Mobile Star Finder

Nathan Duthie, Brian P. DeJong (advisor)
Central Michigan University, duthi1nk@cmich.edu, b.dejong@cmich.edu

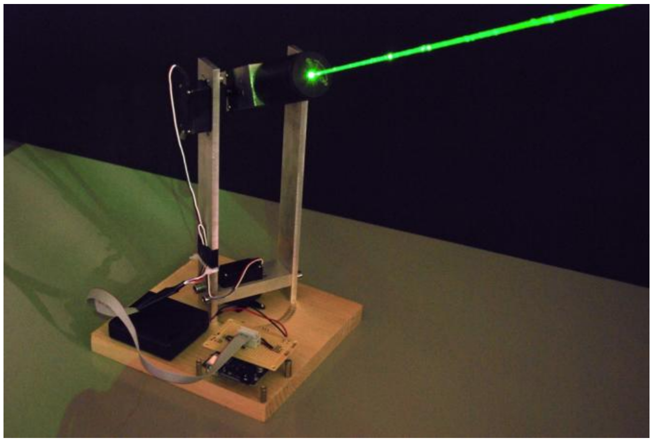

FIGURE 1
THE MOBILE STAR FINDER.

***Abstract* – This paper presents the build, construction, and control of the Mobile Star Finder. Not all people are as good as others when it comes to locating objects in the sky. That is where the Mobile Star Finder comes in. It is a device that can be connected to a laptop and the user can input a predetermined code number that will initiate calculations through a microcontroller. The calculations convert celestial coordinates to the horizon coordinates of the object. Once the MSF is fully operational it will allow for accurate locating of celestial objects. This project is still not complete. There is still much work to be done on the program that will run and control the motion of the MSF. Design changes are also likely to come for better functionality. Once all work is done the user could use it in locating objects on a regular basis and improve their ability to locate these objects without this aid at a later time.**

Index Terms – astronomy, star finder, mechatronic, servomotor

INTRODUCTION

Astronomy enthusiasts, students, and casual stargazers alike should welcome any help in locating specific stars or celestial objects in the night sky. The goal of the Mobile Star Finder (MSF) is just that: to help amateur astronomers locate desired stars by pointing to them with a visible-light laser. The main application of the MSF is naked-eye observation, but it can be used to sight a telescope as well. This paper describes the background, mechanical design, and control of the MSF as an out-of-class summer engineering project.

BACKGROUND

Currently, there are no high-quality star-pointing devices available for amateur astronomers. Two toy-like devices exist but are low-quality and geared towards children [1][2]. Both of these devices have a small speaker for audible descriptions of celestial objects and quick multiple-star "tours" of the night's sky. Furthermore, these devices require the user to hold them the entire time. One of the devices does have a method of guiding the user to a particular object through indicator lights inside the viewfinder, but it does not automatically locate the star.

Most amateur astronomers and star enthusiasts have a several ways of locating celestial objects. Some have located objects for so long that they just know where to look, others use star charts, such as a planisphere; a chart calibrated to a particular general location (usually a range of latitudes). This type of chart can be set to a date and time and gives the user a visual window of what is in the sky at that day and time; thereby allows the user to orientate themselves so that they can locate the object of interest (see Figure 2). If the observer is using a telescope they may have the opportunity to use a self guiding telescope. These telescopes have motors attached to the telescope and have a (usually hand held) module that allows the user to input either coordinates, or is a catalog type system, or even both options are available.

FIGURE 2
Planisphere

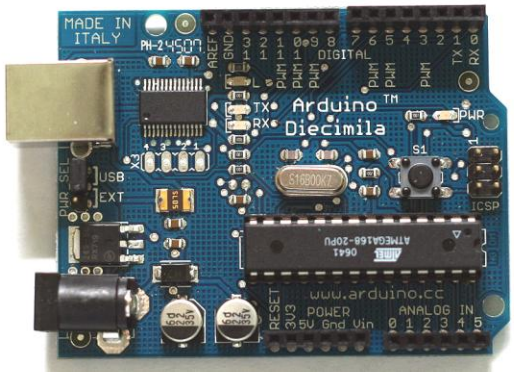

FIGURE 3
THE ARDUINO MICROCONTROLLER.

There are two basic types of telescope mounts, Equatorial and Alt-azimuth [3]. These types are based on how it allows the telescope to locate a star. In the case of the Equatorial mount it uses the normal celestial coordinates of Right Ascension and Declination. While the Alt-azimuth mount uses Horizon coordinates. Each type can be achieved by various variations; such as a fork with either mount, ball and socket or rocker box for Alt-azimuth, and a German Equatorial mount [3].

The goal of the MSF is to be tabletop and self-actuated, and thereby better for a longer night of observing and learning the locations of objects of interest.

DEVICE DESCRIPTION

The MSF is shown in Figure 1. It consists of a visible green laser mounted in a two degrees-of-freedom apparatus with two servomotors controlled by an Arduino microcontroller. One motor controls altitude while the other controls azimuth. At the top of the apparatus is the mount for the

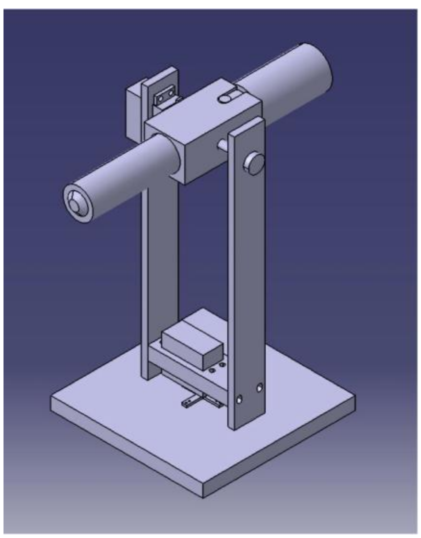

FIGURE 4
CATIA MODEL OF THE MACHINED COMPONENTS

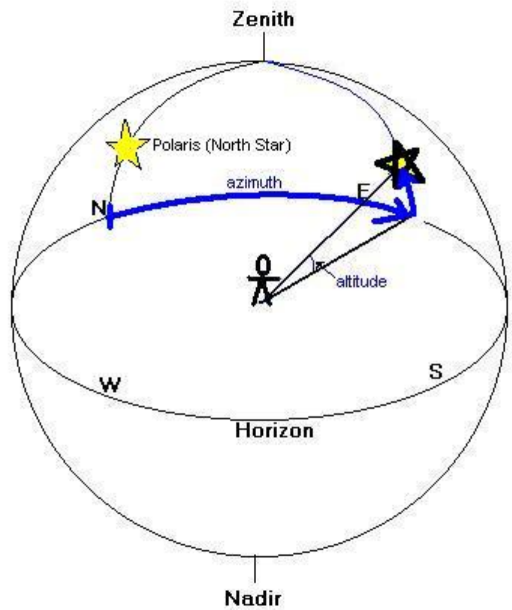

FIGURE 5
HORIZON COORDINATE SYSTEM

visible laser, designed specifically for the current laser.

I. Mechanical Design

For simplicity, the device is a forked Alt-azimuth, mount, as forked mounts are commonly used for telescopes (see Figure 4 for a CATIA sketch). Alt-azimuth mounts use Horizon coordinates which consist of an up/down motion (altitude) and a left/right motion (azimuth) to locate an object in the sky, see Figure 5. To articulate these motions the forked design was used so that the laser would be in the center supported by the arms. The two motions are each actuated by a servomotor – the altitude motor is located in one arm of the fork, while the azimuth motor is located underneath the fork.

The bottom of the device is a rectangular wood block with a large hole drilled through the center and a clearance slot for the push button on/off switch.

The fork and base are made out of aluminum. Aluminum was chosen because it's relatively lightweight, easy to machine, and readily available.

All design, fabrication and construction was done in-house at CMU. The design was done on CATIA, and the parts were machined by hand in the machine shop.

II. Control

The motion of the device is controlled by an Arduino microcontroller [4]; see Figure 3. The user chooses the star or body of interest and inputs the corresponding identifier on a laptop. The Arduino takes that input via a serial port, and calculates the body's corresponding altitude and azimuth for the MSF, and sends the signals to the servomotors. Through these calculations the microcontroller has converted normal celestial coordinates into more useful coordinates for ground based observing. Celestial coordinates were made so that the coordinates themselves do not change other time and to be the same for all locations. However, being based on the surface of Earth the objects in the sky are not stationary. To locate an object the coordinates are based on the observer's location and the time. As time goes on the coordinates of the object also changes. Two separate locations on Earth will give different coordinates even with the same time.

III. Calculations

The calculations that the microcontroller performs are as follows.

For time and subsequent calculations there are a few things that are needed: Julian date (JD), Local Sidereal Time (LST), and Universal Time (UT). The JD is simply the number of days since noon on January 1, 4713 BC in Greenwich, England [5]. For the calculations, two JDs are needed: J2000 (the JD for noon, Jan. 1, 2000) and the other is the current JD (the date of the observations). These dates can be calculated but it is usual to look them up on the Internet or in an astronomical book.

LST is the time required in a particular location for the stars to reach a default position in the night sky. Since stars do not have to travel as far as the sun to reach the same position in the sky a local sidereal day is about 4 minutes shorter than a normal 24 hour day [6].

Universal time is the time for Greenwich, England.

The following formula gives the approximate LST [6]:

$$\mathrm{LST} = 100.46 + 0.985647d + l + 15UT, \qquad (1)$$

where d is the decimal days since J2000, l is the longitude as a decimal in degrees, and UT is the UT in decimal hours

The program then calculates the angle to the body from the body's normal coordinates of Right Ascension (RA) and Declination (Dec). RA is measured in hours and minutes so this is converted to degrees, easily done by multiplying by 15. This is because there are 15 degrees of arc for every hour. Dec is measured in degrees and minutes so this is converted to decimal degrees. To do the calculations RA is converted to an Hour Angle (HA), since the HA increases as the LST increases; the Dec stays the same since it measures the angle from the celestial equator. The angle between the observer's meridian and the hour circle on which the celestial body lies is the defined HA (see Figure 6) [7]. The HA is determined by:

$$\mathrm{HA} = \mathrm{LST} - \mathrm{RA}. \qquad (2)$$

The Altitude (Alt) can now be found by:

$$\begin{aligned}\sin(\mathrm{Alt}) &= \sin(\mathrm{Dec})\cdot\sin(\mathrm{Lat}) \\ &\quad + \cos(\mathrm{Dec})\cdot\cos(\mathrm{Lat})\cdot\cos(\mathrm{HA}),\end{aligned} \qquad (3)$$

where Lat is the devices latitude. Finally, Azimuth (Az) is calculated using:

$$\cos(\mathrm{Az}) = \frac{\sin(\mathrm{Dec}) - \sin(\mathrm{Alt})\cdot\sin(\mathrm{Lat})}{\cos(\mathrm{Alt})\cdot\cos(\mathrm{Lat})}. \qquad (4)$$

The Arduino performs the calculations when the user selects a celestial object on the attached computer. Within the code is a small library of objects (currently only stars). The program uses a simple index to find an object: when the user inputs an index, such as "3", the program locates and points to the object designated as "3". The information that is sent to the Arduino from the computer are the UT and JD corresponding to that time; this is done because to accurately calculate the position the time needs to be accurate since the HA, which will effect Alt which effects azimuth, changes over time. Once these are sent, the program keeps track of how many milliseconds have passed since the starting of the code and thereby keeps time for as long as the code has been running, once power is removed the time is no longer valid. Then, the Arduino does the calculations and sends signals to the servomotors.

CONCLUSION AND FUTURE WORK

The programming of the MSF is still ongoing. As of now, the MSF is movable and the equations and internal index are defined.

The overall design and concept is functional but there is much room for improvements. To reduce machining time and to keep things simple, the fork slides freely on the base. Also, the base is just a piece of wood. Unfortunately, there is significant friction between the wood and the arms. This is not entirely bad; this does keep the rotation speed down, but will not be in the final design. The laser that is currently being used is a ~150 mW 532 nm Aries-150 portable laser [8]. This is a very nice laser and the beam is very good, but it is rather large. Because of its size, the laser wobbles when moving. This wobbling can be reduced if the speed of the servomotor is reduced. A fix to this problem would be to use a smaller laser or an entirely different

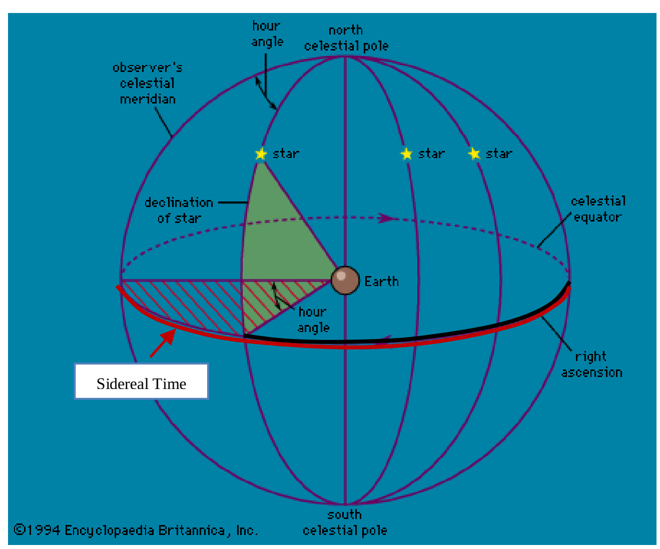

FIGURE 6
HOUR ANGLE [7]

component to control movement.

Work will continue on this project to finalize the program and start testing for functionality and accuracy. Design changes are likely but still to be determined. For example, it would be nice to make the device self-contained so a laptop is not necessary. Another idea is to make it more marketable to the general public, or for classroom use.

Overall, the project was a success and a useful learning experience. I am looking forward to future work with new additions and improvements to the design and functionality.

ACKNOWLEDGMENT

This project was funded by the School of Engineering and Technology at Central Michigan University.

REFERENCES

[1] Amazon. Web. May 2009. <http://www.amazon.com/gp/product/B000CNPAAA/ref=s9_k2a_gw_tr03?pf_rd_m=ATVPDKIKX0DER&pf_rd_s=center-4&pf_rd_r=1BBEC1C96P21ZX7KTPVE&pf_rd_t=101&pf_rd_p=470939031&pf_rd_i=507846>.

[2] Setelescope. Web. May 2009. <http://www.setelescope.com/myskyfrommeade.aspx>

[3] How Stuff Works. Web. Feb. 2010. <http://science.howstuffworks.com/telescope5.htm>.

[4] Arduino. Web. July 2009. <http://www.arduino.cc/>.

[5] Google. Web. Feb. 2010. <http://www.google.com/search?hl=en&q=define:+Julian+Date&aq=0&aqi=l1g10&oq=what+is+julian>.

[6] *Stargazing*. Web. Aug. 2009. <http://www.stargazing.net/kepler/altaz.html>.

[7] Encyclopedia Britannica. Web. Feb. 2010. <http://www.britannica.com/EBchecked/topic/273010/hour-angle>.

[8] LaserGlow. Web. Aug. 2009. <http://enthusiast.laserglow.com/catalog.htm>.

AUTHOR INFORMATION

Nathan Duthie is a second-year undergraduate mechanical engineering student at Central Michigan University. He has a previous degree in physics with a concentration in astronomy. Astronomy has been an interest now for many years. The ability to be able to work on my own idea and concept and also being astronomy related is quite an experience; both educational and fun. He can be contacted through the School of Engineering and Technology, Central Michigan University, Mount Pleasant, MI, 48859, or at duthi1nk@cmich.edu.

Brian P. DeJong is an Assistant Professor of Mechanical Engineering at Central Michigan University, with a Ph.D. in mechanical engineering from Northwestern University. His primary research interest is in using auditory occupancy grids with a mobile robot, although he has concurrent research in robotics, teleoperation, human-robot interfaces, lower-limb exercise robots, and haptics. He can be contacted through the School of Engineering and Technology, Central Michigan University, Mount Pleasant, MI, 48859, or at b.dejong@cmich.edu .