

HOW CONSTRUCT A WLAN MULTI-DATA ACQUISITION SYSTEM BASED ON THE INTEGRATION OF ARDUINO AND NI-LABVIEW PLATFORMS FOR EDUCATIONAL APPLICATIONS



Revista
Desafios

Artigo Original
Original Article
Artículo Original

Como construir um sistema de aquisição de multi-dados WLAN baseado na integração do Arduino e plataforma NI-LabVIEW para aplicações educacionais

Cómo construir un sistema de adquisición de multidatos WLAN basado en la integración de Arduino y plataforma NI-LabVIEW para aplicaciones educativas

Nilo Maurício Sotomayor Choque^{*1}, Liliana Yolanda Ancalla Dávila¹, Welliton Leite da Silva¹, Alexsandro Silvestre da Rocha¹

¹ Curso de Licenciatura em Física, Universidade Federal do Tocantins, Araguaína, Tocantins, Brasil.

*Correspondência: Universidade Federal do Tocantins, Campus Universitário de Araguaína, Curso de Licenciatura em Física. Av. Paraguai esq. com R. das Uxiramas s/n – Cimba CEP: 77824838 - Araguaína, TO – Brasil. e-mail: nmsch@uft.edu.br

Artigo recebido em 08/06/2017 aprovado em 21/11/2017 publicado em 27/12/2017.

RESUMO

A construção de uma Rede Local Sem Fio (RLSF) com um Sistema de Aquisição de multi-Dados (SAD) para aplicações educacionais é relatado, onde um banco de dados para a intensidade da luz, temperatura e potencial elétrico de um painel fotovoltaico foi gerado. Os sinais analógicos de três diferentes sistemas de sensores/transdutores são recolhidos e enviados para um placa Arduino Uno Revisão 3. Os sinais analógicos são convertidos em dados digitais através de um código guardado no microprocessador Arduino e, em seguida, transmitido via Internet através da utilização da tecnologia sem fios suportada por um servidor Arduino Uno WiFi acoplada à placa Arduino Uno. Dados da Internet são adequadamente recuperados, armazenados em uma base de dados, e apresentado continuamente pelo software NI-Labview desenvolvido. Esta ferramenta fornece o professor controle remoto e monitoramento de experimentos físicos desenvolvidos pelos alunos, e seria muito relevante em Educação a Distância (EaD) onde as aulas experimentais podem ser desenvolvidas pelo tutor e aquisição de dados remotamente monitorado em tempo real pelo docente.

Palavras-Chave: Arduino; Rede Local Sem Fio, Aplicações Educacionais.

ABSTRACT

The construction of a Wireless Local Area Network (WLAN) multi-Data Acquisition System (DAS) for educational applications is reported, where a database for light intensity, temperature and electric potential for a photovoltaic panel was generated. Analog signals from three different sensors/transducer systems are collected and sent to an Arduino Uno Revision 3 board. The analog signals are converted to digital data through a code saved on the Arduino microprocessor and then broadcasted to the internet by the use of the WLAN technology supported by an Arduino WiFi Shield server coupled to the Arduino Uno board. Data from the internet are properly retrieved, stored as a database, and continuously displayed by the development of a NI-Labview software application. This tool provides the teacher remote control and monitoring of physical experiments developed by students, and it would be very relevant in Distance Education (DE) where experimental classes can be developed by the tutor and the acquisition of data remotely monitored in real time by the teacher.

Key words: Arduino; Wireless Local Area Network; Educational Applications

RESUMEN

Es relatada la construcción de un sistema de adquisición de datos múltiples (DAS) con base en redes de área local inalámbrica (WLAN) para aplicaciones educativas, en el cual que se generó una base de datos para la intensidad luminosa, la temperatura y el potencial eléctrico de un panel fotovoltaico. Las señales analógicas procedentes de tres diferentes sistemas de sensores/transductores se recogen y se envían a una placa Arduino Uno Revisión 3. Las señales analógicas son convertidas a datos digitales a través de un código guardado en el microprocesador de la placa Arduino y luego se difunden pela Internet mediante el uso de la tecnología WLAN soportada por un servidor de internet en un shield WiFi compatible, acoplado con la placa Arduino Uno. Los datos de Internet se recuperan correctamente, se almacenan como una base de datos y se muestran continuamente a través de una aplicación de software desarrollada en NI-Labview. Esta herramienta provee al profesor de control remoto y monitoreo de los experimentos físicos desarrollados por los estudiantes, y sería muy relevante en la Educación a Distancia (DE) donde las clases experimentales pueden ser desarrolladas por el tutor y la adquisición de datos remotamente monitoreados en tiempo real por el profesor.

Descriptor: Arduino; Red inalámbrica local; Aplicaciones educativas.

INTRODUCTION

Data acquisition is a general instrumentation procedure of sampling analog signals, resulting from arrays of sensors/transducers, to test real world physical phenomena and convert them into digital numeric values that can be displayed and stored by a computer. Today, data acquisition systems are essential and reliable tools in any scientific research laboratories allowing high speed multi data sampling in numerous applications, however, proprietary systems are often expensive and the details of the hardware-software design are not always available. In common applications data acquisition must often be realized in remote locations distant from laboratory facilities making it difficult to monitor the collected data in real time, in this situation, the use of Wireless Local Area Networks (WLAN), technologies to provide connection to the internet for data transference must be desirable. The Arduino is a simple automation tool that associated with WLAN provide the teacher control and real-time

monitoring (by computer) the physical experiments developed by the students. The use of Wireless in remote monitoring is employed in animal (HANDCOCK *et al*, 2009), sensor network on an active volcano (WERNER-ALLEN *et al*, 2006), landslide detection (RAMESH, 2009), monitoring system for fluoride affected areas (KUMAR *et al*, 2014), Habitat Monitoring (MAINWARING *et al*, 2002), for intelligent building energy management (GRINDVOLL *et al*, 2012) and combining with Arduino (POZZEBON, 2015). The capability of sending digital data from a DAS to a website in real time would the information worldwide available for recovering and storing after proper filtering. Up to date, analog to digital signal converters (ADC) and WLAN technologies that allows an electronic device to participate in computer networking using electromagnetic radio bands are available under the open-source electronics platform based on Arduino hardware and software (WHAT IS ARDUINO?, 2015). The low cost of Arduino provides your application in LED stimulator system for vision research

(TEIKARI *et al*, 2012), experiments in psychological and neurophysiological settings (D'AUSILIO, 2012), in cross-disciplinary teaching of biology and computer science (GRASEL *et al*, 2010) controller for the smart assistive mobility hardware (LENGVENIS *et al*, 2012), to measure response latencies (SCHUBERT *et al*, 2013) and Physics/Educational demonstrations (SOUZA *et al*, 2011; LÓPEZ-RODRÍGUEZ and CUESTA 2015; KUBÍNOVÁ and ŠLÉGR 2015; PEREIRA *et al*, 2016; VARANIS *et al*, 2016; AMORIM *et al*, 2015). In this paper, the construction of a wireless local area network (WLAN) multi-data acquisition system based on Arduino Uno Revision 3, Arduino WiFi shield board and specific NI-Labview (GRAPHICAL DEVELOPMENT ENVIRONMENT LABVIEW SYSTEMS, 2015) software applications is reported. This would also be very important in DE (BRASIL, 2006), where experimental physics classes could be carried out only with the presence of a tutor and remotely monitored by teacher. We use an experimental example, where the database for light intensity, temperature and electric potential for a photovoltaic panel was generated. This WLAN multi-data acquisition system based on the integration of Arduino and NI-Labview platforms can be applied to many physical experiments.

MATERIALS AND METHODS

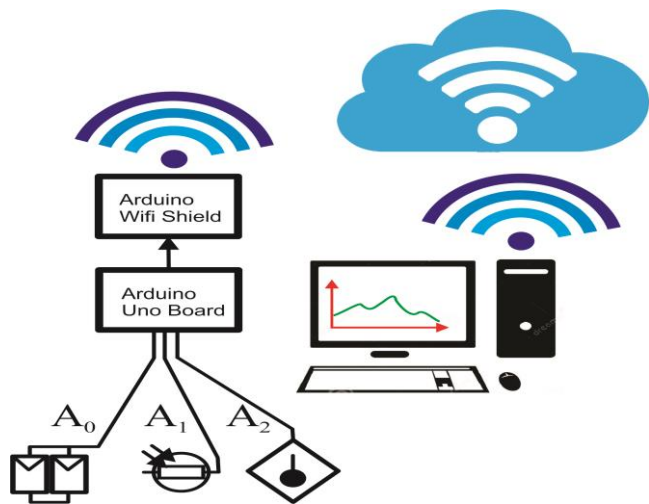
The basic characteristics of the system are depicted in Figure 1, analog data from three serial channels can be used as inputs to an Arduino Uno

board that converts the signals to digital values, the Arduino Uno board broadcast the digital data to the internet through the Arduino WiFi shield that works as an internet server. Client browsers may navigate to the WiFi shield address and with the use of specific NI-Labview tools may recover, store and display the collected data. Data collection distance is an important tool to extrapolate the school environment, as provided with Arduino WiFi shield board and access to Internet connection, the teacher can propose scientific experiments as homework and monitor the data remotely.

RESULTS AND DISCUSSION

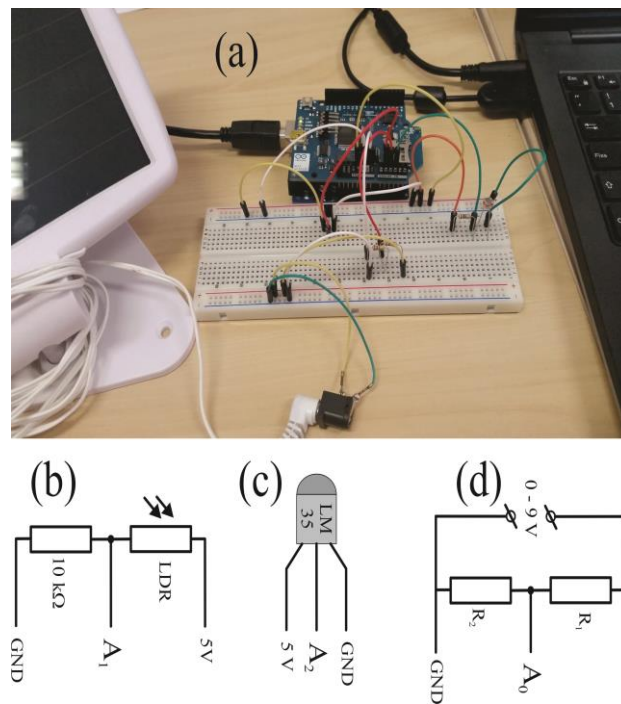
An Arduino Uno Revision 3 and an Arduino WiFi Shield (THE ARDUINO WIFI SHIELD, 2015) boards are coupled through their electrical contacts, this system is connected to a computer through a Universal Serial Bus. By the other side, analog electrical signals from a light-dependent resistor (to test solar irradiance), temperature from an LM35 precision integrated-circuit device, and electric potential from a photovoltaic system (all three of them as function of time), are connected to the A0, A1, A2 inputs of the coupled system (Figure 1).

Figure 1: Scheme of the WLAN data acquisition system, analog electric signals from a solar panel, a photoresistor and a LM35 temperature sensors are sequentially collected and digitalized by the Arduino Uno board. The digital data is broadcasted to the internet by the Arduino WiFi Shield, then, a NI-Labview application establishes a communication with the server and recovers all the available information. After a proper filtering process the desired information is stored and continuously displayed.



The Figure 2(a) depicts the physical structure of the data acquisition system, the array of sensors/ transducers are organized in a protoboard and the analog signals are sent to the Arduino Uno board inputs. Part (b) shows a scheme of the input circuit that collects the sunlight intensity through a $10k\Omega$ light dependent resistor (LDR). Part (c) depicts the input circuit for temperature acquisition, and part (d) shows a scheme of the input circuit that collects the direct voltage (DC) of a photovoltaic panel. A computer program is developed to convert the analog signals in numerical data and broadcast to the internet through the WiFi shield, the code is transferred to the microcontroller via USB and then the bus is disconnected, then a external 9 V DC power supply is connected to the Arduino system to operate.

Figure 2: (a) Physical structure of the data acquisition system. (b) Scheme of the input circuit that collects the sunlight intensity through a light dependent resistor. (c) Input circuit for temperature acquisition. (d) Input circuit that collects the direct voltage (DC) of a photovoltaic panel.

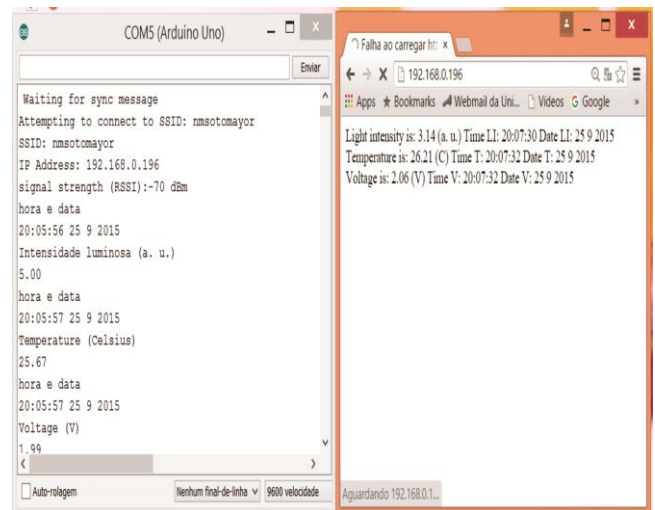


The ATmega 328 microcontroller of the Arduino Uno board is a 10 bit analogic digital converter ADC so it has the capability to detect 1.024 (2¹⁰) discrete analog levels of electric potential for each analog voltage source. A code in the Processing/Wiring Integrated Development Environment of the Arduino project was written to sequentially read the analog inputs and convert them to digital values, simultaneously, by the use of the specific Arduino Time library the exact date and time of the capture of each analog physical magnitude being registered are also sequentially recorded. These numerical values are synchronized as a function of the UNIX/POSIX time (THE UNIX TIME, 2015) that is a system for describing instants in time, defined as the number of seconds that have elapsed since

00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970. For the acquisition of the external DC voltage of the photovoltaic panel that reaches up to 9 V, the internal reference of the Arduino Uno board is used (1.1 V), for the sunlight intensity and temperature the default AVcc reference of the board (5 V) is used. After a serial lecture of the analog signals, the digital values and their corresponding date and time are sent to be printed at the serial monitor and also broadcasted to the internet. The WiFi shield starts communication with an open network using WLAN technology that allows electronic devices to participate in computer networking using 2.4 GHz Ultra high frequency (UHF) and 5.0 GHz Super high frequency (SHF) radio broadcast bands. First, the Service Set Identifier SSID is broadcasted for the shield to connect, and WiFi Protected Access II (WPA2) Personal encryption protocol is used. By using the serial monitor of the Arduino software the actual UNIX/POSIX time is introduced to synchronize the WiFi shield server. When the system is connected to the internet the IP address and the collected digital data is displayed at the serial monitor, then, the USB cord can be disconnected from the Arduino Uno – Arduino WiFi shield system which must be reloaded with the aid of the external DC power that can be connected to a photovoltaic system in places where AC electric power is not available. Figure 3 shows the serial monitor of the Arduino board (on the left), and a web browser (on the right), connected to a Uniform Resource Locator (URL) corresponding to the WiFi shield server, both windows display the basic digital data being

broadcasted for light intensity, temperature and DC Voltage all as a function of the time and date.

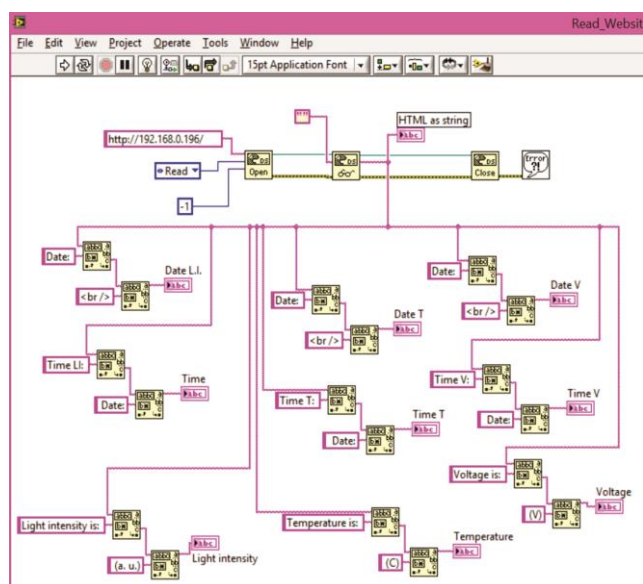
Figure 3: Picture on the left: serial monitor of the Arduino board showing the converted numerical data together with the time and date of collection. Picture on the right: web browser window showing the response of the WiFi shield server upon request, all the information broadcasted is recovered in HTML format.



For this work, the collected data is reloaded at a rate of 1.0 Hz, however, this refreshing time can be set according to particular purposes. At the internet, each digitalized data is public available temporarily and in our case will be used to make a database after proper and continuous recovering. The WiFi Shield does not store the collected data instead, broadcast it at variable time rates, the process of recovering, storing and displaying the digital data collected by the sensors/transducers is concretized through the employment of developed NILabview application. In order to capture the desired numerical data it is necessary to recover all the information in the Hyper Text Markup Language (HTML) response obtained upon request by the web browser, to do this, the DataSocket tool of NI-Labview was used. The DataSocket is an internet programming technology based on the

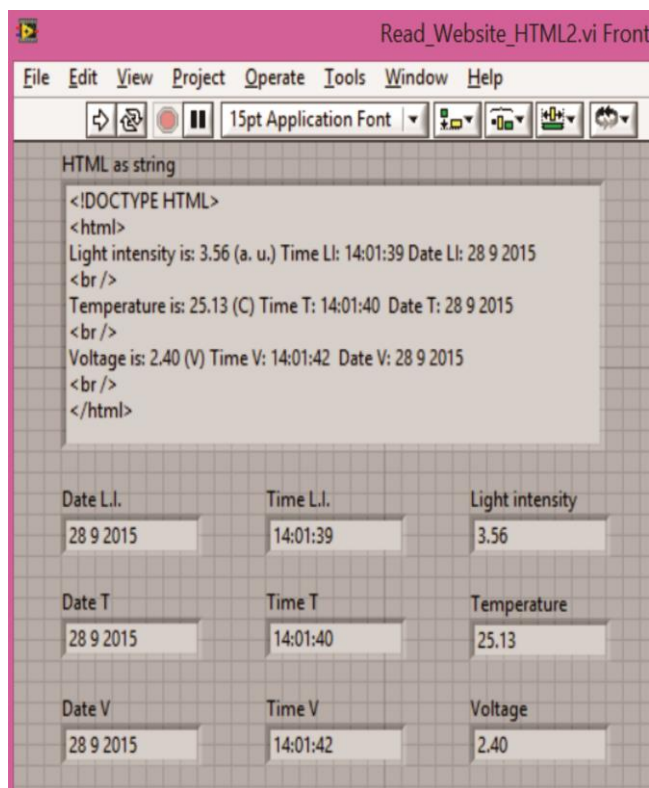
internet protocol suite TCP/IP that simplifies data exchange between computers and applications sharing and publishing live data in measurement and automation applications between different applications and between machines across the Internet. In the next, the process of building the software application is briefly described, first a DataSocket virtual instrument is opened, to start the communication with the WiFi Shield the Uniform Resource Locator (URL) must be introduced at the opening of the DataSocket. After connection, all the HTML content broadcasted by the server is recovered by the DataSocket Read virtual instrument as specified. Figure 4 shows, partially, the block diagram of the built application in the G programming language depicting the part of the Data Socket session, the HTML text recovering, and the process of data filtering (for a single sequence capture).

Figure 4: Partial view of the software block diagram in the G programming language, only the process of capture of a single sequence of HTML data is presented. At the beginning the Data Socket session is opened, all the HTML data is recovered, and then the session is closed. After the data capture, a filtering process starts to select the numerical values from the set of strings recovered in HTML format.

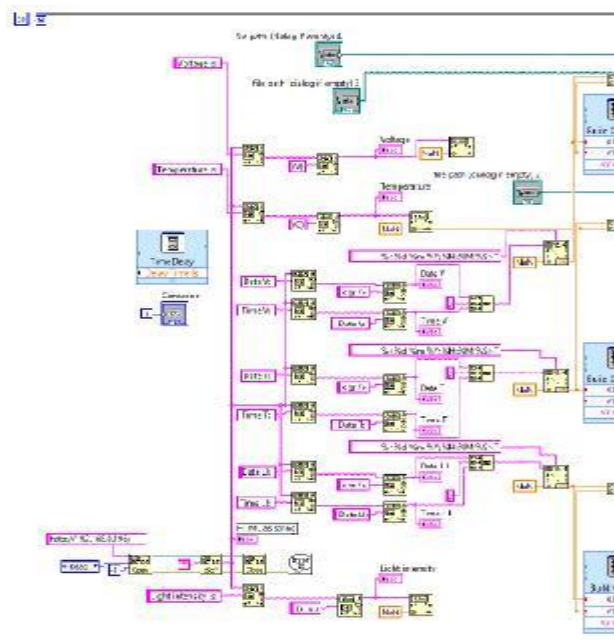


To select the desired numerical data from the HTML content it is necessary to indicate the position of these data within the chains of string characters, the identification and selection is realized by means of a specific tool of the NI-Labview called Match Pattern, the process consists basically in detecting particular chains of string before and after the desired numerical data and then to pick up and display the digits corresponding to the converted electric analog signals. Figure 5 shows a partial view of the front panel corresponding to the G-code shown in Figure 4. In the upper part, a window with the all HTML text recovered from the server is shown, in the lower part, individual windows show the numerical values selected with the use of the Match Pattern tool. As the data from the Arduino Uno board is reloaded at a specific time rate the process of filtering must be repeated in order to capture the sequential information, in this way, the G-code was implemented to repeat the capture process at a specific time rate compatible with the time rate of the Arduino Uno board- Arduino WiFi Shield system, these processes were implemented by means of the for loop tool.

Figure 5: Partial view of the front panel corresponding to the G-code shown in Figure 4. A window with the HTML text recovered from the server is shown in the upper part, in the lower part, individual windows show the numerical values selected with the use of the Match Pattern tool.



Once the desired numerical data is selected through the NI-Labview application it must be recorded, stored and displayed continuously in order to build a database of the physical magnitudes being measured. The numerical values for light intensity, temperature, and solar panel voltage are sequentially and continuously stored in different text files with the use of the Write To Spreadsheet File tool, at the same time, the numerical values are also continuously displayed at the computer monitor by means of the Build XY Graph tool. In the case of the plotting of the physical magnitudes as a function of time and data it is necessary to specify the format for the lecture of the POSIX/UNIX universal time in order convert the time stamp (sequence of characters or encoded information

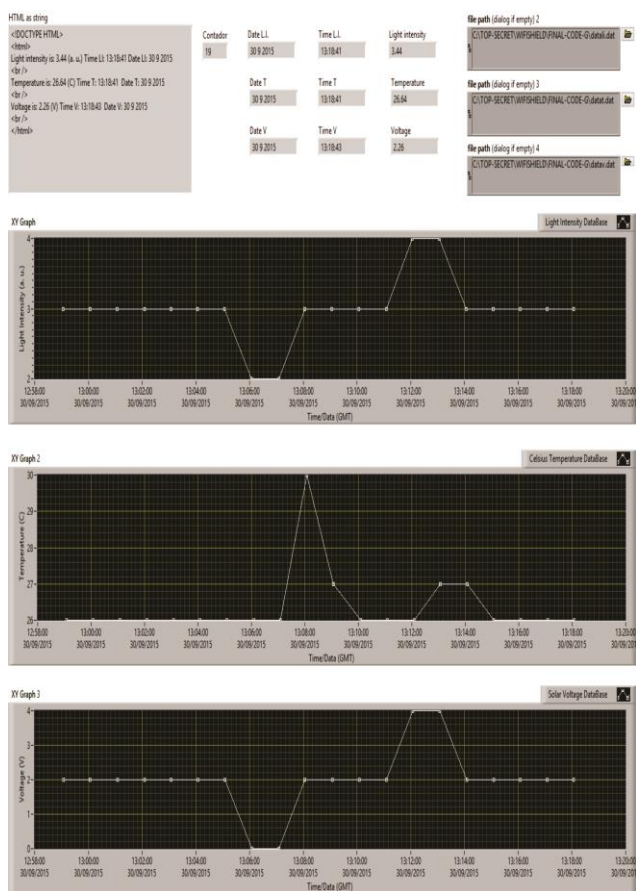


identifying when a certain event occurred), to custom time/data format that can be displayed for the Build XY Graph tool. Figure 6 shows a picture of the whole block diagram or G-code for the developed application software, the location and organization of the icons was optimized with the Clean Up tool.

Figure 6: Complete block diagram or G-code for the application software developed in this work for the treatment of the electrical analog data collected by the Arduino Uno board and sent to the internet by the Arduino Wifi shield.

The Figure 7 shows the front panel of the final form of the NI-Labview application, for starting the data acquisition process the name and location of the text files for light intensity, temperature and solar panel voltage must be introduced. Three partial plots displaying the collected numerical data are also shown.

Figure 7: Final front panel of the software application that construct a database with the digitalized signals broadcasted to the internet by the developed data acquisition system.



As the Arduino Uno and Arduino WiFi Shield are continuously working the application can be initialized anytime and the digital data can be stored continuously for long times.

CONCLUSION

We have constructed a simple data acquisition system that broadcast wirelessly digitalized data to the internet and recovers, stores and displays it with a software application in G-language. The available data is used to construct a database for light intensity, temperature and electric potential for a photovoltaic panel.

ACKNOWLEDGMENTS

We would like to thank INEO/MCT/CNPq for financial support.

Todos os autores declararam não haver qualquer potencial conflito de interesses referente a este artigo.

REFERENCES

AMORIM, H.S. do; DIAS, M.A.; SOARES, V. 2015 Sensores digitais de temperatura com tecnologia one-wire: Um exemplo de aplicação didática na área de condução térmica. *Revista Brasileira de Ensino Física.*, **37**(4): 4310-1-4310-9.

BRASIL. 2006. Decree No. 5.800, OF 8 JUNE 2006. Dispõe sobre o Sistema Universidade Aberta do Brasil – UAB. Available at: http://www.planalto.gov.br/ccivil_03/_Ato2004-2006/2006/Decreto/D5800.htm. Date access: 02/10/2016.

D'AUSILIO, A. 2012. Arduino: A low-cost multipurpose lab equipment. *Behavior Research Methods.* **44**(2):305-313.

GRAPHICAL DEVELOPMENT ENVIRONMENT LABVIEW. 2015. *Systems*. Available at: <http://www.ni.com/labview/pt/>, Date access: 30/09/2016.

GRASEL J.; VONNEGUT W.; DODDS Z. 2010. Bitwise biology: crossdisciplinary physical computing atop the Arduino. *AAAI Spring Symposium Series*. Available at: <https://www.aaai.org/ocs/index.php/SSS/SSS10/paper/view/1042>. Date accessed: 30/09/2016.

GRINDVOLL, H.; VERMESAN, O.; CROSBIE, T.; BAHR, R.; DAWOOD, N.; REVEL, G.M. 2012. A Wireless Sensor Network for Intelligent Building Energy Managementbased o Multi Communication Standards–A Case Study. *Journal of Information Technology in Construction.* **17**(3): 43-62.

HANDCOCK, R.N.; SWAIN, D.L.; BISHOP-HURLEY, G.J.; PATISON, K.P.; WARK, T.; VALENCIA, P.; CORKE, P.; O'NEILL, C.J. 2009. Monitoring Animal Behaviour and Environmental Interactions Using Wireless

Sensor Networks, GPS Collars and Satellite Remote Sensing. *Sensors*. **9**(5): 3586-3603.

KUBÍNOVÁ, Š. ŠLÉGR, J. 2015. Physics demonstrations with the Arduino board. *Physics Education*. **50**(4): 472-474.

KUMAR, K.A.; LINGAM, P.G.; K.MALLIKARJUNA RAO, K.M. 2014. Wireless Sensor Based Remote Monitoring System For Fluoride Affected Areas Using GPRS and GIS. *International Journal of Computer Trends and Technology*. **7**(4): 178-182.

LENGVENIS, P.; MASKELIUNAS, R.; RAUDONIS, V. 2012. Arduino based Controller for the Smart Assistive Mobility Hardware. *Elektronika ir Elektrotechnika*. **18**(9): 75-78.

LÓPEZ-RODRÍGUEZ, F.M.; CUESTA, F. 2016. Andruino-A1: Low-Cost Educational Mobile Robot Based on Android and Arduino. *Journal of Intelligent and Robotic Systems*. **81**(1): 63-76.

MAINWARING, A.; POLASTRE, J.; SZEWCZYK, R.; CULLER, D.; ANDERSON, J. 2002. Wireless Sensor Networks for Habitat Monitoring. *WSNA '02 Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*. Atlanta/Georgia, 2002, USA :88-97.

PEREIRA, A. M.; SANTOS, A.C.F.; AMORIM, H. S. 2016. Estatística de contagem com a plataforma Arduino. *Revista Brasileira de Ensino Física*, **38**(4), e4501- e4501-8.

POZZEBON, A. 2015. Integrating RFID Transponders as Data Loggers in Wireless Sensor Nodes for Outdoor Remote Monitoring Operations. *International Journal of Wireless Information Networks*. **22**(4): 399-406.

RAMESH, M.V. 2009. Real-time Wireless Sensor Network for Landslide Detection. *Third International Conference on Sensor Technologies and Applications*. Athens/Glyfada, Greece. 405-409.

SCHUBERT, T.W.; D'AUSILIO, A.; CANTO, R. 2013. Using Arduino microcontroller boards to

measure response latencies. *Behavior Research Methods*. **45**(4): 1332-1346.

SOUZA, A.R.; PAIXÃO, A. C.; UZÊDA, D.D.; DIAS, M.A.; DUARTE, S.; DE AMORIM, H.S. 2011. The Arduino board: a low cost option for physics experiments assisted by PC. *Revista Brasileira de Ensino de Física*. **33**(1):1702-1702 - 5.

TEIKARI, P.; NAJJAR, R.P.; MALKKI, H.; KNOBLAUCH, K.; DUMORTIER, D.; GRONFIER, C.; COOPER, H.M. 2012. An inexpensive Arduino-based LED stimulator system for vision research. *Journal of Neuroscience Methods*. **211**(2): 227– 236.

THE ARDUINO WIFI SHIELD. 2015. Available at: <https://www.arduino.cc/en/Main/ArduinoWiFiShield>. Date access: 30/09/2016.

THE UNIX TIME. 2015. Available at: https://en.wikipedia.org/wiki/Unix_time. Date access: 30/09/15.

VARANIS, M.; SILVA, A.L.; BRUNETTO, P.H.A. End GREGOLIN, R.F. 2016. Instrumentation for mechanical vibrations analysis in the time domain and frequency domain using the Arduino platform. *Revista Brasileira de Ensino Física*. **38**(1): 1301-1301-19.

WERNER-ALLEN, G.; LORINCZ, K.; WELSH, M.; MARCILLO, O.; JOHNSON, J.; RUIZ, M.; LEES, J. 2006. Deploying a Wireless Sensor Network on an Active Volcano. *IEEE Internet Computing*. **10**(2):18-25.

WHAT IS ARDUINO? 2015. Available at: <https://www.arduino.cc/>. Date access: 30/09/2016.