



Title: Microcontroller lab with remote connectivity and control of virtual instruments

Authors: ISLAS-CERÓN, Alejandro, MARTÍNEZ-HERNÁNDEZ, Haydee Patricia, CORTES-MALDONADO, Raúl and MORALES-CAPORAL, Roberto

Editorial label ECORFAN: 607-8695
BCIERMMI Control Number: 2022-01
BCIERMMI Classification (2022): 261022-0001

Pages: 23
RNA: 03-2010-032610115700-14

ECORFAN-México, S.C.
143 – 50 Itzopan Street
La Florida, Ecatepec Municipality
Mexico State, 55120 Zipcode
Phone: +52 1 55 6159 2296
Skype: ecorfan-mexico.s.c.
E-mail: contacto@ecorfan.org
Facebook: ECORFAN-México S. C.
Twitter: @EcorfanC

www.ecorfan.org

Holdings		
Mexico	Colombia	Guatemala
Bolivia	Cameroon	Democratic
Spain	El Salvador	Republic
Ecuador	Taiwan	of Congo
Peru	Paraguay	Nicaragua

INDEX

1

Abstract

2

Introduction

3

**Virtual
Instrumentation**

4

Printed Circuit

5

Communication

6

Remote Control

7

**Results &
Discussion**

8

Conclusion

9

References

10



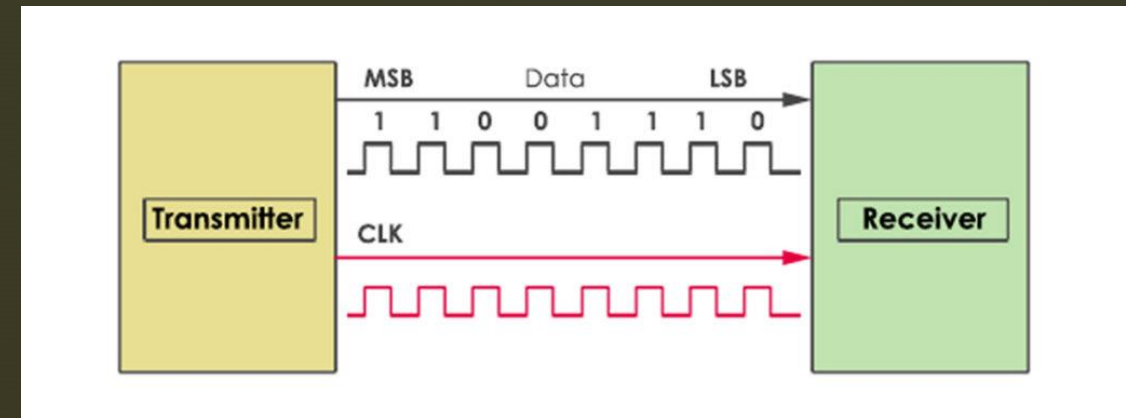
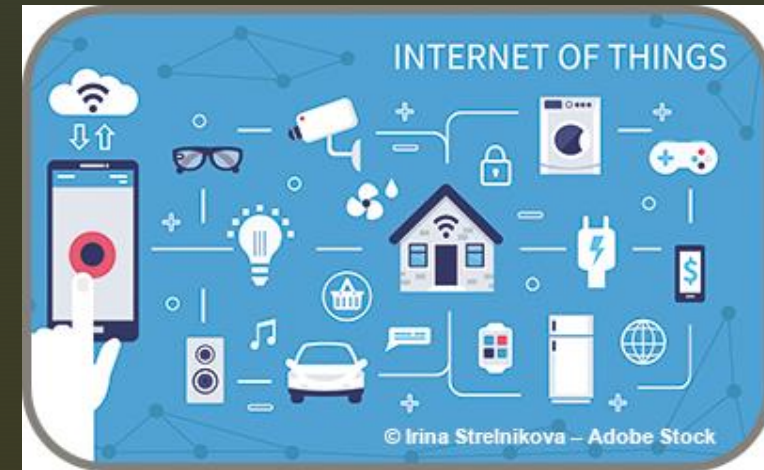
HOME

► ABSTRACT

- **This article shows a learning alternative for undergraduate engineering students, who due to force majeure do not attend their practices or projects in a face-to-face manner in the subject of microcontrollers. This guarantees a hybrid teaching-learning process, where students can interact remotely with control boards located in laboratories and workshops. To achieve this task, the syllabus of this subject was consulted, a series of practices were selected and with them the use of graphical interfaces for visualization and control of virtual instruments was developed, which through a USB-Serial RS232 adapter communicates with the transmission (Tx) and reception of data (Rx) ports of the PIC18f4550 microcontroller, thus allowing control actions on output actuators. Finally, the remote connection is made, through the NodeMCU development board based on the ESP8266 chip and a mobile application developed in Blynk, allowing the student to learn in a more didactic way. The implementation of this type of alternative is intended to ensure quality education at the higher level, in situations where students do not have the availability to attend the facilities of their academic institution.**

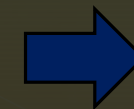
INTRODUCTION

- Teaching-Learning Process
- Microcontrollers
- Virtual Instrumentation
- Serial Protocols
- Internet of things
- Mobile Apps
- Remote Laboratories

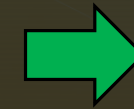


Virtual Instrumentation

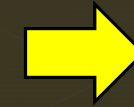
- A virtual instrumentation system is computer software that a user would employ to develop a computerized test and measurement system, for controlling from a computer desktop an external measurement hardware device, and for displaying test or measurement data collected by the external device on instrument-like panels on a computer screen.



Frontal Panel



Programming Environment



On-Off DC motor



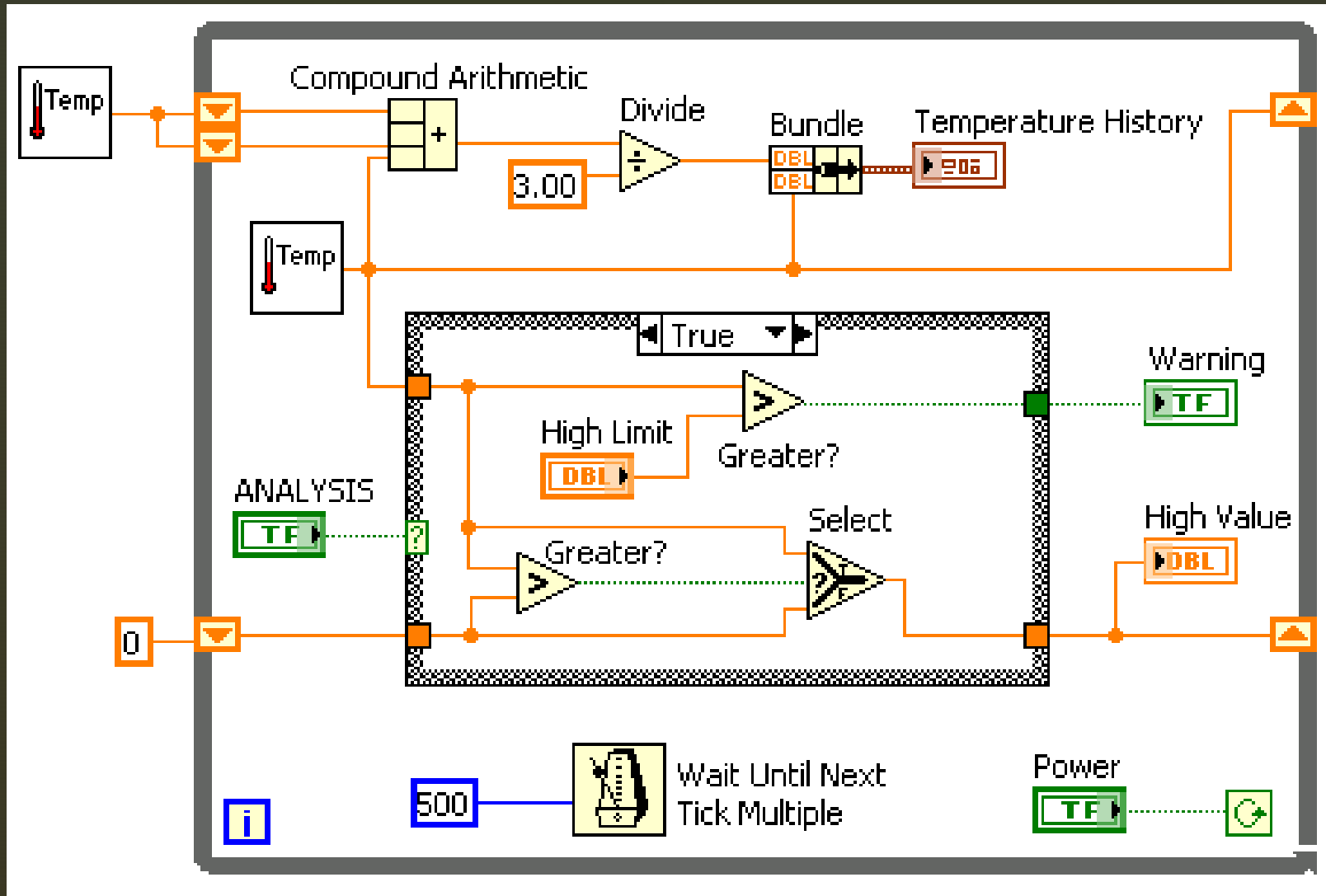
Frontal Panel

The screenshot displays a LabVIEW Frontal Panel with the following elements:

- Top Left:** A green square indicator with the text "<- click button".
- Host IP Address:** A text box containing "http://10.119.83.180:6341" with the instruction "Host IP adress ->Copy this adress in your browser !".
- Bytes Sent:** A numeric display showing "5920" and a label "# bytes send".
- Gauge:** A circular gauge with a red needle pointing to approximately 10.
- Plot 0:** A line graph showing "Amplitude" on the y-axis (0 to 1) and "Time" on the x-axis (8524 to 8624). The plot shows a fluctuating signal.
- Status Code:** A numeric display showing "0" with a green checkmark icon.
- Source:** A text box for entering a source address.
- Control Panel:** A vertical stack of controls: "Gauge", "# bytes send", and "Waveform Chart".
- Beer Level:** A vertical bar indicator showing a yellow liquid level, labeled "Beer" and "Slide Beer - ^".
- Digital Data:** A table with the following content:

	0
10	A1
20	B 0 0
- Mouse on/off:** Two toggle switches.
- Thermometer:** A vertical thermometer indicator showing a red liquid level, labeled "Thermometer" and "Slide thermometer".
- Horizontal/Vertical/Scrolling:** A control panel with a dropdown menu set to "Horizontal", a numeric display showing "751", and a numeric display showing "0".
- Buttons:** Four buttons labeled "button 1", "button 2", "button 3", and "button 4".
- STOP Button:** A large button with the text "STOP" in red.
- Note:** A text box stating: "Note: Do not use control and indicator labels ! The LabVIEW get picture method can not handle these labels !".

▶ Programming Environment



On-Off DC Motor

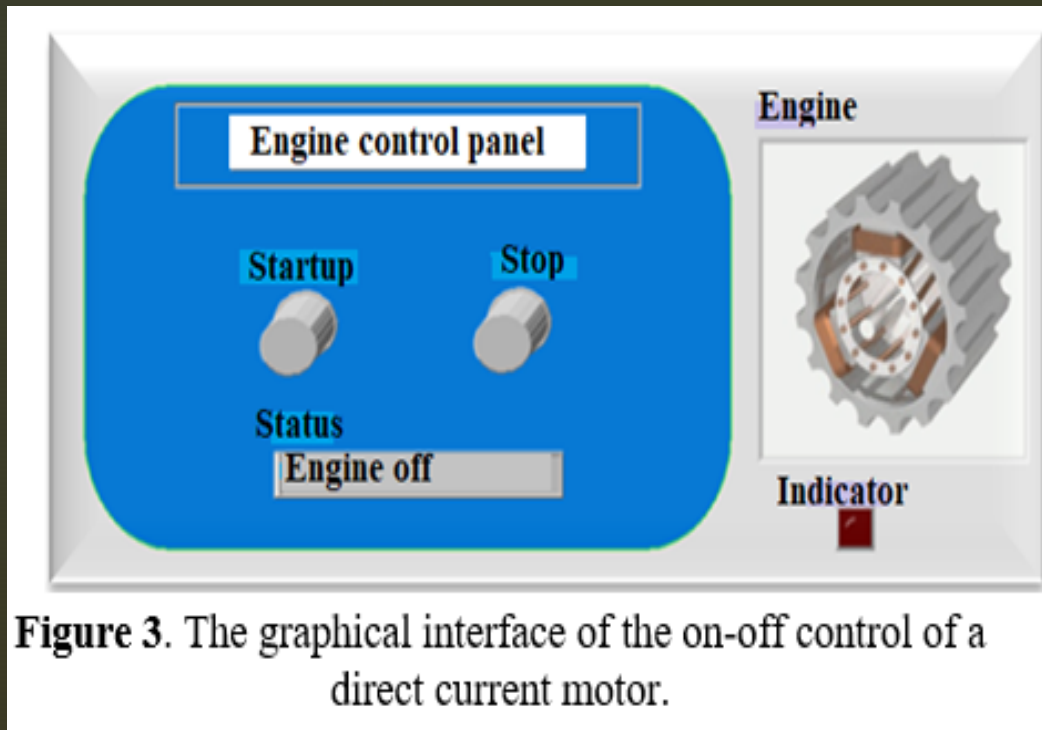


Figure 3. The graphical interface of the on-off control of a direct current motor.

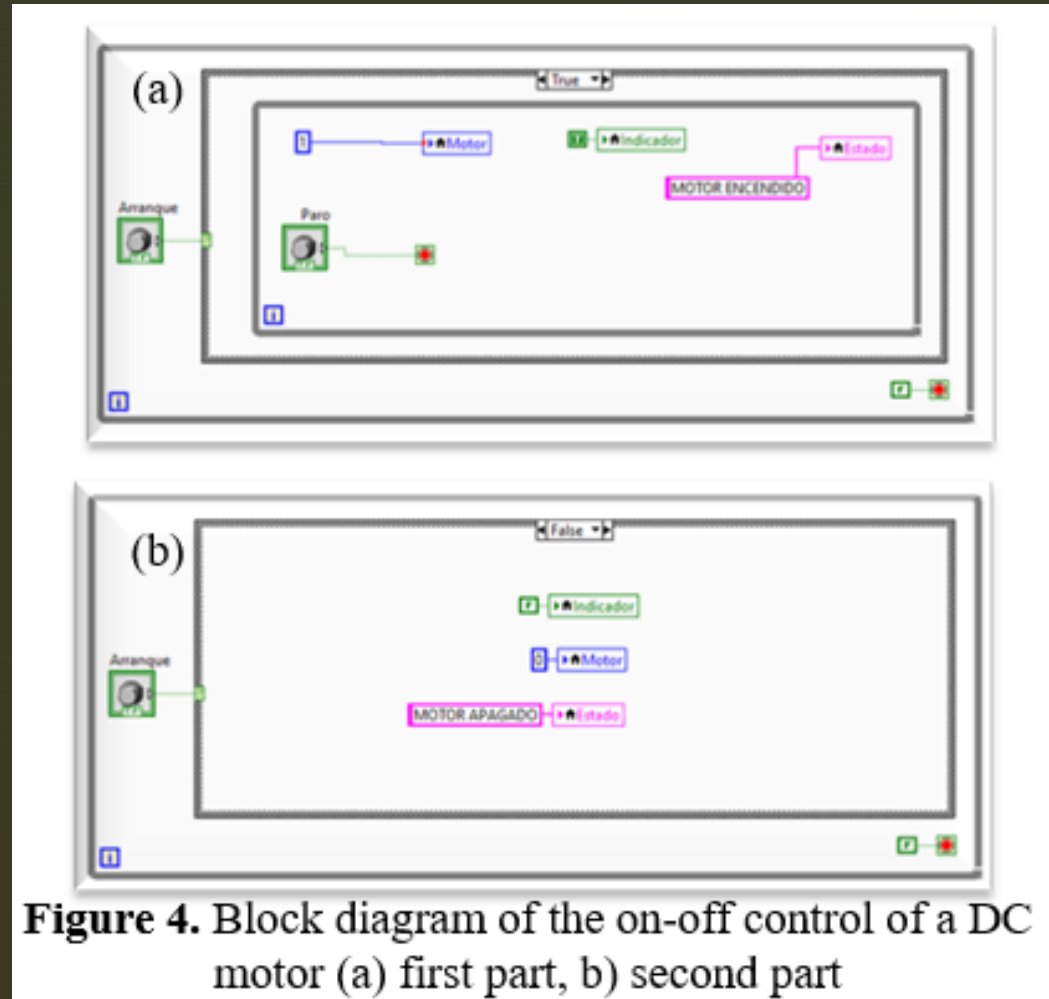
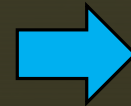


Figure 4. Block diagram of the on-off control of a DC motor (a) first part, b) second part

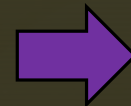
Design & Printed Circuit

The design of the control board allows us to control various physical actuators.

In this case, the PCB Wizard software was used for this purpose. The control board has 2 microcontrollers (PIC18F4550, and the PIC16F628a). We can also observe that the board already has integrated pins for the connection of its programmer, in this case, the PICKit 2.



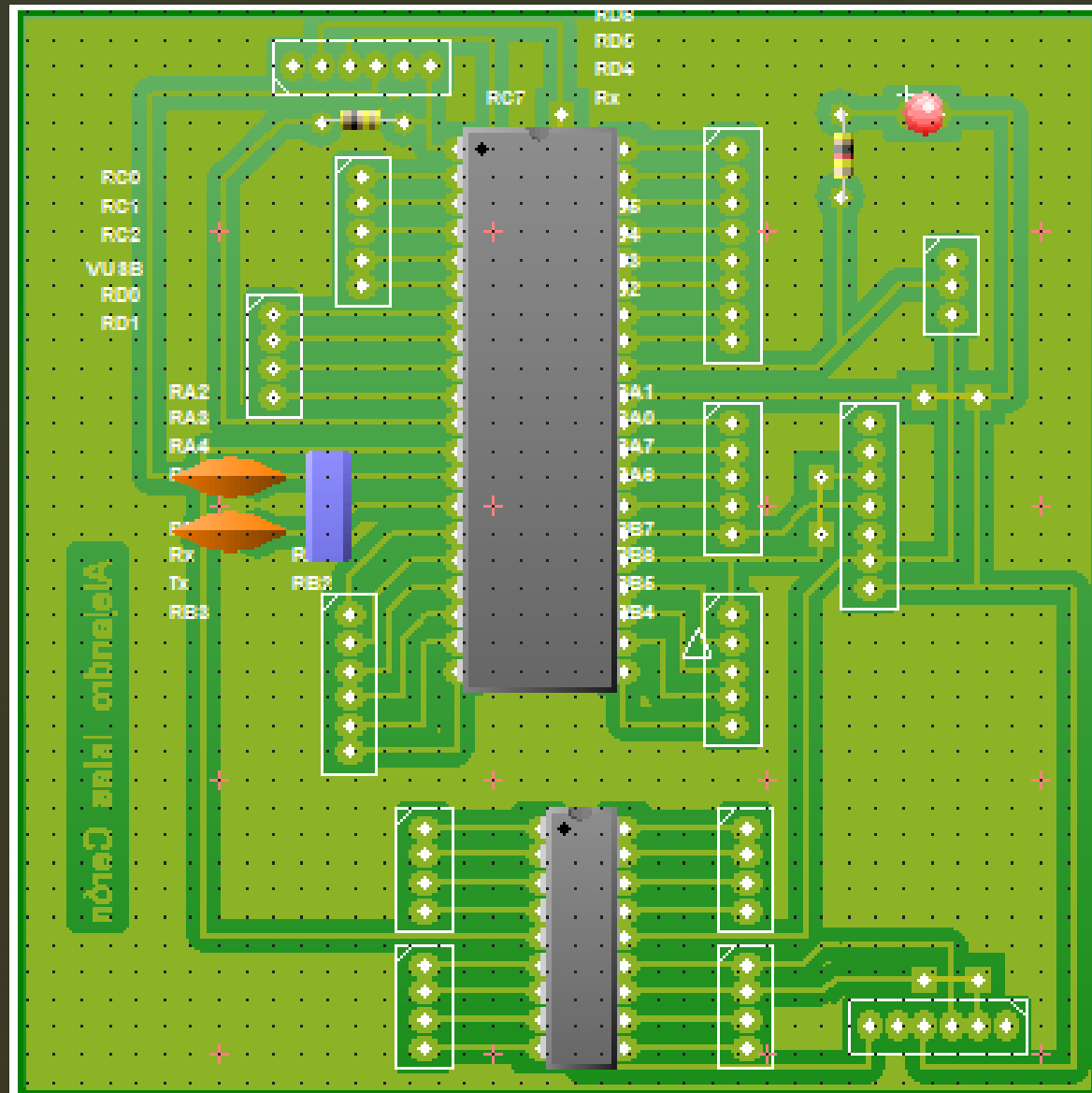
PCB (Real World)



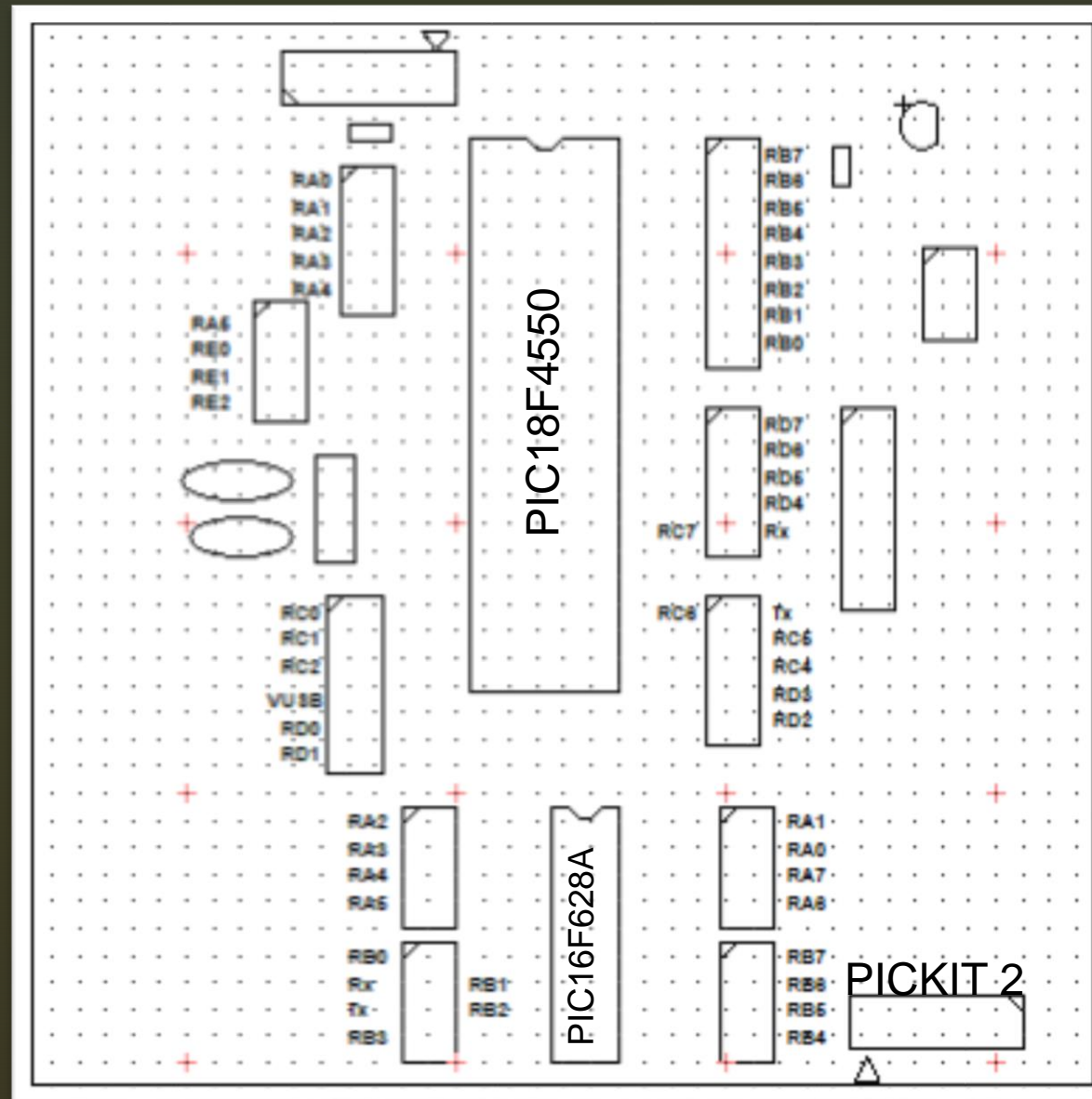
PCB (Silk Screen Artwork)



PCB (Real World)

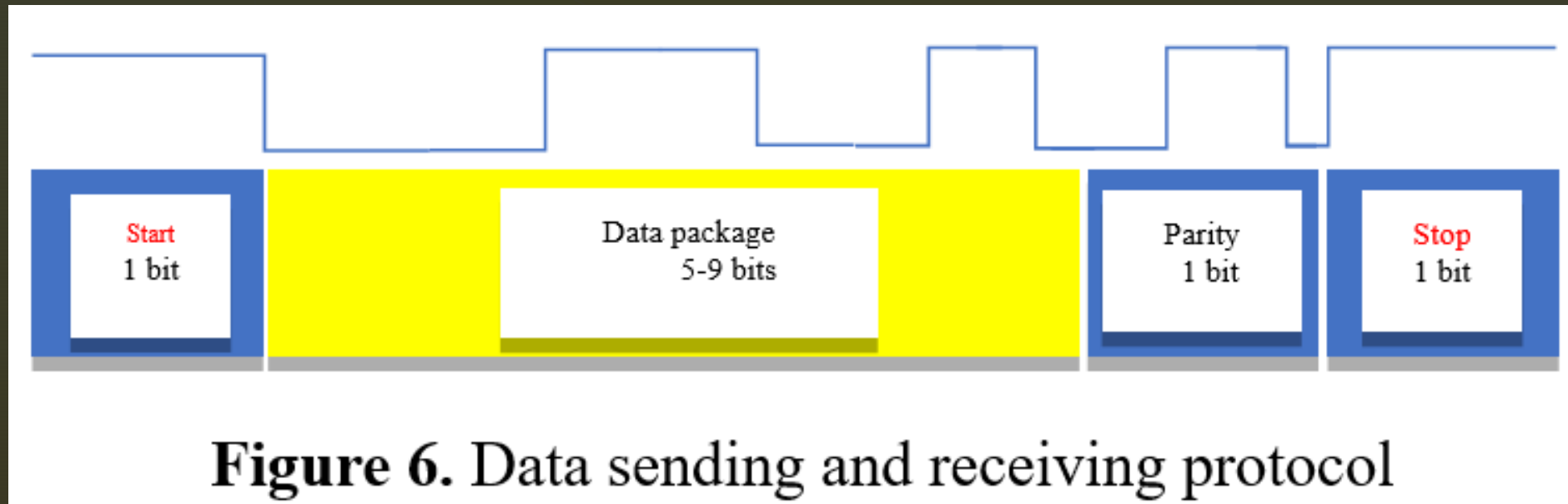


PCB(Silk Screen Artwork)



Communication Protocol

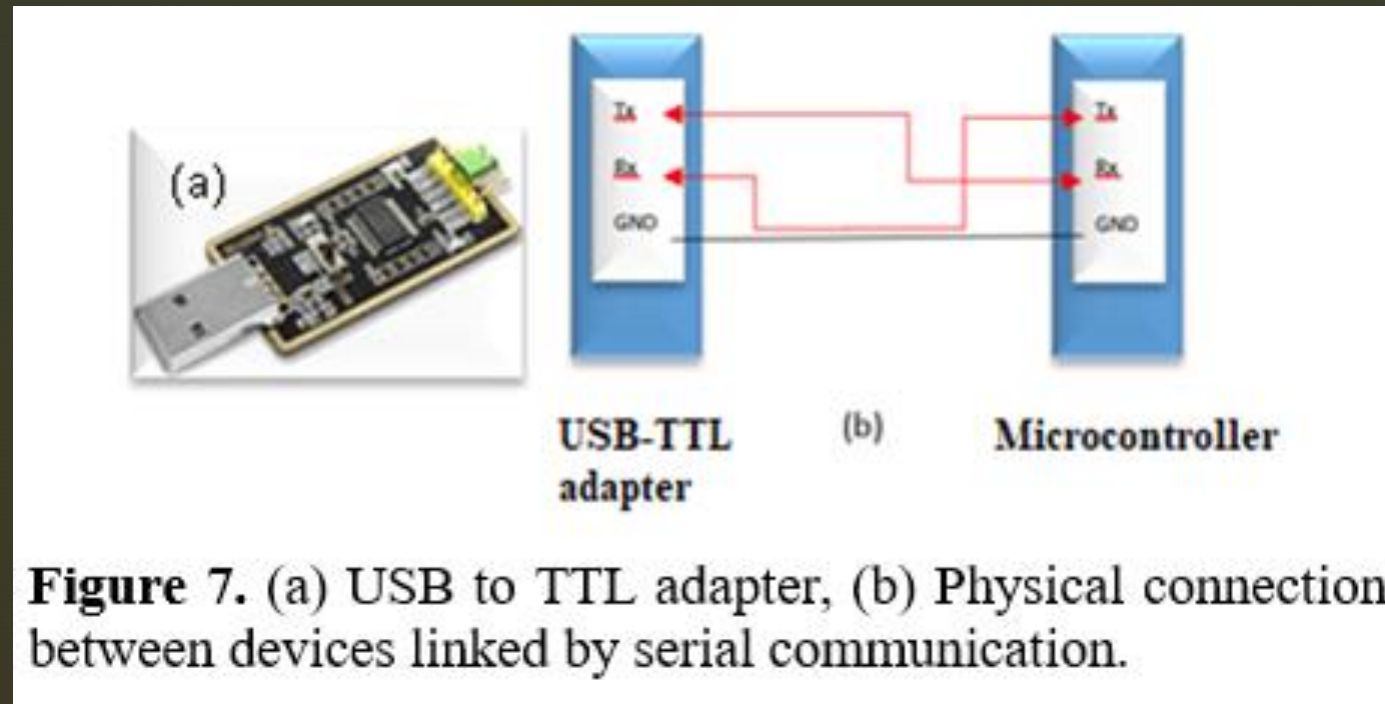
- The communication between virtual instruments and the control board described before is done through the transmission and reception of serial data.



Physical Connection

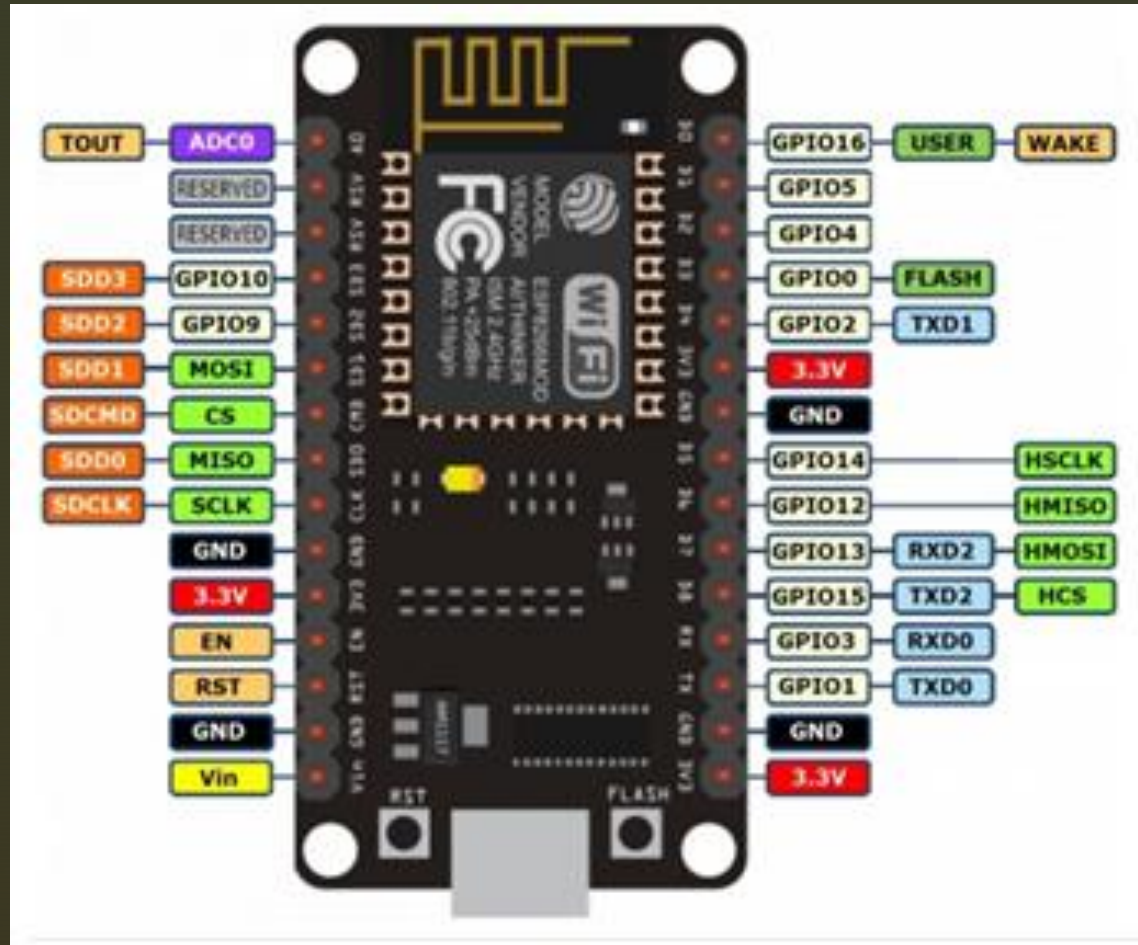
► Physical Connection

The physical connection is achieved through a USB – TTL device as shown in Figure 7.



Remote Control

- For remote control of the electronic devices, the NodeMCU development board based on the open source ESP8266 SoC and a mobile application developed in Blynk were used.



Results & Discussion

- 1.- At the moment of connecting our control card employing the USB-TTL adapter, a COM was automatically generated, which allowed the communication with Labview.

 COM Generated

- 2.- By integrating the LabView graphical interfaces with the serial communication protocol, a correct data flow could be observed.

 Data Flow

- 3.- From the mobile application created in Blynk, it was possible to send and received voltage signals in the Node MCU board successfully.

 Code and App

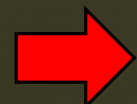
- 4.- Once the mobile application was linked to the NodeMCU board, the data was sent to the PIC18F4550 (transformation of voltage logic levels was considered).

 DC-DC converter

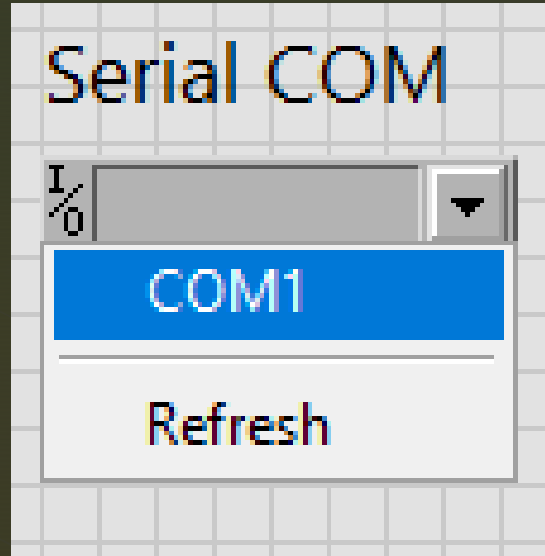
- 5.- The programs in Necto Studio and the Arduino IDE were compiled without errors and allowed reading and writing of the data sent to Labview and received from the mobile application.

 Necto Studio Code

- 6.- The electrical connection diagram was made in Fritzing.

 Electrical Connection Diagram

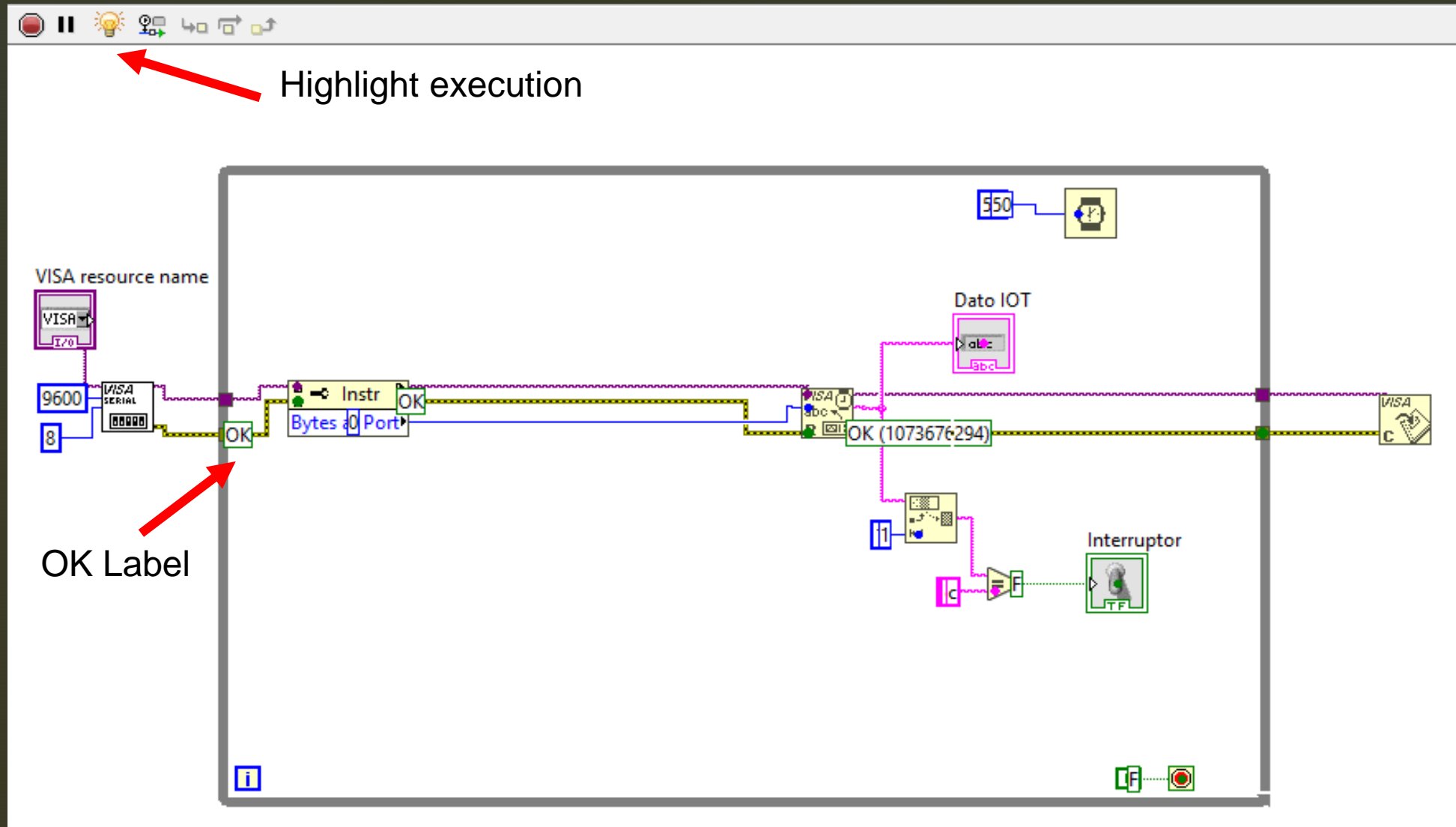
► COM Generated Successfully



- On some devices, it was observed that the control board did not show any response, but this problem was solved by installing the **CP2102 driver**.
- It is also important to add that to avoid any error in Labview, the NI-VISA and **NI Serial libraries** must be downloaded, otherwise, the proper connection will not be achieved.



Successful data flow from beginning to end



BLYNK & NODE MCU Board



Figure 12. Control interface developed in Blynk.

```
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "l2bR0pMDljdmiQA17xFord9bcA7cq42m";

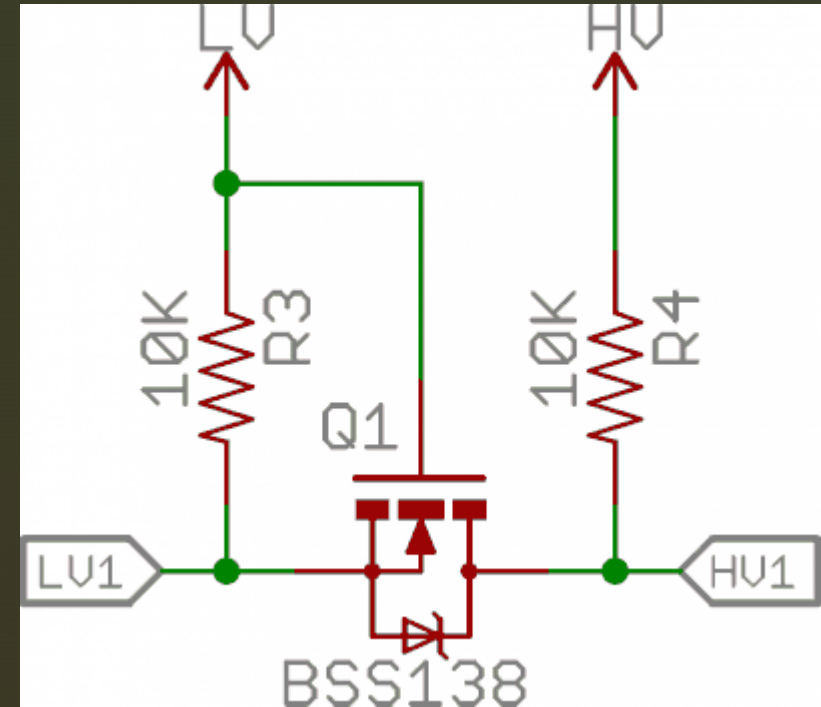
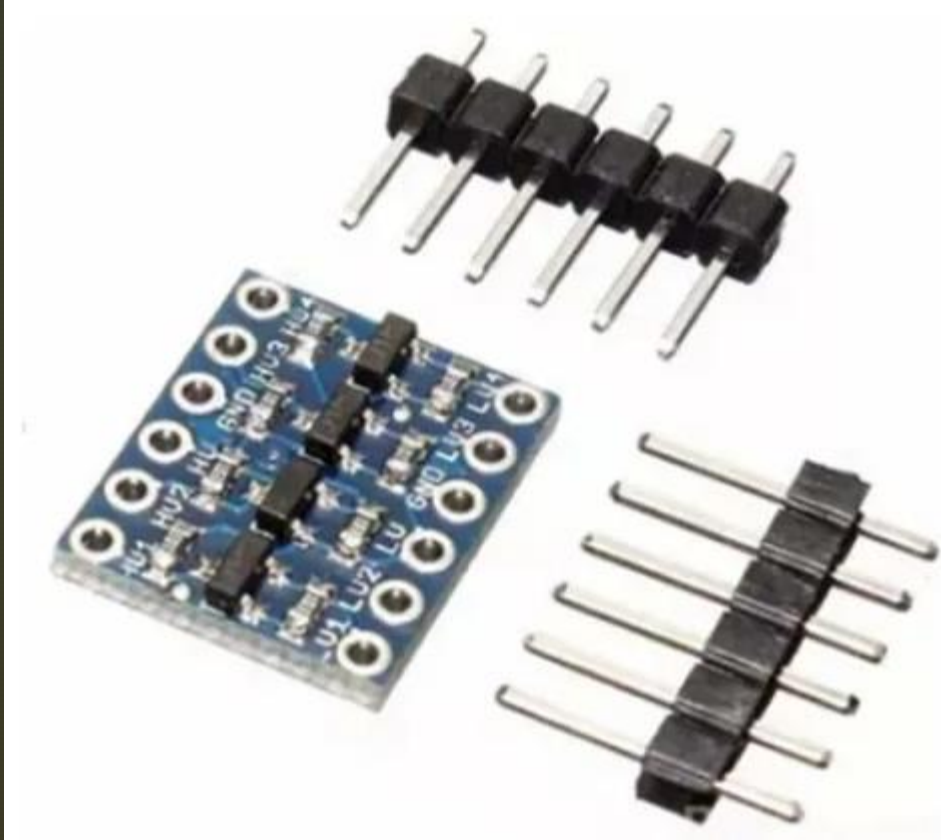
// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "INFINITUM639D_2.4";
char pass[] = "k6SR59eD3s";

void setup()
{
  // Debug console
  Serial.begin(9600);
  pinMode(16, OUTPUT);
  Blynk.begin(auth, ssid, pass);
}

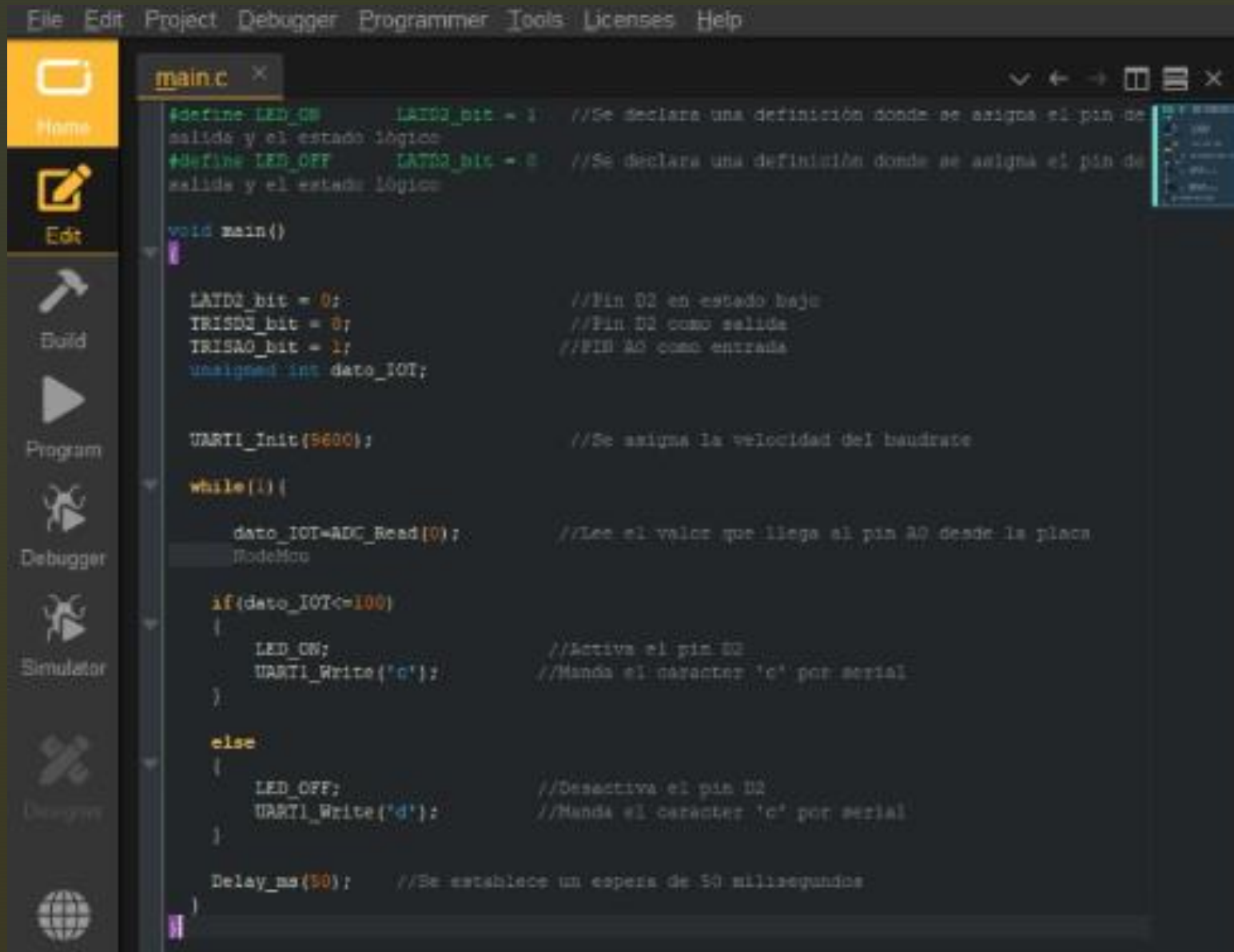
void loop()
{
  Blynk.run();
}
```

Figure 13. Shows the resulting code, which allowed the control of the widget labeled "BUTTON" created in Blynk.

Bi-directional 5v To 3.3v Logic Level Converter



Necto Studio Code



The screenshot displays the Necto Studio Code IDE interface. The left sidebar contains navigation icons for Home, Edit, Build, Program, Debugger, Simulator, and Designer. The main editor window shows a C file named `main.c` with the following code:

```
#define LED_ON LATDQ_bit = 1 //Se declara una definición donde se asigna el pin de salida y el estado lógico
#define LED_OFF LATDQ_bit = 0 //Se declara una definición donde se asigna el pin de salida y el estado lógico

void main()
{
    LATDQ_bit = 0; //Pin D2 en estado bajo
    TRISDQ_bit = 0; //Pin D2 como salida
    TRISAQ_bit = 1; //PIN A0 como entrada
    unsigned int dato_IOT;

    UART1_Init(9600); //Se asigna la velocidad del baudrate

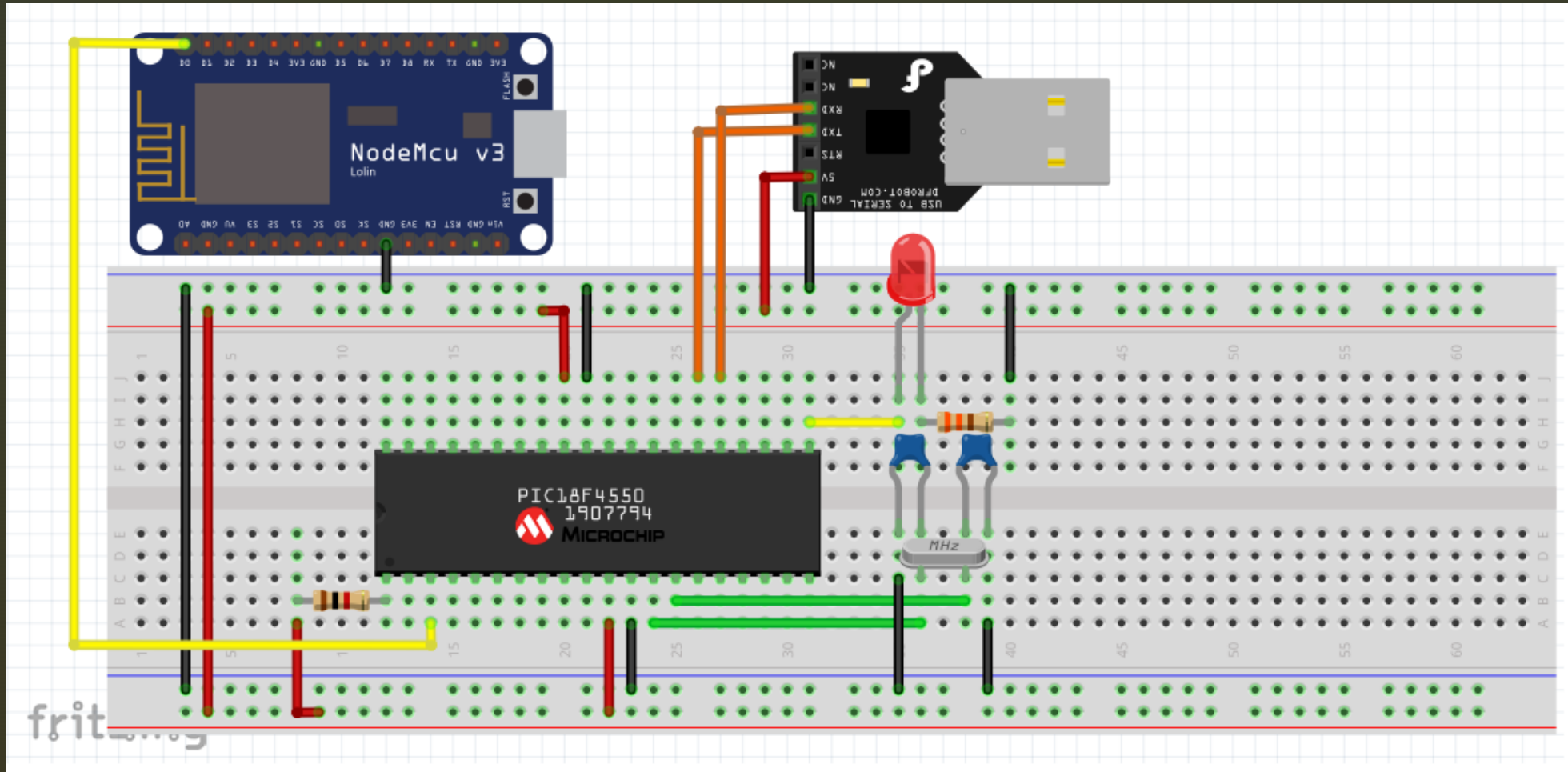
    while(1){
        dato_IOT=ADC_Read(0); //Lee el valor que llega al pin A0 desde la placa
        //ModoHex

        if(dato_IOT<=100)
        {
            LED_ON; //Activa el pin D2
            UART1_Write('c'); //Manda el carácter 'c' por serial
        }

        else
        {
            LED_OFF; //Desactiva el pin D2
            UART1_Write('d'); //Manda el carácter 'c' por serial
        }

        Delay_ms(50); //Se establece un espera de 50 milisegundos
    }
}
```

Electrical Connection Diagram in Fritzing



Conclusion

The implementation of applications focused on the Internet of Things (IoT), allows students to perform practices and projects remotely, strengthening the teaching-learning process.



References

- Genevieve Bakam, E.I Edoun, Anup Pradhan, Nita Sukdeo. (February 2022). A framework for good health governance during the Covid-19 pandemic response and recovery in South Africa. *Technium Social Sciences Journal*, 28, 1-16.
- Tito-Manuel Piamba-Mamian, Leidy Estefanía Zambrano, Luis Alexander Montaña-Rúales, Francisco Arturo Rojas-Gonzales. (2021: 3-19). Implementación de un sistema de monitoreo IoT aplicado a una piscicultura de trucha. *Informador Técnico* , 85, 3-17.
- Rodríguez González, D. 2013. Arquitectura y Gestión de la IoT. *Telemática*. 12, 3 (Nov. 2013), 49–60.
- Abu Bakar Siddique, Sanaullah Memon, sanaullah. (2019). IoT Web Based Internet of Things for Home Automation. *Sukkur IBA University Journal of Emerging Technologies*, 2, 13-19.
- A. Sanjay, Meenu Vijarania, Vivek Jaglan. (06 August 2020). Security Surveillance and Home Automation System using IoT. *EAI Endorsed Transactions on Smart Cities*, 5, 1-6.



ECORFAN®

© ECORFAN-Mexico, S.C.

No part of this document covered by the Federal Copyright Law may be reproduced, transmitted or used in any form or medium, whether graphic, electronic or mechanical, including but not limited to the following: Citations in articles and comments Bibliographical, compilation of radio or electronic journalistic data. For the effects of articles 13, 162,163 fraction I, 164 fraction I, 168, 169,209 fraction III and other relative of the Federal Law of Copyright. Violations: Be forced to prosecute under Mexican copyright law. The use of general descriptive names, registered names, trademarks, in this publication do not imply, uniformly in the absence of a specific statement, that such names are exempt from the relevant protector in laws and regulations of Mexico and therefore free for General use of the international scientific community. BCIERMMI is part of the media of ECORFAN-Mexico, S.C., E: 94-443.F: 008- (www.ecorfan.org/booklets)