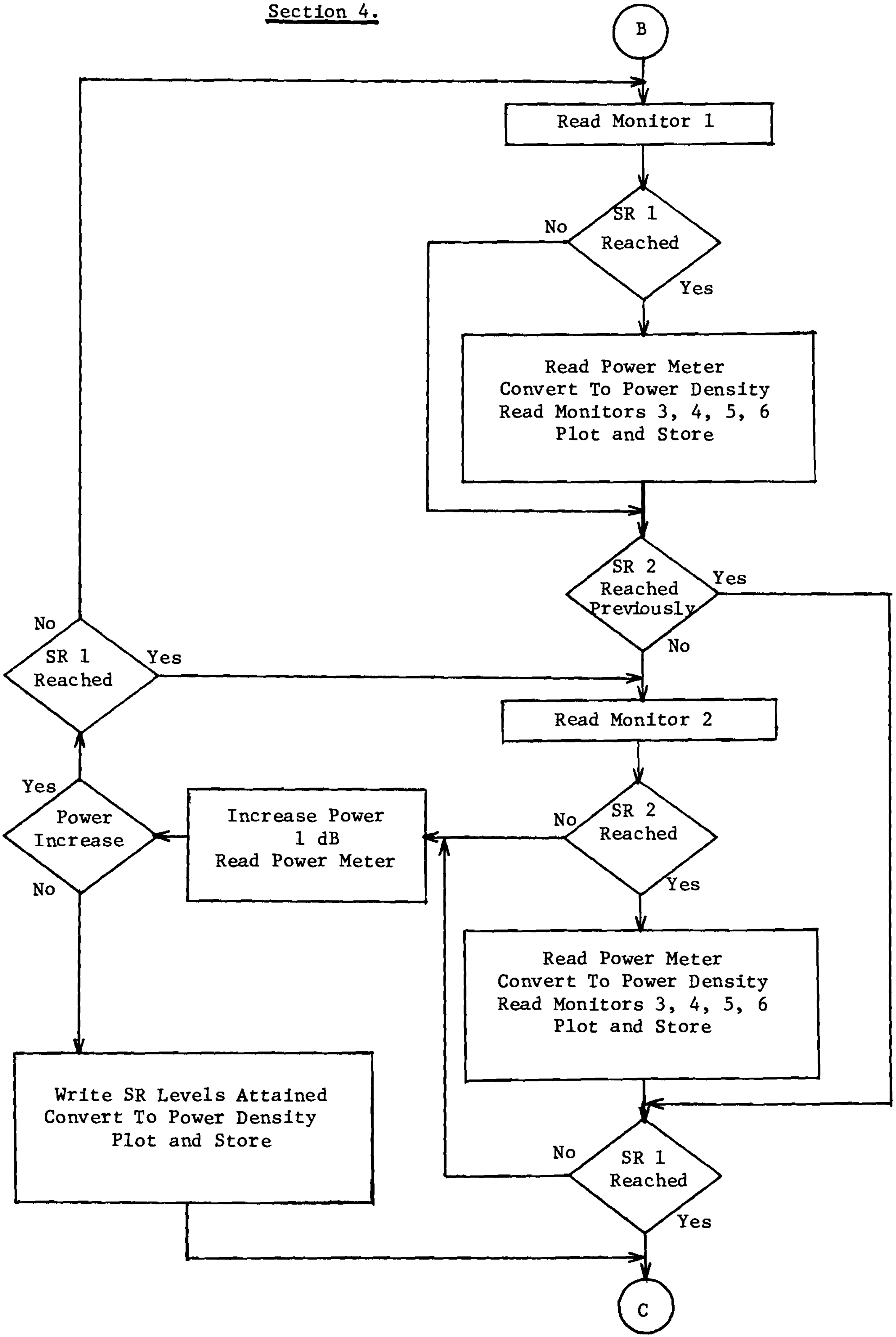


Figure B-2. (Continued)

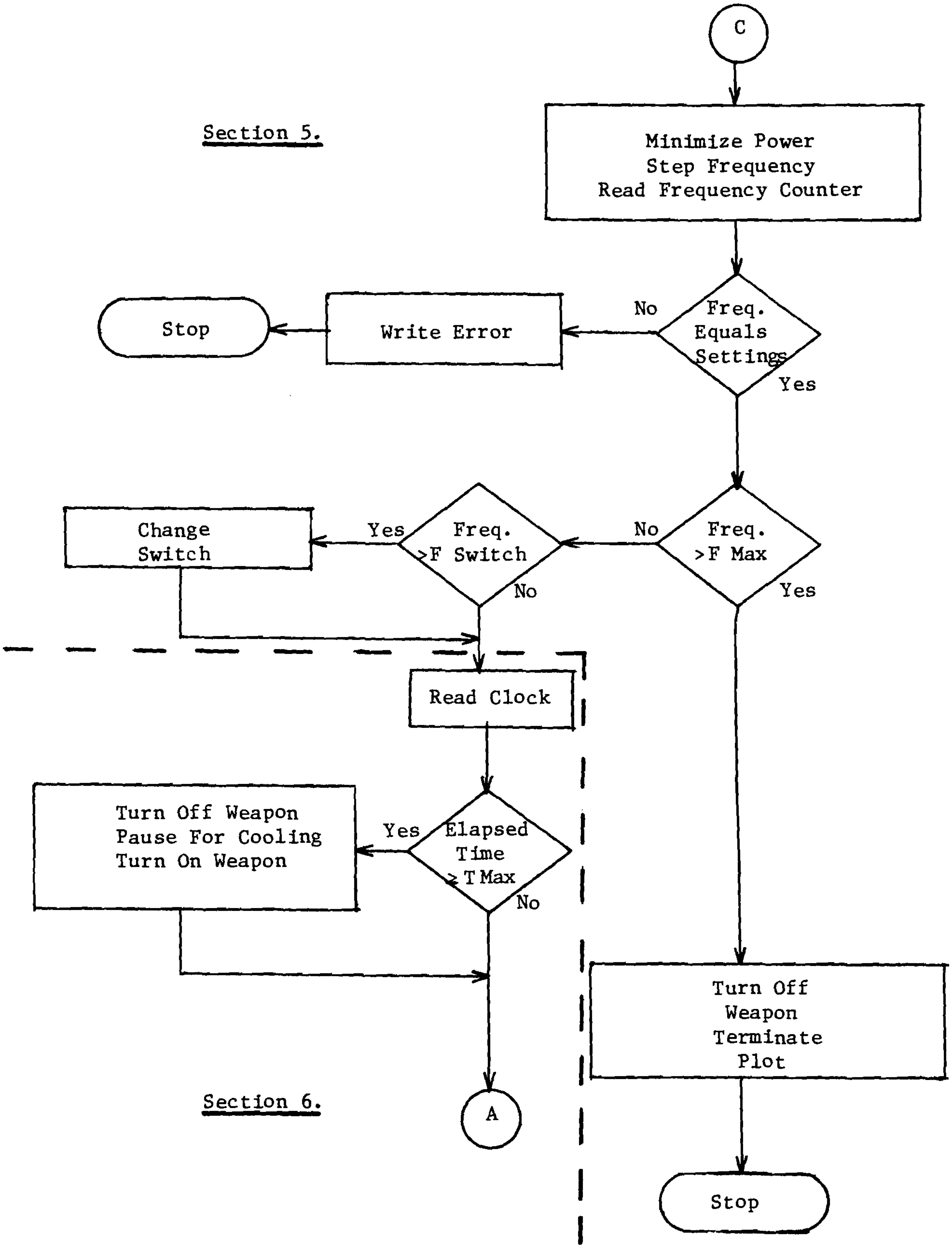


Figure B-2. (Continued)

does verify the increase, this check is then redone for smaller power level increases.

Section 4 comprises the actual measurement process. The test points are polled to determine if the first expected standard response has been achieved. If not, the next standard response is checked. Power is increased until all standard responses have been reached. If the transmitted power does not increase and one or more standard responses have not been achieved, the standard response level that is achived is recorded. As each SR is reached, the power level is converted to power density and the value plotted vs. frequency. After a respective SR is achieved, it is ignored on subsequent passes through the loop. The loop is exited after all standard responses have been achieved or after maximum transmitted power has been reached.

Section 5 contains the instructions that are required to step the frequency to the next value after the measurements have been completed at the present frequency. Power is reduced to the minimum level, and the new frequency is set by stepping the old value by the frequency increment. Checks are made to verify the new frequency. If the frequency counter does not agree with the command value, the weapon is shut-down and the test is ended. If the new frequency is in agreement with the commanded value, a check is made to ensure that the maximum desired frequency has not been surpassed. If the maximum frequency has been exceeded the weapon is shut-down and the test ended. If not, normal flow continues.

Section 6 consists of instructions that control the running time of the weapon testing. Time control is accomplished by reading the clock and comparing elapsed time with the maximum time the weapon may be powered-up. If maximum time has not been exceeded, program flow continues normally back through the main loop. However, if the maximum time has been equaled or exceeded, the weapon is cycled off and a cool down pause is effected after which the weapon is cycled on and program flow is continued.

After measurements have been taken and plots made at all frequencies of interest, the weapon is cycled off and the Frequency Scan Test halted.

The Frequency Scan algorithm serves to illustrate the inherent advantages to programming in a hierarchial fashion with separate and distinct levels of independent modules oriented in a tree-like structure. The implementation of the FS Test into a modular, table-drive, menu-selection approach would be straighforward.

The lower levels of a multi-level software hierarchy tend increasingly toward "mission-specific" related issues; namely, introducing algorithms of greater detail to address areas of smaller function. The focusing of attention from the higher level, including an FS Test itself, to the inclusion of the aforementioned six separate sections is an example of the increased detail that occurs with inclusion of lower levels of programs in the software hierarchy. Note that this progression to successively greater detail is made possible by the independent, functionally-oriented modular construction. The test operator selects those modules from archival memory (or writes them, if necessary) that provide the required function at the given level. Thus, interchangeability of modules and ease of modification of existing modules permits measurement sequences such as Frequency Scan to be tailored to particular situations as the need arises.

APPENDIX C[*]

EVALUATION OF THE HP1000/45 HARDWARE

RADC recently acquired a computer system, the HP1000/45, for use in automating various test procedures. The intent is for the computer to serve the following purposes:

(1) Assist in test definition and set-up through interaction between the test engineers and the operating software;

(2) Run the actual test under the supervision of an operator and simultaneously display in real-time or near-real-time a sufficient number of variables to allow the operator to determine how the test is progressing;

(3) Automate the data collection process;

(4) Support data analysis both during and after the actual test;

(5) Support a multi-user environment such that one user may employ a work station for pre-test planning, while another user may run a test from his work station, and a third user can perform post-test data analysis concurrently (See Figure C-1).

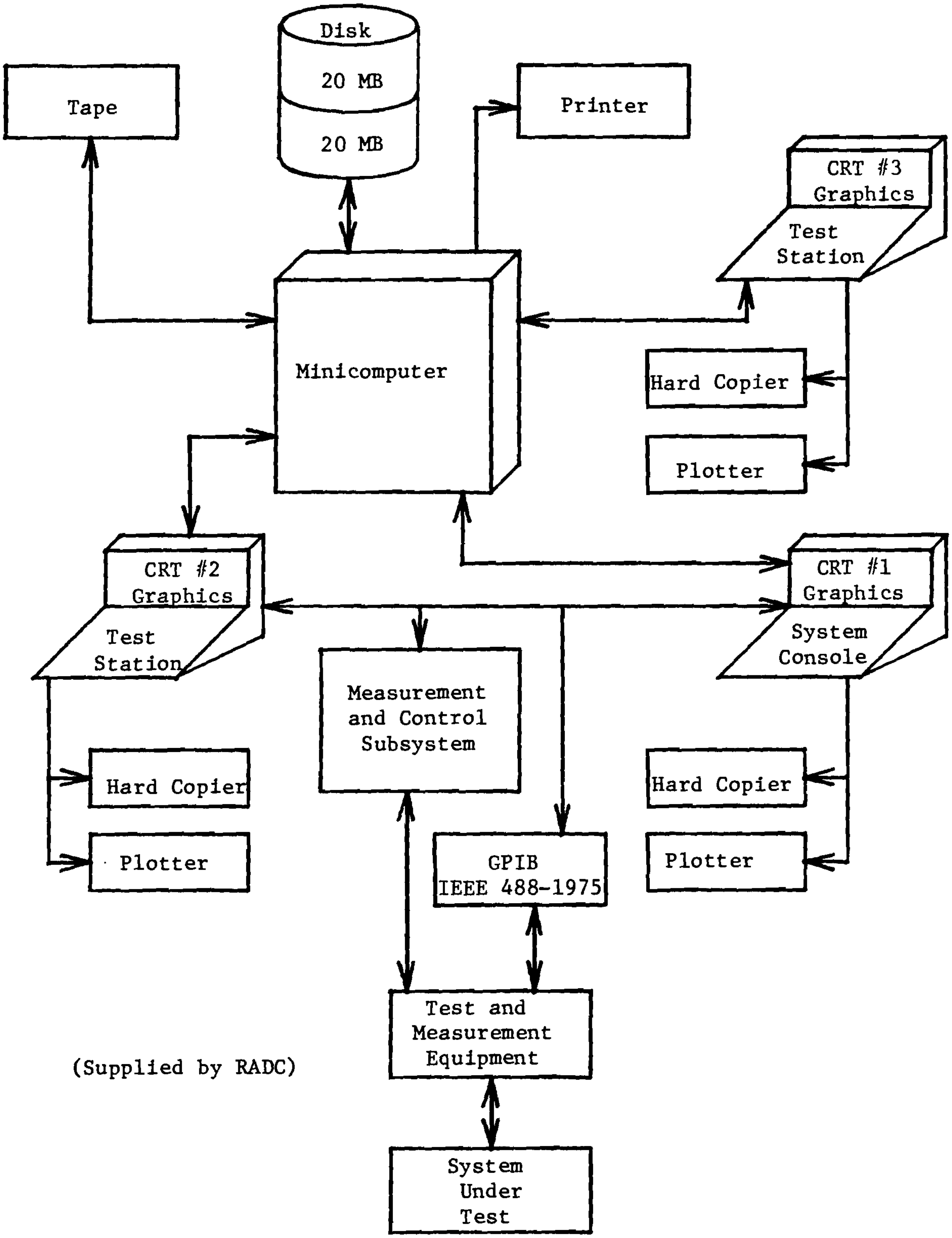
It is important to realize that each of the objectives mentioned above is achieved through combinations of both hardware and software. The present discussion will address the suitability of the hardware for meeting the above objectives.

Test definition and set-up is most heavily influenced by software considerations. The user-written software will dictate whether the set-up procedure is interactive and whether a menu selection approach is used to define the test program. The processor and its associated operating system will influence this capability in three indirect ways:

(1) The desire to simultaneously support multiple users for test set-up or other purposes dictates the use of an operating system which supports a multi-user environment. RTE-IV, the provided operating system for the HP1000/45, has this capability.

---

[*] Prepared by R. W. Rice.



(Supplied by RADC)

Figure C-1. Conceptual Hardware Arrangement for ADACS.

(2) The speed with which multiple users can be supported is related to the speed of the processor's various elements. This fact has been recognized and the Arithmetic Logic Unit (ALU), the Floating Point Processor (FPP), the Fast FORTRAN Processor (FFP), and the Memory have been chosen to maximize computational speed.

(3) The efficiency with which a multi-user environment can be supported is in part tied to the amount of available local memory.

There may be some noticeable constraints on system use imposed by available resident memory. Consider the situation in which two of the terminals are being used for data analysis and one terminal is being used to observe graphically the data for a test in progress. Such an arrangement would most likely utilize the following pieces of software:

Software	Resident Memory Requirements
RTE-IV	48 Kbytes
Graphics 1000 + User Routine	20 Kbytes (typical)
Image 1000	40 Kbytes (typical for 2 routines)
Total	108 Kbytes

The above total does not include the actual data, which could amount to several thousand words, being manipulated by the Image 1000 data base management program. In short, the desired mode of operation may lead to available memory problems.

The ability of a processor to perform real time control and simultaneously display test data is linked to three areas:

(1) The speed of both the hardware and software,

(2) The availability of a foreground/background structure in the operating system software, and

(3) The amount of available local memory.

The architecture of the processor is well suited to real-time process control; however, two limitations appear to be present. The processor and its associated memory have been chosen for their high speed characteristics. To realize the processing speed of which these elements are capable, it will be necessary to use the writable control store and the user control store to handle some of frequent, but traditionally slow, processes such as device I/O. Such programming is done at the assembly level.

HP's operating system, RTE-IV, does provide a foreground/background structure, and therefore, if a certain test sequence has a particularly high speed data collection or control sequence, the routine data display routine could be run as a background routine.

Where immediate display of test data is not required, it is possible to allow the processor to act as a pipeline between the sensors and the system's mass storage facilities, but when immediate display is required, then reasonable amounts of local memory may be required. This memory may be used either for raw data which must be processed for display of correction factors which are used to adjust the raw data. Inputs from RADC personnel suggest that correction factors may require arrays of up to 40,000 words. The example cited previously indicates that in a multi-user enviroment storage of 40,000-word arrays is not possible with the current memory allocation.

Automating the data collection function requires both time efficient I/O routines and reasonable amounts of data storage capability. If high speed sampling is anticipated, it should be recognized that such is facilitated by micro-coded device handlers which would employ either the user control store or the writable control store. This speed enhancement is acquired by sacrificing the transportability associated with high-level programs.

The problems of local memory availability have been discussed above, but there is at least one potential problem with the mass storage facility. The mass storage facility consists of two disk drives and one magnetic tape unit. The two disk drives will allow large amounts of data storage and will also permit disk-to-disk data transfers. The single magnetic tape unit provides a comparable amount of data storage, but since there is only one tape tansport, tape-to-tape transfers can be performed only through the use of an intermediate storage medium such as a disk. Accommodation of such transfers may require the development of special formatting routines.

The ability of the system to support data analysis concurrently with test operation is part of the multi-user operation previously discussed, and the more users involved, the less space, i.e., memory, there is for each user. There is, however, another aspect of the multi-user consideration. Users of test sytems similar to the one planned at RADC have expressed reservations about having multiple users on the system during a test. The stated concern is that a user not involved in the actual test could cause a processor fault which would interrupt the test sequence. This is certainly possible, but the ideal operating system would not allow this to happen. (We have not attempted to determine how close RTE-IV comes to the ideal.)

The capacity of the computer to support real time control and data collection is enhanced by the use of a pre-processor as part of the measurement and control system. The pre-processor is the heart of the real time control program since it is through this package that outgoing control signals and incoming data flows. For low speed applications, the pre-processor may be programmed in FORTRAN, but for high speed applications assembly level coding will be required. Unfortunately, assembly level coding will introduce an element of machine dependence into the software.

Peripheral to the measurement and control system are the analog and digital inputs and outputs. Twelve bit resolution is used for both analog inputs and outputs; this degree of resolution appears to limit the dynamic range of these variables to approximately 64 dB (assuming that compression systems such as logarithmic amplifiers are not used). The sampling rate on the analog channels is a function of the number of channels in use, but the nominal rate is on the order of 1 kHz. This rate is certainly fast enough to observe even the traditional transients on systems such as servos and guidance systems, but it may not be adequate to observe either natural or induced oscillations on such systems.

Figure C-2 shows the configuration of the HP1000/45 hardware which is to be used to automate RADC's measurement facility, and Attachment 1 lists the important characteristics of each of the major hardware items.

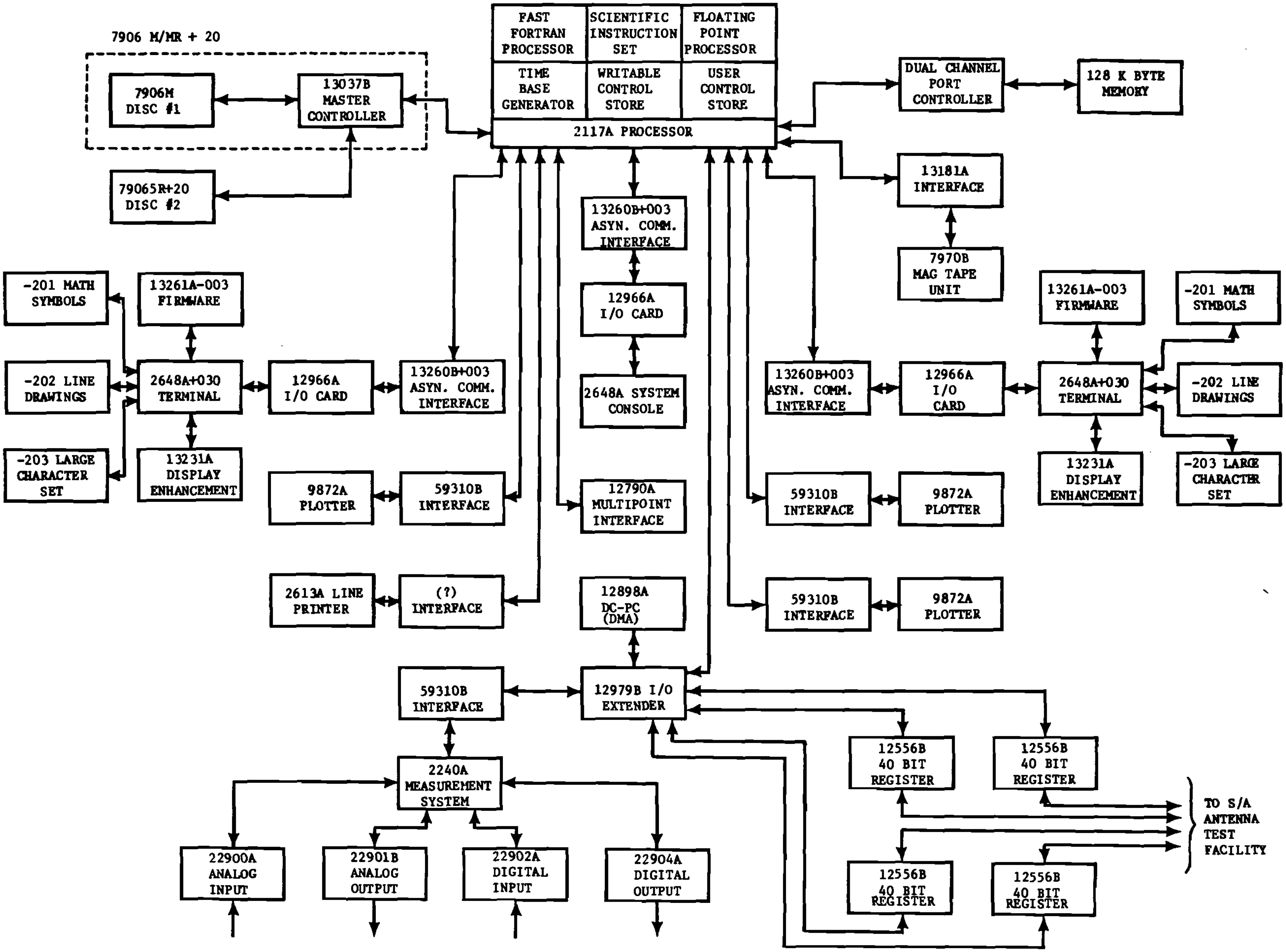


Figure C-2. Hardware Interface for the EMCAF.

ATTACHMENT 1

SUMMARY OF PROPERTIES OF MAJOR ELEMENTS OF ADACS.

PROCESSOR

Arithmetic Logic Unit (ALU)
- Data Word: 16 Bits
- Instruction Expansion: Up to 176 Microprogrammed Instructions
- Interrupts: 50 Distinct Levels
- Control Word: 24 Bits

Floating Point Processor (FPP)
- Word Size:
  - 32 Bit (Single Precision)
  - 48 Bit (Extended Precision)
  - 64 Bit (Double Precision)
- Functions:
  - Add/Subtract
  - Multiply
  - Divide
  - Format Conversion

Scientific Instruction Set (SIS)
- Functions:
  - Sine
  - Cosine
  - Tangent
  - Arc Tangent
  - Hyperbolic Tangent
  - Exponential (e^x)
  - Square Root
  - Natural Logarithm
  - Base 10 Logarithm

Polynomial Evaluation Instruction (PEI)
- Functions:
  - $A_0 + A_1 + A_2X^2 + \ldots + A_nX^n$
  - Ratios of Polynomials

Fast FORTRAN Processor (FFP)
- Functions:
  - Moves
  - X^n
  - Rounding/Normalization
  - Data Packing/Unpacking
  - Complementing
  - Transfer of Control (GO TO)
  - Evaluation of $(1-X)/(1+X)$

Time Base Generator (TBG)
- Time Increments: 100 microsec to 1000 sec in decade steps
- Programming: Assembly level

Memory
Type: Mos/Ram
Amount: 128 Kbytes
Cycle Time: 350 ns.

Writable Control Store
Amount: 1K words
Cycle Time: 175 ns.
Programming: Assembly Level

User Control Store
Amount: 512 words
Cycle Time: 325 ns.
Programming: Assembly Level

I/O Extender
Number of I/O Slots: 16

MASS STORAGE

Disc
Storage Capacity: 39.2 Mbytes

Magnetic Tape
Speed: 45 IPS
Data Rate: 36 Kbytes/sec.

WORK STATIONS

Terminals
Type: CRT (Raster Scan) and Keyboard
Display Capacity: 37 lines of 80 characters (alphanumeric)
720 dots by 360 runs (graphics)
Automatic plotting
Hardware Pan and Zoom
Character Sets: Roman
Math
Line
Large Characters
Enhancements: Blinking
Half-Bright
Underline

Plotter
Plot Size: 11 in. x 15.75 in. (Maximum)
Type: Pen

Printer
Types: Impact (Alphanumeric)
Xerographic (Graphics and Alphanumerics)

MEASUREMENT SYSTEM

Analog Inputs (A/D)
- Number of Inputs: 16 differential or 32 single ended
- Resolution: 12 Bit
- Sampling Rate: 20,000/sec.

Analog Outputs (D/A)
- Number of Outputs: 4
- Resolution: 12 Bit
- Rate: 5000 Points/sec.

Digital Inputs
- Number of Channels: 32
- Rate: 11,000 Points/sec.

Digital Outputs
- Number of Channels: 32
- Rate: 2,200 Points/sec.

APPENDIX D[*]

ADACS DATA ANALYSIS AND DISPLAY REQUIREMENTS

D.1 INTRODUCTION

The data analysis and display routines required for ADACS are primarily intended to reconfigure the measured "raw" data into a more suitable format for interpretation and storage. In such systems, large amounts of data will be acquired over relatively short periods of time. Thus the system must have the capability to display the data at various stages of processing so that a decision can be made as to whether it will be kept or discarded. It is generally most convenient and efficient to store the raw data and perform the data processing at some later time. (The raw data can be stored on a disc, magnetic tape, or an instrumentation recorder.)

In a system operating in both an acquisition and a processing mode, it is important that the system operator be careful so as not to overburden the computer during a data taking operation since the data is available only for a finite time period. The data manipulation routines can always be performed at a later time when the likelihood of interference with the data acquisition operations is at a minimum.

Once the raw data has been obtained, it must be corrected to include calibration factors. A correction capability will require development of software to accommodate linear multiplying factors and/or fixed increment addition or subtraction. The raw data should be printed out on a hard copy before correcting for calibration.

The system must have the capability to merge multiple data sets into a single file and the operator must have the option to edit a given file. Thus, operations such as addition, deletion, and insertion of data will be necessary. Software can be readily designed around the Hewlett-Packard Image/1000 data base management system for the above requirements. Image/1000 contains many of the routines necessary to meet the needs; it will only be necessary to combine them in the proper sequence to accomplish the required task.

---

[*] Prepared by L. A. Jackson

A general sort routine will be necessary since the merging of data files may require re-ordering. For example, suppose that a group of data are taken that fall within the range of another data set taken previously. This might be done to investigate more thoroughly the behavior of a subset of another data set. In order to have a continuous single data set when combined, a sort routine will be required to arrrange the data of both sets in increasing or decreasing sequential order. This routine should be capable of sorting using either X or Y as the selected sort variable. Since data is often taken in overlapping regions, the capability to merge the data points in the overlapping regions must also exist. It is desirable to have operator control how the data should be merged. He should have the option to select the worst case, the average, or a "best estimate" of what he thinks the overlapped region should be. Thus, it will be necessary to employ software that will locate the overlapping data entries and will prompt the operator on the selection of the method by which the data is to be merged. The system must provide the user with a choice between examination of the data entries in the overlapping region directly by listing the data file, or by plottinng the data points on the graphics terminal. The operator would then select the appropriate merge routine, re-examine the new data listing and/or plot, and continue the process until the data is suitable for storage or final plotting.

Before the data can be conveniently plotted or compared with other data, the data units must be consistent. Therefore, routines to convert frequency to Hz, kHz, MHz, or GHz and routines to convert power density to W/m^2, mW/cm^2, or dBm/cm^2 will be necessary. Other helpful conversion routines which should be considered include English to Metric and Degrees to Radians and vice-versa.

D.2 DATA ANALYSIS

In order to gain the maximum amount of information from the data, analysis routines should include as a minimum the following:

Curve Fitting
Extrapolation and Interpolation
Curve Smoothing
Differentiation and Integration

Statistics
Fast Fourier Transform (FFT)
Probability Distribution
Auto/Cross Correlation
Matrix Operations

With the exception of the extrapolation, interpolation, and curve smoothing routines, all of the above routines are available from HP's Library of Contributed User Software (LOCUS). (The attached cross referenced index of LOCUS provides the part numbers for the routines.) These routines are available on paper tape, magnetic tape, or mini-cartridge, and can be used without modification or can be used as a starting point for further development.

Curve fitting algorithms such as weighted averaging, least squares regression, and polynomial regression are available from LOCUS. Extrapolation, interpolation, and curve smoothing routines will require additional development. (A technique for curve smoothing employing the use of software simulated digital filters is described in Digital Signal Processing by Alan V. Oppenheim and Ronald W. Schafer.)

General statistical routines including mean, deviation, and correlation analysis and probability distribution routines such as Normal distribution and Chi-Square Goodness-of-Fit tests are available from LOCUS as are general purpose matrix operations and a Cooley-Tukey FFT algorithm.

Also included in the attachment is a listing of the routines available in one software package, STAT-PACK, that can be obtained on magnetic tape for $175 from HP's softfware center. This single package contains many of the necessary routines for the data analysis section. Although much of this software was developed for the HP2100 and 21MX computers, upward compatibility is expected to allow program operation on the HP1000 System 45.

An alternate approach to obtain the above software routines is to purchase the Sensor-Based DAS Utility Library for the HP1000 computer. This software package, HP part number 92400A, is a collection of routines and functions providing ready-to-use calculation capabilities. Data interpolation, statistical analysis, curve fitting routines, and data integration are among the applicable programs included.

A more specialized routine or group of routines than is available from Hewlett Packard is necessary for complete data analysis. This routine, referred to as delta-dB, specifically shifts the breaklock curve in accordance with other modulation parameters, different azimuth configurations, various target conditions, etc. The magnitude of the shift, Δ-dB, is determined from either a look-up table or from a curve-fit of data taken in preliminary tests that relate breaklock to the particular parameter. For example, a plot might show a curve relating breaklock to frequency and power density for a given modulation (pulse width and pulse repetition frequency) at a given distance. This will usually be the worst case modulation, azimuth, target condition, etc., in terms of breaklock. Since it is desirable to show the threat level of emitters of various modulations, frequencies, polarizations, bandwidth, etc., the breaklock curve will require shifting to show the relative threat at a different pulse repetition frequency (PRF), for example. As a preliminary test, data would be taken to show a dB change in the standard respone variable relative to the PRF at the worst case frequency, azimuth, etc. Therefore, to determine the threat level at the new PRF, the dB change in the standard response variable taken from a look-up table or from the curve-fit would then be added to the breaklock curve to show worst case plot of breaklock relative to frequency and power density at the new PRF. Figure D-1 gives a graphical representation of the above routine. The software required to accomplish the delta-dB function will require development around curve-fitting routines, linear multiplying factors or fixed-increment-addition routines, sorting routines, and plotting routines.

In order to graphically depict the threat level of known emitters, as in Figure D-1, the system must be capable of selecting emitters according to name, ELINT designation, frequency, PRF, pulse width, polarization, beamwidth (horizontal and vertical), scan characteristics, peak transmitter power, antenna gain, and the typical platform characteristics. This information, obtained from the ECAC data base, can be organized through the Image/1000 data base management system to provide the necessary operations.

The overall software requirements of the data analysis and display routines can be summarized in the following manner: it is either available from Hewlett-Packard as a supplemental software package; it exists

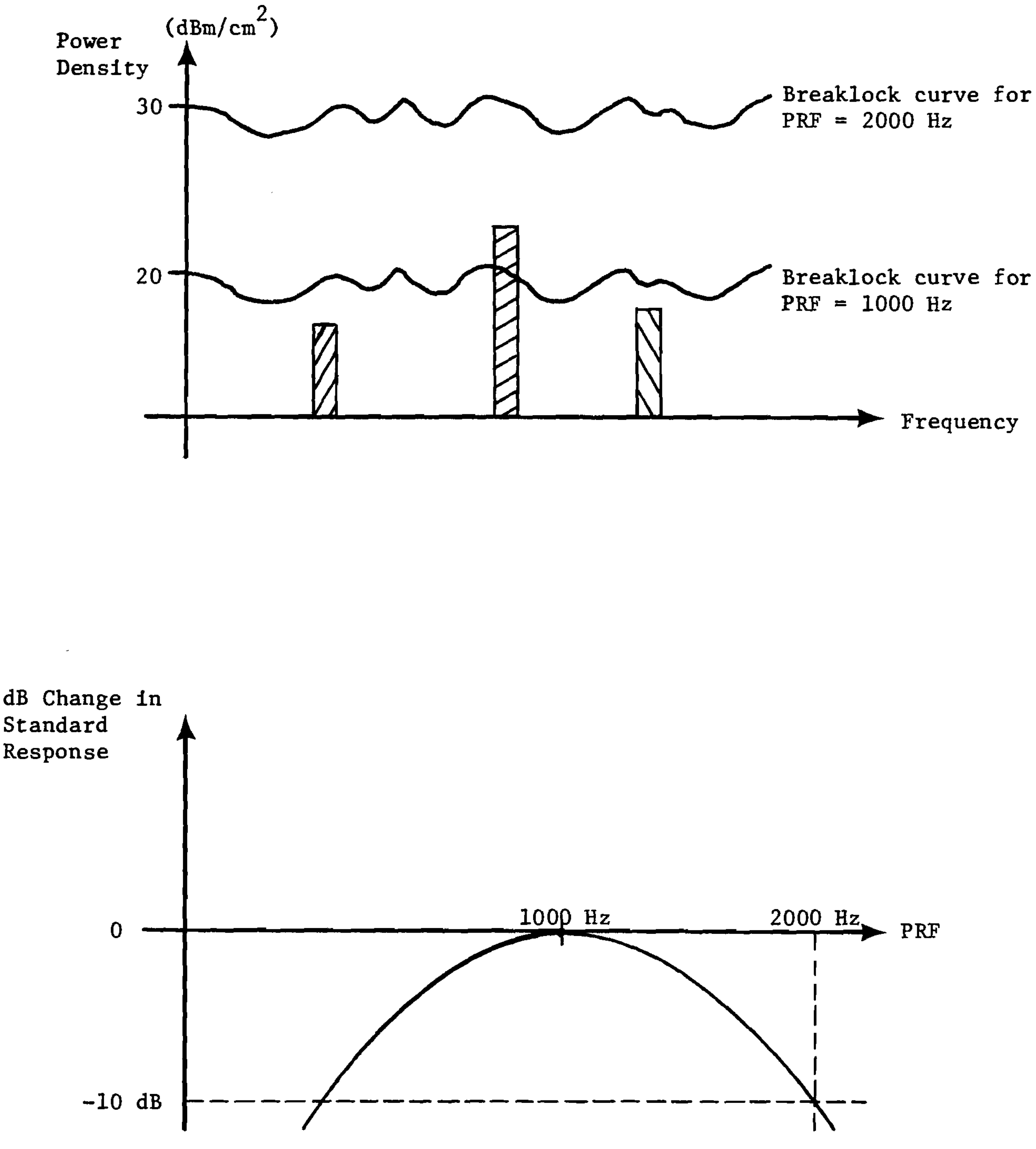


Figure D-1. Graphical Depiction of Delta-dB Operation. (The 10 dB change in response in the lower graph gives a corresponding 10 dB change in the breaklock curve of the upper graph.)

as part of the original purchase; or it will require additional developmental effort. This information is summarized in Table D-1.

D.3 DATA DISPLAY

The basic requirements for the data plotting routines can be met with a minimal amount of supplemental software development effort. The data plotting requirements are essentially met by the capabilities of the HP2648A graphics terminal, HP9872A graphics plotter, and the HP7245A plotter-printer, all of which are supported by the HP Graphics/1000 software package. With Graphics/1000, plotting routines can be developed independent of the hardware which allow the operator to select the particular graphics output device for any given plot.

The Graphics/1000 software package consists of a set of modular plotting support routines for FORTRAN, Basic, or Assembly language programs. It contains 53 device-independent plotting subroutines including a user-defined coordinate system, automatic axis and grid drawing, and labeling. Most of the frequently used subroutines are included to enhance quick and easy graphics program development. These subroutines can be accessed and used to meet most plotting requirements including linear-linear, linear-log, log-log, and polar plots.

The graphics capabilities of the HP2648A, HP9872A, and the HP7245A are described in the following paragraphs.

The HP2648A Graphics terminal is a raster scan alphanumeric/graphics display. For quick and easy graphics generation, an automatic plotting capability exists whereby the operator follows a simple menu. Once the data parameters are defined, the plot is produced by a single keystroke. Although automatic plotting tends to be slower, it provides the flexibility to generate plots of different types without modification of existing software. The "Rubber Band Line" capability allows trial graphics with or without the computer by drawing line segments according to cursor control. This feature would be well suited for data merging in overlapping regions. For example, the operator could merge curves in a "best estimate" fashion through use of the cursor control. Another important feature of the 2648A is that the alphanumeric and graphic display memories are independent which allows interaction with the computer without obscuring the graphics display. Since

TABLE D-1

SOURCES OF REQUIRED ADACS ROUTINES

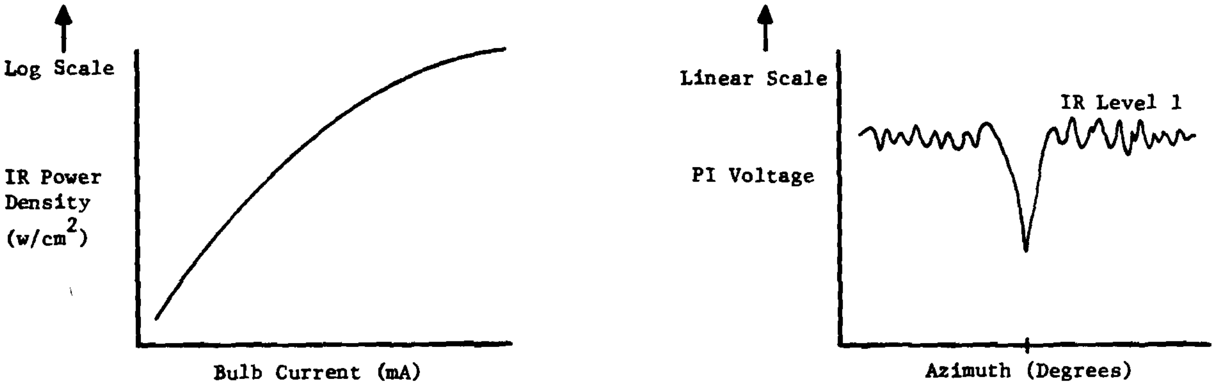
Required Routines	Included in Original Equipment Software	Available from HP Software Center	Requires Development
Add, Delete, Sort	x		
Merge			x
Display	x		
Storage			x
Cal. Factor Correction			x
Plotting	*		
Curve Fitting		x	
Extrapolation			x
χ^2 Goodness of Fit		x	
Auto/Cross Correlation		x	
Differentiation			x
Integration		x	
Statistics			x
Matrix Operations		x	
FFT		x	
Unit Conversion			x
Plot Enlargement	x		
Curve Smoothing			x
ΔdB Routines			x
Plot with Symbols	x		
ECAC Data Base	x		

*Most necessary subroutines are provided by Graphics/1000 it is a raster scan terminal, the capability exists to modify selected portions of the graph or plot. The graphics memory is a dot matrix of 720 x 360 points. The alphanumeric memory can retain 1.5 pages of text or up to 37 lines of 80 characters. Another useful feature allows the user to zoom and pan a graphics display and magnify the selected portion up to 16 times. User-defined soft keys can be set up to issue a string of characters or several control sequences. With the mini-cartridge option, the 2648A can store up to 110,000 characters, perform rapid data transfer, and perform high speed bi-directional searches. This capability is well suited for sorting data or graphics information and is an excellent substitute for paper tape.

The HP9872A graphics plotter is a microprocessor-based HP-IB (IEEE 488-1975 compatible) plotter that features programmable 4 color pen selection, error free off-scale data handling, point digitizing, selectable pen velocity, 7 dashed line fonts, and 5 built-in character sets. Window plotting permits the handling of off-scale data by halting the plot at the mechanical limit and resuming the plot automatically when on-scale data are received. The plot size can be set as desired by adjusting the scale controls. Symbol mode plotting allows the user to select any ASCII character to be placed at a given data point location. With only a few commands, X and Y tick marks or a full grid can be drawn. Again, with Graphics/1000, a plot routine can be developed independently of the device and the HP9872A is supported by Graphics/1000.

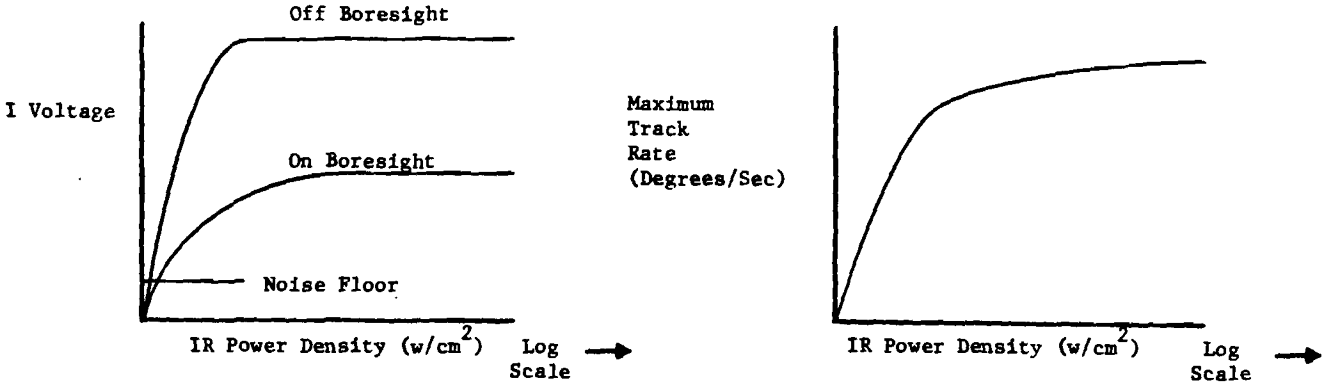
The HP7245A plotter-printer is an HP-IB desktop thermal plotter and printer. An optional CRT raster dump capability provides hard copy records of the HP2648A terminal. Another important feature of this device is the capability for up to 5 meter (16.4 ft.) long axis plots. This device is also supported by the Graphics/1000 software and is therefore capable of printing or plotting under control of generalized graphics software.

Further requirements of the data display software include a capability for plotting multiple curves on the same sheet and axes. It must also be capable of producing multiple plots of the same type without additional user input. The capability to plot continuous curves on the same scale from separate data sets is necessary. Samples of the general graphics outputs are shown in Figures D-2 through D-4.



Target Calibration

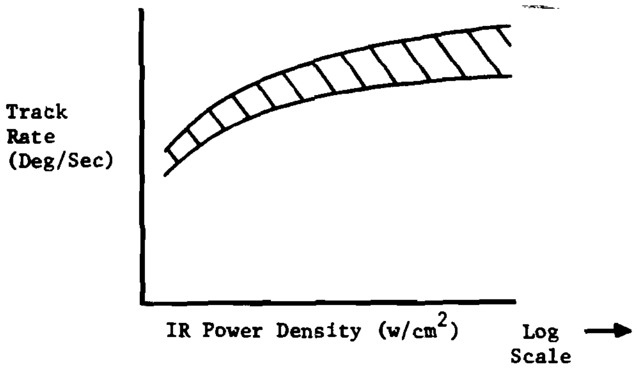
Seeker Characteristics



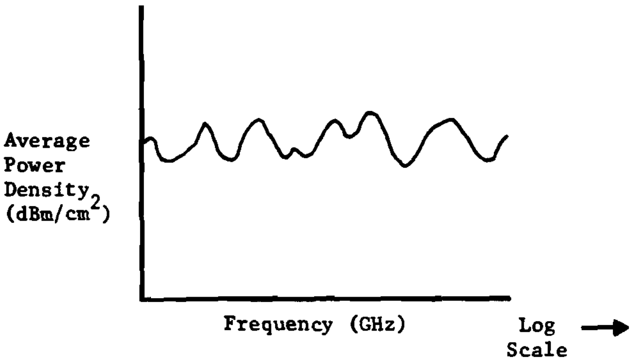
PI Dynamic Range

Track Rate Characteristics

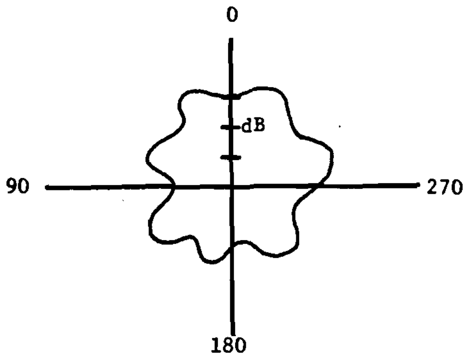
Figure D-2. Examples of Graphical Displays Associated With Preliminary Tests.



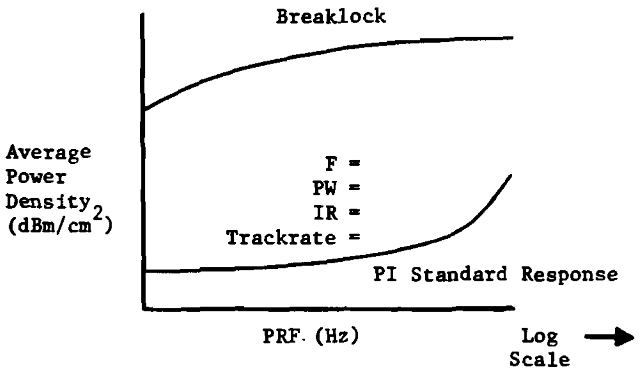
Track Rate Capability Envelope
(Preliminary tests - no EMI)



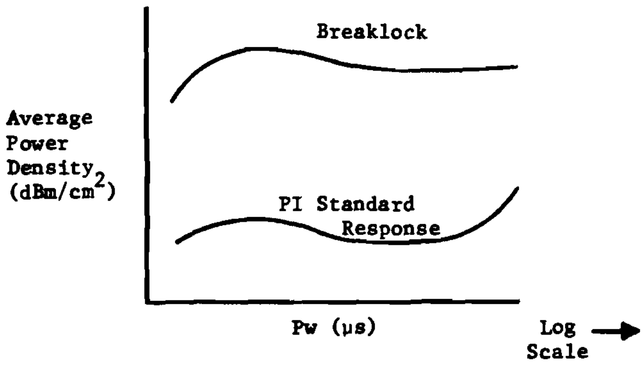
Frequency Scan



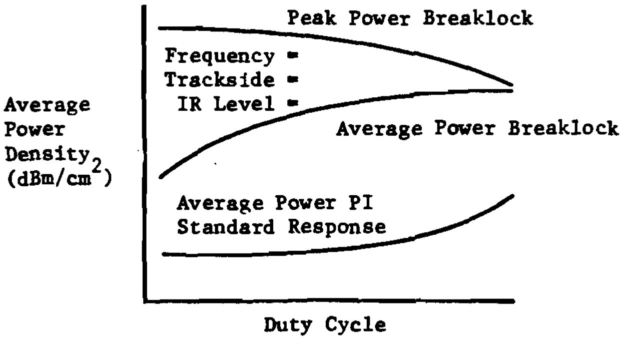
Azimuth Scan



PRF Scan

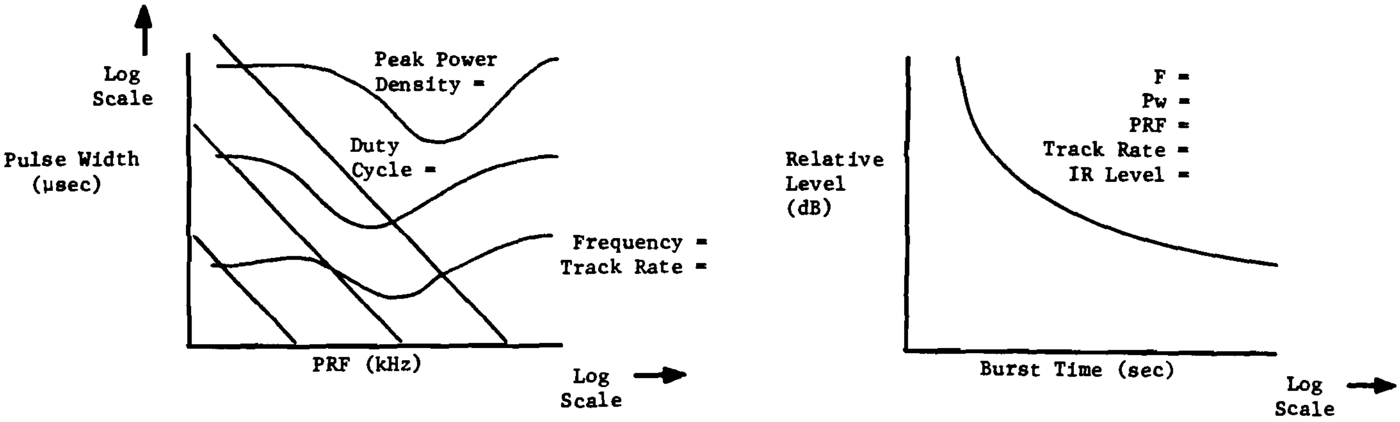


PW Scan



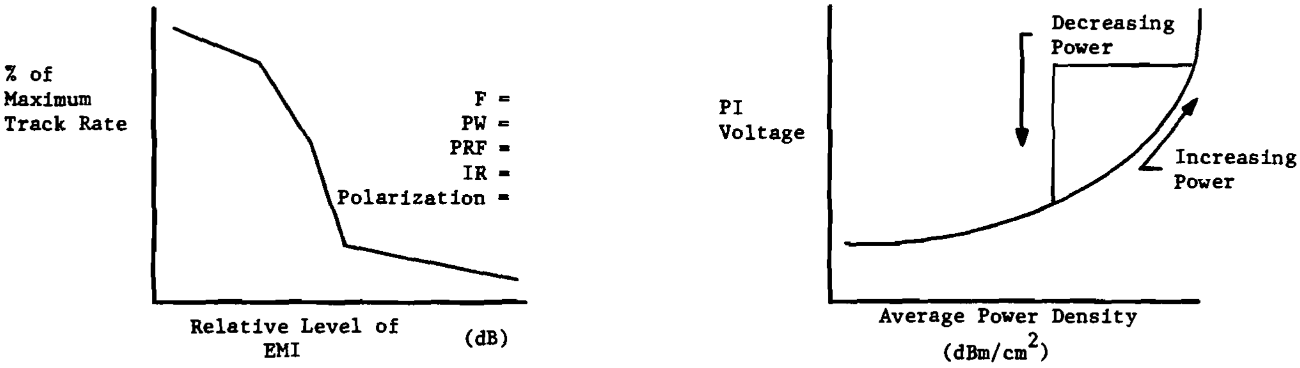
Duty Cycle

Figure D-3. Examples of EMI Test Data Displays.



Contours of Constant Breaklock

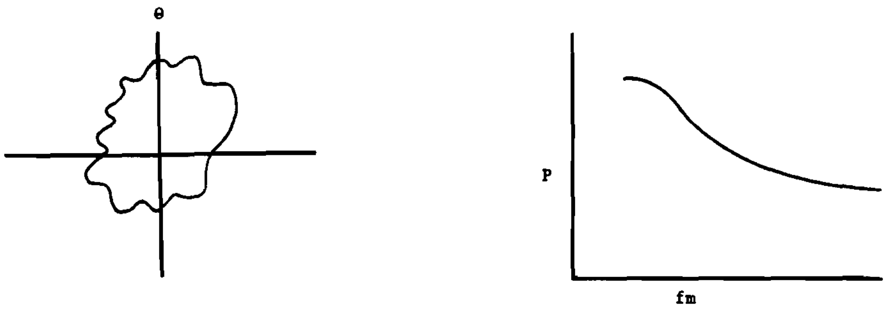
Burst Time Sensitivity



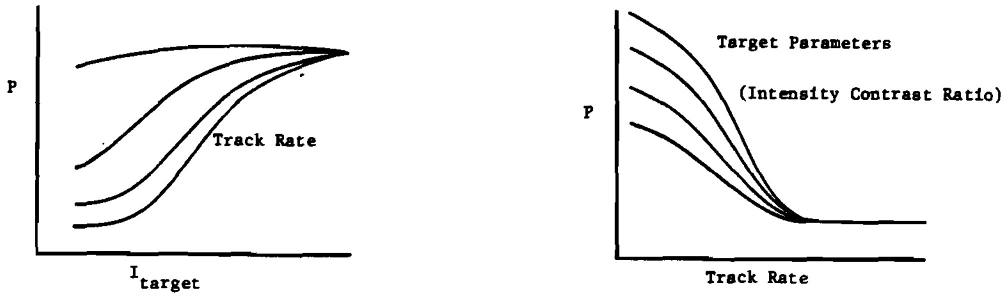
Track Rate Degradation
(Fixed IR Level)

PI Interference
Characteristics

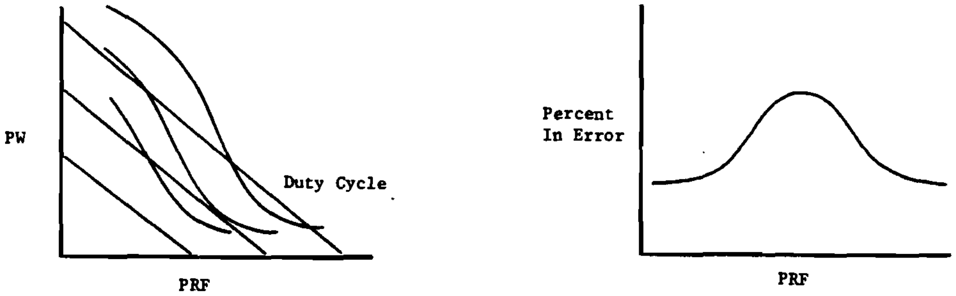
Figure D-3. Examples of EMI Test Data Displays. (Continued)



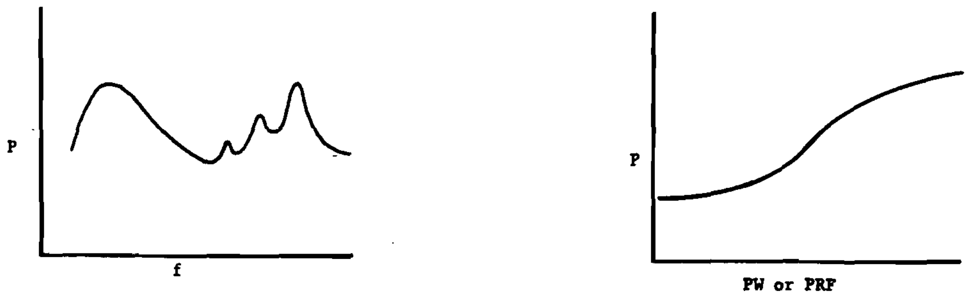
Azimuth Scan Modulation Frequency Scan



Breaklock Track Rate



Duty Cycle Contours Pulde Count · (No EMI)



Frequency Scan PW/PRF Scan

Figure D-4. Illustrative Examples of Data Analysis and Display Requirements.

In addition, the graphics software must include the ability to produce overlap plots, as shown in Figure D-5.

To achieve all the required capabilities, the data display software will require some additional development effort. This effort is reduced considerably by the Graphics/1000 software package in that most of the required graphics subroutines are included in the existing software. The major effort will consist of selecting the appropriate subroutines to perform the desired tasks.

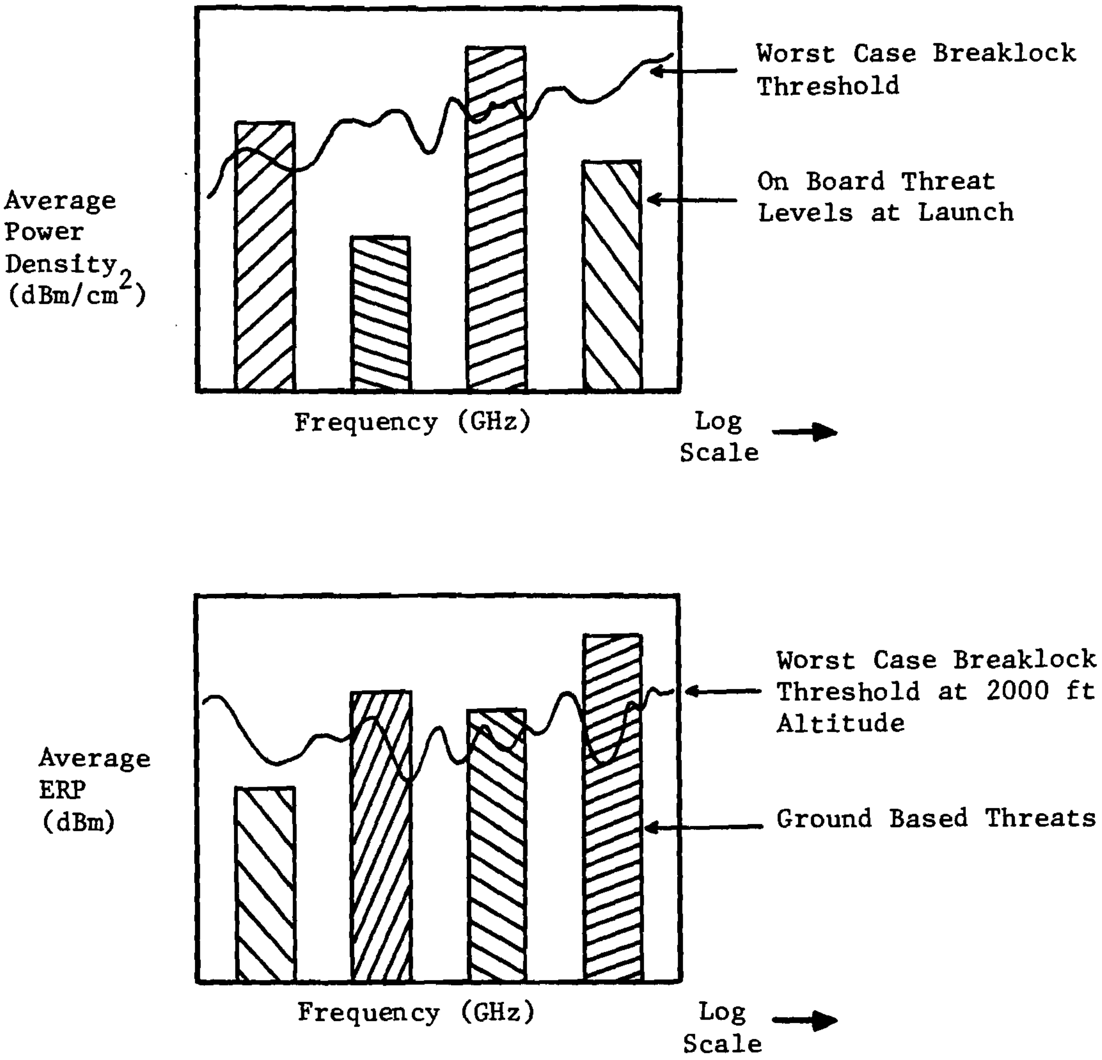


Figure D-5. Representative Vulnerability Analysis Results.

APPENDIX E[*]

TABLE-DRIVEN MENU APPROACH TO EMS/V SOFTWARE

As noted elsewhere, the ADACS software must efficiently control various programs which can perform a variety of test functions in an anechoic chamber, e.g., chamber calibration, antenna pattern measurements, EMS/V tests, data acquisition and reduction. Each different test problem must be supported by an appropriate program, which is in turn supported by numerous subprogram modules (a library). The objective of a composite design is to produce highly reliable and efficient programs/modules through a structured analysis, design, coding, implementation, and documentation process (see Appendix F).

Composite design consists of a set of design measures, strategies, and techniques, constrained by certain guidelines and tradeoffs. The tradeoffs usually considered are development cost, run time, and quality. The quality of the design is measured by its reliability (number of program "bugs"), maintainability (effort and time to fix "bugs"), modifiability (ease of extending the code without requiring extensive changes elsewhere in the program, i.e., the ripple effect), generality (scope), useability (human factors), efficiency (execution speed, storage requirements, resource usage, and mean-time-to-failure), and "portability" (ease of transferring to another system).

The basic recommended concept is to use small, highly independent, single function modular programs. Such programs offer reduced complexity and are, therefore, more reliable, of higher quality, more efficient, less costly, cleaner, and more high optimized than non-modular programs. Each module has a name and is referenced by that name, is lexically bounded together with START/END statements, and returns to the calling program. A module is characterized by the function it performs, i.e., the transformation of input into output, by its internal logic or program flow, and by its interfaces with other modules.

The key measures of the value of a module are its strength and coupling. Module strength refers to the relationships within the module; module coupling refers to the relationships to other modules. A module should have high strength and low data coupling. The strength of a

---

[*] Prepared by J. D. Norgard.

module can be increased by maximizing the relationships within the module. This maximization can be achieved by coding modules which have a single function or goal and, therefore, perform a single transformation of the input data to produce the output data. Module data coupling can be reduced by minimizing the relationships between modules; this can be achieved by coding modules which pass all IO as parameters.

Other measures of the value of a module are its simplicity (which allows flexible data, but avoids flexible functions), size (maximum recommended size is 100 lines of code, since small subdivisions allow for ease of testing, understandability, and independence), predictability (behavior is identical on similar inputs and is, therefore, independent of the environment), initialization (place as far down in structure as possible), decision structure (arrange module affected by a decision subordinate to module containing the decision to avoid "upward" propagation of logic by "hiding" the results of a decision in the subordinate module), and data access (minimize amount of data that a module can reference).

To achieve the extra flexibility needed to determine efficiently and accurately the EMS/V of old (inventoried) and newly developed weapon systems, it is recommended that the composite design of the software system be coded and implemented using a "Table-Driven" menu concept. (Such a concept is currently being developed at the EMVAF facility at WSMR/DR site--see trip reports, Apendix A.) This concept seems to be the most reasonable way to design a general menu-type program, which can be easily modified from one test to another without the need for extensive recoding of the program. (This conclusion was reached after several discussionns with numerous individuals who work in the EMS/V testing area and/or have produced very similar software systems at other measurement facilities, viz. NSWC/Dahlgren, MIRADCOM/Huntsville, Alabama, SA/Atlanta, and Eglin AFB.)

The "General Purpose Test Program" (GPTP) for data acquisition at the EMVAF facility at WSMR will be used as an example to describe the Table-Driven menu concept. Specifically, the tables which control the Stinger Test Program menu item will be described. (GPTP was written for data <u>acquisition</u> only and not for <u>data reducion</u>. The GPTP software pre-processes the raw data collected in a chamber test with numerous statistical routines to reduce it to a form suitable for storage.)

The approach used by WSMR is to use a modular structure called by an "executive" program containing a menu. The executive program allows the user to select and execute a menu item. The particular instructions contained within the menu item are performed subject to numerous parameters which are stored in a table (data array) on a disk file (thus the concept of a Table-Driven Menu). The instructions in the menu are fixed; the parameters in the table are variables. The executive itself has a menu item which can edit the stored parameter table.

At the EMVAF facility, airline driven relays and fiber optic TM packages are used to isolate the weapon under test from the chamber. The table contains a sequencing array which controls the operation and sequencing of the relay bank connected to the test equipment available at the facility. (This part of the table controls the mechanical aspects of the test and is obviously very facility dependent.) The mechanical contents of the table would normally change only if a new piece of hardware were added to the facility. The table also contains a patching array which selects the TM channels to be processed and contains information about the type of data on each channel and how it should be processed. (This part of the table controls the electrical aspects of the test and is obviously very weapon dependent.) The electrical contents of the table would normally change only when a radically different weapon system was under test. Usually, the table can be edited and changed to accommodate most new weapon tests without reprogramming the software.

Most of the weapon systems recently tested at the EMVAF facility were limited to no more than 1 minute of missile continuous run time to avoid possible overheating and wear. The software system, therefore, keeps an automatic record of all test parameters, including test number, date, purpose, start/stop times, total lapsed time, spin-up time, etc. In order to perform a test quickly, all computer interrupts from other users are suspended. Also, the operating system is locked-out. (Extensive program overhead is required to interrogate the terminal for any abort instructions.) Each test has a definite start-up and shut-down sequence. If a test is aborted, the shut-down sequence is initiated. Before a test can be resumed or re-started after an abort, a definite cool-down period must elapse. Timing programs must be written to prevent an early re-start.

This software system design using a "Table-Driven Menu" seems to provide the needed flexibility in a menu program for EMS/V testing and has been adapted to handle all of the tests at the EMVAF facility. It, obviously, does not give total flexibility in changing the testing procedures without making any software changes, but it does allow a great deal of flexibility in changing between "similar" tests of typical weapon systems, with a minimum of software recoding requirements.

GPTP contains an operating system (OS) which manages the system resources in a way to perform a sequence of operations. The OS consists of a "supervisor," an "executive/manager," and a large number of instructions divided into "subprograms" (system tasks) which can perform the various operations allowed by the system. The "supervisor" directs the overall operation of the system. The "executive" is the interface between the operator and the system. It interprets (decodes) the operator commands, searches a privileged menu for the command (which returns a number), does a computed GOTO on the number (which establishes the proper subprogram operation), and executes the subprogram instructions. The "manager" acts as the "executive" for the users but has a restricted (non-privileged) menu. The non-privileged menu for the "manager" is a subset of the privileged menu for the "executive." In all other respects, the "manager" acts just like the "executive."

A "direct" menu is located in memory and contains all of the direct operations that the system can perform; an "indirect" menu (an extended menu) is stored on disk and contains user/operator written operations. The user/operator written items in the menu are "procedural" programs consisting of various combinations of the basic operations, i.e., chains/links of the basic operations. These procedural programs are called Command Substitution System (CSS) programs. The system looks first into the menu in memory, then, if it cannot find the called item, it searches the menu on disk. The commands entered by the operator are called Console Commands; the commands entered by a user are called User Commands. Either command has the form

mnemonic, operation (list)

The "executive" or the "manager," if it has the privilege, executes the instructions corresponding to commands found in either menu according to the options/instructions present in the operand list. The various options are implemented through a command processor which generates a supervisory call (SVC) which handles the subprogram linkage. When an SVC is generated, the supervisor is interrupted, critical parameters are stored, a set of instructions are executed (depending on the directions contained in the data stored in a "parameter block") and then an interrupt return is issued returning control back to the supervisor. The "parameter block" contains (a) a function code to direct its operation, (b) logical unit number(s) of the device(s) necessary to perform the function, and (c) an error status code that can be checked by the supervisor. Examples of SVC functional directives are:

1. IO: read/write (and proceed, pause, wait)
2. System utility - pause
    get/release storage
    set status
    fetch positioner
    unpack binary number
    log message
    interrogate clock (timing)
    fetch data
    time of day wait
    timmer management (set clocks)
    interval wait
    pack numerical data
    pack file descriptor
    scan mnemonic table
    move ASCII characters
    peek
    expand/contract allocation
3. End of task
4. Overlay call
5. Intertask coordination
6. File handling services
7. User SVC

All of the subprograms that are available in the operating system through the use of console or user commands are usually automatically incorporated in a higher level program like FORTRAN at compilation time. This is accomplished in the program by calling a FORTRAN extended subroutine which contains a direct call to an SVC. If an extended FORTRAN subroutine is not available, a user subroutine can be written using the "user SVC" call listed as item 7 above. The "parameter table" must now, however, be written in Assembly Language. User SVC calls are usually written only if the extended FORTRAN that comes with the system will not support a particular task or if the FORTRAN program is too slow in performing a particular task. User SVC calls are much more efficient and much less time consuming.

The OS also controls the Table Driven Menu, since each menu item is just a CSS procedural file. As a specific example of the use of this design concept, see Figure 1 of Appendix A which contains the "state diagram" for the Stinger Test Program. On this Figure 1, each square block represents a basic module, i.e., a particular, single purpose, independent function which can be performed during the test of a weapon system. Each circle represents a local "executive" containing a menu. The mnemonic for each module is the letter placed near the line leaving the "executive" to the module. Each module is a FORTRAN subroutine stored on disk in source, assembly, and object codes. For each module, a CSS procedural file is also stored on disk which will link the various object codes into an executable machine code and load and execute the program. Since each module is an independent entity, it can be compiled and saved. If a module is changed, it, of course, must be recompiled and stored in the updated form. The only change then required to update a CSS procedural file is to use the linkage editor to relink the entire code. No further recompilation is necessary. In this sense, each module is thought of as an element in a library file. For this reason, all subroutines must be reentrant and all subroutines common to all simultaneous users should be placed in a run-time library if multiple task programming (time sharing) is to be used. In a reentrant subroutine, any impure code (data addresses assigned inside the subroutine) must be designated and kept separate from any pure code (data addresses assigned outside the subroutine). The code, therefore, must not contain direct memory references on storage, and the calling routines must pass to all storage information the subroutine through the call list. This passing is automatically done in the higher level programs like FORTRAN and only becomes important in user written assembly language subroutines to be called by a FORTRAN program.

To enter the highest level executive state in the Stinger code, a console or user command, e.g., "Stinger," is entered to identify the Stinger menu item and to begin the Stinger CSS file. Once in the Stinger menu item, a prompt is issued to the calling console by the local executive to indicate a new menu procedural item is requested. Some of the menu items are single purpose manual operations; whereas, others form a normal test run sequence, such as item "S" (see Figure 1 of Appendix A). Item "S" requires that the parameters contained in several "mechanical" relay tables and an "electrical" data table be set before it is called. This setting of parameters is accomplished by menu item "SP" which contains a lower-level "executive" which can edit the tables and change the table entries to values other than their present (default) initial values. The two "mechanical" tables are used in a "Start Normal Run Sequence" (SNRS) subroutine and a "Normal Shut Down Sequence" (NSDS) subroutine. The SNRS subroutine locks out all other tasks, assigns exclusive control of the relay banks to the calling executive, logs all comments, conditions, parmeters, etc., and turns on all devices per a "Start Up Delay Table" (SUDT), cf. Table IV of Appendix A. The NSDS subroutine turns off all devices per a "Shut Down Delay Table" (SDDT), again cf. Table IV of Appendix A. The "electrical" table is stored in a common block and contains a "Data Type Table" (DTT), for TM channels and probes, cf. Table V of Appendix A. The data type code is:

0: not used
1: use one data point only
2: read peak value
3: read minimum value
4: use average value
5: read RMS value

The SUDT contains the time delay (in seconds), from the start of the test, before the relay is energized. A "Start Up Time Table" (SUTT) is generated by calling the clock at the beginning of the test to convert the delay times after the start of the test to actual real times which can be compared to other calls to the clock throughout the remainder of the test. The SDDT contains the time delay, in seconds, from the end of the test (normal stop, or abort) before the relay is de-energized. A "Shut Down Time Table" (SDTT) is generated by calling the clock at the end of the test to convert the delay times after the end of the test to actual real time which can be compared to other calls to the clock. A "State Diagram" for the above described procedures is contained in Figure 2 of Appendix

The mechanical table also contains a "relay protect mask" which gives the calling executive exclusive control of four relay banks (each bank contains four relays) if a "1" (as opposed to a "0") is present in the appropriate position of the relay mask hexadecimal word. Also, the range of the "control word" and the number of increments it is allowed to have are contained in the table. All DO-loops are controlled by the number of increments of the "control word."

At the EMVAF facility, the dialogue between the user and the "supervisor" is kept to a minimum, since most users are not interested in programming. A single character prompt is used for most menu items, and all of the default values of the parameters to be used are stored in a file. The user, therefore, changes only those parameters that are necessary, eliminating the need for the user to retype those numbers each time the program is executed.

In summary, with the Table-Driven Menu approach being implemented at WSMR, the user guides the program, which is only a tool to support the test plan, by entering a single letter prompt to select various menu items. Since all of the parameters which are used to control the test are stored in a table which can be easily edited, the user has a great deal of flexibility in tailoring the available modular library routines to fit the weapon system under test without the necessity of frequent recoding of the current software system.

APPENDIX F[*]

MODERN SOFTWARE DEVELOPMENT METHODOLOGIES

F.1 INTRODUCTION

While the primary emphasis in computer system procurement and development in the 1960's and early 1970's was on hardware, the late 1970's have seen more and more emphasis on application software development. This is an area that is not amenable to normal management and cost controls. Thus instances of actual costs of several times the initial budgeted cost and a time to initial operational capability sometimes twice as long as planned are more often the case than not. To minimize the occurrence of such problems and to assure an orderly software development and completion process, several new software development and management methodologies have been developed [2], [3], [4][*]. In this appendix, an overview of the more relevant aspects of these recent developments is presented. The material is divided into two parts. The first part addresses the management of a software development project. The second part addresses new software development methodologies.

F.2 SOFTWARE LIFE CYCLE MANAGEMENT

The traditional software life cycle can be divided into five phases. These are:

1. Planning
2. Definition
3. Design
4. Coding
5. Testing

In a typical large procurement, these phases are divided into as many as four separate contracts, each requiring a separate proposal from the software contractor. After a definition of each of the five phases

---

[*] Prelared by W. M. Leach.

[**] The references on which this Appendix is based are listed in Section 5, page 19, of the main body of the report.

is given, four different procurement strategies will be described and compared.

F.2.1 Phase 1 - Planning

The first phase of software development is the planning phase. During this phase, the customer, working directly with project engineers, generates his requirements. If a project requires both hardware and software, the project engineers must be knowledgeable in both areas.

One of the major pitfalls in the planning stage is the lack of means to produce clear, concise, and unambiguous statements of user requirements. A clear understanding must be established as to what can be accomplished with software. It is crucial to not overestimate the state of the art. A failure to establish realistic requirements in the planning stage is one of the biggest contributors to the high cost of software.

F.2.2 Phase 2 - Definition

In the definition phase, two systems processes are initiated simultaneously. The first is to develop a definition of the system to be developed from the system requirements established in Phase 1. The second is the planning for and the initial structuring of the organizational system that will do the analysis, design, and implementation of the proposed system.

In the definition phase, the systems and test requirements should be clearly defined to the extent that the software designers know what is required and can develop their design from the specifications themselves. A requirements document should be developed that contains a complete specification - clear, consistent, and unambiguous - describing "what" the software product must accomplish and with "how much." This document should not describe "how" the software will satisfy the requirements, for this is to be done during the design phase.

Also during the definition phase, test requirements should be developed. The test requirements identify the functions to be tested and the performance characteristics to be satisfied. They should identify the number of test cases with specifications for the ranges, limits, and timings for the data values. The criteria for successful test execution and for customer acceptance must be carefully and unambiguously defined so that there can be no misunderstanding as to what constitutes successful system operation.

The software project is itself a system, and it must be planned and organized using good systems management techniques. A software development plan and a work breakdown struucture must be developed and refined, and detailed resource plans, schedules, and budgets drawn up. Procedures must be established for such items as project monitoring and reporting, for project review and concurrence, for producing documentation, and for controlling design changes.

As part of a software develpoment contract, a software plan should be required as a deliverable data item. This plan should give the methodology by which the management team will balance products, services, resources, and schedules. It should also establish the means for appraising and redirecting activities. The customer requirements may dictate the outline and content of the delivered plan.

A convenient tool which is required by some contractual terms is the hierarchal work breakdown structure. The work breakdown structure resembles the product structure and provides a basis for work assignments and cost accounting. The structure is convenient for allocating all budgeted resources and requirements. The contractor should have control of the work breakdown structure to prevent it from becoming an arbitrarily directed standard.

F.2.3 Phase 3 - Design

The design phase is concerned with the design of a software structure that can satisfy the requirements. One of the products produced during this phase is a specification which describes how the system is to be structured to satisfy all the requirements set forth in the requirements specifications. If the costs to maintain and operate a system are to be minimized, it is important that adequate emphasis be paid to design-to-cost and software life cycle cost considerations.

New design techniques such as top-down design, composite design, and structured design provide a methodology for addressing the problems of function, performance, and structure early in the design process and for determining the design approaches to take. They also force early attention to be focused on the problems of interface definition rather than waiting until final system integration to see if all the interfaces have been identified and properly specified.

The other major products of the design phase are test specifications, standards, and the implementation plan. The test specifications detail the testing plan to be followed during the remainder of the project to satisfy the test requirements. Standards are necessary to define the conventions to be followed during the development and should be made part of a software quality assurance program to provide the discipline necessary to control the quality of the final software product. The implementation plan becomes a primary management control document giving measurable milestones, assignments, resources, and schedules.

The output of the design phase contains no code. This concept is an important one. If coding is started too early in the project, there is a high probability that a large percentage of the code must subsequently be redone before the software passes evaluation. Software projects which place a primary emphasis on generating code early in the development cycle usually experience testing phases which last significantly longer than the combined definition, design, and coding phases. Before the writing of code is begun, a structured "walk-through" should be performed to verify the accuracy of the design. This should be attended by the project leader, chief programmer, programmers, and the supervisor.

The output of the design phase can be summarized as follows:

1. An update of all technical documentation described in Phase 2 will be done.
2. Program hierarchy charts which show control flow between segments will be available.
3. Prologues which describe the function performed by each segment within a module will be written. Definition will be made of all data inputs to each segment and data outputs from each segment.
4. Test plans required to test the operation of each module will be generated.
5. Detailed data maps and description of each table and item will be provided.

F.2.4 Phase 4 - Coding

Because the entire program is completely specified after Phase 3, segment coding can be begun after the structured "walk-through" of the design is completed. The resultant code should be simple, straightforward, and easy to understand. Where applicable, structured coding techniques should be employed.

At the completion of the coding, a structured "walk-through" should be performed to detect design flaws and misinterpretations. This "walk-through" should be attended by the chief programmer, coders who are responsible for software directly interfacing with each module being reviewed, the first level supervisor, and the programmer who will be responsible for maintaining the software after the development is complete.

F.2.5 Phase 5 - Testing

The testing phase consists of four stages of software testing. First, each segment is tested to ensure that its function works as specified in the design phase. Second, the segments are strung together to ensure that groupings of segments, or modules, work as specified in the design phase. These two test activities are performed by the design programmers. After the successful completion of these two steps, the chief programmer and the programmer who will be responsible for maintenance of the software verify the operation of each subprogram. The maintenance programmer must agree that each subprogram works according to the specifications generated in the definition and design phases. He must also agree that each subprogram meets commercial standards, i.e., well-documented, structured design, easy to understand code. The last step checks the interoperation of all subprograms. A team consisting of chief programmers, senior maintenance programmers, and evaluation engineers should perform this "total program" or system test. All functions of the program should be tested under stress conditions. The objective of system testing should be to make the system fail in order to locate any problems. Any problems should be passed on to the project leader to ensure prompt resolution.

After system testing is complete, the responsibility for modifying or maintaining the source program is relinquished by the chief and design programmers and passed on to the maintenance programmer. The software maintenance phase now begins and the total system is evaluated by an independent group of system experts. During this evaluation, a complete set of acceptance tests are executed. Design problems are resolved by the maintenance programmers. The effort expended in evaluation testing is not considered part of the software development cost.

F.3 SOFTWARE PROCUREMENT STRATEGIES

There are three strategies which are typically used for software development. These strategies are illustrated in Figure F-1. Each strategy requires two or more contracts. Software maintenance is illustrated as a separate contract because DoD Directive 5000.29 specifies a separate maintenance agent. The principal inputs and outputs of each strategy can be summarized as follows:

1. Strategy A. The principal technical outputs of a full scale development contract are the requirements, design, and software products baselines which are the principal inputs to the agent performing the maintenance contract.
2. Strategy B. The principal technical outputs of the definition contract are the requirements and design baselines. The output may include a model contract or a proposal for a full scale development contract.
3. Strategy C. The principal technical output of the definition contract is a requirements baseline. The output may include a model contract or a proposal for a full scale development contract.

Design, coding, and testing can be merged into a continuous process in strategies A and C. These strategies also permit the use of new software development technologies. Strategy B breaks the continuum by allocating design and coding/testing to separate contracts. If the delay between contracts is long, the design contractor, if he wins the coding/testing contract, will reassign most of his original team and the odds of reassembling the team are small. The net result can be a poorly developed software system. If a new software contractor is selected, further losses can occur. Thus strategies A and C are preferable. However, a contract that follows these strategies should specify that the contractor will not begin the coding phase until the software design has been approved by the customer. This approval should not be given before the design is verified during the structured "walk-through" at the end of the design phase.

F.4 NEW SOFTWARE TECHNOLOGIES

Several new "structured" methodologies have been developed to aid in software development projects. These include structured programming, composite design, HIPO, top-down development, structured analysis, structured walk-throughs, and chief programmer teams. This section briefly reviews each of these new methodologies and makes suggestions about their use.

F.4.1 Structured Programming

Structured programming is considered to be the "first" new software development methodology. A more appropriate name might be "structured coding," for this methodology is one for writing code. Its most important feature is a procedural logic based on combinations of IF-THEN-ELSE, DO-WHILE, and "sequence" structures. Such logic usually eliminates the need for GOTO statements and unconditional branching instructions. Structured code is easy to comprehend. Thus it is easier to maintain, and it is more likely to be correct code. Structured code adds an overhead of 5-10% to the memory requirements and execution time of a program. However, because it is better organized, structured code is usually more efficient than unstructured code.

F.4.2 Composite Design

Composite design is a very powerful technique for software system design. It concentrates on building systems from small, highly independent, single-purpose program modules. It can be described as the process of deciding which modules, interconnected in which way, will best solve some well-stated problem. Because each module is highly independent, composite design results in improved maintainability with little or no "ripple effect" when a module is modified. In addition, composite design provides greater reliability and understanding of how the software system works. Compared to structured programming, composite design has a far greater impact on maintainability.

F.4.3 HIPO and Other Documentation Techniques

Documentation of software has classically been done with flowcharts, decision tables, and narrative text. Along with the recent developments of structured programming and composite design, new documenation techniques have been developed which can describe the procedural logic represented by structured programming and the architectural design represented by composite design. The most widely known documentation technique is known as HIPO - an abbreviation for "Hierarchy, plus Input, Process and Output." This is a better technique than flowcharts because HIPO diagrams show data flow through the programs as well as logic flow.

F.4.4 Top-Down Development

It has become popular to code and test software systems from the top down. The "top-down" approach to implementing systems requires the coding and testing of the top-level or executive module first with the lower-level modules taking the form of "stubs." (A typical example of a stub is a module which exists immediately without doing any real processing.) Subsequent development of the system involves substitution of real modules for the stubs. This is in contrast to the classical approach, referred to as "bottom-up," of unit testing low-level modules and then integrating them into larger entities.

Top-down implementation has a number of benefits, some of which are "political" in nature. It tends to distribute system testing and integration throughout the entire project rather than saving it for the end of the project. It also tends to expose major interface problems early in the project rather than at the end. Equally important, it allows the project manager to demonstrate a working subset of the system to the customer at an early date.

F.4.5 Structured Analysis

Structured analysis is a new discipline which addresses the problem of figuring out what the user wants. Its basic objective is to provide a formal description of the user's requirements, expressed in logical terms (i.e., with as little reference as possible to the peculiarities of a specific machine, a specific data base management system, etc.), using standard tools and building blocks. Its key ingredients are communications tools to improve the communication between the analyst and user, and a new approch to the "systems development life cycle" that encourages both user and analyst to view the development of a software system as an iterative process rather than a sequential process.

F.4.6 Structured Walk-Throughs

One of the most powerful of the new structured methodologies is the structured walk-through. In its simplest form, a walk-through is an informal procedure for reviewing the correctness and quality of the analysis, design, code, test data, and documentation associated with a software project. The review is normally carried out by the programmer's peers rather than his supervisors. The benefits are increased reliability, more comprehensive and maintainable code, greater learning and sharing of information among team members, and a greater chance that a partially completed program can be salvaged if a programmer leaves in the middle of the project.

F.4.7 Chief Programmer Teams

The basic concept of a chief programmer team is to organize a software development project around a person who has (a) programming abilities substantially greater than other programmers in the team, (b) ability to provide the documenttion for the code, the operational procedures, and the user manuals for the system, and (c) the ability to supervise a team of specialists including an apprentice chief programmer, an expert in the programming language or operating system or data base management system being used, a person who develops debugging packages and other software development tools for the project, and a person who organizes and controls the source programs, object programs, listings, and other documents associated with the project. The concentrated talents of one superprogrammer makes it possible for medium-sized software development project to be accomplished with a much smaller group than would be necessary otherwise.

APPENDIX G[*]

ILLUSTRATIVE STATEMENT OF WORK FOR DATA ACQUISITION SOFTWARE SYSTEM

1. OBJECTIVE

1.1 The overall objective of this effort is to design, code, install, test, and document a flexible software system for the Automated Data Acquisition and Control System (ADACS) in the Electromagnetic Compatibility Analysis Facility (EMCAF) at the Rome Air Development Center (RADC). This system is being developed to provide the capability for handling a greater diversity of munitions systems with an increased throughput of electromagnetic susceptibility testing and vulnerability analysis (EMS/V) of such systems. The software system to be developed under this procurement will provide the capability for automatic computer control of the tests and for data reduction, analysis, and display after the tests.

2. SCOPE

2.1 The scope of this effort encompasses the detailed design, development, assembly, integration, test, and documentation of the ADACS software package. It also covers contractor assistance in training, evaluation, and operation of this software.

3. CONTRACT WORK BREAKDOWN STRUCTURE

3.1 The attached Contract Work Breakdown Structure graphically portrays the work to be accomplished consistent with the scope of the contract.

4. RELATED DOCUMENTS

4.1 The Contract Data Requirements List, Delivery Schedule, military specifications and standards, and other documents, to the extent they are referred to in this Statement-of-Work (SOW), further define the work required under this contract. In particular, the Delivery Schedule and the Contract Data Requirements List define the requisite periods

---

[*] Prepared by W. M. Leach.

of performance and delivery dates applicable to all tasks defined in this SOW and to the products of these tasks. These related documents are either included in this request for proposal or may be obtained through the Contracting Officer at RADC.

5. BACKGROUND

5.1 The development of the ADACS is part of the program under Project Have Note to upgrade the EMCAF facilities at RADC. This new facility is being developed to provide a means to efficiently and accurately determine the EMS/V characteristics of Air Force munitions systems, both in inventory and under present or future development. The EMCAF consists of an RF anechioc chamber in which the munitions system under test is mounted, high-power RF transmitters and antennas which generate the interfering electromagnetic radiation, and a computer controlled instrumentation and measurement system which will automate the tests and collect the data. The computer is also to be used to reduce and display the data collected and to use the data in performing vulnerability analyses of munitions systems.

5.2 Diagrams of the EMCAF and ADACS and a list of the hardware items in the system are provided. Computer automation of the measurements is provided by a Hewlett-Packard 1000/45 computer system with disk and tape storage, graphics display terminals, hard copy units, plotters, and a measurement and control subsystem. The software supplied with the computer system includes a real-time multi-user operating system, FORTRAN and Basic compliers, a plotting package, and a file management system. The software to be procured under this SOW will, in part, control the measurement and control subsystem. The types of equipmennt in this subsystem include RF power meters, RF synthesizer, RF attenuators, pulse generators, multimeters, and an antenna positioner.

5.3 The testing with the present system is semi-automatic. Programmable calculators provide limited instrument control by automatically setting some of the test parameters, while others are set manually by the operator. After all parameters are set for each test condition, the calculators record all data and test conditions. The calculators are interfaced to the instrumentation through the IEEE 488 bus. At the conclusion of a test, the data are sorted into the proper order and corrected to account for such variables as the frequency variation of antenna gain, etc. It is then displayed in graphical form by a calculator-controlled plotter and stored on cassette tape for future reference.

5.4 There are five basic measurement and data reduction programs that have been developed for the calculator controlled system. These have been written in HP 9830 Basic Language. They are available for informational purpose through the Procuring Contracting Officer at RADC. A brief description of these programs is given in the following five paragraphs.

5.4.1 An antenna gain program is used to measure the gain versus frequency of the antennas used in the anechoic chambers. The resulting gain data is used in the chamber calibration program described in the following paragraph. The gain measurement can be made with either one of two models. These are the "two-identical-antenna" method or the "three-antenna" method.

5.4.2 A chamber calibration program is used to calibrate the electric field intensity at a given location inside the chamber as a function of frequency for a known RF power input to the chamber. The electric field intensity at that location is determined from the output of a receiving antenna which has been calibrated with the gain measurement program. The output of the chamber calibration program is the power transfer function from the chamber RF input port to the specified location inside the chamber. Between these two points, the RF signal must propagate through transmission lines and/or waveguides and directional couplers, be transmitted by an antenna, and propagate from the antenna to the speciific location.

5.4.3 A frequency scan program is used to measure the effects of RF radiation on a munitions system as a function of frequency. The system under test (SUT) is mounted inside the anechoic chamber at a position where the RF calibration has been performed. The internal control signals inside the SUT are monitored in order to record the effects of the RF interference. During the measurement procedure, the RF power input to the chamber is monitored at a directional coupler. The program sets the frequency, then raises the power level until a selected signal (which indicates the degree of interference) inside the SUT increases to a predetermined level. This level is called the Standard Response level, and it is chosen to be below that which would occur at breaklock of the SUT. When this level is reached, the program records all test data, lowers the power level, steps the test frequency, and repeats the procedure until the frequency scan is complete. The data from the chamber calibration program are then used to convert the measured RF level from the directional coupler at the chamber input into RF power density at the SUT for each test frequency.

5.4.4 A second test program is used to measure the effects of RF radiation on the SUT as a function of either the RF modulation pulse repetition frequency, the pulse width, or the azimuthal angular position of the weapon with respect to the transmitting antenna. The operation of this program is similar to that of the frequency scan program with the exception that the independent test variable is not frequency.

5.4.5 A data manipulation and plotting program is used to reduce, display, and store the data taken in the test. This program is written to plot the data in standard format for reports and to display it in a suitable format for a subsequent vulnerability analysis of the SUT. This program also provides for modifications, additions, deletions, and sorting as required.

6. SOFTWARE DEVELOPMENT STANDARDS

6.1 The contractor shall provide the engineering resources to design, code, install, test, and document the software tasks listed in this SOW. All software code developed under this effort shall be coded to be executed on the resident computer system of EMCAF. The system will be available for software development during the duration of the contract. A complete set of operating manuals will be supplied to the contractor during this period.

6.2 The design, coding, and installation of the software developed under this contract shall be done in accordance with RADC Computer Software Specifications No. CP 0787796100E (dated 30 May 1979) paragraphs 1.1, 5.3.1.2, 5.3.2.1, and 5.3.3.1 through 5.3.3.4. All software code shall be written in the FORTRAN language provided with the ADACS computer. The system contains a writable control store and a user control store. These can be programmed at the assembly level to handle frequent but slow processes, such as device input/output, in order to realize the ultimate speed of which the computer processor and memory are capable.

6.3 The software shall be designed using the top-down development technique. Before coding of the software is begun, a formal structured walk-through shall be performed to verify the design. The contractor shall then provide the Government with the design for approval. The design structure must conform to modern software design concepts in order to ensure program quality. In the software design, the contractor shall demonstrate that the design possesses the desirable features of reliability, maintainability, modifiability, and efficiency. To achieve these goals, a design technique, such as composite design, where each program is decomposed into a set of small highly independent modules is specified.

6.4 After government approval of the software design, the contractor shall develop the software code for each program module using structured programming concepts. The code shall be written so as to eliminate or minimize the number of GOTO statements. In cases where GOTO statements are used, they shall always go downward in the code and never upward. The only lines with statement numbers shall be CONTINUE statements and nonexecutable statements. In each program module, the smallest statement number shall appear as the first statement number, and all following statement numbers shall increase in numeric order.

6.5 The coding and testing of the program modules shall be performed using top-down development. Critical program modules which require early testing may be coded out of turn after the top-down design has been completed and approved by the Government. (See RADC Computer Specification No. CP 0787796700E, paragraph 5.3.1.1.)

6.6 The contractor shall provide complete documentation of all software developed under this contract. The documentation shall consist of a complete description of the programs and a description of the intermodule interfaces within each program. Each module shall be described in terms of its name, function, inputs, outputs, and external effects. Module names shall be the same as the name used in the CALL statement and shall be descriptive of the function perfomed by the module. The function performed by each module shall be described by narrative description, decision tables, graphs, and/or other material. A precise description of all input data from each module shall be given including output parameters, their physical order, size, type, range, and error information (e.g., return codes). The external effects of each module (e.g., the reading of a tape recorder, the plotting of a graph, etc.) shall be described. If any external effect is conditional, it shall be related to the inputs in a cause and effect manner.

7. CONTRACTOR TASKS

7.1 TASK I. The contractor shall develop a flexible menu test program designed to characterize the EMS/V of a munitions system.

(To define a computer program designed "to characterize the EMS/V of a munitions system" is, in general, impossible to do. This follows because it is impossible to anticipate the test requirements on systems which have not been tested, whether they be in present inventory or under development. In the absence of being able to specify a particular test methodology, the software requirements must be stated in terms of the functions of the equipments which are to be controlled by the software. The first software requirement is a series of program elements (modules) which are designed to exercise the equipments in the ADACS. The second requirement is a software system that will allow the user to effect a logical ordering of the events controlled by these program elements. This ordering of events is the automated implementation of the test plan, prepared in advance, to test a particular munitions system.

This task description should include, first, a description of the requirements of the software modules that control the hardware items that are controlled by the ADACS. If these modules are to be provided by RADC, a description of their capabilities should be provided. The description need not describe the requirements (or capabilities) of each individual module, but rather the overall generic requirements (or capabilities).

The second part of this task description should describe the functional requirements of the software system which is to control the modules. For munitions testing, this description must include the requirements to power up the system under test (SUT), to record the total test time, and to power down the system if a preset time has elapsed or if the test has been completed. In the former case, a capability to automatically resume the test after a preset "cool down" period should be specified.

Because a test plan is an ordering of test events, the software system must be capable of ordering the functions controlled by the software modules in the order specified by the test plan. The capability to repeat the ordered sequence over any range of specified independent variable (e.g., frequency, azimuth, modulation, pulse width, etc.) is required. For each value of the independent variable, the capability to measure and record data, as specified by the test plan, from preselected measuring instruments controlled by the ADACS is required.

The requirements for a standard response test must be described. The standard response variable is a dependent variable which has no specified range of values. Instead, it has a specified upper bound which is not to be exceeded for any value of the independent variable. Once the main independent variable is set by the computer, a second independent variable (e.g., RF power density) is to be increased until the standard response variable reaches its specified upper bound. After the specified test parameters are measured and recorded, the second independent variable is decreased and the main independent variable is stepped to a new value and the procedure is repeated. The option to measure the test parameters as a function of the main independent variable without controlling the standard response variable should also be specified.

The test program must provide the capability of stopping the test sequence and powering down the SUT at any time chosen by the operator. It should also have the provision to resume testing from the point at which testing was terminated.)

7.2 TASK II. The contractor shall develop a facility calibration program designed to measure antenna gain versus frequency and to measure the transmission function from RF power input to the EMCAF anechoic chambers to RF power density at any point inside the chambers versus frequency. (Reference existing procedures)

7.3 TASK III. The contractor shall develop a flexible program to process and analyze the data collected during EMS/V tests in the EMCAF.[*]

7.4 TASK IV. The contractor shall develop an interactive graphics program designed to display, plot, and document test conditions, data,[*] and results.

---

[*] See Appendix D for information on which to complete these task descriptions.