

Moon Tracker

By ... Tim Meyers
Email: tmeyers33@comcast.net

Programs in this package:

- *Manual Scan4a.bs2*; This program lets you ‘trim’ the Moon tracker and setup the EEPROM for collecting readings by the next program. Additionally, you can study the sensor heads output real time on an Excel graph, or move the head with manual overrides via the scroll bars on the control dialog. In addition to all the above, the program is also used to take the first reading of the moon. The next program will begin its tracking operation using this first found point in the sky. **Caution**. Do not load this program before downloading the previous night’s data collection. The mere act of loading this program clears the old readings stored there!
- *Lunar_Scan.bs2*; This program uses the first reading collected by the tracker from the previous program above, to begin collecting readings ... about every 5 minutes – at present. Once loaded, it will operate on its own. It’s designed to maximize the battery life, by entering the *sleep* mode (almost zero current) during each 5 minute pause. You **do not** need to leave the lap-top connected for the tracker to operate. However, you can leave it connected to watch the debug messages that are generated by the trackers activity. Such as, “Waking Up ... I need my coffee!” or, “Writing Head position.” Etc. It does not support the Excel *graphing* and *dialog* over-rides available in any of the *manual scan* program versions.
- *Lunar Data Downloader.bs2*; This program transfers the collected readings out of EEPROM and into Excel for storage and study.

Introduction

When I originally decided to build a robot, it was at my daughter’s request that I help her “build a robot.” So I started where everybody these days begins, and I looked online for a kit. However, nothing I saw really *tickled my fancy*.

I was faced with an underlying question, “What form should my robot take on?” If I looked at existing robots, most provide some means of remotely operating them. They either play back commands that make something (industrial) or operate in environments that are too harsh¹ for humans to enter, or they entertain us, as with ‘*battle bots*’ that try to destroy each other in a cage, like modern day ‘gladiators’ that suffer no injury. In every case, they are operated (at least in part) by a *remote* of some sort.

¹ Like deep sea robots

None of these were acceptable! What I wanted to construct, was a device that *resolved* all its own movements and decisions, absent *any external intelligence*. Additionally, the robot should not be so complex that the builder would lose sight of the *forest* for the *trees*. That was a difficult prerequisite to satisfy. Until, I came across an individual who had used a [single CdS cell](#) to measure the light of the moon.

I realized in an instant I had enough spare parts to quickly build a device that **would have to be robotic**, in order to *track* the light of the moon. So that was the birth of my moon tracker. Once constructed, it was unable to track the moon. It needed programming to perform the *tracking* operation. This was exactly what I wanted from my robot!

To put it poetically:

I wasn't so much interested in tracking the MOON, as I was in TRACKING the moon.

