PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND
SCIENTIFIC RESEARCH

University of Tissemsilt

Faculty of Science and Technology
Science and Technology Department

End-of-study thesis for obtaining the academic Master's degree in
Faculty: Electronic
Speciality: Instrumentation

Presented by: **BELKHELIFA Imad**

&

Loai Shakib Shaapan Herzallah

Thesis

Realization of an acquisition card using Arduino and Visual basic

Before the Jury:

MEHARRAR Aoued	President	MCA	U. Tissemsilt
BERBARA Djilali	Supervisor	MCB	U. Tissemsilt
DJOUDI Lakhdar	Examiner	Prof	U. Tissemsilt

Academic year: 2021-2022

Dedication

We would like to dedicate this work to our family’s.

Thanks

First of all, we thank ALLAH, the creator of ours having given us the strength, the will and the courage to accomplish this modest work. We are very grateful to our supervisor Dr. Djilali BERBARA who proposed the theme of this dissertation, thank you for your interest in this work and for your valuable advice and comments.

Finally, we would like to express our deep gratitude to OUR FAMILIES who always supported us and to all who participated in achieving this thesis. As well as all of the TEACHERS who contributed to our journey

Index

Contents

List of Figures

List of Tables

Abbreviation List

Abbreviation list

DAQ	Data Acquisition
UART	Universal asynchronous receiver-transmitter
ADC	Analog-to-digital converter
IC	Integrated circuit
RTD	Resistance Temperature Detector
USB	Universal serial bus
DOS	Disk operating system
DMA	Direct Memory Access
LED	Light-emitting diode
TTL	Transistor-transistor logic
BIT	Binary digit
S/H	Sample and Hold
CPU	Central processing unit
RAM	Read Only Memory
ROM	Read Access Memory
AC/DC	Alternating current/Direct current
GND	Ground
I2C	Inter-Integrated Circuit
TWI	Two-Wire Interface
CMOS	Complementary metal-oxide semiconductor
EEPROM	Electrically erasable programmable read-only memory
GUI	Graphical user interface
PCB	Printed-Circuit-Board
COM	Communication port
RISC	reduced instruction set computer
MIPS	Million Instructions per Second
MODEM	Modulator-demodulator
COM	Communication

Introduction

INTRODUCTION

Humans get information from the environment that surrounds them through their senses, electronic systems acquire information using sensors, today they are present in all human activities as in systems for the industrial control of processes, automobiles, airplanes, medical devices, household appliances, etc. and have become an essential part of our lives.

The development of technology has given us many advantages, including data acquisition, it provides greater control over the operations of the organization and a faster response to failures that may occur, and this increases the company's efficiency and quality. By mentioning the development of technology that produced Arduino, and this little treasure that opened many doors for people to create electronic applications and systems.

Different projects in different engineering specialties and research, have shown the need for inexpensive but accurate data acquisition systems, and Arduino nano allows us to create low-cost data acquisition and control system.

In this concept, our project represent itself as a cheap solution to acquire analog data and display it in our windows application in digital or graphic form, we can also set digital pins to input or output individually, then we can set the output to high or low level and read the state of the input. More clearly, in our thesis, we propose to simulate and realize an acquisition card and command, we use LM35 sensor which measure the temperature in an environment, we have also created a Voltmeter using various probes to measure the voltage, our application allows the user to see the results in two interfaces which can monitor and save it as text file.

Our work is divided into three chapters. In the first chapter, we present our project by introducing a few concepts such as: data acquisition systems, the system specifications, and we define the procedure to follow by talking in general about the basic elements of Data acquisition systems and the signal conditioning, so we talk in detail about the principle of operation of DAQ. In the second chapter we give a brief overview and description of the electronic components used in the design of our board. Finally, the third chapter will be devoted to the steps of creating our application in visual basic and overview each window and we followed it with simulation and realization results of our system. We will end this thesis with a general conclusion.

Chapter I:

Data Acquisition Systems

I.1 Introduction

The main action of data acquisition systems is sampling signals that measure real-world physical conditions (voltage, current) and convert the resulting samples into digital values that can be handled, for example, by a computer. This chapter describes the fundamental concepts of data acquisition systems; in particular sensors, transducers, communication protocols, and system configurations.

I.2 Data acquisition and control

Data acquisition is the process by which physical phenomena from the real world are transformed into electrical signals that are measured and converted into a digital format for processing, analysis, and storage by a computer.

In a large majority of applications, the data acquisition (DAQ) system is designed not only to acquire data but to act on it as well. In defining DAQ systems, it is, therefore, useful to extend this definition to include the control aspects of the total system. Control is the process by which digital control signals from the system hardware are convened to a signal format for use by control devices such as actuators and relays. These devices then control a system or process. Where a system is referred to as a data acquisition system or DAQ system, it may include control functions as well. [1]

I.3 Data Acquisition Systems fundamentals

Data acquisition (DAQ) systems are the main instruments used in laboratory research from scientists and engineers; in particular, for test and measurement, automation, and so on. Typically, DAQ systems are general-purpose instruments that are well suited for measuring voltage or current signals. However, many sensors and transducers output signals must be conditioned before that a board can acquire and transform it to digital signals. The basic elements of DAQ are shown in Figure. 1 and are:

- Sensors and transducers
- Field wiring
- Signal conditioning
- DAQ hardware
- DAQ software
- PC (with an operating system)

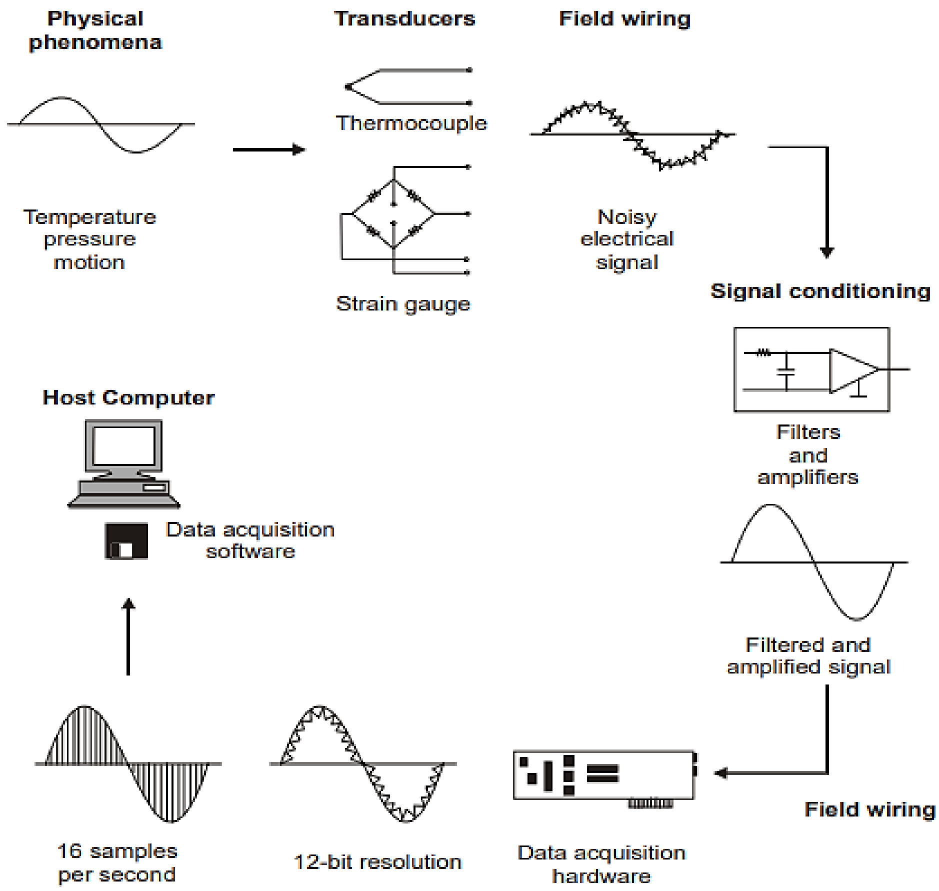


Figure. 1 - Functional diagram of a PC-based Data acquisition system.[3]

Transducers can be used to detect a wide range of different physical phenomena such as movement, electrical signals, radiant energy, and thermal, magnetic, or mechanical energy. They are used to convert one kind of energy into another kind. The type of input or output of the transducer used depends on the type of signal detected or process controlled; in other ways, we can define a transducer as a device that converts one physical phenomenon into another one. Devices with input functions are called sensors because they detected a physical event that changes according to some events for example heat or force. Instead, devices with output.

Functions are called actuators and are used in control systems to monitor and compare the value of external devices.

There are many different types of transducers; each transducer has input and output characteristics and the choice depends on the system: for example, from the type of signal that must be detected and the control system used to manage it.

Table 1 - common transducers.

Quantity being measured	Input device (Sensor)	Output device (actuator)
Light level	Photodiode photo-transistor solar cell	Lamps-LED-fiber optics
Temperature	Thermistor-thermocouple	Heater-fan
Force/Pressure	Pressure switch	Electromagnetic vibration
Position	Potentiometer-encoder	Motor
Speed	Tacho-generator	AC/DC motors
Sound	Carbon microphone	Buzzer-loudspeaker

Each element of the total system is important for the accurate measurement and collection of data from the process or physical phenomena being monitored. [2]

I.3.1 Transducers and sensors

Transducers and sensors provide the actual interface between the real world and the data acquisition system by converting physical phenomena into electrical signals that the signal conditioning and/or data acquisition hardware can accept. Transducers available can perform almost any physical measurement and provide a corresponding electrical output. For example: thermocouples, resistive temperature detectors (RTDs), thermistors, and IC sensors convert temperature into an analog signal, while flow meters produce digital pulse trains whose frequency depends on the speed of flow. Strain gauges and pressure transducers measure force and pressure respectively, while other types of transducers are available to measure linear and angular displacement, velocity and acceleration, light, chemical properties (e.g., CO concentration, pH), voltages, currents, resistances, or pulses. In each case, the electrical signals produced are proportional to the physical quantity being measured according to some defined relationship.

I.3.2 Field wiring and communications cabling

Field wiring represents the physical connection from the transducers and sensors to the signal conditioning hardware and/or data acquisition hardware. When the signal conditioning and/or data acquisition hardware is remotely located from the PC, then the field wiring provides the physical link between these hardware elements and the host computer. If this physical link is an RS-232 (Recently USB to UART) or RS-485 communications interface, then this component of the field wiring is often referred to as communications cabling. Since field wiring and communications cabling often physically represents the largest component of the total system, it is the most susceptible to the effects of external noise, especially in harsh industrial environments. The correct earthing and shielding of field wires and communications cabling is of paramount importance in reducing the effects of noise. This passive component of the data acquisition and control system is often overlooked as an important integral component, resulting in an otherwise reliable system becoming inaccurate or unreliable due to incorrect wiring techniques.

I.3.3 Signal conditioning

Electrical signals generated by transducers often need to be converted to an acceptable form for the data acquisition hardware, particularly the A/D converter which converts the signal data to the required digital format. In addition, many transducers require some form of excitation or bridge completion for proper and accurate operation. The principal tasks performed by signal conditioning are:

- Filtering
- Amplification
- Linearization
- Isolation
- Excitation

I.3.3.1 Filtering

In noisy environments, it is very difficult for very small signals received from sensors such as thermocouples and strain gauges (in the order of mV), to survive without the sensor data being compromised. Where the noise is of the same or greater order of magnitude than the required signal, the noise must first be filtered out. Signal conditioning equipment often contains low pass filters designed to eliminate high-frequency noise that can lead to inaccurate data.

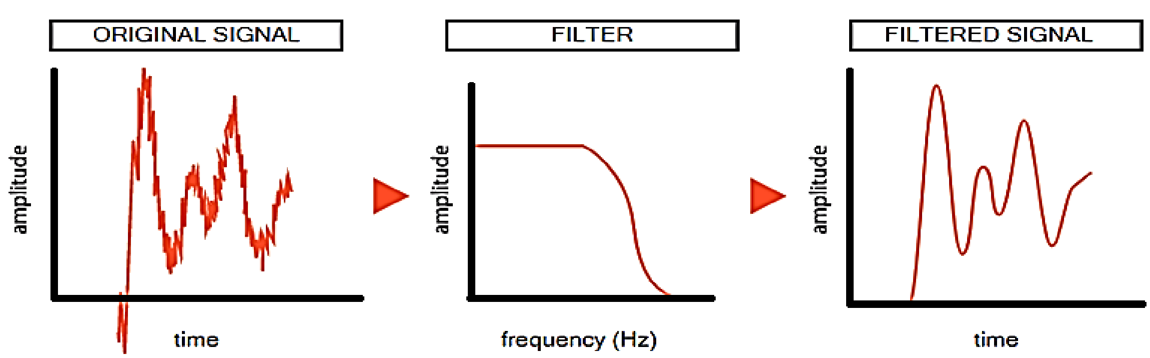


Figure. 2 - Signal filtrations

Having filtered the required input signal, it must be amplified to increase the resolution. The maximum resolution is obtained by amplifying the input signal so that the maximum voltage swing of the input signal equals the input range of the analog-to-digital converter (ADC), contained within the data acquisition hardware. Placing the amplifier as close to the sensor as physically possible, reduces the effects of noise on the signal lines between the transducer and the data acquisition hardware.

I.3.3.2 Linearization

Many transducers, such as some type of thermocouples, display a non-linear relationship to the physical quantity that they are required to measure. The method of linearizing these input signals varies between signal conditioning products. For example, in the case of thermocouples, some products match the signal conditioning hardware to the type of thermocouple, providing hardware to amplify and linearize the signal at the same time. A cheaper, easier, and more flexible method is provided by signal conditioning products that perform the linearization of the input signal using software.

I.3.3.3 Isolation

Signal conditioning equipment can also be used to provide isolation of transducer signals from the computer a where there is possibility that high voltage transients may occur within the system being monitored, either due to electrostatic discharge or electrical failure.

Isolation protects expensive computer equipment from damage and computer operators from injury. In addition, where common-mode voltage levels are high or there is a need for extremely low common-mode leakage current, as for medical applications, isolation allows measurements to be safely obtained, some types of isolation are:

- Galvanic isolation (transformer)
- Optically (opto-isolator)
- Relay
- Magnetoresistance
- Hall effect
- Capacitor

I.3.3.4 Excitation

Signal conditioning products also provide excitation for some transducers. For example, strain gauges, thermistors and RTDs, require external voltage or current excitation signals. [4]

I.3.4 Data acquisition hardware

Data acquisition and control (DAQ) hardware can be defined as that component of a complete data acquisition and control system, which performs any of the following functions:

- The input, processing and conversion to digital format, using ADCs, of analog signal data measured from a system or process, the data is then transferred to a computer for display, storage, and analysis
- The input of digital signals, which contain information from a system or process
- The processing and conversion to analog format, using DACs, of digital signals from the computer the analog control signals are used for controlling a system or process
- The output of digital control signals

Data acquisition hardware is available in many forms from many different manufacturers. Plug-in expansion bus boards, which are plugged directly into the computer's expansion bus, are a commonly utilized item of DAQ hardware. Other forms of DAQ hardware are intelligent stand-alone loggers and controllers, which can be monitored, controlled, and configured from the computer via an RS-232 interface (USB to UART), and yet can be left to operate independently of the computer. Another commonly used item of DAQ hardware, especially in Research and Development (R&D) and test environments, is the remote stand-alone instrument that can be configured and controlled by the computer, via the communication interface.

I.3.5 Data acquisition software

Data acquisition hardware does not work without software, because it is the software running on the computer that transforms the system into a complete data acquisition, analysis, display, and control system. Application software runs on the computer under an operating system that may be single-tasking (like DOS) or multitasking (like Windows, Unix, OS2), allowing more than one application to run simultaneously. The application software can be a full-screen interactive panel, a dedicated input/output control program, a data logger, a communications handler, or a combination of all of these. There are three options available, with regard to the software required, to program any system hardware:

- Program the registers of the data acquisition hardware directly
- Utilize low-level driver software, usually provided with the hardware, to develop a software application for the specific tasks required
- Utilize off-the-shelf application software – this can be application software, provided with the hardware itself, which performs all the tasks required for a particular application; alternatively, third party packages such as LabVIEW and Labtech Notebook provide a graphical interface for programming the tasks required of a particular item of hardware, as well as providing tools to analyse and display the data acquired

I.3.6 Host computer

The PC used in a data acquisition system can greatly affect the speeds at which data can be continuously and accurately acquired, processed, and stored for a particular application. Where high-speed data acquisition is performed with a plug-in expansion board, the throughput provided by bus architectures, such as the PCI expansion bus, is higher than that delivered by the standard ISA or EISA expansion bus of the PC. Depending on the particular application, the microprocessor speed, hard disk access time, disk capacity, and the types of data transfer available, can all have an impact on the speed at which the computer can continuously acquire data. All PCs, for example, are capable of programmed I/O and interrupt-driven data transfers. The use of Direct Memory Access (DMA), in which dedicated hardware is used to transfer data directly into the computer's memory, greatly increases the system throughput and leaves the computer's microprocessor free for other tasks. Where DMA or interrupt-driven data transfers are required, the plug-in data acquisition board must be capable of performing these types of data transfers. In normal operation the data acquired, from a plug-in data acquisition board or other DAQ hardware (e.g., data logger), is stored directly to System Memory. Where the available system memory exceeds the amount of data to be acquired, data can be transferred to permanent storage, such as a hard disk, at any time. The speed at which the data is transferred to permanent storage does not affect the overall throughput of the data acquisition system. [1]

I.4 Communications Cabling

Field wiring is the physical connection from the transducers/sensors to the DAQ hardware. When the signal conditioning and/or DAQ hardware is remotely located from the PC, then it is necessary to use field wiring that provides the physical link. In this case, it is very important to estimate the effects of external noise, especially in industrial environments. One characteristic of all electronic circuits is represented by Noise: it is a random fluctuation in an electrical signal generated by electronic devices. In communication systems, noise is an undesired random disturbance of a useful information signal, kind of noise

I.5 Parameters of a DAQ System

I.5.1 Accuracy and Precision

In the fields of science, the accuracy of a measurement system is the degree of closeness of measurements of a quantity. The precision of a measurement system, instead, is called reproducibility or repeatability of measurements. Relative accuracy is a measure that indicates the capability of the DAQ systems to correct output codes according to its full-scale range.

I.5.2 Noise

Each measurement generates noise as combination of more signals. It is an interference between two terminals. One factor, common-mode noise, indicates the interferences that appear on both measurements' inputs. The majority of common mode interference is attributable to 50 Hz (or 60 Hz) power frequency. [2]

I.5.3 Settling Time

The settling time of an electronic device is the time elapsed from the application of an ideal step input to the time at which the value output has entered and remained within a specified error range. The parameters that can describe settling time are the following: propagation delay and time required to obtain the set output value (Figure. 3)

I.5.4 Acquisition Time

Acquisition Time is a feature of the DAQ systems that indicates the presence of an analog-to-digital converter, which defines the time for going from one situation to a new one according to the system accuracy.

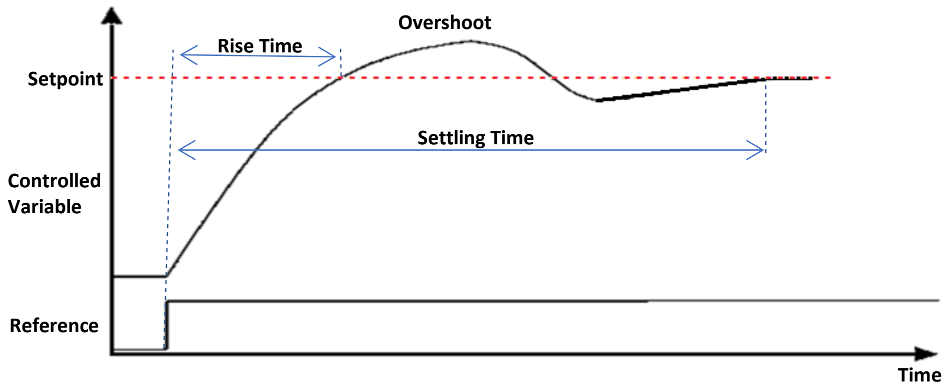


Figure. 3 - Settling time. [3]

I.5.5 DC Input Characteristics

It indicates the value of offset voltages, offset currents, and bias current of an electronic devices.

I.6 Signals classification

Electrical signal can be classified in two classes, analog and digital. [5]

I.6.1 Analog and digital signals

In the real world, physical phenomena, such as temperature and pressure, vary according to nature laws and exhibit properties that vary continuously in time; that is, they are analog time-varying signals.

Transducers convert physical phenomena into electrical signals such as voltage and current for signal conditioning and measurement within DAQ systems. While the voltage or current output signal from transducers has some direct relationship with the physical phenomena they are designed to measure, it is not always clear how that information is contained within the output signal. For example, in the case of a flow meter, the output is a digital pulse train whose frequency is directly proportional to the rate of flow. While the change in the flow rate of a fluid may be varying slowly with time, the output signal is a digital pulse train that may vary quickly in time, dependent on the flow rate, and not on the speed of change in the flow rate.

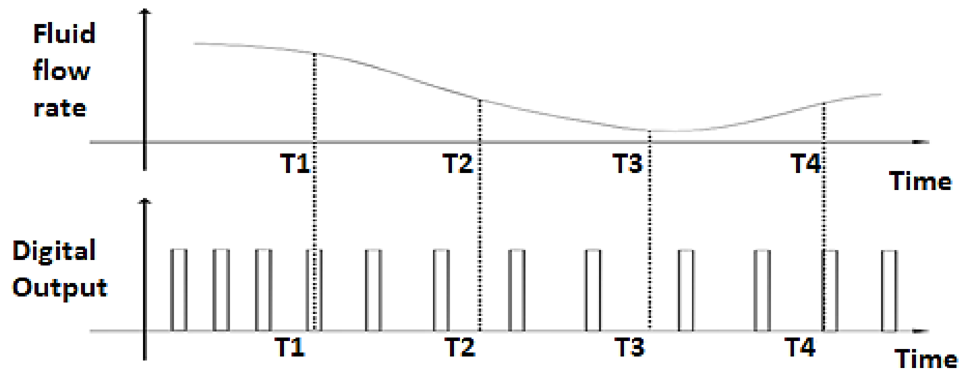


Figure. 4 - The rate of fluid flow and signal at output from a flow meter transducer [7]

I.6.1.1 Digital signals binary signals

A digital, or binary, signal can have only two possible specified levels or states; an "ON" state, in which the signal is at its highest level, and an "OFF" state, in which the signal is at its lowest level. For example, the output voltage signal in a TTL technology can only have two states – the value in "ON" state is 5 V, while it is 0 V in "OFF" state. Control devices, such as relays, and indicators such as LEDs, require digital output signals like those provided on digital I/O boards.

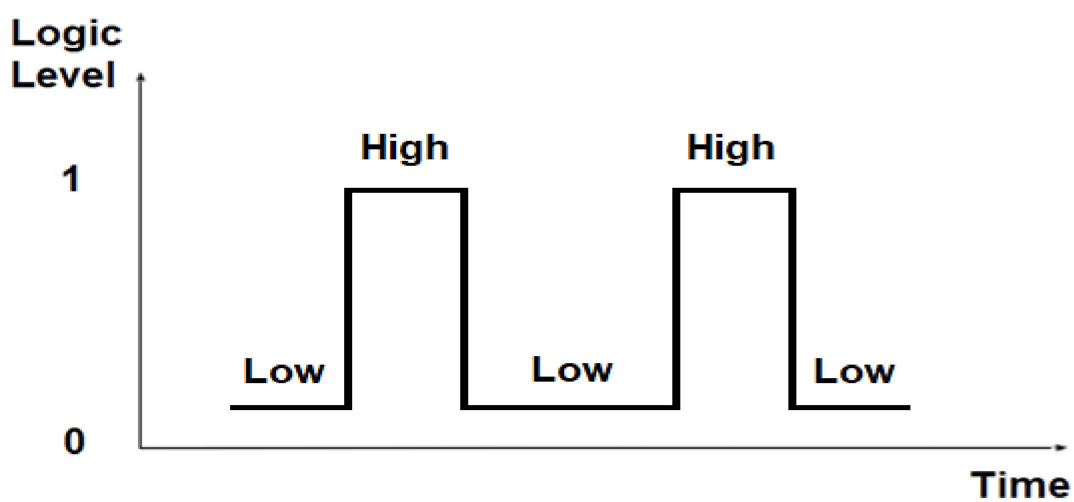


Figure. 5 - A binary digital signal [7]

I.6.1.2 Analog signals

Analog signals contain information within the variation in the magnitude of the signal with respect to time. The relevant information contained in the signal is dependent on whether the magnitude of the analog signal is varying slowly or quickly with respect to time, or if the signal is considered in the time or frequency domains

- Analog DC signals

Analog DC signals are static or slowly varying DC signals. The information conveyed in this type of signal is contained in the level or amplitude of the signal at a given instant in time, not in how this level varies with respect to time.

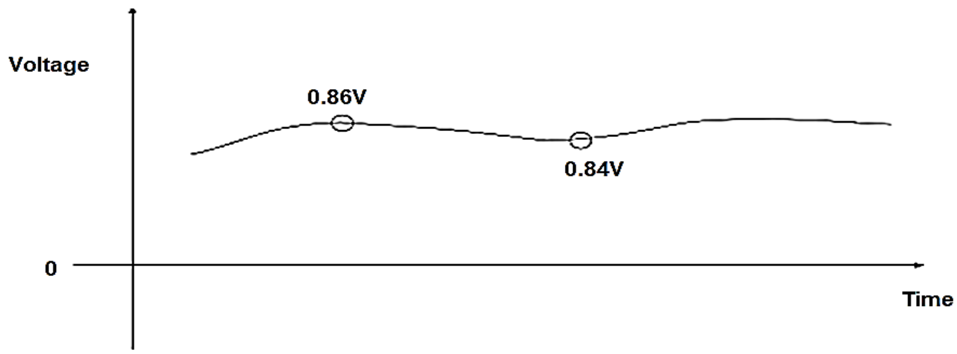


Figure. 6 - An analog DC signal [7]

As the timing of the measurements made of slowly varying signals is not critical, the DAQ hardware would only be required to convert the signal level to a digital form for processing by the computer using an analog-to-digital converter (ADC). Low speed A/D boards would be capable of measuring this class of signal. Temperature and pressure monitoring are just two examples of slowly varying analog signals in which the DAQ system measures and returns a single value indicating the magnitude of the signal at a given instant in time.

- Analog AC signals

The information conveyed in analog AC signals is contained not only in the level or amplitude of the signal at a given instant in time, but also how the amplitude varies with respect to time. The shape of the signal, its slope at a given point in time, the frequency, and location of signal peaks, can all provide information about the signal itself. Further to this, the information extracted from the signal may vary depending on when the measurement of the signal started and ended. DAQ hardware used to measure these signals would require an ADC, a sample clock, to time the occurrence of each A/D conversion, and a trigger to start and/or stop the measurements at the proper time, according to some external event or condition, so that the relevant portion of the signal can be obtained. A high-speed A/D board would be capable of performing these functions.

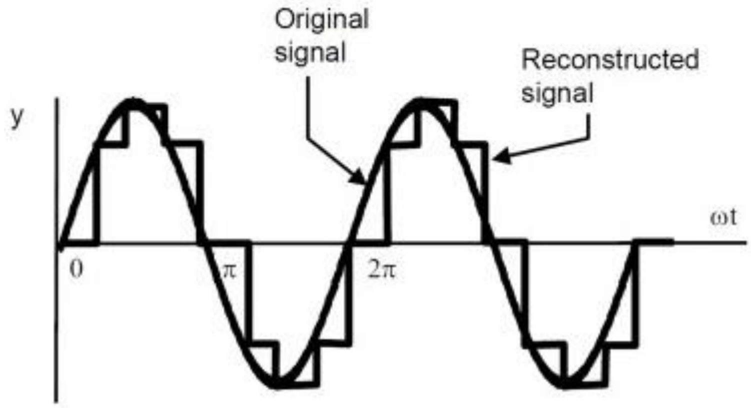


Figure. 7 - an analog AC signal [8]

Since an analog AC signal may vary quite quickly with respect to time, the timing of measurements made of this type of signal may be critical. Hence, as well as converting the signal amplitude to a useful digital form for processing by the computer using an ADC, the DAQ hardware would be required to take the measurements close enough together to reproduce accurately the shape, and therefore the information, contained in the signal. [6]

I.7 Analog to digital conversion A/D

Analog-to-Digital Converters (ADCs) transform an analog voltage to a binary number (a series of 1's and 0's). The number of binary digits bits (n) that represents the digital number and the reference voltage determine the ADC resolution (resolution= $V_{ref}/2^n$). However, the digital number is only an approximation of the true value of the analog voltage at a particular instant because the voltage can only be represented (digitally) in discrete steps, so an ADC works by sampling the value of the input at discrete intervals in time. Provided that the input is sampled above the Nyquist rate, defined as twice the highest frequency of interest, then all frequencies in the signal can be reconstructed.

The connection of digital circuit to sensing device can be done only if the sensors are inherently digital themselves. However, when analog signals are involved in the project, the interface becomes much more complex. In this case, it needs a way to translate analog signals into digital form, an ADC; digital-to-analog converter or DAC performs the opposite operation typically, an ADC is an electronic device that converts an input analog voltage or current to a digital number proportional to the magnitude of the voltage or current. [9]

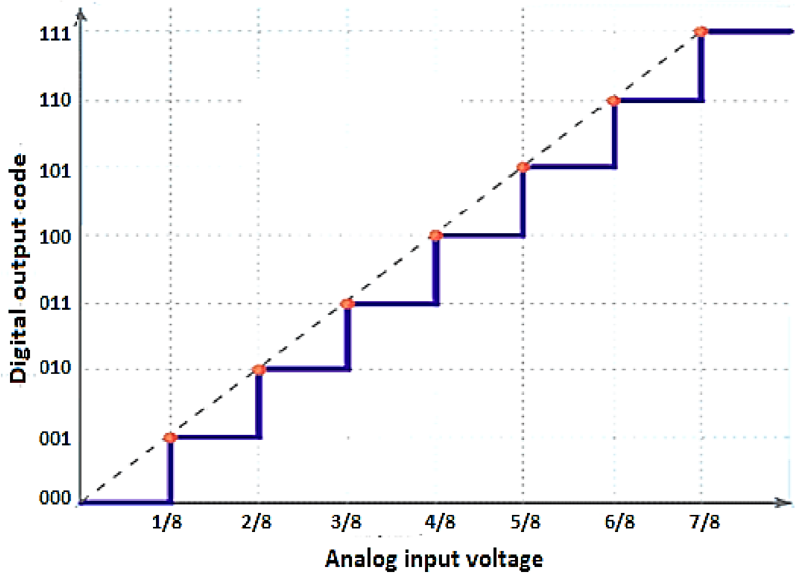


Figure. 8 - Ideal characteristic of 3-BIT analog to digital converter [3]

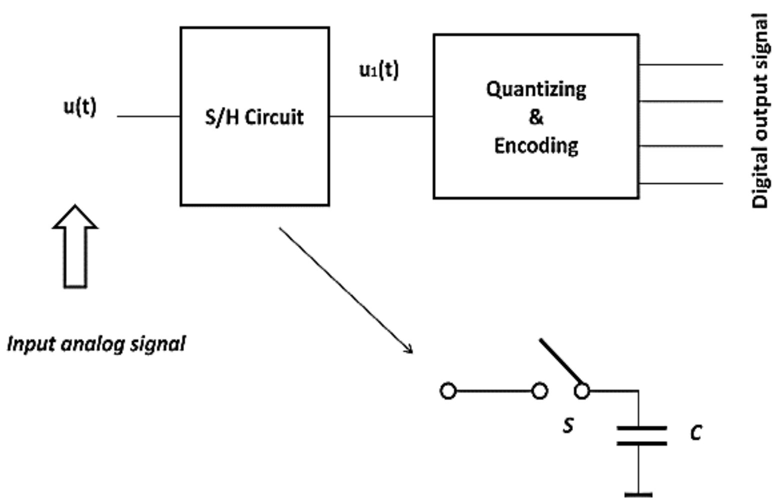


Figure. 9 - analog to Digital Converter Circuit [10]

I.8 Conclusion

In this part, we were interested in the general structure of data acquisition systems, to get a better understanding of how they work, and use it to build our system using Arduino Nano, and acquire accurate measurements in real-time. In the next chapter we will talk about our system components.

Chapter II:

Used Materials Description

II.1 Introduction

In this chapter, we will present the components that make up our acquisition card and deals with the important components of the development board (Arduino nano), type of communication and sensors. In addition, other tools that may help us to make our acquisition card.

The following is a list of topics in this chapter:

- Review development board (Arduino)
- Review some analog sensors
- Material conception

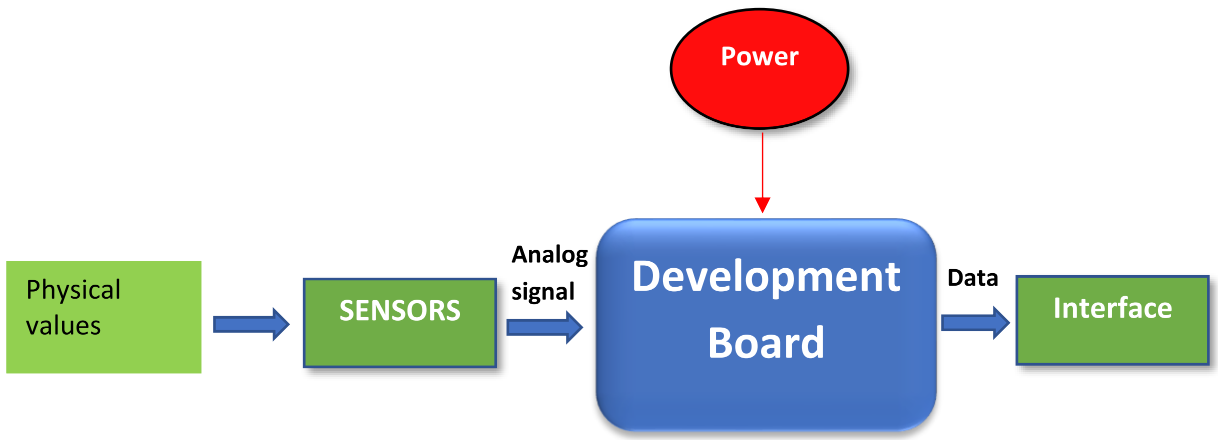


Figure. 10 - Acquisition card diagram.

II.2 Project description

The acquisition card in our project can receive voltage proportional to physical value from an analog sensor, and send it to the computer through serial communication using USB to UART converter embedded in an Arduino card, and it can also read and write digital pins.

Our acquisition card contains four analog channels that can be connected to an analog sensor, The value is received and displayed by a pc application that we have developed using visual basic software, through which we can select the sensor connected to each channel.

The block diagram below presents our acquisition card diagram:

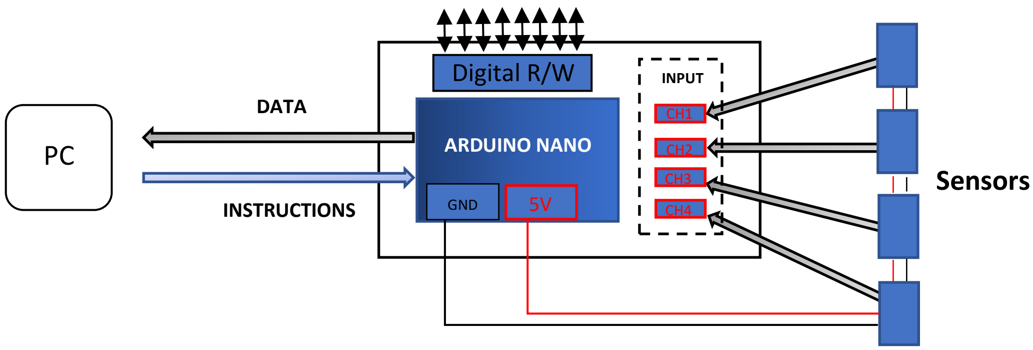


Figure. 11 - Simple Diagram of our system

II.3 Components description

II.3.1 development board (Arduino)

Arduino is an open-source electronics platform based on easy-to-use hardware and software and it is a small microcontroller board with a USB plug to connect to a computer and several connection sockets that can be wired to external electronics such as motors, relays, light sensors, laser diodes, loudspeakers, microphones, and more. They can either be powered by computer through the USB connection, or up to 12V battery or a power supply. They can be controlled or programmed by the computer and then disconnected to work independently. [11]

II.3.1.2 Measurement system

A measuring system exists to provide information about the physical value of some variable being measured. In simple cases, the system can consist of only a single unit that gives an output reading or signal according to the magnitude of the unknown variable applied to it. However, in more complex measurement situations, a measuring system consists of several separate elements, these components might be contained within one or more boxes, and the boxes holding individual measurement elements might be either close together or physically separate. The term measuring instrument is used commonly to describe a measurement system, whether it contains only one or many elements. [12]

II.3.2 serial communication UART

UART, or universal asynchronous receiver-transmitter, UART can work with many different types of serial protocols that involve transmitting and receiving serial data. Data is transferred bit by bit in serial communication using a single line or wire. In two-way communication, we use two wires for successful serial data transfer. Depending on the application and system requirements, serial communications need less circuitry and wires, which reduces the cost of implementation.

By definition, UART is a hardware communication protocol that uses asynchronous serial communication with configurable speed. Asynchronous means there is no clock signal to synchronize the output bits from the transmitting device going to the receiving end. [13]

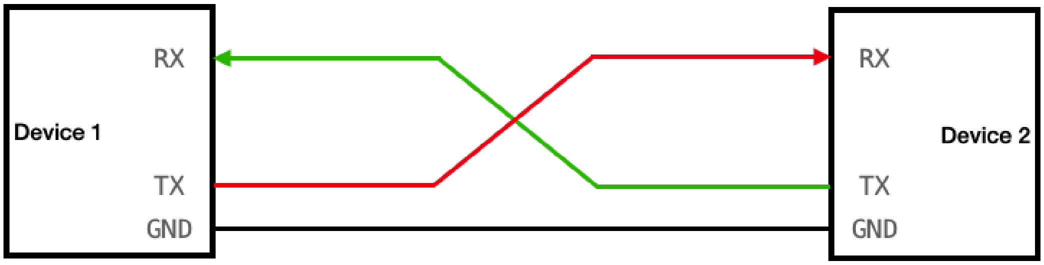


Figure. 12 - Two UARTs directly communicate with each other. [14]

The two signals of each UART device are named:

- Transmitter (TX)
- Receiver (RX)

II.3.3 Analog to digital converter (ADC)

Analog to digital converter is an electronic integrated circuit which transforms a signal from analog (continuous) to digital (discrete) form, Analog signals are directly measurable quantities and Digital signals only have two states. For digital computer, we refer to binary states, 0 and 1. Look at chapter 1. [15]

II.3.4 microcontroller

A microcontroller is a small computer with most of the necessary support chips onboard. All computers have several parts in common, namely:

- A central processing unit (CPU) that 'executes' programs written in ROM.
- A random-access memory (RAM) where variable data can be stored and processed.
- A read only memory (ROM) where programs to be executed can be stored.
- Input and output (I/O) devices that enable communication to be established with the out-side world. [16]

II.4 This is Arduino Nano

The Arduino Nano is a small, a complete and a breadboard-friendly designed board based on ATmega328P. It has more or less the same functionality of Arduino Duemilanove but in a different package. It lacks only a DC power jack and works with a Mini-B USB cable instead of a standard one.

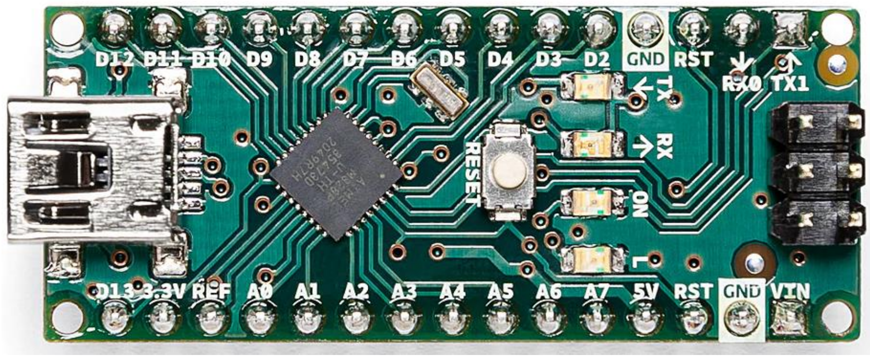


Figure. 13 - Arduino Nano.

II.4.1 Arduino types

There are many types of Arduino boards, and each has a specific shape and features, and we can also use it in our project.

The given below are some Arduino boards we can use to make acquisition card:

- Arduino Uno
- Arduino mega
- Arduino Leonardo
- Arduino due
- Arduino Duemilanove[17]

II.4.2 Why Arduino nano?

Arduino Nano has many features despite its small size and low-cost, this little board permits us to make various interesting projects and applications.

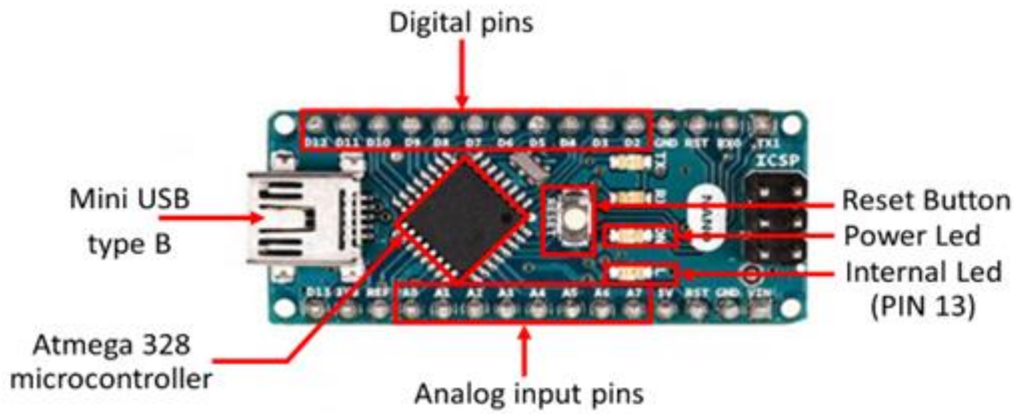


Figure. 14 - Arduino Nano components.

In the picture above, the Arduino Nano block is where the processor and memory are located, the rest ensures external communication, USB port, voltage regulator and etc.

II.4.3 Technical specification

The board comes with a crystal oscillator of 16 MHz frequency. It's used to create a precise frequency clock. There's one limitation of using Arduino Nano i.e., it doesn't come with a DC power jack, which means you must add connector to supply an external power source through a battery for example. [18]

The table below represents Arduino Nano characteristics:

Table 2 - Arduino Nano characteristics.

Column1	Column2
Microcontroller	ATmega328
Architecture	AVR – 8 Bits
Operating Voltage	1.8 – 5.5V
Flash Memory	32 KB of which 2 KB used by the bootloader
SRAM	2 KB
Clock Speed	16 MHz
Analog IN Pins	8
EEPROM	1 KB
DC Current per I/O Pins	40 mA (I/O Pins)
Input Voltage	7-12V
Digital I/O Pins	22 (6 of which are PWM)
PWM Output	6
Power Consumption	19 mA
PCB Size	18 x 45 mm
Weight	7 g
Product Code	A000005

II.5 Atmega328P

The picoPower ATmega328/P is a low-power CMOS 8- bit microcontroller grounded on the AVR enhanced RISC armature. By executing important instructions in a single clock cycle, the ATmega328/P achieves throughputs close to 1 MIPS per MHz. This empowers system contrivers to optimize the device for power consumption versus processing speed.

In this work, we target an 8-bit AVR microcontroller, the Atmel ATmega328P. It has 32 KB of flash memory, 1 KB of EEPROM, and 2 kB of RAM. [19]

The figure below represents the pinout configuration of ATmega328P in SMD package:

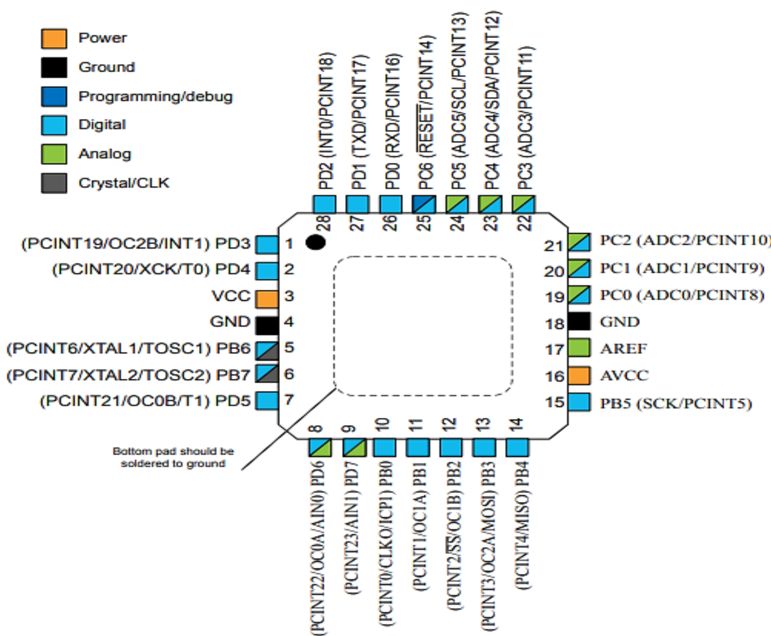


Figure. 15 - ATMEGA328P pins description.

II.6 USB to TTL conversion chip (CH340)

CH340 is a USB bus converter chip, that converts USB to a serial port or printer port. In serial port mode, CH340 provides a common MODEM signal, to expand the UART interface of the computer or upgrade common serial devices to USB bus directly. [20]

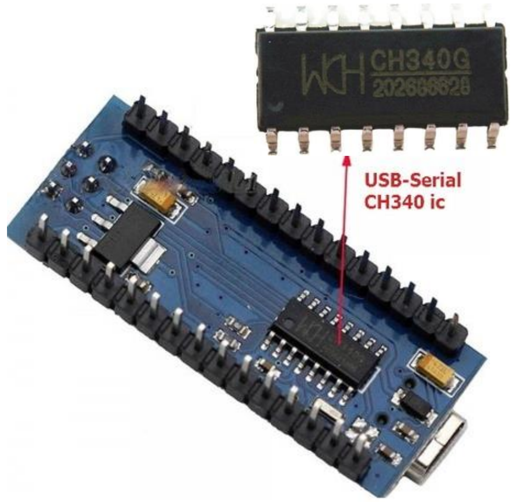


Figure. 16 - CH340.

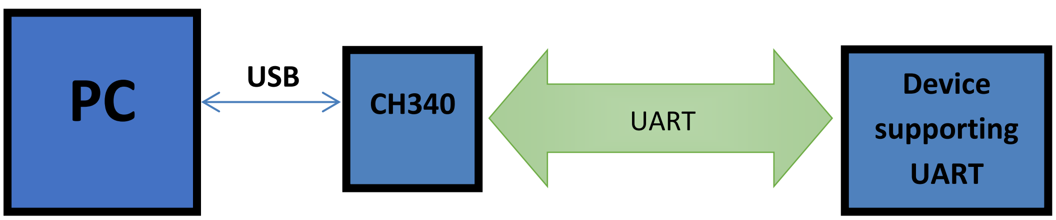


Figure. 17 - CH340 Communication diagram.

II.7 Sensors

The first element in any measuring system is the primary sensor: this gives an output that is a function of the measurand (the input applied to it). For most but not all sensors, this function is at least approximately linear. Some examples of primary sensors are a liquid-in-glass thermometer, a thermocouple, and a strain gauge. In the case of a mercury-in-glass thermometer, because the output reading is given in terms of the level of the mercury, this particular primary sensor is also a complete measurement system in itself. However, in general, the primary sensor is only part of a measurement system. [21]

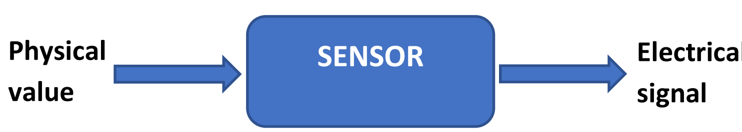


Figure. 18 - Sensor simple diagram.

II.7.1 LM35

LM35 device has an advantage over linear temperature sensors calibrated in Kelvin, as the user is not required to add additional circuit to convert the output to obtain convenient Centigrade scaling. The LM35 device does not require any external calibration or trimming to provide typical accuracies of ±¼°C at room temperature and ±¾°C over a full −55°C to 150°C temperature range. Lower cost is assured by trimming and calibration at the wafer level. The low-output impedance, a linear output, and precise inherent calibration of the LM35 device. [22]

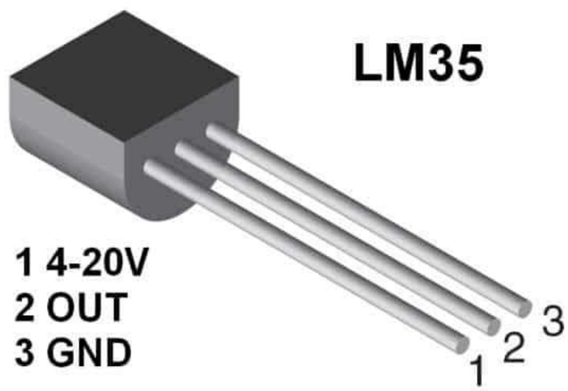


Figure. 19 - LM35 pinout.

II.7.2 LM35 features

- Calibrated Directly in Celsius (Centigrade)
- Linear + 10-mV/°C Scale Factor
- 0.5°C Ensured Accuracy (at 25°C)
- Rated for Full −55°C to 150°C Range
- Operates from 4 V to 30 V
- Less than 60-µA Current Drain
- Low Self-Heating, 0.08°C in Still Air
- Non-Linearity Only ±¼°C Typical
- Low-Impedance Output, 0.1 Ω for 1-mA Load

II.8 Bluetooth (HC-06)

The HC-06 Bluetooth module is a slave Bluetooth module designed for wireless serial communication. It is a slave module meaning that it can receive serial data when serial data is sent out from a master Bluetooth device (device able to establish the connection and send serial data through the air like smartphones, PC).

When the module receives wireless data, it is sent out through the serial interface exactly at it is received. No source code specific to the Bluetooth module is needed at all in the Arduino chip. [23]

Figure. 20 - shows the HC-06 Bluetooth module.

II.9 Conclusion

This chapter focused on the description of our project, to detail its functioning as well as its task. We have mentioned the electronic components that make up our system. when we briefly presented ARDUINO NANO, and we notice that with its simplicity we can use it as a control unit for our system. In the next chapter, we will talk about simulation and realization part of our acquisition card.

Chapter III:

Simulation and Realization

III.1 Introduction

After general studies of the various elements constituting our acquisition card, we move now to the physical realization of our project, in this chapter we will talk about different tools and programs that help us to make the simulation and the realization of our card.

III.2 Progress of the Project

Our realization project was made of two parts Software and Hardware:

- In the software: build an application interface in visual basic and programming the card in Arduino IDE.
- In the hardware: design our PCB using Proteus software and make it with tonner transfer method.

III.3 Our system functioning

Our system runs on Arduino's internal reference 1.1V to get better accuracy and resolution (1.1/1024≈1mV), but we can run it on an external reference VCC≈5V, our card is able to measure positive voltage between 0 and ≈ 1.1V or higher volts using voltage divider (probe 1, probe 2... etc.), while internal reference in Atmega328p is uncertain we can calibrate our device by adjusting the reference voltage to real value. The connection between our card and the computer is done through a serial port (USB to TTL) integrated in Arduino card, we have chosen the maximum Baud Rate supported by Arduino nano (115200), to start data transfer VB sends "ANA" for analog and "DIG" for digital, when Arduino reads it, it starts the data transfer. received data are split and displayed separately (ch1, ch2, ch3, ch4) in VB.

Note: Analog and digital don't work at the same time when we start receiving data after sending "ANA" we can only write with the digital pins.

III.4 Simulation

Before moving on to the realization, we carried out a simulation in proteus software; the simulation allowed us to test and verify the smooth running of our system and to validate the previous stages to move on to the final stage which is the practical realization of this project. Simulation plays a very important role, it allows us to test if our circuit works well before making the real one, this is why we opted for a simulation on Proteus in order to have an idea on the functioning of our system.

III.4.1 Windows application

Using programs like visual basic; a program that is written to run under Microsoft Windows operating system can be made, also called a "Windows app".

III.4.1.1 Visual basic

Visual Basic (VB) is an object-oriented programming language and environment from Microsoft that provides a graphical user interface (GUI) which allows programmers to modify code by simply dragging and dropping objects and defining their behaviour and appearance. VB is derived from the BASIC programming language and is considered to be event-driven and object-oriented. [24]

Figure. 21 - visual basic program icon

III.4.1.2 Brief exposition of visual basic steps to make application

- After running Visual Basic, the start interface appears, through which we can create a new application.

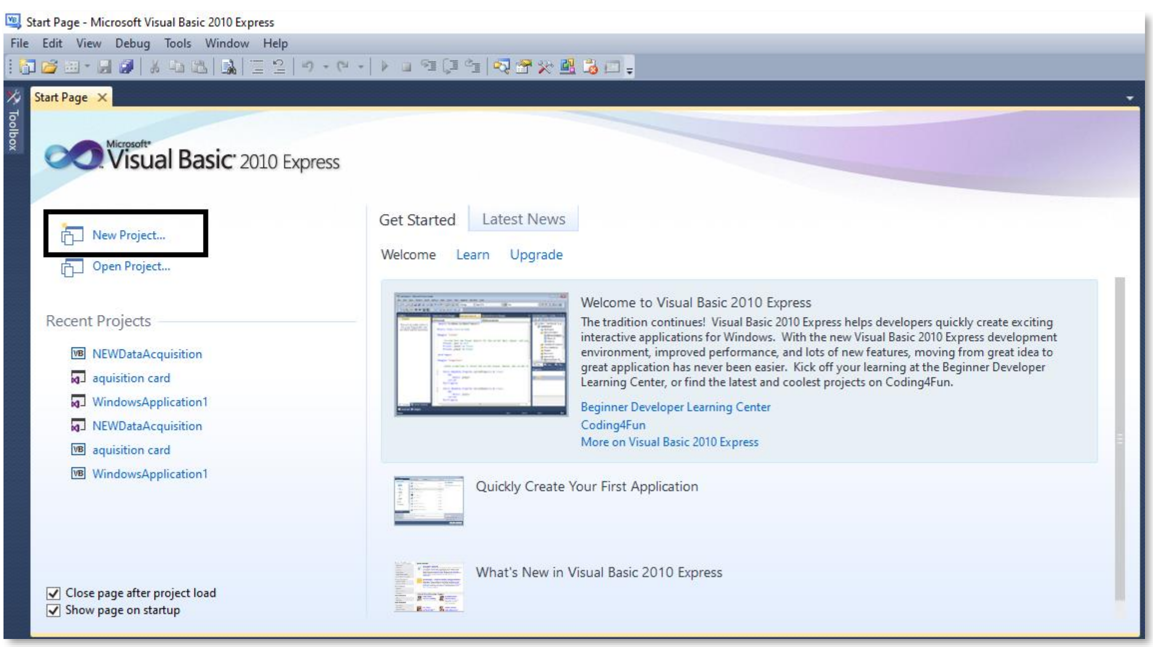


Figure. 22 - Create new project

❖ Then we choose the type of application we want to work on, which is <<Windows Forms Application >>.

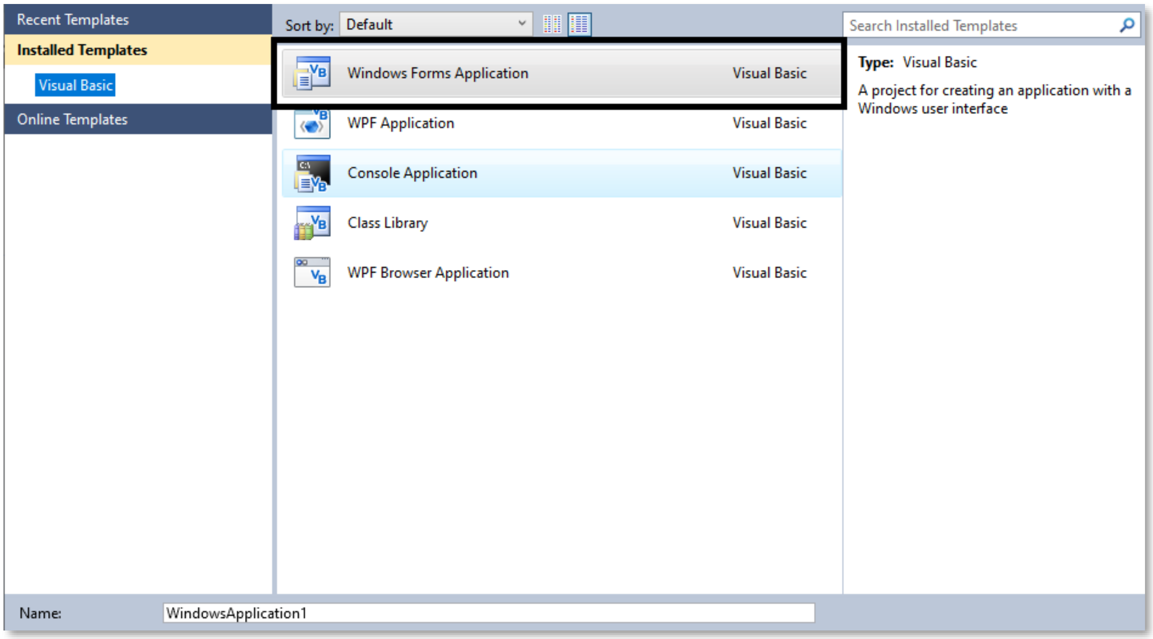


Figure. 23 - Select type of application

❖ After these steps, the application interface will appear, in which we will find all the objects and their properties through which we will make the application, such as (Button, Panel, Label … etc).

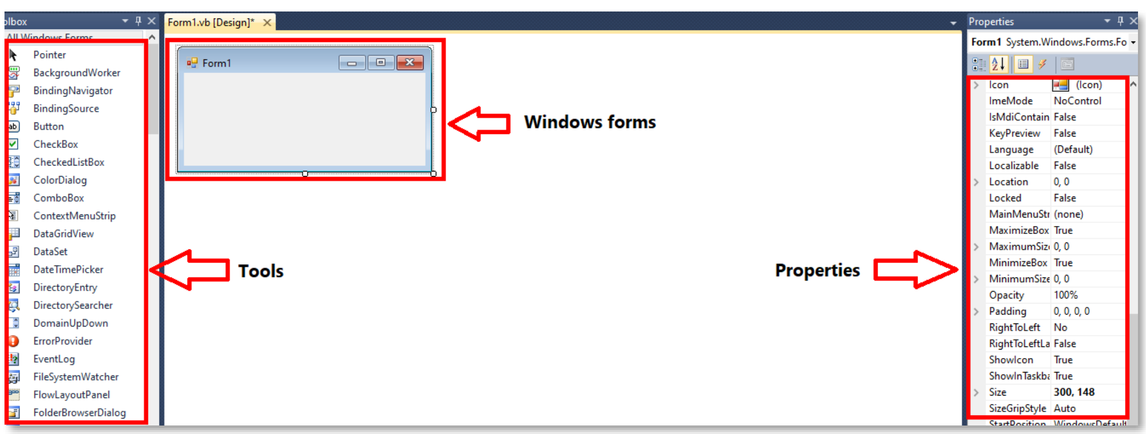


Figure. 24 - New form interface

❖ In this case, we are ready to start programming our form, and we can find also an errors list that can help us to solve some problems we could face.

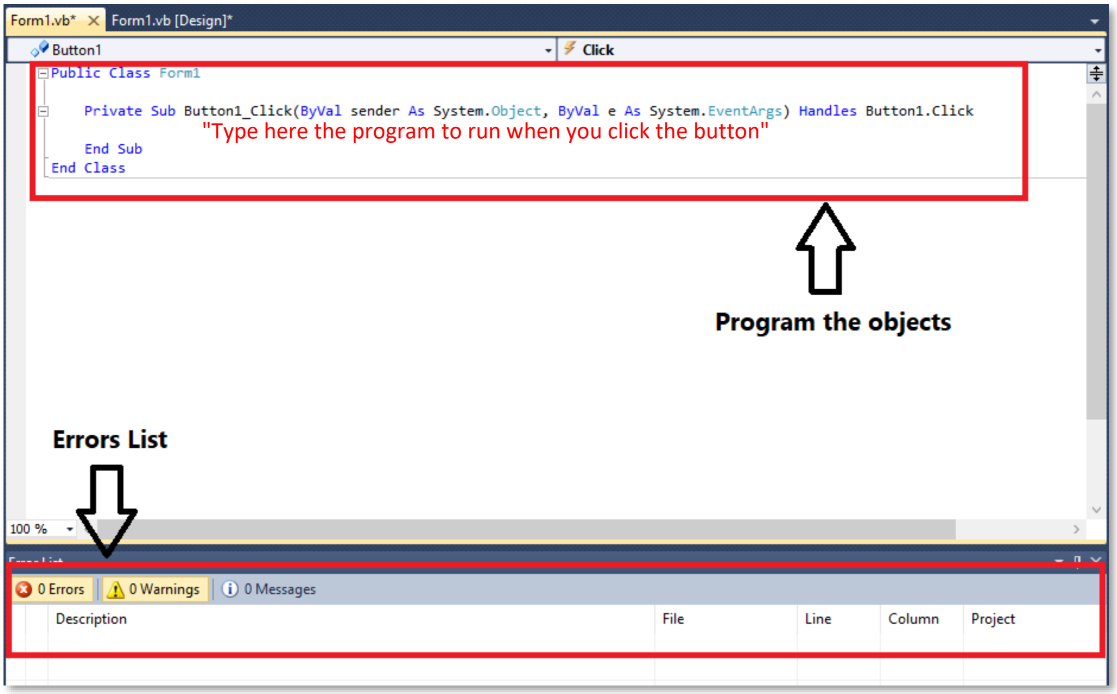


Figure. 25 - Form program

III.4.2 Our application overviews

In the following, we present the fruit of our work in Visual Basic by exposing the different interfaces.

III.4.2.1 Application interfaces

We have created five interfaces which are included in the topics below:

- The main interface: it shows several applications and setting.

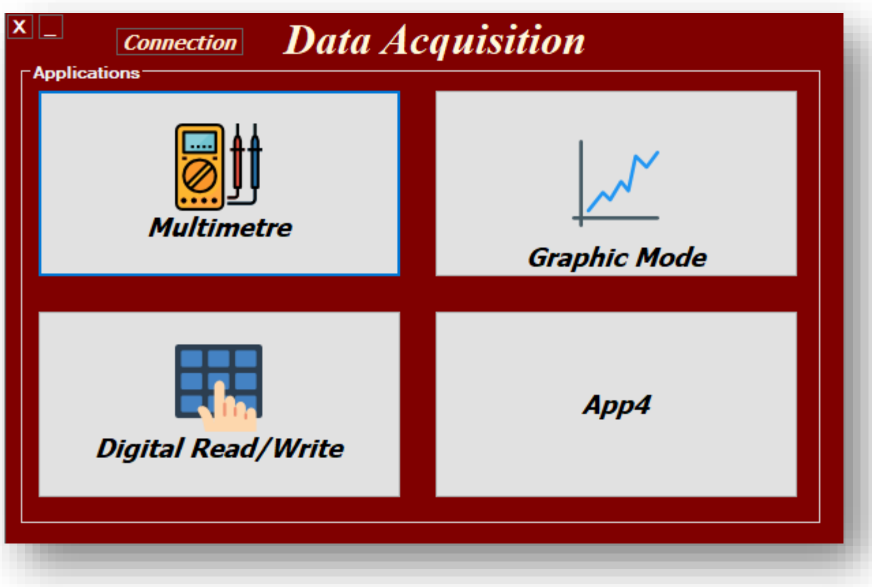


Figure. 26 - Main application

- Digital Multimetre interface: we can read the values of Analog sensors, and we can also use them as a voltmeter by changing the settings.

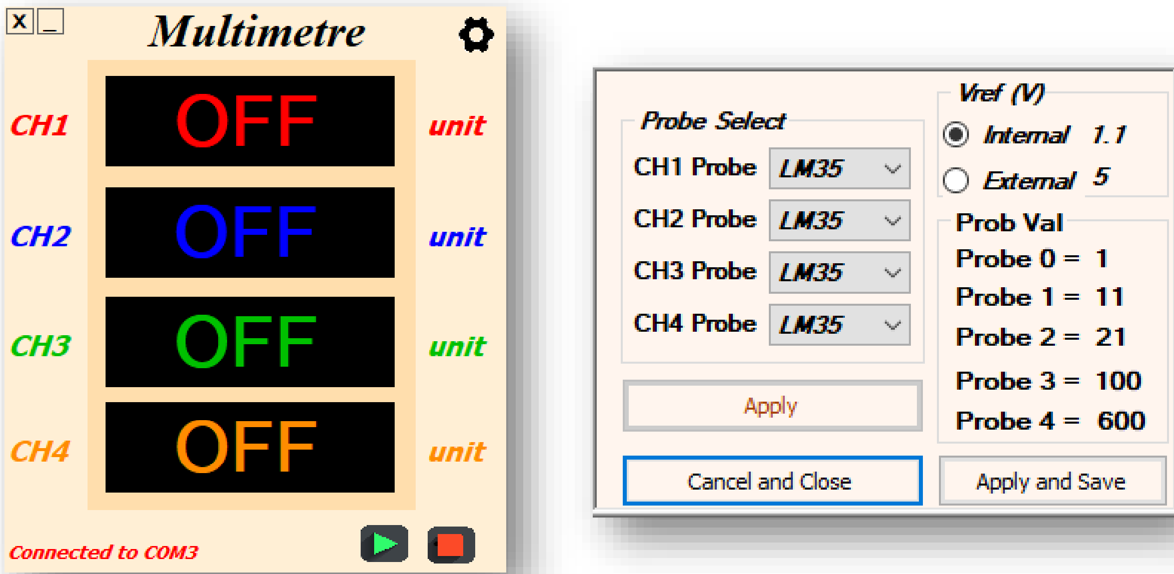


Figure. 27 - Digital multimetre

- Graphic interface: this graphic mode can plot the values read by analog inputs with respect to time, and it contains many features like:
  - ✓ Select the period of acquiring data (end time) or let it run continuously.
  - ✓ Select window limit (start time, end time, V max and V min).
  - ✓ We can hide or show the graph of anu channel that we want to see.
  - ✓ We can also store and export all the values in respect to time in our computer as text file.

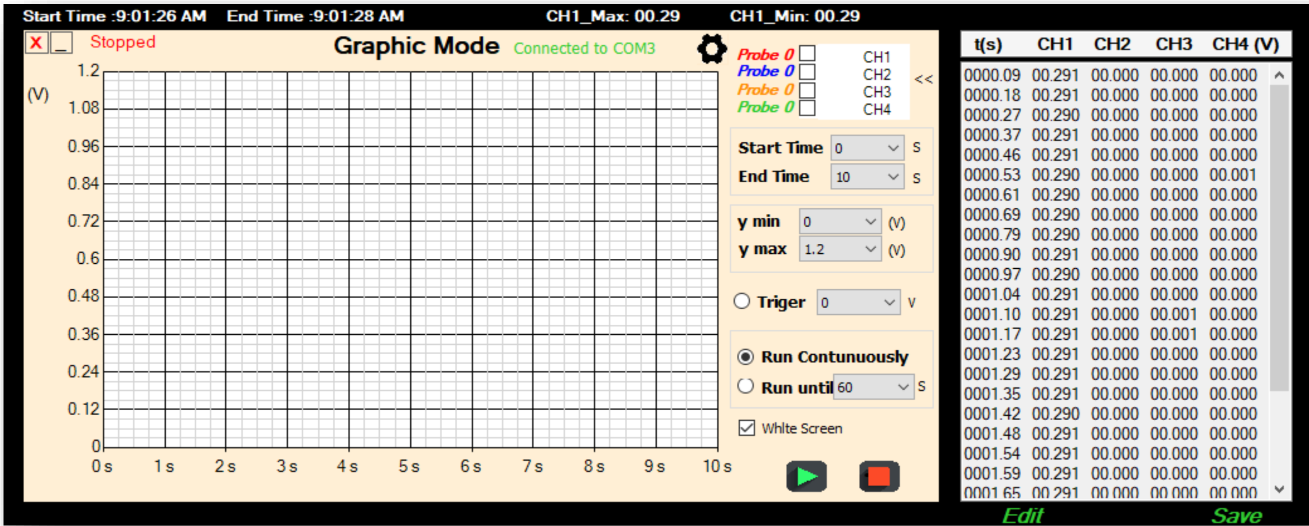


Figure. 28 - Graph mode

- Connection setting interface: this form allows us to show the available COMs in the computer and connect to our card COM (via USB to serial or Bluetooth COM).

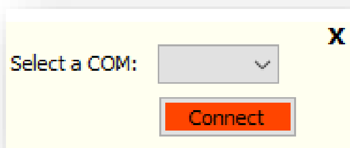


Figure. 29 - COM setting

- Digital read write (R/W) interface: A digital signal can have two states (open or closed, high or low, on or off). This form below allows us to configurate any of those 8 pins as an input, so we can read its state or as an output to change its state.

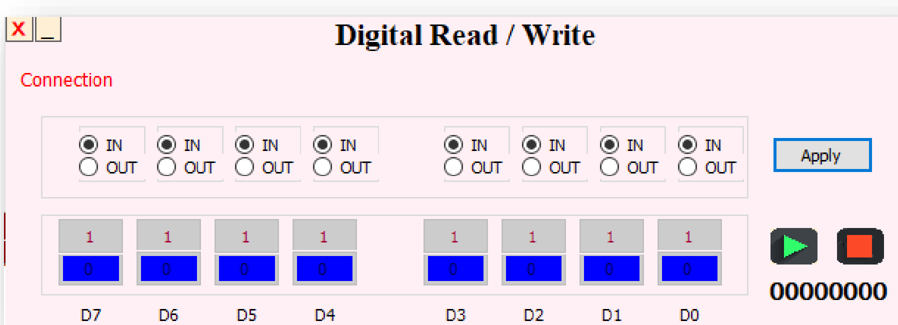


Figure. 30 - Digital write

III.5 Arduino IDE

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board. we used Arduino 2.0 software.

The Arduino language is inspired by several languages. We find similarities with C, C++ and Java languages imposes a structure characteristic typical of on-board computing.

- The "**Setup**" function contains all the operations necessary for configuration of the card (input/output directions, serial communication speeds, etc.).
- The "**Loop**" function is executed in a loop until the card is powered off or restarted by the reset button. This loop is absolutely necessary on microcontrollers since they have no operating system.

If we omit this loop, at the end of the produced code, it will be impossible to regain control of the Arduino board. [25]

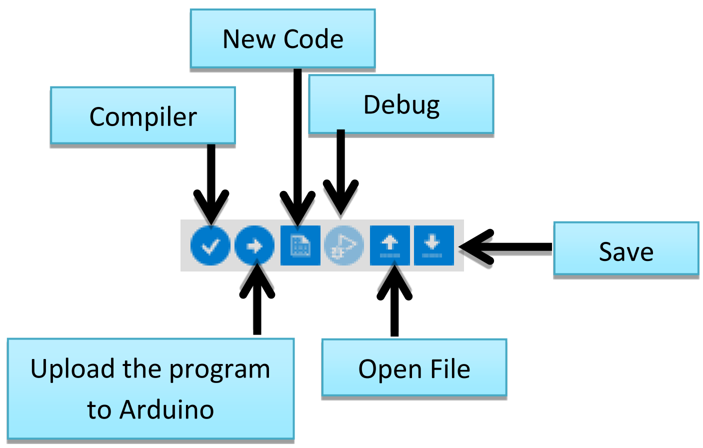


Figure. 31 - Arduino button bar

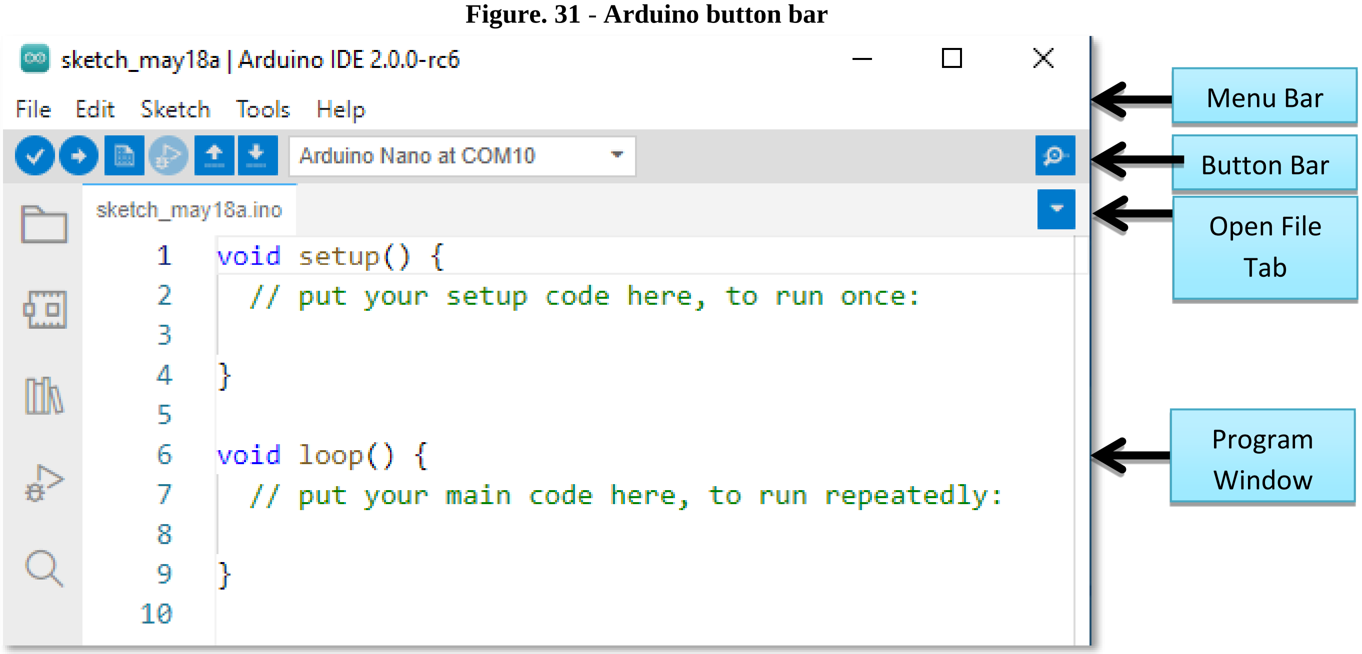


Figure. 32 - Arduino programming Platform interface

The software also includes a serial monitor which allows us to display text messages (Data) sent by the Arduino board and to send characters to the Arduino board, we can also select the baud rate.

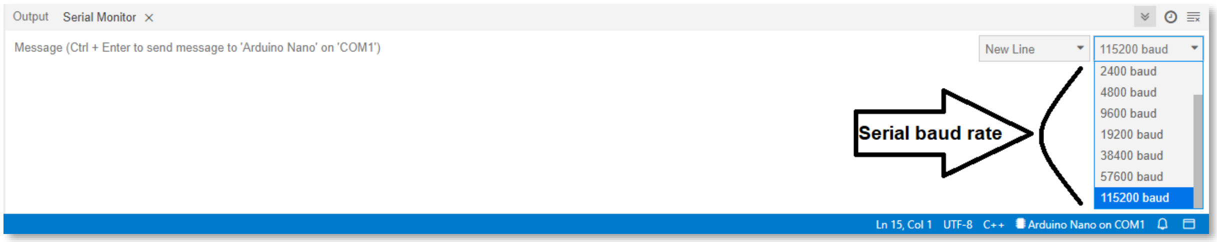


Figure. 33 - Arduino Serial Monitor

III.5.1 Program

Before sending the file (.hex) to the Arduino nano board, it is necessary to select the type of card (Arduino nano ATMEGA328P) and COM port (COM1) as an example in the following figure:

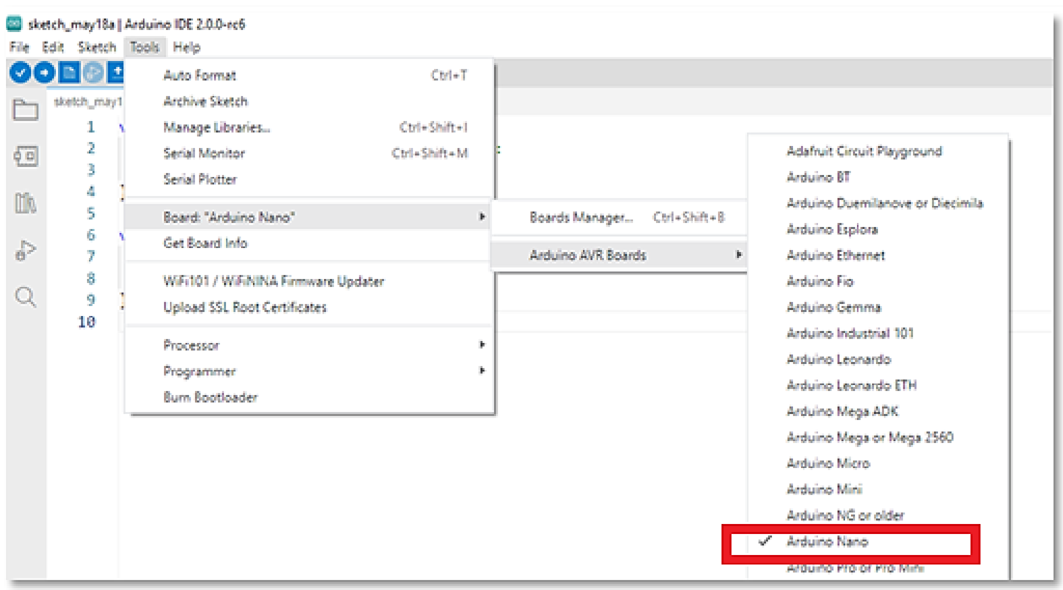


Figure. 34 - Arduino Board selection

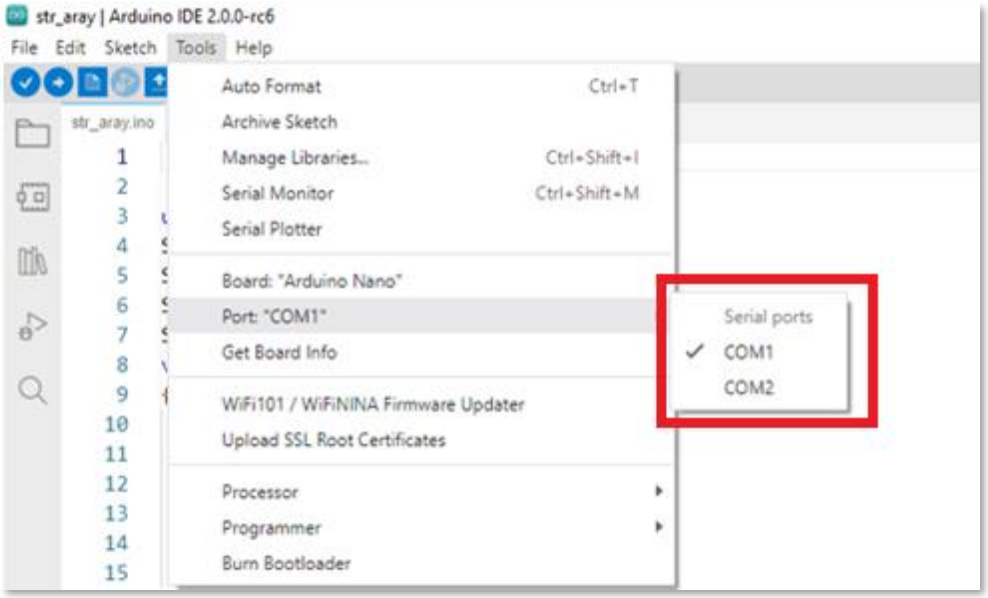


Figure. 35 - Port selection

III.5.1.1 Program upload steps

A simple chained manipulation must be followed in order to inject a code to Arduino through the USB port.

1. We design or open an existing program with the Arduino IDE software.

2. We check this program with the Arduino software (compiler).

3. If errors are reported, the program is modified.

4. The program is loaded into the card.

5. We wire the compiled program.

6. Program execution is automatic after a few seconds.

7. The card is powered either by the USB port or by an autonomous power source for example (5-volt battery).

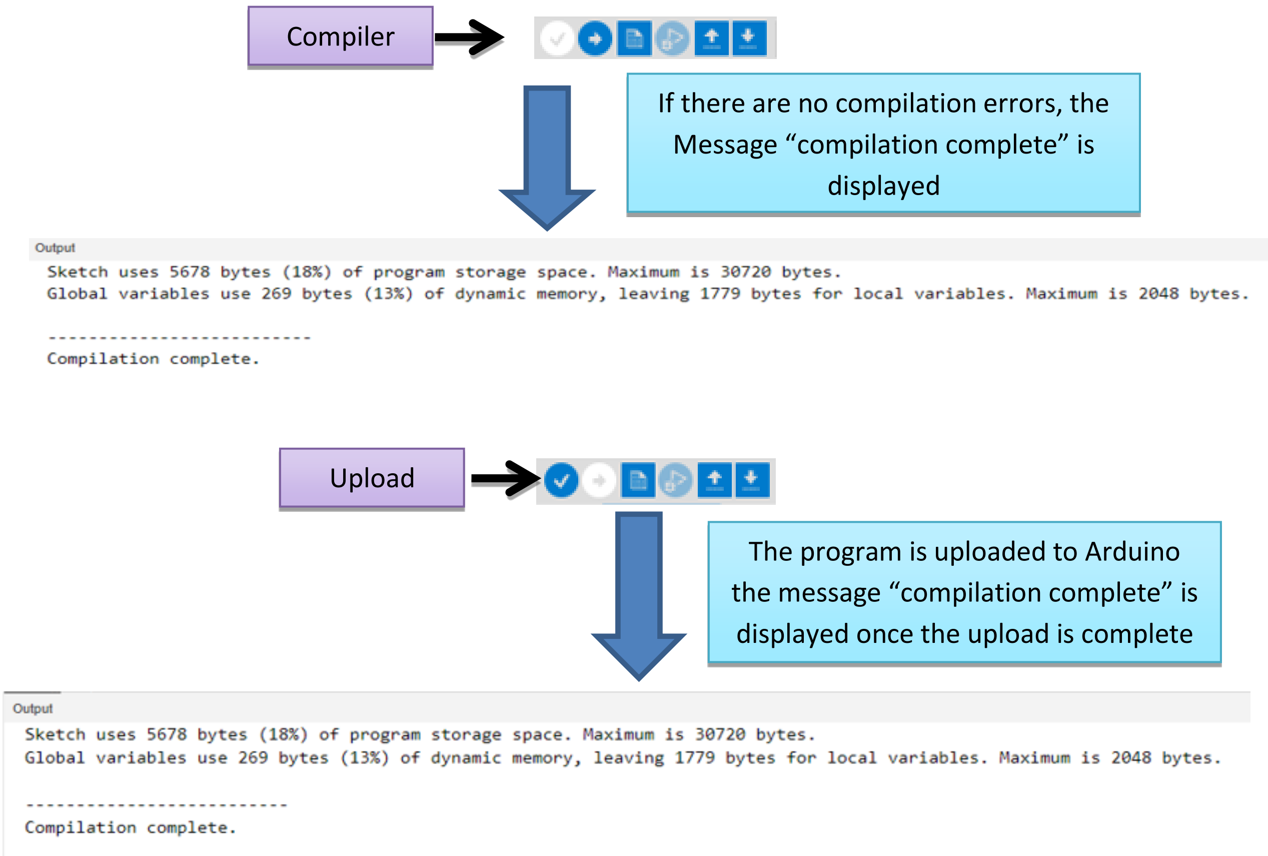


Figure. 36 - Arduino program upload steps

III.6 Proteus 8 professional

Proteus 8 Professional is a software which can be used to draw schematics, PCB layout and even simulate the schematic. It is developed by Labcenter Electronics Ltd.

III.6.1 Starting simulation in Proteus

After starting the program and create new project, we picked the components we need from <<Pick devices>> shown in the image bellow:

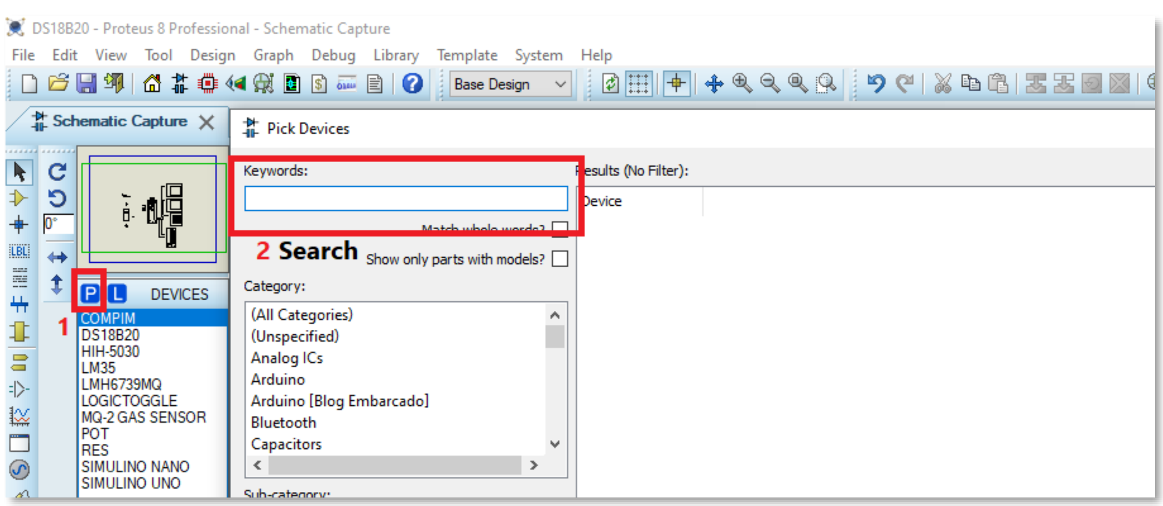


Figure. 37 - Pick devices

Now we can connect between our components and upload the program (.hex) in Arduino and start the simulation. This is the final simulation schematic of our project.

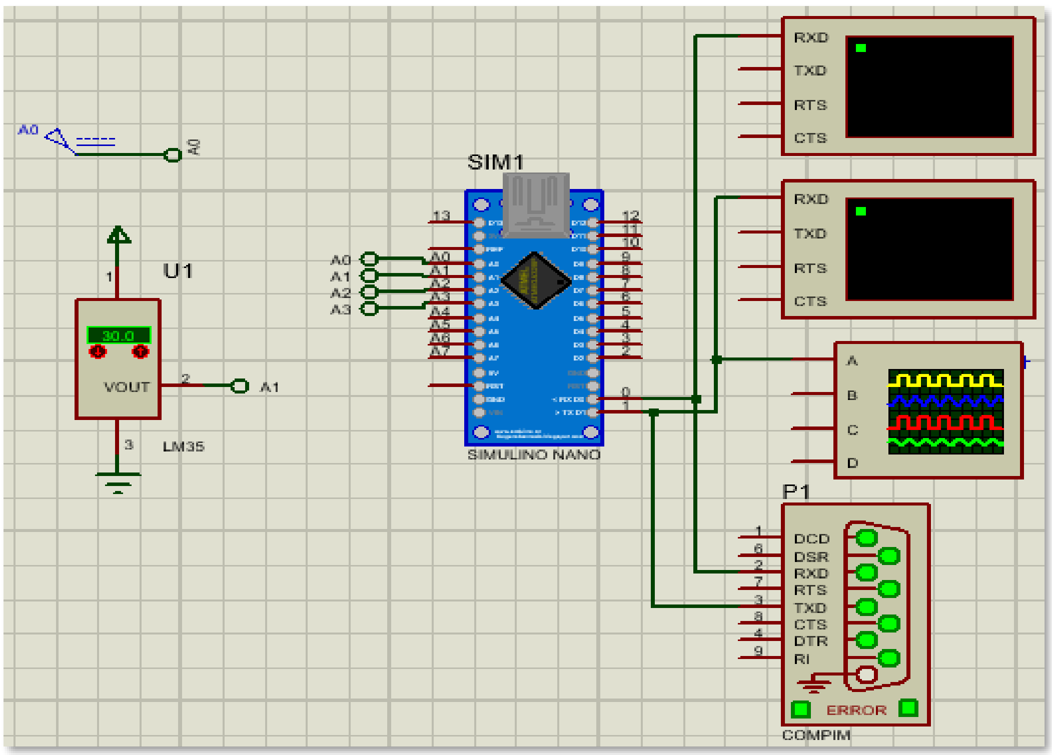


Figure. 38 - simulation schematic

III.6.2 Simulation results

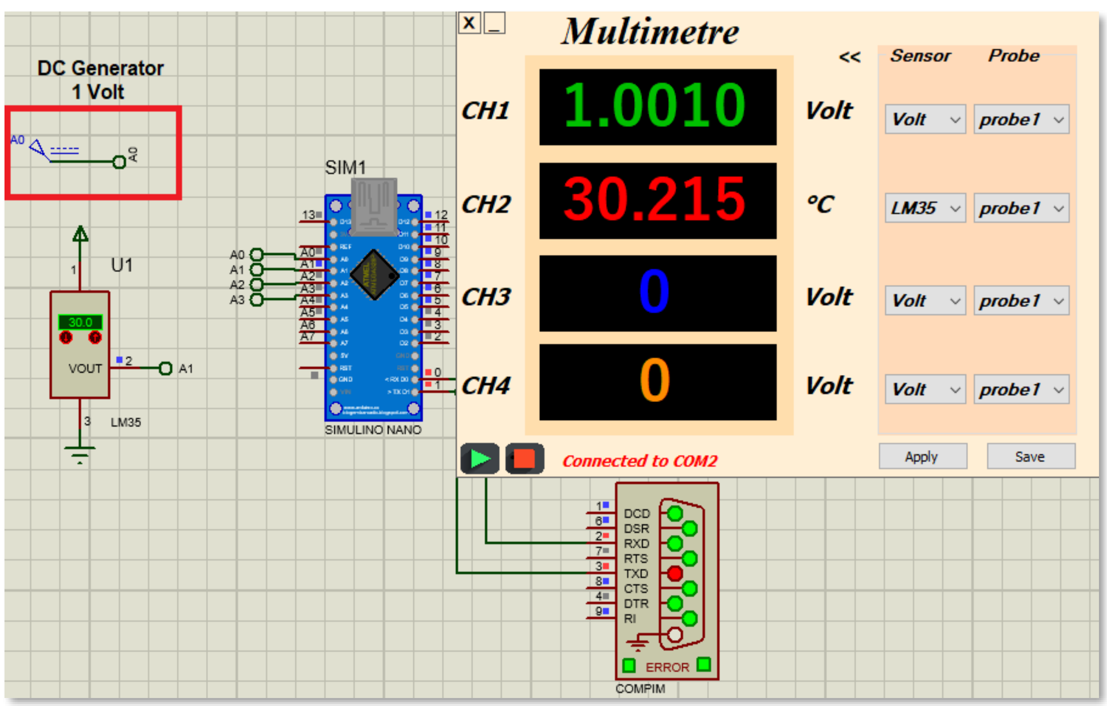


Figure. 39 - Simulation results in digital multimetre

III.7 Virtual serial port driver

Virtual Serial Port Driver program allows creating many virtual COM ports, it supports high speed of data transfer between virtual COM ports. We have used VSPD to connect between Proteus and our acquisition app to make the simulation

- First, we create virtual ports (COM10/COM20)

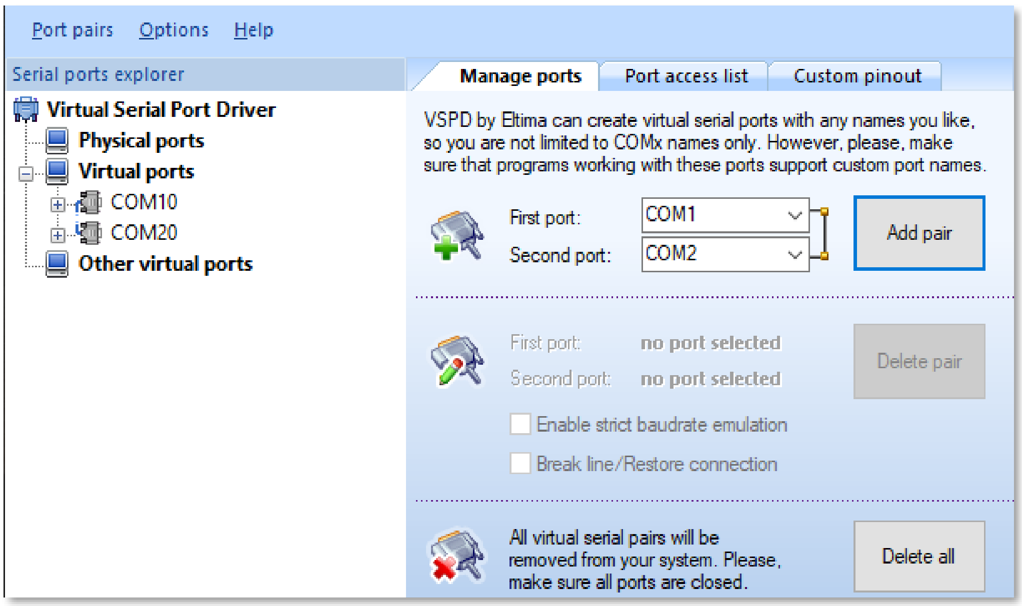


Figure. 40 - Create virtual ports

- Second, we selected the COM10 in Proteus

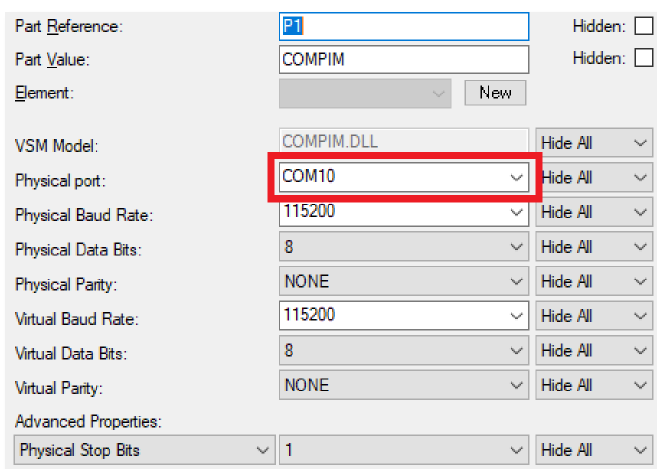


Figure. 41 - selected COM

- Third, we selected and connected to COM20 from our acquisition app

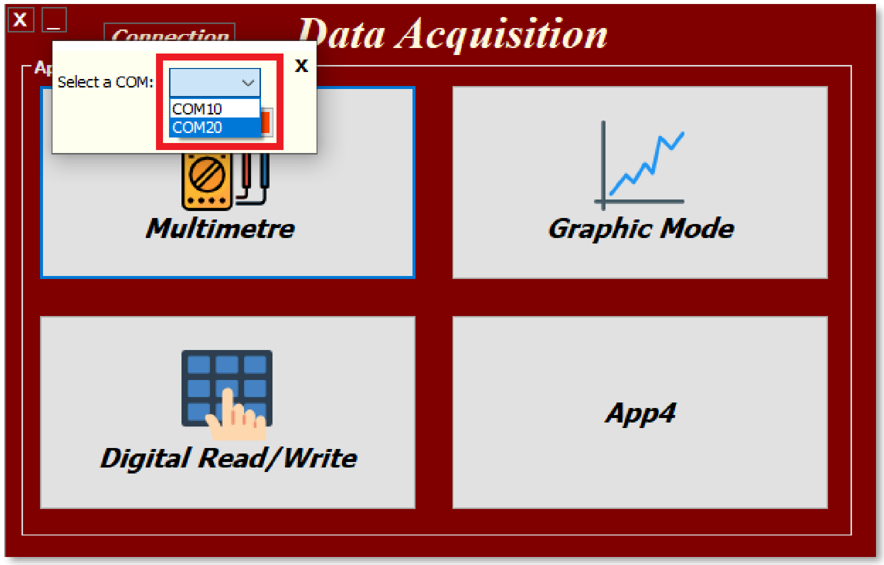


Figure. 42 - select and connect

III.8 Realization

Realization is the assembly of all the components, software and hardware to make the project come to life.

III.8.1 Circuit realization

After the simulation we start to make the schematic in Proteus for the PCB plate also create our custom connect pin <<CON3PIN22>> and <<CON3PIN>> so they can fit in the real components.

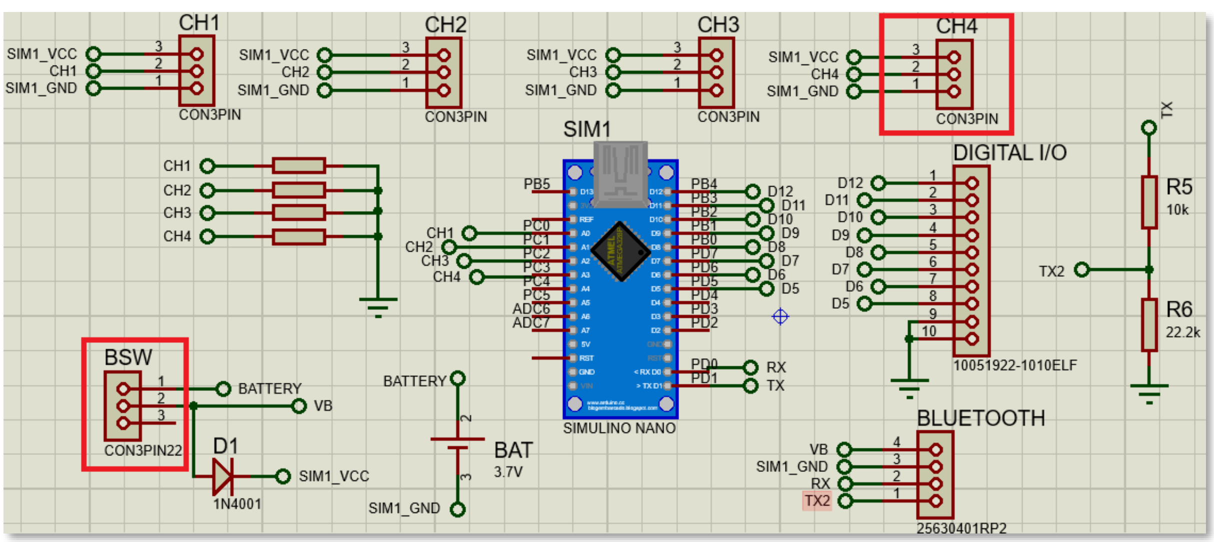


Figure. 43 - Realization final schematic

Now we move to the PCB layout when we can manage the wires and place every component, we can see and check the appearance of our acquisition card before we make it, we have relied on Autodesk Fusion 360 program to create the 3D shape of the analog inputs, this picture bellow describes the board in 3D visualizer:

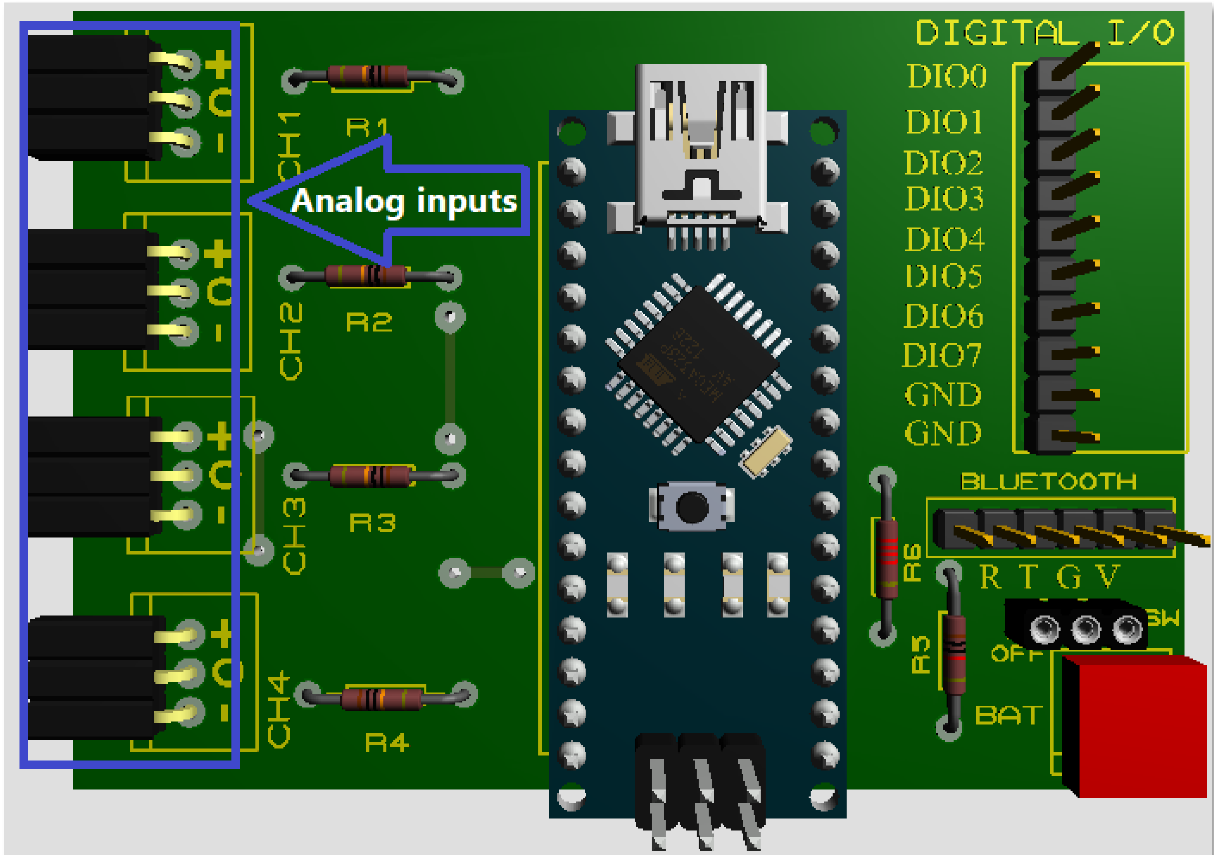
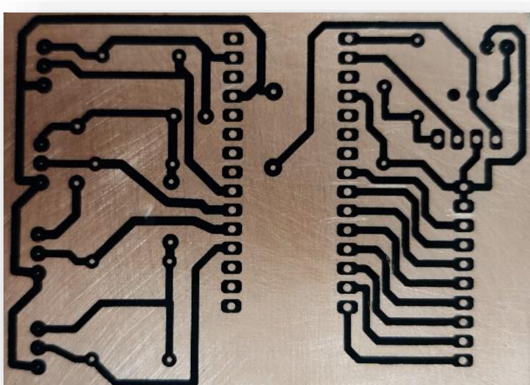


Figure. 44 - acquisition Card in 3D visualizer

❖ In this case we can start realise the PCB plate, so we got these results:



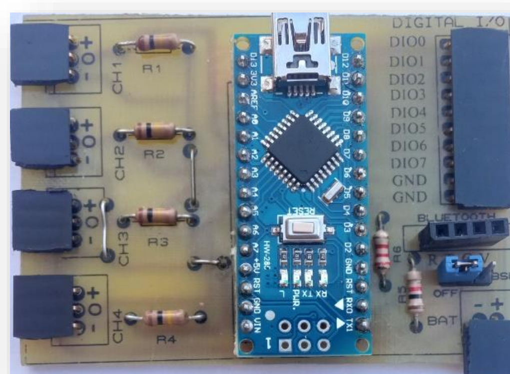


Figure. 45 - PCB wires **Figure. 46 – Final acquisition card PCB**

❖ We assembled the following different components to make the final result:

- Arduino nano (ATMEGA328P) x (1).
- Bluetooth HC-06 x (1).

- Battery (3.7-volt) x (1).
- Cable USB (5-Volts) x (1).
- PCB.

III.8.2 Data acquisition results

Finally, we have tested our system and we have obtained good results as showed below:

- **Multimetre test:** we have put LM35 sensor in CH1 to measure room temperature, and used probe 1 in CH2 to measure voltage of a battery.

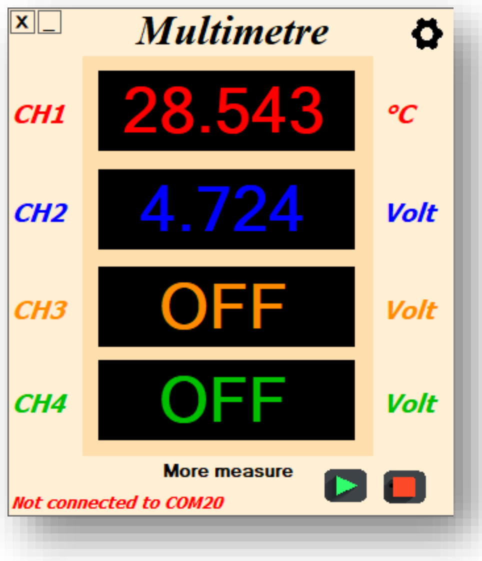


Figure. 47 - multimetre test result.

- **Graphic mode test:** we inserted LM35 in CH1, the graph shows room temperature in the beginning, after about 3 seconds we touched the sensor and the graph shows the increasing of the temperature until it stabilizes at hand temperature.

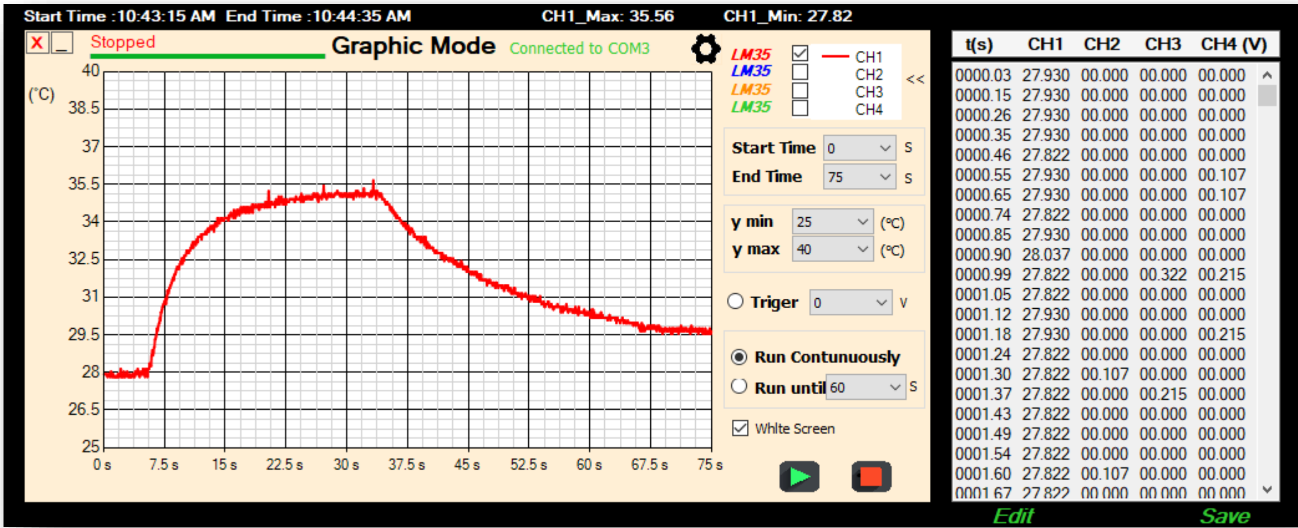


Figure. 48 - Graphic mode result.

III.9 Conclusion

In this chapter, we have explained the different stages of realization of our acquisition card and simulate it using proteus software. The latter allowed us to make the PCB card, which is an acquisition and control card built around a visual basic application.

Conclusion

It is possible to build a low-cost Data acquisition system based on Arduino platform, using different devices that are available in the market at affordable prices. The inclusion of an analog multiplexer in ATmega328p gives the system the ability to obtain 8 analog inputs such as temperature and humidity...etc.

The use of Arduino platform makes our system very versatile, simple and reliable, since it can be reprogrammed easily for many different variables and can be implemented quickly. Data loss in network power interruptions is avoided by adding a 3.7-volt rechargeable Li-ion battery as a secondary power supply.

This card can be used in research works (analyses) or in any situation need to monitor temperature, voltage and current (we can add any analog sensor). Using our windows application, we can display the acquired data and save it, plot live graphs of the various acquired data and set digital pins to Input/output and control them individually.

Our system worked perfectly and in real time precision. The future scope of this work is to include more sensors and negative voltage in the system. Another goal is to connect our system to the internet or GSM module to get accesses to the acquired data remotely and analyse it, and make the digital and analog work at the same time.

Reference

Reference:

[1] Park, J., & Mackay, S. "Introduction: Definition of data acquisition and control" Practical data acquisition for instrumentation and control systems. Oxford: Elsevier, 2003, pp. 1-12.

[2] Di, P, E. M. "Introduction: Fundamentals of Data Acquisition Systems" Data Acquisition Systems: from fundamentals to applied Design. New York, NY: Springer (2013). pp. 1-10.

[3] Di, P. E. M. (2013). *Data acquisition systems: From fundamentals to applied design*. New York, NY: Springer.

[4] National Instruments (2002) Data Acquisition Fundamentals, Application Note 007, National Instruments Corporation.

[5] Taylor J (1986) Computer-based data acquisition system. Instrument Society of America, Research Triangle Park.

[6] Park, J., & Mackay, S. "Analog and digital signals " Practical data acquisition for instrumentation and control systems. Oxford: Elsevier, 2003, pp. 13-35.

[7] Park, J., & Mackay, S. (January 01, 2003). Practical data acquisition for instrumentation and control systems. Oxford: Elsevier.

[8] Li Tan, Jean Jiang. (2013). Chapter 1 - Introduction to Digital Signal Processing, Digital Signal Processing (Second Edition): Academic Press, Pages 1-13, https://doi.org/10.1016/B978-0-12-415893-1.00001-9.

[9] https://www.mccdaq.com/PDFs/specs/Analog-to-Digital.pdf. Retrieved Avril 2022.

[10] https://microcontrollerslab.com/analog-to-digital-adc-converter-working/ Retrieved Mai 2022.

[11] Monk, Simon. (2016) Programming Arduino: Getting Started with Sketches. New York: McGraw-Hill Education.

[12] Morris, Alan & Langari, Reza. (2012). Measurement and Instrumentation: Theory and Application.

[13] Eric Peňa, Mary Grace Legaspi, (DEC 2020)" UART: A Hardware Communication Protocol Understanding Universal Asynchronous Receiver/Transmitter"

[14] https://vanhunteradams.com/Protocols/UART/UART.html , Retrieved Mars 2022

[15] Dapeng Zhu, Analog to Digital Converters,(January 2022) Online Cours

[16] Calcutt, D. M., Cowan, F. J., & Parchizadeh, G. H. (2004). 8051 microcontrollers: An applications-based introduction. Amsterdam: Elsevier, Newnes.

[17] https://store.arduino.cc/products/arduino-nano, Retrieved Mars 4, 2022

[18] https://www.theengineeringprojects.com/2018/06/introduction-to-arduino-nano.html, Retrieved Mars 2022

[19] ATMEL [ATMEL Corporation]. ATmega328/P. retrieved from https://pdf1.alldatasheet.com/datasheet-pdf/view/392243/ATMEL/ATMEGA328.html

[20] WCH [Jiangsu Haoheng Co., Ltd.]. USB to Serial Port Chip CH340 (3A). retrieved from https://pdf1.alldatasheet.com/datasheet-pdf/view/1132602/WCH/CH340.html

[21] Ferri, Giuseppe, Gianluca Barile, and Alfiero Leoni. 2021. "Electronics for Sensors" *Sensors* 21, no. 6: 2226. https://doi.org/10.3390/s21062226

[22] https://www.engineersgarage.com/lm35-description-and-working-principal, Retrieved Mars 2022

[23] Bluetooth Module (HC-06) + Arduino. (2018, August 18). ESE205 Wiki, . Retrieved 05:39, May 20, 2022 from https://classes.engineering.wustl.edu/ese205/core/index.php?title=Bluetooth_Module_(HC06)_%2B_Arduino&oldid=12261

[24] https://www.techtarget.com/whatis/definition/Visual-Basic-VB. Retrieved Mai 2022.

[25] Hamouchi, Hamid. (2015). Conception et réalisation d’une centrale embarquée de la domotique « Smart Home ». 10.13140/RG.2.1.4484.3608.

[26] Tom Almy, Far Inside the Arduino: Nano Every Supplement, Amazon Digital Services LLC - Kdp, 2019

Annex

ANNEX.A. Arduino Nano

ANNEX.A.1. Pin explanation

This part discusses the technical specs most importantly the pinout and functions of every pin in the Arduino Nano board, as you can see this image described each pin of the Arduino Nano:

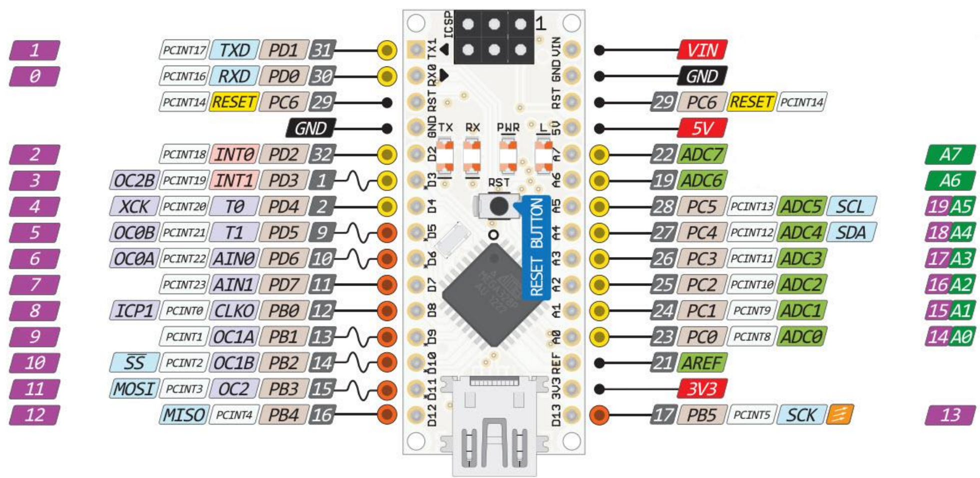


Figure. 49 - Arduino nano pins description.

- ❖ For the pin description of Arduino Nano, let us assume some basic numbering. Let the numbering begin with the TX pin (D1). So, TX is Pin 1 RX is Pin 2, RST is Pin 3, and so on. On the other side, D13 is Pin 16, 3V3 is Pin 17, etc.

Table 3 - Arduino Nano description pins.

Pin Number	Pin Name	Description	Alternative Functions
1	TX / D1	Digital IO Pin 1Serial TX Pin	Generally used as TX
2	RX / D0	Digital IO Pin 0Serial RX Pin	Generally used as RX
3	RST	Reset (Active LOW)	
4	GND	Ground	
5	D2	Digital IO Pin 2	
6	D3	Digital IO Pin 3	Timer (OC2B)
7	D4	Digital IO Pin 4	Timer (T0/XCK)
8	D5	Digital IO Pin 5	Timer (OC0B/T1)
9	D6	Digital IO Pin 6	
10	D7	Digital IO Pin 7	
11	D8	Digital IO Pin 8	Timer (CLK0/ICP1)
12	D9	Digital IO Pin 9	Timer (OC1A)
13	D10	Digital IO Pin 10	Timer (OC1B)

14	D11	Digital IO Pin 11	SPI (MOSI) Timer (OC2A)
15	D12	Digital IO Pin 12	SPI (MISO)
16	D13	Digital IO Pin 12	SPI (SCK)
17	3V3	Power	
18	AREF	Analog Reference	
19	A0	Analog Input 0	
20	A1	Analog Input 1	
21	A2	Analog Input 2	
22	A3	Analog Input 3	
23	A4	Analog Input 4	I2C (SDA)
24	A5	Analog Input 5	I2C (SCL)
25	A6	Analog Input 6	
26	A7	Analog Input 7	
27	5V	+5V Output from regulator or +5V regulated Input	
28	RST	Reset (Active LOW)	
29	GND	Ground	
30	VIN	Unregulated Supply	

- The following table describes the pins of the ICSP Connector.

Table 4 - ICSP Connector pins.

Column1	Column2
MISO	Master In Slave Out (Input or Output)
5V	Supply
SCK	Clock (from Master to Slave)
MOSI	Master Out Slave In (Input or Output)
RESET	Reset (Active LOW)
GND	Ground

In the Arduino board, we will see 30 pins so of these pins there are 22 pins associated with input and output. In those 14 pins (D0 to D13) are true digital IO pins, which can be configured as per your application using pinMode(), digitalWrite(), and digitalRead() functions.

All these Digital IO pins are capable of sourcing or sinking 40mA of current. An additional feature of the Digital IO pins is the availability of an internal pull-up resistor (which is not connected by default). The value of the internal pull-up resistor will be in the range of 20KΩ to 50KΩ.

There are also 8 Analog Input Pins (A0 to A7). This is a couple more than Arduino UNO (which only has 6). All the analog input pins provide a 10-bit resolution ADC feature.

An important point about Analog Input pins is that they can be configured as digital IO pins if required (all analog pins except A6 and A7 can be configured as digital IO).

Digital IO pins 3, 5, 6, 9, 10, and 11 are capable of producing 8-bit PWM Signals.

This was a brief overview of the Arduino Nano board layout, specialized specifications, important features, and most importantly pinout information. [18]

ANNEX.A.2. communication

The Arduino Nano has several facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328P provide UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An FTDI FT232RL on the board channels this serial communication over USB and the FTDI drivers (included with the Arduino software) provide a virtual com port to software on the computer. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the FTDI chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A SoftwareSerial library allows for serial communication on any of the Nano's digital pins. The ATmega168 and ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify the use of the I2C bus; see the documentation for details. To use the SPI communication, please see the ATmega168 or ATmega328 datasheet.

ANNEX.A.3. Configuration

The Arduino Nano can be programmed with Arduino software (IDE), The ATmega328P on the Arduino Nano comes pre burned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files). You can also bypass the bootloader and program the microcontroller through the ICSP (in-circuit Serial Programming) header; see these instructions for details. [26]

ANNEX.B. Probe

ANNEX.B.1. voltage divider

We used the voltage divider to reduce the volta

$$ADC\ V_{in} = \frac{R2}{R1 + R2} \times Probe\ V_{in}$$

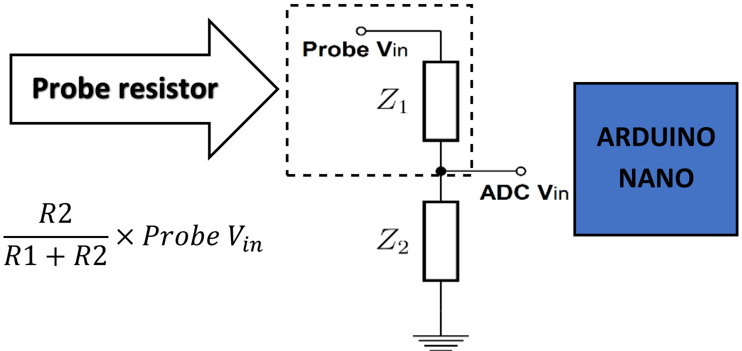


Figure. 50 – Probe divider voltage

Abstract

Our project is a realization of a low-cost data acquisition system based on Arduino Nano, it can measure temperature, voltage and current. Using Visual basic we designed a windows application that displays the data acquired by our system with a possibility to save them as text file, our card can be connected to the computer by USB cable or wirelessly by Bluetooth.

Keywords: Data acquisition system (DAQ), Arduino Nano, LM35, Visual basic

ملخص

مشروعنا عبارة عن انشاء نظام استحواذ بيانات منخفض التكلفة و الذى يعتمد على اردوينو نانو ، يمكنه قياس درجة الحرارة والجهد والتيار. باستخدام الفيجوال بيسك صممنا تطبيقً ويندوز لعرض البيانات التي حصل عليها نظامنا حيث يمكن حفظها على شكل ملف نصى ، كما انه يمكن لبطاقتنا ان تتصل بالحاسوب عن طريق الكابل يو اس بى أو لاسلكيا عبر البلوتوث.

الكلمات المفتاحية: اردوينو نانو ، نظام الحصول على البيانات ، فيجوال بيسك ، مستشعر حرارة

Résumé

Notre projet est un système d'acquisition de données à faible coût basé sur Arduino Nano, il peut mesurer la température, la tension et le courant. En utilisant Visual basic, nous avons conçu une application Windows qui affiche les données acquises par notre système et les enregistre comme un fichier texte, notre carte se connecte à l’ordinateur à travers un câble USB ou un module Bluetooth.

Mots clés : Système d'acquisition de données, Arduino Nano, LM35, Visual basic