

Michael Goldman  
December 29, 2000

## **Range Scan Program**

### **Proposed Scan Scheduler, Conventions, Missions**

#### **Introduction.**

The Scheduler is a software module of the rangefinder metrology system, which remains to be defined and implemented. Measurement missions to be carried out by the ranging system have to be scheduled for implementation. This is controlled by the Scheduler, which consists of two computational sub-units: a client unit resident in the GBT control room computer, and a server unit resident in the ZIY metrology computer. The present note discusses the nature of the Scheduler, its possible structure and implementation. It also suggests nomenclature and conventions for mission scan scheduling.

We start from a viewpoint that the rangefinder metrology system performs a set of formal, well-defined metrology missions. For each mission we classify the range scan procedure to be used, by partitioning the mission into a sequence of scan programs. Each program of a mission commands performance of a set of scan scenarios. Each scenario is characterized by a list of available rangefinders, a sequence of "scan frames" to be acquired, and a list of targets to be scanned sequentially by each ranger during frame acquisition. A frame is a set of range measurements made simultaneously by different rangefinders.

In this note we also try to define the metrology missions in a formally structured way. We do this to provide guidelines for the software codes which will control execution of the missions by the metrology rangefinders. We also provide flow charts to indicate the operations to be performed in the individual missions.

A mission request is initiated by keyboard input to a computer in the GBT control room. One brings up a "Mission" menu, available in the metrology system's graphic user interface, on the computer's monitor screen. For a telescope observing run, a sequence of missions for the run is entered. For each of the missions to be performed in sequence, one selects the mission from among the GUI menu's list of missions, sets the mission start time and certain mission-specific parameters and options using the menu. A "Mission Validate" option is selected

from the menu, and the return key is pressed to enter the mission request. The full sequence of missions for the observing run is validated and initiated by selecting an option “Start Mission Sequence” from the GUI menu and pressing the keyboard return key.

On initiation, the request is transmitted to a software module, the scheduler client. For GBT closed loop phase III operation an option will be included to allow mission request modifications to be made directly to the scheduler client by the Active Surface Module during mission operations.

The scheduler client generates a full list of timing parameters for the mission, and timing for the event triggers for the rangefinders and measurement data processing. It also provides a-priori information about the telescope’s commanded track for the mission, when possible. Some of the timing parameters are event start times for individual scan scenarios, others may be periods for repeated scans. The scheduler client then sends the request, as a formatted data list, to the scheduler server.

To perform a mission, measurements for the mission must be specified in detail and timing triggers generated to initiate laser scanning, computation of raw ranges, archiving of measured raw ranges, and subsequent data processing and handling. These tasks are controlled by the scheduler server module. In more detail, they include: naming the mission, naming the rangefinders and targets used, selecting visible targets, providing timing triggers for each range measurement, providing aiming coordinates for the targets at time of scan, converting electronic phases acquired by each scan to a raw range distance (unadjusted), archiving raw ranges, making range adjustments specific to the mission (such as trilateration of ranges or time-interpolated ranges taken among several rangefinder-target pairs), archiving adjusted ranges and, finally, analytically post-processing either the unadjusted or adjusted ranges to obtain derived quantities needed by the mission. (An example of such analytical processing is computation of position coordinates for scanned targets using the adjusted range information).

Let us review our concept. The scheduler client is served by a “scheduler server” in the ZIY metrology computer. The scheduler client request to the ZIY computer is a list. The list consists of a mission name, a start time, a full list of timing parameters, a flag, and (for some missions) parameter values relating

to telescope commanded track for the mission. The flag is included to initiate running of appropriate computational modules in the RANGEFINDER MODULE in the ZIY computer. The list also contains parameter values sufficient to describe the telescope commanded track for the mission a-priori when this is possible. For static scan missions, when the telescope is not moving, the list includes the commanded telescope encoder azimuth and elevation coordinates for the measurements.

The targets and rangefinders for each Mission are contained within the mission object class in the ZIY scheduler server, together with scan programs, scan scenarios, and scan frames for the mission. These generate ordered lists of the rangefinder/target scans to be made. The mission request from the scheduler client provides the timing information required to generate event trigger signals to initiate and perform each particular scan.

### **Proposed Scan Missions and Programs.**

We propose to define the following set of scan missions:

#### **01. Full Surface Measurement, Static.**

This mission is to obtain position coordinates for  $\sim 2000$  main dish surface points when the telescope sits at a fixed commanded elevation and azimuth. It takes about seven minutes to accomplish. Raw acquired range distances are adjusted by a least squares procedure to get best estimated range values and reflector frame coordinates, and then archived. The measured coordinates are those of surface retrotarget reference points. Coordinates of corresponding points on the dish surface are then computed using the measured target coordinate data, and archived. The resulting surface point data can be used in several ways. For example the astronomy observer might subsequently decide to change the surface shape (using its actuators) and transform the surface to some optimal shape (best fit paraboloid for this elevation). In this case, the telescope surface system would compute actuator displacements needed to reshape the main dish. The adjusted coordinate data from the mission would be used as input data for the surface-reshaping computations. Another example: archived adjusted mission coordinate data could be compared with pre-existing dish coordinate data, to look for significant shifts in position, as an indicator of telescope wear or mechanical change; if

coordinate changes above some preset level occurred, a flag would be set to warn the telescope operator that changes have occurred in the telescope geometry, and signal a need for maintenance inspection.

#### 02. Canonical Surface Reflector Scan, Static.

This mission is a reduced version of the previous mission. The purpose of the mission is to provide a check of surface shape at a given telescope elevation, by scanning a subset of about 100 surface retroreflectors from the feed arm lasers. Requirements for this mission are not yet defined, and should be set by the astronomy users. In defining the mission, specifications relating to the following issues, among others, should be given: Are measurements to measure the configuration of the local feed arm ranger scan point control net to be made, or should previous available results be used? How are raw acquired ranges to be analyzed and used? Are measurements to ground targets to be made, if so how are they to be analyzed and used?

#### 03. Canonical Surface Reflector Scan, Dynamic.

The goal of this mission is to obtain time-stamped raw ranges from the six feed arm rangefinders to a subset of  $\sim 100$  main dish targets. The ranges are acquired while the telescope is in motion, observing. Raw ranges to the dish targets are obtained and archived. The archived data is subsequently analyzed and the pointing of the telescope along the observed object's trajectory during the observation is later reconstructed by calculation using the time-stamped raw range data as calculation input information.

#### 04. Dish Rim Reflector Scan, Dynamic.

This mission may be considered to be an independent activity, where the rim triplet ball reflectors' ground frame coordinates are measured by ground rangers, to provide an estimate of the pointing trajectory of the telescope main dish during an observing run.

#### 05. Subreflector Scan, Static.

The goal of this mission is to locate the subreflector in position and orientation

with respect to the reflector coordinate system, by making and analyzing range measurements from the feed arm rangers to the six subreflector target prisms.

#### 06. Subreflector Scan, Dynamic.

This mission relates to dynamical range measurements made from feed arm rangers to the six subreflector prisms during astronomical observations. The details, methodology and analysis of this mission should be described by astronomy users.

#### 07. Alidade Target Scan.

In this mission ranges are measured between retroreflector targets on the alidade structure and ground rangers. The elevation of the tipping structure is held fixed, at some arbitrary user-selected angle during this scan. The alidade azimuth is stepped over a user-selected range, with user-selected step size. At each of the stepped alidade azimuth angles static range measurements are made of target-ranger pairs with mutual visibility.

This scan mission would be used to characterize the static geometric properties of the telescope.

#### 08. Tipping Structure Target, Ground Ranger Scan, Static.

This mission is similar in nature to the previous one. Reflector frame and ground frame coordinates of targets on the feed arm, box structure, main dish rim, and feed arm laser scan points would be determined (by ranging from both ground and feed arm rangers) while holding telescope azimuth fixed at an arbitrary user-selected angle, for a set of stepped user-selected commanded telescope elevation angles.

#### 09. Free Scan-Pair List, Static.

This is a general mission where ranges for an arbitrary set of {retroreflector, rangefinder} pairs are to be measured, with the telescope fixed at a user-selected elevation and azimuth. The person selecting this mission makes a keyboard entry of the desired telescope orientation angles and the list of target-ranger pairs to be

measured, using the “Mission” menu of the metrology GUI of the GBT control room computer.

#### 10. Target Reference Coordinate Determination, Static.

In this mission, the telescope is brought to its rigging elevation angle (i.e. the elevation encoder readout is the rigging elevation angle 50.8 degrees), and held there while range data is acquired. Targets on the box structure, horizontal and vertical feed arms and alidade, elevation bearings, and main dish rim balls are measured by the ground rangefinders, to acquire reference coordinates for the fiducial reference points for these retroreflector targets. Concurrently, the feed arm laser scan point coordinates are determined by ranging from the feed arm lasers to one another, to the ground, to the main dish targets and to the dish rim prism targets are measured. The raw range distances are least-squares adjusted to generate target reference coordinates in the appropriate geometric reference frame (reflector or elevation frame coordinates).

#### 11. Pointing Determination by Rangefinders, Static.

In this mission, the telescope encoders are set to bring the telescope to a commanded azimuth and elevation. When the telescope has arrived at this orientation, as confirmed by the encoder readouts, appropriate range scans are made to the telescope from the full set of rangefinders. The raw range data is least-squares adjusted to obtain best target ranges and coordinates, and is subsequently analyzed to provide best estimated pointing elevation and azimuth angles from the adjusted target position coordinates.

#### 12. Pointing Trajectory Determination, Dynamic.

Here, the rangefinders acquire, track and range appropriate targets on the alidade, main reflector rim, and tipping structure while the telescope is in motion. The resulting rangefinder phase data is reduced to time-stamped raw range information which is archived. The archived ranges are interpolated and extrapolated using appropriate algorithms to provide estimates of range trajectories of targets during the telescope motion. The range trajectories are further analyzed to provide an estimate of the telescope’s pointing trajectory during the motion.

## **Scan Programs, Scan Scenarios, and Scan Frames.**

A scan mission is broken down into scan programs, scenarios and frames. To illustrate the meaning of these terms, before defining them formally let us review in detail how one particular mission is to be accomplished. We choose to examine the mission "Canonical Surface Reflector Scan, Static."

The goal of the mission is assumed to be the following. One wishes to obtain least-squares-adjusted coordinates of a set of  $\sim 100$  main reflector surface points in some Cartesian coordinate system, whose orientation and origin are determined with respect to the ground frame. The purpose of such a mission might ultimately be, for example, to sample the shape of the main reflector surface by measuring 100 points, so that the surface actuators might subsequently reshape the surface to some desired parabola and correct the telescope elevation angle. To carry out this seemingly modest mission, the metrology system must carry out a complex set of tasks.

To measure the surface targets, feed arm rangars are used. Their reference scan points must be located relative to one another to establish a local coordinate reference frame. This is done by having these rangars scan one another and also scan their internal reference reflectors. While these scans are being performed, refractive index measurements needed for range distance determination are made by ground based rangars. The orientation of the feed arm ranger reference coordinate frame must be determined. To do this one has three procedures available: ground rangars and feed arm rangars can scan one another, feed arm rangars can scan ground benchmark monument targets, ground and feed arm rangars scan the main dish rim targets. The data is subsequently least squares adjusted to generate reflector coordinate system coordinates and ground coordinate system coordinates for the feed arm rangars.

The feed arm rangars then scan the surface targets. The raw ranges from feed arm ranger to surface target are then trilaterated to generate (reflector coordinate system) coordinates of the surface points. This trilateration must be scheduled. Six feed arm rangefinders are available to scan about 100 surface target points. Six range determinations can be made simultaneously. Given the targets to be scanned concurrently, an illuminating ranger is assigned to each target. For this same set of targets, one may follow a scan by another scan of the same targets,

permuting the illuminating rangers. For carrying out extended static main dish scan missions, it is convenient to partition targets into sets of six nearby or adjacent targets, illuminate a set of 6 targets concurrently, then rescan the same 6 targets but permuting the illuminating rangers, and subsequently rescan the same targets with various permutations of the illuminating rangers. Each target is scanned by at least 4 rangers if a trilateration with error estimate is desired; if a stronger measurement is desired the targets are scanned with 6 rangers. Some targets are not visible from all six feed arm rangers.

Sets of concurrent range measurements can be grouped into frames. A scan frame is a structure consisting of a set of scan rangers, a set of scan targets, and a scenario matrix, whose rows list the pattern of targets to be scanned sequentially by each ranger and whose columns list the pattern of targets to be scanned concurrently by the rangers. A listing of the scan frames used for each scenario of the mission is stored in a database for the mission. Examples of possible scan frames are given in Figures 1 to 3. If each surface target required 4 sequential scans for a trilateration, and the targets were scanned concurrently six at a time, the 2209 surface targets would require  $2209 \times 4 \times \frac{1}{6} \times \frac{h}{60}$  minutes to scan the entire main dish surface, where  $h$  is the number of seconds required between successive scans of a ranger. For  $h \simeq 0.2$  second, the surface scans would take 4.9 minutes. The time to achieve the mission is estimated at 7 minutes.

The advantage of using such an indexing scheme for the scan measurements is that time required for scanning is minimized, because the scans are permuted among nearby targets and require minimal motions of the rangefinder drive motors. Successive frames may be chosen by moving to a new set of targets which are near to one another but are also near to the targets of the previous frame.

To check that the surface has not changed appreciably during the scan it may be advisable to also include some redundant split-scan frames where the frame targets lie at the dish rim, in three diametrically opposed pairs. Also, at the price of greater total scan time, one could significantly strengthen the surface measurement by including redundant frames such that each of the extra frames overlapped four of the initial frames. A full secondary scan of this nature would lengthen the measurement time by 25%.

We suggest that the missions, programs, scenarios, and frames be indexed for

access in the following format:

```
mission[mission no.], program[mission no.][program no.],  
scenario[mission no.][program no.][scenario no.],  
frame[mission no.][program no.][scenario no.][frame no.].
```

### **The ZIY Scheduler Server Unit.**

A flow chart summarizing the operation of the scheduler server unit is shown in Figure 3.

### **Timing Input Signals.**

A flow chart describing the generation of rangefinder timing signals by the scheduler server is shown in Figure 4.

### **C++ Object Class Definitions.**

In Appendix C, C++-language code is suggested to define the Mission, Program, Scenario and Frame classes. A possible class structure for the scheduler is presented there.

### **Range Reduction Timing.**

When measuring range to a target, the logged time of measurement is the mid-time of the target illumination. The range which is measured is the range at that time.

### **Dynamic Range Measurements.**

When ground rangefinders are used to scan targets on the alidade, box structure or main dish rim during sidereal tracking motion of the telescope, they are used to produce range trajectories of the scanned targets.

Interpolation codes are given in [Gold-2]. Code will have to be defined and written to handle the data flow for the interpolation computations.

### **Range Adjustments And Adjustment Scheduling.**

For stationary telescope measurement data, range trilaterations can be made using the commercial STAR\*NET codes. Data handling interfaces are being written to handle the data flow for static adjustments. The issue of real-time data handling has not yet been addressed.

### **Data Transfer And Logging.**

Data logging protocols for the various observing missions do not yet exist, and have to be defined. Appropriate code will have to be written.

### **Discussion And Summary.**

The nature of the scheduler has been discussed, and suggestions have been given for its implementation in code. The events to be scheduled are presented as flow charts. Formal definitions were suggested for the C<sup>++</sup>-language object classes: Mission, Program, Scenario, Frame. By using these formal classes the scheduler module has a consistent and convenient way of controlling metrology scans.

## References.

- [Gold-1] M.A. Goldman, *GBT Coordinates And Coordinate Transformations*, GBT Memo 165, February 1997.
- [Gold-2] M.A. Goldman, *Dynamical Rangefinder Measurements*, Memo, June 1998.
- [Gold-3] M.A. Goldman, *GBT Laser Ranging System, Laser Metrology Interface Configuration Diagram*. GBT Drawing D35420K014, May 23 1997.
- [Payne-1] J.M. Payne, *Pointing The GBT*, GBT Memo 84,
- [Wells-1] D.C. Wells, *The GBT Precision Pointing System*, GBT Memo 85, September 1992.
- [Wells-2] D.C. Wells, *Gaussfit for Windows 9x & NT*, Limited distribution memo, September 29, 1998.

## **Appendix A. Rangefinder Mission Trees.**

Mission trees, giving mission, programs, scenarios and scan frames are given below.

## **Appendix B. Scheduler Generation Of Event Timing.**

The Scheduler generates detailed event timing for the rangefinders in accordance with the scheme outlined below.

## **Appendix C. Mission And Scheduler Object Class Structure.**

Proposed class objects for Missions, Programs, Scenarios and Frames are given below. A proposed class object for the Scheduler is discussed.

Mission No. = 1 (Full Surface Measurement, Static)  
 Program No. = 4 (Scan Surface and BenchmarkTargets)  
 Scenario No. = 4 (Scan Each of 6 Targets by 4 Rangers)  
 Frame No. = 57 (Frame 57 of Scenario 4)

Frame ID = Frame[1][4][4][57]

Scenario Scan Matrix

A [d, a, b, c]  
 B [c, f, e, d]  
 C [a, b, c, f]  
 D [f, e, d, a]  
 E [e, d, a, b]  
 F [b, c, f, e]

Frame Ranger List

R[01] = ZY113 = A  
 R[02] = ZY114 = B  
 R[03] = ZY115 = C  
 R[04] = ZY116 = D  
 R[05] = ZY117 = E  
 R[06] = ZY118 = F

Frame Target List

T[01] = a (= ZPG\_\*\*\*\*)  
 T[02] = b (= ZPG\_\*\*\*\*)  
 T[03] = c (= ZPG\_\*\*\*\*)  
 T[04] = d (= ZPG\_\*\*\*\*)  
 T[05] = e (= ZPG\_\*\*\*\*)  
 T[06] = f (= ZPG\_\*\*\*\*)

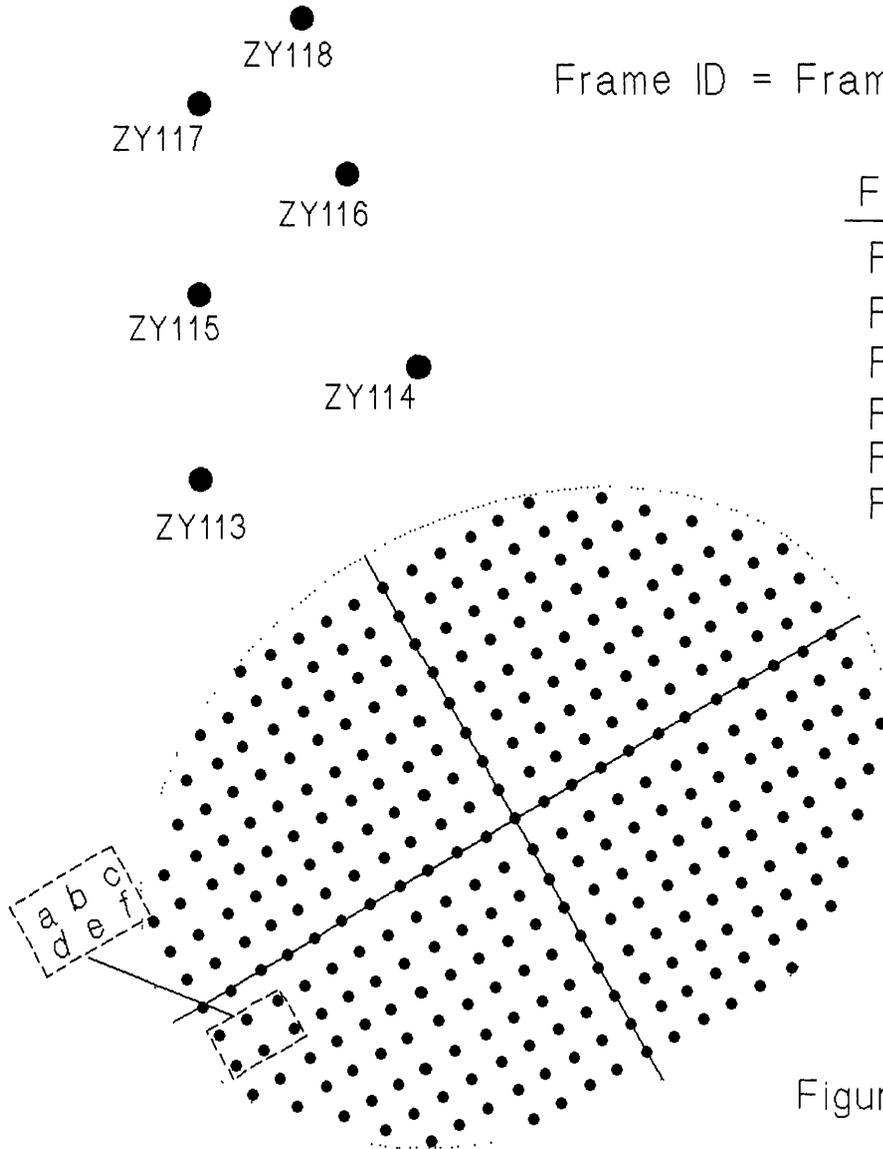
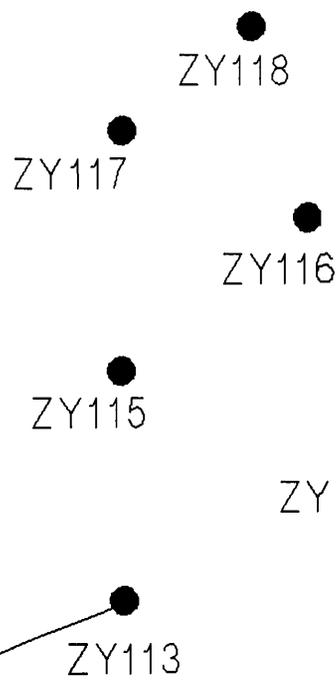


Figure 1. Example Of A Possible Scan Frame.



Mission No. = 1 (Full Surface Measurement, Static)  
 Program No. = 4 (Scan Surface and Benchmark Targets)  
 Scenario No. = 5 (Scan 7 targets by 6 rangers)  
 Frame No. = 3 (Frame 3 of Scenario 5)

Frame ID = Frame[1][4][5][3]

Frame Ranger List

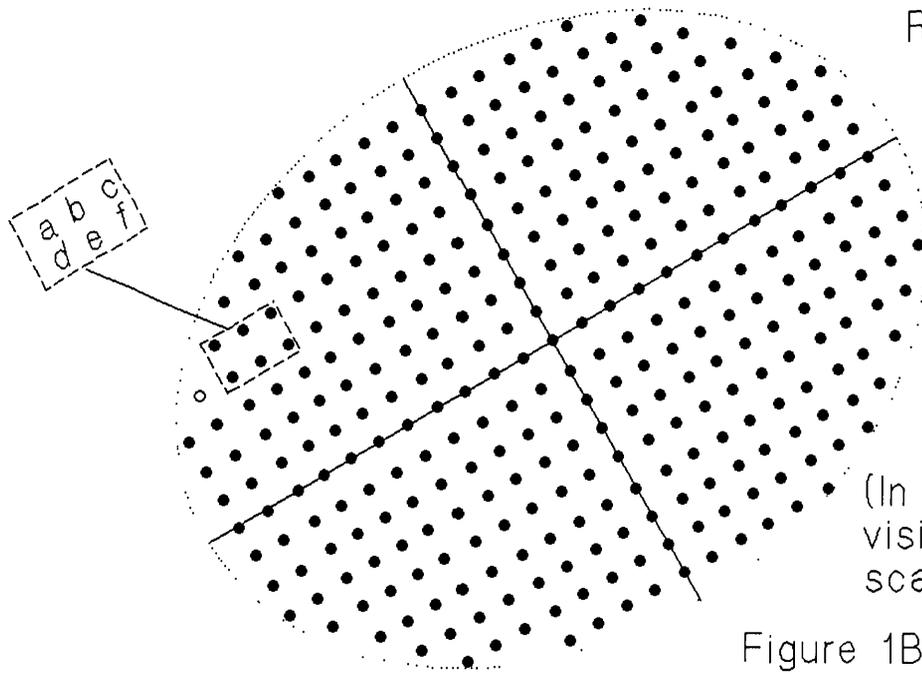
- R[01] = ZY113 = A
- R[02] = ZY114 = B
- R[03] = ZY115 = C
- R[04] = ZY116 = D
- R[05] = ZY117 = E
- R[06] = ZY118 = F

Scenario Scan Matrix

- A [g, g, g, g, g, g]
- B [c, f, e, d, a, b]
- C [a, b, c, f, e, d]
- D [f, e, d, a, c, c]
- E [e, d, a, b, c, f]
- F [b, c, f, e, d, a]

Frame Target List

- T[01] = a (= ZPG\_\*\*\*\*)
- T[02] = b (= ZPG\_\*\*\*\*)
- T[03] = c (= ZPG\_\*\*\*\*)
- T[04] = d (= ZPG\_\*\*\*\*)
- T[05] = e (= ZPG\_\*\*\*\*)
- T[06] = f (= ZPG\_\*\*\*\*)
- T[07] = g (= ZBG\_\*\*\*\*)



(In this example, targets a...f are assumed not visible from ZY13, which is used to concurrently scan a benchmark target).

Figure 1B Example Of A Possible Scan Frame.

● ZY118  
 ● ZY117  
 ● ZY116  
 ● ZY115  
 ● ZY114  
 ● ZY113

Mission No. = 1 (Full Surface Measurement, Static)  
 Program No. = 3 (Scan Rim Triplets)  
 Scenario No. = 1 (Scan Rim Prisms)  
 Frame No. = 1 (First Three Triplet Units)  
 Frame No. = 2 (Next Three Triplet Units)  
 Scenario[1][3][1] contains four frames.

Scenario Scan Matrix

- A [b, a, d, c, f, e]
- B [a, b, c, d, e, f]
- C [a, b, c, d, e, f]
- D [b, a, d, c, f, e]
- E [b, a, d, c, f, e]
- F [a, b, c, d, e, f]

Frame[1][3][1][1]  
Frame Target List

- T[01] = a = ZUG11
- T[02] = b = ZUG01
- T[03] = c = ZUG12
- T[04] = d = ZUG02
- T[05] = e = ZUG13
- T[06] = f = ZUG03

Frame Ranger List

- R[01] = A = ZY113
- R[02] = B = ZY114
- R[03] = C = ZY115
- R[04] = D = ZY116
- R[05] = E = ZY117
- R[06] = F = ZY118

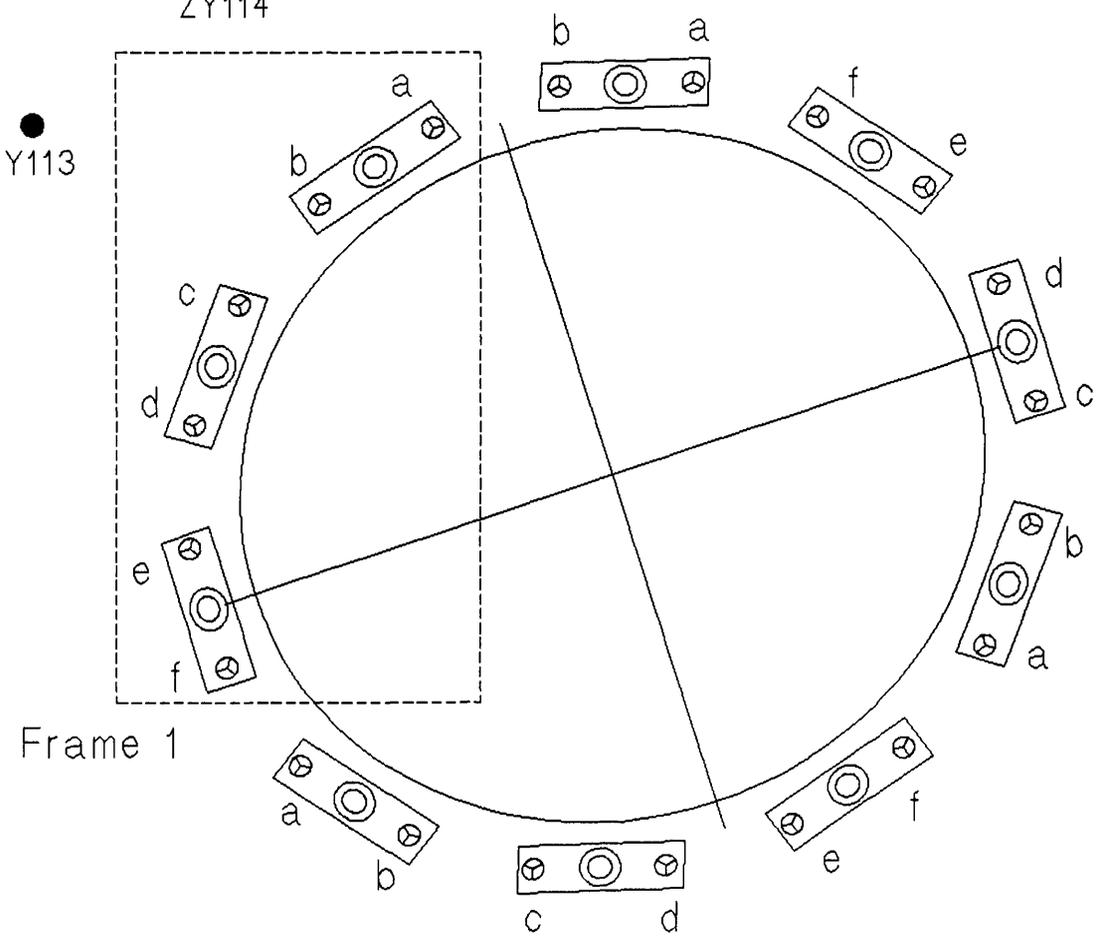


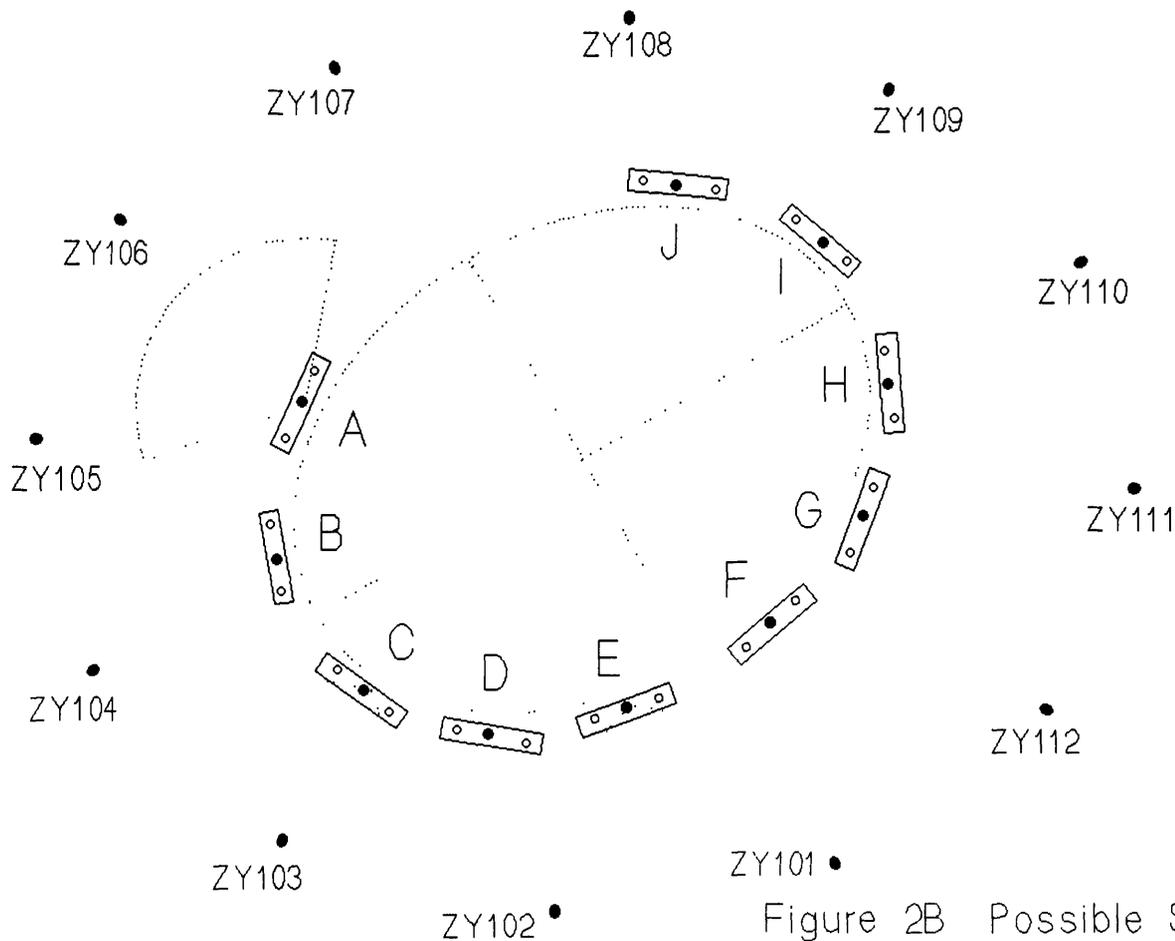
Figure 2A. Possible Scan Frame Scheme For Rim Prisms.

Mission No. = 1 (Full Surface, Static)  
 Program No. = 3 (Scan Rim Triplets)  
 Scenario No. = 2 (Scan Rim Balls)  
 Frame No. = 1 (Balls ZDG01, ZDG03, ZDG05)  
 Scenario[1][3][2] contains 4 frames.

Scenario Scan Matrix

ZY101	[ZDGj, vis(01,j)=1, j=1... 10]
ZY102	[ZDGj, vis(02,j)=1, j=1... 10]
ZY103	[ZDGj, vis(03,j)=1, j=1... 10]
ZY104	[ZDGj, vis(04,j)=1, j=1... 10]
ZY105	[ZDGj, vis(05,j)=1, j=1... 10]
ZY106	[ZDGj, vis(06,j)=1, j=1... 10]
ZY107	[ZDGj, vis(07,j)=1, j=1... 10]
ZY108	[ZDGj, vis(08,j)=1, j=1... 10]
ZY109	[ZDGj, vis(09,j)=1, j=1... 10]
ZY110	[ZDGj, vis(10,j)=1, j=1... 10]
ZY111	[ZDGj, vis(11,j)=1, j=1... 10]
ZY112	[ZDGj, vis(12,j)=1, j=1... 10]

Scan in order of increasing j.



Frame[1][3][2][1]  
 Frame Target List

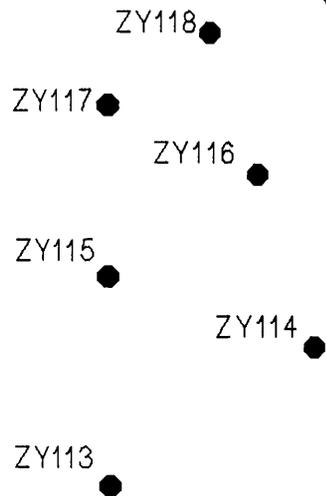
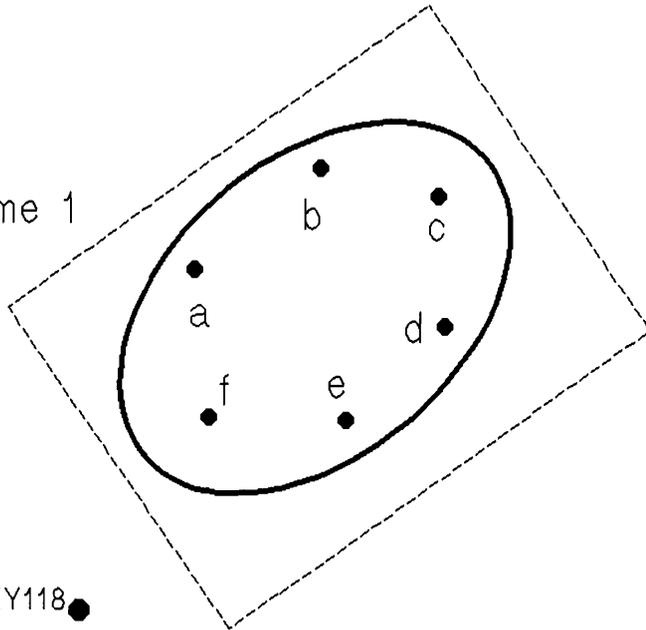
- T[01] = A = ZDG01
- T[02] = C = ZDG03
- T[03] = E = ZDG05

Frame[1][3][2][2]  
 Frame Target List

- T[01] = B = ZDG02
- T[02] = D = ZDG04
- T[03] = F = ZDG06

Figure 2B Possible Scan Frame Scheme For Rim Balls.

Frame 1



Mission No. = 5 (Subreflector, Static)  
 Program No. = 1 (Subreflector, Static)  
 Scenario No. = 1 (Only Scenario, Full Scan)  
 Frame No. = 1 (Only Frame)

A smaller subreflector survey, scanning the subreflector by only 4 of 6 feed arm rangers is generated by deleting any 2 columns of the scan matrix.

Scenario Scan Matrix

A [d, a, b, c, f, e]  
 B [c, f, e, d, a, b]  
 C [a, b, c, f, e, d]  
 D [f, e, d, a, b, c]  
 E [e, d, a, b, c, f]  
 F [b, c, f, e, d, a]

Frame[5][1][1][1]  
 Frame Target List

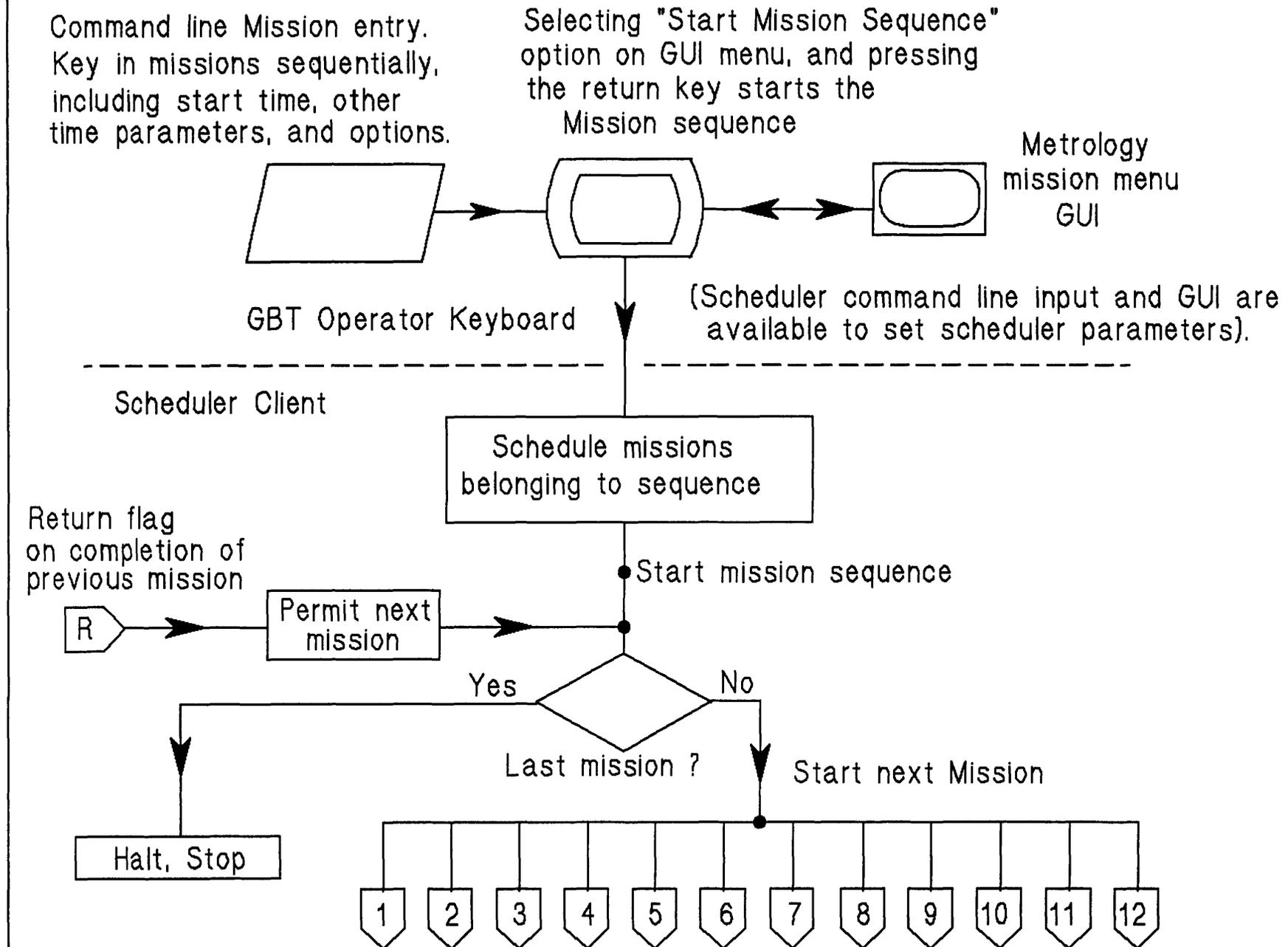
T[01] = a = ZSG01  
 T[02] = b = ZSG02  
 T[03] = c = ZSG03  
 T[04] = d = ZSG04  
 T[05] = e = ZSG05  
 T[06] = f = ZSG06

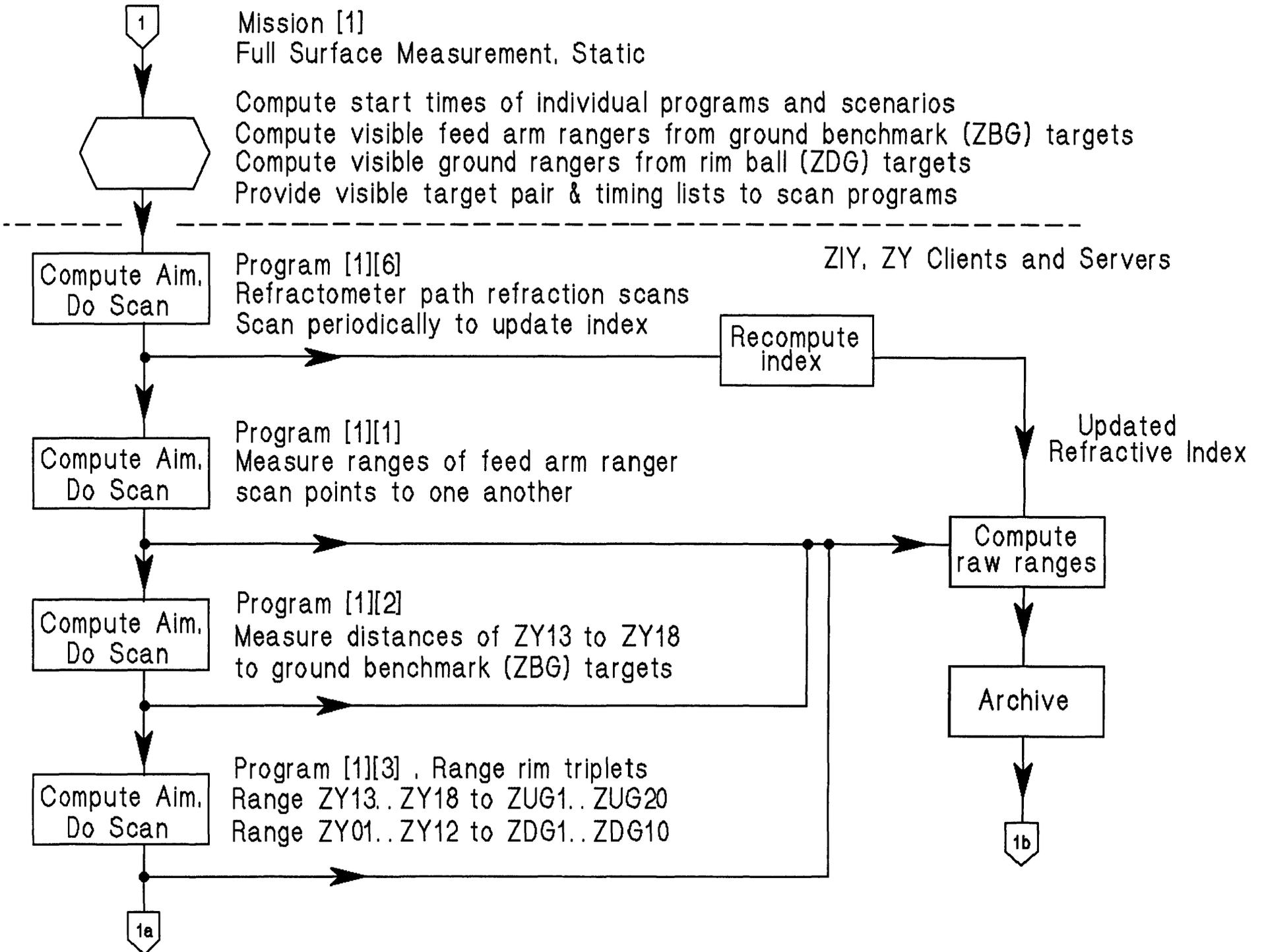
Frame[5][1][1][1]  
 Frame Ranger List

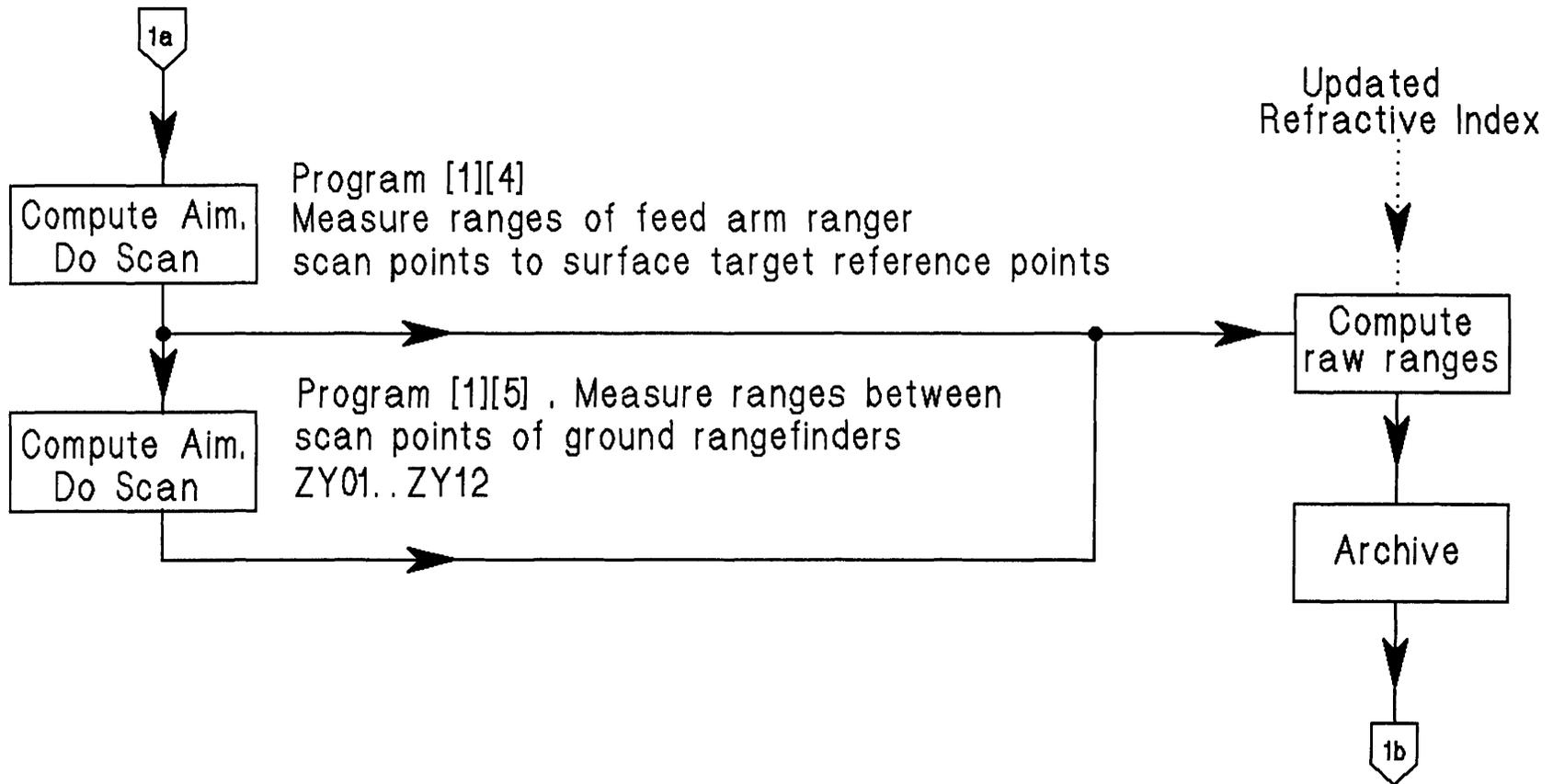
R[01] = A = ZY113  
 R[02] = B = ZY114  
 R[03] = C = ZY115  
 R[04] = D = ZY116  
 R[05] = E = ZY117  
 R[06] = F = ZY118

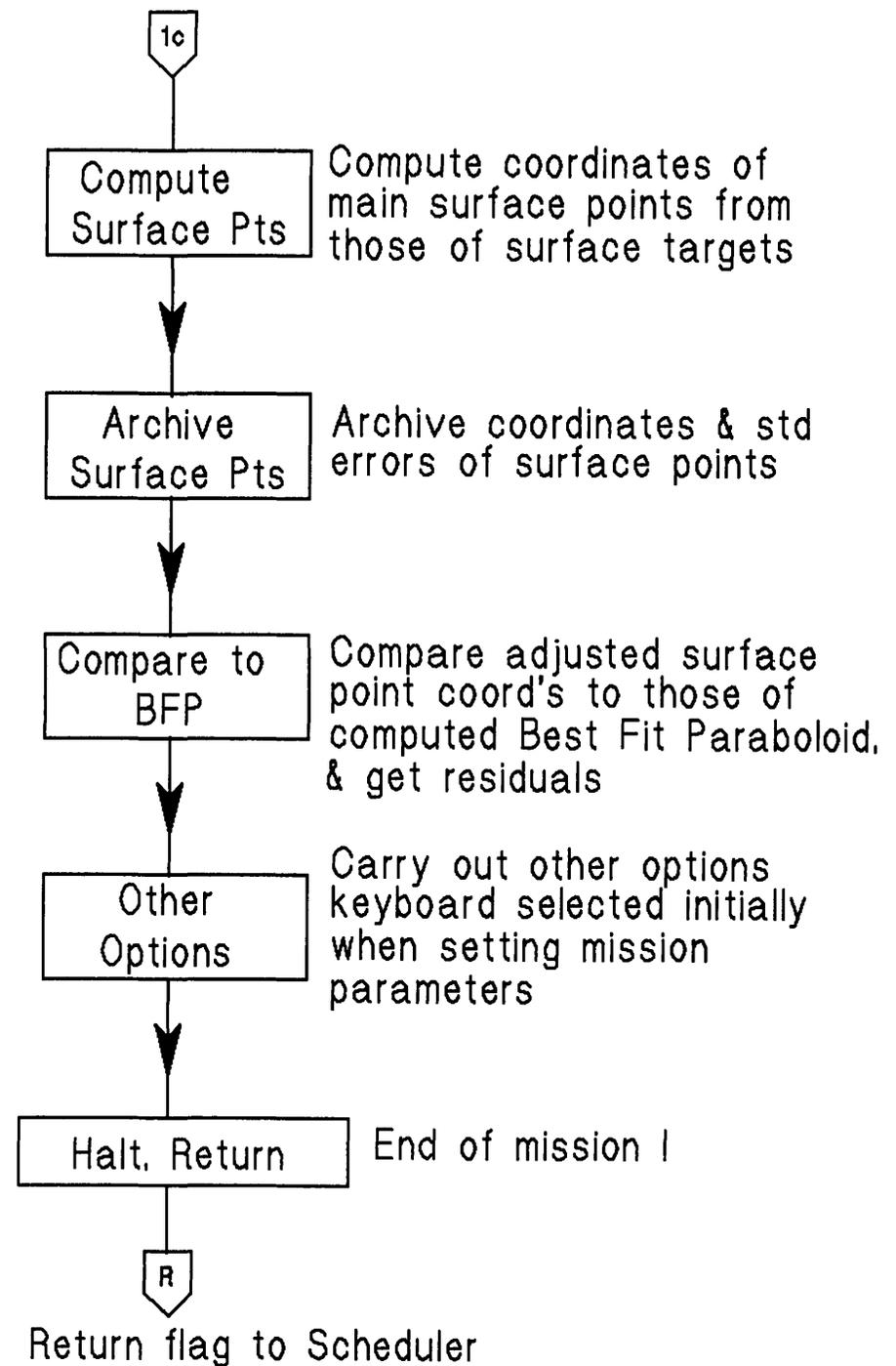
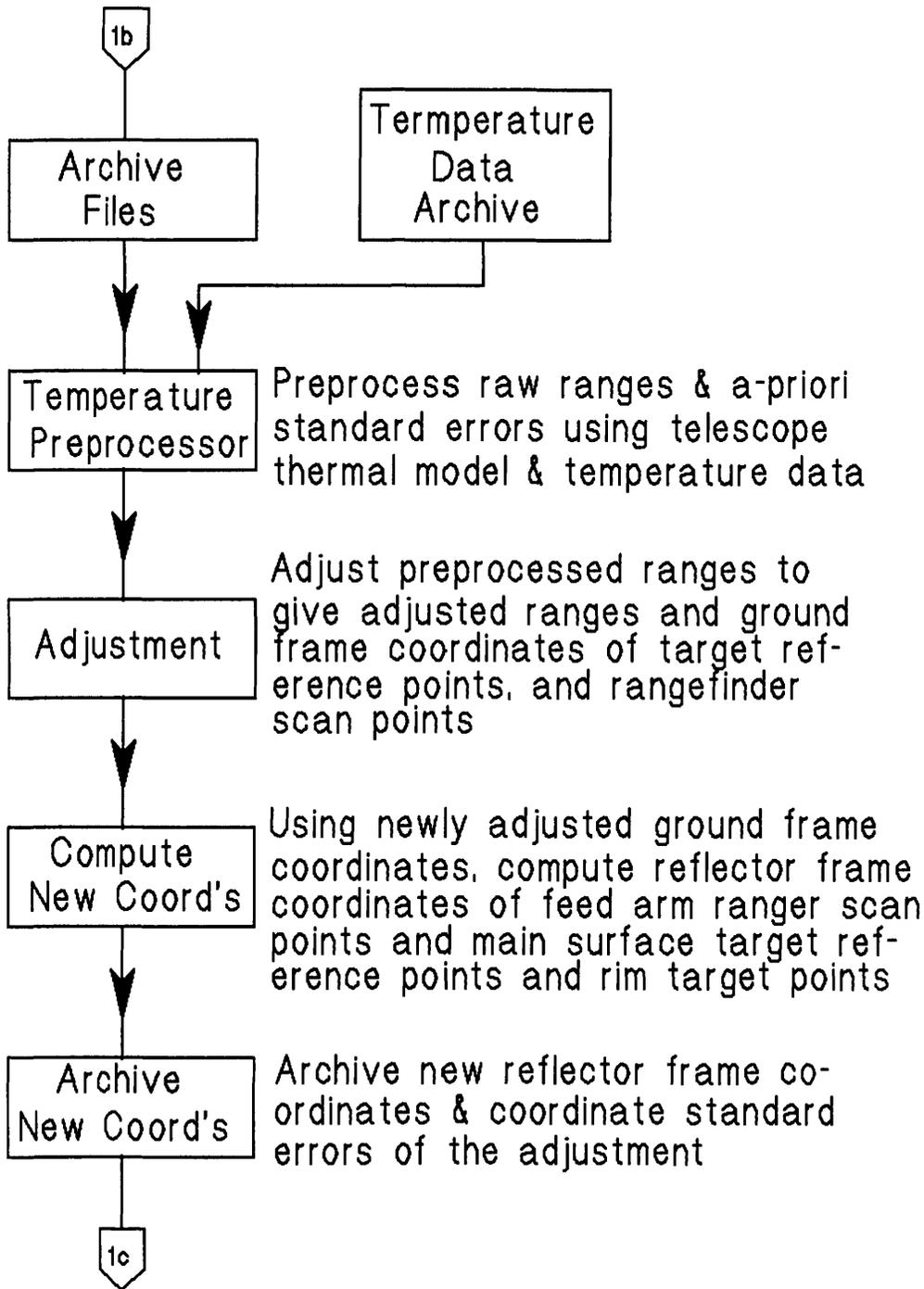
Figure 3. Possible Scan Frame Sequence For The Subreflector.

Figure 4. Metrology Mission Flow Charts.









Mission [5]  
Subreflector Scan, Static

