

# Identification of wavelengths from the visible spectrum by means of Arduino for the generation of a knowledge base managed by PROLOG

*Pinedo Arriaga O. Ts'zul*<sup>1,\*</sup>, *Pinedo Arraiga Carlos D.*<sup>2,\*\*</sup>, *Herrera Alvarado Eduardo*<sup>3,\*\*\*</sup>, and *Tinoco Varela David*<sup>4,\*\*\*\*</sup>

<sup>1,2,3</sup> Universidad Nacional Autónoma de México, FESC, IME, México.

<sup>4</sup> Universidad Nacional Autónoma de México, FESC, Departamento de Ingeniería, México.

**Abstract.** PROLOG is a programming language widely used in the generation of expert and intelligent systems, generally limited to data that is entered directly by a user in the form of software, having little or no interaction with data that is captured directly from a physical environment. This paper presents an implementation of an interface that detects the wavelengths of the visible spectrum, that is, identifies colors, colors that are stored in a knowledge base and then managed by PROLOG. This interface consists of two parts, software and hardware. The hardware is designed by means of the Arduino UNO development board, where a TCS3200 sensor is used. For the development of the software, two tools have been used, on the one hand, the standard programming of the Arduino IDE terminal has been used to manage the inputs and outputs of the Arduino board, and on the other hand, a data management system has been generated, in which PROLOG manages all the data obtained from hardware. This scheme seeks to generate color classifications in a dynamic and intelligent way in the future.

The proposed system has the advantage that it is highly economical, easy to perform, uses the logical paradigm of programming, and opens the way to the design of intelligent systems managed by PROLOG from a monitoring of physical variables.

**Keywords:** PROLOG, Arduino, Wavelength sensor, Interfaces.

## 1 Introduction.

In a world where there is a growing hodgepodge between digital and physical environments, the acquisition of data from a real environment, and processed by a virtual environment, has become a topic of great interest, mainly due to the need to generate artificial ambiances that are responsive to the needs of a user.

There are many methodologies and tools that allow generating interfaces that achieve the interaction between the two mentioned worlds, however, many of them are expensive, and some others require that a user elaborates from scratch, intelligent but complex algorithms.

One of the tools mentioned is the Arduino development board, this board has been used in numerous applications as a data acquisition system [1] [2], however, for the purposes of the project presented in this article, only some applications related to color or *wavelength sensors* will be described.

The wavelength sensors can be useful in applications related to the textile industry, the pigmentation industry, and even the agricultural industry, we can see in [3] an example of this last sector, where the authors state that in

the automatic inspection of fruit and vegetables, the ability to identify specific wavelengths is important because such a sensor provides information about individual components, or fruit damage that can be perceived only at certain wavelengths.

Arduino has played an important role in the application of this type of sensors in different branches of knowledge, for example, in 2015 *Morais* and other authors [4] described a photometer for the detection of plumb ions, this device was based on the TCS230 sensor, and an Arduino board as a data acquisition system. Another application of this type is described in [5], where a device which measures the color of the urine of people with diabetes and healthy people was designed, in which, based on the identification of the color could be detected if a person is healthy or sick, for this system was used in addition to the Arduino board, a color sensor type RGB TCS3200.

One of the most important aspects to consider when making this type of devices, is the economic aspect, since according to *Agudo* and other authors [6], "color measurements have traditionally been related to expensive and difficult to handle equipment", with the intention of minimize this fact, they developed a system which captures, process, and manages the color of any

\* [tszulpinedo@gmail.com](mailto:tszulpinedo@gmail.com), \*\*[carlos.devir.pinedo.arriaga@gmail.com](mailto:carlos.devir.pinedo.arriaga@gmail.com),  
\*\*\*[eduardo.herrera.alvarado@gmail.com](mailto:eduardo.herrera.alvarado@gmail.com), \*\*\*\* [dativa19@hotmail.com](mailto:dativa19@hotmail.com)

non-luminescent object using a low-cost hardware, based on Arduino and a color sensor TCS3414CS, they implemented it with the help of *Matlab*. In the same line of thought, are *Anzalone* and other authors [7], who mention that "the high cost of what have been sensors and sophisticated tools related to research, has limited their adoption to a relatively small group of well-funded researchers", with this thought in mind, they designed an open source colorimeter, putting the Arduino board as a main processing system, and using a TSL 230R sensor, according to their own words, the development of this type of systems "will allow unprecedented access to sophisticated instrumentation based on low-cost sensors to those who need it the most, laboratories of the underdeveloped and developing world".

As shown, Arduino has actively participated in the development of data acquisition systems, in this case, acting as an identification system of colors or wavelengths, however, the intention of this project is to implement a system that not only recognizes the colors, but also, it can manage them by means of logical programming and in a future, it can obtain logical conclusions from the data obtained, processed, and stored. For this purpose, it is proposed to use PROLOG.

PROLOG is a programming language that has been developed, and used mainly for the generation of artificial intelligence, which gives it an inherent power when it comes to developing intelligent systems or expert systems. This language allows a developer to focus more on the solution of the problem than on the implementation of the algorithm, achieving that an intelligent environment is not initiated from scratch, however, in most cases it has been used to acquire knowledge entered just by software, limiting its possibilities since it does not generate an interaction with the physical environment that surrounds it.

In this section we have reviewed how wavelength sensors have been used in different applications, and how Arduino has been in charge of managing the information obtained from these sensors, however, in this project we will not focus on the detection of colors as a theme principal, but we will focus on the fact of being able to generate a knowledge base that can be managed through PROLOG, a knowledge base that is obtained directly from the measurement of physical quantities existing in a real environment through Arduino and different kind of sensors, in this particular case, information obtained through the color sensor. This first step is aimed at generating an expert system that can intelligently manage and process changing and dynamic data in an intelligent manner.

## 2 Preliminaries

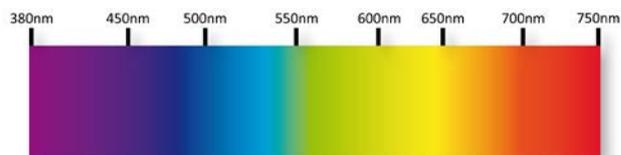
In this section some basic concepts for the generation of the presented project are presented.

### 2.1 Visible light and wavelength

*Electromagnetic waves*, which is the term that describes the way in which electromagnetic radiation travels

through space, cover a wide range of wavelengths, to all this range of values, they are called the *electromagnetic spectrum*.

Within the electromagnetic spectrum are the wavelengths among which belong the gamma rays, and the wavelengths that correspond to the radio waves, including acoustic and optical wavelengths. Within this wide range of wavelengths from the electromagnetic spectrum, we have the *visible spectrum*, which is the region of the spectrum that the human eye is capable of perceiving. An electromagnetic radiation in this range of wavelengths is called visible light or simply light, which is divided between the colors that range from violet to red. Although formally, a range of values in wavelength is assigned to each color, in reality it is difficult to define specifically the range corresponding to each color, since as it approaches the end of the range of one color, the following color is already starting, even so, a human eye is able to perceive wavelengths from 400 to 700 nm. In Figure 1, it is possible to see the spectrum of the colors, as well as their approximate limits of wavelength.



**Fig. 1.** Visible spectrum, in the upper part it is possible to see the estimated limits of the wavelengths corresponding to the different colors, given in nanometers.

### 2.2 Arduino

According to the official page of the *Arduino project* (<https://www.arduino.cc/>), Arduino is an open source electronic platform based on easy-to-use hardware and software. One of the main advantages of this development board, is that Arduino can read inputs through sensors and keypads, process these inputs, and generates controls of different types of actuators in their outputs, all of this in a very simple way.

This electronic board, has become the central process of thousands of projects, ranging from very simple projects to highly complex scientific projects, some of these can be seen in [8], [9] and [10].

Arduino was developed in the *Ivrea Interaction Design Institute* as a tool that could be easily used by anyone without experience in electronics and programming for the generation of technological prototypes. The most used board is called *Arduino UNO*, and we can see it in figure 2.



**Fig. 2.** Development board Arduino UNO.

### 2.3. PROLOG

PROLOG is a language based on the logical programming paradigm, this language is used mainly for artificial intelligence applications. The development of PROLOG began in 1970 with *Alain Coulmeauer* and *Philippe Roussel*. The name corresponds to "PROgramming in Logic".

PROLOG uses a language based on statements of facts, questions, and rules. It seeks to relate the different facts existing in its knowledge base through linked logical sequences, in order to reach a logical conclusion starting from certain predicates, It generates a chain of true predicates, and if all of them are true, the result will be true; if a single predicate within the chain is false, the final result will be false.

An interesting aspect of this language is that it bases its programming on recursive schemes and functions.

Some characteristics of this language are:

- It is based on logical predicates.
- It focuses on solving the problem, rather than on how to reach that solution.
- Only its execution continues, if the objectives are met.
- The user focuses more on knowledge than on algorithms.
- It begins from known to the unknown.

Although the language is applied mainly in intelligent systems and expert systems [11, 12, 13], it can have more varied applications, such as the generation of computer security protocols [14].

## 3 Project development

This section gives a description of how the project was carried out, the development in hardware and software, and how the data obtained was communicated.

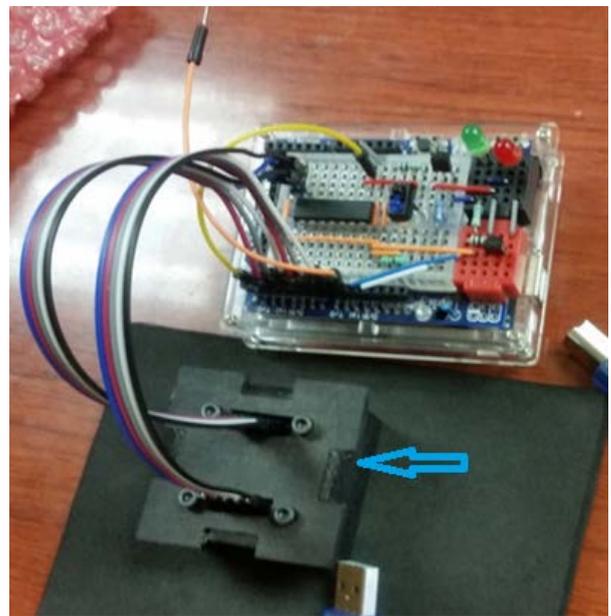
### 3.1. Hardware

As already mentioned, the TCS3200 sensor has been used, which is a light-frequency converter, and with the

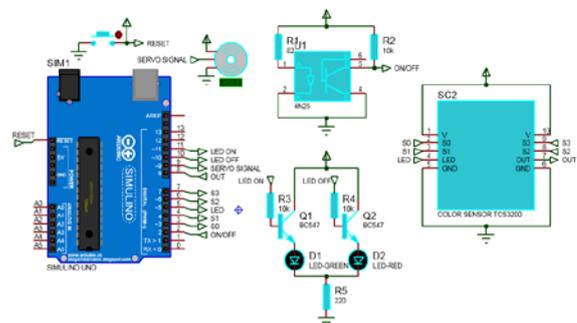
reading of the sensor's output frequency, Arduino can easily calculate the color that is being measured. When first tests were performed, the circuit had problems to correctly identify the test colors, so a completely black cover was designed to obtain better results at the time of testing. This cover allows to isolate the entry of ambient light since it affects the readings made by it.

The sensor is equipped with a pair of LEDs that illuminate the object to analyze, each color has a specific wavelength, so what the sensor perceives is a different frequency for each color, this signal is sent to the Arduino board. Within the Arduino, the signal is mapped to determine the color that is being measured. The implemented circuit can be seen in figure 3.

The electronic diagram of the implementation can be seen in Figure 4.



**Fig. 3.** Implementation of the sensor and the Arduino board, where the blue arrow points to the cover that protects the TCS3200 sensor.



**Fig. 4.** Electronic diagram of the presented implementation.

Once the implementation is ready, a base of knowledge of colors is produced, the knowledge base is generated through the color sensor, and color samples that are entered. When a user wants to enter a new value to the knowledge base, a color card is placed under the sensor, and the hardware is indicated that it is time to

take the measurement, for this example, we use the cards shown in figure 5.



Fig. 5. Color cards used to fill the knowledge base.

### 3.2. Software

For the software part we have used three tools: Arduino IDE, which is responsible for managing the inputs and outputs of the TCS3200 sensor, as well as performs the calculations that define the tested color; PROLOG, which is responsible for managing the knowledge base related to colors, indicates whether the information of a specific color exists; and finally, the tool called *Hyperterminal*, which allows us to read the information from the serial channel, and therefore from Arduino, it serves as a communication bridge between Arduino and PROLOG.

Arduino IDE, performs the calculation and determines the color that is being tested, this information is sent to the serial monitor, from this point, the Hyperterminal obtains the acquired information and displays it on the screen in real time, which generates a file in which will be sent the information received. Through this tool, we can stop and resume the process (Call and hang), that is, we can define the moment in which we want to continue with the data acquisition.

When a user wants to enter a new color, we have the following behavior:

1. We begin with the reading of the data when activating the circuit, we place the blue card to carry out the test, so that, in the Hyperterminal monitor we can visualize the color that is being charged (Figure 6).
2. PROLOG can identify if there is a new data or a lack of data, these data are provided by the Hyperterminal. The manager in PROLOG starts by means of the word "start." (Figure 7), which enables the system to access the database.
3. Activation of the knowledge manager shows on the screen a message with the color request that you want to know if it exists in the database (figure 7).
4. As the color has already been saved, it is shown in the database and corroborated with the message that is printed on the screen, as can be seen in figure 8.
5. A second query is made, where an unsaved color is searched, as there is no knowledge of

that color, the message is printed where it denies the existence of it and asks to be entered by Arduino through the TCS 3200 sensor, and thus save it in the database, this can be seen in figure 9.

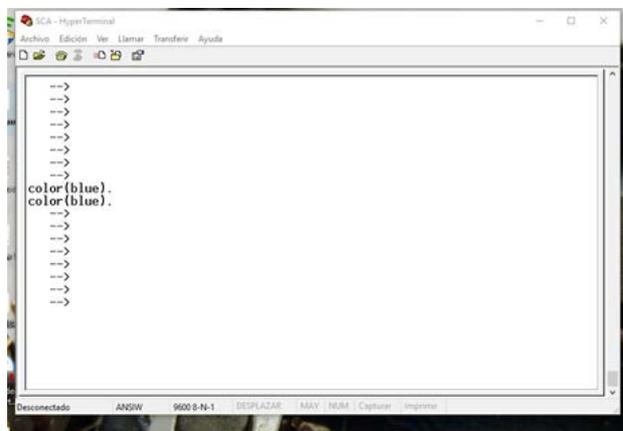


Fig. 6. The Hyperterminal shows on screen the color that is being stored in the PROLOG knowledge base.

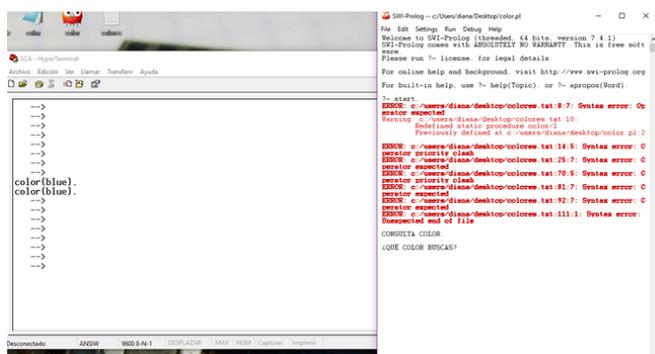


Fig. 7. Activation of the knowledge base manager, and data search.

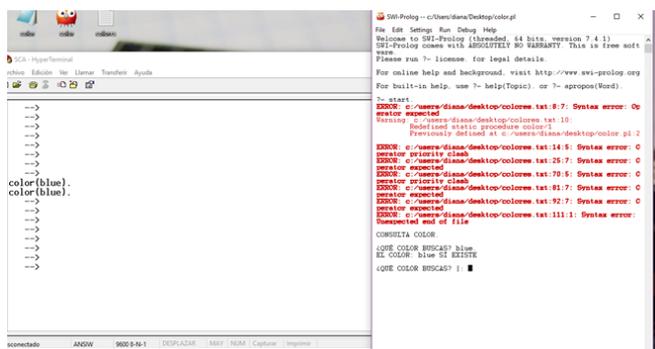
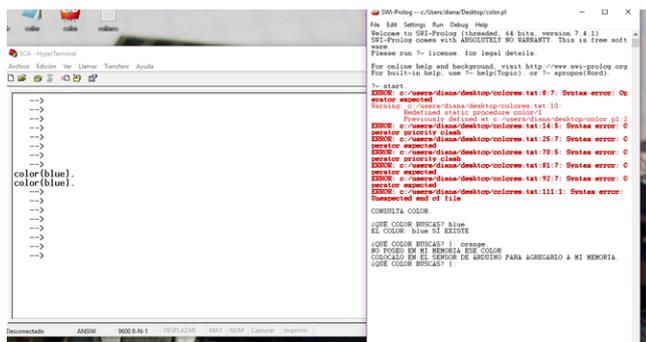


Fig. 8. Corroboration that the searched color has been saved in the knowledge base.



**Fig. 9.** A color that does not exist in the knowledge base is sought, so the terminal of PROLOG requests that such color could be placed on the sensor in order to register its existence.

### 3.3. Advantages

The advantages that the proposal presents are:

1. As mentioned by other authors, it is very important to generate inexpensive interfaces that can be accessed by anyone, the implementation shown is extremely cheap, and so it complies with this idea.
2. Carry out an implementation of this type, does not require hard-to-get tools, in fact it is very easy to access them.
3. The sensor used is quite easy to calibrate and connect to Arduino, in fact, there are many programs on the internet that can be used for free for this purpose.
4. PROLOG is focused on the generation of knowledge bases, and the logical processing of stored data, so in the future it is possible to make an intelligent system based on data obtained from the real world, using only the inherent characteristics of the language, without the need for implementation of highly complex algorithms.
5. Based on this scheme, in the future, proposals for smart buildings managed through PROLOG could be made.

### 4 Conclusions

ROLOG is a programming language that has mostly been used to manage data entered by software, in this scheme, PROLOG is working with data entered directly by hardware. The knowledge base managed by PROLOG is being filled with data that comes from the real world, generating an interface for data acquisition.

In the proposed scheme, a knowledge base focused on colorimetry has been developed, however, this same scheme can be implemented to acquire different types of data such as solar radiation levels or humidity levels, only by changing the acquisition sensors and modifying the software a bit inside the Arduino IDE.

PROLOG has adequately managed the data that was entered, and communication between the three tools used has been easy and coordinated.

As can be seen, the system that was generated is extremely cheap, since the Arduino board is a low cost and easy to acquire board; PROLOG can be programmed in various free access tools; and the Hyperterminal has a reduced economic cost.

PROLOG is used, due to the facility it has to develop intelligent systems, and we want to implement an expert system which can use dynamic data.

### 5 Future work

By entering data into a knowledge base through Arduino, the possibility of generating intelligent management of data acquired from any environment is opened. In this article a way of interaction of the PROLOG language with a physical environment was implemented, so, as future work, the development of an expert system written in PROLOG that can interact with the environment is proposed, making it not a system that is dependent only of the data entered by the user, but can be generating its own base of knowledge from the environment where it is, and thus generate a dynamic expert system.

**Acknowledgment:** This paper was in part supported by the PAPIIT IN 113316 project, and the PIAPI 1634 project, of the UNAM.

### References

1. Faris, D. M., & Mahmood, D. M. F. M. B. Data acquisition of greenhouse using Arduino. *Journal of University of Babylon*, 22(7), 1908-1916.(2014).
2. Shajahan, A. H., & Anand, A. Data acquisition and control using Arduino-Android platform: Smart plug. In *Energy Efficient Technologies for Sustainability (ICEETS), 2013 International Conference on* (pp. 241-244). IEEE. (2013, April).
3. Cubero, S., Aleixos, N., Moltó, E., Gómez-Sanchis, J., & Blasco, J. Advances in machine vision applications for automatic inspection and quality evaluation of fruits and vegetables. *Food and Bioprocess Technology*, 4(4), 487-504. (2011).
4. de Moraes, C. D. L., Carvalho, J. C., Sant'Anna, C., Eugênio, M., Gasparotto, L. H., & Lima, K. M. A low-cost microcontrolled photometer with one color recognition sensor for selective detection of Pb 2+ using gold nanoparticles. *Analytical Methods*, 7(18), 7917-7922. (2015).
5. Wardana, H. K., Indahwati, E., & Fitriyah, L. A. Measurement of Non-Invasive Blood Glucose Level Based Sensor Color TCS3200 and Arduino. In *IOP Conference Series: Materials Science and Engineering* (Vol. 336, No. 1, p. 012019). IOP Publishing. (2018, April).

6. Agudo, J. E., Pardo, P. J., Sánchez, H., Pérez, Á. L., & Suero, M. I. A low-cost real color picker based on arduino. *Sensors*, *14*(7), 11943-11956.
7. Anzalone, G. C., Glover, A. G., & Pearce, J. M. (2013). Open-source colorimeter. *Sensors*, *13*(4), 5338-5346. (2014).
8. Piyare, R., & Tazil, M. Bluetooth based home automation system using cell phone. In *Consumer Electronics (ISCE), 2011 IEEE 15th International Symposium on* (pp. 192-195). IEEE. (2011, June).
9. Baraka, K., Ghobril, M., Malek, S., Kanj, R., & Kayssi, A. Low cost arduino/android-based energy-efficient home automation system with smart task scheduling. In *Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on* (pp. 296-301). IEEE. (2013, June).
10. Akbar, M. A. Simulation of fuzzy logic control for DC servo motor using Arduino based on MATLAB/Simulink. In *Intelligent Autonomous Agents, Networks and Systems (INAGENTSYS), 2014 IEEE International Conference on* (pp. 42-46). IEEE. (2014, August).
11. Protopapas, C. A., Psaltiras, K. P., & Machias, A. V. An expert system for substation fault diagnosis and alarm processing. *IEEE Transactions on Power Delivery*, *6*(2), 648-655. (1991).
12. Fuglsang-Frederiksen, A., Rønager, J., & Vingtoft, S. PC-KANDID: an expert system for electromyography. *Artificial Intelligence in Medicine*, *1*(3), 117-124. (1989).
13. Goethe, J. W., & Bronzino, J. D. An expert system for monitoring psychiatric treatment. *IEEE Engineering in Medicine and Biology Magazine*, *14*(6), 776-780. (1995).
14. Blanchet, B. An Efficient Cryptographic Protocol Verifier Based on Prolog Rules. In *csfw* (Vol. 1, pp. 82-96). (2001, June).