DIGITAL CONTROL OF AN ALTITUDE-AZIMUTH TELESCOPE TO SIMULATE EQUATORIAL TRACKING

Thesis

Submitted To

Graduate Engineering & Research

School of Engineering

UNIVERSITY OF DAYTON

In Partial Fulfillment of the Requirements for

The Degree

Master of Science in Electrical Engineering

by

Darryl W. Brayman

UNIVERSITY OF DAYTON

Dayton, Ohio

August, 1993

DIGITAL CONTROL OF AN ALTITUDE-AZIMUTH TELESCOPE TO SIMULATE EQUATORIAL TRACKING

APPROVED BY:

Dana B. Rogers, Ph.D.
Professor of Electrical Engineering
Committee Chairperson

Donald L. Moon, Ph.D.
Chairman, Department of Electrical Engineering
Committee Member

Anthony J. Evers, M.S.E.E.
Associate Professor of Electrical Engineering
Committee Member

Donald L. Moon, Ph.D.
Interim Associate Dean/Director
Graduate Engineering & Research
School of Engineering

Joseph Lestingi, D.Eng., P.E.
Dean
School of Engineering

ABSTRACT

DIGITAL CONTROL OF AN ALTITUDE-AZIMUTH TELESCOPE TO SIMULATE EQUATORIAL TRACKING

Brayman, Darryl Wayne
University of Dayton, 1993

Advisor: Dr. D. B. Rogers

This report presents the system requirements and engineering design issues and constraints involved in the development of a computer control system for an altitude-azimuth telescope to perform slewing, setting, and equatorial tracking functions.

As a basis for the design, an astronomy tutorial is included, followed by a derivation of the coordinate conversion algorithms. This is followed by a detailed discussion and analysis of many of the major technical issues involved in determination of torque, velocity, and other system requirements. Equations of altitude and azimuth track velocity and acceleration are derived, analyzed, and presented in three-dimensional plots versus hour angle and declination. Polar and zenith tracking constraints are discussed, as well as requirements for slew and track accuracy and resolution.

The paper presents a hardware design solution capable of simultaneously controlling two stepping motors, using microstepping techniques to produce track resolution to 1.125 arc seconds. The design incorporates a programmable-frequency pulse generator

circuit capable of producing any of 765 distinct velocities over a dynamic range of more than seven orders of magnitude. A software-controlled ramping routine permits fast slew operation to five degrees per second. In addition, the design incorporates position feedback loops sensed at both rotational axes, as well as the motor shafts. A multi-register routing network controls data flow in and out of the PC controller.

Finally, a preliminary software design is provided with some low-level coding. In addition to providing control for ramping, the software design provides for both programmed and manual slews, slow-motion manual slews for setting on the target object, and autonomous tracking functions.

ACKNOWLEDGEMENTS

I wish to thank Dr. Dana Rogers, my advisor, for lending his technical expertise through guidance and advise, and especially for his patience during the extended course of this project. I also wish to thank Bell Industries of Dayton, Ohio, for providing data books, which were used extensively in the hardware design stage.

I am also grateful to my mother, Priscilla Brayman, for assisting with typing some of the earlier chapters. Finally, let me express my deepest appreciation to my wife, Joan, and both of my parents for their patience and encouragement, which was instrumental in bringing this project to completion.

Darryl W. Brayman

PREFACE

The author's intent of this project was to do an electrical engineering thesis with an application into astronomy, the interest of which had been triggered by an undergraduate course on the subject. Real time control was also a strong interest, developed in a graduate electrical engineering course in robotics. Naturally, telescope control was a logical combination of these two disciplines.

Telescope control technology matured considerably in the late 1970s and 1980s. The most apparent examples are the popular Meade and Celestron portable Schmidt-Cassegrain telescopes, which have advanced rapidly, due to the fierce competition between the two. These telescopes are of equatorial design, mounted in such a way that one of the telescope's pointing axes is aligned with the earth's rotational axis. Although difficult to construct mechanically, the equatorial design simplifies the automated celestial tracking solution.

The altitude-azimuth (also known as altazimuth or "alt-az") mount is a much simpler design to construct, and much more economical, especially for larger apertures. The so-called "Dobsonian" alt-az telescopes, sold largely by Coulter Optical Co., sell for far less than equatorials of comparable size. As an example, a 17 1/2" Dobsonian, such as the Coulter Odyssey II, is currently priced

for as little as $1145.00 (plus shipping and handling). A 16" equatorial like the Meade DS-16A starts at about twice that price.

The problem with altazimuth instruments is that the automatic tracking feature is difficult to implement, and requires complex computer control. However, savings like that described above can buy a lot of computer control. Certainly, this was begging the question: "Could a computer-controlled alt-az telescope be developed to track equatorially with reasonable tracking precision?" Perhaps the bigger question one might ask is: "Does such a system have a practical application sufficient to justify the cost and the effort?"

The answer to these questions is "yes" for larger telescopes. Early in the research effort, it was learned that some of today's largest research instruments in development and the new multiple-mirror telescopes are designed in this fashion. The answer to the question of practicality is "unlikely" for smaller telescopes, those less than eight inches in aperture. However, medium- to large-aperture portable telescopes may provide a practical application. This paper will discuss some of the major challenges facing anyone who should try this with a medium-aperture telescope.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Purpose

The purpose of this paper is to discuss the technical issues involved in the development of a computerized control system for an altitude-azimuth telescope to perform simulated equatorial tracking. More specifically, the purpose is two-fold: 1. to identify the technical requirements and engineering constraints involved in the retrofit of an 8-inch home-built Dobsonian telescope with a digital control system. 2. to document an initial design of such an experimental digital controller for proof of concept.

Statement of Problem

Astronomical telescope tracking is generally preferred during visual observations and required for astrophotography. Normally, this is accomplished by driving the telescope tube assembly on an equatorial mount. Generally, a servo or timing motor is used to turn the telescope at a constant angular velocity equal and opposite to the earth's sidereal rotation rate, 359° in 24 hours.

Electronically-controlled motor drives are now standard equipment on many commercially available telescopes. These drive

systems are used to precisely maintain track of celestial "targets" as they are observed, studied, or photographed. Without this automatic tracking feature, even small (8-inch) reflectors at medium to high powers become cumbersome, since they require frequent pointing adjustment to counter the earth's rotation, and keep the object in the field of view. For instance, at 200×, to keep an object in the middle third of the field (where the best image is obtained), the scope must be repointed at least every 20 seconds! For photography, even (relatively) short exposures make precise automatic tracking a must, and manual corrections are still required for long exposures.

Of course, it also becomes necessary to properly align the drive axle with the earth's rotational axis. Such alignment is achieved by slanting the drive axle from the vertical by an angle complementary to the viewing latitude. For purely visual use, accuracy of polar alignment is reasonable within a few degrees (°). Photography requires a precision polar alignment within a few arc minutes (′).

At most latitudes, however, the weight of a large telescope tube assembly on an equatorial shaft can cause substantial bending stresses. This can result in increased bearing wear and tracking errors. In addition, some equatorial designs require the use of counterweights, applying further loading to the shaft. "Whereas an 8-inch (reflector) in a simple altazimuth mounting (may weigh only) a total of 40 pounds or less, a 20-inch telescope, properly mounted, requires a sturdy equatorial support, weighing a total of 2 tons!

(65: 13)" Furthermore, if portable, the large equatorial will be weakened by the adjustable wedge, required to accommodate a range of geographical latitudes.

Proposed Solution

A few large research instruments, such as the multiple-mirror telescope (MMT), have managed to avoid this problem by using an altitude-azimuth mount. Tracking is accomplished by using a complex computer control system to drive both altitude and azimuth axes simultaneously. Since the azimuth shaft is vertical, the axial loading that caused such a problem with the equatorial only tends to compress the shaft, while bending stresses are virtually zero, as long as the tube assembly is properly balanced. The altitude shaft is really the tube assembly itself, and as long as the tube truss or housing is of sound construction, the only appreciable flexure should be along the length of the tube. Tube flexure is a separate problem that must be dealt with independent of the mounting used.

This same idea can be applied to the large portable instrument. By using an altazimuth mount, not only is the polar shaft loading eliminated, but so is the adjustable equatorial wedge and the need for counterweights, if carefully balanced by design. The result is a potentially lighter, more transportable telescope structure that is less costly and simpler to construct. In addition, latitude adjustments can be made with a few keystrokes (software-regulated), as opposed to the required mechanical adjustment of an equatorial wedge.

Equatorial tracking can be simulated by the use of two digitally-controlled motors, one for each of the two degrees of freedom. By programming a microcomputer with the appropriate coordinate system transformation algorithms, one could generate a tracking trajectory to approach that of a true equatorial system.

Although the payoffs have been discussed, there are drawbacks introduced when pursuing this avenue for tracking capability. Unlike a standard equatorial, both axes must be driven simultaneously over a wide range of speeds, just to create the proper tracking trajectory. If using stepping motors, the discreteness of steps causes a complex variance of positional error as a function of position [$\Delta x = f(x)$]. In addition, it requires software to perform real-time trigonometric calculations to control and monitor position. Finally, for long-exposure photography, a third driven axis is required to correct for the rotating field of view, as described in Chapter 2. Thus, all three degrees of freedom, pitch, yaw, and roll, must be controlled, unless the telescope is used exclusively for visual work or short-exposure photography.

Procedure and Scope

Cost and time limitations precluded the approach of building a new large portable telescope. The next alternative was to retrofit an existing eight-inch home-built reflector on an altazimuth mounting, by adding motors, position sensors, and a computer controller. Due to additional time constraints, the scope of the project was finally restricted to carry only through a preliminary design stage. Jean

Texereau defines the "standard telescope" as an eight-inch f/6 or f/8 Newtonian reflector that employs an altazimuth mounting. This is perhaps the simplest and least costly design for an amateur to build for the quality it can produce. However, it is commonly accepted that automatic tracking with such a system is impractical. On the other hand, applying the concept to a relatively inexpensive system (as this) can lead to a more practical system application, such as a 17 1/2" or larger instrument.

One goal of this project would be to minimize cost. According to one source, "several large telescopes have been successfully computerized, but the expenses these projects experienced, both in hardware and software, are beyond the reach of many smaller observatories. (66: 1)" The good news is, "...although computerization can bring about greater cost savings on larger telescopes per dollar spent on the computerization project, the dramatic price drops in recent years for fast and reliable computer hardware have made computerization of smaller telescopes economically feasible. (66: 7)"

Before proceeding, it is important to reiterate that this will involve the design of an experimental system, which is not intended to meet all the requirements of a "practical" system. Instead, it will differ in a variety of ways:

a. As previously suggested, if one is going to go through the trouble of computerizing an altazimuth telescope for tracking, one would probably wish to apply this to a larger scope (17 1/2" or larger) whose performance could fully justify it.

b. A more practical system would use a small microprocessor-based hand controller, rather than a whole personal computer. Such controllers have been developed for equatorials for guiding, slewing, and remote focusing.

c. Due to the high cost of precision motors and position sensors, as well as precision machining, photographic tracking accuracy will not be pursued. Instead, the goal will be to maintain track of an object in a half-degree (30′) field of view in any direction (except passing within 5° of the zenith) for one hour. Some actual software coding will be written, but not in every detail. Attempts will be made to identify certain applications where software models might be used to compensate for hardware anomalies.

d. No compensation will be made for the rotating field problem (See Chapter 2) that comes with the altazimuth tracking approach.

The remaining chapters are arranged in the following manner: Chapter 2 is a tutorial which provides a more detailed picture of the telescope, in general, its basic design and application. It is intended as an astronomy background for the engineer/layman astronomer. In Chapter 3, the mathematical tracking algorithms are derived. Chapter 4 establishes the design requirements (i.e., speed, acceleration, torque, etc.) Chapter 5 discusses the hardware design and construction of the control system driving function, including motor selection, gearing schemes, and computer control hardware interface. Position sensors are introduced, and methods of performing coarse and fine position sensing are discussed. Where

possible, cost, precision, and availability are considered. Chapter 5 also addresses some hardware configuration and integration issues, primarily those dealing with the telescope modifications necessary to implement the control system. In Chapter 6, a strawman software design is presented. Finally, Chapter 7 is a summary, providing any final conclusions and/or recommendations.

CHAPTER 2

ASTRONOMY AND TELESCOPES - A TUTORIAL

Background

Considered by many as man's oldest science, astronomy dates back to the days of the ancient Greeks of 5th and 6th centuries B. C. At that time, known as the "Age of Reason", Plato, Pythagoras, and others postulated various theories on the makeup of the universe. Although these theories were more philosophical than scientific, they soon gave way to the first attempts to measure sizes and distances of the sun and moon (Aristarchus, 310 B.C.) and explanations of planetary retrograde motion.

By A. D. 1600, Copernicus' improved model of the Solar System and Tycho Brahe's observations and measurements began what might be called the "age of astronomy". During the 17th century, Kepler and Newton respectively produced mathematical models of orbital characteristics and gravitation, while Galileo's two-inch telescope first revealed lunar craters, phases of Venus, Saturn's rings, and four moons of Jupiter, to name a few.

Today we have optical telescopes of various sizes and designs to as large as a few meters in diameter. Professional astronomers are now largely dedicated to mathematical applications of theoretical

physics to study details of objects to determine their origins. Meanwhile, observations are left to be performed almost exclusively by amateurs. Astronomy may be considered a hobby for most amateurs, but often serious amateurs select particular areas of interest or "specialties". Further, they may be involved in scientific research in support of astronomical and/or astrophysical studies. In any case, the telescope has gained a wide variety of applications for both casual observing and scientific research.

Applications

In addition to visual observations, here are a few of the other areas in which telescopes are largely utilized:

Astrometry

Precise measurement of an object's position with respect to other "fixed" objects (stars) for ephemeris predictions (especially a comet, asteroid, or other object with readily apparent motion with respect to background stars.) Stars do move relative to each other; however, all but a few very close stars change positions so slowly that movement can be considered as negligible over decades of time.

Astrophotography

Photographic recording of celestial images includes both short- and long-exposure photography. Short exposures are used to record brighter objects such as the moon and planets, and may range

from fractions of a second to several seconds in duration. Long exposures are required for dimmer objects (comets, asteroids, nebulae, stars, and galaxies) and generally range on the order of a few minutes to over an hour in duration.

Photometry

Involves the use of a light sensing device (usually photoelectric) optically coupled to the telescope to measure the relative brightness of an object. This quantity, commonly referred to as the "apparent magnitude" of the object, is a measure of its perceived luminosity.

Spectroscopy

The study of emission and absorption spectra of a star or group of stars, as recorded photographically through a diffraction grating or similar device. Stellar spectral classification can then be used to determine the star's age, mass, temperature, and absolute brightness (magnitude). Distance can then be derived from the r^2 relationship between absolute and apparent magnitude. Spectral characteristics of extragalactic objects are also commonly studied for Doppler shifts. Corresponding relative velocities can infer the object's distance and rotation rate (if applicable), and provide input into the determination of the universal expansion rate and other significant cosmological data.

Basic Telescope Optics/Definition of Terms

As a review of basic optics, the simplest telescope has two essential elements: an ocular and an objective. The objective may be a lens or series of lenses (in a refractor - see Figure 1) or a mirror (in a reflector - see Figures 2 and 3). In either case, the parabolic surface of the objective focuses the incident light at the focal point, F, located at some distance away, referred to as the focal length, f_{obj}. The focal length is inversely proportional to the curvature of the objective.

One of the most important optical specifications of the objective is the focal ratio, FR, equal to the focal length divided by the diameter (aperture, in photographic terms), where FR is the ratio:

$$FR = f_{obj}/D_{obj}$$

The focal ratio, conventionally denoted as "f/FR" and pronounced "f-(FR)", represents the inherent light gathering capability of the objective for a given magnification. This number is particularly important in astrophotography. As in ordinary photography, a focal ratio of f/5 produces an exposure in one-fourth the time it would at f/10. Thus f/5 is said to be a "faster" focal ratio than f/10.

The ocular or eyepiece, a lens or (more often) a series of lenses, is placed at a distance beyond the objective focal point, F, equal to its effective focal length, f_{OC}. This is to correct the refracted light, allowing it to emerge parallel (in focus) for the eye of the observer.

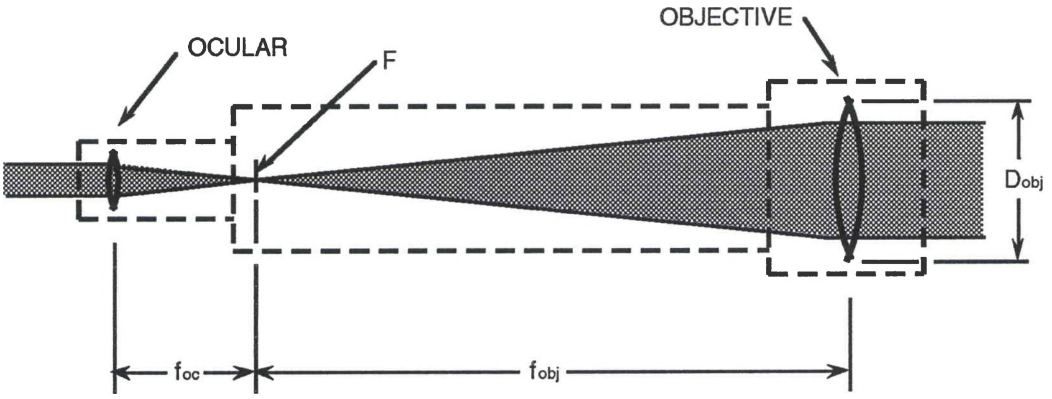


Figure 1

Refractor Telescope

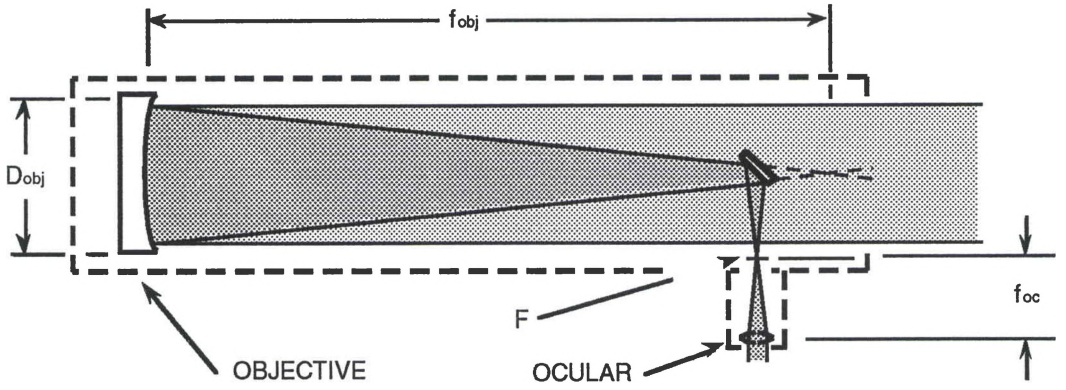


Figure 2

Reflector Telescope

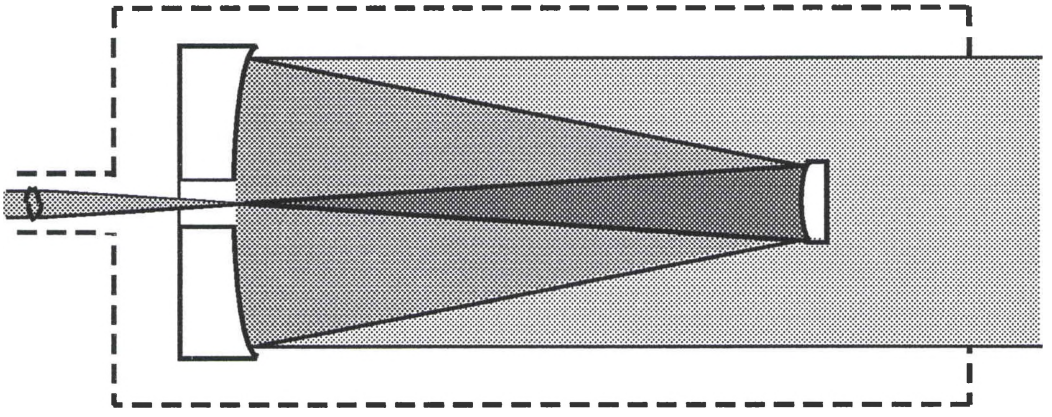


Figure 3

Cassegrain Telescope

The effective magnification, M, is the ratio:

$$M = f_{obj}/f_{oc}$$

Since f_{obj} is a fixed quantity, magnification (power) is varied by changing eyepieces. Although highly regarded by the most inexperienced observers, magnification is usually less significant than other factors, since power is increased at the expense of light gathering ability. In fact, this is an inverse-square relationship (i. e., doubling the power reduces the brightness to one-fourth its original magnitude.)

Wide-field objects, those of relatively large angular size, are best seen at low powers, since angular field of view is really an inverse of magnification. The proportionality constant varies, however, depending on the apparent field of view, as determined by the design of the ocular. A typical field of view would be about 30′ (one-half degree) at 100× magnification, 15′ at 200×, etc.

Resolution is another important optical specification to consider. Good image fidelity first demands a precisely-shaped objective surface (both surfaces in the case of a refractor objective), down to millionths of an inch. Standard home mirror-grinding techniques are capable of producing such precision in a six- or eight-inch mirror with an inexpensive mirror-grinding kit and a few simple tools. Nonetheless, diffraction establishes a practical limit on resolution that cannot be improved upon, regardless of the surface

quality. It can be shown (65: 3-6) that this angle of resolution (in radians), known as the "diffraction limit", is determined as:

$$\rho_r = 1.22\ \lambda / D_{obj} \quad \text{where} \quad \lambda = \text{wavelength}$$

Although the visible spectrum ranges from 4000 to 7000 Å, the eye is most sensitive to 5600 Å, or 0.00056 mm. For this purpose, this wavelength is selected in resolution calculations. Since λ is a fixed quantity, resolution can only be further improved by using a larger objective.

Example:

Objective:

focal length: 2000 mm

diameter: 200 mm

Oculars:

focal lengths: 32 mm, 20 mm, and 10 mm

Focal Ratio: FR = 2000 mm / 200 mm = f/10

Magnification:

w/ 32 mm: M = 2000 mm / 32 mm = 62.5×

w/ 20 mm: M = 2000 mm / 20 mm = 100.0×

w/ 10 mm: M = 2000 mm / 10 mm = 200.0×

Resolution (diffraction-limited):

ρ_r = (1.22) (0.00056 mm)/200 mm

$= 3.42 \times 10^{-6}$ radians

or 0.705″ (arc sec)

Telescope Designs

The previous section clearly indicates the advantage of larger aperture telescopes, namely light gathering ability and resolution. Long focal lengths can produce high magnifications at the expense of light gathering power. This important tradeoff must always be dealt with first, depending on the application and telescope design.

Refractors (Figure 1) generally have long focal lengths and small objectives, producing focal ratios of f/10 to f/20, typically. Inexpensive versions of these instruments are frequently seen advertised with the high-power marketing ploy, but these are generally of very limited use in astronomy. Good refractors are usually three inches and larger in diameter. They are also expensive, and the price goes up rapidly with the smallest increase in aperture. Good refractors are most suitable for observing bright objects of small angular size, such as planets and lunar craters. The refractor offers unobstructed optics, and a sealed enclosure for reducing internal convection, both serving to enhance image quality.

Reflectors can be produced with larger objectives and shorter focal lengths than refractors, and at a lower cost. With typical focal ratios of f/5 to f/15, they are perhaps more suitable for dim wide-field objects (i.e., comets, star clusters, nebulae, and galaxies), but are satisfactory for planets as well.

Reflectors come in a variety of designs. The simplest and most common is the Newtonian (Figure 2). A reasonably simple design to

build for the amateur, the Newtonian consists of a parabolic objective (also called a primary) mirror and a small flat (secondary) mirror to deflect the converging light cone out of the incident light path to the ocular. Unfortunately, the diagonal secondary also creates an obstruction for the light entering the tube to the primary. The reduction of available light, usually about 15%, is not significant, but the diffraction impact can be. The obstruction can cause an apparent resolution that detracts from the diffraction-limited figure discussed earlier. In addition, the impact of the secondary support vanes (often called a spider) can further contribute to this problem.

Sometimes an optical glass "window" is placed at the front end of the tube to reduce convection and seal out dust and moisture. However, the window could also be used to support the secondary, eliminating the need for a spider. Another version is the Schmidt-Newtonian, in which the window is, itself, a refracting medium, used to optically correct for a spherically-, rather than parabolically-, ground primary mirror. Newtonians are most common with focal ratios of f/4 to f/10, but they can be slower.

Another common reflector design is the Cassegrain (Figure 3). The Cassegrain requires a slightly convex secondary mirror to extend the focal length of the primary and project the emerging light cone through a hole in the primary to the ocular. A flat secondary in this design would need to be more than half the diameter of the primary. The Cassegrain is more costly than the Newtonian, but it provides a long focal length in a much shorter tube, lending itself to portability and a reduced moment of inertia. This last advantage is especially

important in relaxing drive system requirements. Focal ratios of Cassegrains are typically on the order of f/15, or f/10 in the case of a Schmidt-Cassegrain.

Various other reflector designs have been developed, ranging from the unusual to the exotic. One example is a Coude´ design, in which the final optical path traces one of the rotational mounting axes, allowing the observer the freedom to slew a large telescope to any pointing direction without having to change his viewing position. Another interesting design is the Schiefspigler, a type of folding telescope which reflects the light cone slightly off-axis. This unique design eliminates the obstruction problem seen in other reflector designs.

Mounts and Coordinate Systems

Astronomical telescope mounting structures generally come in one of two varieties - the altitude-azimuth mount (also altazimuth or alt-az) and the equatorial mount. The alt-az method (see Figure 4) is a simple design in which the telescope tube can be slewed about a horizontal axis (altitude), which is, itself, mounted on a vertical axis for slewing in azimuth. This design is often seen supporting inexpensive refractors. An instrument mounted in this fashion is aimed by utilization of the altitude-azimuth coordinate system. In this case, a star's position (direction) is, in essence, measured as a compass bearing and an elevation angle above the horizon at that bearing.

During the past few years an altazimuth mount designed by John Dobson of the San Francisco Sidewalk Astronomers has become very popular. Originally intended for instruments in the 20-inch range, it is equally practical for smaller telescopes. It's main attributes are portability, stability, and extreme simplicity... In its simplest form, the Dobsonian mount relies upon relatively low magnification and tracking by the observer gently pushing the tube from time to time. The secret of this design is that it is held in position only by friction. The friction is large enough so that light wind or adjusting the eyepiece does not cause movement. (65: 385)

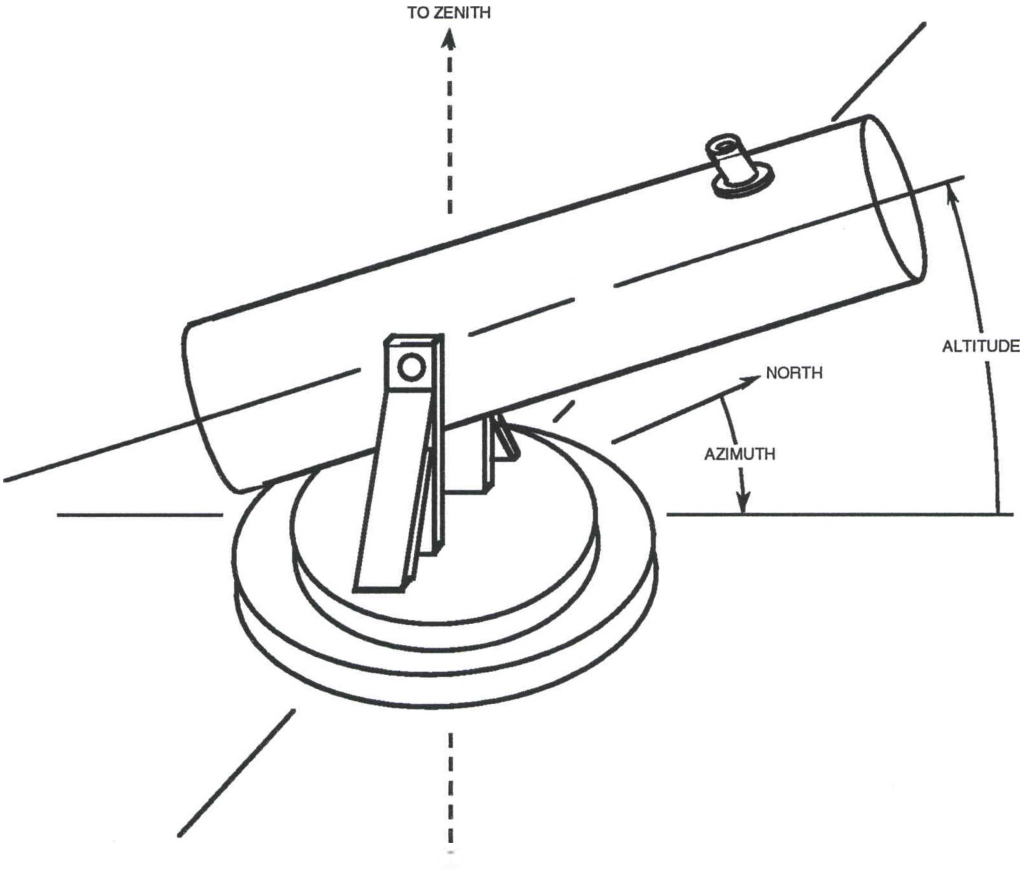


Figure 4

Telescope with Altitude-Azimuth Mount

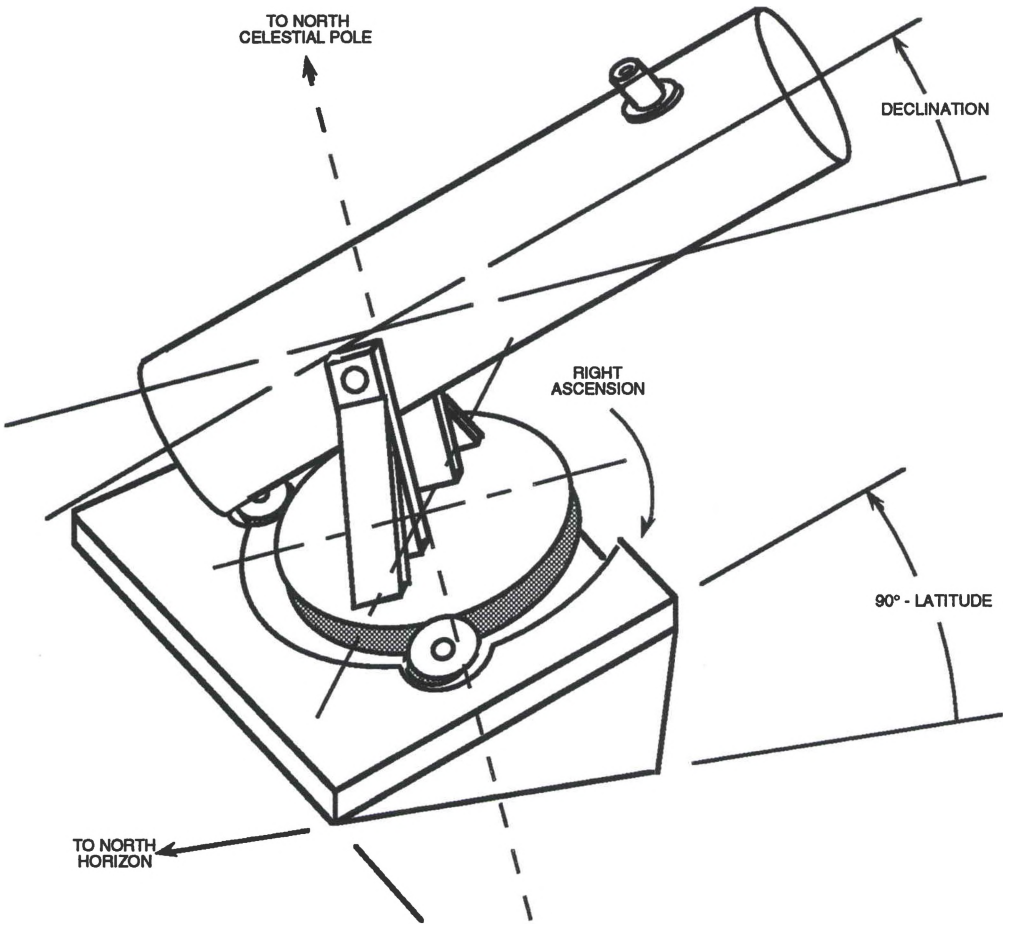


Figure 5

Telescope with Equatorial Mount

The equatorial method (see Figure 5) involves the alignment of one axis or rotation (called the polar axis) with the earth's rotational axis, which permits scanning East and West. The other axis allows slewing North and South, tracing an angle up to ±90° with respect to the plane of the earth's equator. Essentially, an equatorial system is an alt-az configuration in which the azimuthal axis has been offset from the vertical by an angle whose complement is the viewing latitude: slant angle = 90° - latitude. Thus, at the North or South pole, an equatorial mount is the equivalent of an altazimuth mount.

The equatorial celestial coordinate system is analogous to the terrestrial coordinate system used for world mapping. Similarly, stars are mapped on the celestial "sphere" according to their position in right ascension (RA) and declination (DEC). Right ascension is measured about an axis parallel to the earth's axis, and thus measures East and West, similar to longitude. Units of RA are hours and minutes, and there are 24 hours of RA around the sphere (i.e., 15° of arc per hour). Declination, on the other hand, serves as the measurement above or below the equator plane. Like earth latitude, declination is measured in degrees in the range, $-90° \leq DEC \leq +90°$.

Tracking

Equatorial Systems

The primary advantage of an equatorial system is in its tracking utility. The equatorial telescope is first carefully calibrated for alignment with the earth's axis, by any of several polar alignment processes, depending on the degree of precision required. Once this is accomplished, the tube assembly can be motor-driven about the right ascension axis. This "clock drive" is turned at a precise constant rate of about 15° per hour (one RA hour per hour), equal and opposite to the terrestrial rotation rate. "This will immobilize the entire sky with respect to the instrument." (65: 232)

The rate of 15° per hour is actually the solar rate, the earth's angular velocity relative to the sun. If high precision tracking is not required (i.e., visual observation), the solar rate can be used as a

close approximation. The reason is that the earth completes one rotation in about 23 hours, 56 minutes and 4 seconds. In order to maintain solar synchronization, an additional 0.986° advancement is required to compensate for the earth's orbital advancement. The true star tracking rate or "sidereal" rate, is actually 15.041° per hour. (Sidereal time is discussed further in Chapter 3.) Some telescope drive gears (worm wheels) are deliberately made with 359 teeth. This way, the worm can be turned at a standard 1/4 RPM to achieve the desired sidereal rate. However, software can now be easily used to scale the rate for use with the standard 360-tooth gear, and experienced amateurs claim this is a more practical approach. All tracking is accomplished about the right ascension axis; the declination axis "is used only to aim at the object at the start of observation, and is not again adjusted during observation if the equatorial mount is properly aligned." (65: 232)

Altazimuth Systems

Until recent years, celestial tracking with an altazimuth system was generally impractical. For the average amateur astronomer, however, this was not a serious problem. Small telescopes of eight-inch diameter or less could be equipped with an equatorial mount and achieve very accurate tracking without significant mechanical loading of the right ascension shaft. As the "large-aperture generation" developed, however, requirements of equatorial mounts became significantly more demanding, particularly in size, weight, and rigidity.

In choosing the altazimuth tracking system design approach, various challenges quickly emerge. Both axes must be driven simultaneously at widely varying speeds to simulate the equatorial trajectory. The problem of accelerating and decelerating in both degrees of freedom must be addressed, and moments of inertia must be kept to a minimum. Like polar alignment for the equatorial, the alt-az system requires initial calibration in the form of zenith alignment. (66: 106) Additionally, however, the alt-az will need to be "direction-calibrated" (i.e., told where South is). Finally, the altazimuth design must deal with the problem of a rotating field during astrophotography. The rotation follows a complex relationship involving many factors, including the observer's position (latitude and longitude), the star's position (RA/DEC), as well as the time and date. In general, the deviation rate is severe at East and West, and relatively minimal at North and South.

No matter what design is used, one thing is for certain: a precise driving mechanism is required to prevent any apparent motion of objects in the field of view at the eyepiece. For example, a 45° apparent field at 100× magnification presents an actual field of 0.45° or 27′. At the sidereal tracking rate of 15.041°/hour, a star crosses the diameter of this field in 1 minute, 48 seconds, with no tracking at all. More practically, to keep an object initially centered within the middle third of the field of view for three minutes or more of observation, tracking accuracy must be within 10% of the sidereal tracking rate. This may not sound like precision, but it also must be noted that tracking corrections must be made smoothly, so

as to avoid "fluttering" or "jerky" movement of objects in the field, such as might occur with a stepping motor with insufficient resolution. This may increase the demand of precision to within 1% or less. For astrophotography, an accuracy of 0.05% or better may be required to prevent smearing or streaking in long exposures, for instance, 10 minutes or longer.

Poncet Mount

A fairly simple, and interesting compromise between equatorial and alt-az mounting methods is the Poncet mount or "equatorial table". In effect, a Dobsonian can be converted into a limited equatorial system, if assembled with an equatorial table. "Reduced to its simplest elements, the Poncet mounting has a South pivot point and on the North end two Teflon bearings riding on a smooth inclined plane. (65: 387) The inclined plane lies parallel to the earth's equator and supports a platform which can then slide in an equatorial trajectory. This design makes short-term tracking relatively simple. The Poncet mount can be driven with a nut and threaded rod acting as a worm, attached to a timing motor. However, the design of the Poncet mount limits its tracking trajectory (duration) to about an hour. The key limiting factor is the tracking error generated by the straight tangential threaded rod, as it deviates from the desired curved path. An additional drawback is that the Poncet mount can be somewhat top-heavy, limiting its stability.

Other Driving Considerations/Requirements

Tracking motion is only one of the problems to be solved in computer telescope control. Unless slewing is intended to be purely manual, one may wish to add a very high speed motion control capability about both rotational axes, no matter which mounting method is used. Once an object is in the field of the finderscope, a somewhat slower setting speed may also be desired to center the object and bring it into the main telescope field of view. Other than the tracking speed, this is the most important speed for purely visual requirements.

If long-exposure photography is to be an application, the tracking speed and polar alignment must be very precise. In addition, a fourth speed is necessary for guiding. Guiding is the process of making small corrections during an exposure at high magnification to compensate for imperfect polar alignment, aberrations, and other sources of tracking error. Guiding speed would normally be on the order of 10% of the tracking speed, but relative to the tracking motion. (64: 555) Declination guiding and setting in an equatorial telescope would require an additional motor controlled separatelym but the altazimuth system requires two motors anyway.

Assuming all four speeds are required, the following summarizes what these speeds might be, expressed as an absolute rate, and relative to the standard tracking velocity of 15.041´´ per sec.

Bidirectional rotation is required for all speeds (except tracking, unless the scope is to be used in the Southern hemisphere).

Application	Absolute Speed	Relative to Tracking
Guiding	1.5-3´´/sec	0.1 - 0.2
Tracking	15.041´´/sec	1
Setting	4-5´/sec	16-20
Slewing	3-9°/sec	720 - 2160

This setting speed requirement was based on the assumption that centering an object in the field of view (FOV) from one-third of the field away should take 2 to 30 seconds. In an 8×50 finder, the FOV is 6° wide, while the telescope field at 100× might be about 30´ wide. Unless these two speeds are handled separately, the following tradeoff must be made: 2-2.5 sec in the telescope corresponds to 24-30 seconds in the finder. The slew speed was based on the assumption that a long slew should traverse an appreciable portion of the sky (45° to 135°) in 15 seconds or less. In summary, operating speeds should range over about three orders of magnitude, four orders for photographic applications.

CHAPTER 3

COORDINATE CONVERSION EQUATIONS

Basis

In order to define operational system requirements, it is first necessary to discuss the complex mathematical basis behind these requirements. Hence, a separate chapter has been dedicated for this purpose. Many of the equations presented and derived here would comprise a large portion of the software discussed in Chapter 6. In a sense, this chapter would provide the lowest level of documentation for that software, since it provides the theory behind it.

Geometric Symbology

To facilitate formula manipulation and derivation, it is first necessary to assign symbols to the relevant parameters used in this chapter, listed in the order they are introduced. Expressed units are in parentheses. From this point on, unless otherwise noted, let:

Y = Year

M = Month

D = Day

JD = Julian Day Number (Days)

C = Century - 1 (Centuries)

R = Gregorian Reform Factor (Days)

k = Sidereal/Solar Proportionality Factor

T = Julian Century Relative to Epoch (Centuries)

G_0 = Greenwich Mean Sidereal Time (GMST) at 0^h UT
(Hours, H/M/S, or Revolutions)

G = Greenwich Mean Sidereal Time (GMST) (Hours or H/M/S)

U = Universal Time (Hours or H/M/S), abbreviated UT

S = Local Sidereal Time (Hours or H/M/S)

θ = Longitude (Degrees or D/M/S)

ϕ = Latitude (Degrees or D/M/S)

A = Azimuth (Degrees)

E = Altitude [Elevation] (Degrees)

α = Right Ascension (H/M/S, Degrees, or D/M/S)

δ = Declination (Degrees or D/M/S)

Note: Subsequent uses of "A" and "E" as subscripts refer to parameters pertaining to the azimuth and altitude axes, respectively.

Julian/Gregorian Calendars

Standard calendar dates are cumbersome to use when doing astronomical calculations. To facilitate this, astronomers use the Julian and Gregorian calendars, which provide an actual count of days taken place since January 1, -4712 (4713 B. C.) The Julian calendar is only useful to October 15, 1582. The modern Gregorian reformed calendar should be used for subsequent dates. Each calendar produces a corresponding Julian Day Number, JD. By definition, a "Julian Day begins at Greenwich mean noon, that is at 12^h

Universal Time. (26: 23)" Therefore, dates at 0^h (Greenwich midnights) end in .5 exactly.

Meeus (26: 24) uses the following method for determining the Julian Day number. Consider a date in the following format:

YYYY.MMDDdd,

where dd represents decimal days.

Let Y = YYYY and M = MM.

If $M \leq 2$, let M = M + 12 and Y = Y - 1.

The Julian Day Number is then

JD = INT [365.25Y] + INT [30.6001 (m + 1)] + DD.dd + 1,720,994.5

where the INT function produces a truncated, rather than rounded, integer. However, if the date takes place after October 15, 1582 (Gregorian calendar), calculate the following quantities:

C = INT [Y/100] and R = 2 - C + INT [C/4]

Then add R to the previous total.

Sidereal Time

Chapter 2 briefly mentioned sidereal time, but, at this point, it should be explained in more detail. The commonly-used local civil time system is based upon geo-solar interaction, earth's rotation with respect to the sun. So is the less common Universal Time (UT) used

for world-wide standardization, also referred to as Zulu Time or Greenwich Mean Time (GMT). Neither of these time systems are adequate for astronomical activities and calculations, however. The reason is that while the earth completes 365.25 solar-respective rotations, it has completed 366.25 rotations relative to objects outside the solar system. The two rates differ by the factor, k = 1.002738. This anomaly requires the establishment of a sidereal-rate time system, one that "tells time" relative to the stars.

Sidereal time is approximately 00:00:00 at Greenwich at 12:00:00 Noon (12^h UT) on March 21 of each year. Local sidereal time for an observer anywhere on the earth is determined by subtracting the respective time-longitude offset of one hour per 15° Westward from Greenwich. The observer's local sidereal time gives the right ascension of all objects on the local meridian, a line drawn North to South through the zenith.

There are several ways to determine the sidereal time at Greenwich (GMST), depending on the standard epoch used. Sinnott (54: 558) uses epoch 2000 and calculates sidereal time in seconds (a very large number), then converts to hours, minutes, and seconds (H/M/S). Meeus' (26: 39) version uses epoch 1900 and calculates in revolutions, then converts to H/M/S. Sample calculations run using each method agree within 0.1 seconds. The Meeus formula is more attractive for current dates, since it avoids negative numbers, which become cumbersome in the conversion process. For this reason, the Meeus method was selected.

To determine sidereal time, it is first necessary to calculate JD at 0^h UT, as described earlier. Then determine the number of Julian centuries since 1900 by the following formula:

$$T = \frac{JD - 2415020}{36525}$$

The GMST is then determined as

$$G_0 = 0.276919398 + 100.0021359\ T + 0.000001075\ T^2 \quad \text{(revolutions)}$$

The result would then be converted from revolutions to hours, minutes, and seconds. However, since 1 second = $86{,}400^{-1}$ revolution, it can be shown that five decimal places is sufficient for one arc second accuracy for $|T| \leq 3.62$, or $1538 \leq Y \leq 2262$. Further, the T^2 term can be dropped, leaving

$$G_0 = 0.27692 + 100.00214\ T \quad \text{(revolutions)}$$

This can be converted by multiplying the decimal remainder by 24 to get hours, that remainder by 60 to get minutes, and the last remainder by 60 to get seconds. It should be kept in mind here that decimal seconds will not be accurate.

The previous formula gives GMST only at 0^h UT on the given Julian Day, however, and Julian Days ending in other than .5 will give erroneous results. "To find the sidereal time at Greenwich for any instant in hours and decimals, multiply by k (=1.002738), and add the result to the sidereal time at O^h UT. (26: 40)"

$$G = G_0 + kU$$

Finally, to convert this to local sidereal time (LST), subtract the time respective to the local longitude, West taken to be positive, or

$$S = G - \theta$$

where θ (longitude) is expressed as a time, using the rate of four minutes per degree of longitude.

Hour Angle

The local hour angle is the angle Westward (+) or Eastward (-) of the local meridian to a star or other object of known right ascension (α). It is determined simply by subtracting the right ascension from the LST. ($H = S - \alpha = G - \theta - \alpha$.) Thus, the right ascension of a star on the meridian is equal to the LST.

Equatorial to Altazimuth Conversion

Three angular quantities are required to determine the altitude and azimuth of a celestial object: the hour angle (H), derived from the right ascension (α), expressed in degrees; the declination (δ), and the observer's latitude (ϕ). The latitude is the key parameter that links the two coordinate systems. Figure 6 shows these relationships.

To derive the coordinate system transformation algorithms, consider Figure 7. Let the xyz coordinate system describe the altitude-azimuth domain and the $x'y'z'$ system represent equatorial

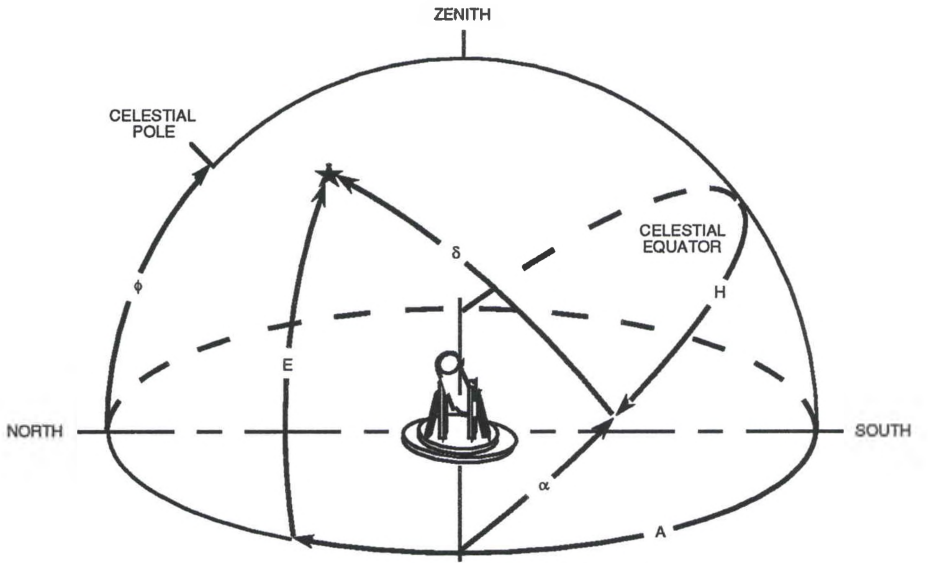


Figure 6

Relationships Between Coordinate Systems

coordinates. The diagram in Figure 7a shows the latitude relationship between the two systems, which gives the following:

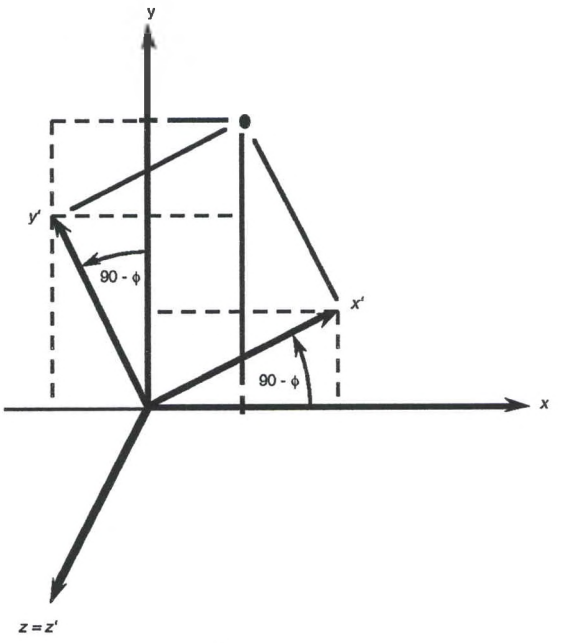
$$x = x' \cos (90°-\phi) - y' \sin (90°-\phi)$$
$$y = x' \sin (90°-\phi) + y' \cos (90°-\phi)$$
$$z = z'$$

which is equivalent to

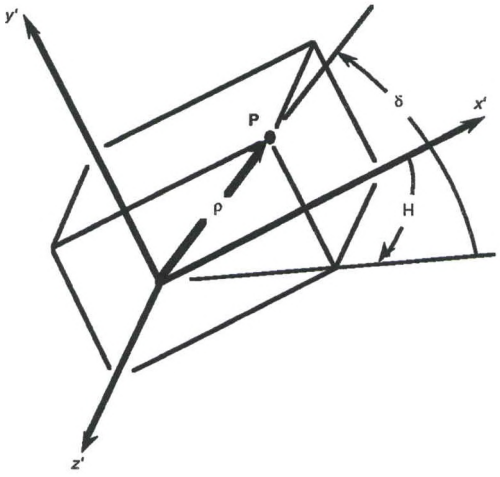
$$x = x' \sin (\phi) - y' \cos (\phi)$$
$$y = x' \cos (\phi) + y' \sin (\phi)$$
$$z = z'$$

which, in turn, can be expressed in matrix form:

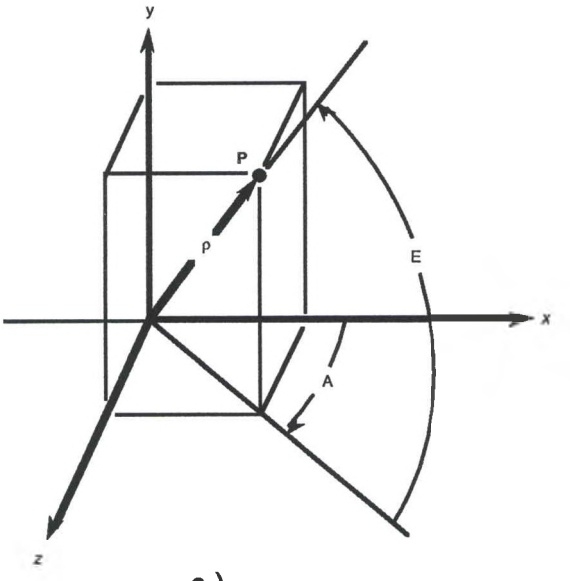
$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \sin \phi & -\cos \phi & 0 \\ \cos \phi & \sin \phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}$$



a.)



b.)



c.)

Figure 7

Derivation of Transformation Equations

Next consider a point P(x', y', z'), expressed in equatorial coordinates (H, δ) as shown in Figure 7b. Assume that the point P is a star, and the distance to it is ρ. The ρ is really a "dummy" parameter, since it is really irrelevant in pointing a telescope. By using ρ, however, a complete spherical coordinate system has been established. Therefore, P(ρ, H, δ) can be used to replace the Cartesian P(x', y', z'). From the diagram, it can be shown that

$$x' = \rho \cos H \cos \delta$$
$$y' = \rho \sin \delta$$
$$z' = \rho \sin H \cos \delta$$

Likewise, P(ρ, A, E) can be used to replace the Cartesian P(x, y, z) in the altitude-azimuth system (see Figure 7c). Note that ρ is the same. Similarly, this diagram shows that

$$x = \rho \cos A \cos E$$
$$y = \rho \sin E$$
$$z = \rho \sin A \cos E$$

Substituting these new expressions for x, y, z, x', y', and z' into the latitude equations generates the following:

$$\rho \cos A \cos E = \rho \cos H \cos \delta \sin \phi - \rho \sin \delta \cos \phi$$
$$\rho \sin E = \rho \cos H \cos \delta \cos \phi + \rho \sin \delta \sin \phi$$
$$\rho \sin A \cos E = \rho \sin H \cos \delta$$

Since every term has a ρ, these can now be dropped. The second equation produces the formula for altitude:

$$E = \sin^{-1}\left(\cos H \cos \delta \cos \phi + \sin \delta \sin \phi\right)$$

Dividing the third equation by the first produces the tangent of azimuth, such that

$$A = \tan^{-1}\left(\frac{\sin H \cos \delta}{\cos H \cos \delta \sin \phi - \sin \delta \cos \phi}\right)$$

Dividing numerator and denominator by cos δ yields

$$A = \tan^{-1}\left(\frac{\sin H}{\cos H \sin \phi - \tan \delta \cos \phi}\right)$$

Since $0° \leq A \leq 360°$, care should be taken when using this formula to ensure that the signs of the numerator and denominator are extracted to preserve the proper quadrant. Otherwise, a quadrature ambiguity will result.

The same process could be used to produce the inverse equations. In this case, the transformation matrix is identical, except for the signs on the cosine terms.

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \sin \phi & \cos \phi & 0 \\ -\cos \phi & \sin \phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

This, in turn, produces the following inverse expressions:

$$\delta = \sin^{-1}\left(\sin E \sin \phi - \cos H \cos E \cos \phi\right)$$

$$H = \tan^{-1}\left(\frac{\sin A}{\cos A \sin \phi + \tan E \cos \phi}\right)$$

As in the case for the azimuth expression, routines must be developed to preserve quadrature for the hour angle equation.

The following flow chart (Figure 8) illustrates the basic relationships of the parameters described in this chapter. In general, it summarizes that a telescope's pointing position, expressed in terms of altitude and azimuth, is a function of

1. the current date and time,
2. the observer's position, and
3. the star's position.

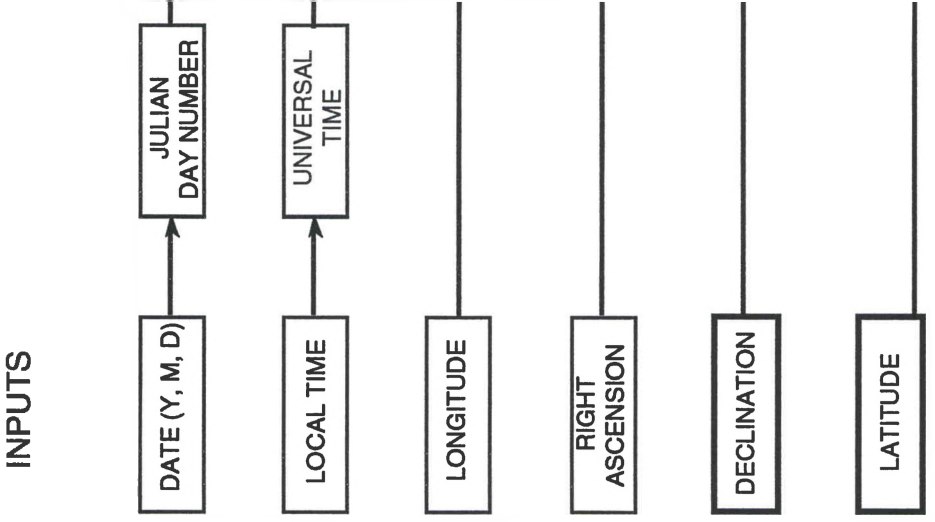
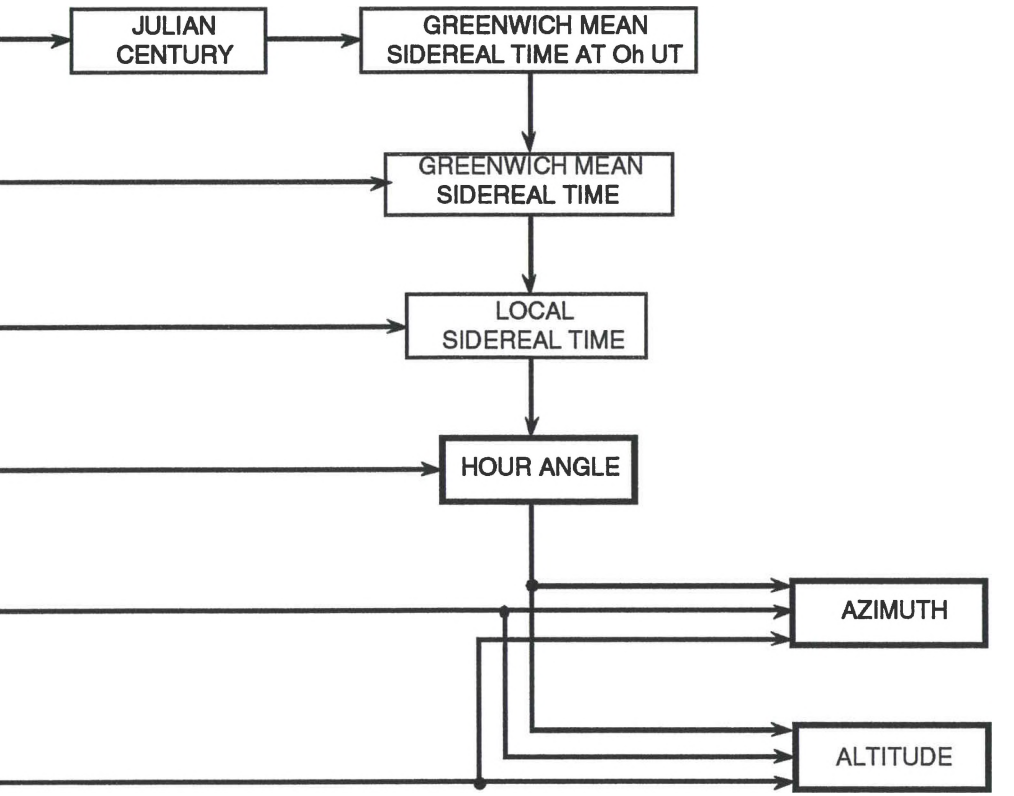


Figure 8

Flow Chart Showing Relationships Between Parameters


OUTPUTS
JULIAN CENTURY
GREENWICH MEAN SIDEREAL TIME AT Oh UT
GREENWICH MEAN SIDEREAL TIME
LOCAL SIDEREAL TIME
HOUR ANGLE
AZIMUTH
ALTITUDE

CHAPTER 4

SYSTEM REQUIREMENTS DEFINITION/PRE-DESIGN ISSUES

Overview

Before designing the control system, it is first necessary to establish the drive system requirements. Chapter 2 took a brief look at basic velocity requirements and provided insight into the relative velocities required of an equatorial telescope system to perform tracking, guiding, setting, and slewing. Obviously, no such system can switch instantaneously among several fixed velocities differing over several orders of magnitude. Further, the alt-az system track velocities are rarely constant in themselves. It is now necessary to adapt these requirements for an altazimuth system and incorporate other key dynamic issues, such as acceleration and torque.

To reiterate, the goal is to design a drive system for an existing altazimuth telescope to perform equatorial tracking for purely visual use. An additional design goal is to reasonably trade off fast slew capability for tracking smoothness. Since the drive system will not be required to support long-exposure astrophotography, the rotating field problem discussed earlier will not be dealt with. Further, since astrophotography is not a requirement, tracking accuracy becomes less stringent, and no guiding speed is required.

Since dynamic torque is dependent on the load moment of inertia (MOI), a detailed moment of inertia analysis will be presented and incorporated. An estimate of bearing friction will also be determined and used in both the static and dynamic torque requirement estimate.

The last section of the chapter will discuss whether open- or closed-loop architecture should be employed. More specifically, the advantages of a closed-loop system will be weighed against the difficulty of adding the feedback components. This should provide a natural lead-in to the hardware and software designs presented in subsequent chapters.

It is believed that this procedure is consistent with the advice of Trueblood and Genet:

> To design a control system, you should first decide what the system will do. Next choose a design approach to making the system work in the desired fashion. . . . At this stage, specific pieces of hardware can be examined to see if which ones, if any, are capable of meeting the system and component performance requirements. Finally, the control system is built and integrated with this telescope. (66: 3).

Motor Type Selection

This topic is really a design issue and would normally be saved for the next chapter. However, certain requirements depend on the type of motor used. Granted, it is certainly premature to select a specific motor, but the general type to be used can be determined based on information pertaining to the application discussed so far

and some limited research into some standard motor types. Three types of motors were considered: large DC servo or "torque" motors, small DC servo motors, and stepping motors.

Large DC servo motors, as the name suggests, are powerful motors that use a permanent magnet stator. Due to the high torque they can generate, they can be mounted in a direct drive configuration and offer the "brute force" approach, whereas smaller motors must be geared down. As a result, torque motors can operate at high speeds for slewing, provide optimum performance, and impressively high acceleration. Further, direct drive coupling prevents gearing errors.

For all these advantages, large DC servos have some significant drawbacks. For one, the brushes cause considerable motor friction, requiring an appreciable control voltage to overcome. Torque motors can be used for digital applications, but the control gain (voltage per unit angle) varies widely according to the size of the current position error. Otherwise, the telescope inertia would cause large overshoots at slew speeds. Even in an analog configuration, the pointing resolution, (called the deadband), is the position error associated with that voltage; in a direct drive configuration, it is unlikely that sufficient resolution can be achieved.

Finally, and perhaps most importantly, torque motors are large, heavy, expensive, and require large amounts of power. They may weigh as much as 190 lb, cost thousands of dollars, and consume several kilowatts of power. Therefore, one can conclude that torque

motors are not practical for telescope control applications. Instead, weight, cost, power, and resolution can each be reduced by perhaps two or three orders of magnitude or more, by simply choosing a different type of motor in a gear-reduced configuration.

Small DC servo motors offer such a candidate solution. Small permanent magnet (PM) motors are lightweight and have a linear speed-torque curve. Moving coil motors are PM motors that use a flat disk rotor. This lightweight armature can generate a high acceleration, but like torque motors, are very costly. Wound field motors, as the name implies, use a coil for the stator as well as the rotor. Speed can be controlled by varying the current in either winding.

Gear reduction makes small DC servos a viable option for telescope control in general. Drive velocity is reduced to the driven axis to accommodate relatively slow tracking rates, while the torque requirement at the drive motor is reduced as well. A large telescope may have a large moment of inertia, but a gear reduction ratio of N reduces the MOI seen at the motor by a factor of N^2. Finally, if analog control is used, the deadband would also be reduced.

Perhaps the main disadvantage of DC motors is that the motor position is not easily computer controlled. Servos are useful for constant velocity control and could be very useful to drive an equatorial in track mode. However, position control becomes an important consideration while slewing. For an alt-az system, tracking involves tight control of a velocity that varies constantly

over a wide range. Such velocity variation can cause slippage and overshoot, so position should be monitored as well. "The performance of the geared servo motor makes it entirely acceptable as a telescope drive, with the only disadvantage being that it is more difficult to interface to most computers than a stepper motor. (66: 36)"

The last type, the stepper motor, is a type of DC synchronous motor which uses multiple poles and field windings (phases). When a particular winding or combination of windings is excited, the motor shaft rotates a fixed angular displacement and comes to rest. By energizing different combinations of windings in a specified sequence, the motor continues to rotate in steps. Stepper driver circuits can be used to convert an incoming pulse train to a series of winding excitations, such that the motor travels one step per pulse. As a result, the stepper motor can be used to achieve very accurate position control.

Figure 9a shows a typical step sequence for a four-phase stepper motor. Essentially, each phase undergoes a series of logic step changes, making the stepper ideally suitable for digital control, and easily integrated with a computer. Furthermore, since the angular size of a step is known, the stepper can be used in an open-loop control configuration in certain applications.

Stepper motors are manufactured from a few steps to several thousand steps per revolution. One method of increasing motor position resolution is to use a half-stepping routine (Figure 9b). In

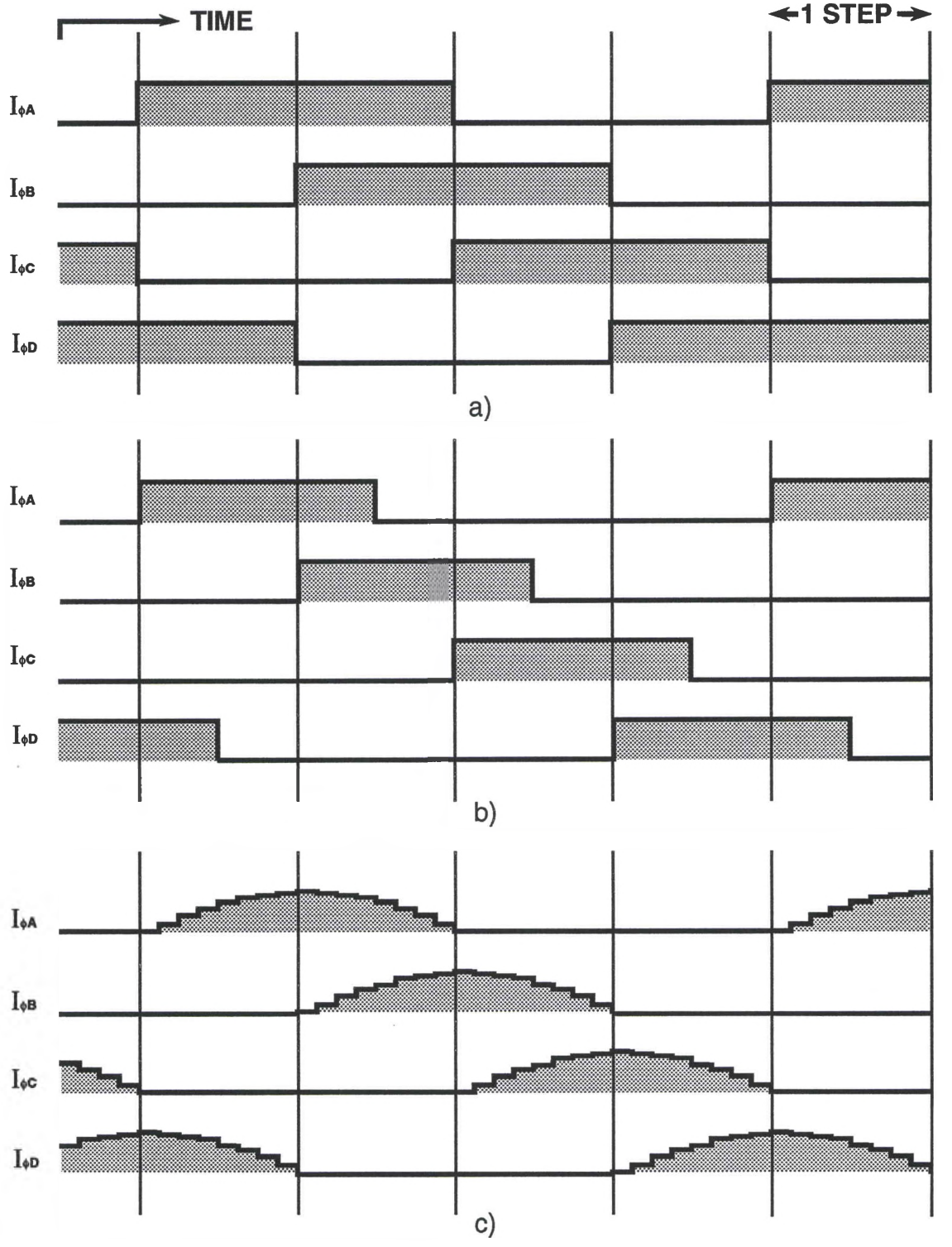


Figure 9

Motor Stepping Routines

this case, one phase changes state at a time rather than a pair of phases simultaneously. This effectively doubles the number of steps per revolution.

For especially fine resolution, the number of steps per revolution can be increased dramatically by a method known as microstepping. (See Figure 9c) Microstepping involves use of a D/A converter to ramp up (or down) the winding voltages, and thereby the currents, in gradually increasing (or decreasing) steps, as opposed to a single discrete change between minimum and maximum current values. The average angular size of a microstep is the nominal step size, divided by the number of voltage increments. The actual size varies cyclically within a nominal step by about ±20%, due to variations in holding torque caused by the phased operation of the motor.

Steppers can operate over a fairly wide range of speeds. There is virtually no lower bound on the rate of stepping, but the upper bound generally resides at about one thousand to a few thousand steps per second, resulting in a maximum speed somewhat lower than that of DC servos. Like DC motors, the steppers face a significant reduction in torque at high speeds. One way to overcome this is to generate an overvoltage spike on the leading edge of the step pulse, thereby increasing torque, and thus performance at high speeds.

The speed-torque curve of a stepper is non-linear. In fact, a typical stepper speed-torque curve is quite irregular, due to mechanical resonances at certain speeds that may cause loss of

position and reduce the available torque. As a result, some steppers may be able to deliver higher torque than DC servos at high speeds, but may suffer dropouts in getting there. Fortunately, microstepping generally reduces or eliminates such resonances.

Although gear reduction greatly relaxes the torque requirement and reduces velocity to a reasonable magnitude, it also introduces several sources of error. Backlash can be reduced by mounting the worm and motor on a spring-loaded plate and greasing the mesh to reduce wear. Without ramping properly, the driven axis can become the driving axis, and the gear teeth are essentially at the mercy of the load moment of inertia.

In addition to causing gear damage, significant changes in speed can cause position errors due to overshooting and/or missed steps. The control system should start the motor slowly and ramp it up to speed gradually, then ramp it back down as it approaches the desired position. This should not be a problem while tracking alone, but ramping will certainly be necessary to accommodate the acceleration during a fast slew.

Many of these challenges must also be met with the small DC servo. Though it has its disadvantages, the stepper's excellent digital controllability over both position and speed and its adaptability through control system modification make it the motor of choice for this application.

Mechanical Description of Telescope Assembly

In order to establish system requirements, it is necessary to learn as much as possible about the mechanical characteristics of the load to see how they will affect the requirements. Figure 10 illustrates the mechanical construction of the telescope without any control hardware. The system is an eight-inch f/6 Dobsonian.

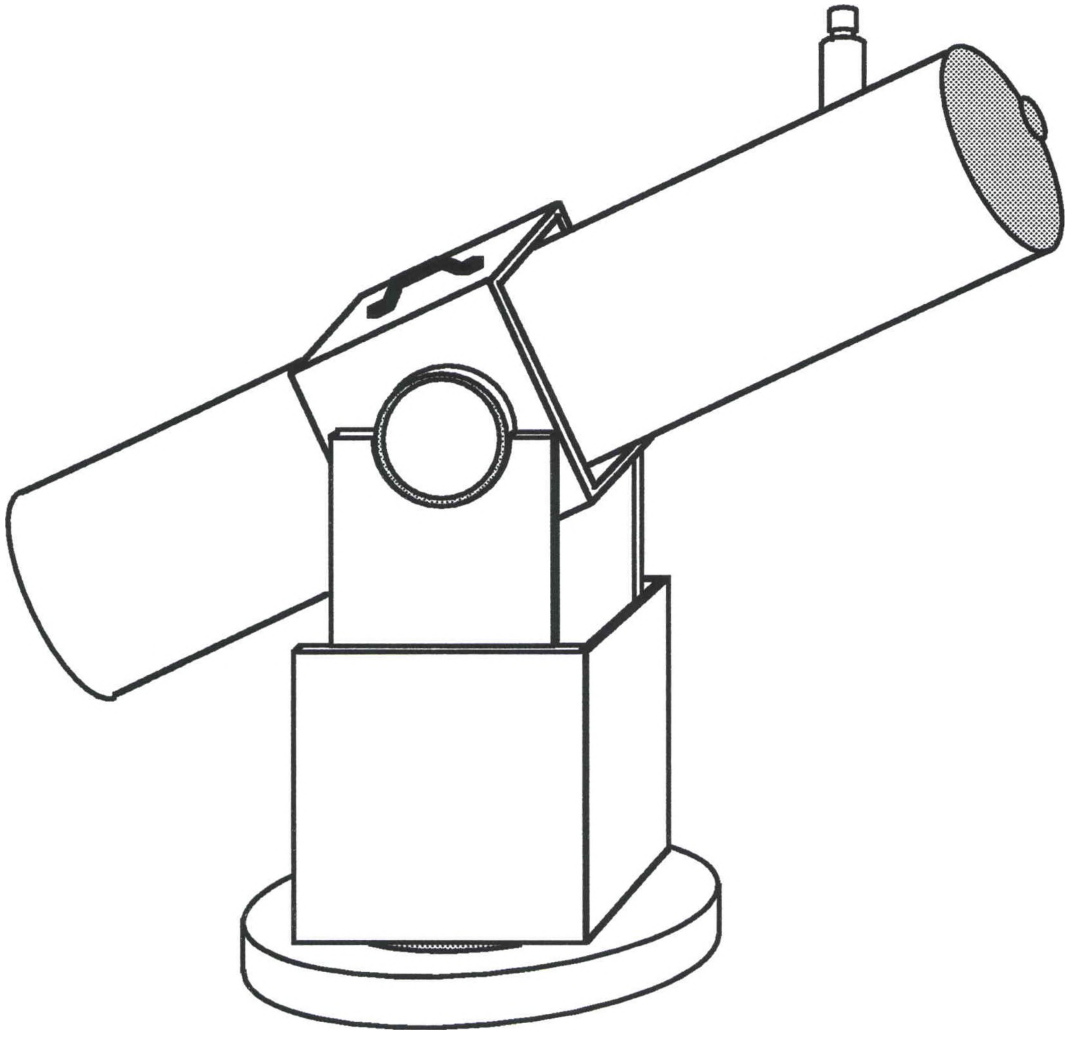


Figure 10

Telescope System to be Retrofitted

The base, rocker box, and tube box are constructed of primarily 3/4" plywood. The polyvinyl chloride (PVC) tube is held in place by the pressure of two sides of the box which compress it into a slightly oval cross section; it is further secured by four machine screws, which also hold the handle in place. The two altitude bearings are each constructed of a plywood disk with a ring of PVC riding in a curved semicircular slot on two Teflon pads. The azimuth bearing is composed of a formica-coated particle board disk, riding on three Teflon pads. Supporting the Pyrex mirror is a mirror cell constructed of two phenolic plates and a sidewall of PVC pipe.

Appendix A provides a complete description of telescope parts when disassembled, including the individually-measured weight of each element. Centers of mass are also listed, as determined by symmetry where appropriate, or otherwise, by experimental balancing.

Torque to Overcome Friction

Both static and dynamic torque requirements are heavily influenced by load frictional characteristics. This is especially true of the standard Dobsonian, since it is not designed to minimize bearing friction. In fact, the Dobsonian's crude Teflon/formica bearing design was not only intended to be simple to build, but also to provide a degree of friction *high enough* to withstand a light bump or wind and maintain its current position.

Experimental bearing friction data was obtained from tests run with an apparatus consisting of series of pulleys and a cable attached to the telescope tube. Tension on the cable was incrementally increased until the tube began to move. As expected, due to the rough bearing construction, friction measurements varied considerably with respect to the starting position in both azimuth and altitude. The resultant static frictional parameters for both axes are summarized in Table 1.

Table 1

Tabulation of Torque to Overcome Friction

Axis	Direction	Moment Arm Dist. (in)	Static Frictional Force (oz)		Equivalent Torque (oz-in)	
			Avg.	High	Avg.	High
Azimuth	CW	28.5	43.1	57.0	1228.	1625.
Azimuth	CCW	28.5	36.1	39.3	1029.	1120.
Altitude	Rising	25.2*	7.3	10.0	184.	252.
Altitude	Setting	18.8*	30.2	31.4	568.	590.

*The moment distance varied; numbers represent the average moment distance.

Measurements for kinetic friction were also attempted, but were not successful. In any case, kinetic friction rarely exceeds static friction for the same surfaces. In fact, with Teflon, data suggests static and kinetic friction coefficients are essentially equal. (5: 642). Due to the wide variation in the static friction measurements, and sizeable sources of error in the measurement apparatus, an

engineering margin of 25% over the highest torque measurement for each axis will be established as the minimum frictional torque requirement. This produces the following numbers:

$$T_{f_A} = 2031 \text{ oz-in} \qquad T_{f_E} = 738 \text{ oz-in}$$

Moment of Inertia

Dynamic torque requirements are determined not only as that necessary to overcome bearing friction, but also to accelerate the load to some known maximum speed during tracking and slewing. In its simplest form (without gearing), the dynamic torque is determined as follows:

$$T = J\alpha + T_f, \text{ where } \alpha > 0 \quad \text{(accelerating)}$$
$$\text{or } \alpha < 0 \quad \text{(decelerating)}$$

Since the load moment of inertia, J_L, is a key element of the dynamic torque, a comprehensive analysis was performed on each major mechanical portion of the telescope to determine its contribution to the overall MOI about each axis of interest. (See Appendix B.) The azimuth moment, J_{L_A}, was determined with the tube pointing horizontally, since the quantity is maximized at that altitude position. The altitude moment, J_{L_E}, remains constant, since it is independent of azimuth position. The contributions of the altitude drive system components (motor and worm gear) have been estimated, as the actual components to be used are yet to be determined. The azimuth worm gear is assumed to be constructed as a toothed strip wound around the azimuth bearing; therefore, it is

understood to be integrated with the rocker box subassembly and incorporated with its MOI contribution. The MOIs for each of the major load elements about each axis are summarized in Table 2. Rotor inertia estimates are to be handled separately, since all other load elements must be translated via the gear ratio, N.

Table 2

Tabulation of Load Moments of Inertia

Element	MOI, J_L, Slug-Ft^2 (oz-in-sec^2)	
	Azimuth Axis	Altitude Axis
Tube	1.567 (301)	1.567 (301)
Upper Plate	0.093 (18)	0.094 (18)
Tube Box	0.058 (11)	0.049 (9)
Secondary Mirror w/ Mount	0.089 (17)	0.089 (17)
Finder w/ Bracket	0.173 (33)	0.159 (31)
Eyepiece w/ Focuser	0.072 (14)	0.075 (14)
Mirror w/ Cell	0.809 (155)	0.809 (155)
Rocker Box	0.221 (42)	N/A
Altitude Gear	0.019 (4)	0.002 (0+)
Altitude Motor	0.046 (9)	N/A*
TOTAL	3.147 (604)	2.844 (546)

*rotor inertia to be handled separately

Slew Pointing Accuracy

Slew pointing accuracy can be achieved in a variety of ways. A common open-loop approach is to initially calibrate the pointing position at system startup, and continuously count steps throughout the observing session in order to maintain position information. However, after several slews, it is likely that the accumulated pointing error will be of such magnitude as to require recalibration. This is mainly due to mount fabrication errors, gearing errors, missed steps, and any other form of drive train anomaly. A more practical approach is a closed-loop architecture to accurately monitor and update the true pointing position throughout the slew.

Slew accuracy is important not only to provide feedback to locate an object, but also to properly predict the tracking profile in order to keep the object fixed in position within the field of view. In order to aim the telescope at an object of known location, and to be able to ensure the object will appear in the 30′ field of view after slewing to that location, pointing accuracy should be maintained on the order of ±12′. Stepping errors should not be a problem for slew pointing as long as plenty of torque is available and acceleration and deceleration are properly controlled to prevent missed steps.

Perhaps the most obvious is the error caused by the worm gear backlash. Special design techniques will be used to keep this error to a minimum, but this should really only be a problem when changing the direction of acceleration, as particularly encountered during slewing. For instance, assume the system is operating in a

trapezoidal velocity profile, with a system azimuth backlash error of ε_B. When acceleration begins from rest, a backlash slip, β, may occur of magnitude $-\varepsilon_B \leq \beta \leq 0$, relative to the starting rest position. The uncertainty lies in the starting position of the worm with respect to adjacent wormwheel teeth. When top speed is reached and acceleration terminated, no change in β will occur, since the load friction is expected to hold the worm against the same side of the wormwheel tooth. However, when deceleration begins, the wormwheel will slip forward by the increment, ε_B, unless the system is still forward-driven to produce a more gradual decelerating profile than a purely frictional ramp. Thus, the new position error is $0 \leq \beta \leq \varepsilon_B$. Assuming this to be the case, no further change in β will occur when the load comes to rest. If motion recommences in the same direction, two full slips of magnitude ε_B will occur, one in each direction. Hence, the error after a complete cycle is still $0 \leq \beta \leq \varepsilon_B$. If motion resumes in the reverse direction, one full slip will occur, such that the new error is $-\varepsilon_B \leq \beta \leq 0$. Therefore, it can be seen that backlash error is non-cumulative, and will never exceed $|\varepsilon_B|$ in either direction from reference position, nor will the algebraic sum of extreme errors in both directions during any given observing session.

There are three basic types of mounting errors in an altazimuth system. (59: 196) Type 1 is leveling error, ε_L, the deviation of the azimuth axis from the vertical, which could be a degree or more. However, this can be compensated for in software by simply adding an altitude leveling correction factor of

$$\Delta E_L = \varepsilon_L \cos (A_L - A)$$

where A_L is the azimuth at the line formed by projecting the azimuth axis onto a horizontal plane. It can be also be shown that the azimuth leveling correction can be determined as

$$\Delta A_L = A_L - A - \tan^{-1}\left(\frac{\cos E \sin (A_L - A)}{\sin E \sin \varepsilon_L + \cos E \cos (A_L - A)}\right), \quad |A_L - A| \leq 90°$$

To preserve quadrature in the previous equation, add 180° to the result if $90° < |A_L - A| \leq 180°$. Although this equation can produce azimuth corrections much larger in angular magnitude than ε_L, itself, the actual pointing error remains as ε_L.

Type 2 is the perpendicularity error, ε_P, which represents the angular deviation of the altitude axis from a perpendicular to the azimuth plane. This is normally small, and can be modelled, if necessary. Trueblood and Genet (66: 113) derive the correction for perpendicularity error as

$$\Delta A_P = \varepsilon_P \sin E$$

$$\Delta E_P = \varepsilon_P \Delta A$$

These equations have been simplified by the assumption that ΔA_P and ε_P are small angles. Note that the second equation will only work with ε_P expressed in radians. Otherwise, the following should be used

$$\Delta E_P = \varepsilon_P \tan \varepsilon_P \sin E$$

The previous expression also illustrates that $\Delta E_P << \varepsilon_P$ for small ε_P. Further, the earlier equation, $\Delta A_P = \varepsilon_P \sin E$, determines that $\Delta A_P \leq \varepsilon_P$, so that the maximum perpendicularity pointing error is ε_P.

The last fabrication error to be considered, Type 3, is the alignment error, which is the deviation of the telescope's optical axis from a perpendicular to the altitude axis along the tube. This error is mitigated more easily if broken into two components, one measured perpendicular to each axis. It can be shown that the correction to the alignment error about the azimuth axis, ε_A, is

$$\Delta A_{AL} = \frac{\varepsilon_A}{\cos E} = \varepsilon_A \sec E$$

As in Type 1, this equation can produce azimuth corrections much larger in angular magnitude than ε_A, itself, but the actual pointing error is still ε_A. Similarly, the correction to the alignment error about the altitude axis, ε_E, is simply the error itself, or

$$\Delta E_{AL} = \varepsilon_E$$

To summarize, all three of the mounting error types can be effectively modelled in software; precision is limited essentially by accuracy of measurement. Taki (59: 196) has developed a BASIC program for coordinate conversion which incorporates models for all three mounting errors. However, care should be taken whenever possible to minimize mounting errors to begin with. Since this is a design for an existing telescope, mechanical adjustments should be made as practical, and modelling should be utilized only as required.

To attempt to model backlash would be very difficult at best. Therefore, it is expected that backlash will be the largest error contributor. If it is assumed that the root-sum-square (RSS) of the combined errors must not exceed 12′, or

$$RSS = \sqrt{\varepsilon_B^2 + \varepsilon_L^2 + \varepsilon_P^2 + \varepsilon_A^2 + \varepsilon_E^2} \leq 12',$$

then the average allowable error for a single contributor is

$$\varepsilon_{AVG} = 5.4'$$

Tracking Resolution/Accuracy

There are three key differences between pointing accuracy and track accuracy issues:

1. angular range over which errors are accumulated,
2. one is strictly a position error while the other involves both position and velocity control, and
3. precision required.

The previous section took into account several sources of error that affect slew pointing accuracy. Such factors are those that cause inconsistencies in position determination before and after slewing over a wide span of the sky (perhaps 90° to 180°), independent of the rate of speed and acceleration in getting there. Slew velocity and acceleration requirements are established in later sections, and may somewhat influence pointing accuracy. However, pointing accuracy is really concerned with the total position error accumulated during a

given slew. Further, these issues are required to maintain accuracy on the order of several arc minutes in order to acquire the image in the subdegree field of view.

Factors affecting tracking accuracy are those that impact the ability to maintain the object in the field of view and at a specific given location within the field of view, but only over that angular distance (typically, a degree or less) covered during a single tracked observation. Unlike pointing error concerns, tracking error issues must be dealt with constantly throughout the track; i.e., both position and velocity must be tightly controlled. Finally, certain elements of tracking error must be maintained with much greater precision than slew pointing considerations. Although the goal is to maintain the object in the 30′ field, an effort should be taken to stabilize the image within the field, which requires accuracy on the order of arc seconds. In short, pointing accuracy must be conserved to the arc *minute* level while traversing large angles, while tracking accuracy must be constantly maintained at the arc *second* level, but over much shorter angular distances.

There are several factors to consider in order to determine the accuracy of tracking in this application. Since mount imperfections are not detrimental for small changes in position, and generally do not cause short-term perturbations in track velocity or acceleration, these can probably be ignored as far as tracking considerations are concerned. The telescope's drive train assembly is the primary source of tracking error. Basically, these errors fall into two categories: gear errors and motor stepping errors.

High-precision gears are relatively expensive, and not justifiable for this particular design effort. Therefore, the required precision will be limited to keep costs down. However, gear indexing errors should not exceed 1%, or 36˝/tooth for 360 teeth, and should be non-cumulative. If this error is significantly larger, yet repeatable, software modelling may be required to maintain track. If not, such an error could really only hinder track accuracy over a long observation or when tracking close to zenith. Backlash can also be a problem for tracking, but only when changing direction of altitude or azimuth.

Several other mechanical factors may also affect tracking ability. As mentioned earlier, a problem with stepping motors is that they may occasionally miss steps. Care should be taken to insure that the required torque is well below the motor torque limits. If not, bearing friction may cause a tracking lag error. This is due to the fact that the motor's rated holding torque is not valid as the motor poles approach alignment toward the end of a full step. Instead, the rotor may come to rest prematurely when the remaining motor torque is no longer great enough to overcome the load torque. The result is a position offset error, which not only causes microstepping to be somewhat non-linear, but it could also lead to missed microsteps. Careful analysis of manufacturer-supplied torque curves during the motor selection process can minimize this concern.

An additional tracking issue to be dealt with is tracking resolution, which must be selected to ensure not only that the object remains in the field, but also to prevent an unstable or fluttering

image. Consequently, arc-second resolution must be maintained. In general, tracking resolution can be thought of as the largest tracking correction tolerable, and if a DC servo motor were used, it would be just that. With a stepping motor, however, the stepping action is not only responsible for the tracking motion itself, but is, in a sense, a continuous correction to a track of zero velocity. In this application, track resolution is essentially the step size, σ, divided by the gear ratio, N. If the "seeing" or optical resolution is on the order of one arc sec (1´´), then any stepping motion of an angular magnitude much greater than that would induce unacceptable quantization errors, rendering the appearance of "vibrating" stars. Table 3 illustrates the resolution achieved using various step sizes and gear ratios (N).

Table 3

Angular Resolution Achieved By Gear Reduction

	Resolution (Arc Sec)					
Steps/Rev	N=90	N=180	N=270	N=360	N=450	N=540
50	288	144	96	72	57.6	48
100	144	72	48	36	28.8	24
200	72	36	24	18	14.4	12
400	36	18	12	9	7.2	6
500	28.8	14.4	9.6	7.2	5.8	4.8
1000	14.4	7.2	4.8	3.6	2.9	2.4
2000	7.2	3.6	2.4	1.8	1.4	1.2

From the table, it is apparent that additional reduction is required, especially since steppers with more than 200 primary steps/revolution are generally not available. (Manufacturer claims of thousands of steps/revolution generally refer to microsteps.) There are two ways to achieve further reduction, mechanically and electronically. The mechanical solution involves adding an additional gearing stage. Given the numerous fabrication problems this brings about, the electronic solution, microstepping, is more favorable. For a 200-step motor and N = 360 teeth, a 4-bit (16 μsteps/step) microstepping routine will produce a resolution of σ_μ = 1.125″, which should suffice. It should also be noted that typical manufacturer's step size tolerances (non-cumulative) are on the order of 3%; this essentially sets the reliable limit on microstep resolution at 33 μsteps/step. Therefore, the D-to-A circuitry to produce finer microstepping could be implemented, but step size tolerances would limit resolution to that achieved with the five most significant bits.

Assuming mechanical concerns can be dealt with, perhaps the single largest factor in the tracking error budget is the accuracy of the tracking control algorithm itself. If steps are missed, the algorithm should update position and tracking data and parameters often enough to avoid significant error accumulation. If the track velocity is based on current position, the position will need to be determined with sufficient accuracy to compute a reliable track rate, and one that does not change drastically with each new position update. Further, it may be necessary to use programmed rate variations that predict the velocity profile while awaiting the next

update. This will be especially of concern when approaching zenith, where the azimuth track acceleration rises significantly, as will be seen in the next section.

In order to prevent drifting outside of the 30′ field of view for one hour, the tracking velocity will have to be maintained to within 12′/hour (0.2′′/sec), or approximately 1.3% of the nominal equatorial track velocity. Track position must also be maintained to arc sec accuracy by choosing the proper gearing and stepping schemes, and choosing the proper motor to minimize load torque offset.

Tracking Velocity

In the previous chapter, the following equations were derived for altitude and azimuth, as a function of latitude, declination and hour angle.

$$A = \tan^{-1}\left(\frac{\sin H}{\cos H \sin \phi - \tan \delta \cos \phi}\right)$$

$$E = \sin^{-1}\left(\cos H \cos \delta \cos \phi + \sin \delta \sin \phi\right)$$

It may also be recalled that the hour angle is

$$H = S - \alpha = G - \theta - \alpha = G_o + kU - \theta - \alpha$$

where all quantities, including time (U), are expressed as an angle in degrees (°). In order to maintain sidereal tracking, the telescope must follow a trajectory which allows it to remain pointed at the exact altitude and azimuth of the object of interest. Right ascension (α), declination (δ), and latitude (ϕ) of a typical celestial object are all

fixed quantities, independent of time. So are longitude (θ) and GMST @ O^h UT (G_0) for a given date. In fact, the only quantity on the right side of the hour angle equation that changes as a function of time is time itself, expressed here as Universal Time (U). Therefore, all other quantities can be represented as constants, a through e, where

$$a = G_0 - \theta - \alpha$$

$$b = \sin \phi$$

$$c = \tan \delta \cos \phi$$

$$d = \cos \delta \cos \phi$$

$$e = \sin \delta \sin \phi.$$

In addition, it is apparent that

$$H = a + kU$$

and the transformation equations become

$$A = \tan^{-1}\left(\frac{\sin H}{b \cos H - c}\right) = \tan^{-1}\left(\frac{\sin (a + kU)}{b \cos (a + kU) - c}\right)$$

$$E = \sin^{-1}\left(d \cos H + e\right) = \sin^{-1}\left(d \cos (a + kU) + e\right)$$

Note that U is expressed here as an angular equivalent of time, measured at the solar rate of four minutes of time per degree. At this point, it may be appropriate to use the term "deg" to denote time measurement in degrees. For angle measurements, the degree, minute, and second symbols (°, ′, ″) will be used whenever possible to avoid confusion. The whole word "degree" will be used interchangeably when no potential for confusion exists.

The tracking velocity equations of motion as a function of time can be obtained by taking derivatives with respect to U. Note that H′(U) = k. The first derivative of A provides the azimuth track velocity

$$v_{t_A} = \frac{dA(U)}{dU} = \frac{k\,(b - c\cos H)}{(b\cos H - c)^2 + \sin^2 H}$$

$$= \frac{k\,(b - c\cos(a + kU))}{(b\cos(a + kU) - c)^2 + \sin^2(a + kU)}$$

where $b \neq c$ ($\phi = \delta$) if $H = a + kU = 0$, the conditions at zenith.

Similarly, the altitude track velocity is

$$v_{t_E} = \frac{dE(U)}{dU} = \frac{-kd\sin H}{\sqrt{1 - (d\cos H + e)^2}}$$

$$= \frac{-kd\sin(a + kU)}{\sqrt{1 - (d\cos(a + kU) + e)^2}}$$

where $d + e \neq 1$ ($\phi = \delta$) if $H = a + kU = 0$, the conditions at zenith.

In both equations, units are expressed in degrees/degree (°/deg). Essentially, these expressions represent the tracking velocities as a multiple of (i.e., normalized to) the standard solar tracking rate. Note that if the k is removed from the numerator (i.e., divide through by k), the velocities are expressed as multiples of the standard sidereal rate. This is a useful way to look at v_{t_A} and v_{t_E}, since it provides an idea of the dynamic range of speeds (magnitude of velocities) required relative to the standard equatorial track rate.

To more fully illustrate how azimuth and altitude track velocities vary as a function of hour angle and declination, Mathematica® software was used to generate three-dimensional plots showing these relationships. Note that the plots (Figures 11 and 12) were created for a fixed latitude of 45°N, and show only the the Northeast celestial quadrant.

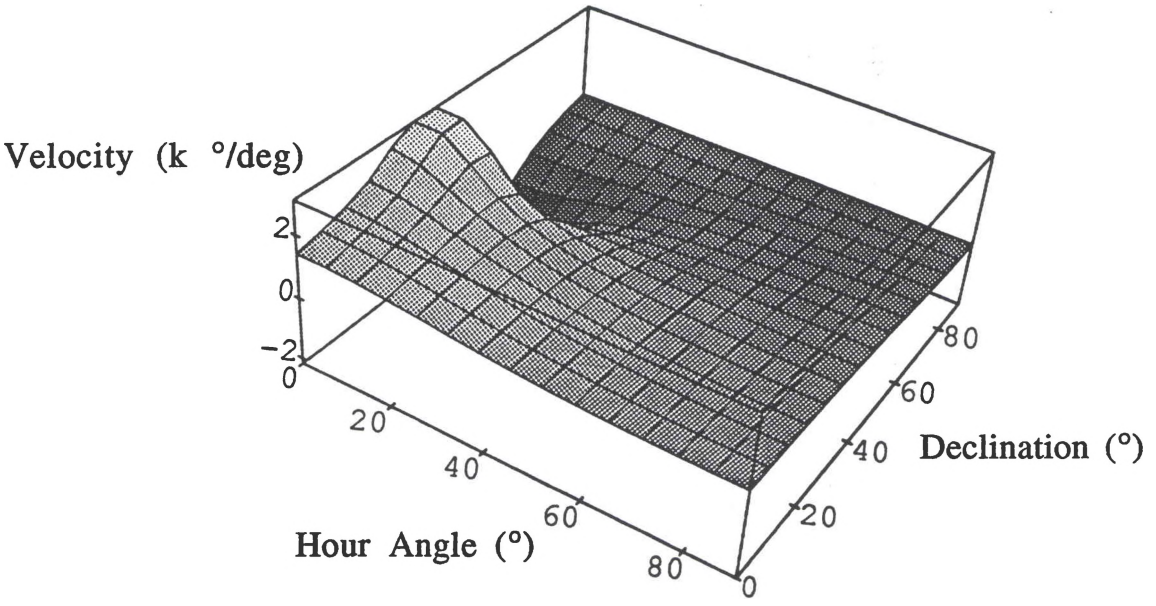


Figure 11

Azimuth Velocity vs Hour Angle and Declination (ϕ = 45°N Lat.)

Figure 11 shows a single discrete point at which the surface plot peak velocity occurs. This point is at zenith, where a discontinuity is formed, as $v_{t_A} \rightarrow \infty$. As a result, it becomes necessary to establish a prohibited region of perhaps five degrees around zenith to keep the azimuth track velocity manageable and maintain accuracy.

Since velocity is a vector quantity, a directional convention must be established that is consistent with the foregoing equations:

v_{t_A} is positive if clockwise (East to South to West, etc.), while v_{t_E} is positive when rising. The maximum positive (clockwise) azimuth velocity occurs at the Southern "edge" of the prohibited zone, whereas the maximum negative velocity (absolute value) occurs at the Northern "edge". These velocities for 40°N latitude, $v_{t_A}(\phi, H, \delta)$ are calculated to be

$$v_{t+A} = v_{t_A}(40°, 0°, 35°) = 9.40\ k\ °/deg$$

$$= 9.40 \times \text{sidereal track rate}$$

$$= \frac{(9.40\ °/deg)(3600\ ''/°)(1.002738)}{240\ sec/deg}$$

$$= 141.4''/sec \quad or \quad 2.4'/sec$$

$$v_{t-A} = v_{t_A}(40°, 0°, 45°) = -8.11\ k\ °/deg$$

$$= -8.11 \times \text{sidereal track rate}$$

$$= \frac{(-8.11\ °/deg)(3600\ ''/°)(1.002738)}{240\ sec/deg}$$

$$= -122.0''/sec \quad or \quad -2.0'/sec$$

Figure 12 shows a surface plot of altitude velocity as a function of hour angle and declination for 45°N latitude. By inspection, there appears to be a maximum (absolute value) v_{t_E} at H = 90°, δ = 0°. Note from the equation that

$$v_{t_E}(\phi, 90°, 0°) = -k \cos \phi, \text{ or}$$

$$v_{t_E}(45°, 90°, 0°) = -k \cos 45° = -0.707\ k\ °/deg$$

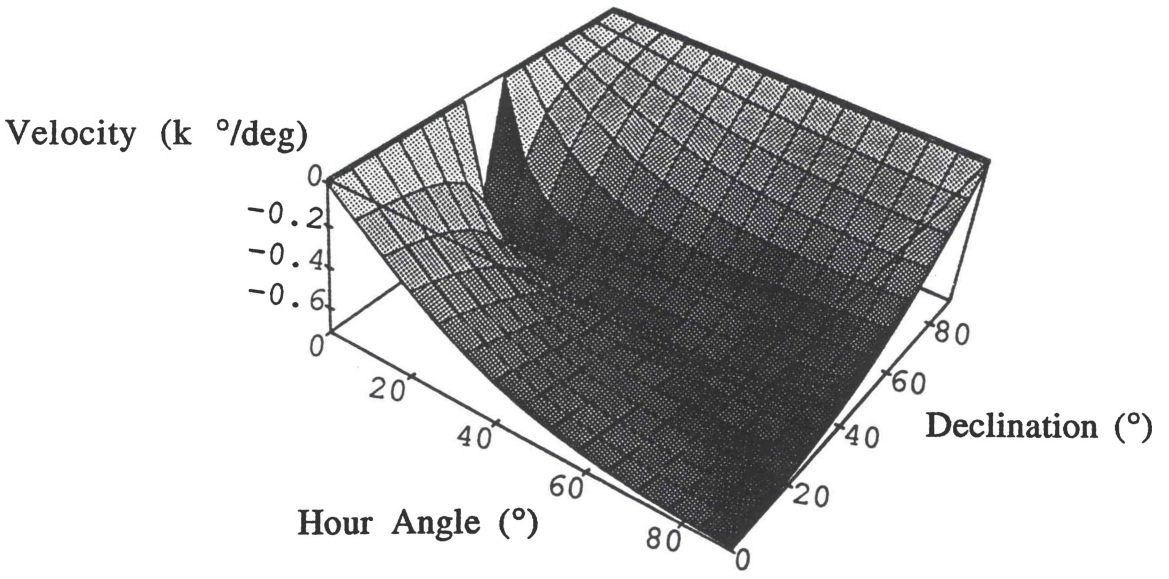


Figure 12

Altitude Velocity vs Hour Angle and Declination (ϕ = 45°N Lat.)

However, this location is on the Western horizon, where tracking would be unnecessary. Detailed analysis indicates data from the zenith region also appears to converge on this value. Again assuming a 5° prohibited zone, the maximum altitude velocities for 40°N latitude will occur at the Eastern (+) and Western (-) edges of this zone. These values are

$$v_{t+E} = v_{t_E}(40°, -5°, 40°) = 0.77 \text{ k °/deg}$$

$$= 0.77 \times \text{sidereal track rate}$$

$$= \frac{(0.77 \text{ °/deg})(3600 \text{ ´´/°})(1.002738)}{240 \text{ sec/deg}}$$

$$= 11.6\text{´´/sec}$$

$$v_{t\text{-}E} = v_{t_E}(40°,\ 5°,\ 40°) = -0.77\ k\ °/deg$$

$$= -0.77 \times \text{sidereal track rate}$$

$$= \frac{(-0.77\ °/deg)(3600\ ''/°)(1.002738)}{240\ sec/deg}$$

$$= -11.6''/sec$$

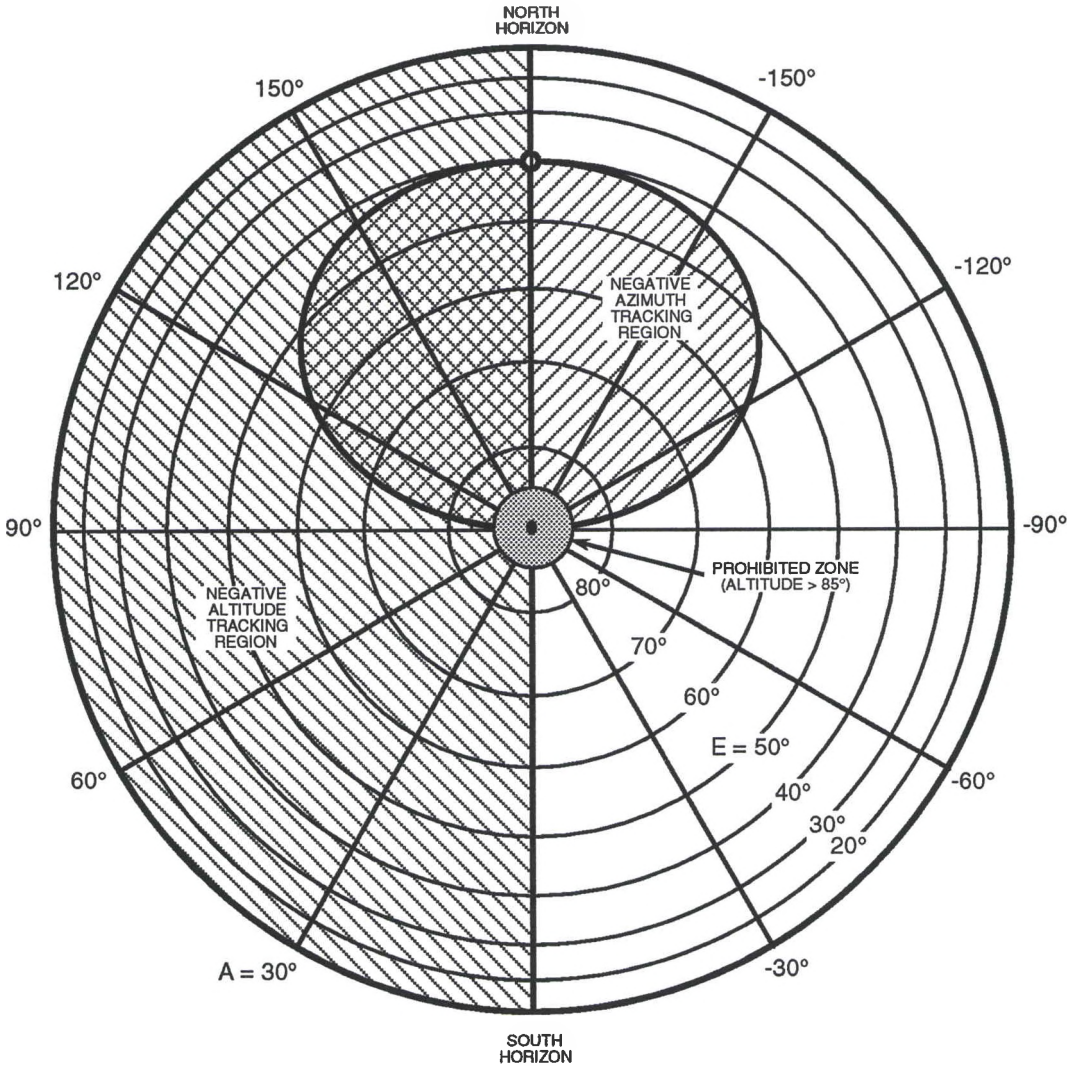


Figure 13

Positive/Negative Track Regions for 40°N Lat. (Alt-Az Coord.)

As already indicated, one significant difference between the equatorial drive and the alt-az is that the latter must perform bidirectionally about both axes. To better illustrate this, Figure 13 shows the regions where v_{t_A} and v_{t_E} are positive and negative, respectively, relative to azimuth and altitude. The vertical line down the center represents $v_{t_E} = 0$, where the altitude track reaches its

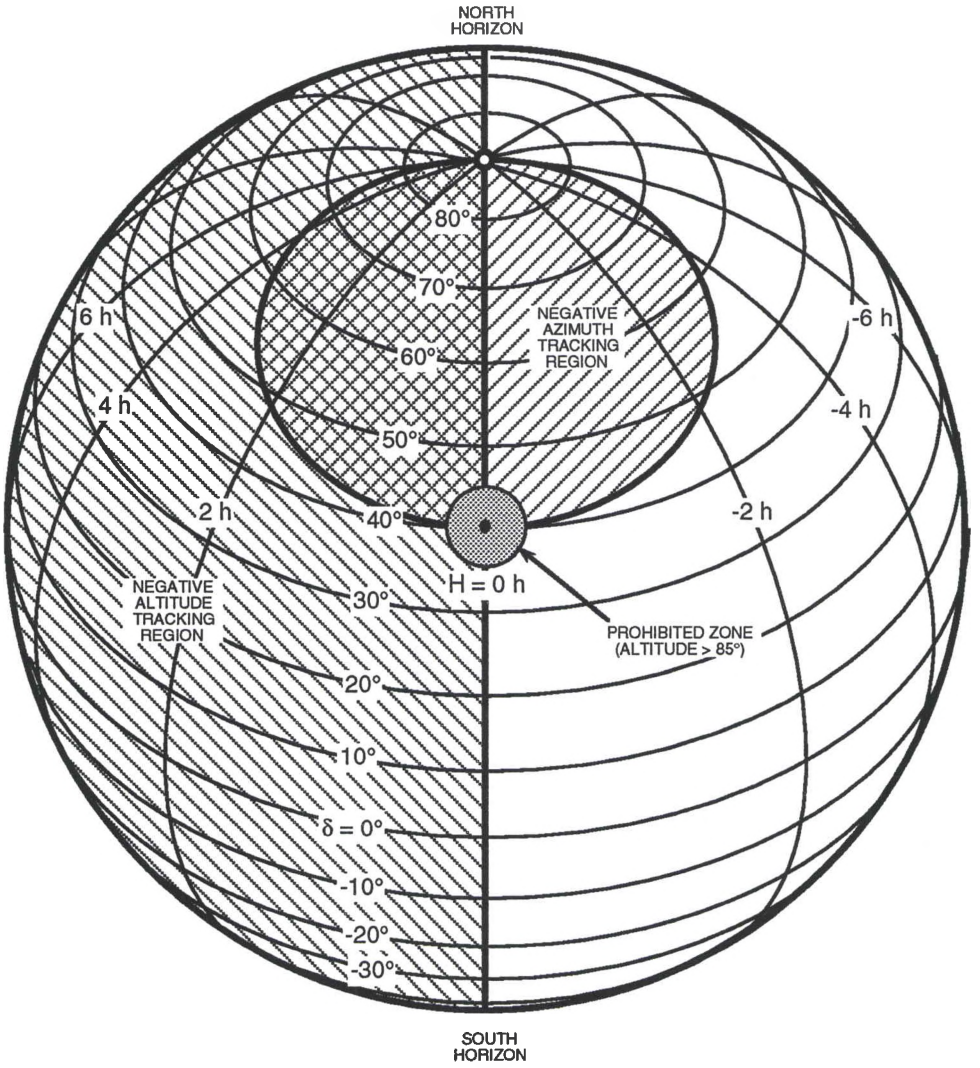


Figure 14

Positive/Negative Track Regions for 40°N Lat. (Equatorial Coord.)

peak and changes direction. Likewise, the rim of the ellipse in the upper half of the plot represents $v_{t_A} = 0$, where the azimuth track changes direction. Figure 14 shows the same information superimposed on an equatorial grid. The declination grids can also be thought of as track curves (i.e., the path of a star of given declination horizon-to-horizon.

Minimum Track Velocity

Since velocity with a stepping motor is accomplished through movement in discrete steps, it may also be necessary to know the minimum speed (greater than zero) necessary to maintain track. This quantity is directly indicative of the longest expected delay between pulses. Unfortunately, calculations indicate that tracking velocities tend to be very close to zero over a broad trajectory. As a result, it may be necessary to place further boundary conditions on the operating position envelope.

In order to determine the minimum tracking velocities, it is first necessary to determine where small relative velocities occur. By superimposing the track curves of Figure 14 onto the altitude-azimuth grid of Figure 13, zero velocities can be expected wherever the track curves shown (or any interpolated curves) coincide in slope with grid lines, shown or interpolated. The locations of interest are those where the slopes are nearly coincident for the longest period of time. The azimuth zero crossings all fall on the edge of the ellipse, but only at zenith is there a sustained coincidence between the track

curves and the radial altitude (elevation) grid. Since a prohibited zone has already been established around zenith, it may be best to consider the altitude tracking velocity, v_{t_E}. Analysis indicates that v_{t_E} becomes infinitesimally small as it approaches the celestial pole ($\delta = 90$) along the meridian, and that it is smaller near the portion of the meridian at H = 180°, vice H = 0° for a given declination, δ. Therefore, very long delays between pulses can be expected here. Placing a limit on the maximum delay means establishing a prohibited zone around the pole, as well as zenith.

If track accuracy is to be guaranteed for $\delta \leq 89°$, the goal is to determine the pulse interval, and thereby, the average v_{t_E} over one step interval for $\delta = 89°$ at 40° latitude near the 180° meridian. The first step is to determine the altitude at this location, then solve for H (time) when the altitude changes by one step (1.125′′).

$$E(40°,\ 180°,\ 89°) = 39.000971°$$

Since the altitude track at H = 180° is rising,

$$E(40°,\ H_{+1},\ 89°) = 39.000971° + (1.125'') \left(\frac{1°}{3600''}\right) = 39.001284°$$

Through additional calculations, the hour angle at this new location, H_{+1}, is determined to be

$$H_{+1} = 178.555° = 1.445° \text{ from the meridian}$$

At the equatorial track rate, a target star advances this angular displacement in 346 sec, or 5 min, 46 sec. Therefore, the minimum altitude track velocity becomes

$v_{t_E}(\text{min}) = 346^{-1}$ steps/sec or 0.002891 steps/sec = 0.003251 ´´/sec

Tracking Acceleration

Taking the second derivative produces the tracking acceleration equations:

$$a_{t_A} = \frac{d^2A(U)}{dU^2} = \frac{k^2\sin(H)\left(c(1-b^2c)\cos^2(H)-2b(b^2-1)\cos(H)+c(c^2-2b^2+1)\right)}{\left[\left(b\cos(H)-c\right)^2+\sin^2(H)\right]^2}$$

$$= \frac{k^2\sin(a+kU)\left(c(1-b^2c)\cos^2(a+kU)-2b(b^2-1)\cos(a+kU)+c(c^2-2b^2+1)\right)}{\left[\left(b\cos(a+kU)-c\right)^2+\sin^2(a+kU)\right]^2}$$

where $b \neq c$ ($\phi = \delta$) if $H = a + kU = 0$, the conditions at zenith.

$$a_{t_E} = \frac{d^2E(U)}{dU^2} = k^2d\left[\frac{d\left(d\cos(H)+e\right)\sin^2(H)+\left[\left(d\cos(H)+e\right)^2+1\right]\cos(H)}{\left[1-\left(d\cos(H)+e\right)^2\right]^{3/2}}\right]$$

$$= k^2d\left[\frac{d\left(d\cos(a+kU)+e\right)\sin^2(a+kU)+\left[\left(d\cos(a+kU)+e\right)^2+1\right]\cos(a+kU)}{\left[1-\left(d\cos(a+kU)+e\right)^2\right]^{3/2}}\right]$$

where $d + e \neq 1$ ($\phi = \delta$) if $H = a + kU = 0$, the conditions at zenith.

Units are expressed in terms of °/deg^2. From the complexity of these equations, the need for surface plots should now be apparent, since setting the numerators equal to zero will not provide a simple max/min solution with multiple unknowns. Figures 15 and 16 show plots of track acceleration for 45°N latitude with respect to H and δ. Again, note that maximums appear at or near zenith.

However, acceleration in units of $k^2\,°/deg^2$ is not especially useful, since it bears no relevance to the equatorial realm, and thus, it should be converted to a unit of angle per unit time-squared.

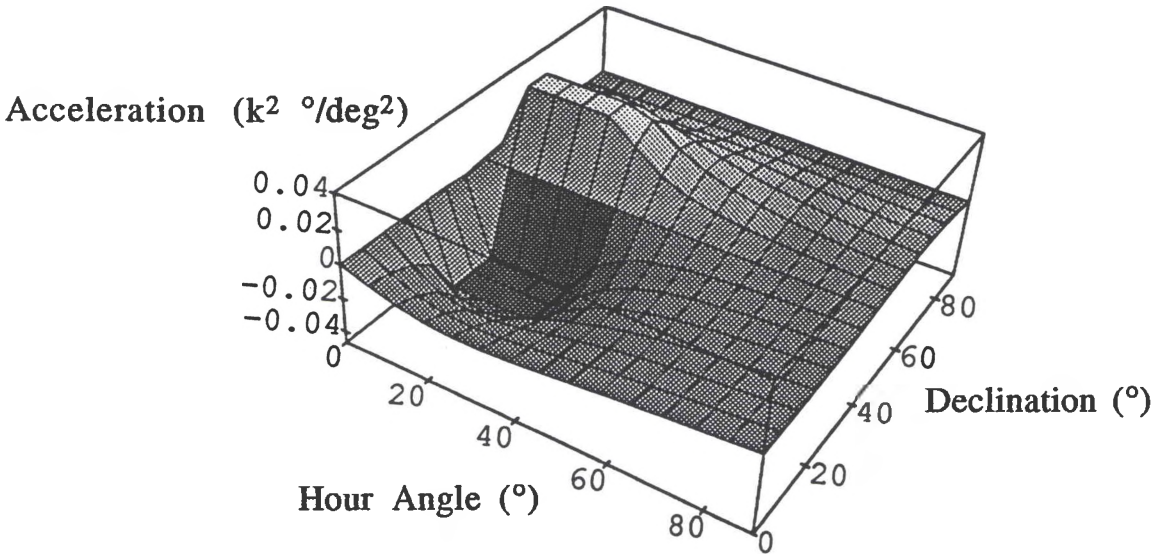


Figure 15

Azimuth Acceleration vs Hour Angle and Declination

(ϕ=45°N Lat.)

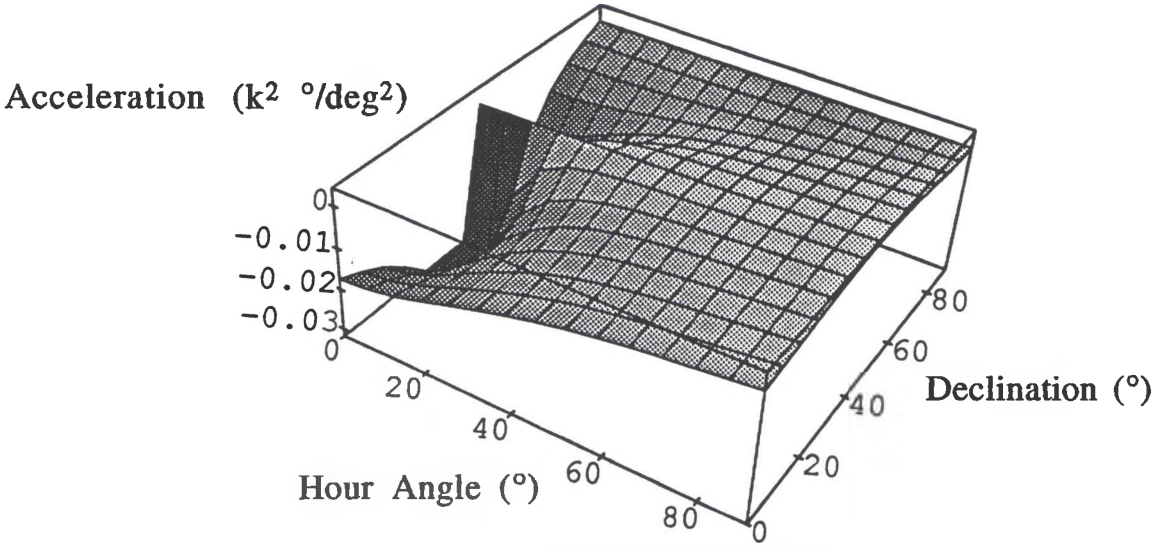


Figure 16

Altitude Acceleration vs Hour Angle and Declination

(ϕ=45°N Lat.)

Once again, considering the 5° radial prohibited zone at zenith, the maximum accelerations for 40°N latitude are

$$\alpha_{t+_A} = \alpha_{t_A}(40°,\ 5°,\ 40°) = 2409\ k^2\ °/deg^2$$

$$= \frac{(2409\ k^2\ °/deg^2)(3600\ ''/°)(1.002738)^2}{(240\ sec/deg)^2}$$

$$= 151.4''/sec^2 \quad or \quad 2.5'/sec^2$$

$$\alpha_{t-_A} = \alpha_{t_A}(40°,\ -5°,\ 40°) = -2409\ k^2\ °/deg^2$$

$$= \frac{(-2409\ k^2\ °/deg^2)(3600\ ''/°)(1.002738)^2}{(240\ sec/deg)^2}$$

$$= -151.4''/sec^2 \quad or \quad -2.5'/sec^2$$

$$\alpha_{t+_E} = \alpha_{t_E}(40°,\ 5°,\ 40°) = 266{,}880\ k^2\ °/deg^2$$

$$= \frac{(266{,}880\ k^2\ °/deg^2)(3600\ ''/°)(1.002738)^2}{(240\ sec/deg)^2}$$

$$= 16{,}771''/sec^2 = \quad or \quad 4.7°/sec^2$$

$$\alpha_{t-_E} = \alpha_{t_E}(40°,\ -5°,\ 40°) = -266{,}880\ k^2\ °/deg^2$$

$$= \frac{(-266{,}880\ k^2\ °/deg^2)(3600\ ''/°)(1.002738)^2}{(240\ sec/deg)^2}$$

$$= -16{,}771''/sec^2 = \quad or \quad -4.7°/sec^2$$

One final note regarding track velocity and acceleration. The software design must not allow an input of $\delta = 90°$ (star position at North celestial pole), since the constant $c = \tan \delta \cos \phi$ is undefined. This represents an additional discontinuity, but not nearly as troublesome as the problem at zenith. However, we have already

established a 1° prohibited zone around the celestial pole to accommodate the minimum track velocity requirement.

Slew Velocity/Acceleration

Another challenge that must be addressed is slew velocity. If the same motor is used for both track and slew, there is a tradeoff between slew velocity and tracking resolution. As discussed earlier, stepper motors have a practical limit on speed, normally one thousand to a few thousand steps/second. Therefore, a 200-step motor might be limited to only 5 revolutions/second (using whole steps). With a 360-tooth worm gear, it would take 36 seconds to slew 180° (5°/sec) in azimuth, plus ramp up/down time. For this application, 5°/sec will be considered to be acceptable. If not, to increase the slew speed, one might need to use a motor with a larger step size or use less gear reduction, either of which sacrifices resolution. Another option is to use the overvoltage spike option described earlier, thereby generating a faster winding current rise time. The result is an increase in the torque-speed curve, and an extension of the motor's maximum speed. Some systems avoid the problem altogether by using a separate high-speed motor (or one with less gear reduction) to handle the slewing requirement. Unfortunately, adding motors and additional gearing schemes with clutches increases cost, creates additional sources of error, and can be difficult to implement mechanically.

Given the above assumptions, however, the nominal running slew velocity (also the maximum required) will be established at

5°/sec for each axis. If necessary, the altitude requirement could be relaxed by a factor of two, since the maximum travel is only 90°, half that of the azimuth axis. However, it is assumed at this point that identical slew motors will be used for each axis.

This specification, in turn, requires a 200-step motor to run at 1000 steps/sec (5 revs/sec) if a 360-tooth worm gear reduction is utilized. If a 4-bit microstepping routine is employed, a μstep rate of 16,000 pulses/sec (pps) would be required. If an 5-bit D/A is used, the pulse repetition rate will need to reach 32,000 pps. Alternately, a half-step routine could be used for slewing, as fine resolution is not required here. However, one stepping routine might be preferred for both track and slew if possible; otherwise, additional complex hardware and software may be required to handle the transition from microstep tracking to half-step slewing, and vice versa.

Ramp up and ramp down times of one second each between standstill and the maximum slew velocity should be reasonable for both axes. Given a linear acceleration/deceleration to produce a trapezoidal slew velocity profile, the acceleration rate would be

$$a_s = a_{s_A} = a_{s_E} = \frac{5°/\text{sec}}{1\ \text{sec}} = 5°/\text{sec}^2$$

Since the altitude track acceleration requirement has previously been established as

$$a_{t_E} = 4.7°/\text{sec}^2,$$

the slew acceleration requirement is an insignificant increase. In any case, the overall acceleration required is 5°/sec^2, which, in turn, converts to 1000 steps/sec^2 or 16,000 μsteps/sec^2 using a 4-bit routine with N=360 gear reduction.

Setting Velocity/Acceleration

As discussed at the end of Chapter 2, a reasonable setting velocity is about 4-5′/sec or 16 to 20 times the standard sidereal track rate. Therefore, 5′/sec will be selected for this requirement. However, this velocity must be measured relative to whatever tracking velocity happens to be at the position of setting, which is quite large when approaching zenith. Setting should be considered as a low-velocity slew, as well as an extension to the upper limit of the track velocity. At the maximum azimuth positive track velocity of 2.4′/sec, the setting velocity would then be 7.4′/sec or -2.6′/sec, depending in which direction the setting movement was commanded. At the maximum azimuth negative track velocity of -2.0′/sec, the setting velocity would then be 3.0′/sec or -7.0′/sec. Altitude setting velocities would fall in the range of 4.8′/sec to 5.2′/sec.

The setting acceleration is really insignificant as a requirement, especially when compared to slewing. Setting involves much smaller overall changes in velocity and much shorter travel distances than slewing. Therefore, the setting acceleration requirement is considered insignificant in comparison with the slew acceleration requirement.

Dynamic Torque and Rotor Inertia

Up to this point, various preliminary design decisions have evolved as an integral part of the requirements development process. Perhaps the most important system requirement is that of the overall dynamic torque necessary to drive each axis during both track and slew. Although the slew torque requirement is likely to be the driver to sizing the motor, it is probably wise to determine requirements for track as well, in case separate drive systems are selected in the design stage.

Early in this chapter, the maximum dynamic torque required was defined as

$$T = J\alpha + T_f, \text{ where } \alpha > 0 \quad \text{(accelerating)}$$
$$\text{or } \alpha < 0 \quad \text{(decelerating)}$$

The first term, $J\alpha$, represents the torques required to accelerate or decelerate a given load inertia, J, in a frictionless environment. T_f is not only the torque required to overcome friction, but will be considered to be that required to be that necessary to maintain a constant angular velocity. However, this expression does not provide the complete answer for the motor torque requirement. Instead, it provides only the torque to be delivered to the load. Several other factors need to be taken into consideration, namely, the gear ratio, N, the gear efficiency, n, and the motor rotor inertia, J_m (to be determined). A complete expression would be

$$T = T_m + T_L = J_m\alpha_m + \left(J_L\alpha_L + T_f\right)\left(\frac{1}{nN}\right)$$

Since $\alpha_m = N\alpha_L$, the required torque becomes

$$T = J_m N\alpha_L + \left(J_L\alpha_L + T_f\right)\left(\frac{1}{nN}\right)$$

Ideally, the motor inertia, J_m, should match the reflected load inertia, which is

$$J_{L_{refl}} = \frac{J_L}{nN^2}$$

At a minimum, J_m should be $J_{L_{refl}}/4$. Therefore, the acceptable range for J_m is

$$\left(0.25 \le J_m \le 1\right)\frac{J_L}{nN^2}$$

Recall that the total load moments of inertia, J_L, were determined to be

$$J_{L_A} = 604 \text{ oz-in-sec}^2 \qquad J_{L_E} = 546 \text{ oz-in-sec}^2$$

Buchsbaum (5: 290) suggests the typical efficiency of a worm gear to be on the order of 50%. Given a gear ratio of N=360, the range of J_{m_A} is

$$\left(0.25 \le J_{m_A} \le 1\right)\frac{604 \text{ oz-in-sec}^2}{(0.5)(360)^2}$$

$$2.33 \times 10^{-3} \text{ oz-in-sec}^2 \le J_{m_A} \le 9.32 \times 10^{-3} \text{ oz-in-sec}^2$$

Similarly, the acceptable range for J_{m_E} is

$$\left(0.25 \le J_{m_E} \le 1\right)\frac{546 \text{ oz-in-sec}^2}{(0.5)(360)^2}$$

or

$$2.11 \times 10^{-3} \text{ oz-in-sec}^2 \leq J_{m_E} \leq 8.43 \times 10^{-3} \text{ oz-in-sec}^2$$

Next, recall that the static torque requirement for each axis (i.e., that necessary to overcome friction), T_f, were determined to be

$$T_{f_A} = 2031 \text{ oz-in} \qquad T_{f_E} = 738 \text{ oz-in}$$

In addition, the slew acceleration requirement was set at

$$\alpha_{s_A} = \alpha_{s_E} = \pm 5.0°/\text{sec}^2,$$

while that for track acceleration was

$$\alpha_{t_A} = \pm 2.5'/\text{sec}^2 \qquad \alpha_{t_E} = \pm 4.7°/\text{sec}^2$$

Given that the required altitude track acceleration is approximately equal to the slew requirement, there would appear to be little value in using separate motors for track and slew. Based on this information, the maximum required azimuth torque becomes

$$T_A = J_{m_A} N \alpha_{L_A} + \left(J_{L_A} \alpha_{L_A} + T_{f_A} \right) \left(\frac{1}{nN} \right)$$

$$T_A = \left(9.32 \times 10^{-3} \text{ oz-in-sec}^2\right)(360)\left(\frac{5.0°/\text{sec}^2}{57.3°/\text{rad}}\right)$$

$$+ \left(\left(604 \text{ oz-in-sec}^2\right)\left(\frac{5.0°/\text{sec}^2}{57.3°/\text{rad}}\right) + 2031 \text{ oz-in} \right)\left(\frac{1}{(0.5)(360)}\right)$$

$$= 11.87 \text{ oz-in}$$

Similarly, the maximum altitude torques required is

$$T_E = J_{m_E} N \alpha_{L_E} + \left(J_{L_E} \alpha_{L_E} + T_{f_E} \right) \left(\frac{1}{nN} \right)$$

$$T_E = \left(8.43 \times 10^{-3} \text{ oz-in-sec}^2 \right) (360) \left(\frac{5.0°/\text{sec}^2}{57.3°/\text{rad}} \right)$$

$$+ \left(\left(546 \text{ oz-in-sec}^2 \right) \left(\frac{5.0°/\text{sec}^2}{57.3°/\text{rad}} \right) + 738 \text{ oz-in} \right) \left(\frac{1}{(0.5)(360)} \right)$$

$$= 4.63 \text{ oz-in}$$

However, this is not a final requirement, as frictional torque offset (to be discussed later) may increase this value.

Slew Deceleration

It is also important to address the deceleration profile. Note that in both cases above, the torque requirement is composed almost completely of T_f, as $T_f >> J\alpha$. This is a significant consideration in determining the the deceleration characteristics of the system. Since the deceleration rates were selected to be of the same magnitude as the acceleration rate requirements, T_f will be much more than sufficient to decelerate the system rapidly. However, it is necessary to maintain the programmed track deceleration rate. This will require some additional forward torque (i.e., applied in the direction of motion) to keep frictional deceleration from occurring too rapidly. Even during slew deceleration, it is necessary to keep the motor advancing in approximate synchronism with the load rotation as it slows. Since the slew pulse rate will be slowed in a linear fashion,

small forward and reverse torques can be expected to act on the load, since frictional characteristics vary, and frictional deceleration will be non-linear.

The average (purely frictional) slew deceleration rate could be calculated by using the average friction readings from Table 1. Note that $T_f < 0$, since frictional torque opposes the direction of motion.

$$\alpha_{LS\text{-}A}\ (\text{CW avg}) = \frac{T_{f_A}\ (\text{CW avg})}{J_{L_A}} = \frac{-1228\ \text{oz-in}}{604\ \text{oz-in-sec}^2}$$

$$= -2.03\ \text{rad/sec}^2 = -116.5°/\text{sec}^2$$

$$\alpha_{LS\text{-}A}\ (\text{CCW avg}) = \frac{T_{f_A}\ (\text{CCW avg})}{J_{L_A}} = \frac{-1029\ \text{oz-in}}{604\ \text{oz-in-sec}^2}$$

$$= -1.70\ \text{rad/sec}^2 = -97.6°/\text{sec}^2$$

$$\alpha_{LS\text{-}E}\ (\text{rise}) = \frac{T_{f_E}\ (\text{Rise})}{J_{L_E}} = \frac{-184\ \text{oz-in}}{546\ \text{oz-in-sec}^2}$$

$$= -0.337\ \text{rad/sec}^2 = -19.3°/\text{sec}^2$$

$$\alpha_{LS\text{-}E}\ (\text{set}) = \frac{T_{f_E}\ (\text{Set})}{J_{L_E}} = \frac{-568\ \text{oz-in}}{546\ \text{oz-in-sec}^2}$$

$$= -1.04\ \text{rad/sec}^2 = -59.6°/\text{sec}^2$$

Frictional Torque Offset Accuracy

As suggested earlier, the motor's maximum holding torque is only valid when the energized stator-rotor pole pairs are separated by an electrical (phase) angle of 90°. In fact, the actual holding torque varies sinusoidally as a function of the pole separation. Thus,

the holding torque is zero when the poles are perfectly aligned. Consequently, a frictional torque position offset error will exist about the point of alignment at an angular distance of

$$\varepsilon_f = \sin^{-1}\left|\frac{T_f}{nNT_{MH}}\right|$$

where T_{MH} is the maximum motor holding torque. This source of error is perhaps better clarified in Figure 17. Specifically, the error

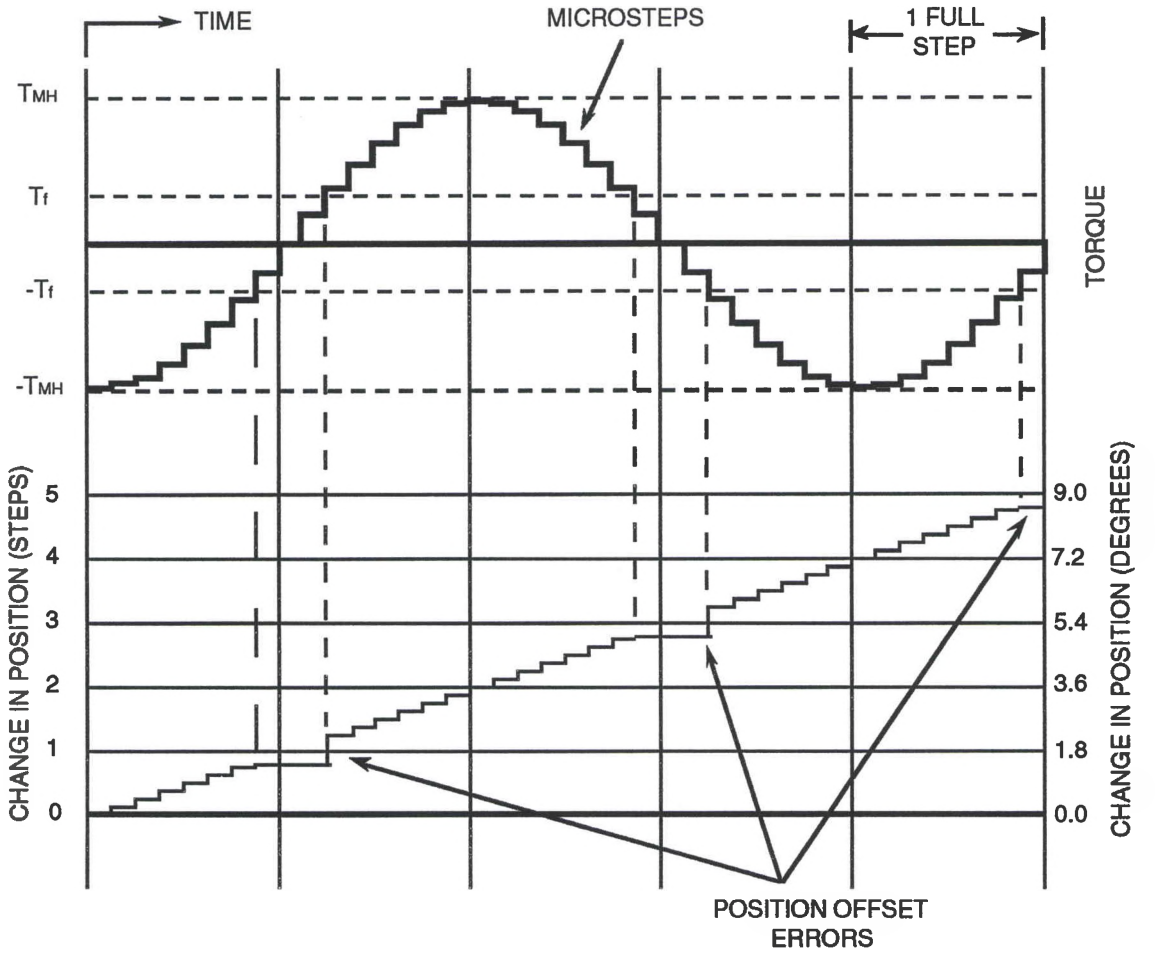


Figure 17

Frictional Torque Offset Error

represents the angular offset from the point of alignment at the position where static torque equilibrium takes place, or

$$T_f = T_{MH} nN \sin \varepsilon_f$$

Since T_f is known, the intent is to select a motor with a holding torque large enough to keep ε_f within acceptable limits. It should be noted that ε_f is a periodic error that only occurs as the poles approach alignment. However, software models cannot compensate for this periodic error; correction can only be achieved by increasing torque or reducing the load friction.

Further, it should be recognized that the total track position accuracy will be on the order of the largest of the microstep resolution, σ_μ, the torque offset error, ε_f, and the motor's specified nominal step accuracy. Thus, ε_f should also be kept down to about one arc second (1´´). Given a 200-step motor (σ=1.8°), and N=360, Table 3 indicates that one full step will move the load an angular displacement of σ/N = 18´´. Since one step represents 90° (electrical degrees) in a typical step motor, then ε_f = 1´´= 1/18 step = 5° (electrical). Given the values for n and T_f used previously,

$$T_{MH_A} = \frac{T_{f_A}}{nN \sin \varepsilon_f} = \frac{2031 \text{ oz-in}}{(0.5)(360) \sin (5°)} = 129.5 \text{ oz-in}$$

$$T_{MH_E} = \frac{T_{f_E}}{nN \sin \varepsilon_f} = \frac{738 \text{ oz-in}}{(0.5)(360) \sin (5°)} = 47.0 \text{ oz-in}$$

So it can be seen that the torque requirement established earlier is insufficient to meet the position accuracy requirement - in fact, by an order of magnitude. However, if these new torque requirements

cannot be met, they could be relaxed by reducing bearing friction. Significant reductions could probably be realized by adding lubricants. Further reductions could be achieved by implementing ball or roller bearings.

Step Generation/Motor Time Constant

Given a maximum slew velocity of 5°/sec, a 200-step motor, with a gear reduction ratio of N=360, would need to turn at 5 rev/sec or 1000 steps/sec (16,000 μsteps/sec with 4-bit μstep routine). In order for the motor to achieve and maintain this speed, it is necessary that the winding currents can rise within 1 msec to that necessary to achieve the required torque. The maximum azimuth tracking torque for microstep accuracy has just been calculated to be 129.5 oz-in. However, microstep accuracy is not necessary for slewing. The required torque for slewing would be that required to achieve maximum slew acceleration, calculated earlier to be 11.87 oz-in. Therefore, only about 9.2% of the peak current is required for slewing, or

$$1 - e^{-t/\tau} = 1 - e^{-(.001 \text{ sec})/\tau} \geq 0.092$$

Accordingly, the motor time constant, τ, defined as L_M/R_M must be

$$\tau \leq 10.4 \text{ msec.}$$

Typical steppers have time constants ranging from about a half to a few msec, so this should not be a problem. Otherwise, it would be necessary to reduce τ by adding a series resistor, R_S, such that

$$\tau = \frac{L_M}{R_M + R_S} \leq 10.4 \text{ msec}$$

This approach is not desired, since it would increase both the power loss and the required driver voltage.

Discussion of Open- Vs. Closed-Loop Feedback Architecture

As mentioned earlier, the stepper motor has several key advantages over other types of motors for digital real-time control applications. One of these is its capability to run open loop. Since the position angle of each step is known, counting pulses from an established reference point can sometimes be sufficient to maintain position control. Velocity information can be derived directly from the pulse repetition frequency, and can be controlled in that way.

Open-loop architecture, however, has its disadvantages. To begin with, pulse generation does not necessarily guarantee step advancement. For instance, peak bearing friction estimates are based on experimental data, and it is possible that greater friction could be encountered. If the drive circuit fails, motor resonance occurs, or something disturbs/obstructs the tube's trajectory, the reliability of open loop control is in jeopardy. Further, many sources of error have been discussed, and there is no way to reconcile these errors without some sort of position feedback. Finally, since trajectory will probably be determined by velocity control to some extent, and since track velocity is position-dependent, open-loop position errors could be further compounded very easily.

Position sensing/feedback could be accomplished by a variety of methods. Ideally, one would prefer what Trueblood and Genet (66: 10) refer to as the non-classical approach. In such a system, the object's position in the eyepiece is actually monitored by a charge coupled device (CCD), photometer, or similar light-detecting device, while corrections are made until the object reaches the desired position in the field of view, normally the center. Unfortunately, these sensors are expensive and difficult to implement.

The more popular classical approach involves sensing the angular position of the driven shaft with respect to the driving assembly. Normally this would refer to measurement of rotation at the axis bearing, but it could also refer to the rotation of "the final worm, which is one step removed from the axis. . . (66: 9)." This is a much simpler method to implement, but errors between the final axis of rotation and the optical axis are not monitored. This part of the system is subject to tube or mount flexure and other errors, which can be generally modeled.

While some equatorial telescope control systems have been designed to run open loop, an altazimuth design may be difficult to implement without some form of feedback. This is largely because absolute position needs to be constantly monitored in order to predict and command the correct trajectory. However, it should also be noted that feedback can add significant cost to the system, especially if a high degree of accuracy is required. Optical shaft encoders cost from about $100 to many thousands of dollars. If the

system is designed to run closed loop, obviously maximum effort should be made to use encoders at the low end of this cost range.

Given some form of feedback is required, the question remains as to at what stage(s) the error should be sensed. If one sensor per axis is sufficient, the goal would be to select the sensor location best suited for the application and providing the most efficient feedback for the cost. The error can be measured at the actual axis of rotation (i.e., azimuth, altitude) or at the respective worm axis. Another option would be to implement feedback at both locations, but this should be recognized as more costly, as four position sensors are required instead of two.

Several factors should be considered in making this determination. Primarily, each method should be evaluated for its drive error detection capability, absolute position accuracy, and mechanical implementation considerations. Drive error detection involves the sensing of missed steps, motor slips, drive circuit failure, etc. Absolute position measurement, as mentioned earlier, is necessary to predict the proper tracking trajectory, and can also correct for mounting errors (if not modeled) and gear backlash.

An axis-mounted position sensor is most effective for absolute position measurement since it does so directly, largely with a single calibration per axis. While it can be done, direct mechanical coupling to the axis is not always easily accomplished. However, the most significant problem with the axis-mounted sensor is that it lacks the sensitivity to detect drive errors over a short term, if at all. Thus, if

any intolerable drive errors are expected, a standalone axis-mounted sensor will not provide the required feedback.

On the other hand, a worm-mounted position sensor can detect drive errors quite readily. Stepper motors sometimes come with position sensors (encoders) already attached to the shaft, and even if not, mechanical interface is still not a major issue. In addition, worm-mounted sensors could also be used to measure absolute position in an open-loop manner. Specifically, this would involve incrementally counting motor rotations since power-up, a calibration-intensive process. A single sensor at this location could suffice, but a second axis-mounted sensor would be preferred for this measurement if affordable.

Another closed-loop question deals with the feedback resolution. Since microstep sizes vary, it is best that the sensing resolution be somewhat less than the microstep resolution, thereby averaging the motion of multiple microsteps. Parker Hannifin (13) suggests at least four control steps for every sensor step. Based on this recommendation, a system consisting of a 200-step motor with 4-bit microstepping (i.e., 3200 μsteps/rev) should have no more than 800 worm-mounted sensor steps per revolution.

Axis-mounted sensor resolution is really not subject to this limitation, since it is geared down considerably from the worm shaft. A reasonable axis sensor resolution for absolute positioning would be of an order necessary to acquire and maintain the object within the

field of view. This translates to the concept of an angular position "cell" of less than 15′, or half the field of view.

Summary

This chapter has established the major system requirements necessary to design a control system for the intended application. In so doing, many of the preliminary design issues have been addressed as well. These issues will provide the foundation for the detailed design to follow. Most of the requirements established in this chapter are summarized in Table 4 below. Angular units have been normalized in arc sec (′′) for comparison. The asterisks (*) represent calculated system characteristics which are not technically requirements.

Table 4

System Requirements

	AZIMUTH	ALTITUDE
Gear Index Errors (%)	≤ 1	≤ 1
Track Resolution (˝)	≈1	≈1
Step Accuracy (˝)	≈1	≈1
Slew Pointing Error (˝)	Total ≤ 720 (12′)	Total ≤ 720 (12′)
Max Track Velocity (˝/sec)	-122.0 to +141.4	-11.6 to +11.6
Min Track Velocity (˝/sec)	0.003251	0.003251
Relative Set Velocity (˝/sec)	300 (5′/sec)	300 (5′/sec)
Slew Velocity (˝/sec)	-18000 to +18000	-18000 to +18000
Slew Pulse Rep Rate (pps)	16,000	16,000
Motor Time Constant (msec)	≤10.4	≤10.4
Track Acceleration (˝/sec^2)	-151.4 to +151.4	-16771 to +16771
Slew Acceleration (˝/sec^2)	-18000 to +18000	-18000 to +18000
Slew Deceleration (˝/sec^2)	-18000 to +18000	-18000 to +18000
Expected Purely Frictional Deceleration (˝/sec^2)*	CW: -419400 CCW: -531360	RISE: -69480 SET: -214560
Rotor Inertia (oz-in-sec^2)	2.33 to 9.32×10^{-3}	2.11 to 8.43×10^{-3}
Static Load Torque (oz-in)*	2031	738
Dynamic Torque (oz-in)	11.87	4.63
Dynamic Torque for 1˝ Position Offset (oz-in)	129.5	47.0

CHAPTER 5

CONTROL SYSTEM HARDWARE DESIGN

Basis and Top Level Approach

Information in previous chapters have laid the groundwork for detailed design of the altazimuth tracking system. In the process of formulating the requirements in Chapter 4, several hardware design assumptions/decisions were made. These include the following:

1. Stepper motors were chosen as the primary control actuators for the system, each with 200 steps per revolution. This decision was made, based on the high resolution required and the limited availability of motors with smaller step sizes.

2. To limit cost, only one motor per axis would be used for slew and track. However, if the resolution requirement is tightened or maximum track/slew speeds are raised, separate track and slew motors may be necessary.

3. Given one motor per axis to handle high-resolution tracking and relatively fast slew and track speeds, some type of microstepping routine would be employed.

Conversion circuitry will likely involve a counter-D/A converter combination with 4-bit accuracy.

4. The drive should use the same microstepping circuit for slew, if possible. If not, a separate circuit would need to be developed to switch over to a full- or (preferably) half-stepping method if the required pulse repetition rate becomes a limiting factor.

5. Tracking will be largely velocity-controlled (generated), but corrections should be made based on position monitoring.

6. Single-loop feedback, at a minimum, is required, sensed at the worm and motor shaft. However, double-loop feedback is preferred, with the second sensor located at the driven axis.

The remainder of the hardware design can now be addressed. The final design is intended to satisfy the requirements in Chapter 4, but a few less significant requirements are not specifically addressed.

The design procedure begins by defining an appropriate system architecture and top level design. From there, the design is broken down and discussed in more detail, one subsystem at a time, beginning with the mechanical interface between the motor and the load axis. Next, the motor is selected, and each of the drive circuit elements is designed in sequence, such that the system develops

progressively from the load back to the controller. Once this has been accomplished, the feedback architecture is readdressed and a design configuration selected.

The Standard Digital Motion Control System

Figure 18 represents a simplified or generic system block diagram for a typical stepping motor motion control application. Power supplies for each unit are not shown. Solid lines are indicative of electronic interfaces, while dotted lines denote the mechanical interfaces.

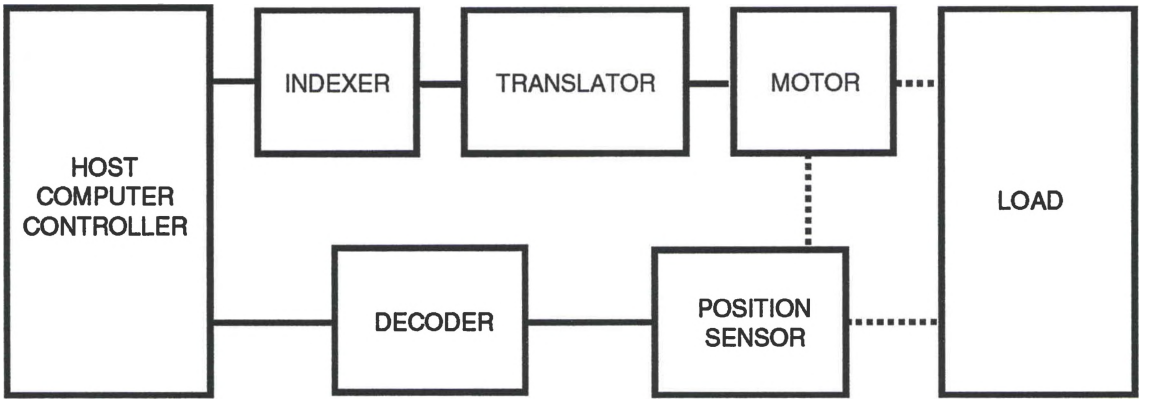


Figure 18

Generic Block Diagram

The function of the host computer controller is to generate position or velocity information in the form of digital commands. All motion control profiles are usually generated by the controller. The indexer converts these commands into a pulse train where each pulse is intended to trigger a one-step advance. The indexer must also supply a direction bit or transfer one from the host computer.

The translator is then used to convert the pulses into a combination of voltage levels to be applied to each phase winding of the stepper motor. This generally involves some logic circuitry to coordinate phase energizing relationships and a power transistor amplifier to produce the high current required to drive the motor. If microstepping is used, the translator becomes much more complex, since it must generate an analog staircase function which, ideally, would be superimposed on a sine or cosine function. The translator is sometimes referred to as the "drive" or "amplifier", although these terms are not as specific.

The motor in Figure 18 refers only to the motor itself, while the load refers to the driven axis and all pertinent mechanical characteristics associated with it (i.e., moment of inertia, frictional torques, etc.) Position sensors may be connected to either the motor, the driven axis, or both, depending on the level(s) of feedback desired or required. Finally, a decoder is needed to convert the sensor information into a format that can be used by the host computer.

Overall Design/System Architecture

As a first step to designing the control hardware, a tailored system block diagram was created for this particular application, as illustrated in Figure 19. The circuit for control of each axis is identical, except for the number of feedback bits processed.

The feedforward configuration is composed primarily of four elements, including the motor. The routing network permits the necessary communication with several I/O devices from a single 8-bit port. Although pulses could be generated by the PC controller, external implementation of this function would save considerable CPU time. Therefore, the controller is used to generate an 8-bit word and a 2-bit range selector that can be converted by the pulse generator into a pulse stream with an overall frequency range of seven orders of magnitude. The microstep converter counts pulses, provides synchronization, and creates the relative voltages necessary for microstep generation, while the amplifier is left to generate the required output winding currents necessary to drive the motor.

Two feedback loops are shown. The inner loop functions as a fine position feedback sensed at the motor axis to permit monitoring of the stepping action as necessary to ensure proper tracking. The outer loop is used to sense the position of the driven axis for independent pointing location confirmation and input for velocity calculation. The feedback portions of each loop are composed of a position sensor to produce an electronic signal representation of the given angular position and direction of rotation, and a set of decoders to generate a digital interpretation for the PC controller. The only difference in these circuits is that the azimuth axis loop processes 11 bits, the altitude axis nine, and each motor shaft loop eight.

The PC controller processes user inputs, displays user outputs, generate velocity/direction commands and ramping routines, monitor position/direction inputs, and services interrupts. It also

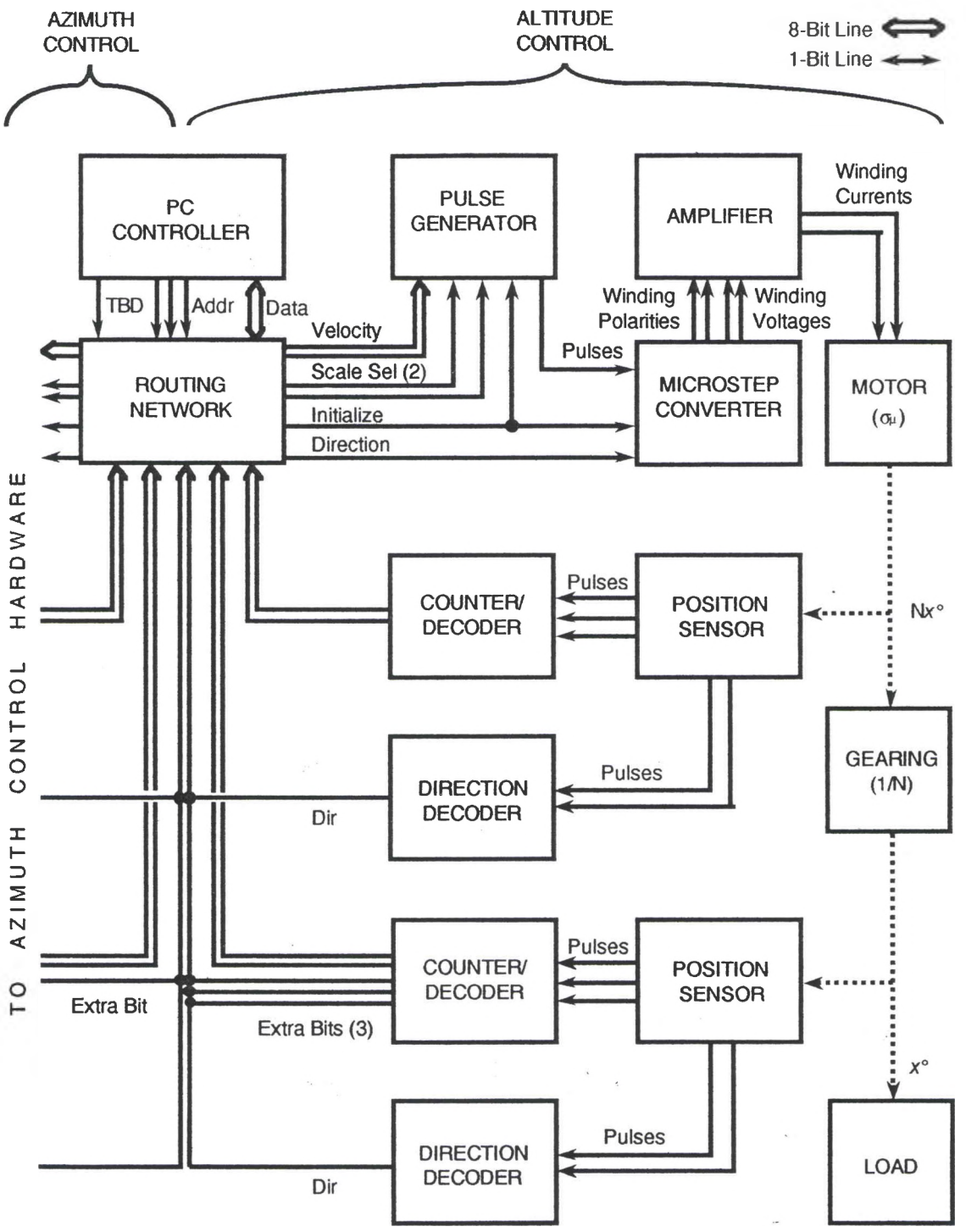


Figure 19

Application-Specific Block Diagram

stores and accesses position, direction, and velocity data, as well as look-up tables. Finally, the PC performs all calculations, including sidereal time, coordinate conversions, position error determination and correction, and the required tracking velocity, based on position.

To limit power consumption in the logic circuitry, Linear and CMOS technologies will be incorporated to the maximum extent possible. In general, National Semiconductor's CD4000 series and other chips will be used.

Mechanical Drive Interfaces

The mechanical drive interface for each axis is shown in Figure 20. Both consist of a single-thread worm that meshes with a 360-tooth worm gear. The inexpensive worm gear (wheel) is fabricated by cutting 360 concave teeth into a 45-inch strip of material (eight threads per inch) in the shape of worm and bending this strip around the circumference of a disk of appropriate size. The thickness of the strip and the disk diameter are chosen such that the total diameter at half the thread depth is 14.324 inches.

To reduce backlash, the motor/worm/position sensor assemblies could be mounted on spring-loaded (not shown) brackets. However, a better option is to mount the worm wheel with a rotary spring to keep it biased in one direction, thereby keeping the worm in constant contact with the same side of each tooth. Gears should be greased at all times to ensure smoothness. Position sensors are in place to monitor angular rotation of both axes and both motor shafts.

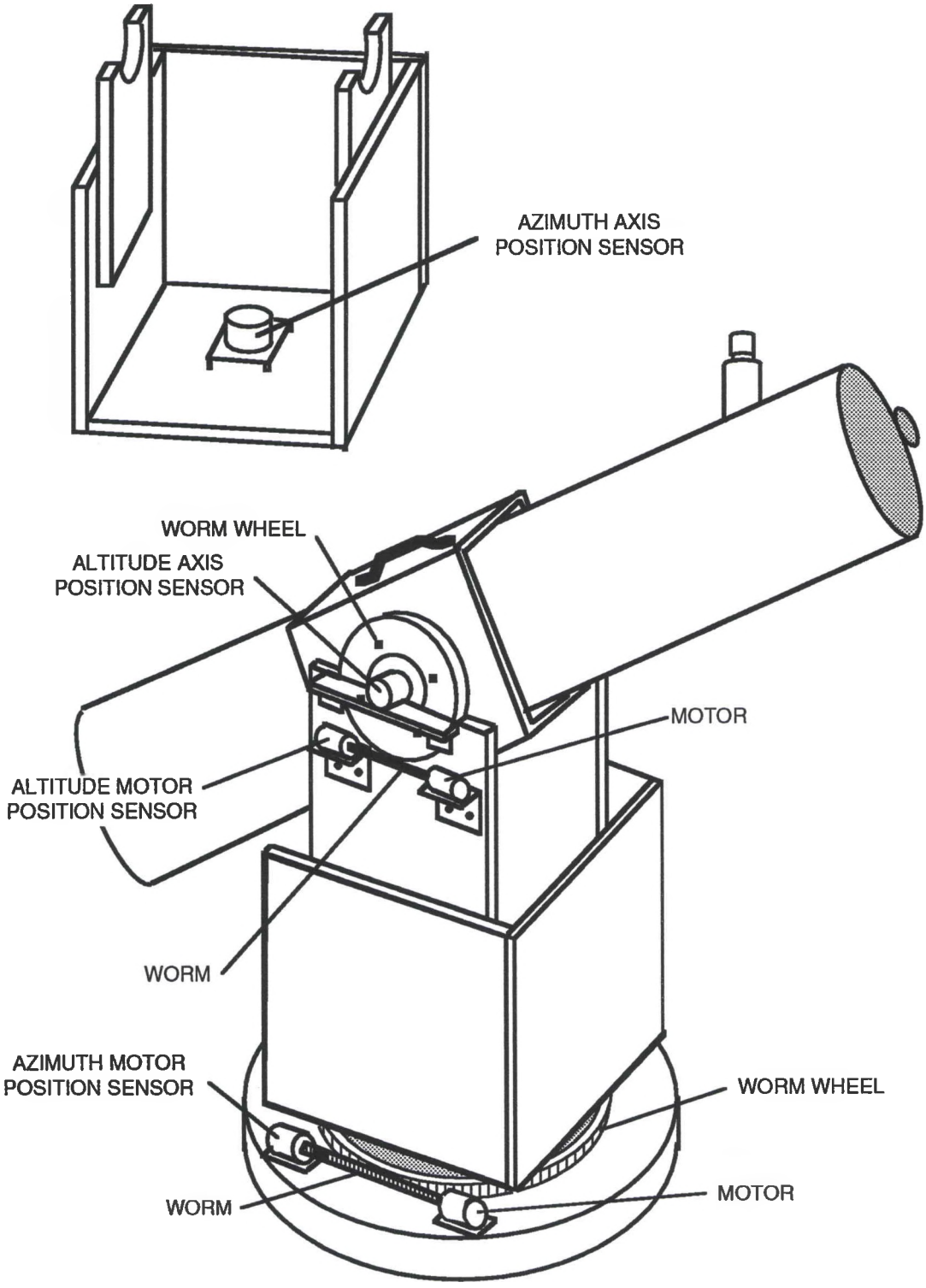


Figure 20

Mechanical Drive Interface

The azimuth axis must be free to rotate continuously through a full 360° and recirculate. The axis position sensor is located on the inside base of the rocker box as shown. Its shaft is attached to the pivot bolt or rod which is, in turn, affixed to the base plate. The position sensor housing is then free to rotate with rocker box to measure azimuth position. The azimuth worm wheel (drive gear) is essentially "sandwiched" between the rocker box and the base plate. To accommodate this, a new azimuth bearing will need to be installed below the worm wheel. The existing bearing could be used, but would need to be detached from the base of the rocker box. No matter what type of bearing is used, it is essential that lubricants or other means are implemented to reduce friction as much as possible.

The altitude worm wheel is attached to the side bearing disk at the outside of the fork. The altitude position sensor is bracket-mounted to the fork as shown, with its shaft connected to the center of the worm wheel. It could also be mounted on the other side bearing if the hardware gets too crowded. The altitude axis is only required to operate over a 90° range; hence, mechanical stops should be placed just below the horizon and at or near zenith to limit movement to this range during calibration. However, since tracking is prohibited within 5° of zenith and 1° of the celestial pole, the software will be required to prevent such encroachment during normal operation anyhow. Nonetheless, the software should also prevent the azimuth axis from dropping below the horizon and running up against the mechanical stop.

Drive Amplifier Type

Since stepping motor selection is partially dependent on the type of drive amplifier used, a discussion of amplifier techniques is necessary at this point. Since a motor's output torque is directly proportional to current in the windings, current, rather than voltage, is the parameter to be controlled. The voltage is selected solely to control current. The function of the drive amplifier is to provide the necessary current amplification to energize the stepper windings to advance the motor in synchronization with the properly phased input signals. There are two main classes of stepper drive amplifier technologies: unipolar and bipolar.

The unipolar drive (Figure 21), as its name suggests, allows current flow in only one direction through any particular motor terminal. This is a simple, yet widely used drive configuration with four-phase operation. The drive can only be used with a motor with bifilar windings, a set of two nearly identical windings for each phase, wound together and connected at one end, forming a

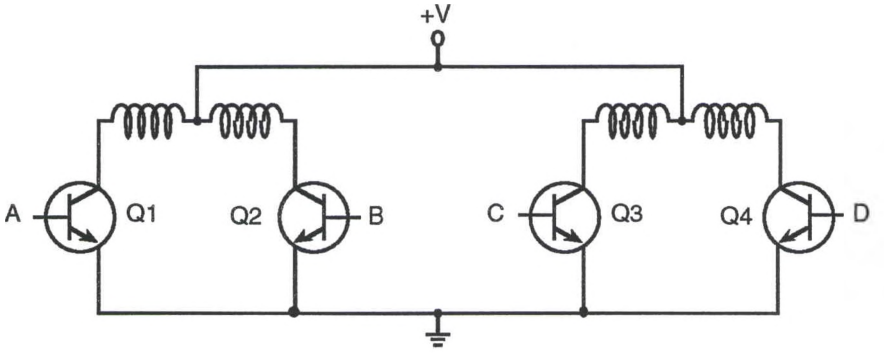


Figure 21

Unipolar Drive Amplifier

centertap. By applying current through one half of the winding and then through the other, the stator field is reversed as is necessary for normal stepping operation. The unipolar works well for slow step rates, but lacks efficiency, since only one or each pair of windings are energized at any one time.

The bipolar drive in Figure 22 offers the fairly simple method of alternately switching the full phase winding between the terminals of two power supplies of opposite polarity. By using both halves of a winding simultaneously, this two-phased approach will produce a 40% increase in torque over the unipolar method, for the same power dissipation. So this bipolar drive solves the problem of winding utilization, but it also causes others. For one thing, two power supplies are required. Secondly, only one power supply is

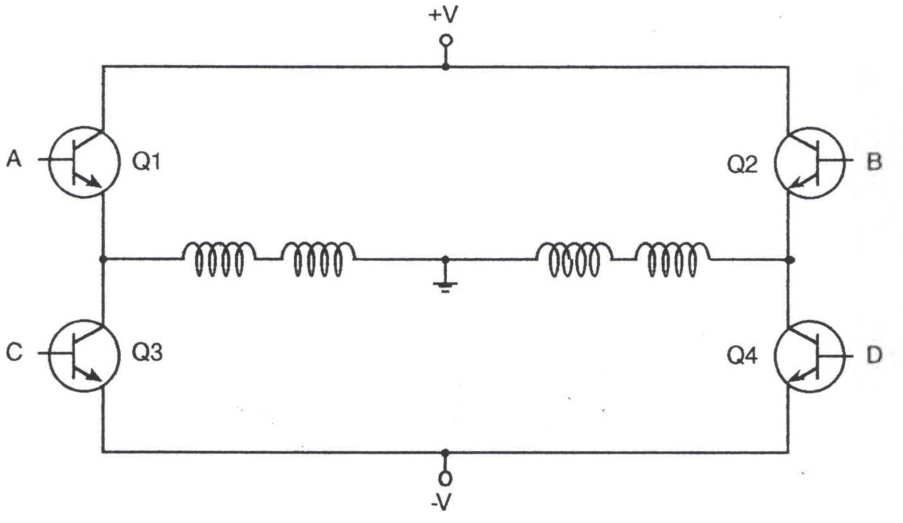


Figure 22

Bipolar Drive Amplifier

providing current at any one time. Finally, the transistors used must be rated at twice the normal operating voltage.

Fortunately, a third possibility exists. The bipolar bridge (Figure 23) a special variant of the bipolar drive, offers a highly efficient operation with a single power supply. Also a two-phased approach, the bridge essentially reverses the power supply leads to the motor winding by alternately turning on the diagonally-configured pairs (i.e., Q1 and Q4, then Q2 and Q3) of transistors. This drive offers all the advantages of the original bipolar drive without the drawbacks. The only disadvantage to the bridge is that eight transistors are required. Some designs, called chopper drives, also use a small current sense resistor placed between the Q3-Q4 emitter connection and ground to drive a comparator that temporarily shuts off the supply voltage to maintain constant current.

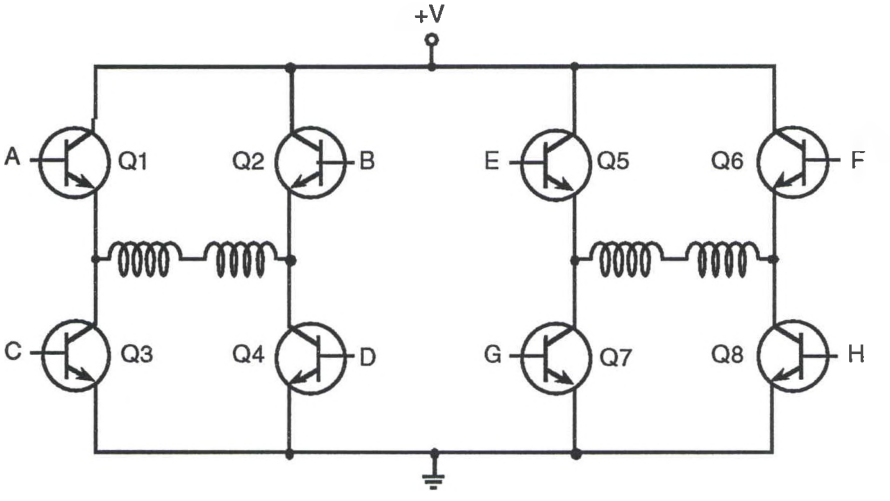


Figure 23

Bipolar Bridge Drive Amplifier

Based on the above considerations, the bipolar bridge circuit is selected for this application. For maximum efficiency, the bridge will run two-phase-on operation. However, one other question remains: should the motor bifilar windings be connected in series or parallel? Series connection produces higher torque at low speeds while parallel connection provides best performance for high-speeds. Sufficient torque is essential to ensure proper microstepping, which governs smoothness of tracking. Also, since tracking is primarily achieved at low speed, a series connection of windings is used. For series bipolar applications, motor current ratings should be normally divided by $\sqrt{2}$ since they are generally given for unipolar connection. Finally, with bipolar operation, timing is important to ensure that no other pair of transistors are on simultaneously at any time.

Motor Selection

Due to the similarities of the application and requirements for both azimuth and altitude, and given the presumed cost savings, as well as the advantage of having matching circuits, two identical motors are preferred. Recall from Chapter 4 that the following requirements apply to the stepping motors to be used.

Steps/Revolution:	200 (1.8° step size)
Holding Torque :	129.5 oz-in (min)
Time Constant:	10.4 msec (max)
Rotor Inertia:	2.33-8.43 $\times 10^{-3}$ oz-in-sec^2

The permanent magnet/hybrid stepping motors in Table 5 were considered for this application. They were selected from catalogues of various manufacturers, based largely on the specifications above. This initial selection of candidate motors are all

Table 5

Specifications of Candidate Stepping Motors

Vendor	Model No.	Holding Torque (oz-in)	Phase Current (A)	Volts/ Phase (V)	Time Constant (msec)	Inertia (10^{-3} oz in-sec^2)
EAD	LA23BCK-10	150	1.27	6.1	4.4	3.31
EAD	LA23BCK-11	150	1.09	7.6	4.5	3.31
EAD	LA23BCK-47	150	0.88	8.5	4.8	3.31
EAD	LA23DGK-2	210	2.00	4.8	4.8	4.97
EAD	LA23DGK-23	210	1.25	8.5	4.9	4.97
Oriental	PH268-21	125*	1.10	7.6	1.7	2.85
SDP	2232190A8	114*	1.4	5.9	2.6	3.26
Superior	M062-FC06	150	2.1	4.8	2.5	4.55
TRW 423A	1102-A301	120*	1.1	7.6	2.5	3.26
TRW 423A	1102-B301	120*	1.1	7.6	2.5	3.26
TRW 423A	1102-A302	120*	1.3	5.9	2.6	3.26
TRW 423A	1102-B302	120*	1.3	5.9	2.6	3.26
TRW 423A	1103-A401	139	2.1	4.8	2.5	4.56
TRW 423A	1103-B401	139	2.1	4.8	2.5	4.56

Note: EAD = Eastern Air Devices, SDP = Stock Drive Products

(10, 12, 40, 56, 67)

1.8° Size 23 (2.3" diameter), with rotor inertia and time constant within the acceptable range. Since the high torque requirement is based on microstep accuracy only, this requirement could be relaxed slightly, if necessary. In cases where torque values fell slightly outside the specified range (marked with an asterisk), the motors were still included. However, the list was limited to those with phase voltage requirements in the 4-12 volt range under bipolar series conditions. Motors employing special technologies (i.e., those with magnetic disk rotors) were eliminated, due to cost considerations. In those cases where the manufacturers listed voltage and current parameters for unipolar connection, the equivalent series bipolar values were derived and listed in Table 5.

Recall from Chapter 4 that the rotor inertia should be at least one-fourth of the load inertia and, ideally, should match. The highest inertias listed in Table 5 are approximately half the load inertias of

$$J_{L_A} = 9.32 \times 10^{-3} \text{ oz-in-sec}^2 \quad J_{L_E} = 8.43 \times 10^{-3} \text{ oz-in-sec}^2$$

Based on this consideration, the list can be easily narrowed to five with the highest inertias. Of those, the two EAD motors have much higher time constants than the others and, therefore, may be eliminated. This leaves the Superior and the last two TRW motors, which are all nearly identical by their characteristics. Given all other factors essentially equal, the Superior motor is rated for slightly higher torque than the TRW motors, which makes it the motor of choice.

Due to the extra torque margin, the maximum operating current can be reduced to 1.9 A, which should still produce 136 oz-in of torque.

Drive Amplifier Design

Now that the drive type and the motor have been selected, the drive amplifier design can be completed. The circuit with motor is shown in Figure 24. Motor phases are wired such that a positive quadrature advance results in clockwise rotation. Preliminary bench testing with the motor may be required to confirm this.

Due to their high impedance, excellent performance, ease of use, thermal stability, and compatibility with CMOS, N-channel power MOSFETs (enhancement type) were substituted for the power switching transistors (Q1-Q8) shown in Figure 23. Two more MOSFETs (Q9, Q10) were also added as current amplifiers to feed the bridge network. Shunting diodes (standard 1N4001 or equivalent), have been added for transient suppression. The 47 Ω, 1/2 W resistors are used to limit current through the diodes.

Since Q1-Q8 are only used for switching, selection is not critical. The only requirements are that they be rated at least 3 A continuous operation and have as low an ON-resistance, R_{DS}, as possible, preferably on the order of 0.1 Ω. The reason for this is that each motor winding pair, R_m, connected in series, is only 2.3 Ω, and additional voltage drops will tend to limit the accuracy of microstep

generator to produce the right winding currents. Based on its very low ON-resistance (0.16 Ω), the IRF100 was selected for Q1-Q8 (55: 143).

The polarity inputs are synchronized with the sine wave generation counter in the microstep generator as discussed in that section. The 4049 inverters at these inputs are used to turn on the correct pair of MOSFET switches. Since microstepping phase currents vary sinusoidally, polarity changes should occur when the drain current, I_D, is approximately zero. If the input voltages, V_1 and V_2, are unipolar, resembling an unfiltered full-wave-rectified stepped

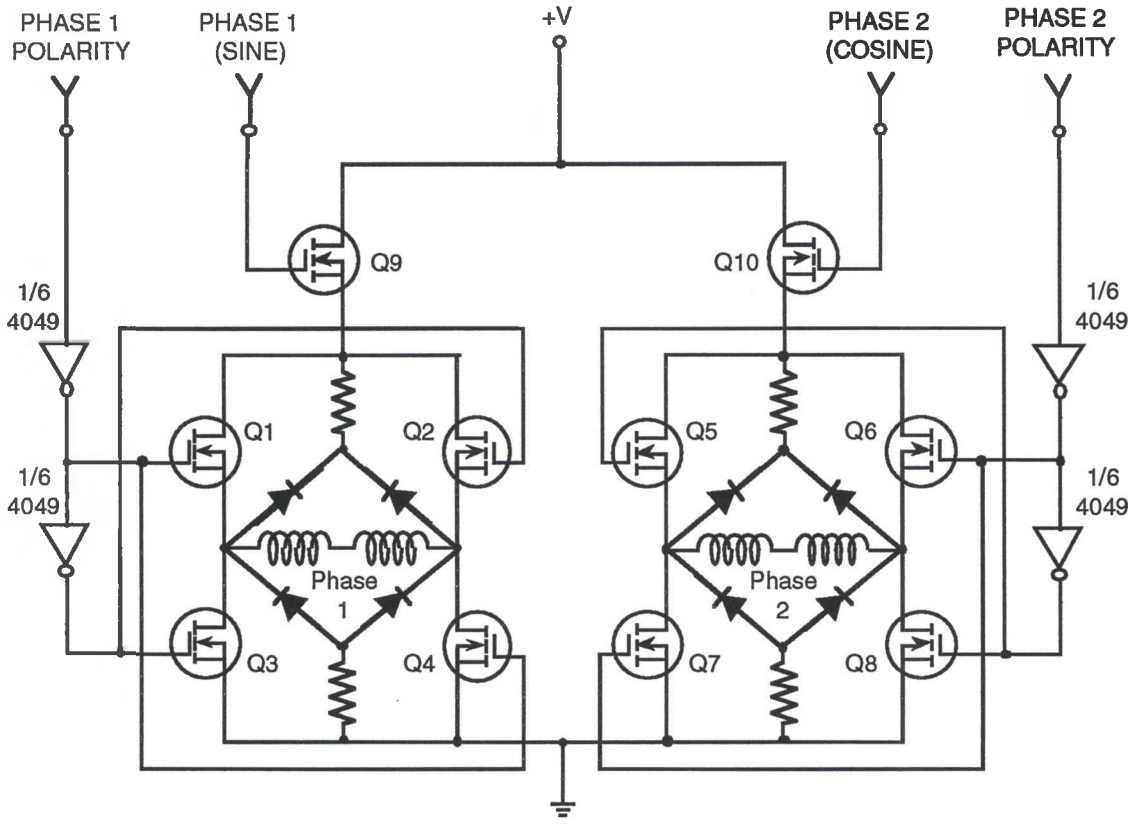


Figure 24

Bipolar Drive Amplifier Design

sine wave, then the bridge polarity switch can be used to correct them. Since this occurs at $I_D \approx 0$, the 20-nsec simultaneous ON-condition of all transistors (inverter propagation time) should not be a problem.

Gate voltages for Q1-Q8 should be supplied high enough to ensure that both are operating in their ohmic regions (i.e., where the gate-to-source voltage, V_{GS}, has minimal impact on I_D) over the entire range of I_D. Consider the steady state equivalent circuit for one-phase-on, shown in Figure 25. For an IRF100, V_{GS} should be maintained on the order of 10 V or more. At maximum current (1.9 A), the necessary voltages supplied to Q1 and Q3 should then be

$$V_{GG}\,(Q1) = V_{GS} + I_D\,(R_m + R_{DS}) = 10 + 1.9\,(2.3 + 0.16)$$

$$= 14.67 \text{ V}$$

$$V_{GG}\,(Q3) = 10 \text{ V}$$

Therefore, the minimum gate input voltage for Q1-Q8 is 15 V.

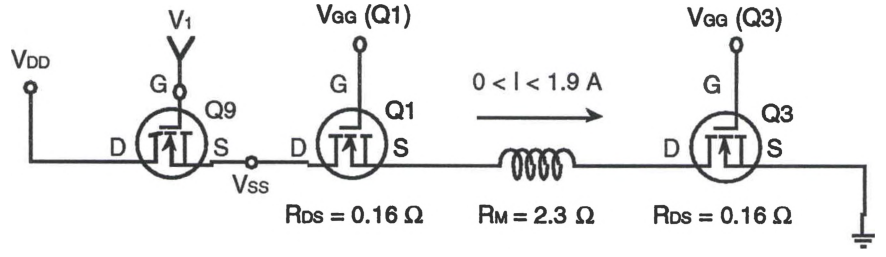
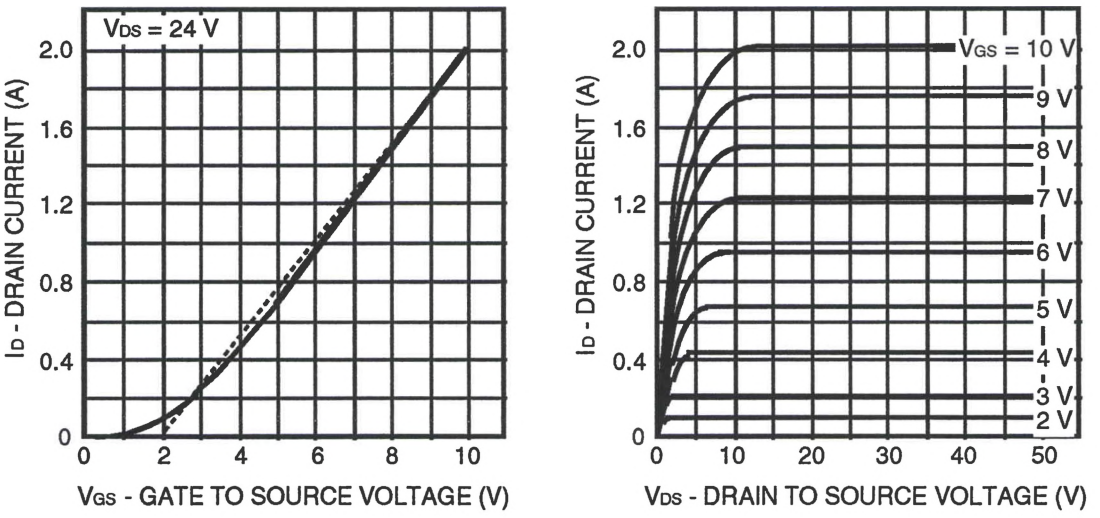


Figure 25

Single Phase ON Equivalent Circuit

The pseudo-sine and pseudo-cosine inputs are waveforms that are actually synthesized in the microstep.generator to be rough approximations of stepped full-wave-rectified sine and cosine patterns. Unlike Q1-Q8, MOSFET transistors Q9 and Q10 should be carefully chosen, since they must control winding currents to 4-bit accuracy for a given voltage input. Therefore, in addition to meeting the maximum current ratings, the transfer characteristics of these two MOSFETs must be highly linear over the 0 - 1.9 A range of operation. Further, it is necessary that the threshold voltage, V_T, be as sharp and well-defined as possible, and preferably small. In other words, it is necessary that linearity extend as close to zero as possible. The 2N6656 N-Channel Power MOSFET reasonably fits this description. The MOSFET's transfer and output characteristics are shown in Figure 26, as provided by Semtech Corp., the device manufacturer.



(55: 157-158)

Figure 26

Transfer and Output Characteristics of 2N6656

To determine the operating ranges of V_1 and V_2, again consider the single-phase equivalent circuit in Figure 25. The voltage at the source of Q9 is

$$V_{SS}\,(Q9) = I_D\,(R_m + 2\,R_{DS}) = I_D\,\big(2.3 + 2\,(0.16)\big) = 2.62\,I_D$$

Since Q9 is an amplifier, it must be operating in its constant current or "active" region, where I_D is affected by V_{GS} independent of V_{DS}. From Figure 26, it appears that this relationship can be modeled by a line as shown, represented by the following formula

$$V_{GS}\,(Q9) = V_T + (\text{Slope})\,I_D = 2 + 4\,I_D$$

so that

$$V_1 = V_{SS}\,(Q9) + V_{GS}\,(Q9) = 2.62\,I_D + 2 + 4\,I_D = 2 + 6.62\,I_D$$

Therefore, the input voltage at the gate of Q9, V_1, must be varied over the range of

$$2\ V \le V_1 \le 14.58\ V \qquad \text{for } 0 \le I_D \le 1.9\ A$$

Additionally, to maintain operation in the constant current region, it can be seen from the output characteristic in Figure 26 that V_{DS} must be at least 13 V. Therefore, the high-current supply voltage, V_{DD}, must be

$$V_{DD}\,(Q9) = V_{SS}\,(Q9) + V_{DS}\,(Q9) = (2.62\ \Omega)(1.9\ A) + 13\ V \approx 18\ V$$

Further, the same voltages are required for Q10, but the gate voltage, V_2, must be 90° out of phase with respect to V_1.

Microstep Generation

The function of the microstep generator stage is to create the sine and cosine magnitude and polarity signals necessary to control the drive amplifier developed in the previous section. The magnitude portion of both waveforms must be incremented with each step pulse input by 1/16 of the output voltage range, consistent with the 4-bit microstepping routine suggested in Chapter 4. The output is produced with an 8-bit D/A converter. Binary counters are used to count pulses and provide timing for polarity and magnitude slope reversals. By far, most complex and difficult to implement is the sinusoid generator, which is required to convert a linear input into a digital sinusoidal output to be fed to the D/A converter, in order that a proportional voltage output waveform can be generated.

Since there are 16 microsteps per step and each full step corresponds to 90° electrical, pulse counts to 64 are necessary to complete one full cycle. Therefore, the four least significant bits (LSBs) of the counter stage can be used to create a 16-step binary ramp, while the next two bits will designate quadrature, as 00 = first quadrant, 01 = second quadrant, etc. As long as quadrature is maintained, motor direction can be reversed by simply changing the up/down input to the counters. From these six bits, all the necessary information for microstep generation can be derived. Figure 27 describes the evolution of waveforms throughout the process. Bit assignments are as follows: X = Q6, Y = Q5, A = Q4, B = Q3, C = Q2, D = Q1, where Q1 is the LSB.

From the figure, it can be seen that the polarities can be easily derived from X and Y, as

$$P_{SIN} = \overline{X} \qquad P_{COS} = \overline{X \oplus Y}$$

Figure 27 also shows an intermediate step, whereby a triangular waveform is generated to correspond with the sign of the desired magnitude slope for each trigonometric function. The sign of this slope is also readily derived from Y, as

$$MAG'_{SIN} = \overline{Y} \qquad MAG'_{COS} = Y$$

The positive ramps are simply represented by the count, ABCD, while the negative slopes can be created by subtracting ABCD from 16. This can be achieved by producing a two's complement and adding one. For the sine function, this value is used when Y is high. For the cosine function, it is used when Y is low. Thus, Y should be inverted between the sine and cosine generation circuits.

The four-bit word is then converted with a customized circuit to produce an 8-bit sinusoid magnitude. However, there is still another concern: there are actually 17 values that must be generated, not 16. The 4-bit value, 0000, that produces an 8-bit value, abcdefgh, of 11111111 (255) must be distinguishable from the 0000 that produces 00000000 (0). To make this distinction, the fifth bit, Y, is used. When YABCD = 10000, dedicated logic must be invoked to invert the sinusoid magnitude generator output from all 0's to all 1's. To accomplish this, the following expression is used

$$\text{Output to D/A} = YYYYYYYY \oplus abcdefgh$$

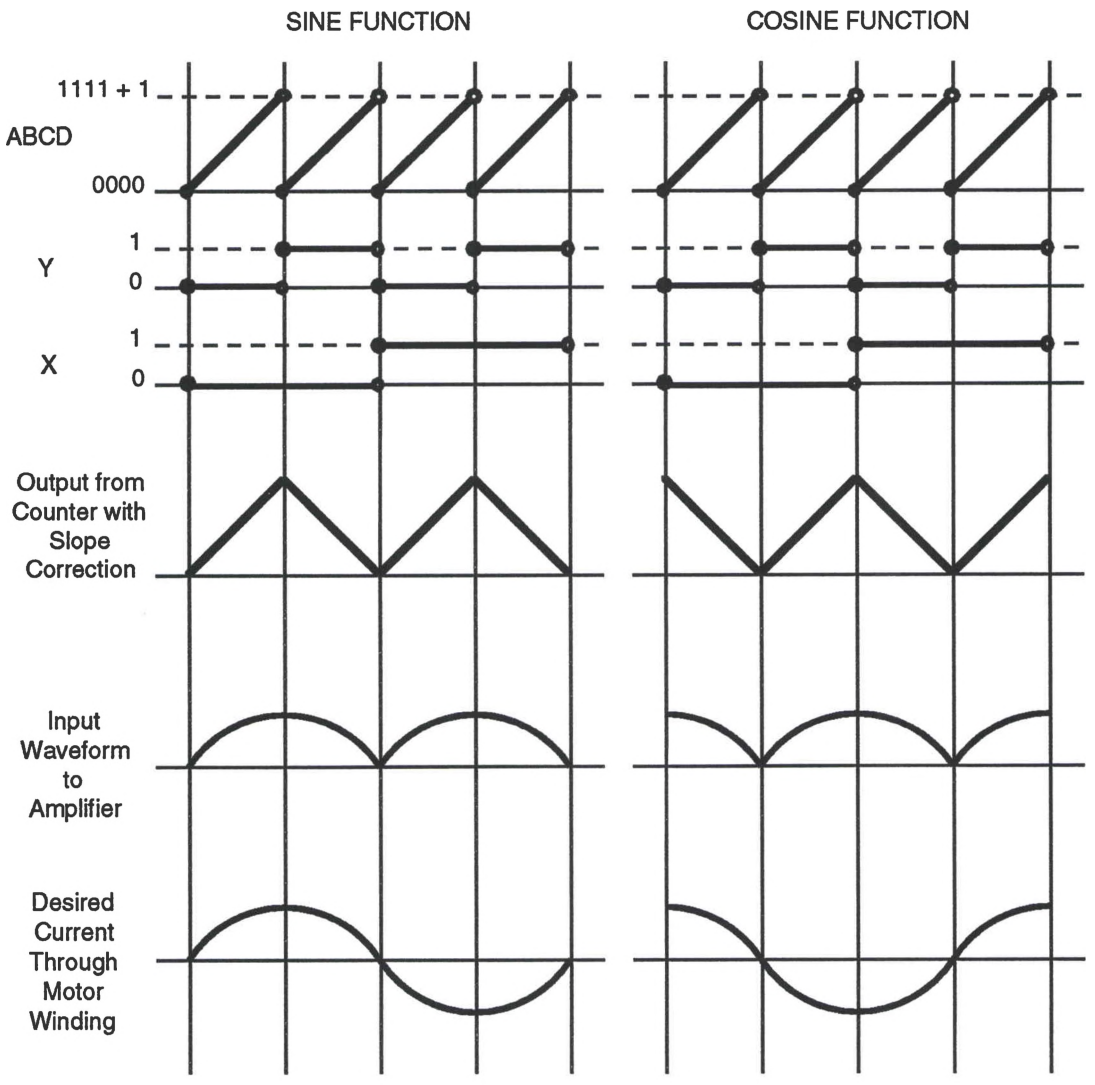


Figure 27

Evolution of Microstepping Waveforms

Before designing the sinusoidal magnitude generator and the D/A circuit, enough information is available to produce the rest of the schematic configuration for the microstep generator, shown in Figure 28. The cosine polarity function is shown with an Exclusive NOR gate, rather than requiring an additional Hex Inverter for the

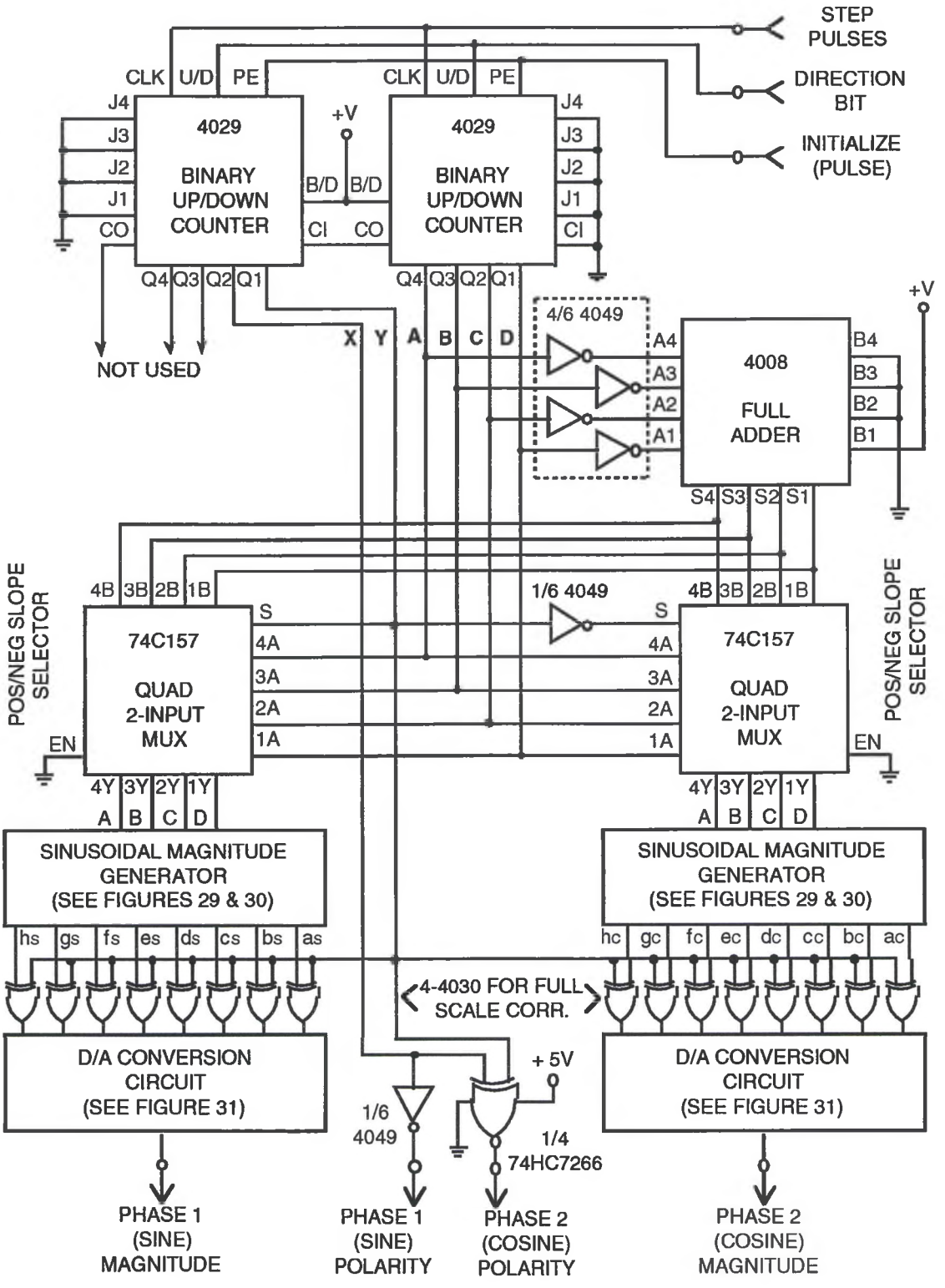


Figure 28

Microstep Generator Circuit

single NOT function. However, it should be noted that the 74HC7266 requires an additional supply voltage of 5 V.

The 4-bit word that produces the bidirectional ramp is representative of an angular increment of 90°/16, or 5.625°. At this point, a circuit is needed to convert this 4-bit word into an 8-bit word, S = abcdefgh, where

$$S = \sin (5.625N)$$

where N is the number of microsteps offset from motor pole alignment. The Truth Table, Karnaugh maps, and Quine-McKluskey analysis for each of the eight outputs can be found in Appendix C. Based on these analyses, the following Boolean expressions were derived:

$$a_S = A + BC$$

$$b_S = \bar{B}C\bar{D} + B\bar{C} + AD + AC$$

$$c_S = \bar{A}\bar{B}C\bar{D} + A\bar{C}\bar{D} + AC\bar{D} + B\bar{C} + BD + AB$$

$$d_S = \bar{A}\bar{C}D + \bar{B}C\bar{D} + A\bar{B}\bar{D} + B\bar{C}D + ABC$$

$$e_S = \bar{A}\bar{B}D + \bar{A}\bar{C}D + BC\bar{D} + AB\bar{D} + ABC$$

$$f_S = \bar{A}BC\bar{D} + A\bar{B}\bar{D} + ABD + A\bar{C}$$

$$g_S = \bar{A}B\bar{D} + \bar{A}C + BC$$

$$h_S = \bar{B}\bar{C}D + A\bar{B}D$$

There are several ways to implement these expressions, including multiplexers, Read-Only-Memory (ROM), Programmable Logic Array (PLA), Programmed Array Logic (PAL), and use of standard logic gates. Multiplexers are normally very efficient for single-function applications, but not for multiple functions. ROMs generally come with thousands of gates, much more than necessary for this application. The other three options, PLA, PAL, and standard logic, are all easily derived from the above expressions, and any of these approaches are well-suited for the application.

Before designing the circuit, however, it would be best to compare the terms in the above Boolean equations, and look for common elements. For simplification, each term is assigned a number as shown in Table 6. The previous expressions can then be rewritten as follows:

$$a_S = [26] + [23]$$
$$b_S = [15] + [24] + [22] + [19]$$
$$c_S = [2] + [11] + [10] + [24] + [25] + [18]$$
$$d_S = [12] + [16] + [7] + [14] + [3]$$
$$e_S = [9] + [12] + [13] + [5] + [3]$$
$$f_S = [1] + [7] + [4] + [20]$$
$$g_S = [8] + [21] + [23]$$
$$h_S = [17] + [6]$$

Of the 31 terms, five ([3], [7], [12], [23], and [24]) are repeated once each, leaving 26 unique terms. However, since [26] is a single-variable term, it can be wired directly. Of the 25 remaining, there

Table 6

Boolean Terms Required for Sinusoidal Magnitude Generator

Number	1	2	3	4	5	6	7	8
Term	$\bar{A}B\bar{C}\bar{D}$	$\bar{A}\bar{B}\bar{C}\bar{D}$	ABC	ABD	AB$\bar{D}$	A$\bar{B}$D	A$\bar{B}\bar{D}$	$\bar{A}$B$\bar{D}$
Variables	4	4	3	3	3	3	3	3

Number	9	10	11	12	13	14	15	16	17
Term	$\bar{A}\bar{B}$D	ACD	AC$\bar{D}$	$\bar{A}\bar{C}$D	BC$\bar{D}$	B$\bar{C}$D	$\bar{B}$CD	$\bar{B}$C$\bar{D}$	$\bar{B}\bar{C}$D
Variables	3	3	3	3	3	3	3	3	3

Number	18	19	20	21	22	23	24	25	26
Term	AB	AC	A$\bar{C}$	$\bar{A}$C	AD	BC	B$\bar{C}$	BD	A
Variables	2	2	2	2	2	2	2	2	1

are two 4-variable, 15 3-variable, and eight 2-variable terms. Of the eight functions, there are one 6-term, two 5-term, two 4-term, two 3-term expressions, and one 2-term expression. An equivalent standard configuration using all AND and OR logic gates is shown in Figures 29 and 30.

Note that the same AND array is used in both Figures. The two drawings actually show the same input circuit, which is shared by all eight outputs. Each copy of the Sinusoidal Magnitude Converter would require the circuit shown in Figures 29 and 30 (combined). If constructed of standard logic gates, one of these circuits would require the following chips:

1 4049 - Hex Inverter (to invert inputs)

2 54C08 - Quad 2-Input AND Gates

5 4073 - Triple 3-Input AND Gates

1 4082 - Dual 4-Input AND Gates

1 4075 - Triple 3-Input OR Gate

1 4072 - Dual 4-Input OR Gate

3 54HC4078 - 8-Input OR Gates

However, this could probably be best accomplished using a PLA, PAL, or similar device, based on availability of configurations with the appropriate number of AND and OR gates. An ideal PLA implementation would be a single-chip circuit nearly identical to that shown in Figures 29 and 30. Since PAL allows programmed AND inputs only, the five repeated terms would require five additional AND gates.

A newer technology, Gated Array Logic (GAL), promises to be the best option yet. A single GAL16V8 integrated circuit provides adequate logic on a single programmable chip with a propagation time as short as 10-25 nsec. This chip allows up to eight independent inputs and provides internally-buffered and inverted versions of each. The GAL chip is capable of ANDing as many of these inputs as necessary. It also provides eight outputs, each with an OR gate fed independently by eight of its own AND gates. Since only four independent inputs (ABCD) are needed, inversion is provided as necessary. Finally, since no more than eight inputs are required for any one OR gate, one GAL16V8 can easily satisfy the requirements of one copy of the circuit.

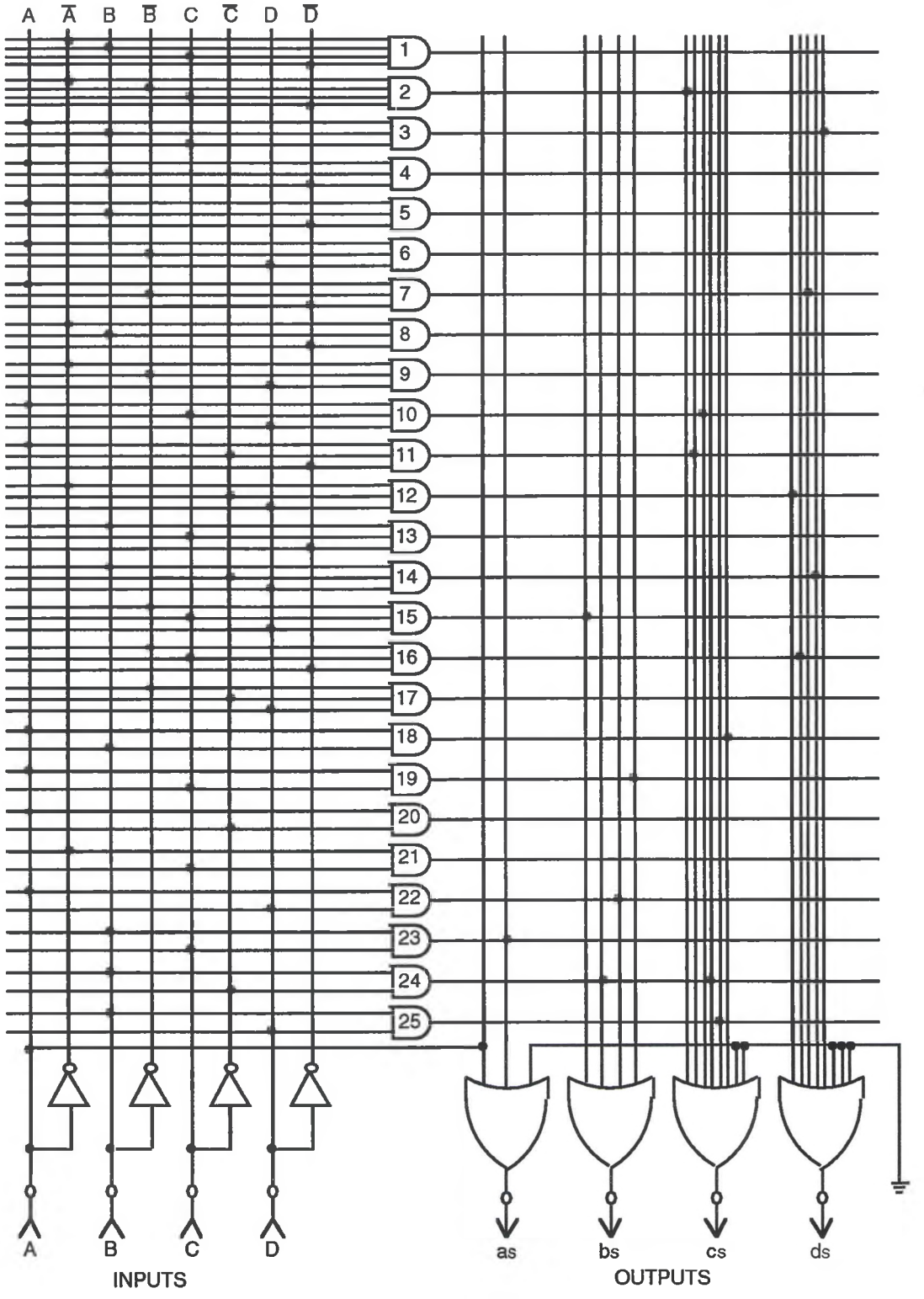


Figure 29

Sinusoidal Magnitude Generator (a_S - d_S)

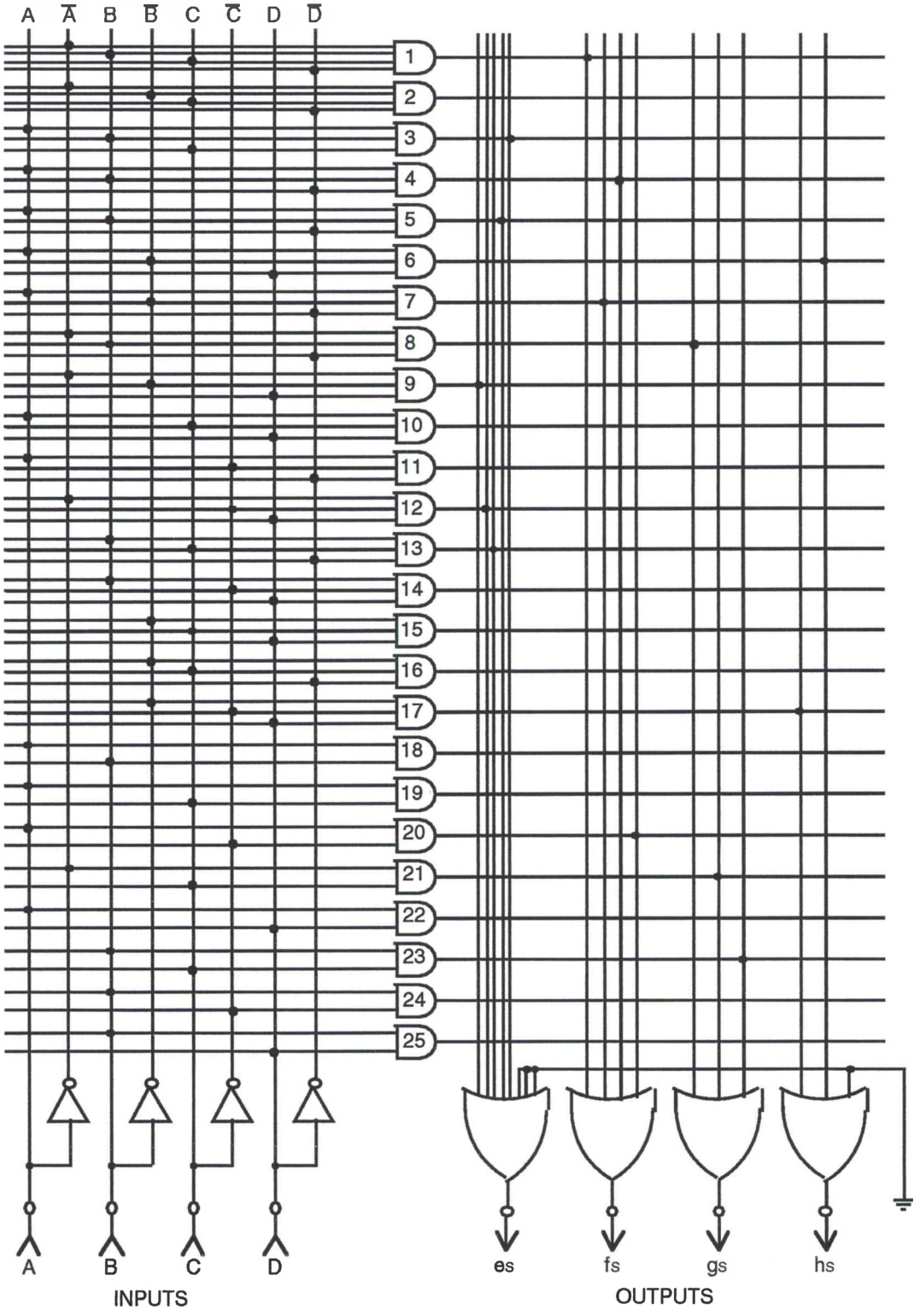


Figure 30

Sinusoidal Magnitude Generator (e_S - h_S)

Recall from the last section that the amplifier input voltages, V_1 and V_2, must be varied over the range of

$$2\ V \le V_1 \le 14.58\ V \qquad \text{for } 0 \le I_D \le 1.9\ A$$

The D/A circuit must be capable of producing an output voltage covering this range, given digital inputs ranging from 00000000 to 11111111. The diagram in Figure 31 is one way of implementing this. The DAC0830 D/A converter and the LM324 Op Amp were selected based on their unipolar operation, eliminating the need for a negative power supply. The Op Amp requires a supply voltage at least 1.5 V higher than the output voltage, which is 14.58 V, plus

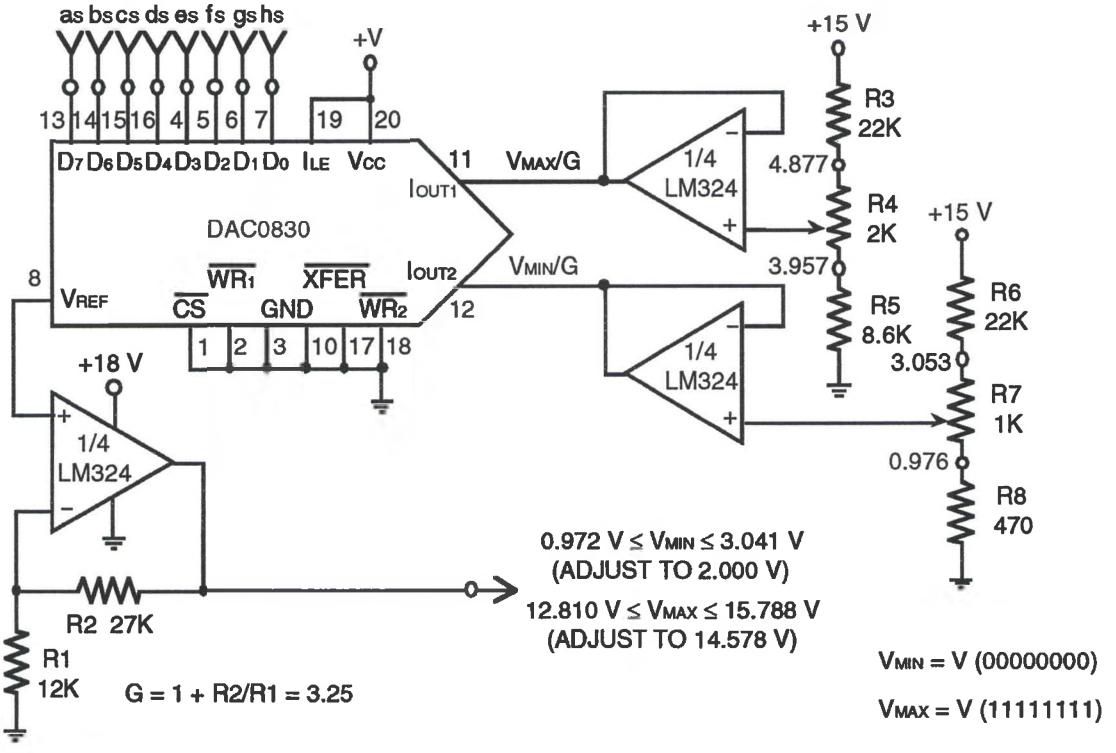


Figure 31

D/A Conversion Circuit

some room for adjustment. Therefore, the Op Amp supply voltage should be 18 V. This circuit is designed with level shift and span-adjustable output to produce a voltage proportional to the binary input between the boundary values of V_{MIN} and V_{MAX}. The voltages at the DAC0830 I_{OUT} pins must both be in the range of 0 to 5 V. In order to produce an output maximum adjusted voltage on the order of 16V while not exceeding 5 V at I_{OUT1}, the output Op Amp resistors, R1 and R2, were selected for a voltage gain of

$$G = 1 + \frac{R2}{R1} = 1 + \frac{27K}{12K} = 3.25$$

The input resistors, R3-R8, were chosen to produce the following adjustable ranges for V_{MIN} and V_{MAX} for calibration purposes:

$$V_{MIN}\ (\text{min}) = \left(\frac{0.47K}{0.47K + 1K + 22K}\right)\left(\frac{255}{256}\right)(15\ V)\ (3.25) = 0.972\ V$$

$$V_{MIN}\ (\text{max}) = \left(\frac{0.47K + 1K}{0.47K + 1K + 22K}\right)\left(\frac{255}{256}\right)(15\ V)\ (3.25) = 3.041\ V$$

$$V_{MAX}\ (\text{min}) = \left(\frac{8.6K}{8.6K + 2K + 22K}\right)\left(\frac{255}{256}\right)(15\ V)\ (3.25) = 12.810\ V$$

$$V_{MAX}\ (\text{max}) = \left(\frac{8.6K + 2K}{8.6K + 2K + 22K}\right)\left(\frac{255}{256}\right)(15\ V)\ (3.25) = 15.788\ V$$

The 255/256 is the proportionality factor associated with the maximum DAC output, given an input value of 11111111_2 (255_{10}).

Pulse Generation and Velocity Control

Although pulse generation could be handled in a number of ways, the velocity-sensitive (rather than position-sensitive) nature of the application suggests that a frequency generator rather than a time generator should be used. Furthermore, if step duration were time delay controlled, the maximum speed would be based on one delay period, and the second highest speed on two delay periods, which converts to half the maximum speed. Even if the maximum speed were based on several time delay periods, there would be significant discrete speed changes at the high end of the range, which cannot be tolerated in a stepper controller.

The circuit in Figure 32 consists of a D/A conversion circuit, followed by a precision voltage-to-frequency converter, and a cascade of counters for down-scaling. The digital input will be an 8-bit word representing one of 255 separate proportional speeds. An additional 2-bit range selector input will provide a selection of three speed ranges, plus an OFF position for calibration, initialization, user interrupts, error condition interrupts, etc. Since each of the two lower scales provide a divide-by-256, a continuous range of 765 speeds can be realized over a dynamic range of 256^3, or 16,777,216 to one.

The D/A converter in this circuit produces a voltage proportional to the value of the 8-bit word representing the desired pulse repetition frequency. However, the full scale voltage required for the input to the voltage-to-frequency converter is 10 V.

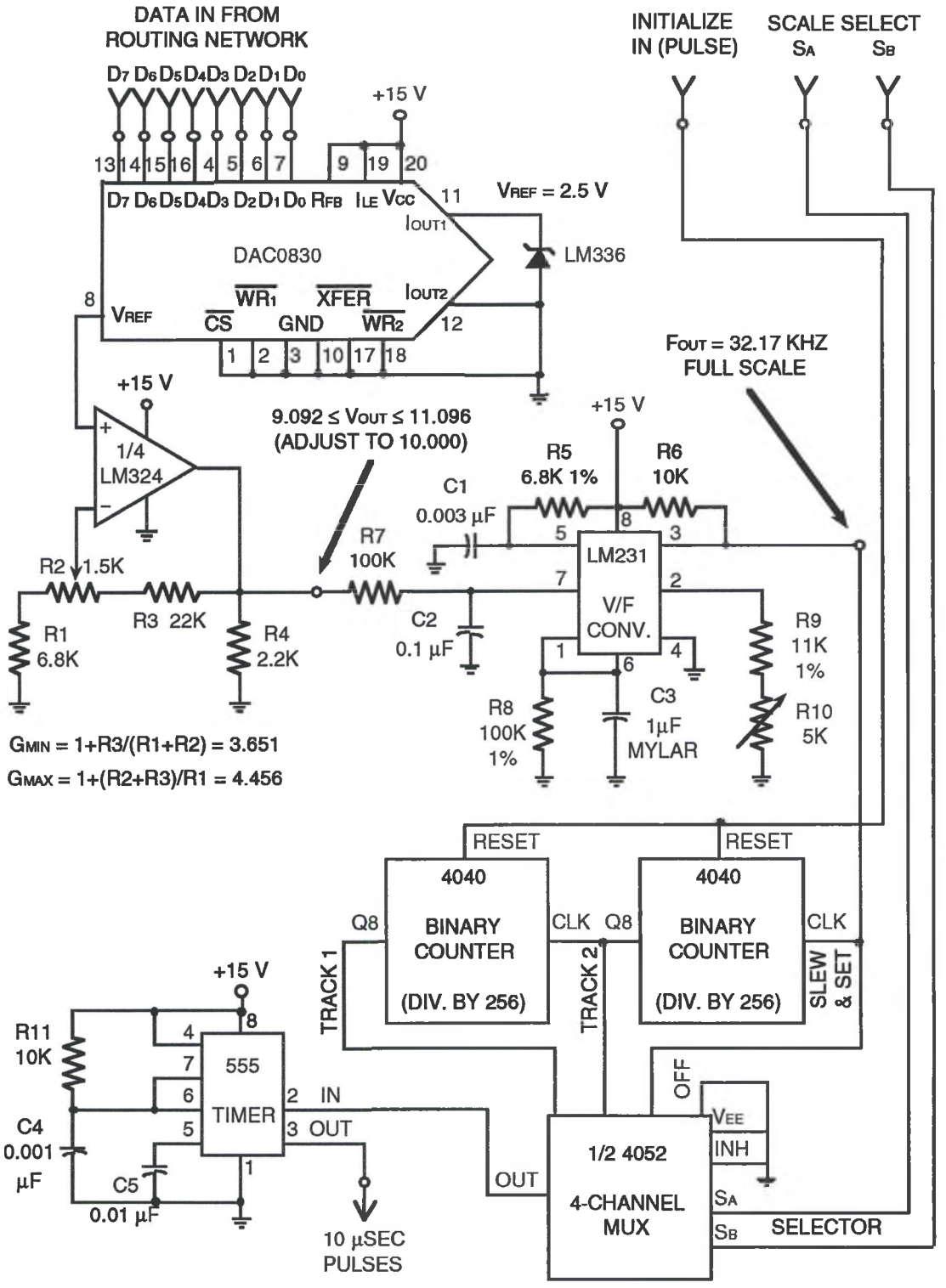


Figure 32

Pulse Generation Circuit

Therefore, it is necessary to make use of the remaining Op Amp (1/4 LM324) left over from the circuit in Figure 31. This non-inverting voltage amplifier configuration should have a gain of four to produce 10 V, given the 2.5 V reference provided by the LM336 Zener diode. However, to produce a precise 10 V output, a calibration circuit is used to provide a ±1 V adjustment. Resistors, R1-R3, were chosen to produce a variable gain over the following range.

$$G_{MIN} = 1 + \frac{R3}{R1 + R2} = 1 + \frac{22K}{6.8K + 1.5K} = 3.651$$

$$G_{MAX} = 1 + \frac{R2 + R3}{R1} = 1 + \frac{1.5K + 22K}{6.8K} = 4.456$$

This, in turn, produces a variable full scale voltage output of:

$$V_{MIN} = \left(\frac{255}{256}\right)(2.5\ V)\ (3.651) = 9.092\ V$$

$$V_{MAX} = \left(\frac{255}{256}\right)(2.5\ V)\ (4.456) = 11.096\ V$$

R4 is a pull-down resistor to reduce Op Amp saturation effects, which may impact the zero code voltage. The amplified operating voltage, calibrated to 10.000 V full scale, is then determined as

$$V_{OP} = \left(\frac{N}{256}\right)(10.000\ V)$$

where N is the decimal equivalent of the 8-bit binary input. This voltage is then fed to the LM231 precision voltage-to frequency converter circuit, which produces a pulse stream with a frequency proportional to the operating input voltage.

The maximum output frequency was chosen to correspond to twice the required slew step rate of 16 KHz, or 32 kHz, and is given by the formula:

$$f_{OUT} = \frac{V_{IN}\,(R9 + R10)}{(2.09\ V)\,R8\ R5\ C1}$$

The potentiometer, R10, provides approximately a 20% adjustment. Accordingly, the circuit was designed to produce minimum and maximum full scale output frequencies of

$$f_{OUT}\,(min) = \frac{(10\ V)\,(11\ K\Omega + 0\ K\Omega)}{(2.09\ V)\,(100\ K\Omega)\,(6.8\ K\Omega)\,(0.003\ \mu F)} = 25{,}800\ Hz$$

$$f_{OUT}\,(max) = \frac{(10\ V)\,(11\ K\Omega + 5\ K\Omega)}{(2.09\ V)\,(100\ K\Omega)\,(6.8\ K\Omega)\,(0.003\ \mu F)} = 37{,}527\ Hz$$

Essentially, the calibration range, relative to the center of adjustment, is 31,663 Hz ± 5864 Hz.

As discussed in Chapter 4, however, tracking velocities will vary over a wide range. Recall that the azimuth track velocity ranges from 0.003251 to 141.4 ´´/sec. Given a 1.125 ´´/sec step size, this converts to a pulse repetition frequency range of 0.00289 to 125.7 Hz. This minimum rate is perhaps better thought of as expressed in terms of its pulse interval: 346 seconds. It is an interesting coincidence that the maximum track rate is almost exactly 1/128 of the established slew rate. This is why the maximum frequency was chosen to be double the slew rate.

Furthermore, the minimum track rate is slightly more than $1/256^2$ of the maximum track rate. Therefore, by twice dividing the

maximum frequency by 256 and through the use of the D/A converter/voltage-to-frequency conversion process, three continuous stacked frequency ranges can be achieved. Essentially, the three ranges compose 24 continuous bits of programmable frequency. Further, due to the 128:1 ratio between maximum track and slew speeds, the two lower ranges can be dedicated for tracking, while the high range can be used solely for setting and ramping to slew speed. These ranges can be summarized as follows, given calibration for a full scale slew frequency of 32.17 KHz (256 × 125.7 Hz). The accompanying hexadecimal code is provided to signify the hex version of the respective range of velocity input commands available for each velocity scale.

Set/Slew:	126.2 Hz - 32,170 Hz	(010000 - FF0000)
Track 2:	0.493 Hz - 125.7 Hz	(000100 - 00FF00)
Track 1:	0.00192 Hz - 0.491 Hz	(000001 - 0000FF)

The two-stage divide-by-256 can be achieved using an 8-stage ripple-carry counter/divider. Since the counter is used only as a divider, it need only count in one direction, and need not be presettable to a given value. Unfortunately, National Semiconductor makes only 7-stage, 12-stage, and 14-stage ripple-carry counters. Therefore, the 4040 12-stage counter was selected.

The "two-stage, eight-stage" configuration is shown at the bottom of Figure 32. Any of the three ranges, as well as an OFF option, can be selected by the 4052 4-Channel Multiplexer. The second half of the 4052 can be utilized by the second motor. The two

select bits must be supplied by the routing network, separate from the 8-bit data bus.

Finally, since the counters will only produce a periodic state-change, the 555 Timer is used in its monostable, or "one-shot", mode to produce a standardized 10 μsec pulse each time the Multiplexer output goes low. The pulse width is determined by the RC time constant, or

$$PW = \tau = (R11)\,(C4) = (10\ K\Omega)\,(0.001\ \mu F) = 10\ \mu sec$$

Position Sensor Selection

As mentioned earlier, two position feedback loops are preferred - one sensed at the motor/worm axis and one at the axis of rotation. As indicated in Chapter 4, the resolution of worm-mounted feedback sensor is not practical beyond 800 position increments per motor revolution. In terms of the outer loop, sensed at the rotational axis, 15′ resolution can be achieved with 1440 position increments per axis revolution.

Several position sensor options are available. However, many of these are AC devices (i.e., mutual inductance potentiometers, inductive potentiometers, reluctance potentiometers, etc.), and thus, can be immediately eliminated. The remaining DC devices can be categorized as either analog or digital. DC analog sensors include resistance potentiometers, Hall Effect potentiometers, strain guages, ferromagnetic sensors, and photo-optic devices. Digital sensors include incremental and absolute shaft encoders.

One immediately obvious disadvantage of all analog sensors is the required used of an A/D converter, since a digital interface is required, although this fact, alone, is not a major concern. Two of the analog types can be immediately eliminated: strain guages, since their application is limited to very small relative position changes, and photo-optic devices, which are not yet readily available commercially. (42: 16)

Resistance potentiometers are simple voltage-divider circuits, very inexpensive, and can provide infinite resolution, but they have many significant drawbacks. They are very noisy devices, highly sensitive to loading effects, are electrically discontinuous, and are often mechanically discontinuous, although multi-turn versions are readily available.

Hall Effect potentiometers also act as voltage dividers, but are more reliable and slightly less load-sensitive than resistance potentiometers. However, linearity tolerance tends to be on the order of 2.0%, versus a typical 1.0% for resistance potentiometers. Finally, Hall Effect potentiometers tend to be particularly sensitive to temperature and humidity. (42: 16)

> Ferromagnetic position sensors are very accurate (and) display infinite resolution. . . . Their chief drawbacks are size; they are relatively large compared to similar devices, and their need for extensive peripheral circuitry; usually supplied external to the actual transducer (42: 16).

In addition to the need for A/D conversion for interfacing, each of the DC analog sensor options has some significant disadvantages which are of concern for this application. Consequently, it appears that a digital problem calls for a digital solution. Shaft encoders provide position information with high precision, albeit of finite resolution, very low noise, and which requires little or no digital processor interface. Further, they are popular, easy to use, readily available, and unlike most DC analog sensors, encoders are not load sensitive.

Incremental optical encoders contain a thin shaft-mounted disk with a series of evenly spaced slits which intermittently pass between a (usually infrared) LED and a phototransistor, generating a stream of pulses, normally square, with a 50% duty cycle. By using a digital counter, relative position information is easily derived. Given one square pulse per slit, the leading and trailing edges respectively provide two counts per pulse. Normally, a second band of slits with its own optical transceiver is offset from the first by half the slit width to produce a second channel 90° out of phase from the first. This produces a quadrature relationship, as shown in Figure 33, permitting twice the resolution and a directional tracking capability, as well. These two channels are designated A and B, as indicated. A third band with a single slit per revolution, called an index, zero, or Z channel, is normally provided to designate a "home" position.

Absolute encoders contain a disk with a pattern of several bands, each of which contains alternating transparent and opaque sectors of various angular length. Each successive band has twice as

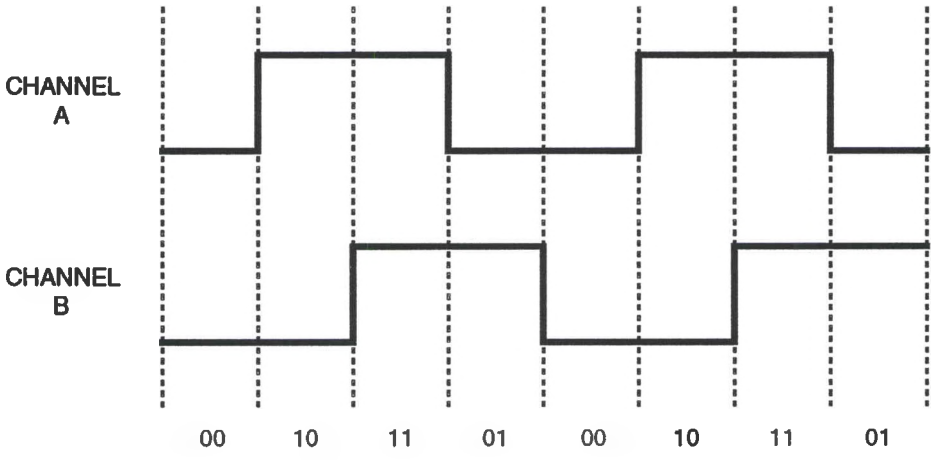


Figure 33

Encoder Output Signals

many sectors of half the angular length of the previous one, providing finer angular resolution. The pattern is arranged so that at any given shaft angle, a unique digital code is generated, usually a gray code, but sometimes binary or BCD. The output is a digital word of N parallel bits, where N is the number of bands.

Perhaps the bigger question is not whether to use an encoder, but which type, and what resolution is required. Absolute encoders are the easiest to use, since they directly provide a unique digital shaft angle code identifier, and therefore, require no further processing. Unfortunately, they are very expensive, and not as readily available as the more popular incremental devices. For 16-bit resolution, an absolute encoder costs about four times that of an equivalent resolution incremental encoder, and the ratio increases for higher resolution. (66: 66) As a result, absolute encoders should only be employed in applications where "homing" is impractical, or

perhaps in the rare instance where motor speeds generate pulse rates which exceed the encoder frequency response, typically on the order of 100 kHz.

Based on these considerations, incremental encoders are the practical choice. To meet the goal of 1440 increments per revolution for rotational axis angle measurement, 11 bits (512 slits × 2 counts per slit × 2 quadrature channels = 2048 steps per revolution) is sufficient. For altitude measurement, only the nine LSBs of data will be needed, however, since the range of rotation is limited to 90°. A nine-bit encoder could be used here with a 1:4 gear increase, if the encoder cost savings outweighs the cost of the additional gearing and the increase in load inertia is tolerable. However, this is highly unlikely. Also, since the altitude range is limited to 90°, it is essential that the "home" channel pulse be generated within that quadrant, preferably very close to the horizon. The azimuth axis encoder should be positioned such that the Z-channel pulse occurs as close to South as possible.

For motor shaft angle monitoring, eight bits per revolution (64 slits in two-channel quadrature) should be sufficient, especially given that this loop is only to ensure microstepping occurs properly. Since zeroing is unnecessary here, an index channel is not required. Eight bits will provide 256 increments per revolution, compared with 3200 motor microsteps per revolution. Hence, one angle position increment will be fed back for every 12.5 microsteps of motor advancement.

A variety of incremental encoders from several manufacturers were considered. PMI Motion Technologies' M23 series encoders were found to meet the above specifications. They are available in a variety of cycle counts and output configurations. The selection for the axis-mounted encoders is the M23-2048-ABI-15-S, a 2048-count (11-bit/512-slit) encoder with two channels plus index and CMOS-compatible 15-volt squarewave outputs. For the motor shaft-mounted sensors, the choice would be the M23-256-AB-15-S, a comparable 256-count encoder without an index channel. (45)

Feedback Mechanism

Given the position sensors have been chosen, the next step is to design the feedback circuit to perform counting and direction decoding. The input signals from the encoder consist of three channels, A, B, and Z, where A and B are the two square wave channels in quadrature and Z is the zero or "home" channel, producing a single narrow pulse per revolution. The digital feedback circuit will need to detect changes of state in both A and B channels and increment or decrement a counter, given a change in either channel. Additional circuitry will be required to determine the current direction of rotation (0 = CCW, 1 = CW). This bit would then be fed to the UP/DOWN input of the counting portion of the circuit. The counter should be reset with the Z-channel pulse. The same circuit can be used for all four feedback loops, except that the counter stage of altitude and azimuth axis feedback loops will be

required to count to nine and 11 bits, respectively, while the motor loops need only count to 8 bits.

Consider the circuit in Figure 34. The state transition detector portion consists of 1/2 of a 40175 Quad D Flip-Flop, followed by two XNOR gates, NANDed together, and fed to the clock of the counter circuit. Since they would share the same clock, the other half of the 40175 flip flop could be used for one of the other feedback circuits.

Since 12.5 microstep pulses should occur for every motor position feedback increment, this pulse train can be used as a clock to control the state transition detector. Essentially, the D flip flops monitor the state of the A and B channels by sampling them at every microstep pulse. The logic is then compared with the state of the previous sample to see if the condition has changed. The XNOR outputs both remain high, and therefore the NAND that follows remains low, unless a change has occurred. If a change does occur on either channel, one XNOR output will go low momentarily until the next sample. This, in turn, produces a pulse at the NAND output, incrementing or decrementing the counter circuit on the negative edge.

The rest of the circuit is used to determine the direction of rotation. In order to develop the design, it is first necessary to understand the quadrature relationship of the two shaft encoder output Channels A and B. Figure 33 illustrates the corresponding logic levels of each channel, moving left to right for counter-clockwise (CCW) encoder rotation. This, in turn, converts to clockwise

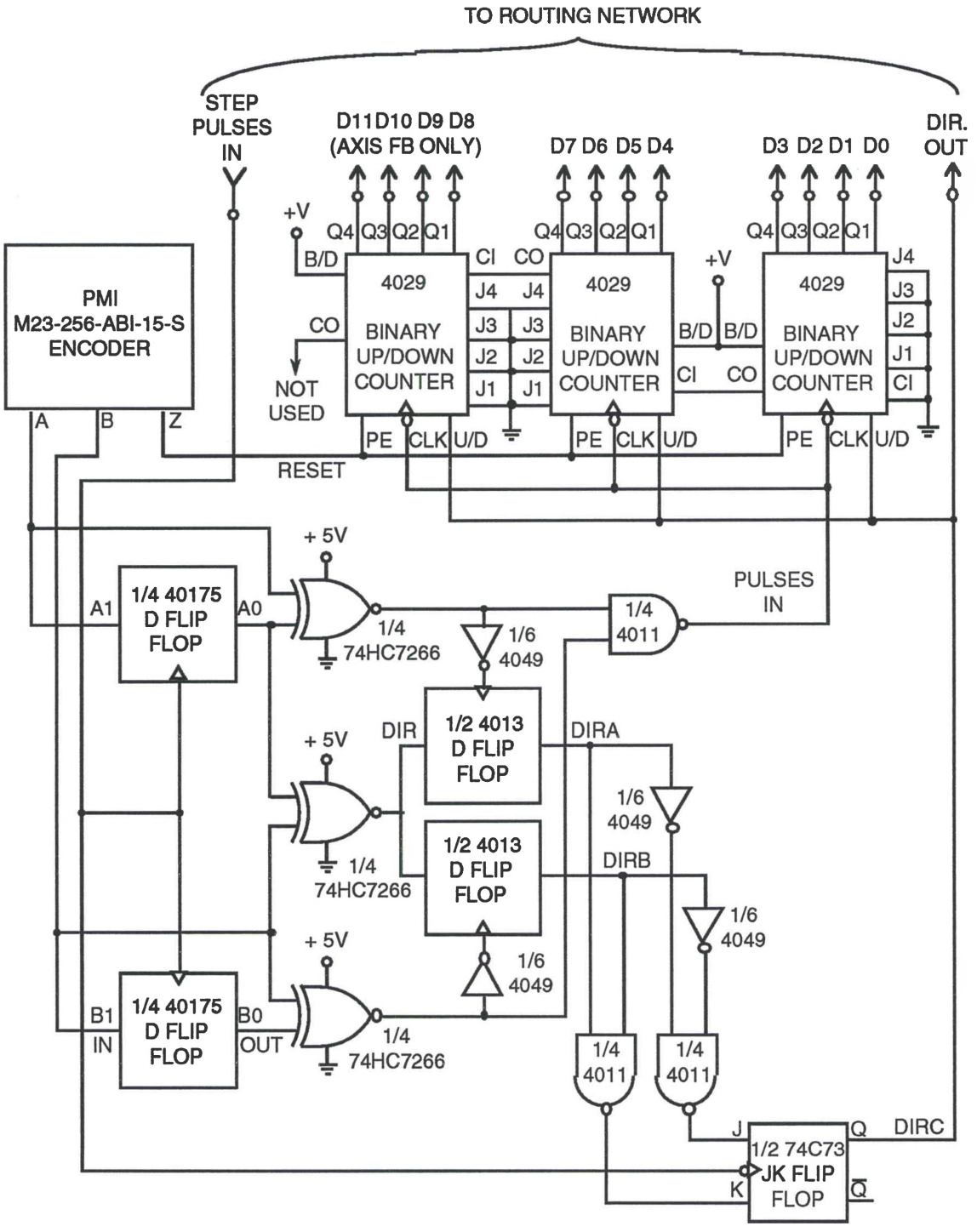


Figure 34

Feedback Circuit

(CW) motor rotation, since motor-position sensors are mounted with shafts facing their respective driving motor. Additionally, the azimuth axis encoder, mounted inverted at the base of the rocker box, will move CCW for positive (CW) azimuth rotation (i.e., South to West). Since positive altitude rotation is rising, the altitude encoder should be mounted on the outside of the left fork for CCW encoder movement when rising. Thus, there is a consistent convention of positive movement as the signal logic levels of each channel change from left to right in Figure 33, which represents CCW encoder rotation and CW motor/axis rotation. Conversely, CW encoder rotation (CCW motor/axis rotation) will be designated as negative rotation. This convention will be used to determine what to apply to the Up/Down input to the counting circuit.

The direction can only be determined by the ordered pattern of state transitions as they occur, and therefore, will require additional sequential logic. Consider the current sample of the A-channel as A1, the previous sample A0, and the same for the B-channel. By creating a truth table for each possible set of "before and after" conditions, it is possible to produce an at least partially-representative Boolean expression.

Note from Figure 33 that the two channels produce a Gray code pattern, in which only one channel changes state at a time. Thus, four of the 16 possible combinations of A0, A1, B0, and B1 are eliminated, namely, those that indicate simultaneous changes on both channels. However, it may be less cumbersome to treat state transitions of each channel separately. In so doing, the two truth

tables in Figure 7 were produced, one to monitor A-channel transitions and one to monitor B-channel transitions. A "0" in the direction column is to signify negative rotation, a "1" represents positive rotation, and NC constitutes "No Change".

Table 7

Truth Tables for Intermediate Direction Decoding

A0	A1	B0/B1	DIR
0	0	0	NC
0	0	1	NC
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	NC
1	1	1	NC

A0/A1	B0	B1	DIR
0	0	0	NC
0	0	1	0
0	1	0	1
0	1	1	NC
1	0	0	NC
1	0	1	1
1	1	0	0
1	1	1	NC

The tables show 16 conditions, but only 12 are unique. Eight of the conditions shown are actually four unique conditions, those that are indicative of no change of state. In these cases, the desired output would be "No Change", since rotation in the same direction would be assumed until a state transition occurs on one of the

channels. To produce Boolean expressions, substitute A for the A0/A1 combination, since the logic does not care which sample is used (i.e, they are the same), and likewise, substitute B for the B0/B1 combination. Eliminating the No Change conditions, the logic can be represented by the following XOR expressions:

$$DIR = A1 \oplus B, \text{ given that A changes state, or}$$
$$DIR = A \oplus B0, \text{ given that B changes state.}$$

Alternately, XNOR logic could be used as follows:

$$DIR = \overline{A0 \oplus B}, \text{ given that A changes state, or}$$
$$DIR = \overline{A \oplus B1}, \text{ given that B changes state.}$$

Since A represents A0/A1 and B represents B0/B1, the above XOR expressions can be reduced to a single Boolean equation,

$$DIR = A1 \oplus B0$$

which can be done with a single gate. Alternately, with XNOR logic,

$$DIR = \overline{A0 \oplus B1}$$

However, the No Change conditions still need to be satisfied. If DIR is fed into a flip flop triggered by a state transition bit from Channel A, a direction bit, DIRA, can be stored and updated only when a Channel-A transition takes place. Similarly, DIR can be fed to another flip flop triggered by a state transition bit from Channel B, and another direction bit, DIRB, can be stored and updated only when a Channel-B transition takes place. The 4013 Dual D Flip Flop

was selected for this application, due to the need for independent clocks.

The question now remains as to what to do with DIRA and DIRB. If their logic values agree, it means two consecutive transitions, one on each channel, have detected that rotation is still in the same direction. In this case, the answer is simple, and that logic level, designated DIRC, is presented to the Up/Down bit of the counter circuit. If DIRA and DIRB disagree, then a change of rotational direction has occurred, and DIRC should be changed from its previous value, designated DIRP. Table 8 shows shows a truth table relating DIRA and DIRB to the desired DIRC output. Noting that this could be implemented with a J-K Flip Flop, a J-K truth table with output DIRC is included for comparison, and finally a third truth table relating DIRA and DIRB with the required J-K inputs. "NA" indicates the condition is not allowed, as it never occurs.

Table 8

Truth Tables for Final Direction Decoding

Desired Logic

DIRA	DIRB	DIRC
0	0	0
0	1	$\overline{\text{DIRP}}$
1	0	$\overline{\text{DIRP}}$
1	1	1

J-K Logic

J	K	DIRC
0	0	NA
0	1	0
1	0	1
1	1	$\overline{\text{DIRP}}$

Required J-K Input

DIRA	DIRB	J	K
0	0	0	1
0	1	1	1
1	0	1	1
1	1	1	0

From the last truth table, J is easily recognized as an OR function, while K is a NAND function. However, by DeMorgan's Theorem, J can be transformed to a NANDed function of inverted inputs. The Boolean expressions are then

$$J = DIRA + DIRB = \overline{(\overline{DIRA})\,(\overline{DIRB})}$$

$$K = \overline{(DIRA)\,(DIRB)}$$

From these expressions, the counter Up/Down input circuit was designed as shown in Figure 34. The DIRC output is also provided back to the computer via the routing network, discussed in the next section.

The counting circuit consists of a cascade of up to three 4029 Presettable Binary/Decade Up/Down Counters, configured to count in binary. The presettable inputs are only used for reset, as triggered by the Z-Channel input. Thus, all Jam inputs are grounded. Only two of these counters in cascade are required for the two motor shaft angle feedback loops. Since 11 and nine bits are needed for azimuth and altitude axis position monitoring, respectively, the third counter would be used in these two cases. Since only one additional bit is required for the azimuth axis channel, a flip flop could also be used to produce the ninth bit.

As in the Microstep Generator circuit, a 5 V supply is required for the XNOR gates. To eliminate the 5 V supply, the two XNOR gates used in the state transition detector followed by the NAND could be replaced by two XOR gates, followed by an OR gate. The two

inverters that feed the 4013 clocks could then be eliminated. The additional XNOR that begins the direction-detecting process could be replaced by an XOR, and changing the inputs to A1 and B0.

Routing Network

The last major hardware design element is the routing control circuit, required to perform the task of managing data flow in and out of the computer controller. Put another way, it is required to send 22 command bits and monitor 40 bits of feedback data, all via an 8-bit parallel port. The best way to accomplish this is to design a circuit with three 8-bit output registers and five 8-bit input registers. Registers could then be selected and connected, thereby multiplexing the inputs and demultiplexing the outputs sequentially by software control. Three address lines (designated A2, A1, and A0) would be required for channel selection, a Read/Write Enable (R/W EN) line to permit port isolation by disconnecting all of the eight registers, and an Initialize (INIT) line to perform various reset functions at powerup. Care must be taken to ensure that only one channel is connected at any given time to prevent conflicting inputs, inputs contaminating outputs, and misrouted outputs.

The following is a listing of the 22 output command bits that are necessary:

8 Bits	Azimuth Velocity Value
8 Bits	Altitude Velocity Value
2 Bits	Azimuth Range Selector

2 Bits	Altitude Range Selector
1 Bit	Azimuth Direction
1 Bit	Altitude Direction

The following is a listing of the 40 input bits to be monitored during system operation:

11 Bits	Azimuth Axis Position
9 Bits	Altitude Axis Position
8 Bits	Azimuth Motor Position
8 Bits	Altitude Motor Position
1 Bit	Azimuth Axis Direction
1 Bit	Altitude Axis Direction
1 Bit	Azimuth Motor Direction
1 Bit	Altitude Motor Direction

The first step is to arrange all the inputs and outputs in logical groupings forming 8-bit words that can be easily transferred between the registers and the computer port. Each of these groupings is then given a channel designation. Table 9 shows such a reasonable arrangement.

The routing network design is shown in Figure 35. A 74HC237 3-to-8 Decoder with Address Latches was used to perform the multiplexing/demultiplexing function. Given a separate 5-bit port or combination of five bits from a number of ports, three are used to address the 74HC237, which selects or "turns on" one of the eight registers by connecting it to the 8-bit port. A R/W Enable line is

Table 9

I/O Channel Designations

Channel	Type	No. of Bits	Bit No.	Function
0	Output	2	0-1	Azimuth Range Select
		2	2-3	Altitude Range Select
		1	4	Azimuth Direction
		1	5	Altitude Direction
		2	6-7	TO BE DETERMINED
1	Output	8	0-7	Azimuth Velocity
2	Output	8	0-7	Altitude Velocity
3	Input	1	0	Azimuth Motor Dir
		1	1	Altitude Motor Dir
		1	2	Azimuth Axis Dir
		1	3	Altitude Axis Dir
		1	4	Az Axis Encoder, Bit 8
		1	5	Az Axis Encoder, Bit 9
		1	6	Az Axis Encoder, Bit 10
		1	7	Alt Axis Encoder, Bit 8
4	Input	8	0-7	Az Axis Enc, Bits 0-7
5	Input	8	0-7	Az Motor Enc, Bits 0-7
6	Input	8	0-7	Alt Axis Enc, Bits 0-7
7	Input	8	0-7	Alt Motor Enc, Bits 0-7

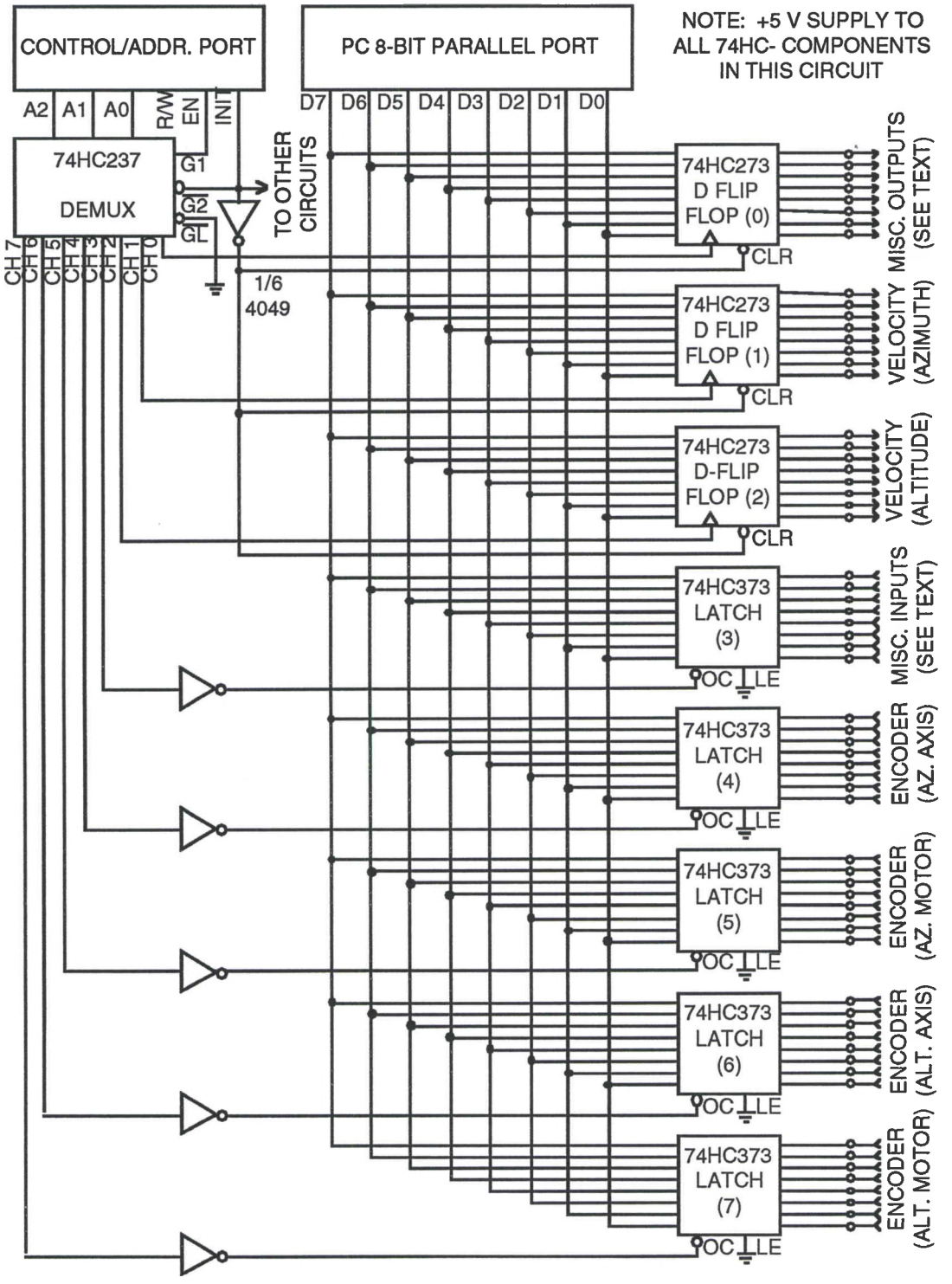


Figure 35

Routing Network Circuit

connected to the G1 (enable) input while the Initialize line is connected to the $\overline{G2}$ (inhibit) input. If G1 is low or $\overline{G2}$ is high, as it is at initialization, all outputs are low, which isolates the port from all registers. Only if G1 is high and $\overline{G2}$ is low, the addressed register is selected. The $\overline{GL}$ input is tied low to disable the latching function, which is neither required, nor desired.

To provide registers for the output channels, three 74HC273 Octal D Flip-flop with Clear chips were selected. The corresponding channel select line from the 74HC237 is tied to the clock input of each flip flop. When a flip flop output channel is selected, the clock input goes high and the data on the port is transferred to the flip flop outputs, and is latched there to provide control to other circuits. Since these are octal flip flops, all eight bits can be controlled from one IC package. The Clear function is very important. At initialization, the inverted INIT bit is presented to the inverted clear input on each output flip flop, clearing all output lines. This not only ensures stability of the D/A converters in the Pulse Generation circuit, but also ensures that the velocity range selectors for both motors are initially set to the OFF position.

To provide registers for the input channels, five 74HC373 Octal D-Type Latches were used. In this case, the corresponding channel select line from the 74HC237 is inverted and fed to the Output Control input of the latch. When a latch input channel is selected, the Output Control input goes low and the incoming data is transferred to the port. There, it can be read into computer memory, when so commanded. Once the data has been read, the line is deselected, and

the latch Output Control goes high. When this occurs, the latch assumes its TRI-STATE high impedance (HI-Z) state, essentially disconnecting it from the port. The Latch Enable input is tied low, as the latching function, itself, is not required.

One problem with this configuration, as was also the case for the Microstep Generator and the Feedback circuit, is the need for the 5 V supply. The main driver for this requirement is the 74HC273 Octal D Flip-flop with Clear. There exists a 15-volt 74C374 Octal D Flip-flop, but it lacks the Clear function, necessary for initialization. If the 5-volt supply is to be eliminated, all the 74HC-series chips would need to be replaced by an equivalent 4000-series or 74C-series implementation. These include the 74HC273 flip-flops, the 74HC373 latches, the 74HC7266 Exclusive NOR gates in previous circuits, and the 74HC237 decoder. The flip-flops could each be replaced by two 40175 Quad D Flip-flops, which have Clear capability, but this adds three more chips. The 74HC373 latches are directly replaceable by 74C373s. The 74HC7266 Exclusive NORs could be replaced by 4030 Exclusive ORs while adding or removing inverters.

Finally, the 74HC237 decoder could be replaced by a number of options. One way would be to use a 4051 Single 8-Channel Analog Multiplexer/Demultiplexer as a digital demultiplexer by tying the input high. Another option is to use an 8-Bit Addressable Latch, such as a 4099 or a 4724, but disabling the latching function to allow operation as a normal demultiplexer. In any of these cases, additional logic will be required to select the case where Init = 0 and

R/W En = 1 to connect the selected register (flip-flop or latch) to the 8-bit port. This logic is already built into the 74HC237, which has independent enable and inhibit inputs.

PC Controller

The routing network could be compatible with most any personal computer with an 8-bit parallel port that can be controlled as a user port. Since numerous models exist in a variety of configurations, one may wish to maintain the flexibility to chose whatever system/configuration is available that would be suitable for this application. Therefore, rather than make a selection and "lock in" on a system-specific design, this section will discuss the computer controller only in generality. The required processing frequency and memory capacity will be dependent on a number of factors, including the speed of the programming language chosen. However, using a language of reasonable speed (i.e., FORTRAN, PASCAL, C, FORTH, etc.), it is believed that most any standard 4.77 MHz or 8.00 MHz machine should do the job with as little as 1 MB of memory or less.

Interfaces required include a standard 8-bit parallel connection and five additional bits for addressing and control. These include three address bits to select the desired I/O channel, one R/W enable bit to enable data transfer, and one bit for initialization. Generally, a Centronics or other parallel printer port and/or mouse/joystick ports can be used for this purpose. However, whichever port is used for the 8-bit data exchange must be configurable for input or output.

This is normally done by resetting or setting the bits in a given memory location reserved for data direction designation. Control and address lines, of course, need not be bidirectional.

Power Supplies

All supply voltages are unipolar 15 V, unless otherwise specified. Due to the motor winding resistance, maximum winding current required, and the MOSFET characteristics, the drive amplifier supply voltage must be 18 V. Also, in the D/A conversion circuit, at the output stage of the Microstep Generator, the LM324 Op Amp at the DAC0830 output also requires 18 V, for the reasons specified. However, 15 V is also separately required, since 18 V exceeds the supply voltage ratings of some components, such as the DAC0830. Finally, a 5 V supply is also required for the high-speed 74HC-series CMOS integrated circuits used mostly in the routing network. This extra supply could be eliminated as explained earlier. All supply voltages can be obtained from the same main power supply by use of separate voltage regulator circuits. The 18 V supply should be rated for at least 3.0 A, while the 5 V and 15 V supplies should be rated for 1.0 A each.

To aid the initialization procedure, it is suggested that an additional high-current switching device be added to the drive amplifier circuit, placed in series with each set of motor windings. This would provide the flexibility to power up all other control hardware first, thus allowing time for all registers to reset, before

manually switching on the motor windings. Otherwise, the motors may start running in an unstable, uncontrolled condition.

Summary of Hardware Configuration

Of the hardware circuits developed in this chapter, several require duplicate fabrications to produce all the necessary I/O channels for a completed system. Such a complete system would include:

2 Amplifier circuits with motors
2 Microstep Generator circuits
4 Sinusoidal Magnitude Generator circuits
(Actually part of Microstep Generator)
4 D/A Conversion circuits
(Actually part of Microstep Generator)
2 Pulse Generation circuits
2 Axis Position Feedback circuits
2 Motor Position Feedback circuits
1 Routing Network circuit

Based on this information, the following table summarizes the total parts required. Parts listed for the Microstep Generator do not include those for the D/A Conversion or Sinusoidal Magnitude Generator circuits. Fractions represent number of modules required versus modules per package.

Although this completes the detailed hardware design, it does not necessarily represent a final design. The design should remain

somewhat flexible throughout the software design stage, as software may still impact the hardware design, and some modifications can be expected. In accordance with good engineering practices, multilevel subsystem tests should be made during the fabrication process as the hardware is constructed.

Table 10

Parts List

Part Description	Circuit	No. in Circuit	Total Pkgs.
Motor, Superior M062-FC06	Amplifier	1	2
Encoder, PMI M23-256-ABI-15-S	Motor Feedbk	1	2
Encoder, PMI M23-2048-ABI-15-S	Axis Feedback	1	2
IC, 4008 4-Bit Full Adder	Microstep Gen.	1	2
IC, 4011 Quad 2-Input NAND Gate	All Feedback	3 / 4	3
IC, 4013 Dual D Flip Flop (Ind. Clk)	All Feedback	2 / 2	4
IC, 4029 4-Bit Up/Down Counter	Microstep Gen. Motor Feedbk Axis Feedback	2 2 3	14
IC, 4030 Quad 2-Input XOR Gate	Microstep Gen.	16 / 4	8
IC, 4040 12-Stage Binary Counter	Pulse Gen.	2	4
IC, 4049 Hex Inverter	Amplifier Microstep Gen. All Feedback Routing Net.	4 / 6 6 / 6 4 / 6 6 / 6	7
IC, 4052, Dual 4-Input Multiplexer	Pulse Gen.	1	2

Table 10 - Continued

Part Description	Circuit	No. in Circuit	Total Pkgs.
IC, 40175 Quad D Flip Flop	All Feedback	2 / 4	2
IC, 74C73 Dual J-K Flip Flop	All Feedback	1 / 2	2
IC, 74C157 Quad 2-Input Mux	Microstep Gen.	2	4
IC, 74HC237 3-to-8 Decoder	Routing Net.	1	1
IC, 74HC273 Octal D Flip Flop	Routing Net.	3	3
IC, 74HC373 Octal D Latch	Routing Net.	5	5
IC, 74HC7266 Quad 2-Input XNOR	Microstep Gen. Routing Net.	1 / 4 3 / 4	2
IC, DAC0830 8-Bit D/A Converter	D/A Conv. Pulse Gen.	1 1	6
IC, GAL16V8 Programmable Logic	Sin. Mag. Gen.	1	4
IC, LM231 Voltage/Freq. Converter	Pulse Gen.	1	2
IC, LM324 Op Amp	D/A Conv. Pulse Gen.	3 / 4 1 / 4	4
IC, LM555 Timer	Pulse Gen.	1	2
Transistor, 2N6656 Power MOSFET	Amplifier	2	4
Transistor, IRF100 Power MOSFET	Amplifier	8	16
Diode (Zener), LM336	Pulse Gen.	1	2
Diode, 1N4001	Amplifier	8	16
Capacitor, 0.001 µF	Pulse Gen.	1	2
Capacitor, 0.003 µF	Pulse Gen.	1	2
Capacitor, 0.01 µF	Pulse Gen.	1	2

Table 10 - Continued

Part Description	Circuit	No. in Circuit	Total Pkgs.
Capacitor, 0.1 μF	Pulse Gen.	1	2
Capacitor, 1.0 μF, Mylar	Pulse Gen.	1	2
Resistor, 47 Ω, 1/2 W	Amplifier	8	16
Resistor, 470 Ω, 1/4 W	D/A Conv.	1	4
Resistor, 2.2K Ω, 1/4 W	Pulse Gen.	1	2
Resistor, 6.8K Ω, 1/4 W	Pulse Gen.	1	2
Resistor, 6.8K Ω, 1/4 W, 1%	Pulse Gen.	1	2
Resistor, 8.6K Ω, 1/4 W	D/A Conv.	1	4
Resistor, 10K Ω, 1/4 W	Pulse Gen.	2	4
Resistor, 11K Ω, 1/4 W, 1%	Pulse Gen.	1	2
Resistor, 12K Ω, 1/4 W	D/A Conv.	1	4
Resistor, 22K Ω, 1/4 W	D/A Conv. Pulse Gen.	2 1	10
Resistor, 27K Ω, 1/4 W	D/A Conv.	1	4
Resistor, 100K Ω, 1/4 W	Pulse Gen.	1	2
Resistor, 100K Ω, 1/4 W, 1%	Pulse Gen.	1	2
Precision Trimmer Pot., 1.0K Ω	D/A Conv.	1	4
Precision Trimmer Pot., 1.5K Ω	Pulse Gen.	1	2
Precision Trimmer Pot., 2.0K Ω	D/A Conv.	1	4
Precision Trimmer Pot., 5.0K Ω	Pulse Gen.	1	2

CHAPTER 6

PRELIMINARY SOFTWARE DESIGN

Design Approach

The purpose of this chapter is to present a preliminary software design only, one intended to be portable, so it can be coded in various languages and implemented on various systems. Most detailed software design/coding will be left to the programmer, who will have the flexibility of choosing a system and programming language.

The functions of the software are to control initialization and all I/O functions, perform date/time and coordinate conversion calculations, determine velocity and direction commands, read and interpret position sensors, provide control for slew, set and track, provide user interface, detect and service error conditions, manage an object data base, and otherwise manage data and memory. These functions can be grouped into three categories: control system operations, positioning functions, and motor control functions. With the exception of the top-level command sequence, they will be developed and discussed in that order. The three categories may be defined as follows:

1. Control System Operations - those functions necessary for basic operation, such as user interface, top-level command sequence, memory management, setup, calibration, initialization, I/O communication, and error detection and servicing.

2. Positioning Functions - those functions necessary for object location, most of which provide data that supports velocity command formulation. These include the object data base, Julian date and sidereal time calculations, coordinate conversions, and error modelling, if appropriate.

3. Motor Control Functions - those functions directly associated with control commands to be sent to the hardware. Examples are slew control, set control, and track control. Slew control can be further designated as manual or programmed slew. Manual slew can be thought of as a real-time user command to simply slew in a given direction, as in the case of a manual search. Programmed slew, on the other hand, is a routine by which the control system automatically moves to a given set of coordinates. The set control is no more than a fine manual slew, except that set velocities must be added or subtracted from the track rate. Track control involves calculating the required azimuth and altitude track velocities for the current pointing position, and scaling that value appropriately for output.

Words in all capital letters in this chapter generally refer to either user-defined commands, commonly-used FORTRAN, BASIC, or FORTH commands, or variables. In general, these should be distinguishable, based on the context in which they are used. However, for additional clarification, some of the more significant variables are defined in the next section.

Variables

It is assumed that the computer operating system has a memory location where the current date and local time is stored, and that can be read by the control program. It is also assumed that this data is correct before entering the program at startup. A DEFVAR command must be executed early to define all variables and arrays, depending on the programming language used. In addition, Table 11 shows some of the variables the software will be required to assign for use during control system operation. This is not intended to be a complete list, as additional variables will also be needed for intermediate calculations. Further, a large array will be need to be defined for the object data base discussed later.

Suggested variable names are used for reference later in this chapter. The table also indicates the minimum number of significant digits required. User interface variables, in most cases, will be defined as real variables, since many are floating point values. Most user variables should be accurate to at least four or five significant figures, exceptions as indicated. Hardware interface variables,

normally one byte each, must be easily accessible in binary format. These can all be defined as integer variables.

Table 11

Variables and Descriptions

Variable	Description	Type	Char
TIMEZ	Time Zone (1-24)	User	2
DAYSV	Daylight Savings (yes/no)	User	1
JDAY	Julian Date (Days)	User	8
SIDT	Sidereal Time (HMS)	User	6
LAT	Observer Latitude (Degrees)	User	4
LON	Observer Longitude (Degrees)	User	4
AZAX	Current Azimuth (Raw Sensor Position)	HW	8 Bits
ALAX	Current Altitude (Raw Sensor Position)	HW	8 Bits
AZMO	Current Azimuth Motor Position (Raw)	HW	8 Bits
ALMO	Current Altitude Motor Position (Raw)	HW	8 Bits
AZAXD	Azimuth Direction Bit (Feedback)	HW	1 Bit
ALAXD	Altitude Direction Bit (Feedback)	HW	1 Bit
AZMOD	Azimuth Motor Dir. Bit (Feedback)	HW	1 Bit
ALMOD	Altitude Motor Dir. Bit (Feedback)	HW	1 Bit
AZEXB	Current Azimuth (Extra Bits)	HW	3 Bits
ALEXB	Current Altitude (Extra Bit)	HW	1 Bit
MISCI	Miscellaneous Inputs (Direction and Extra Position Bits)	HW	8 Bits

Table 11 - Continued

Variable	Description	Type	Char
AZ	Current Azimuth (Degrees)	User	5
AL	Current Altitude (Degrees)	User	5
RA	Current Right Ascension (Degrees)	User	5
DEC	Current Declination (Degrees)	User	5
AZD	Destination Azimuth (Degrees)	User	5
ALD	Destination Altitude (Degrees)	User	5
RAD	Destination Right Ascension (Degrees)	User	5
DECD	Destination Declination (Degrees)	User	5
AZOF	Azimuth Offset Error (Degrees)	User	4
ALOF	Altitude Offset Error (Degrees)	User	4
AZV	Azimuth Absolute Velocity (Deg/Deg)	User	8
ALV	Altitude Absolute Velocity (Deg/Deg)	User	8
AZVEL	Azimuth Scaled Velocity Bits	HW	8 Bits
ALVEL	Altitude Scaled Velocity Bits	HW	8 Bits
AZSCL	Azimuth Velocity Scale Bits	HW	2 Bits
ALSCL	Altitude Velocity Scale Bits	HW	2 Bits
AZDIR	Azimuth Direction Bit	HW	1 Bit
ALDIR	Altitude Direction Bit	HW	1 Bit
MISCO	Miscellaneous Outputs (Scales and Directions)	HW	8 Bits
ADDR	Hardware Communication Channel Address	HW	3 Bits

User Interface/Operational Concept

The user interface (Command: INTERF) should involve coding to set up a command console display to present the current position, velocity, and direction status of each motor and axis in both equatorial and altitude-azimuth coordinates. It might also display the current date, Julian date, local and sidereal time. It should also present a keypad map, showing the user how to provide command inputs. Finally, it must present a listing of known objects in the object data base, and a method of selection.

The first step would be to switch to a fixed-screen mode, rather than scrolling, such as is done with a VT100 emulator or similar software mechanism. After clearing the screen, graphics-mode software would take over to draw a console display, dividing the screen into appropriate sections, featuring a designated location for each function. During normal operation, all displayed data is updated directly from memory.

The user will have access to output commands based on mode of operation. At a minimum, there should be a startup/initialization mode (Command: START), data base access (DATAACC) mode, data base maintenance (DBMAINT) mode, NORMAL mode, and SHUTDOWN mode. In the startup mode, the user is given the opportunity to input time zone, daylight savings, latitude, and longitude. The user is then instructed to perform a series of mechanical adjustments and prompted to switch on the control hardware, but leave off the switch to energize the motor windings. Next the program activates the

initialize bit to reset all control hardware counters and registers. The user then performs hardware voltage calibration, and turns on the motor windings. The motors are then slewed until each axis is in its home position. When initial calibration is complete, the program informs the user and switch back to the TOP-LEVEL mode. The alignment process, however, is still not complete.

For the first time in the data base access mode (Command: DATAACC to execute) the user is provided with a small list of bright stars suggested for alignment. The user then selects an object from the list. Otherwise, the user selects the desired object for observation, or inputs his own desired coordinates. The program then switches to the NORMAL mode.

As previously mentioned, a DBMAINT mode is also necessary to allow the user to enter new data into the object data base. The program should offer this option to the user when initialization is complete.

In the NORMAL mode, the program executes the slew, set, and track commands. It begins by slewing the telescope to the calculated position. However, the first time, this position would likely be several degrees off from the actual object position. The user is then be prompted to manually slew to the object. When the object is in the field of view, the user executes the set command to center the object. When this has been accomplished, the user acknowledges, and a position error value is generated, which is used as an offset for the next programmed slew and updated each time additional manual

slew alignment is required. The program then checks to ensure tracking is not in the prohibited zone. If so, the user is informed and instructed to find another object or try again later. Otherwise, the track command is automatically executed, and continues until either an error condition occurs (i.e., if the track reaches a prohibited zone) or the user enters another command.

The shutdown (Command: SHUTDOWN) mode is fairly simple. When executed, the program completes any programmed slew currently in progress, resets all registers to stop motors, and instructs the user to power down the motor windings first, then the rest of the control hardware. When user acknowledges, the program quits.

System Hardware Powerup, Calibration, And Initialization

While this section deals largely with hardware, it is included as part of the software development, since it is software-interactive. When the control system software is loaded and compiled, and the command START is executed, the program would first prompt the user to input time and location parameters. It should then instruct the user to perform the following electrical/mechanical setup procedure before applying any power to the hardware configuration:

1. Ensure that the telescope is properly mechanically and electrically configured. Be sure it is level, properly, and securely connected to all control hardware, and that nothing obstructs or inhibits rotation of either axis,

including short wires that may wrap around or catch on part of the mount during rotation.

2. Ensure that both axes are set in a reasonable initial calibration position. The altitude axis position sensor should be itself positioned to ensure its zero position closely coincides with the tube pointing position at or near, but not below the horizon. The azimuth axis position sensor should be positioned to ensure its zero position closely coincides with the tube pointing position at or near due South, but not East of South. It is assumed that gearing has been implemented such that clockwise motor rotation produces azimuth axis rotation from South to West, and a rising altitude rotation.

3. Once the axis position sensors have been set, place the azimuth tube position within a few degrees East of due South, while the initial altitude position should be pointed just below the horizon. This will ensure that the zero sensor positions will be encountered shortly after a positive (clockwise) calibration rotation is initiated. Some type of manually-read angular detection device should also be mounted at the axes to aid in rough calibration. A surveyor's compass and inclinometer could be used, but a pair of protractors will suffice, as precise calculation can be achieved later by stellar alignment.

Once the user has acknowledged these actions have been accomplished, the program instructs him to apply power to the control hardware, but not the separately-switched motor windings. When the user acknowledges this, the R/W Enable bit is reset (R/W EN = 0), thereby isolating all registers (latches and flip flops) from the port to avoid bad motor commands, conflicting outputs on the port, and general data miscommunication. The system is then initialized by strobing the Initialization Bit HIGH. This action clears the counters in the Microstep Generation Circuit, so that microstepping begins from an electrical phase angle of 0°. It also clears the position detection counters in the feedback circuits, as well as the frequency division counters, although the latter is not necessary.

Most importantly, initialization will reset all three output registers to 00000000. This ensures that both motors will come up in the OFF position of the velocity scale selection multiplexer. Further, it ensures a predictable initial value at the output of the D/A converter, and a very low initial pulse rate (i.e., low velocity) when the motors are turned on before the first velocity update.

When registers have been cleared, the user is instructed to calibrate all adjustable voltages with an accurate voltage reference. At this point, power can applied to the motor windings.

In the initial position calibration procedure, each motor is ramped up to speed in the slew mode in a clockwise direction until it reaches zero axis sensor position as read on the appropriate input

channel. The normal I/O control communication sequence is described in the next section. When each motor reaches its calibration position, it is immediately decelerated to a stop and reversed at a slower speed until it comes to rest in this position.

At this point, initial calibration is complete, but position errors on the order of a few degrees are still present. Stellar alignment is still necessary to achieve the required precision. This is accomplished in the normal operating mode by performing one or two programmed slews to known stars. In each case, the control system slews to the calculated altazimuth position and stops. The user is then prompted to manually slew to the star's position, then center it using the set command, and the position error values are recorded and stored.

I/O Communication

Recall the complexity of the routing network hardware design described in Chapter 5. It is now necessary to design the software to "talk to" each of the I/O registers. The following series of commands (Table 12) would be followed as a standard data communication routine during normal operation. For later reference, Command IOCOMM will be designated as the command to execute this routine. The intent is to cycle through this sequence of sending velocity and direction output commands and reading position and direction sensor inputs, followed by a period of checking for user or error interrupts, performing calculations and comparisons, decision-making, and output command formulation. These commands are written in no

specific language, but should be easily interpreted by a programmer for implementation in most reasonable "I/O-friendly" programming languages on most any system that meets the guidelines as defined in Chapter 5. Entries in italics do not represent variable references, but values only. DDD refers to the data direction designator, normally a memory location of eight bits designated for the purpose of specifying whether each port line is configured for input of output. Where DDD = 1, the actual value entered is 11111111, since all eight lines will be simultaneously designated as outputs. The same is true for DDD = 0.

Since the MISCO (Miscellaneous Outputs) and MISCI (Miscellaneous Inputs) variables each contain several pieces of data carried in bits as specified in Chapter 5, the MISCOASS and MISCIDIS commands are intended to combine and extract this data, respectively. Some simple routines to achieve this are shown below. The logical ANDs are used to select the appropriate bits, while the multiplication and division is used to shift bits left and right, respectively.

MISCOASS:

```
MISCO = ( AZSCL AND 3 ) + ( ( ALSCL * 4 ) AND 12 )
+ ( ( AZDIR * 16 ) AND 16 ) + ( ( ALDIR * 32 ) AND 32 )
```

MISCIDIS:

```
AZMOD = MISCI AND 1
ALMOD = ( MISCI AND 2 ) / 2
```

AZAXD = (MISCI AND 4) / 4

ALAXD = (MISCI AND 8) / 8

AZEXB = (MISCI AND 112) / 16

ALEXB = (MISCI AND 128) / 128

The AZUPDATE and ALUPDATE commands are used to convert the raw encoder position data into actual angular position information in degrees/decimal degrees. The following are possible

Table 12

I/O Communication (IOCOMM) Command Sequence

COMMAND	PURPOSE	PORT CONNECT.	PORT CONTENTS
DDD = 1	Configure for Output	None	XXXXXXXX
MISCOASS	Assemble Misc. Output Data	None	XXXXXXXX
PORT=*MISCO*	Misc. Output to Port	None	*MISCO*
ADDR = 000	Select Misc. Output Channel	None	*MISCO*
R/W En = 1	Write *MISCO* into Flip Flop 0	Chan 0	*MISCO*
R/W En = 0	Isolate Registers	None	*MISCO*
PORT=*AZVEL*	Azimuth Velocity to Port	None	*AZVEL*
ADDR = 001	Select Azimuth Vel. Channel	None	*AZVEL*
R/W En = 1	Write *AZVEL* into Flip Flop 1	Chan 1	*AZVEL*
R/W En = 0	Isolate Registers	None	*AZVEL*
PORT=*ALVEL*	Altitude Velocity to Port	None	*ALVEL*
ADDR = 010	Select Altitude Vel. Channel	None	*ALVEL*
R/W En = 1	Write *ALVEL* to Flip Flop 2	Chan 2	*ALVEL*

Table 12 - Continued

COMMAND	PURPOSE	PORT CONNECT.	PORT CONTENTS
R/W En = 0	Isolate Registers	None	*ALVEL*
DDD = 0	Configure for Input	None	XXXXXXXX
ADDR = 011	Select Misc. Inputs Channel	None	XXXXXXXX
R/W En = 1	Read *MISCI* from Latch 3	Chan 3	*MISCI*
READ PORT	Read *MISCI* into CPU	Chan 3	*MISCI*
MISCI=*MISCI*	Store Value into Var. MISCI	Chan 3	*MISCI*
R/W En = 0	Isolate Registers	None	XXXXXXXX
MISCIDIS	Disassemble Misc. Input Data	None	XXXXXXXX
ADDR = 100	Select Az Axis Pos Channel	None	XXXXXXXX
R/W En = 1	Read *AZAX* from Latch 4	Chan 4	*AZAX*
READ PORT	Read *AZAX* into CPU	Chan 4	*AZAX*
AZAX=*AZAX*	Store Value into Var. AZAX	Chan 4	*AZAX*
R/W En = 0	Isolate Registers	None	XXXXXXXX
ADDR = 101	Select Az Motor Pos Channel	None	XXXXXXXX
R/W En = 1	Read *AZMO* from Latch 5	Chan 5	*AZMO*
READ PORT	Read *AZMO* into CPU	Chan 5	*AZMO*
AZMO=*AZMO*	Store Value into Var. AZMO	Chan 5	*AZMO*
R/W En = 0	Isolate Registers	None	XXXXXXXX
(REPEAT LAST 5 STEPS FOR ALTITUDE DATA READ - LATCHES 6, 7)			
AZUPDATE	Convert AZAX to AZ (360°)	None	XXXXXXXX
ALUPDATE	Convert ALAX to AL (90°)	None	XXXXXXXX

software implementations of these. The INT function is used to place the azimuth measurement in the range of $-180 \leq AZ \leq 180$.

AZUPDATE:

$$AZ360 = 45 * AZEXB + 0.1758 * AZAX$$
$$AZ = AZ360 - 360 * INT (AZ360 / 180)$$

ALUPDATE:

$$AL = 45 * ALEXB + 0.1758 * ALAX$$

Error Detection and Servicing

At a minimum, this software (Command: ERRORDET) should perform the following error detecting and servicing functions:

1. It should detect any slew that stops at an altitude greater than or equal to 85°, inhibit tracking, and inform the user. It may also be preferable to detect rising slews ending at altitudes between 80° and 85°, and issue a warning to the user that tracking may be halted within 20 minutes. It should also detect the point at which the altitude reaches 85° while tracking, halt tracking at that point, and inform the user.

2. The error detecting routine should also make similar decisions when approaching the horizon. It should halt all tracking and slewing at the horizon, however. The only exception should be at startup when the altitude

axis should be pointed slightly below the horizon. Like the zenith approach, a warning should be issued when the altitude falls below 5°.

3. Similarly, tracking must also be prohibited within 1° of the celestial pole. However, a warning is not necessary here, since by definition, tracking involves movement at constant declination. Therefore, any track beginning outside of this region will stay outside of it.

4. The error routine should also monitor the count of motor encoder steps and compare it with its previously measured value. The delta should then be divided by the time elapsed between measurements and compared with the commanded velocity. If these do not agree, missed or extra steps may have occurred, which the user should be aware of.

5. The measured directions of all encoders should be compared with the commanded direction. If these are found in disagreement, slippage may be occurring, which should also be reported.

Other error conditions could also be monitored. In most cases, coding for these routines is straightforward, involving extracting parameters from two different memory locations, subtracting or otherwise comparing them, and executing some form of interrupt command.

Object Data Base

The object data base is set up by defining a large array. An initial list of some 200 objects would be preferred, expandable to 1024. Values are all stored in hexadecimal format. Each object entry should contain, at a minimum, a 1-up catalog number for reference and indexing (3 bytes), a New General Catalog (NGC) number (4 bytes), Messier No. (if applicable, 2 bytes), 8-character object type description (8 bytes), 16-character object nickname or "also known as" description (16 bytes), followed by its coordinates in right ascension in hours, minutes, and tenths of minutes (5 bytes), and declination in (positive/negative) degrees and minutes (4 bytes). This comes to a total of 42K bytes required.

When DATAACC is executed, the program requests that the user select an object. However, the program could also be expanded and developed with complex routines to search on any of the fields to find the desired object in a variety of ways. In addition, searches could be performed to select smaller lists of objects given boundaries on coordinates.

Julian Date Determination

A command JDAYCAL would be developed to determine the current Julian day number. Given the current date stored in memory by the operating system, the Julian date (Variable: JDAY) can be derived by using the algorithms provided in Chapter 3. The

programmer should be certain to calculate and include the Gregorian offset value.

Sidereal Time Determination

A command SIDTCAL would also be necessary to determine the current sidereal time (Variable: SIDT). Given the current time stored in memory by the operating system and other parameters, the sidereal time can be derived by using the algorithms provided in Chapter 3. Several other intermediate values must be calculated first, however. Among them are the time in Julian centuries, T, and the Greenwich sidereal time at O^h UT, G_0. In addition, the current Universal Time, U, must be determined, taking daylight savings into account, which, multiplied by the sidereal/solar proportionality factor, can be combined with G_0 to produce the current sidereal time at Greenwich, G. The last step is to use the observing longitude, expressed in time units, to convert G to the local sidereal time, S. This procedure was shown in flow chart form in Figure 8, back in Chapter 3.

Equatorial to Altazimuth Conversion Module

One of the key mathematical algorithms (Command: EACONV) is the conversion module to translate a set of equatorial coordinates (right ascension and declination), also given the current sidereal time (SIDT), and observing latitude (LAT), the equivalent azimuth and altitude coordinates can be determined by the coordinate conversion algorithms derived in Chapter 3:

$$A = \tan^{-1}\left(\frac{\sin H}{\cos H \sin \phi - \tan \delta \cos \phi}\right)$$

$$E = \sin^{-1}\left(\cos H \cos \delta \cos \phi + \sin \delta \sin \phi\right)$$

However, care must be taken to ensure the azimuth calculation produces the right quadrant. This is perhaps best achieved by computing numerator and denominator separately. Defining two new variables as

$$NA = \sin H$$

$$DA = \cos H \sin \phi - \tan \delta \cos \phi$$

Therefore, the proper azimuth can be computed by the following expression:

$$A = \tan^{-1}\left(\frac{NA}{DA}\right) + 180 * \text{LOGICAL}\,(DA < 0?) * \left(\frac{NA}{|NA|}\right)$$

The second term provides quadrature correction by adding 180° if the denominator, DA, is negative and the numerator, NA is positive. However, if both are negative, the quadrature correction is achieved by subtracting 180°. The LOGICAL portion of the expression can be achieved with an IF/THEN statement. This expression will always produce an azimuth in the range of $-180° \leq A \leq 180°$.

To ensure input values are in a manageable format, a command, HMCON, would be a few lines of code to convert right ascension from hours/minutes to degrees/decimal degrees.

Similarly, a command, DMCON, must be defined to convert declination from degrees/minutes to degrees/decimal degrees.

Altazimuth to Equatorial Conversion Module

In some cases, the command (AECONV) will also be necessary to convert altazimuth coordinates to equatorial. Again, from Chapter 3, the expressions are:

$$\delta = \sin^{-1}\left(\sin E \sin \phi - \cos H \cos E \cos \phi\right)$$

$$H = \tan^{-1}\left(\frac{\sin A}{\cos A \sin \phi + \tan E \cos \phi}\right)$$

To preserve quadrature for hour angle,

$$NH = \sin A$$

$$DH = \cos A \sin \phi - \tan E \cos \phi$$

Therefore, the proper hour angle is determined as

$$H = \tan^{-1}\left(\frac{NH}{DH}\right) + 180 * \text{LOGICAL}\,(DH < 0?) * \left(\frac{NH}{|NH|}\right)$$

Velocity Calculations

In Chapter 4, the following velocity equations were derived for each axis:

$$v_{t_A} = \frac{k\left(b - c \cos\left(a + kU\right)\right)}{\left(b \cos\left(a + kU\right) - c\right)^2 + \sin^2\left(a + kU\right)}$$

$$v_{t_E} = \frac{-kd \sin (a + kU)}{\sqrt{1 - (d \cos (a + kU) + e)^2}}$$

where all input quantities, including time (U), are expressed as an angle in degrees (°), while the output quantities are expressed in °/deg. The constants, a through e, were used to represent the following quantities:

$$a = S - \alpha$$

$$b = \sin \phi$$

$$c = \tan \delta \cos \phi$$

$$d = \cos \delta \cos \phi$$

$$e = \sin \delta \sin \phi.$$

Given a set of equatorial coordinates and other input parameters, the VELCAL command should be defined to calculate the quantities a, b, c, d, and e. It should not be necessary to check for the zenith error condition where $\phi = \delta$ if $a + kU = 0$, since a zenith check would have already been done in the TRACK routine. The VELCAL program would then determine the instantaneous track velocities for each axis in °/deg.

The actual input variables used by VELCAL would be SIDT, RA, DEC, LAT, UNIT (universal time), and K (the sidereal/solar proportionality factor). VELCAL should first convert these all to units of degrees/decimal degrees. The output variables from this module would be AZV and ALV.

Velocity Conversions

Given a set of track velocities, AZV and ALV, as determined by VELCAL, a VELCONV command is needed to convert these velocities into the properly formatted velocity commands that would be sent to the hardware portion of the control system. Essentially, this routine must convert these velocities into scaled integer values and select the appropriate track scale, Track-1 or Track-2. Since AZV and ALV are expressed in units of °/deg, an equivalent Track-2 command for AZV would be determined by the following formula:

$$\text{AZVT2} = \text{AZV (°/deg)} \left(\frac{3600\ ''/°}{240\ \text{sec/deg}}\right)\left(\frac{1\ \text{step}}{1.125''}\right)\left(\frac{255\ \text{counts}}{125.7\ \text{Hz}}\right)$$

Multiplying out all these factors produces:

$$\text{AZVT2} = 27.057\ \text{AZV}$$

If the value produced is less then the ratio, 255.5/256 (i.e., 0.998 or less), it should be multiplied by 256 and scaled on Track-1. The final number to be used as the velocity command must be a rounded integer value from 1 to 255. A reasonable routine for this would be as indicated below. A warning has been added in case an excessive track velocity is demanded; however, this should never occur in altitude, and should only occur in azimuth when grazing the Southern edge of the zenith prohibited zone. Similarly, a warning is issued if the calculated track falls below the minimum velocity, but this should only occur in azimuth at zenith or in altitude within one degree of the celestial pole.

VELCONV:

```
IF  AZV < 0  THEN  AZDIR = 0  ELSE  AZDIR = 1  ENDIF
AZVS = ABS ( 27.075 * AZV )
IF  ( AZVS ≤ 0.998 )
THEN  AZSCL = 1  ,  AZVS = 256 * AZVS
IF  AZVS < 1
PRINT  "WARNING:  AZIMUTH TRACK VELOCITY IS
        BELOW THE MINIMUM ALLOWABLE.  TRACK MAY
        NOT BE MAINTAINED."
ELSE  AZSCL = 2  ENDIF
AZVEL = INT ( AZVS ) + INT ( ( AZVS - INT ( AZVS ) ) * 2 )
IF  AZVEL > 255
THEN  AZVEL = 255
PRINT  "WARNING:  AZIMUTH TRACK VELOCITY EXCEEDS
        MAXIMUM ALLOWABLE.  TRACK MAY NOT BE
        MAINTAINED."
ENDIF
IF  ALV < 0  THEN  ALDIR = 0  ELSE  ALDIR = 1  ENDIF
ALVS = ABS ( 27.075 * ALV )
IF  ( ALVS ≤ 0.998 )
THEN  ALSCL = 1  ,  ALVS = 256 * ALVS
IF  AZVS < 1
PRINT  "WARNING:  ALTITUDE TRACK VELOCITY IS
        BELOW THE MINIMUM ALLOWABLE.  TRACK MAY
        NOT BE MAINTAINED."
ELSE  ALSCL = 2  ENDIF
```

```
ALVEL = INT ( ALVS ) +
        INT ( ( ALVS - INT ( ALVS ) ) * 2 )
IF  ALVEL > 255
THEN  ALVEL = 255
PRINT "WARNING: ALTITUDE TRACK VELOCITY EXCEEDS
        MAXIMUM ALLOWABLE. TRACK MAY NOT BE
        MAINTAINED."
ENDIF
```

Error Modelling Possibilities

As discussed earlier, certain sources of error can be modeled, if necessary. Several examples are provided in Chapter 4, particularly in regard to mounting errors. If operation reveals that these issues lead to unacceptable position errors that will not calibrate out, the programmer is referred to Chapter 4 to develop appropriate error models for compensation.

Slew Control

As previously discussed, three velocity control modes are required: slew, set, and track. Slew control is unique in two ways. First of all, it takes place at speeds requiring some kind of ramping profile to allow the steppers to stay in synchronization with the pulses. Secondly, two types of slew are necessary, programmed and manual. Programmed slew involves moving to a specific location, defined by a set of input coordinates, normally equatorial, by periodically monitoring the current position and comparing it with

the coordinates of the desired destination until they coincide, all under software control. Manual slew is essentially an open-loop motion, whereby the user deliberately initiates and terminates rotation in real time, perhaps by holding down a key which controls a given motor in a specified direction.

The ramping requirement will be dealt with first. From Chapter 4, the decision was made to require each motor to reach a maximum speed of five revolutions per second (1000 primary steps per second) in one second. It may be recalled that this converts to a maximum pulse repetition frequency of 16,000 pps. Since the maximum required slew speed occurs at a velocity input setting of 10000000_2 (128_{10}), or half the maximum digital control capability, the pulse repetition frequency will rise in 128 increments during the one-second interval.

Assuming a trapezoidal acceleration profile, a linear ramp will produce 128 intervals of 8 msec in 1.024 sec. Upon reaching the maximum pulse repetition frequency of 16 kpps, each 8 msec interval should contain 128 pulses. Essentially, then, for a linear ramp, the first interval should contain one pulse, the second two, the third three, etc. This is really the case on the average, however, as the pulse generation circuit is asynchronous. Approximately one second later, as the motor has reached its maximum slew speed, it has moved over 8000 microsteps (500 primary steps), which has moved the telescope 2.5°. If deceleration then occurs immediately at the same rate, the axis has moved 5° before coming to rest.

It is therefore apparent that a triangular profile will also be required for short slews, assuming a single acceleration/deceleration rate. Consequently, before slew motion begins, the slew routine should identify the angular distance to be moved by subtracting the current position from the destination position. If this difference is more than 5°, the routine will ramp the motor up to maximum speed, and maintain that speed until within 2.5° from the destination, at which time deceleration commences at the constant rate of $5°/sec^2$, stopping one second later. If the distance to travel is less than 5°, the ramp routine should accelerate the motor until half the distance is covered, and begin deceleration.

When a programmed slew (Command: PROGSLEW) is executed, the routine must permit the user to select an object from the data base or input his own coordinates. In either case, the routine must then convert the destination position to altazimuth coordinates. It then checks the current position and determines the point at which it must begin decelerating to stop at the proper position. When ready, the ramping slew motion is performed until the destination has been reached or an interrupt occurs. Such a routine might appear as that shown below.

```
PROGSLEW:
        PRINT "SELECT OBJECT FROM DATA BASE?"
        < User answers "Y" or "N" >
        IF CHAR = Y THEN DATAACC , "SELECT OBJECT: ___"
        < User selects object XXX and ENTER >
        RAD = RAXXX , DECD = DECXXX
```

```
ELSE "ENTER EQUATORIAL COORDINATES:
        RA: ____ HRS ____ MIN    DEC: ____ DEG ____ MIN"
RAD = RAUSER , DECD = DECUSER ENDIF
RAD = HMCON(RAD) , DECD = DMCON(DECD)
SIDTCAL
EACONV (RAD, DECD)
< Altazimuth parameters stored in AZD, ALD >
DELAZ = AZD - AZ + AZOF , DELAL = ALD - AL + ALOF
IF DELAZ ≥ 0 THEN AZDIR = 1 ELSE AZDIR = 0 ENDIF
IF DELAL ≥ 0 THEN ALDIR = 1 ELSE ALDIR = 0 ENDIF
AZSCL = 3 , ALSCL = 3
IF ABS(DELAZ) ≤ 5 THEN MIDAZ = AZ + DELAZ/2
ELSE MIDAZ = AZD - 2.5 ENDIF
IF ABS(DELAL) ≤ 5 THEN MIDAL = AL + DELAL/2
ELSE MIDAL = ALD - 2.5 ENDIF
AZVEL = 1 , ALVEL = 1 , AZINC = 1 , ALINC = 1
BEGIN IOCOMM DELAY
< Delay should be adjusted for 8 msec cycle time >
IF AZ ≥ MIDAZ THEN AZINC = -1 ENDIF
IF AL ≥ MIDAL THEN ALINC = -1 ENDIF
IF AZVEL = 0 OR IF AZVEL = 128
THEN AZINC = 0 ENDIF
IF ALVEL = 0 OR IF ALVEL = 128
THEN ALINC = 0 ENDIF
AZVEL = AZVEL + AZINC , ALVEL = ALVEL + ALINC
UNTIL ( AZVEL = 0 AND ALVEL = 0 ) OR
USERINT = 1 OR ERRORINT = 1
```

AZVEL = 0 , ALVEL = 0

IOCOMM

In manual slew operation, the user will simply have to release the slew key early, and may need to reverse slew to stop at the intended destination. If the minimum manual slew duration (time to depress and release key) is assumed to be 0.1 seconds, the minimum manual slew is then approximately 25 8-msec periods, 13 up and 12 down, in which time the motor has moved approximately 169 microsteps (10.6 steps), moving the axis by 3.2′. This very short incremental slew should be capable of bringing the object well within setting range.

The MANSLEW command, described below, runs the manual slewing routine, which determines which of four possible keypad entries was made by the user, and continues to monitor the key status during the slew. The routine is designed so that pressing the "4" key will move counterclockwise in azimuth, the "6" key will move clockwise in azimuth, the "2" key will cause the tube to rise in altitude, while the "8" key will cause the tube to drop in altitude. If azimuth motion is commanded, the top ten lines are active, whereas the next ten are active if altitude motion is commanded. The entire routine is incorporated in the highest level of nested IF/THEN statements, except for the last two lines. The KEYREL? command is a generic one for any language-dependent command or sequence of commands to detect that a key is no longer depressed.

MANSLEW (Executed only when user touches keys 2, 4, 6, or 8):

```
AZREC = AZ , ALREC = AL
IF CHAR = 4 OR IF CHAR = 6
THEN AZSCL = 3 , AZVEL = 0 , AZINC = 1
IF CHAR = 4 THEN AZDIR = 0 ELSE AZDIR = 1 ENDIF
BEGIN
IF AZVEL = 128 THEN AZINC = 0 ENDIF
AZVEL = AZVEL + AZINC
IOCOMM DELAY
< Delay should be adjusted for 8 msec cycle time >
IF KEYREL? THEN AZINC = -1 ENDIF
UNTIL AZVEL = 0 OR ERRORINT = 1
ELSE ALSCL = 3 , ALVEL = 0 , ALINC = 1
IF CHAR = 2 THEN ALDIR = 0 ELSE ALDIR = 1 ENDIF
BEGIN
IF ALVEL = 128 THEN ALINC = 0 ENDIF
ALVEL = ALVEL + ALINC
IOCOMM DELAY
< Delay should be adjusted for 8 msec cycle time >
IF KEYREL? THEN ALINC = -1 ENDIF
UNTIL ALVEL = 0 OR ERRORINT = 1
ENDIF
AZVEL = 0 , ALVEL = 0
IOCOMM
SIDTCAL
AECONV (AZ, AL)
```

< Equatorial parameters stored in RA, DEC >

PRINT "DO YOU WISH TO STORE THIS MOVE AS AN ERROR OFFSET?"

< User responds Y or N >

IF CHAR = Y

THEN AZOF = AZ - AZREC , ALOF = AL - ALREC ENDIF

Set Control

The set control (Command: SET) is needed to function essentially as a fine manual slew. Because the setting speed is much slower than the maximum slewing speed, ramping will not be required here. However, because the set speed of 5′/sec is only twice the maximum azimuth track velocity, the set velocity command used should be that required relative to track velocity. Otherwise, if operating near the maximum azimuth track velocity region (Southern edge of the zenith prohibited zone), the apparent motion of the target object is fast enough to significantly reduce the apparent (practical) set velocity.

Ideally, the set velocity command should be formulated by adding the nominal or relative set velocity to the track velocity. However, the absolute set velocity of 5′/sec for a zero track speed would require a command of 00000010_2 (2_{10}) on the slew velocity scale. This compares with a maximum azimuth track velocity command of 11111111_2 (255_{10}) on the Track-2 (fast track) scale, which is approximately the same speed as 00000001_2 (1_{10}) on the slew velocity scale. Further, all track velocity commands are

generated in the two track velocity ranges and only eight bits are available with no scale overlap. Since the control system for each axis is only capable of operating in one velocity range or the other, a compromise will be needed: If the running azimuth track velocity is at least half maximum, or 10000000_2 (128_{10}) on the Track-2 scale, then the set velocity command will be adjusted up to 00000011_2 (3_{10}) or down to 00000001_2 (1_{10}), depending on whether set and track motions are in the same or opposite directions, respectively. Since the maximum altitude track velocity is always small in comparison with the set velocity, no adjustment will be necessary for that axis.

Setting could be triggered by the same keys used for manual slew, but the program could be set up for the user to make the distinction by hitting the "0" key prior to initiating a setting motion. Both manual slew and set could function as interrupts to the tracking program. Therefore, whenever SET is executed, the current azimuth track velocity is available in AZVEL (magnitude) and AZSCL (scale). A short program to perform this function is provided below, again consisting of a nesting of IF/THEN statements to treat all possible combinations of track velocities/directions and keystrokes.

SET (Executed only when user touches keys 2, 4, 6, or 8 after 0):

```
IF ( CHAR = 4  OR  CHAR = 6 )
THEN
IF  AZSCL = 2  AND  IF ( AZVEL AND 128 ) = 128
THEN  AZSCL = 3
```

```
IF AZDIR = 0 THEN
IF CHAR = 4 THEN AZVEL = 3 , AZDIR = 0
ELSE AZVEL = 1 , AZDIR = 1 ENDIF
ELSE
IF CHAR = 4 THEN AZVEL = 1 , AZDIR = 0
ELSE AZVEL = 3 , AZDIR = 1 ENDIF
ENDIF
ELSE AZVEL = 2 , AZSCL = 3
IF CHAR = 4 THEN AZDIR = 0
ELSE AZDIR = 1 ENDIF
ENDIF
ELSE ALVEL = 2 , ALSCL = 3
IF CHAR = 2 THEN ALDIR = 0
ELSE ALDIR = 1 ENDIF
ENDIF
BEGIN IOCOMM KEYREL? UNTIL
```

Track Control

The tracking control software is required to check for error conditions and compute the track velocity and direction for each axis, based on the current position. It should also monitor motor position and direction feedback channels to ensure proper tracking. During the normal operating mode, this routine would run continuously unless interrupted by ERRORDET, or by the user to execute PROGSLEW, MANSLEW, OR SET.

The motor monitoring function (Command: MOTORFB, not listed) would extract the motor position feedback values and compare them with the values read during the previous cycle. It will also compute an estimate of the distance each of the motors should have travelled for comparison, and inform the user if the predicted and measured values do not agree. It should also compare the sensed motor and axis directions for comparison to those commanded.

Since most operations are executed either during tracking or as an interrupt to tracking, the tracking operation itself will be a large part of the NORMAL mode of operation. A possible implementation for the NORMAL command could be developed by stringing together various previously-defined functions, as listed below.

NORMAL:

```
AZVEL = 0 , ALVEL = 0
IOCOMM
BEGIN ERRORDET , SIDTCAL , AECONV (AZ, AL)
VELCAL , VELCONV , MOTORFB
UNTIL USERINT OR ERRORINT
IF ERRORINT = 1
THEN AZSCL = 0 , ALSCL = 0 , AZVEL = 0 , ALVEL = 0
PRINT "SYSTEM HALTED BY ERROR DETECTION"
ELSE IF CHAR = Q THEN QFLAG = 1
ELSE IF CHAR = 0 THEN SET
ELSE IF CHAR = 2 OR IF CHAR = 4 OR
```

```
        IF CHAR = 6 OR IF CHAR = 8
THEN MANSLEW
ELSE PROGSLEW ENDIF ENDIF ENDIF ENDIF
UNTIL QFLAG = 1
```

Top-Level Command Mode

When all previously-discussed coding is complete, the only remaining software task is the top-level command mode, which is the highest level program (Command: CONTROL) that is executed to perform all functions. The listing below provides such a routine.

CONTROL:

```
DEFVAR , INTERF , START ,
PRINT " DO YOU WISH TO MAKE ANY NEW
        DATA BASE ENTRIES?"
< User responds "Y" or "N" >
IF Y THEN DBMAINT ENDIF
PROGSLEW
< User selects first object >
NORMAL
QUIT
```

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

The original intent of this report was to examine the technical issues surrounding an experimental telescope control system, determine the technical requirements for the system, and design and develop a workable implementation of that system. The system was not originally intended to perform the tracking function to the extend necessary for astrophotography, largely due to the expected quantization errors inherent in the intended design, and also due to mechanical mount imperfections in the instrument proposed for control system retrofit. In conclusion, however, it is believed that many aspects of the design, modified and refined, could be directly applied to a much more precise system, especially provided the mechanical configuration is precisely machined and otherwise fabricated.

As stated in the preface, the idea for this project was independently originated, but not completely original. Some of the latest technology in the largest aperture telescopes either completed or in development in the last ten years, have been altazimuth designs. These include the Multiple-Mirror Telescope (MMT), the Keck telescope, and the Ohio State Two-Shooter telescope, to name a few. In addition, Mark Trueblood (66: 220) was working on a design

of a computer-controlled 30-inch trailer-mounted altazimuth instrument as early as 1985.

In Chapter 4, exhaustive analysis was performed to determine torque and velocity requirements for the system. Considerable data was generated and was, in many cases, presented in a unique format. Although not all items listed as requirements were really "hard" requirements, they were generally followed in the design.

In terms of the hardware design, several things could be done to improve performance. For one, the use of a non-classical feedback architecture would be highly beneficial. Such a mechanism would likely involve some type of optical feedback device specially interfaced with the optical path of the telescope with an off-axis guider or similar device to constantly monitor a guide star or other object to control tracking.

Another idea would be to add a second slower slew acceleration rate for short manual slews. In the current design, the user may be required to make several very short "choppy" manual slews to get within a reasonable setting range. A second slower acceleration rate would tend to allow the user a more convenient means of bridging the gap between a short slew and a long set.

A major problem that is discussed in detail in Chapter 4 is dealing with the issue of tracking through zenith. The only way to completely eliminate the problem would be to use a different type of mount. Aside from the obvious equatorial option, the possibility exists for an altitude-altitude mount, whereby the instrument is

rotated first in East-West altitude, then in North-South altitude. This is an altazimuth-type version of what is sometimes referred to as a cradle-, English-, or yoke-equatorial. Essentially, the altitude axis is fixed parallel to the horizon along a line running North and South. Two widely-spaced vertical supports are required to contain each of the two altitude bearings. Running between the bearings is a structure consisting of two or more rigid horizontal members containing the other set of bearings, with the tube mounted between them. The equivalent prohibited zone with this type of mount is only approaching the Northernmost and Southernmost points on the horizon, which is not of concern.

Finally, one important feature could be added which is a must for long-exposure astrophotography. This would be to provide a slowly-rotating camera mount to compensate for the rotating field problem discussed earlier.

A number of improvements could be made to the software design. This section was originally intended only as a preliminary or high-level design, but in several cases, detailed design was added down to coding level. Since the goal was not to perform actual coding, no particular programming language or protocol was consistently adhered to. There are many modern engineering programming languages that could be used, but to eliminate confusion, simple commands were used as much as possible. Commands were either user-defined or borrowed from any number of languages, including FORTRAN, BASIC, and FORTH. Further, the author recognizes that there may be various ways to more efficiently

implement the routines provided. Instead, the intent was to use simple coding practices to allow the programmer to read, understand, and reproduce the code in a more efficient manner.

APPENDIX A

TELESCOPE DESCRIPTION BY SUBASSEMBLY

The following diagrams provide a detailed description of each major subassembly of the telescope to be retrofited with the digital control system described in this paper. This information is used extensively in Appendix B to determine moments of inertia about each of the rotational axes (altitude and azimuth). Essentially, the telescope, without the control system hardware, is described below as broken down into eight major subassemblies: tube (including focuser mounting plates), tube box, secondary mirror/holder/spider, finder (including mount), focuser/eyepiece, mirror/mirror cell, rocker box, and base plate.

Tube

The tube is solid polyvinyl chloride (PVC) pipe (see Figure 36), slightly flattened, with oval cut-out at one end. Inlaid around the cut-out are two rectangular aluminum plates stacked vertically. Each plate is solid aluminum, 4" × 9" × 1/4" thick. The lower plate has an oval cutout that measures 2" × 5". The upper plate has a 1 1/2" diameter hole, centered 5" from the end of the plate toward the front of the tube. With the tube pointing horizontally, the plates are slanted at 37° from the horizontal as shown in the figure.

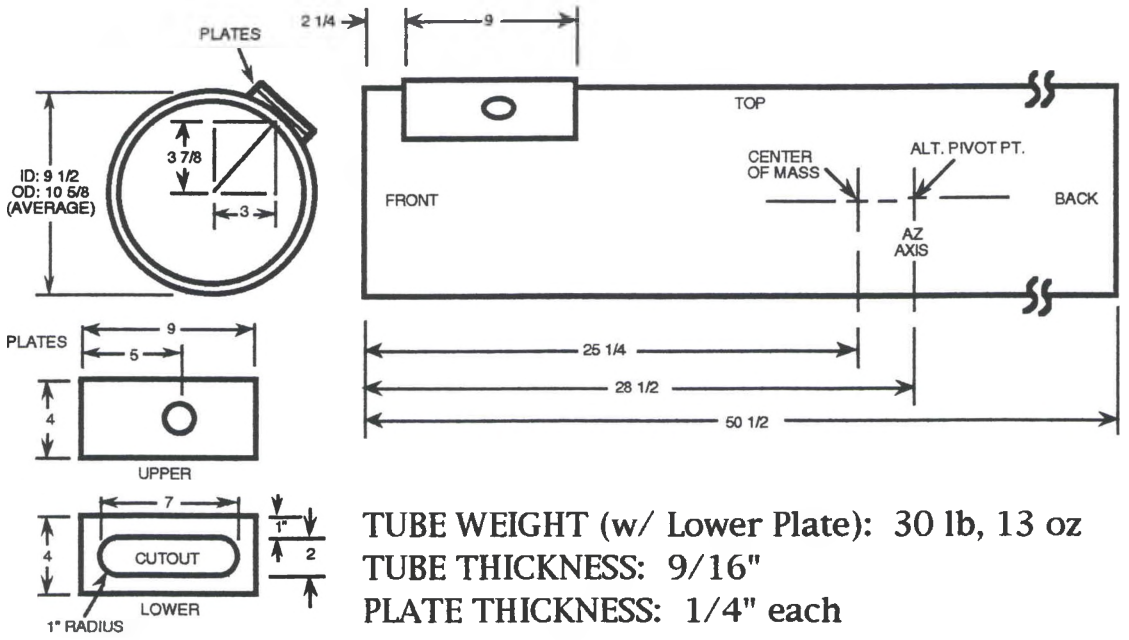


Figure 36

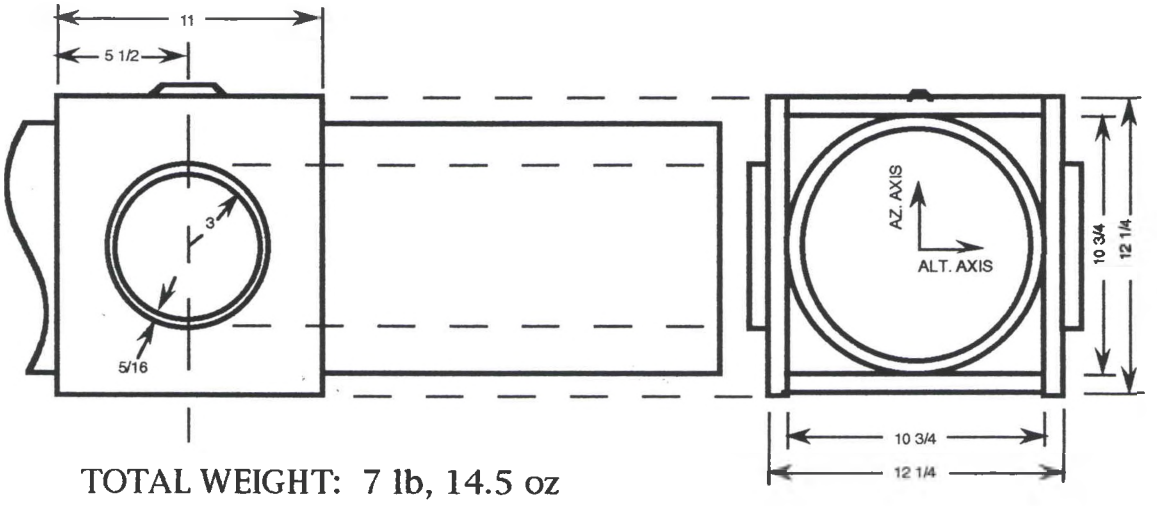
Mechanical Description of Tube

Tube Box

The tube box (see Figure 37) is a square box constructed from four pieces of 3/4" plywood, which surround the tube, attached with wood screws and right-angle joint connectors. A light metal handle is attached at the top. An altitude bearing is attached to each side. The bearings are rings cut from PVC pipe, each mounted on a 6" diameter plywood (3/4" thick) disk.

Secondary Mirror/Holder/Spider

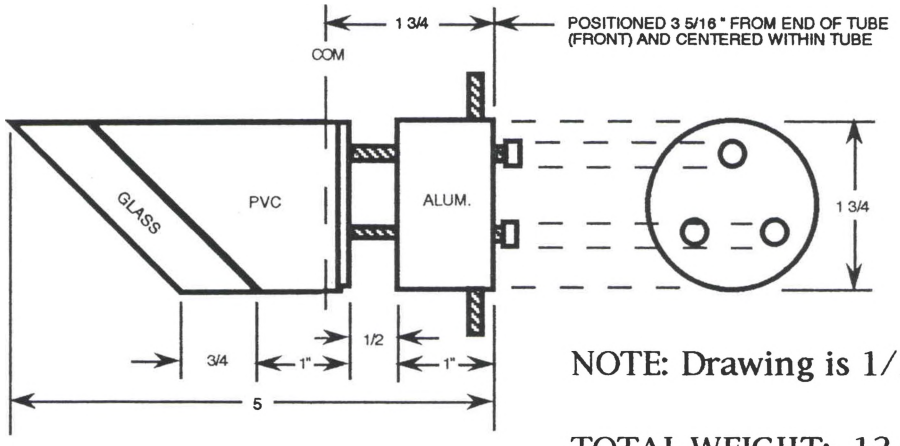
The glass mirror is mounted on a cylinder of PVC (see Figure 38) , in turn, attached to an aluminum cylinder with three adjusting screws. A two-vane spider (threaded rod) extends from the aluminum cylinder.



TOTAL WEIGHT: 7 lb, 14.5 oz

Figure 37

Mechanical Description of Tube Box



NOTE: Drawing is 1/2 scale.

TOTAL WEIGHT: 12 oz

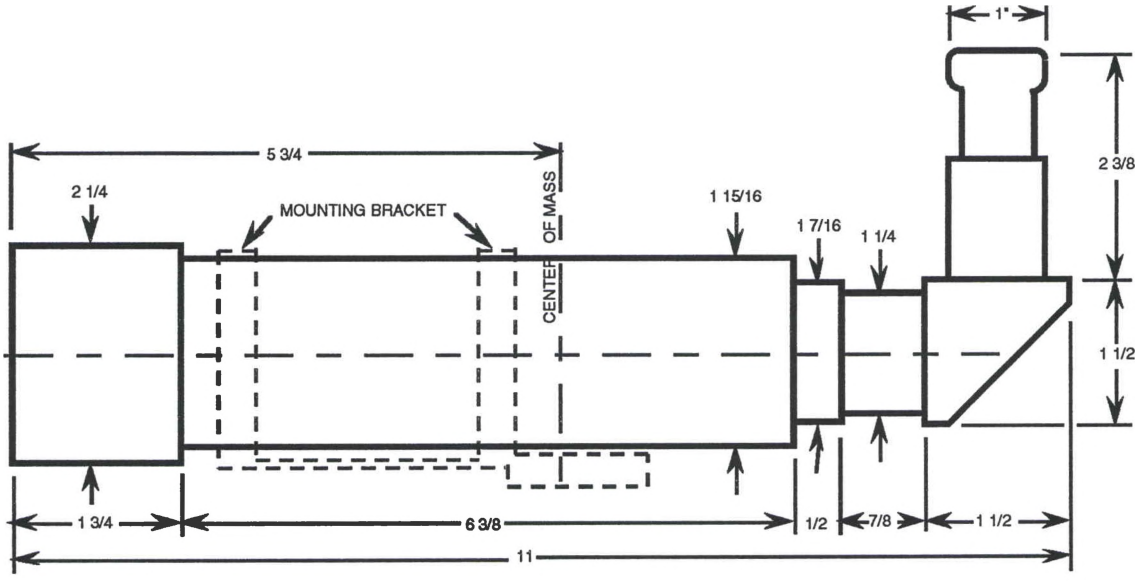
Figure 38

Mechanical Description of Secondary, Holder, Spider

Finder (w/ Mount)

The finder is constructed primarily of metal and glass as shown in Figure 39. This subassembly includes the mounting bracket, diagonal, and eyepiece. The front edge of the finder is positioned

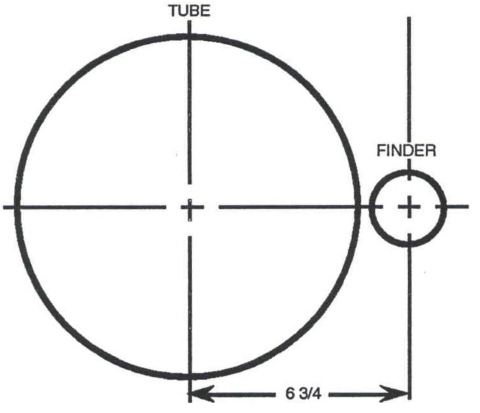
3/8" behind the front edge of the main telescope tube. The finder body is mounted at the side of and parallel to the main tube, spaced as shown below.



NOTES:

Drawing above is 1/2 scale.

Drawing at right is 1/6 scale.



TOTAL WEIGHT: 1 lb, 7 oz

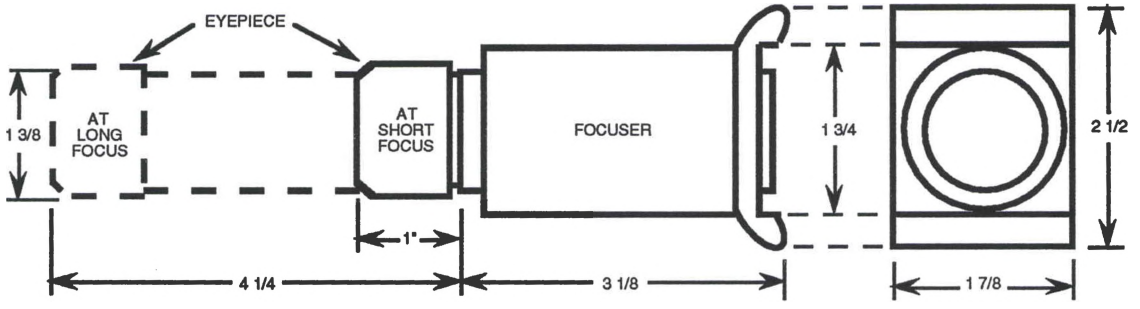
Figure 39

Mechanical Description of Finder

Focuser/Eyepiece

The focuser (see Figure 40) is primarily cast metal, while the eyepiece is largely metal and glass. The rack and pinion mechanism

(not shown) slides the eyepiece in and out of the focuser. Dashed lines portray the position of the eyepiece at maximum extension. Assembly is mounted directly over the hole in the top plate (see Figure 36), centered 7 1/4" from front end of tube. The focuser is hollow, with an inside diameter of 1 1/4". Except for glass lens elements, the eyepiece is also a hollow tube, with an inside diameter of approximated 1 1/8".



Drawing is 1/2 scale. TOTAL WEIGHT: 11 oz

Figure 40

Mechanical Description of Focuser, Eyepiece

Mirror/Mirror Cell

The mirror is Pyrex® glass of 8" diameter, 1 3/8" thick, and is supported in the cell by three Teflon pads. The cell is constructed of two 3/8"-thick phenolic disks (A and B) and a section of PVC pipe for the sidewall. Four 2" x 1 1/8" sidewall cutouts are for handling and removing the mirror. This assembly is shown in Figure 41.

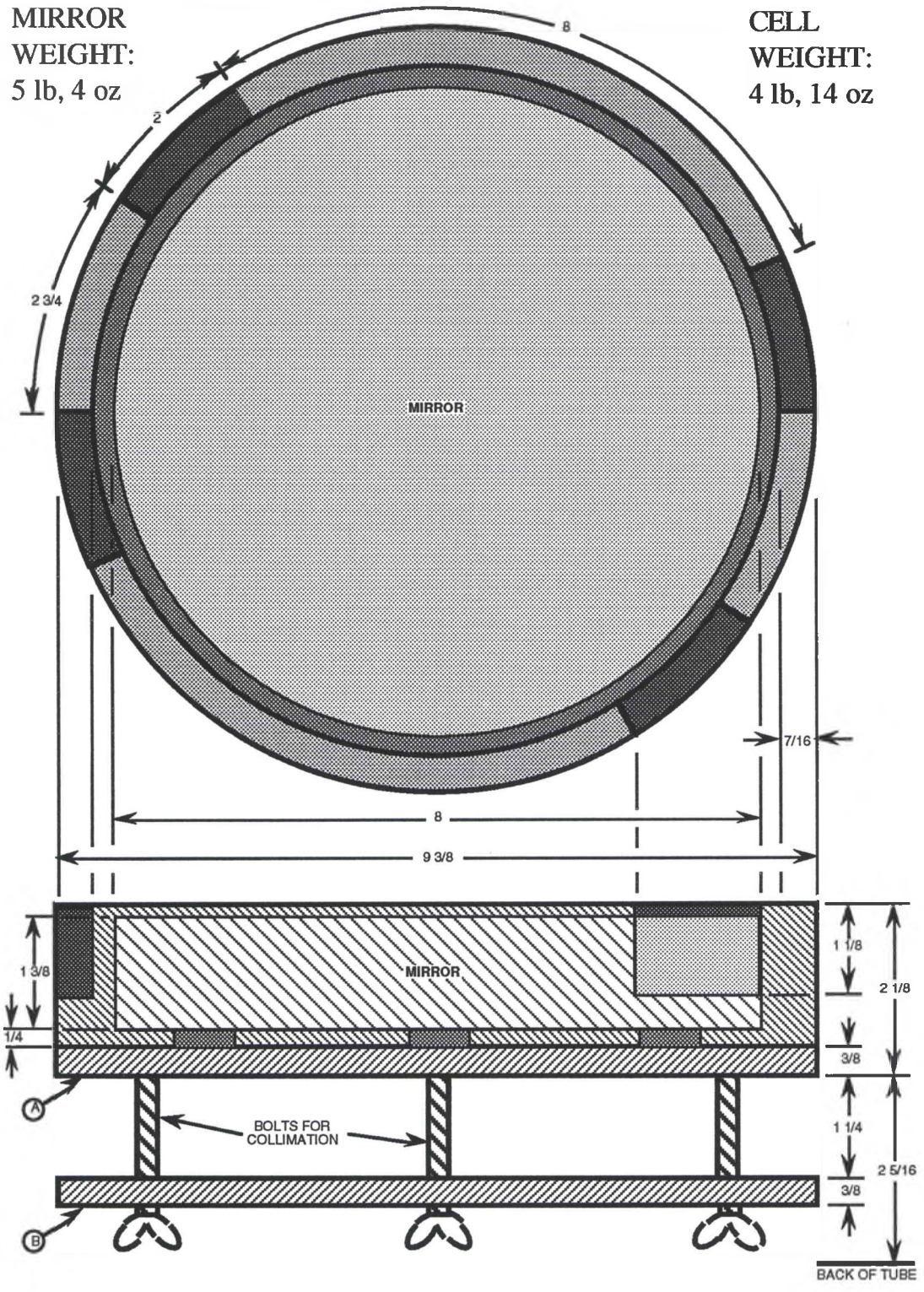


Figure 41

Mechanical Description of Mirror, Cell

Rocker Box

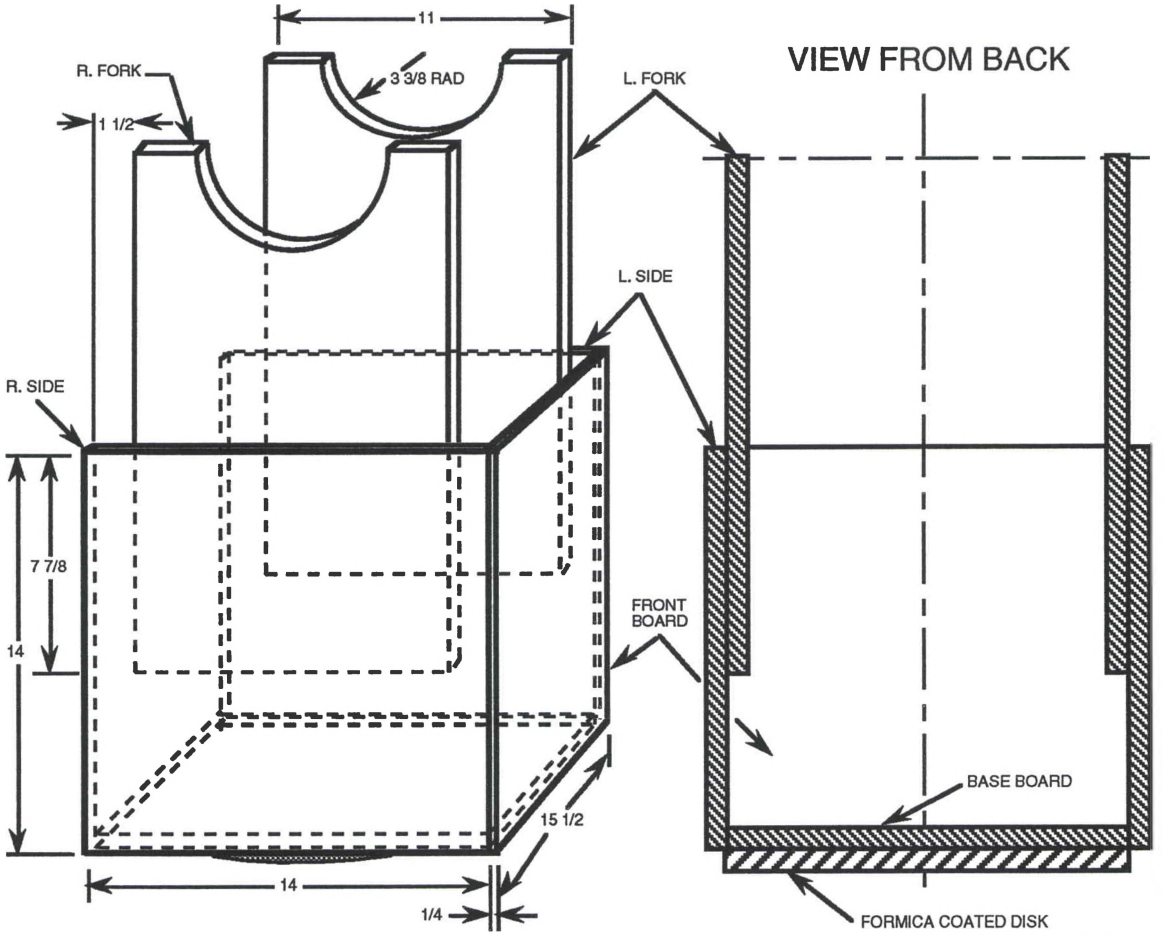
The rocker box (see Figure 42) is constructed from six sections of plywood and a disk of formica on particle board. The assembly is mounted on a pivot atop a larger base assembly (not shown) for azimuth rotation. Elements of construction are labelled as shown below. Dimensions and weights are:

Side Boards (2)	14" x 14" x 3/4"	3.143 lb*
Base Board (1)	14" x 14" x 3/4"	3.143 lb*
Front Board (1)	14 x 15 1/2" x 1/4"	1.160 lb*
Forks (2)	11" x 18" x 3/4"	2.888 lb*
	(w/ 6 3/4" Diameter Semicircular Cutout)	
Formica/Part. Board (1)	14 1/8" Diameter x 3/4"	2.512 lb

*Weights calculated (work not shown) assuming uniform density of all elements.

Base Board

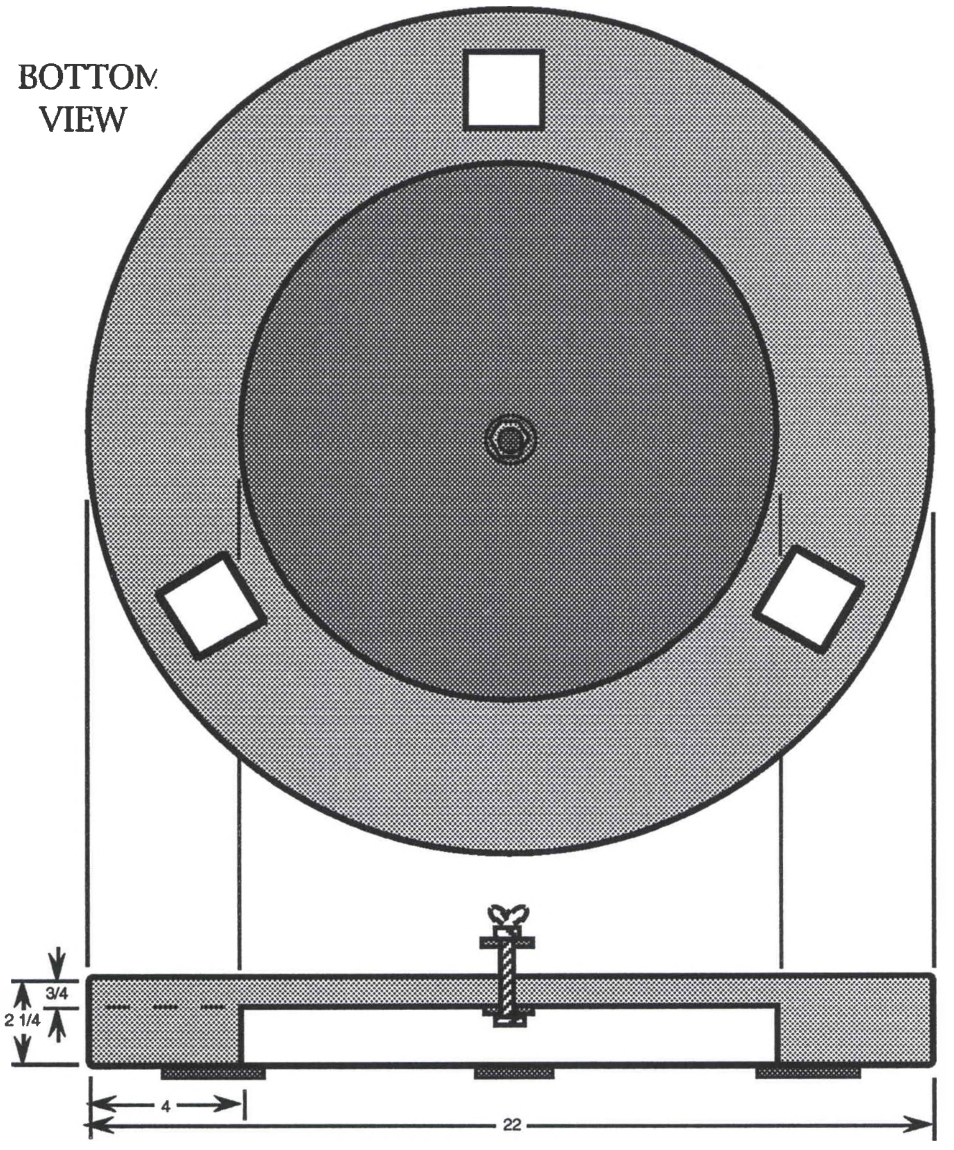
The base board (see Figure 43) is constructed from a 22" diameter plywood disk (3/4" thick) and two 3/4" plywood rings of outside diameter 22" and inside diameter 14". The three feet are 2" x 2" squares of 1/4" plywood. The bearing hardware includes a 3/8" bolt, wing nut, washers, and three 2" x 2" squares of Teflon (not shown) to support the rocker box and minimizing the sliding friction against the formica disk.



TOTAL WEIGHT: 18 lb, 14 oz

Figure 42

Mechanical Description of Rocker Box



TOTAL WEIGHT: 13 lb, 7 oz (excluding bearing hardware)

Figure 43

Mechanical Description of Mirror, Cell

APPENDIX B

MOMENT OF INERTIA (MOI) DETERMINATION

The following exercise was performed to determine the individual contribution of each major telescope subassembly to the total system moment of inertia (MOI) about each of the rotational axes (altitude and azimuth). Essentially, the telescope, without the control system hardware, is described below as divided into nine major subassemblies: tube (including lower focuser mounting plate), upper mounting plate, tube box, secondary mirror/holder/spider, finder (including mount), focuser/eyepiece, mirror/mirror cell, rocker box, and base plate. For each element, all pertinent approaches and assumptions are fully explained prior to calculation. A detailed subassembly description is in Appendix A, if needed.

Tube

Approach

Treat the tube as two solid cylinders of outside and inside diameters, respectively, and subtract one MOI from the other. Calculations will be made for the maximum MOI about the azimuth axis, which occurs when the tube is horizontal, since this places the center of mass (COM) at the maximum distance from that axis.

Assumptions

Since both axes (azimuth and altitude) intersect the tube at right angles to its length at the same pivot point, their respective MOIs should be equal ($J_A = J_E$). Since a portion of the tube around the cutout is displaced by the lower plate (made of aluminum, approximately three times the density of PVC), the lower plate will be treated as part of the tube itself.

Calculations

Let: M_T = Mass of Tube

M_O = Mass of Cylinder of Outside Diameter

M_I = Mass of Cylinder of Inside Diameter

L = Length of Tube = 4.208 ft

H = Displacement Between Parallel Axes

R_O = Outside Radius (Avg.) = 0.443 ft

R_I = Inside Radius (Avg.) = 0.396 ft

$J_T = J_{T_A} = J_{T_E}$ = MOI of Tube

J_{CM} = MOI of Tube About its COM

W_T = Weight of Tube w/ Lower Plate = 30.813 lb

g = Acceleration of Gravity = 32.2 ft/sec^2

$$M_T = \frac{W_T}{g} = \frac{30.813 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.957 \text{ slug}$$

$$M_O = M_T \left(\frac{R_O^2}{R_O^2 - R_I^2}\right) \text{ and } M_I = M_T \left(\frac{R_I^2}{R_O^2 - R_I^2}\right)$$

$$J_{CM} = J_O - J_I = \left(\frac{M_O R_O^2}{4} + \frac{M_O L^2}{12}\right) - \left(\frac{M_I R_I^2}{4} + \frac{M_I L^2}{12}\right)$$

$$= \frac{1}{4}\left(M_O R_O^2 - M_I R_I^2\right) + \frac{1}{12} L^2 \left(M_O - M_I\right)$$

$$= \frac{M_T}{12\left(R_O^2 - R_I^2\right)}\left(3R_O^4 - 3R_I^4 + L^2 R_O^2 - L^2 R_I^2\right)$$

$$= \frac{0.957 \text{ slug}}{12\left((0.443 \text{ ft})^2 - (0.396 \text{ ft})^2\right)}\left[3(0.443 \text{ ft})^4\right.$$

$$- 3(0.396 \text{ ft})^4 + (4.208 \text{ ft})^2(0.443 \text{ ft})^2$$

$$\left. - (4.208 \text{ ft})^2(0.396 \text{ ft})^2\right]$$

$$= 1.497 \text{ slug-ft}^2$$

Note: The altitude axis is displaced from the tube COM by 3 1/4" (0.271 ft). The azimuth axis is displaced from the tube COM by 0.271 cos θ ft, where θ is the altitude angle from the horizon. Again, however, the maximum MOI occurs at θ = 0°, i.e., when the tube is horizontal. In this case, the displacement is H = 0.271 ft. Using the Parallel Axis Theorem, then

$$J_T = J_{CM} + M_T H^2 = 1.497 \text{ slug-ft}^2 + (0.957 \text{ slug})(0.271 \text{ ft})^2$$

$$= 1.567 \text{ slug-ft}^2$$

So,

$$J_{T_A} = 1.567 \text{ slug-ft}^2$$

$$J_{T_E} = 1.567 \text{ slug-ft}^2$$

Top Plate

Approach/Assumptions

This element will be treated as a thin plate. Further, the hole will be considered negligible. In determining the MOI about an axis at the COM, only the area of the plate projected onto a plane perpendicular to the axis need be considered. (See Figure 44.)

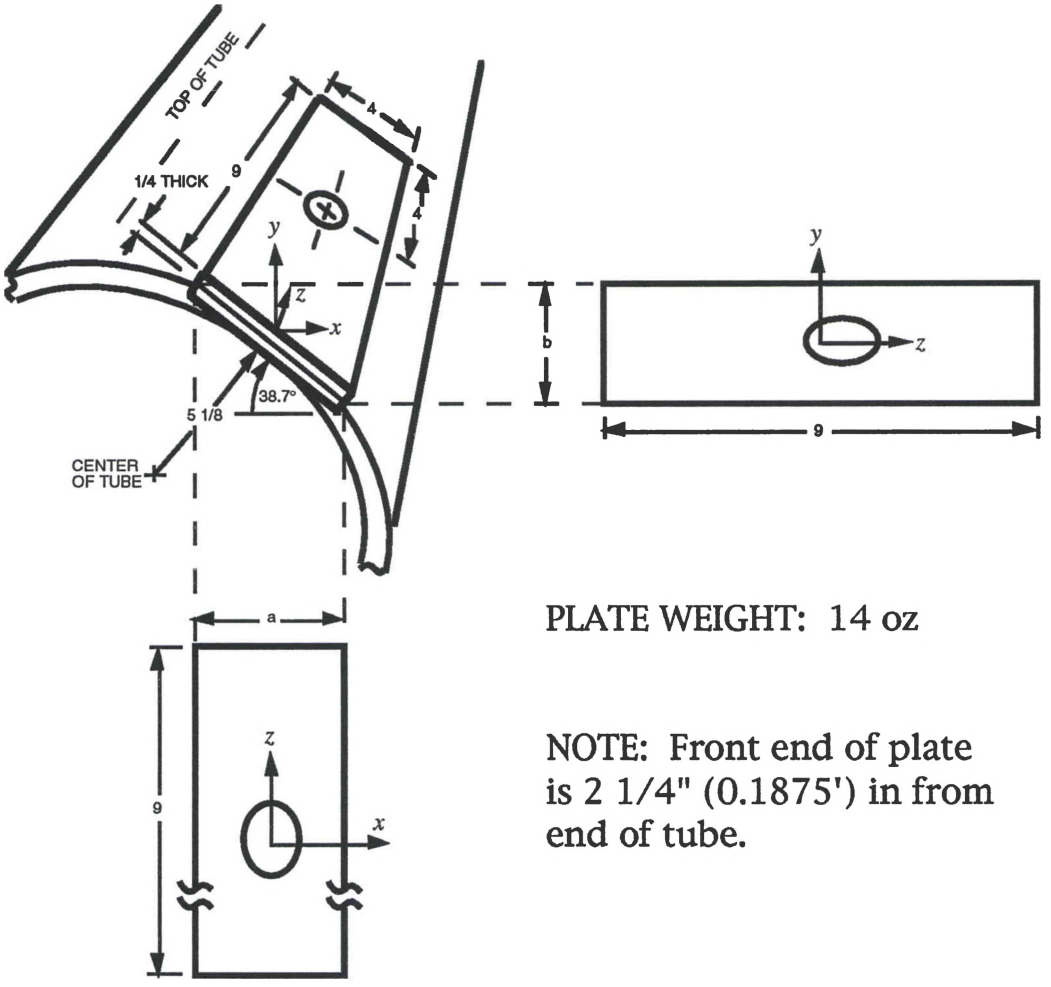


Figure 44

MOI Analysis Approach for Top Plate

Calculations

Let: M_P = Mass of Plate

W_P = Weight of Plate = 0.875 lb

L = Length of Plate = 0.75 ft

w = Width of Plate = 0.333 ft

a = Width in Plane Perpendicular to Azimuth Axis

b = Width in Plane Perpendicular to Altitude Axis

A = Area Projected onto Perpendicular Plane

H = Displacement Between Parallel Axes

g = Acceleration of Gravity = 32.2 ft/sec^2

$J_{CM_A} = J_y$ = MOI About an Axis Parallel to the Azimuth Axis at the COM

$J_{CM_E} = J_x$ = MOI About an Axis Parallel to the Altitude Axis at the COM

r = Moment Distance to Incremental Mass, dM

J_{P_A} = MOI of Plate About Azimuth Axis

J_{P_E} = MOI of Plate About Altitude Axis

$$M_P = \frac{W_P}{g} = \frac{0.875 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0272 \text{ slug}$$

$$a = w \cos 38.7° = (0.333 \text{ ft})(0.8) = 0.267 \text{ ft}$$

$$b = w \sin 38.7° = (0.333 \text{ ft})(0.6) = 0.2 \text{ ft}$$

$$J_{CM_A} = J_y = \int r^2 dM = \int r^2 \left(\frac{M_P}{A}\right) dA = \frac{M_P}{aL} \int_{-L/2}^{L/2} \int_{-a/2}^{a/2} (x^2 + z^2)\, dx\, dz$$

$$= \frac{M_P}{12}\left(a^2 + L^2\right) = \frac{0.0272 \text{ slug}}{12}\left[(0.267 \text{ ft})^2 + (0.75 \text{ ft})^2\right]$$

$$= 0.00144 \text{ slug-ft}^2$$

Similarly,

$$J_{CM_E} = J_x = \int r^2 dM = \int r^2 \left(\frac{M_P}{A}\right) dA = \frac{M_P}{bL} \int_{-L/2}^{L/2} \int_{-b/2}^{b/2} \left(y^2 + z^2\right) dy\,dz$$

$$= \frac{M_P}{12}\left(b^2 + L^2\right) = \frac{0.0272 \text{ slug}}{12}\left[(0.2 \text{ ft})^2 + (0.75 \text{ ft})^2\right]$$

$$= 0.00137 \text{ slug-ft}^2$$

To determine the displacement between the parallel axes through the tube COM and the plate COM, respectively, consider Figure 45.

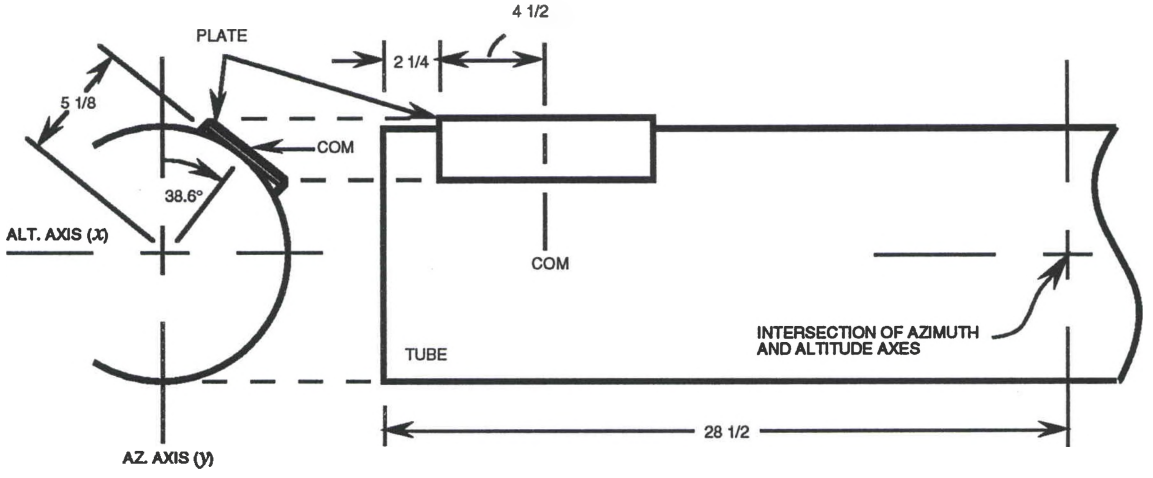


Figure 45

Parallel Axis Theorem Applied to Top Plate

H_z (Comp. Along Tube) = (28.5 in - 2.25 in -4.5 in) $\left(\frac{1}{12}\ \text{ft/in}\right)$

= 1.8125 ft

H_x (Distance Along x Axis) = (5.125 in) (sin 37°) $\left(\frac{1}{12}\ \text{ft/in}\right)$

= 0.256 ft

H_y (Distance Along y Axis) = (5.125 in) (cos 37°) $\left(\frac{1}{12}\ \text{ft/in}\right)$

= 0.342 ft

$$J_{P_A} = J_{CM_A} + M_P H^2 = J_{CM_A} + M_P (H_x^2 + H_z^2)$$

$$= 0.00144 \text{ slug-ft}^2 + 0.0272 \text{ slug } [(0.256 \text{ ft})^2 + (1.8125 \text{ ft})^2] = 0.0926 \text{ slug-ft}^2$$

$$J_{P_E} = J_{CM_E} + M_P H^2 = J_{CM_E} + M_P (H_y^2 + H_z^2)$$

$$= 0.00137 \text{ slug-ft}^2 + 0.0272 \text{ slug } [(0.342 \text{ ft})^2 + (1.8125 \text{ ft})^2] = 0.0939 \text{ slug-ft}^2$$

So,

$J_{P_A} = 0.093$ slug-ft^2

$J_{P_E} = 0.094$ slug-ft^2

Tube Box

Approach

MOI can be determined by summing the MOIs of three pairs of elements: two rectangular plates (11" × 12 14") perpendicular to the

respective axis, two rectangular plates parallel to the respective axis, and two disks - bearings - (6 5/8" diameter × 3/4" thick).

Assumptions

All elements will be treated as thin plates or disks. Assume PVC is approximately the same density as plywood. The handle will be considered negligible.

Calculations

Let: M_{TB} = Mass of Tube Box

$M_{\parallel P}$ = Mass of One Parallel Plate

$M_{\perp P}$ = Mass of One Perpendicular Plate

M_D = Mass of One Disk

A_{TB} = Surface Area of All Elements

$A_{\parallel P}$ = Area of One Parallel Plate

$A_{\perp P}$ = Area of One Perpendicular Plate

A_D = Area of One Disk

r_D = Radius of One Disk

J_{TB} = MOI of Tube Box

$J_{\parallel P}$ = MOI of One Parallel Plate

$J_{\perp P}$ = MOI of One Perpendicular Plate

J_D = MOI of One Disk

$R_{\parallel P}$ = Moment Distance, Axis to Center of Parallel Plate

R_D = Moment Distance, Axis to Center of Disk

W_{TB} = Weight of Tube Box = 7.906 lb

g = Acceleration of Gravity = 32.2 ft/sec^2

$$M_{TB} = \frac{W_{TB}}{g} = \frac{7.906 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.246 \text{ slug}$$

$$A_{\parallel P} = (10.75 \text{ in})\,(11 \text{ in})\left(\frac{1}{144} \text{ ft}^2/\text{in}^2\right) = 0.821 \text{ ft}^2$$

$$A_{\perp P} = (12.25 \text{ in})\,(11 \text{ in})\left(\frac{1}{144} \text{ ft}^2/\text{in}^2\right) = 0.936 \text{ ft}^2$$

$$A_D = \pi\,(3.3125 \text{ in})^2\left(\frac{1}{144} \text{ ft}^2/\text{in}^2\right) = 0.239 \text{ ft}^2$$

$$A_{TB} = \left(A_{\parallel P} + A_{\perp P} + A_D\right)\,2 = 3.993 \text{ ft}^2$$

$$M_{\parallel P} = \left(\frac{A_{\parallel P}}{A_{TB}}\right) M_{TB} = \left(\frac{0.821 \text{ ft}^2}{3.993 \text{ ft}^2}\right)(0.246 \text{ slug}) = 0.0505 \text{ slug}$$

$$M_{\perp P} = \left(\frac{A_{\perp P}}{A_{TB}}\right) M_{TB} = \left(\frac{0.936 \text{ ft}^2}{3.993 \text{ ft}^2}\right)(0.246 \text{ slug}) = 0.0575 \text{ slug}$$

$$M_D = \left(\frac{A_D}{A_{TB}}\right) M_{TB} = \left(\frac{0.239 \text{ ft}^2}{3.993 \text{ ft}^2}\right)(0.246 \text{ slug}) = 0.0147 \text{ slug}$$

$$R_{\parallel P} = \frac{1}{2}\left(\frac{12.25 \text{ in} + 10.75 \text{ in}}{2}\right)\left(\frac{1}{12} \text{ ft/in}\right) = 0.479 \text{ ft}$$

Due to dimensional symmetry, $J_{\parallel P_A} = J_{\parallel P_E}$ and $J_{\perp P_A} = J_{\perp P_E}$. However, $J_{D_A} \neq J_{D_E}$. Since one dimension of each parallel plate is along the axis of rotation, it can be treated as a thin rod of length, 11". Combining this with the Parallel Axis Theorem,

$$J_{\parallel P} \text{ (as a thin rod)} = \frac{1}{12} M_{\parallel P} L^2 + M_{\parallel P} R_{\parallel P}^2$$

$$= \frac{1}{12}(0.0505 \text{ slug})\left(\frac{11}{12} \text{ ft}\right)^2 + (0.0505 \text{ slug})\,(0.479 \text{ ft})^2$$

$$= 0.0151 \text{ slug-ft}^2$$

$J_{\perp P}$: Let (a,b) = dimensions of each perpendicular plate, such that a = 12.25 in, b = 11 in.

$$J_{\perp P} = \int_{a\,b} r^2 dM = \int_{a\,b} r^2 \left(\frac{M}{A}\right) dA = \int_{-a/2}^{a/2} \int_{-b/2}^{b/2} \left(x^2 + y^2\right) dx\,dy$$

$$= \frac{M_{\perp P}}{12}\left(a^2 + b^2\right) = \frac{0.0575 \text{ slug}}{12}\left[(12.25 \text{ in})^2 + (11 \text{ in})^2\right]\left(\frac{1}{144} \text{ ft}^2/\text{in}^2\right)$$

$$= 0.0090 \text{ slug-ft}^2$$

$$R_D = R_{\parallel P} + (0.75 \text{ in})\left(\frac{1}{12} \text{ ft/in}\right) = 0.479 \text{ ft} + 0.0625 \text{ ft} = 0.542 \text{ ft}$$

$$r_D = (3.3125 \text{ in})\left(\frac{1}{12} \text{ ft/in}\right) = 0.276 \text{ ft}$$

$$J_{D_A} = \frac{M_D r_D^2}{4} + M_D R_D^2 = \frac{1}{4}(0.0147 \text{ slug})(0.276 \text{ ft})^2 + (0.0147 \text{ slug})(0.542 \text{ ft})^2 = 0.0046 \text{ slug-ft}^2$$

$$J_{D_E} = \frac{M_D r_D^2}{2} = \frac{1}{2}(0.0147 \text{ slug})(0.276 \text{ ft})^2 = 0.0006 \text{ slug-ft}^2$$

$$J_{TB_A} = 2\left(J_{\parallel P} + J_{\perp P} + J_{D_A}\right) = 2\,(0.0151 \text{ slug-ft}^2 + 0.0090 \text{ slug-ft}^2 + 0.0046 \text{ slug-ft}^2) = 0.057 \text{ slug-ft}^2$$

$$J_{TB_E} = 2\left(J_{\parallel P} + J_{\perp P} + J_{D_E}\right) = 2\,(0.0151 \text{ slug-ft}^2 + 0.0090 \text{ slug-ft}^2 + 0.0006 \text{ slug-ft}^2) = 0.049 \text{ slug-ft}^2$$

So,

$$J_{TB_A} = 0.057 \text{ slug-ft}^2$$

$$J_{TB_E} = 0.049 \text{ slug-ft}^2$$

Secondary Mirror/Holder/Spider

Approach/Assumptions

This subassembly can be modeled as a solid cylinder of homogeneous material (i.e., uniform density), and diameter 1 3/4". Length could be shortened to 3 1/2" to maintain position of the COM with respect to the flat (aluminum) end. Since both the azimuth and altitude axes are perpendicular to the length of the cylinder, we use the standard formula

$$J_{CM} = \frac{MR^2}{4} + \frac{ML^2}{12}$$

for both J_A and J_E. Then, by the Parallel Axis Theorem, add the quantity, MH^2, where H is the distance from the COM to the respective axis. Due to the dimensional symmetry, $H_A = H_E$. (See Figure 46.)

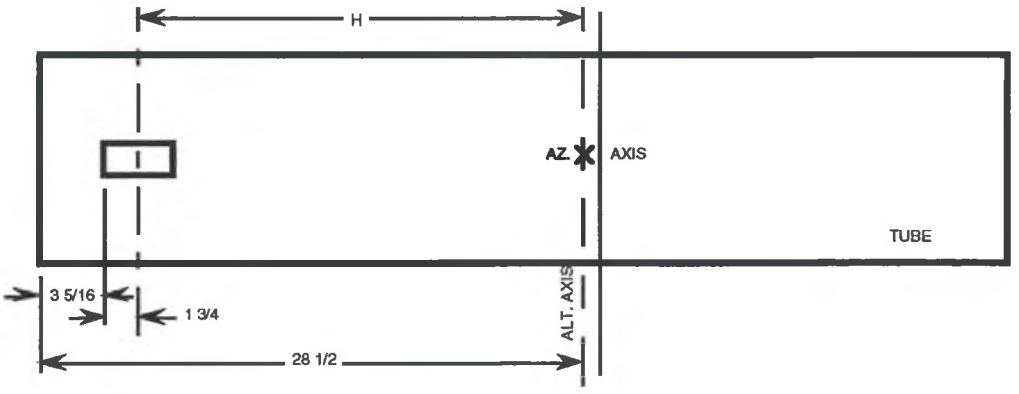


Figure 46

Parallel Axis Theorem Applied to Secondary

Calculations

Let: M_S = Total Mass of Subassembly

$J_S = J_{S_A} = J_{S_E}$ = MOI of Subassembly

R_S = Radius of Effective Cylinder = 0.073 ft

L = Length of Effective Cylinder = 0.292 ft

W_S = Total Weight of Subassembly = 0.75 lb

g = Acceleration of Gravity = 32.2 ft/sec^2

$$M_S = \frac{W_S}{g} = \frac{0.75 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0233 \text{ slug}$$

$$H = (28.5 \text{ in} - 3.3125 \text{ in} - 1.75 \text{ in}) \left(\frac{1}{12} \text{ ft/in}\right) = 1.953 \text{ ft}$$

$$J_S = J_{S_A} = J_{S_E} = \frac{M_S R_S^2}{4} + \frac{M_S L^2}{12} + M_S H^2 = M_S \left(\frac{R_S^2}{4} + \frac{L^2}{12} + H^2\right)$$

$$= (0.0233 \text{ slug}) \left[\frac{1}{4} (0.073 \text{ ft})^2 + \frac{1}{12} (0.292 \text{ ft})^2 + (1.953 \text{ ft})^2\right] = 0.089 \text{ slug-ft}^2$$

So,

$$J_{S_A} = 0.089 \text{ slug-ft}^2$$

$$J_{S_E} = 0.089 \text{ slug-ft}^2$$

Finder w/ Mount

Approach/Assumptions

This subassembly can also be modeled as a solid cylinder of homogeneous material (i.e., uniform density). The problem can be simplified by using an average diameter of 2". Length could be increased to 11 1/2" to maintain position of the COM with respect to the front end of the finder. Since both the azimuth and altitude axes are perpendicular to the length of the cylinder, again we use the standard formula

$$J_{CM} = \frac{MR^2}{4} + \frac{ML^2}{12}$$

for both J_A and J_E. Then, by the Parallel Axis Theorem, add the quantity, MH^2, where H is the distance from the COM to the respective axis. (See Figure 47.)

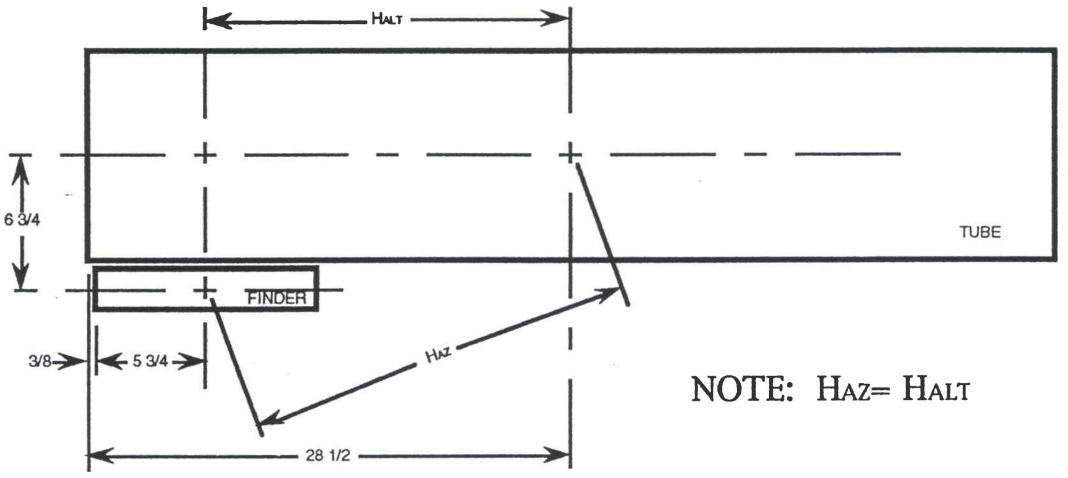


Figure 47

Parallel Axis Theorem Applied to Finder

Calculations

Let: M_F = Total Mass of Finder Subassembly

$J_F = J_{CM} + M_F H^2$ = MOI of Subassembly

R_F = Radius of Effective Cylinder = 0.083 ft

L = Length of Effective Cylinder = 0.958 ft

W_F = Total Weight of Subassembly = 1.4375 lb

g = Acceleration of Gravity = 32.2 ft/sec^2

$$M_F = \frac{W_F}{g} = \frac{1.4375 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0446 \text{ slug}$$

$$H_E^2 = \left[(28.5 \text{ in} - 5.75 \text{ in} - 0.375 \text{ in})\left(\frac{1}{12} \text{ ft/in}\right)\right]^2$$

$$= 3.477 \text{ ft}^2$$

$$H_A^2 = H_E^2 + \left[(6.75 \text{ in})\left(\frac{1}{12} \text{ ft/in}\right)\right]^2 = 3.477 \text{ ft}^2 + 0.316 \text{ ft}^2$$

$$= 3.793 \text{ ft}^2$$

$$J_{CM} = \frac{M_F R_F^2}{4} + \frac{M_F L^2}{12} = (0.0446 \text{ slug}) \left[\frac{1}{4} (0.083 \text{ ft})^2 + \frac{1}{12} (0.958 \text{ ft})^2 \right] = 0.0035 \text{ slug-ft}^2$$

$$J_{F_A} = J_{CM} + M_F H_A^2 = 0.0035 \text{ slug-ft}^2 + (0.0446 \text{ slug}) (3.793 \text{ ft}^2) = 0.173 \text{ slug-ft}^2$$

$$J_{F_E} = J_{CM} + M_F H_E^2 = 0.0035 \text{ slug-ft}^2 + (0.0446 \text{ slug}) (3.477 \text{ ft}^2) = 0.159 \text{ slug-ft}^2$$

So,

$$J_{F_A} = 0.173 \text{ slug-ft}^2$$

$$J_{F_E} = 0.159 \text{ slug-ft}^2$$

Focuser/Eyepiece

Approach/Assumptions

This subassembly can be modeled as a hollow cylinder of homogeneous material of outside diameter, 1.5" (average) and inside diameter 1.2" (average). Length should be modeled as it would be at full extension, 7.375". MOI is first determined about the COM of the subassembly by integrating in three dimensions (cylindrical coordinates) over the volume of the model cylinder. Since the model is hollow, integration should be achieved from the inner to the outer radii.

Derivation (Refer to Figure 48.)

Let: M_{FE} = Mass of Focuser/Eyepiece

L = Length of Model = 7.375 in = 0.6146 ft

V = Volume of Model

α = Offset Angle = 53.1°

$\beta = 90° - \alpha = 36.9°$

r_o = Outer Radius = 0.0625 ft

r_i = Inner Radius = 0.05 ft

R_m = Moment Distance to Differential Volume, dV

J_{CM_A} = MOI About the COM Through an Axis Parallel to the Azimuth Axis

J_{CM_E} = MOI About the COM Through an Axis Parallel to the Altitude Axis

W_S = Weight of Subassembly = 0.688 lb

g = Acceleration of Gravity = 32.2 ft/sec^2

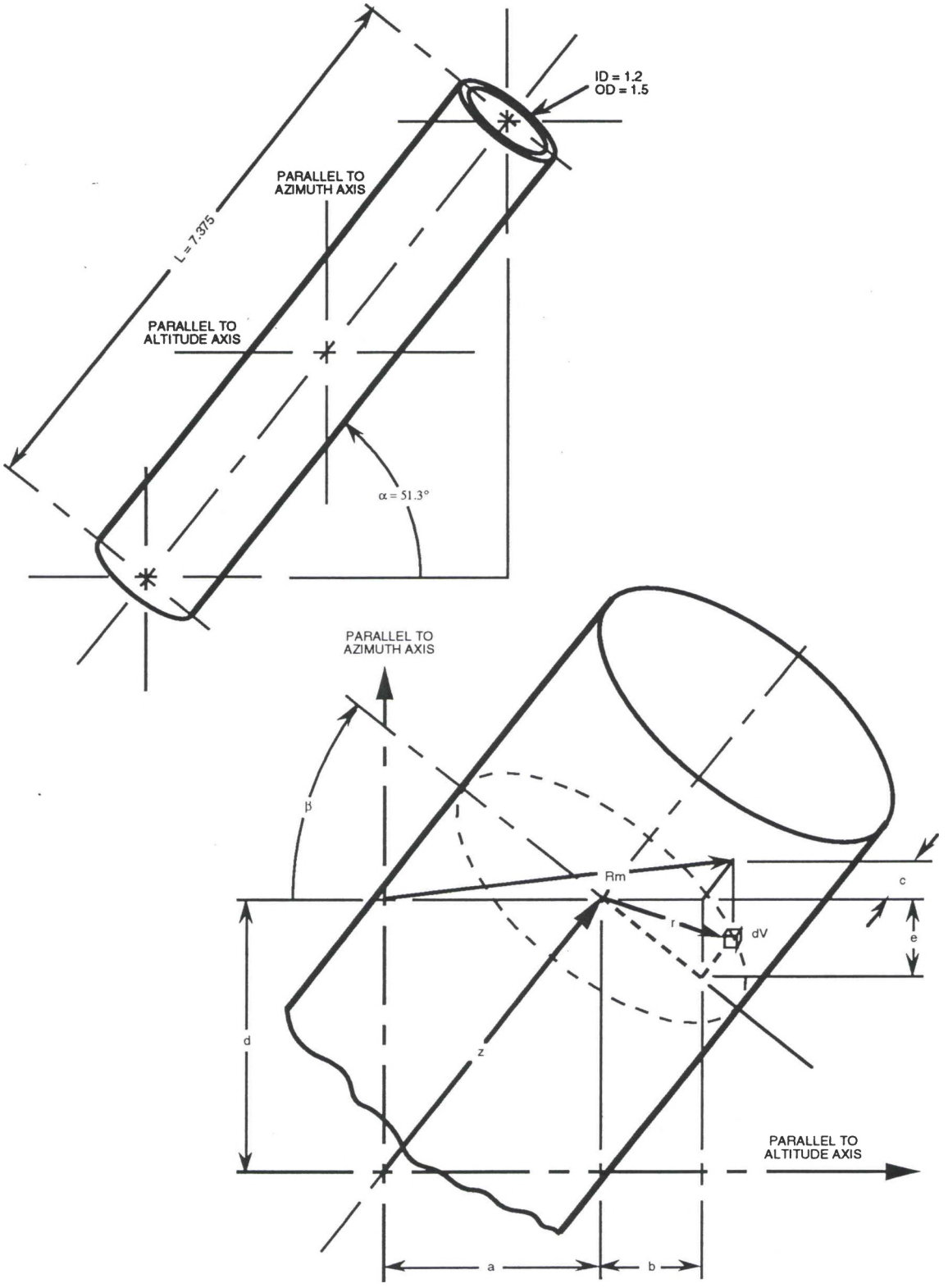


Figure 48

Derivation of MOI for Focuser/Eyepiece

$$J_{CM} = \int R_m^2 dM = \int_V R_m^2 \left(\frac{M}{V}\right) dV = \int_{z_1}^{z_2} \int_{\theta_1}^{\theta_2} \int_{r_1}^{r_2} R_m^2 \left(\frac{M}{V}\right) r \, dr d\theta dz$$

From the drawing, R_{m_A} is the shortest distance between dV and the azimuth axis, or

$$R_{m_A}^2 = (a + b)^2 + c^2$$

From the drawing,

$$a = z \sin\beta \qquad b = r \cos\theta \cos\beta \qquad c = r \sin\theta$$

So,

$$R_{m_A}^2 = (z \sin\beta + r \cos\theta \cos\beta)^2 + r^2 \sin^2\theta$$

$$= (\sin^2\theta + \cos^2\beta \cos^2\theta) r^2 + (2z \sin\beta \cos\beta \cos\theta) r + z^2 \sin^2\beta$$

From the drawing, R_{m_E} is the shortest distance between dV and the altitude axis, or

$$R_{m_E}^2 = (d - e)^2 + c^2 \text{ , where}$$

$$d = z \cos\beta \qquad e = r \cos\theta \sin\beta \qquad c = r \sin\theta$$

So,

$$R_{m_E}^2 = (z \cos\beta - r \cos\theta \sin\beta)^2 + r^2 \sin^2\theta$$

$$= \left(\sin^2\theta + \sin^2\beta \cos^2\theta\right) r^2 - \left(2z \sin\beta \cos\beta \cos\theta\right) r + z^2 \cos^2\beta$$

Both MOIs can now be determined by the following integral:

$$J_{CM} = \frac{M}{V} \int_{-L/2}^{L/2} \int_{0}^{2\pi} \int_{r_i}^{r_o} R_m^2 \, r \, dr d\theta dz \quad , \quad \text{where} \quad V = \pi L \left(r_o^2 - r_i^2\right)$$

Therefore,

$$J_{CM_A} = \frac{M \sin^2\beta}{4} \left(r_o^2 + r_i^2 + \frac{1}{3} L^2\right)$$

$$J_{CM_E} = \frac{M \cos^2\beta}{4} \left(r_o^2 + r_i^2 + \frac{1}{3} L^2\right)$$

Calculations

$$M_{FE} = \frac{W_{FE}}{g} = \frac{0.688 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0214 \text{ slug}$$

$$J_{CM_A} = \frac{M_{FE} \sin^2\beta}{4} \left(r_o^2 + r_i^2 + \frac{1}{3} L^2\right)$$

$$= \frac{(0.0214 \text{ slug}) (0.6)^2}{4} \left[(0.0625 \text{ ft})^2 + (0.05 \text{ ft})^2 + \frac{1}{3} (0.6146 \text{ ft})^2\right] = 0.00025 \text{ slug-ft}^2$$

$$J_{CM_E} = \frac{M_{FE} \cos^2\beta}{4} \left(r_o^2 + r_i^2 + \frac{1}{3} L^2\right)$$

$$= \frac{(0.0214 \text{ slug}) (0.8)^2}{4} \left[(0.0625 \text{ ft})^2 + (0.05 \text{ ft})^2\right.$$

$$+ \frac{1}{3}(0.6146 \text{ ft})^2 \Big] = 0.00045 \text{ slug-ft}^2$$

To determine the offset MOI by use of the Parallel Axis Theorem, it is first necessary to determine the components of H. In doing so, refer to Figure 49.

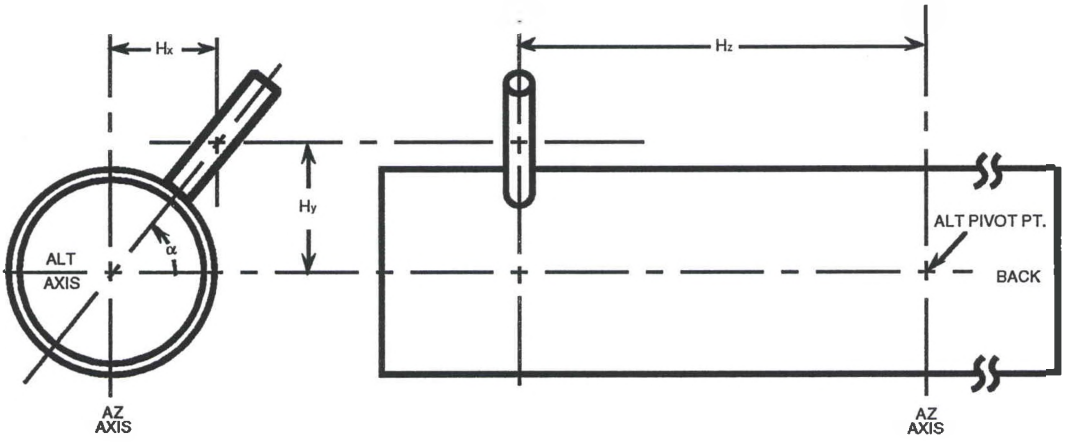


Figure 49

Parallel Axis Theorem Applied to Focuser/Eyepiece

$$H_{xy} = \text{Inside Radius of Tube} + \text{Plate Thickness} + \frac{L}{2}$$

$$= \left(5.125 \text{ in} + 0.25 \text{ in} + \frac{7.375 \text{ in}}{2}\right)\left(\frac{1}{12} \text{ ft/in}\right) = 0.755 \text{ ft}$$

$$H_x = H_{xy} \cos\alpha = (0.755 \text{ ft})\ (0.6) = 0.453 \text{ ft}$$

$$H_y = H_{xy} \sin\alpha = (0.755 \text{ ft})\ (0.8) = 0.604 \text{ ft}$$

$$H_z = \left(28.5 \text{ in} - 2.25 \text{ in} - 5 \text{ in}\right)\left(\frac{1}{12} \text{ ft/in}\right) = 1.771 \text{ ft}$$

$$H_A^2 = H_x^2 + H_z^2 = (0.453 \text{ ft})^2 + (1.771 \text{ ft})^2 = 3.341 \text{ ft}^2$$

$H_E^2 = H_y^2 + H_z^2 = (0.604 \text{ ft})^2 + (1.771 \text{ ft})^2 = 3.501 \text{ ft}^2$

Then, by the Parallel Axis Theorem,

$J_{FE_A} = J_{CM_A} + M_{FE} H_A^2 = 0.00025 \text{ slug-ft}^2$

$+ (0.0214 \text{ slug}) (3.341 \text{ ft}^2) = 0.0716 \text{ slug-ft}^2$

$J_{FE_E} = J_{CM_E} + M_{FE} H_E^2 = 0.00045 \text{ slug-ft}^2$

$+ (0.0214 \text{ slug}) (3.501 \text{ ft}^2) = 0.0752 \text{ slug-ft}^2$

So,

$J_{FE_A} = 0.072 \text{ slug-ft}^2$

$J_{FE_E} = 0.075 \text{ slug-ft}^2$

Mirror/Mirror Cell

Approach

Refer to Figure 41 in Appendix A. Determine MOI of this subassembly by finding the individual contributions of four separate parts: mirror and two disks, modeled as disks (cylinders), and the side wall, modeled as a hollow cylinder.

Assumptions

Due to symmetry, $J_{MC_A} = J_{MC_E}$ for all cell elements. Assume PVC and Phenolic are the same density. All bolts, cutouts, and pads are considered to have a negligible effect on the MOI. However, cutouts are considered in determining mass allocations for cell components.

Calculations

Let: M_M = Mass of the Mirror

$M_A = M_B$ = Mass of Each 3/8" Disk

M_{SW} = Mass of the Side Wall

M_C = Mass of the Cell = $M_A + M_B + M_{SW}$

T_M = Mirror Thickness = 1 3/8 in = 0.115 ft

$T_A = T_B$ = Thickness of Each Disk = 3/8 in = 0.0313 ft

T_{SW} = Height of Side Wall = 1 3/4 in = 0.146 ft

T_{CUT} = Depth of Side Wall Cutout = 1 1/8 in = 0.094 ft

w_{CUT} = Width of Side Wall Cutout = 2 in = 0.167 ft

R_M = Mirror Radius = 4 in = 0.333 ft

$R_A = R_B$ = Disk Radius = 4 11/16 in = 0.391 ft

R_{SW_O} = Outer Side Wall Radius = 4 11/16 in = 0.391 ft

R_{SW_I} = Inner Side Wall Radius = 4 1/4 in = 0.354 ft

$V_A = V_B$ = Disk Volume

V_{SW} = Side Wall Volume

V_C = Total Cell Volume

J_{CM} = MOI About COM for Each Element

J_M = MOI Contribution from Mirror

$J_A = J_B$ = MOI Contribution from Each Disk

J_{SW} = MOI Contribution from Side Wall

J_{MC} = Total MOI for Mirror and Cell

H_M = Parallel Axis Distance to Mirror COM

H_A = Parallel Axis Distance to Disk A COM

H_B = Parallel Axis Distance to Disk B COM

H_{SW} = Parallel Axis Distance to Side Wall COM

W_M = Mirror Weight = 5.25 lb

W_C = Cell Weight = 4.875 lb

g = Acceleration of Gravity = 32.2 ft/sec^2

$$M_M = \frac{W_M}{g} = \frac{5.25 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.163 \text{ slug}$$

$$M_C = \frac{W_C}{g} = \frac{4.875 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.151 \text{ slug}$$

$$V_A = V_B = \pi R_A^2 T_A = \pi(0.391 \text{ ft})^2 (0.0313 \text{ ft}) = 0.0150 \text{ ft}^3$$

$$V_{SW} = V_{OUTER} - V_{INNER} - V_{CUTOUTS}$$

$$= (V_{OUTER} - V_{INNER}) (1 - \text{Cutout Proportion})$$

$$= \left(\pi R_{SW_O}^2 T_{SW} - \pi R_{SW_I}^2 T_{SW}\right) \left[1 - \left(\frac{T_{CUT}}{T_{SW}}\right) \left(\frac{4w_{CUT}}{2\pi R_{SW_O}}\right)\right]$$

$$= \pi T_{SW} \left(R_{SW_O}^2 - R_{SW_I}^2\right) \left[1 - \left(\frac{T_{CUT}}{T_{SW}}\right) \left(\frac{2}{\pi}\right) \left(\frac{w_{CUT}}{R_{SW_O}}\right)\right]$$

$$= \pi (0.146 \text{ ft}) \left((0.391 \text{ ft})^2 - (0.354 \text{ ft})^2\right) \left[1 - \left(\frac{0.094 \text{ ft}}{0.146 \text{ ft}}\right) \left(\frac{2}{\pi}\right) \left(\frac{0.167 \text{ ft}}{0.391 \text{ ft}}\right)\right] = 0.0103 \text{ ft}^3$$

$$V_C = V_A + V_B + V_{SW} = 2 (0.0150 \text{ ft}^3) + 0.0103 \text{ ft}^3 = 0.0402 \text{ ft}^3$$

$$M_A = M_B = \left(\frac{V_A}{V_C}\right) M_C = \left(\frac{0.0150 \text{ ft}^3}{0.0402 \text{ ft}^3}\right) (0.151 \text{ slug}) = 0.0564 \text{ slug}$$

$$M_{SW} = \left(\frac{V_{SW}}{V_C}\right) M_C = \left(\frac{0.0103 \text{ ft}^3}{0.0402 \text{ ft}^3}\right) (0.151 \text{ slug}) = 0.0386 \text{ slug}$$

$$H_M = \left(22 \text{ in} - 2.313 \text{ in} - 0.375 \text{ in} - 0.25 \text{ in} - 1.375 \text{ in}/2\right)\left(\frac{1}{12} \text{ ft/in}\right) = 1.531 \text{ ft}$$

$$H_A = \left(22 \text{ in} - 2.313 \text{ in} - 0.375 \text{ in}/2\right)\left(\frac{1}{12} \text{ ft/in}\right) = 1.625 \text{ ft}$$

$$H_B = \left(22 \text{ in} - 2.313 \text{ in} - 1.25 \text{ in} + 0.375 \text{ in}/2\right)\left(\frac{1}{12} \text{ ft/in}\right)$$

$$= 1.760 \text{ ft}$$

$$H_{SW} = \left(22 \text{ in} - 2.313 \text{ in} - 0.375 \text{ in} + 1.75 \text{ in}/2\right)\left(\frac{1}{12} \text{ ft/in}\right)$$

$$= 1.536 \text{ ft}$$

$$J_{MC} = \sum J_{CM} + MH^2 = J_{MC_A} = J_{MC_E}$$

$$J_M = J_{CM_M} + M_M H_M^2 = M_M\left(\frac{R_M^2}{4} + \frac{T_M^2}{12} + H_M^2\right)$$

$$= (0.163 \text{ slug})\left(\frac{(0.333 \text{ ft})^2}{4} + \frac{(0.115 \text{ ft})^2}{12} + (1.531 \text{ ft})^2\right)$$

$$= 0.387 \text{ slug-ft}^2$$

$$J_A = J_{CM_A} + M_A H_A^2 = M_A\left(\frac{R_A^2}{4} + \frac{T_A^2}{12} + H_A^2\right)$$

$$= (0.0564 \text{ slug})\left(\frac{(0.391 \text{ ft})^2}{4} + \frac{(0.0313 \text{ ft})^2}{12} + (1.625 \text{ ft})^2\right)$$

$$= 0.151 \text{ slug-ft}^2$$

$$J_B = J_{CM_B} + M_B H_B^2 = M_B\left(\frac{R_B^2}{4} + \frac{T_B^2}{12} + H_B^2\right)$$

$$= (0.0564 \text{ slug})\left(\frac{(0.391 \text{ ft})^2}{4} + \frac{(0.0313 \text{ ft})^2}{12} + (1.760 \text{ ft})^2\right)$$

$$= 0.177 \text{ slug-ft}^2$$

From the earlier MOI calculation for the main tube, the J_{CM} formula for a hollow cylinder was derived as

$$J_{CM} = \frac{M_T}{12\left(R_O^2 - R_I^2\right)}\left(3R_O^4 - 3R_I^4 + L^2R_O^2 - L^2R_I^2\right)$$

So,

$$J_{SW} = J_{CM_{SW}} + M_{SW}H_{SW}^2 = \frac{M_{SW}}{12\left(R_{SW_O}^2 - R_{SW_I}^2\right)}\left(3R_{SW_O}^4 - 3R_{SW_I}^4 + T_{SW}^2R_{SW_O}^2 - T_{SW}^2R_{SW_I}^2\right) + M_{SW}H_{SW}^2$$

$$= \frac{(0.0386 \text{ slug})}{12\left((0.391 \text{ ft})^2 - (0.354 \text{ ft})^2\right)}\left(3(0.391 \text{ ft})^4 - 3(0.354 \text{ ft})^4 + (0.146 \text{ ft})^2(0.391 \text{ ft})^2 - (0.146 \text{ ft})^2(0.354 \text{ ft})^2\right) + (0.0386 \text{ slug})(1.536 \text{ ft})^2 = 0\ 0940 \text{ slug-ft}^2$$

Therefore,

$$J_{MC_A} = J_{MC_E} = J_M + J_A + J_B + J_{SW} = 0.387 \text{ slug-ft}^2 + 0.151 \text{ slug-ft}^2 + 0.177 \text{ slug-ft}^2 + 0.094 \text{ slug-ft}^2$$

$$= 0.809 \text{ slug-ft}^2$$

So,

$$J_{MC_A} = 0.809 \text{ slug-ft}^2$$

$$J_{MC_E} = 0.809 \text{ slug-ft}^2$$

Rocker Box

Approach

Calculate MOI contributions for each element as performed earlier for the Tube Box. Then sum all results together.

Assumptions

Since no part of the rocker box is to be rotated about the altitude axis, it is not necessary to calculate the MOI about this axis (J_{RB_E}). All elements are to be treated as thin plates or disks. The formica/particle board disk is assumed to be of the same density as plywood. The pivot hole and all hardware are assumed to have a negligible effect on the MOI.

Calculations

Let: M_{SB} = Mass of One Side Board

M_{BB} = Mass of Base Board

M_{FB} = Mass of Front Board

M_F = Mass of One Fork

M_{PB} = Mass of Formica/Particle Board Disk

R_{SB} = Moment Distance from Axis to Center of Side Board

R_{FB} = Moment Distance from Axis to Center of Front Board

J_{SB} = MOI Contribution from One Side Board

J_{BB} = MOI Contribution from Base Board

J_{FB} = MOI Contribution from Front Board

J_F = MOI Contribution from One Fork

J_{PB} = MOI Contribution from Formica/Particle Board Disk

J_{RB} = Total MOI of Rocker Box

W_{XX} = Weight of Each Element (See Figure 42)

g = Acceleration of Gravity = 32.2 ft/sec^2

$$M_{SB} = \frac{W_{SB}}{g} = \frac{3.143 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0976 \text{ slug}$$

$$M_{BB} = \frac{W_{BB}}{g} = \frac{3.143 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0976 \text{ slug}$$

$$M_{FB} = \frac{W_{FB}}{g} = \frac{1.160 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0372 \text{ slug}$$

$$M_F = \frac{W_F}{g} = \frac{2.888 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0897 \text{ slug}$$

$$M_{PB} = \frac{W_{PB}}{g} = \frac{2.512 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0780 \text{ slug}$$

The side boards, front board, and forks are all plates parallel to the azimuth axis, whereas the base board and formica/particle board disk are perpendicular to the axis. Parallel plates may be considered as thin rods of length, L, where L is the dimension not parallel to the axis of rotation.

$$R_{SB} = \left(\frac{1}{2}(14 \text{ in}) + \frac{1}{2}(0.75 \text{ in})\right)\left(\frac{1}{12} \text{ ft/in}\right) = 0.615 \text{ ft}$$

$$R_{FB} = \left(\frac{1}{2}(14 \text{ in}) + \frac{1}{2}(0.25 \text{ in})\right)\left(\frac{1}{12} \text{ ft/in}\right) = 0.594 \text{ ft}$$

$$R_{SB} = \left(\frac{1}{2}(14 \text{ in}) - \frac{1}{2}(0.75 \text{ in})\right)\left(\frac{1}{12} \text{ ft/in}\right) = 0.552 \text{ ft}$$

$$J_{SB} \text{ (As a thin rod)} = \frac{1}{12} M_{SB} L^2 + M_{SB} R_{SB}^2$$

$$= \frac{1}{12} (0.0976 \text{ slug}) \left[(14 \text{ in}) \left(\frac{1}{12} \text{ ft/in} \right) \right]^2$$

$$+ (0.0976 \text{ slug}) (0.615 \text{ ft})^2 = 0.0479 \text{ slug-ft}^2$$

$$J_{FB} = \frac{1}{12} M_{FB} L^2 + M_{FB} R_{FB}^2$$

$$= \frac{1}{12} (0.0372 \text{ slug}) \left[(15.5 \text{ in}) \left(\frac{1}{12} \text{ ft/in} \right) \right]^2$$

$$+ (0.0372 \text{ slug}) (0.594 \text{ ft})^2 = 0.0183 \text{ slug-ft}^2$$

To determine J_F, it is first necessary to determine the mass of the semicircular cutout, M_{CUT}, and then its MOI, designated J_{CUT}. Using a side board as a reference,

$$M_{CUT} = \left(\frac{A_{CUT}}{A_{SB}} \right) M_{SB} = \left(\frac{\pi r_{CUT}^2}{2 A_{SB}} \right) M_{SB}$$

$$= \left[\frac{\pi (3.375 \text{ in})^2}{2 (14 \text{ in})^2} \right] (0.0976 \text{ slug}) = 0.0089 \text{ slug}$$

$$J_F = \frac{1}{12} \left(M_F + M_{CUT} \right) L^2 + \left(M_F + M_{CUT} \right) R_F^2$$

$$- \frac{1}{2} \left(\frac{1}{4} M_{CUT} R_{CUT}^2 + M_{CUT} R_F^2 \right)$$

$$= \frac{1}{12} \left(0.0897 \text{ slug} + 0.0089 \text{ slug} \right) (11 \text{ in})^2 \left(\frac{1}{144} \text{ ft}^2/\text{in}^2 \right)$$

$$+ \left(0.0897 \text{ slug} + 0.0089 \text{ slug} \right) (0.552 \text{ ft})^2$$

$$- \frac{1}{2} \left[\frac{1}{4} (0.0089 \text{ slug}) (3.375 \text{ in})^2 \left(\frac{1}{144} \text{ ft}^2/\text{in}^2 \right) \right.$$

$$\left. + (0.0089 \text{ slug}) (0.552 \text{ ft})^2 \right] = 0.0355 \text{ slug-ft}^2$$

The base board is considered a "perpendicular plate" as encountered as part of the tube box. Using the general formula

$$J_{\perp P} = \int_{a\,b} r^2 dM = \int_{a\,b} r^2 \left(\frac{M}{A}\right) d\,A$$

for a rectangular or square plate of length, a, and width, b, the integral was solved earlier as

$$J_{\perp P} = \frac{M_{\perp P}}{12}\left(a^2 + b^2\right), \text{ so}$$

$$J_{BB} = \frac{M_{BB}}{12}\left(a^2 + b^2\right) = \left(\frac{0.0976 \text{ slug}}{12}\right)\left((14 \text{ in})^2 + (14 \text{ in})^2\right)\left(\frac{1}{144} \text{ ft}^2/\text{in}^2\right) = 0.0221 \text{ slug-ft}^2$$

The formica/particle board can be solved with the equation,

$$J_{PB} = \frac{1}{2} M_{PB} r_{PB}^2 = \frac{1}{2} (0.0780 \text{ slug}) (7.0625 \text{ in})^2 \left(\frac{1}{144} \text{ ft}^2/\text{in}^2\right)$$

$$= 0.0135 \text{ slug-ft}^2$$

Summing all the elements together:

$$J_{RB} = J_{RB_A} = 2J_{SB} + J_{FB} + 2J_F + J_{BB} + J_{PB} = 2(0.0479 \text{ slug-ft}^2) + 0.0183 \text{ slug-ft}^2 + 2(0.0355 \text{ slug-ft}^2) + 0.0221 \text{ slug-ft}^2 + 0.0135 \text{ slug-ft}^2 = 0.221 \text{ slug-ft}^2$$

So,

$$J_{RB_A} = 0.221 \text{ slug-ft}^2$$

Altitude Gear and Motor

Approach

The actual gear and motor to be used are to be determined. However, MOI contributions of these elements can be estimated, based on the estimates shown in Table 13.

Table 13

Estimated Weight and Dimensions of Altitude Drive Assembly

	Thickness or Length (in)	Diameter (in)	Weight (lb)	Location & Orientation
Gear	0.50	7.00	1.50	See Figure 50
Motor	3.25	2.25	3.00	See Figure 50

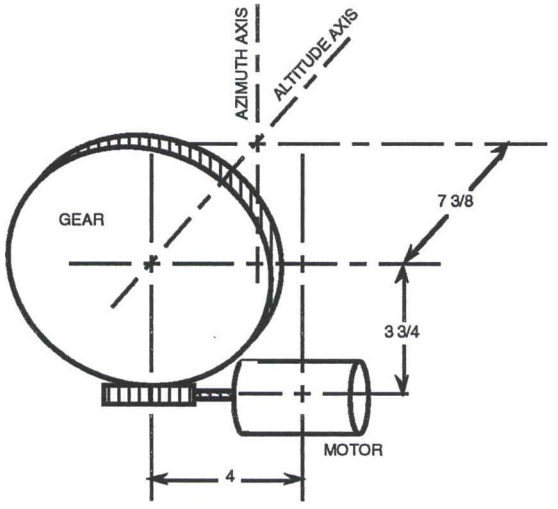


Figure 50

Expected Location/Orientation of Altitude Drive Components

Assumptions

Since the azimuth drive hardware is stationary, only the altitude hardware MOIs need be estimated. Since the altitude motor will not be rotated about the altitude axis, it is not necessary to calculate its MOI about this axis (J_{MOTOR_E}). The gear will be treated as a thin disk, while the motor will be modelled as a solid cylinder.

Calculations

Let: M_{GE} = Mass of Gear

M_{MO} = Mass of Motor

H_{GE} = Moment Distance from Axis to Center of Gear

H_{MO} = Moment Distance from Axis to Center Motor

r_{GE} = Radius of Gear = 3.50 in

r_{MO} = Radius of Motor = 1.125 in

L_{MO} = Length of Motor = 3.25 in

J_{GE} = MOI Contribution from Gear

J_{MO} = MOI Contribution from Motor

W_{GE} = Weight of Gear = 1.50 lb

W_{MO} = Weight of Motor = 3.00 lb

g = Acceleration of Gravity = 32.2 ft/sec^2

$$M_{GE} = \frac{W_{GE}}{g} = \frac{1.50 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0466 \text{ slug}$$

$$M_{MO} = \frac{W_{MO}}{g} = \frac{3.00 \text{ lb}}{32.2 \text{ ft/sec}^2} = 0.0932 \text{ slug}$$

$$H_{GE_A}{}^2 = (7.375 \text{ in})^2 \left(\frac{1}{144} \text{ ft}^2/\text{in}^2\right) = 0.378 \text{ ft}^2$$

$H_{GE_E}^2 = 0$ (Center of Mass is on Axis)

$$H_{MO_A}^2 = H_x^2 + H_z^2 = (7.375 \text{ in}^2 + 4 \text{ in}^2)\left(\frac{1}{144} \text{ ft}^2/\text{in}^2\right)$$

$$= 0.489 \text{ ft}^2$$

$$r_{GE} = (3.5 \text{ in})\left(\frac{1}{12} \text{ ft/in}\right) = 0.292 \text{ ft}$$

$$r_{MO} = (1.125 \text{ in})\left(\frac{1}{12} \text{ ft/in}\right) = 0.094 \text{ ft}$$

$$L_{MO} = (3.25 \text{ in})\left(\frac{1}{12} \text{ ft/in}\right) = 0.271 \text{ ft}$$

$$J_{GE_A} = \frac{M_{GE}\, r_{GE}^2}{4} + M_{GE}\, H_{GE_A}^2$$

$$= \frac{1}{4}(0.0466 \text{ slug})(0.292 \text{ ft})^2 + (0.0932 \text{ slug})(0.378 \text{ ft}^2)$$

$$= 0.0186 \text{ slug-ft}^2$$

$$J_{GE_E} = \frac{M_{GE}\, r_{GE}^2}{2} = \frac{1}{2}(0.0466 \text{ slug})(0.292 \text{ ft})^2 = 0.0020 \text{ slug-ft}^2$$

$$J_{MO_A} = \frac{M_{MO}\, r_{MO}^2}{4} + \frac{M_{MO}\, L_{MO}^2}{12} + M_{MO}\, H_{MO_A}^2$$

$$= (0.0932 \text{ slug})\left(\frac{1}{4}(0.094 \text{ ft})^2 + \frac{1}{12}(0.271 \text{ ft})^2\right.$$

$$\left. + 0.489 \text{ ft}^2\right) = 0.0464 \text{ slug-ft}^2$$

So,

$$J_{GE_A} = 0.019 \text{ slug-ft}^2$$

$$J_{GE_E} = 0.002 \text{ slug-ft}^2$$

$$J_{MO_A} = 0.046 \text{ slug-ft}^2$$

APPENDIX C

DERIVATION OF BOOLEAN EXPRESSIONS FOR SINUSOIDAL MAGNITUDE GENERATOR

It is necessary to design a hardware solution to convert a 6-bit word, XYABCD, representing an electrical angle into an 8-bit word, S = $(abcdefgh)_S$, where S is the sine of that function, scaled to 11111111 (255). The angle, expressed in binary form, is provided in 64 even increments over 360°, such that

$$S = \sin (5.625\ N)°$$

where N is the number of microsteps offset from the zero position. Further, let C = $(abcdefgh)_C$, where C is the cosine of that function, according to the same scale. The Truth Tables for the sine and cosine magnitude functions for the first two quadrants are given in Tables 14 and 15. Since these represent magnitude only, the last two quadrants will be a repeat of the first two. Since the cosine magnitude function contains the same values phase-shifted by 90°, inverting bit Y serves as a hardware "pointer" to find the correct data. Thus, the same logic can be implemented for both functions. Since the values in the lower half of each table are repeated, binary subtraction can be used to point to these. Therefore, only the values in one half of one table are truly unique. It should be noted that a

Table 14

Truth Table for Sine Magnitude Generator

INPUTS						SINE OUTPUTS								
X	Y	A	B	C	D	Mag	a_S	b_S	c_S	d_S	e_S	f_S	g_S	h_S
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	25	0	0	0	1	1	0	0	1
0	0	0	0	1	0	50	0	0	1	1	0	0	1	0
0	0	0	0	1	1	74	0	1	0	0	1	0	1	0
0	0	0	1	0	0	98	0	1	1	0	0	0	1	0
0	0	0	1	0	1	120	0	1	1	1	1	0	0	0
0	0	0	1	1	0	142	1	0	0	0	1	1	1	0
0	0	0	1	1	1	162	1	0	1	0	0	0	1	0
0	0	1	0	0	0	182	1	1	1	1	0	1	0	0
0	0	1	0	0	1	197	1	1	0	0	0	1	0	1
0	0	1	0	1	0	212	1	1	0	1	0	1	0	0
0	0	1	0	1	1	225	1	1	1	0	0	0	0	1
0	0	1	1	0	0	236	1	1	1	0	1	1	0	0
0	0	1	1	0	1	244	1	1	1	1	0	1	0	0
0	0	1	1	1	0	250	1	1	1	1	1	0	1	0
0	0	1	1	1	1	254	1	1	1	1	1	1	1	0
0	1	0	0	0	0	255	1	1	1	1	1	1	1	1
0	1	0	0	0	1	254	1	1	1	1	1	1	1	0
0	1	0	0	1	0	250	1	1	1	1	1	0	1	0
0	1	0	0	1	1	244	1	1	1	1	0	1	0	0
0	1	0	1	0	0	236	1	1	1	0	1	1	0	0
0	1	0	1	0	1	225	1	1	1	0	0	0	0	1
0	1	0	1	1	0	212	1	1	0	1	0	1	0	0
0	1	0	1	1	1	197	1	1	0	0	0	1	0	1
0	1	1	0	0	0	182	1	0	1	1	0	1	0	0
0	1	1	0	0	1	162	1	0	1	0	0	0	1	0
0	1	1	0	1	0	142	1	0	0	0	1	1	1	0
0	1	1	0	1	1	120	0	1	1	1	1	0	0	0
0	1	1	1	0	0	98	0	1	1	0	0	0	1	0
0	1	1	1	0	1	74	0	1	0	0	1	0	1	0
0	1	1	1	1	0	50	0	0	1	1	0	0	1	0
0	1	1	1	1	1	25	0	0	0	1	1	0	0	1
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 15

Truth Table for Cosine Magnitude Generator

INPUTS						COSINE OUTPUTS								
X	Y	A	B	C	D	Mag	a_C	b_C	c_C	d_C	e_C	f_C	g_C	h_C
0	0	0	0	0	0	255	1	1	1	1	1	1	1	1
0	0	0	0	0	1	254	1	1	1	1	1	1	1	0
0	0	0	0	1	0	250	1	1	1	1	1	0	1	0
0	0	0	0	1	1	244	1	1	1	1	0	1	0	0
0	0	0	1	0	0	236	1	1	1	0	1	1	0	0
0	0	0	1	0	1	225	1	1	1	0	0	0	0	1
1	0	0	1	1	0	212	1	1	0	1	0	1	0	0
1	0	0	1	1	1	197	1	1	0	0	0	1	0	1
1	0	1	0	0	0	182	1	0	1	1	0	1	0	0
1	0	1	0	0	1	162	1	0	1	0	0	0	1	0
1	0	1	0	1	0	142	1	0	0	0	1	1	1	0
1	0	1	0	1	1	120	0	1	1	1	1	0	0	0
1	0	1	1	0	0	98	0	1	1	0	0	0	1	0
1	0	1	1	0	1	74	0	1	0	0	1	0	1	0
1	0	1	1	1	0	50	0	0	1	1	0	0	1	0
1	0	1	1	1	1	25	0	0	0	1	1	0	0	1
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	1	25	0	0	0	1	1	0	0	1
0	1	0	0	1	0	50	0	0	1	1	0	0	1	0
0	1	0	0	1	1	74	0	1	0	0	1	0	1	0
0	1	0	1	0	0	98	0	1	1	0	0	0	1	0
0	1	0	1	0	1	120	0	1	1	1	1	0	0	0
0	1	0	1	1	0	142	1	0	0	0	1	1	1	0
0	1	0	1	1	1	162	1	0	1	0	0	0	1	0
0	1	1	0	0	0	182	1	1	1	1	0	1	0	0
0	1	1	0	0	1	197	1	1	0	0	0	1	0	1
0	1	1	0	1	0	212	1	1	0	1	0	1	0	0
0	1	1	0	1	1	225	1	1	1	0	0	0	0	1
0	1	1	1	0	0	236	1	1	1	0	1	1	0	0
0	1	1	1	0	1	244	1	1	1	1	0	1	0	0
0	1	1	1	1	0	250	1	1	1	1	1	0	1	0
0	1	1	1	1	1	254	1	1	1	1	1	1	1	0
1	0	0	0	0	0	255	1	1	1	1	1	1	1	1

special dedicated logic circuit will need to be added to invert the values for an input of 010000 to distinguish it from 000000.

To solve for the Boolean expressions, Karnaugh maps were created for each of the eight output bits for the 000000-001111 range of input values (Tables 16 and 17).

Table 16

Karnaugh Maps for a_S, b_S, c_S, d_S

a_S =

CD \ AB	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	0	1	1	1
10	0	1	1	1

b_S =

CD \ AB	00	01	11	10
00	0	1	1	0
01	0	1	1	1
11	1	0	1	1
10	0	0	1	1

c_S =

CD \ AB	00	01	11	10
00	0	1	1	1
01	0	1	1	0
11	0	1	1	1
10	1	0	1	0

d_S =

CD \ AB	00	01	11	10
00	0	0	0	1
01	1	1	1	0
11	0	0	1	0
10	1	0	1	1

Table 17

Karnaugh Maps for e_S, f_S, g_S, h_S

e_S =

CD \ AB	00	01	11	10
00	0	0	1	0
01	1	1	0	0
11	1	0	1	0
10	0	1	1	0

f_S =

CD \ AB	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	0	0	1	0
10	0	1	0	1

g_S =

CD \ AB	00	01	11	10
00	0	1	0	0
01	0	0	0	0
11	1	1	1	0
10	1	1	1	0

h_S =

CD \ AB	00	01	11	10
00	0	0	0	0
01	1	0	0	1
11	0	0	0	1
10	0	0	0	0

The Boolean expression for a_S is reasonably obvious:

$$a_S = A + BC$$

Minimization of Boolean expressions for the other functions represented above were achieved by use of the Quine-McCluskey algorithm (see Tables 18-24).

Table 18

Quine-McCluskey Algorithm for b_S

b_S:

1's	Minterms		
1	m_4	0100	√
2	m_3	0011	√
2	m_5	0101	√
2	m_9	1001	√
2	m_{10}	1010	√
2	m_{12}	1100	√
3	m_{11}	1011	√
3	m_{13}	1101	√
3	m_{14}	1110	√
4	m_{15}	1111	√

1-Cubes		
4,5	010X	√
4,12	X100	√
3,11	X011	*
5,13	X101	√
9,11	10X1	√
9,13	1X01	√
10,11	101X	√
10,14	1X10	√
12,13	110X	√
12,14	11X0	√
11,15	1X11	√
13,15	11X1	√
14,15	111X	√

2-Cubes		
4,5,12,13	X10X	*
9,11,13,15	1XX1	*
10,11,14,15	1X1X	*
12,13,14,15	11XX	*

	Minterm Table	3	4	5	9	10	11	12	13	14	15
*	3,11	√					√				
*	4,5,12,13		√	√				√	√		
*	9,11,13,15				√		√		√		√
*	10,11,14,15					√	√			√	√
	12,13,14,15							√	√	√	√

$$b_S = \bar{B}CD + B\bar{C} + AD + AC$$

Table 19

Quine-McCluskey Algorithm for c_S

c_S:

1's	Minterms		
1	m_2	0010	*
1	m_4	0100	√
1	m_8	1000	√
2	m_5	0101	√
2	m_{12}	1100	√
3	m_7	0111	√
3	m_{11}	1011	√
3	m_{13}	1101	√
3	m_{14}	1110	√
4	m_{15}	1111	√

1-Cubes		
4,5	010X	√
4,12	X100	√
8,12	1X00	*
5,7	01X1	√
5,13	X101	√
12,13	110X	√
12,14	11X0	√
7,15	X111	√
11,15	1X11	*
13,15	11X1	√
14,15	111X	√

2-Cubes		
4,5,12,13	X10X	*
5,7,13,15	X1X1	*
12,13,14,15	11XX	*

	Minterm Table	2	4	5	7	8	11	12	13	14	15
*	2	√									
*	8,12					√		√			
*	11,15						√				√
*	4,5,12,13		√	√				√	√		
*	5,7,13,15			√	√				√		√
*	12,13,14,15							√	√	√	√

$$c_S = \bar{A}\bar{B}C\bar{D} + A\bar{C}\bar{D} + ACD + B\bar{C} + BD + AB$$

Table 20

Quine-McCluskey Algorithm for d_S

d_S:

1's	Minterms		
1	m_1	0001	√
1	m_2	0010	√
1	m_8	1000	√
2	m_5	0101	√
2	m_{10}	1010	√
3	m_{13}	1101	√
3	m_{14}	1110	√
4	m_{15}	1111	√

1-Cubes		
1,5	0X01	*
2,10	X010	*
8,10	10X0	*
5,13	X101	*
10,14	1X10	*
13,15	11X1	*
14,15	111X	*

2-Cubes	
NONE	

	Minterm Table	1	2	5	8	10	13	14	15
* * *	1,5	√		√					
* * *	2,10		√			√			
* * *	8,10				√	√			
*	5,13			√			√		
*	10,14					√		√	
* *	13,15						√		√
* *	14,15							√	√

$$d_S = \bar{A}\bar{C}D + \bar{B}C\bar{D} + A\bar{B}\bar{D} + B\bar{C}D + ABC$$

or " + " + " + $AC\bar{D}$ + ABD

or " + " + " + ABC + ABD

Table 21

Quine-McCluskey Algorithm for e_S

e_S:

1's	Minterms		
1	m_1	0001	√
2	m_3	0011	√
2	m_5	0101	√
2	m_6	0110	√
2	m_{12}	1100	√
3	m_{14}	1110	√
4	m_{15}	1111	√

1-Cubes		
1,3	00X1	*
1,5	0X01	*
6,14	X110	*
12,14	11X0	*
14,15	111X	*

2-Cubes	
NONE	

	Minterm Table	1	3	5	6	12	14	15
*	1,3	√	√					
*	1,5	√		√				
*	6,14				√		√	
*	12,14					√	√	
*	14,15						√	√

$$e_S = \bar{A}\bar{B}D + \bar{A}\bar{C}D + BC\bar{D} + AB\bar{D} + ABC$$

Table 22

Quine-McCluskey Algorithm for f_S

f_S:

1's	Minterms		
1	m_8	1000	√
2	m_6	0110	*
2	m_9	1001	√
2	m_{10}	1010	√
2	m_{12}	1100	√
3	m_{13}	1101	√
4	m_{15}	1111	√

1-Cubes		
8,9	100X	√
8,10	10X0	*
8,12	1X00	√
9,13	1X01	√
12,13	110X	√
13,15	11X1	*

2-Cubes		
8,9,12,13	1X0X	*

	Minterm Table	6	8	9	10	12	13	15
*	6	√						
*	8,10		√		√			
*	13,15						√	√
*	8,9,12,13		√	√		√	√	

$$f_S = \bar{A}BC\bar{D} + A\bar{B}\bar{D} + ABD + A\bar{C}$$

Table 23

Quine-McCluskey Algorithm for g_S

g_S:

1's	Minterms		
1	m_2	0010	√
1	m_4	0100	√
2	m_3	0011	√
2	m_6	0110	√
3	m_7	0111	√
3	m_{14}	1110	√
4	m_{15}	1111	√

1-Cubes		
2,3	001X	√
2,6	0X10	√
4,6	01X0	*
3,7	0X11	√
6,7	011X	√
6,14	X110	√
7,15	X111	√
14,15	111X	√

2-Cubes		
2,3,6,7	0X1X	*
6,7,14,15	X11X	*

	Minterm Table	2	3	4	6	7	14	15
*	4,6			√	√			
*	2,3,6,7	√	√		√	√		
*	6,7,14,15				√	√	√	√

$$g_S = \bar{A}B\bar{D} + \bar{A}C + BC$$

Table 24

Quine-McCluskey Algorithm for h_S

h_S:

1's	Minterms		
1	m_1	0001	√
2	m_9	1001	√
3	m_{11}	1011	√

1-Cubes		
1,9	X001	*
9,11	10X1	*

2-Cubes	
NONE	

$$h_S = \bar{B}\bar{C}D + A\bar{B}D$$

BIBLIOGRAPHY

1. Aerotech. Motion Control Product Guide. Pittsburgh: Aerotech, 1990.

2. Bishop, Ron. Basic Microprocessors and the 6800. Hasbrouck Heights, NJ: Hayden Book Company, 1979.

3. Boyd, Louis J. "Automating a Telescope." Byte, July 1985, 227.

4. Brodie, Leo. Starting FORTH. London: Prentice Hall, 1981.

5. Buchsbaum, Frank, Ferdinand Freudenstein, and Peter J. Thornton. Design and Application of Small Standardized Components. New Hyde Park, NY: Stock Drive Products, 1983.

6. Burton, Bill. "Ready, Compute, Aim." Astronomy, July 1988, 71.

7. Compumotor Division. Positioning Control Systems and Drives. Rohnert Park, CA: Parker Hannifin Corporation, 1992.

8. Computer Systems Division. Commodore 64 Programmer's Reference Guide. Wayne, PA: Commodore Business Machines, 1982.

9. Davis, Richard M. Thesis Projects in Science and Engineering. New York: St. Martins Press, 1980.

10. Division of Designatronics. Motion Control Products Catalog, SDP-823. New Hyde Park, NY: Stock Drive Products, 1987.

11. Dodson, Stephen. "A 22-Inch Equatorial Dobsonian." Sky and Telescope, August 1984, 167.

12. Eastern Air Devices. Stepping Motors, Linear Actuators. Dover, NH: Eastern Air Devices.

13. Electromechanical Controls Branch. Solutions in Motion, Technical Seminar Series Slide Handouts, set 96-013074-01,

Rev B, and accompanying notes. Petaluma, CA: Parker Hannefin, 1992.

14. Fitzgerald, Arthur Eugene, David E. Higginbotham, and Arvin Grabel. Basic Electrical Engineering. McGraw-Hill series in electrical engineering: Networks and systems, ed. Stephen W. Director. New York: McGraw-Hill, 1981.

15. Genet, Russell M.. "Updating the Oldest Science." Byte, July 1985, 179.

16. Genet, Russell M., Louis J. Boyd, and Mark Trueblood. "Stepper Motor Control of Telescopes" Sky and Telescope, April 1985, 350.

17. Graf, Rudolf F. Modern Dictionary of Electronics. Indianapolis: Howard W. Sams & Co., 1972.

18. Halliday, David, and Robert Resnick. Fundamentals of Physics. New York: John Wiley & Sons, 1974.

19. Hayt, William H., Jr., and Jack E. Kemmerly. Engineering Circuit Analysis. New York: McGraw-Hill, 1978.

20. Hill, Frederick J., and Gerald R. Peterson. Switching Theory and Logical Design. New York: John Wiley & Sons, 1981.

21. Holbrook, John. "A Stepper Motor for Your Telescope." Sky and Telescope, July 1986, 80.

22. Holland, Earle. "Two-shooter Guns for Far-off Stars." OSU Quest, Summer 1986, 6.

23. Hwang, Kai and Faye A, Briggs. Computer Architecture and Parallel Processing. McGraw-Hill series in computer organization and architecture, ed. Eric M. Munson and Jonathan Palace. New York: McGraw-Hill, 1984.

24. Leiker, Steve. "How to Half-Step a Stepper Motor." Sky and Telescope, January 1988, 80.

25. Marino, Chris. "Selecting Step Motors for Incremental Motion Applications." Motion, November/December 1986, 3.

26. Meeus, Jean. Astronomical Formulae for Calculators. Richmond: Willmann-Bell, 1988.

27. Merchant, Michael J., and John R. Sturgul. Applied FORTRAN Programming with Standard FORTRAN, WATFOR, WATFIV, and Structured WATFIV. Belmont, CA: Wadsworth Publishing Company, 1977.

28. Meshkat, Saeid. "Servo System Design, A Tutorial Study." Motion, July/August 1986, 28.

29. Millman, Jacob. Microelectronics: Digital and Analog Circuits and Systems. McGraw-Hill series in electrical engineering: Electronics and circuits. New York: McGraw-Hill, 1981.

30. Mims, Forrest M., III. "Controller for Small DC Motors." Popular Electronics, October 1982, 98.

31. Mims, Forrest M., III. Engineer's Mini-Notebook, Basic Semiconductor Circuits. Forest M. Mims, 1986.

32. Mims, Forrest M., III. Engineer's Mini-Notebook, Digital Logic Circuits. Forest M. Mims, 1986.

33. Mims, Forrest M., III. Engineer's Mini-Notebook, 555 Circuits. Forest M. Mims, 1992.

34. Mims, Forrest M., III. Engineer's Mini-Notebook, Op-Amp Circuits. Forest M. Mims, 1985.

35. Mims, Forrest M., III. Engineer's Mini-Notebook, Optoelectronics Circuits. Forest M. Mims, 1986.

36. Mims, Forrest M., III. "The New Power FETs." Popular Electronics, February 1982, 94.

37. National Semiconductor Corporation. CMOS Logic Databook. Santa Clara, CA: National Semiconductor Corporation, 1988.

38. National Semiconductor Corporation. Linear Databook. Santa Clara, CA: National Semiconductor Corporation, 1982.

39. National Semiconductor Corporation. Programmable Logic Devices Databook and Design Guide. Santa Clara, CA: National Semiconductor Corporation, 1993.

40. Oriental Motor Company. Oriental Motor Catalogue, Cat. No. 83004A. Torrance, CA: Oriental Motor Company, 1983(?).

41. Osborne, Adam. An Introduction to Microcomputers, vol. 1, Basic Concepts. Berkeley: Osborne/McGraw-Hill, 1980.

42. Overcash, Don. "Position Sensor Technology." Motion, January/February 1987, 14.

43. Parker Compumotor. Solutions in Motion Technology Videos. Produced and Directed by Leonard Luna, 58 min. Parker Hannefin Corporation, 1992. Videocassette.

44. Pavlat, Joseph. "A New Look at Microstepping." Machine Design, March 23, 1989.

45. PMI Motion Technologies. Motion Control Components, Short Form Catalog. Commack, NY: PMI Motion Technologies, 1988.

46. Portescap. Escap Motion Systems. La Chaux-de-Fonds 1 Suisse: Portescap, 1989(?).

47. Radio Shack. Semiconductor Reference Guide, Cat. No. 276-4010, Fort Worth: Tandy Corporation, 1986.

48. Raven, Francis H. Automatic Control Engineering. McGraw-Hill series in electrical engineering, ed. Jack P. Holman. New York: McGraw-Hill, 1978.

49. Ravi, S. "The 8-Bit Parallel Interface." Analog Computing, July 1986, 19.

50. RCA Solid State Division. SK Series Solid State Replacement Guide, Pub. No. SKG202D. Somerville, NJ: RCA Corporation, 1985.

51. Robbins Myers Motor & Control Systems Division. Encoderstock Distributor Catalog. Goleta, CA: Renco Corporation.

52. SGS Thompson. GAL Programmable Logic Devices Databook. Phoenix: SGS Thompson Microelectronics, 1992.

53. Sinnott, Roger W. "Taming Our Chaotic Calendar." Sky and Telescope, May 1984, 454.

54. Sinnott, Roger W. "Where Is It in the Sky?" Sky and Telescope, June 1984, 558.

55. Stone, Robert T., and Howard M. Berlin. Design of VMOS Circuits, with Experiments. Indianapolis: Howard W. Sams & Co., 1980.

56. Superior Electric Company. Slo-Syn DC Stepping Motors. Bristol: Superior Electric Company, 1979.

57. Superior Electric Company. Slo-Syn Stepping Motor Controls. Bristol: Superior Electric Company, 1979.

58. Taft, C. K., and R.G. Gauthier, Stepping Motor System Design and Analysis. Durham, NH: University of New Hampshire, 1988.

59. Taki, Toshimi. "A New Concept in Computer-Aided Telescopes." Sky and Telescope, February 1989, 194.

60. Tal, Jacob. "Tal Talks - Microstepping with DC Motors." Motion, January/February 1987, 26.

61. Tal, Jacob. "Tal Talks - Modular Approach to Motion Control." Motion, May/June 1986, 14.

62. Tal, Jacob. "Tal Talks - Quantization Errors in Digital Servo Systems." Motion, September/October 1986, 10.

63. Tarabian, Kate L. A Manual for Writers of Term Papers, Theses, and Dissertations. Chicago: University of Chicago Press, 1987.

64. Tardiff, Alphonse. "The Amazing Stepper Motor." Sky and Telescope, May 1989, 554.

65. Texereau, Jean. How to Make a Telescope. Richmond: Willmann-Bell, 1984.

66. Trueblood, Mark, and Russell M. Genet. Microcomputer Control of Telescopes. Richmond: Willmann-Bell, 1985.

67. TRW Electronic Components Group. [Stepping Motor] Specifications. Dayton: TRW Motor Division.

68. Van Slyke, Paul B. "Four-Speed 17 1/2-Inch Newtonian." Sky and Telescope, February 1986, 199.