

# Computerization of a telescope at secondary education

Alfonso García Santiago and José Martos Jumillas

I.E.S Iulia Salaria SABIOTE (Jaén)

## Abstract

The work we are presenting in this paper is the computerization of a refractor telescope on an EQ3 type equatorial mount through Arduino. The control of the mount is done via three different interfaces: Stellarium, an Android interface for mobile phones and a second interface for PC made with Processing.

The aforementioned work was done by the authors with a double purpose: presenting the interest in astronomy in the Mathematics department, and the development of applications within the subject of Technology in 4th ESO. So, it is a collaborative project between both departments. Except for the telescope and the mount, all the resources we have used can be found in any high school: free software (Guadalinex v9), App Inventor and Processing. The project was carried out under the principle of reducing all possible costs given the economic possibilities of the institution.

## 1 Introduction

What we are presenting here is the result of the work previous to the didactic units we are going to elaborate for our students. We think we should get them involved with the world of programming as soon as possible, and we can do it now, in 4th ESO.

Since 2000, IES Iulia Salaria has got a 120mm aperture, 1m focal refractor telescope made in China with a German equatorial mount with stepper motors in both axes. To control it, it has got a handbox that drives the motors in both directions, with the option of doing it at different speeds and with the sidereal movement that enables star tracking. This sidereal movement is not adjustable, and this can be a problem for the long term exposition photographs of the Moon, the Sun and the planets. Since that year, there has been an astronomy group with the students in the school with some continuity. The stepper motors are the mount original ones. They are in perfect condition. We have noticed that there is a gap between the motors and the gears so that when the direction of rotation is changed, the

motor must travel an angle before dragging the gears. This presents a significant error in the accuracy of the device that we have already quantified but not corrected.

Computer programming has always been within the specialized science margins. The use of low-level, high-performance languages such as Fortran 77 and C was the result of a calculation needs that, outside those circles, made no sense. Now, the popularity of mobile phones (in which audiovisual aspects, games, etc, come into play) and the need to control these devices make programming appealing to other sectors of the population. MIT created a very intuitive application, App Inventor<sup>1</sup>, which brings programming on Android closer to sectors of the population, such as children, very far from the initial programming circles. To make a program for mobile phones that allows to communicate with Arduino<sup>2</sup> via bluetooth is within any high school student's reach. We have made a model of the program that simulates the handbox of the mount but with more features.

Stellarium is an astronomical open source program that comes within the Guadalínex GNU/Linux distribution of the Junta de Andalucía for the Andalusian schools. Both professionals and amateurs agree on its excellence: one of the best and most complete astronomy softwares. Maybe, the most popular of all. Stellarium can control telescopes with computerized mounts in the Meade LX200 range. To do this, it has got some specific commands of this protocol already implemented. However, unlike other software of the same type as Carte du Ciel, the interface to control the telescope does not allow fine adjustments or to establish non-sidereal movements. For the needs of the amateur astronomer, it is insufficient. It is useful only for a "slew", i.e. to indicate a target star and, from its current position, move the motors looking for the direction of the star.

Arduino is managed through programs written in C. It is not that students learn C away from the needs posed by specific projects, but the other way around, i.e. they are started on the principles of that language through the projects. The amount of available libraries facilitates this task. However, we know that programming implemented in Arduino to control the two motors of the mount are beyond the capabilities of high school students, although parts of it are not. We used the shield for Arduino, which provides an intensity close to 1 A, the intensity required for the operation of both motors simultaneously. We used a Mega plate for the simple reason that it has the free ports required to connect the bluetooth. When the motors' shield is connected to Arduino Uno, port connections are covered by the shield.

We found the germ of our work on the Internet. It was developed by David Vidal under the name ArduGoto, a program for Arduino which connects to Stellarium and drives the motors of the mount. This program has serious shortcomings that make it practically useless. We have corrected the mistakes and we have also expanded the program including the possibility of being governed from a mobile phone via bluetooth and from a PC. It is troublesome to use Stellarium only to control it given the extremely austere and not very versatile interface that it has. The interfaces we have made for both Android and PC supplement all those deficiencies.

---

<sup>1</sup><http://ai2.appinventor.mit.edu> [Accessed Jun. 2016].

<sup>2</sup><https://www.arduino.cc> [Accessed Jun. 2016].

The bluetooth used is HC-05, which can be purchased for about 5 euros.

Finally, we point out that the students in 4th ESO work through projects in the subject of Technology. With the project we are presenting now, as noted above, it is intended to introduce students to programming in C as necessary for both Arduino and Processing<sup>3</sup>, and Android mobile programming carried out with App Inventor.

## 2 Development

Confronting our project requires some previous knowledge of astronomy and technology by our students. As it was mentioned before, it is developed in 4th ESO, but we start in 3rd ESO. In this course, the geographical coordinates are studied. It is the right time to move to the equatorial astronomical coordinates so that they know how to place a star in a planisphere. It may also be interesting to get them started in the apparent motion of the celestial orb without taking into account the motion of the planets and the Sun. Stellarium can be then especially useful by introducing different speeds over time and observing several areas of the sky.

Although observations are made in 3rd ESO, it is in 4th when the equatorial mount has to be presented, paying special attention to the fact that the tracking of a star can be done with a constant speed of a motor, that of Right Ascension, when dealing with this type of mounts. Tracking, therefore, is to set the proper speed of that motor. It is also important to notice the importance of starting monitoring station to be accurate. With a bad start telescope setup, it is impossible. During the observation, the types of telescopes and parts of a refractor (like the one the high school has) would also be explained.

Regarding the previous knowledge students must have on Arduino, Processing and App Inventor, they will get it from the area of Technology. After a simple explanation about connection to the PC and load of programs, the student must face a simple program like lighting a led with Arduino via the serial port as well as from the bluetooth already programmed in App Inventor. The experience of several years of our students with App Inventor shows that they acquire programming skills easily in that environment. We think it is because the program offers the immediacy of results by changing the different variables that come into play in the program.

## 3 Results

In this link (<http://iuliasalaria.org/mates/astroiulia/ardueq3/ardueq3.zip>) you can download a zip file containing:

- Program `ardueq3.ino`, which is the file you must upload to the Arduino plate
- Program `ardueq3movil.apk`, which is the application you must upload to your mobile and that will provide a handbox for the telescope.

---

<sup>3</sup><http://processing.org>. [Accessed Jun. 2016].

- Program `ardueq3pc.pde`, which is the text file for Processing if you want to make changes.
- `ardueq3pc` (the previous file compiled for Linux and Windows for both 32 and 64 bits).
- Use and installation manual.

The program `ardueq3.ino` has its germ, as noted earlier, in `ArduGoto`, but it has been corrected and extended. The sentences for the connection and interpretation of commands via the serial port and bluetooth are connected on it. For the latter, we have designed it so that a single character could be read through the port, what is much simpler than commands `LX200`, in which you have to collect them in an array where a character is stored in each cell. You will have to download the `AFmotor` library from the web. It is necessary for the operation.

To install the application `ardueq3movil.apk`, we copied it first on the mobile phone Download folder. It is optimized for 4,3-inch screens, so there will be no problems for larger screens. Complete information about it can be found in the manual. Similarly for `ardueq3pc`.

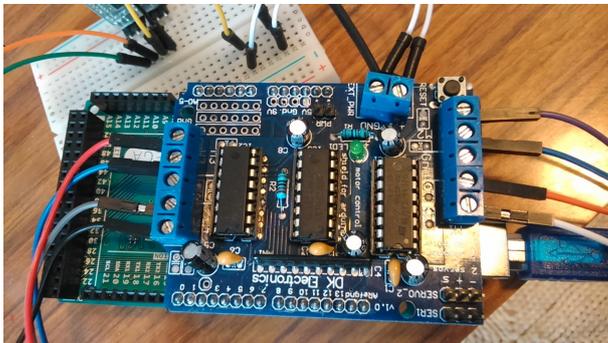
Once tested, the operation is more than satisfactory, although we have found some drawbacks.

The first one is that once the target position has been indicated via `Stellarium`, there is a difference between the position finally adopted by the telescope and the requested one. The difference is not only due to the periodic error, but to a gap in the gears that is evident especially when the direction of the rotation is changed. This was already mentioned before. Fortunately, with the help of the mobile's interface or the PC's, we can adjust the telescope to the desired position. The speed adjustment is important.

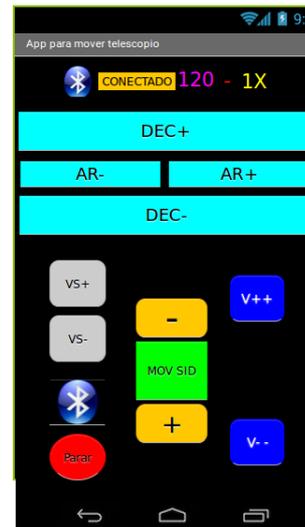
The second drawback is that some mobile phone batteries do not support long observation time. We also had to adjust the brightness of the mobile phone screen to a minimum to help the eye dark adaptation when going back to ocular mode.

The design of the interfaces has been quite simple. We had not raised the possibility of programming complicated interfaces, in the sense that they had visual effects, for example, when pressing the keys. We should not forget that the goal is to introduce students to programming.

Since the Arduino plate supports a wifi shield plate and the high school is equipped with a `Celestron NexImage` camera, we can view the ocular images and control the telescope from a remote computer. The images are collected and sent through the `VLC` program, which allows streaming and is also free software. It is the simulation of a large observatory operation made for and by high school students. The cost of computerization has been minimal: it is less than 30 euros.



(a) arduino and shield



(b) ardueq3movil



(c) stellarium and ardueq3pc