Design and implementation of a serial-USB converter

Erik Patrón†, Candelario Pinto†, Raciel Poot†, Erika Llanes †, Lizzie Narváez† and Victor Chi†

†UADY, Unidad Multidisciplinaria Tizimín

Summary
This project presents the design and implementation of a serial-USB converter, useful for transferring data from devices that do not have RS-232 serial ports. The converter was built using the MAX232 chip and the PIC18F4550 microcontroller and MPLAB was used for the logic programming of the converter. In the final part of the project we present the implementation of the prototype already tested on the printed circuit board (PCB), thereby obtaining the serial-USB converter.

Key words:
RS-232 Protocol, USB Protocol, PIC18F4550, MAX232, MPLAB.

1. Introduction

The serial port is a communication interface between computers and devices for digital data transmission; "serial" because the port "serializes" data, i.e. by a byte of data transmitting the 8 bits of a byte one at a time, figure 1. In the past, RS-232 serial port established one of the most basic connections of a computer, having existed since the early computer era; some examples of using these ports in the old models are keyboards, mice, terminals and modems. Currently in most devices, the RS-232 serial port has been replaced by the USB (Universal Serial Bus) due to transmission speeds. Also, computers are now connected to external devices via USB and very often do not even have an RS-232 serial port, some grounds for why that was removed are: to achieve a reduction in cost and because it considers obsolete port.

Fig. 1. DB9 Serial port

There are currently many projects in which the absence of the serial port had to be addressed and a common option is to use another interface, usually the USB interface. Examples include the creation of a synchronized video system that uses a serial-USB converter to connect a video input with a miniTAG (TAG: Trajectory Attitude and Gravimetry) which is a device that works as a GPS time server (Global Positioning System) [1]. Another example of the application of this device is in [2] where mention about the use of Google Earth as GPS universal, and says that the GPS system is used only in the serial output port while the laptop has only used USB ports, used to resolve this conflict, a serial-USB converter.

Other works consulted were several important developments, for example [3] the use of serial-USB converter to connect the hardware (reader EMG signals) with the computer, which was of vital importance in monitoring electromyography signal, it should be noted that electromyography is a method of muscle testing that tracks and interprets the electrical activity that provided the muscle contractions.

This article is structured as follows, the next section describes the problems and technical solutions posed by research, the third section presents the architectural design of the prototype and description of the modules that comprise it and the fourth section describes the electronics and the methodology for the construction of the drive on the tablet prototyping (breadboard) and logic programming microcontrollers finally describes the implementation of prototype printed circuit board and lists the conclusions.

2. The problem and technology

In addition to individual cases mentioned in the previous section, we mention some other devices and applications using exclusively the RS-232 serial interface. As an example, network devices such as routers and switches that are often used these serial ports to modify its configuration. As applications are some industrial automation systems and industrial products that also use the RS-232 serial interface for data exchange. The problem is that the use of RS-232 serial port is being displaced in many laptops through USB port, although in many devices and applications they are still needed.

This research raises a solution as design and implementation of a device to convey information in an RS-232 serial device to a USB connector and vice versa, so that can be plugged into any laptop, this device is commonly known as a serial-USB converter.

To achieve the development of the converter is used two main chips: the PIC18F4550 is a microcontroller which has two communication protocols supported by hardware (USB and RS-232) and the integrated circuit MAX232 is

used to change voltages that handles PIC18F4550 which are 5 and 0 volts to voltages in accordance with the standard RS-232, figure 2.
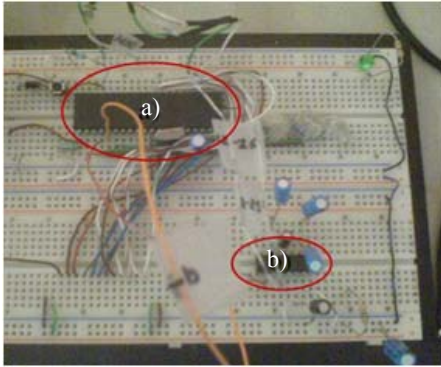

Fig. 2. a) PIC18F4550 b) MAX232

2.1. RS-232 Protocol

The protocol RS-232 (Standard ANSI/EIA-232) is the serial connection found on IBM compatible computers a global standard that governs the parameters of one of the serial communication modes. Through this protocol data transfer speeds standardize, control what uses the transfer, the levels of voltages used, the type of cable allowed the gap between computers, the connectors, etc.

In addition to transmission lines (Tx) and receive (Rx), serial communications have other lines of flow control (Handshake), where its use is optional depending on the connecting   device.

RS-232 is basically the selection of the baud rate (1200, 2400, 4800, etc.), Verification of data or parity (kidding pair, odd parity or no parity), stop bits after each data (1 or 2), and the number of bits per data (7 or 8), which is used for each symbol or character sent.

2.1.1. RS-232D (9-pin DB9 connector):

Table 1 shows the signals from each pin DB9 serial connector (9 pins) that can still be found in some of today's computers.

Table 1. RS-232 signals

SIGNAL	PIN No.	SIGNAL	PIN No.
Carrier Detect	1	Data Set Ready	6
Receive Data	2	Request To Send	7
Transmit Data	3	Clear To Send	8
Data Terminal Ready	4	Ring Indicator	9
Signal Ground	5		

Here are the main signs: Request to Send (RTS) is sent from the computer (DTE) to modem (DCE) to indicate that you want to transmit data. If the modem decides that is OK, sit for the CTS line. Clear to Send (CTS), affirmed by the modem after receiving the RTS signal indicates that the

computer can transmit. Data Terminal Ready (DTR) is asserted by the computer to inform the modem that is ready to receive data. Data Set Ready (DSR) is asserted by the modem in response to a DTR signal from the computer. Receive Line Signal Detect (RSLD) is asserted by the modem and informs the computer that has established a physical connection with another modem; Sometimes known as Carrier detector (CD). Transmit Data (TD) is the line where the data is transmitted one bit at a time. Receive Data (RD) is the line where the data is received one bit at a time, [4].

2.1.2. MAX232

MAX232 is an integrated circuit that converts signals from a RS-232 serial port to the appropriate signals for use in TTL (Transistor-Transistor Logic) compatible with digital logic circuits, available in-house 4-converters for this purpose [5]. See figure 3.
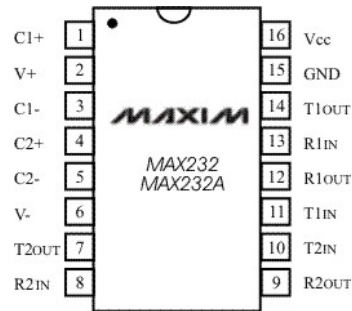

Fig. 3. Integrated circuit MAX232

The integrated circuit has been internally 2 TTL level converters to RS-232 and 2 with RS-232 to TTL so, in total we can handle 4 signals PC serial port, usually the most used are: TX, RX , RTS and CTS, the last two are used for handshaking protocol.

For the MAX232 to function properly it should be placed in external capacitors, all of this is shown in Figure 4, where wired TX and RX lines are the most commonly used for almost any application [6].
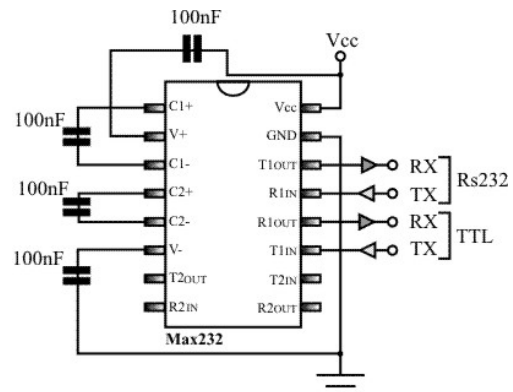

Fig. 4. MAX232 connection diagram

This integrated circuit is used to communicate a microcontroller or digital system with a PC or a system based on RS-232 bus. Its main features are:

- Vcc: 4.5V to 5.5V.
- Consumption: 4 mA (15 mA load to the output of 3 KOhm).
- TTL and CMOS compatible inputs.
- Maximum input voltage RS-232: / - 30v.
- Output voltage RS-232: / - 15v.
- The output voltage is typical of /-8V nominal load of 5 Kohm in RS-232.
- RS-232 Input Resistance: 5 kohm (to ground).
- Input Resistance TTL / CMOS: 400 Kohm (a positive).
- Tickets can be left to air.
- TTL input air is considered a "0" reversed in the output.
- RS-232 input to the air is considered a "1" to reverse the output.
- Outputs short-circuit continuously. RS-232 Outputs: / - 22 mA.
- TTL / CMOS: a mass of -10 mA to 30 mA positive.
- Data Rate: 200 Kbps (min 116 Kbps).

The voltage levels are useful to understand what happens to the voltage levels. When one receives a MAX232 IC to convert TTL, TTL logic changes from 0 to 3 to 15 V, and TTL logic changes from 1 to between -3 and-15V, and vice versa for the conversion of RS-232 to TTL. See table 2.

Table 2. MAX232 voltage

Tipo RS-232 de línea y nivel de la lógica	RS-232 Voltaje	TTL Tensión a/desde MAX232
Transmisión de Datos (Rx / Tx) 0 lógico	+3V a +15 V	0V
Transmisión de Datos (Rx / Tx) Lógica 1	-3V a -15V	5V
Control de Señales (RTS / CTS / DTR / DSR) 0 lógico	-3V a -15V	5V
Control de Señales (RTS / CTS / DTR / DSR) Lógica 1	+3 V a +15 V	0V

2.2. USB Protocol

The USB protocol is a specification for serial transmission of data and power distribution developed by industry leaders in telecommunications and computers introduced in the market for PC's and peripherals to enhance the slow serial and parallel. The USB provides transfer rates of up to 100 times faster compared to the parallel port and 25-pin serial DB-9, DB-25 and RS-232. The original target is relatively slow to connect peripherals (mice, printers, digital cameras, etc.) In a simple, fast way based on serial communications and could connect hard drives [7].

2.2.1. Electrical aspects

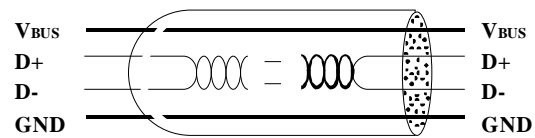Power level, the USB cable transfers the signal and power on 4 wires, figure 5.

Fig. 5. USB cable electrical aspect

A feeding level, the cable provides the rated voltage of 5V. It is necessary to correctly define the wire diameter so that there is not an important voltage drop in the cable. A termination resistor installed on the data line to detect and trace port settings (1.5 or 12 Mbps). A signal level, it is a twisted pair with a characteristic impedance of 90 Ω. The speed can be either 12 Mbps or 1.5 Mbps Receiver sensitivity can be at least 200mV and be able to support a good degree of rejection of common mode voltage. The clock is transmitted in the data stream, the encoding is NRZI type, and there is a device that generates a bit ensures that the clock frequency remains constant. Each packet is preceded by field sync.

The computers manage the consumer, have the capacity to put to sleep (suspend) or up to a USB peripheral. At rest, consumption is reduced (if possible), rendering the USB functionality. This management is aimed especially toward laptops [8].

En With regard to the types of transfer, the virtual link (pipe) can be of four types: control mode for configurations in which the link has no data loss, bulk, fast transfer mode is used for transmission of large amounts of non-critical information; interrupt, the transmission mode used for small packages aimed at human perceptions (mouse pointers), isochronous or stream real-time mode for transmission in real time as audio or video [9].

To implement the USB 2.0 protocol the serial-USB converter is used the PIC18F4550 microcontroller, which was chosen for its high performance and especially by its compatibility with the protocol.

2.2.3. PIC18F4550

The PIC has wide acceptance in the professional community engaged in the design of applications. The PIC has RISC (Reduced Instruction Set Computer) architecture, Harvard type, based on bank records, with a reduced instruction set of the same length and orthogonal, which processing is segmented. Some other features are:

- Easy to use.
- Superb technical information.
- Adjusted price and good availability.
- Understandable instructions and few in number.

- High average performance parameters: speed, power consumption, power, size, etc..
- Abundant, easy and inexpensive development tools.
- Support for software and hardware that facilitate the mobility into higher models.
- Wide range of models to find the most appropriate for each case.
- Disposition from Microchip to support customers.
- Among others. [10]

The PIC18F4550 microcontroller is manufactured by Microchip, is 8 bits and has the following main characteristics [11]:

- Flash program memory.
- Nano Watt Technology (very low power consumption).
- Complies with USB 2.0 protocol specifications.
- Support control transfers, interrupt, bulk and asynchronous.
- Full-speed 12 Mbps and 1.5 Mbps Low-speed.
- 1 Kbyte of RAM dual data cache for USB.
- Module USB 2.0 transceiver independent.
- USB voltage regulator.
- Pull-up resistors to the USB.
- ROM: 16K
- CPU Speed (MIPS): 12
- RAM: 2Kb
- Frequency: 48 MHz external crystal
- Allows ICSP programming
- USB 2.0 allows direct communication to hardware
- Number of pins: 40
- Temperature Range (C): -40 to 85
- Operating Voltage Range (V): 2 to 5.5
- Rango de voltaje operativo(V): 2 a 5.5

3. Architectural design of serial-USB converter.

Figure 6 shows the proposed architecture for serial-USB converter which basically consists of PIC18F4550 and the MAX232 chip and a set of LEDs, resistors, buttons, UTP cables for device connections and communication between them.
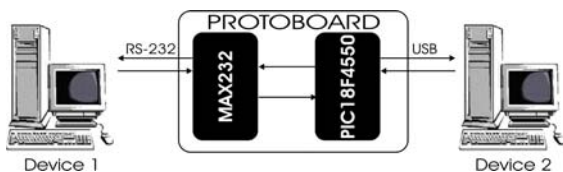

Fig. 6. Connection scheme serial-USB converter

For a better understanding of the design of the breadboard, it modulates its architecture and describes in detail each of these modules, figure 7.
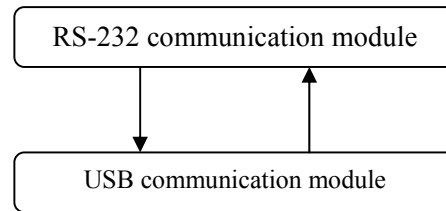

Fig. 7. Prototype modules.

RS-232 communication module

This module contains the MAX232 and RS-232 socket. The MAX232 is an integrated circuit that converts line levels of a RS-232 serial port to TTL levels and vice versa. This IC is responsible for managing data communication RS-232 serial port. Four of the port data cables of the RS-232 female, attached to the legs 7, 8, 13 and 14 of the MAX232, two for input and two for data output and then the MAX232 is connected to the PIC the following configuration: leg 9 of the MAX232 to the leg 5 of the PIC, leg 10 of the MAX232 to leg 4 of the PIC. The MAX232 chip is also used to change a voltage that handles 4550 to 5 and 0 volts to voltages in accordance with the standard RS-232, figure 2.

USB communication module

This module is the PIC18F4550 microcontroller and the USB connector, which provide power to the circuit and serves as a data transmitter and receiver. Data Cables USB port to connect the legs 23 and 24 of microcontroller and two cables: one serves as data input (D) that connects to the leg 24 and another is for output or data transmission (D-) that connects to the leg 23.

4. Implementation of the protoboard architecture

In this section we present the electronic components that were used during the implementation of the prototype, the methodology followed for its construction and the software that was programmed logical operation.

4.1 Construction of the prototype

The prototype design and development involved the construction of a breadboard hardware is required for the following Electronic Components:

- Protoboard
- 4 Capacitors 10uF
- 1 47 UF Capacitor
- 1 100 UF Capacitor
- Female serial port
- PIC18F4550 Microcontroller

- MAX232
- 4 resistors
- 1 Push Bottom1
- LED
- UTP Cables
- 1 20MHz Oscillator

Before any communication with the RS-232 must determine the parameters of initialization because the protocol needs to be manually configured on transmission of data, the parameters are:

- Serial Protocol (no parity).
- 9600 bps port speed
- Protocol flow control (RTS / CTS or XON / XOFF)

The program to control the PC can be anything that manages the serial port. For this project we used free software Terminal version 1.9b.

Figure 8, shows the schematic of the prototype with the PIC18F4550 microcontroller and MAX232 integrated circuit used in the construction of the converter.
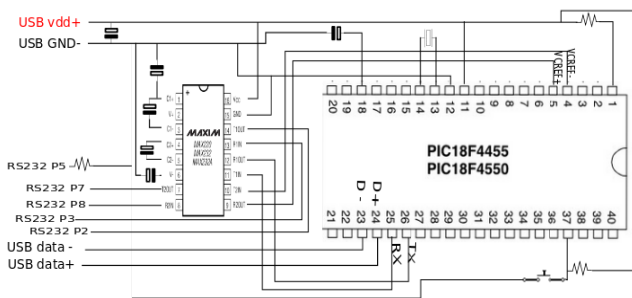
Fig. 8. Schematic of the prototype

For the construction of the breadboard converter capacitors used for power regulation in the MAX232, figure 9.
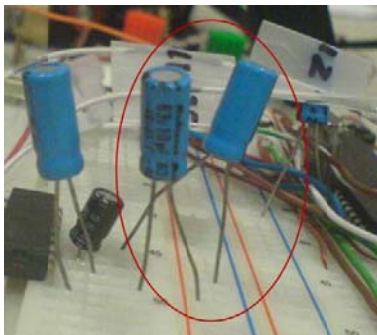
Fig. 9. Capacitors

Similarly resistance is needed as this monitors and regulates the energy required to operate in certain legs of the PIC and the circuit, figure 10.
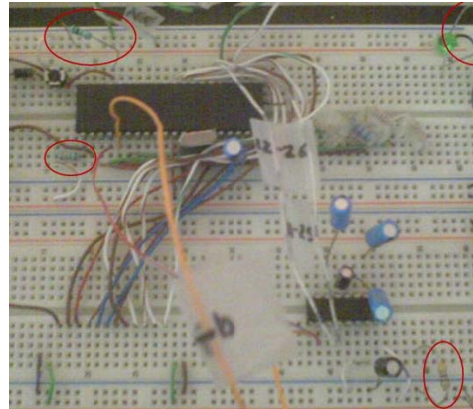
Fig. 10. Resistance.

Later we used a quartz crystal oscillator in the legs 14 and 15 of the PIC to generate clock pulses and so the microprocessor can operate at the speed set by the oscillator, figure 11.
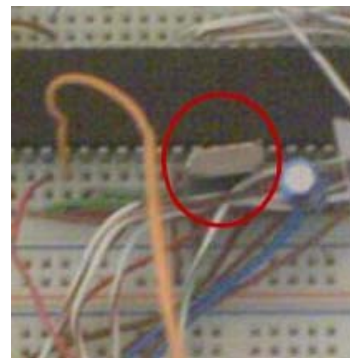
Fig. 11. Oscillator

Finally, the microcontroller is programmed to implement a communication bridge between USB and RS-232 with the microcontroller account.

4.2. Logic Programming microcontrollers

Logic programming for each of the functions performed by integrated circuits used MPLAB project. MPLAB is a tool to write and develop code in assembly language for microcontrollers, this software incorporates all the tools necessary for the completion of any project, as well as a text editor with a simulator which can execute code step by step in order to see its evolution and the state in which your records are at all times. Here is an extract of the code of the "ProcessIO(void)" which takes incoming data from RS-232 port character by character reading them and keeping them in a vector or array of characters and then sends the full array of USB buffer, thus providing all the information in the RS-232 serial port to USB port.

```
if (mTxRdyUSART() == 0 &&
mUSBUSARTIsTxTrfReady() == 0)
 {
        RS232_Input_Data[size]=getcUSART();
        ++size;
        putUSBUSART(RS232_Input_Data,64);
}
```

The following code fragment is part of the same feature and allows data from the USB buffer to be placed in the buffer of the RS-232 by accepting data over USB to serial; this feature along with the previous offer two-way communication between the USB and RS-232 serial port implemented on the device.

```
if (RS232_Out_Data_Rdy == 0)
{
  LastRS232Out = getsUSBUSART(RS232_Out_Data,64);
  if(LastRS232Out > 0)
        {
        RS232_Out_Data_Rdy = 1;
        RS232cp = 0;
        }
}
```

5. Implementation of prototype printed circuit board

Once the prototype presented the expected results, we proceeded to build the hardware, implementing the prototype in the printed circuit board.

5.1 Development of implementation in printed circuit board

For the construction of the hardware on a printed circuit board required the following components:
- Copper Tablet
- 4 Capacitors 10uF
- 1 47 UF Capacitor
- 1 100 UF Capacitor
- 2 Plastic Bases CI
- Female serial port
- PIC18F4550 Microcontroller
- MAX232
- 4 resistors
- 1 Push Bottom
- 1 LED
- 1 20MHz Oscillator
- Ferric chloride
- Heat Transfer Paper
- 1 iron

Then there will be a description of the methodology used in the construction of the printed circuit serial-USB converter, considering that from the beginning to determine the parameters of initialization because the RS-232 protocol needs to be configured manually before data transmission.

Once designed the schematic in Protel DXP program Altium Company, moved to another program called Advanced PCB Design and image generated in it, was imported into Corel Draw to make a black the schematic. In Figure 8, the schematic showed to be printed on thermal transfer paper.

It took the copper plate and washed thoroughly with water and detergent, once dry placed on a table and placed over the thermal transfer paper and printed circuit so that the porous plate is in copper and the flat side is up, once you set focus tape will not move allowing you to iron it. After carefully peeling off the transfer paper to verify that the toner left on the plate and had to repaint the lines with a permanent marker black.

Once marked out the lines it was necessary to place the plate in a plastic container and pour the ferric chloride to a considerable height and start to swing for about 20 minutes until the copper chloride dissolved on the card and only the toner was left. The card was removed from the liquid and washed until toner was removed and copper appeared.

Once clean and dry it was verified that the lines are complete, holes were drilled where they laid the groundwork PIC18F4550 microcontroller and MAX232 IC and the capacitors, resistors, oscillator, LED and other
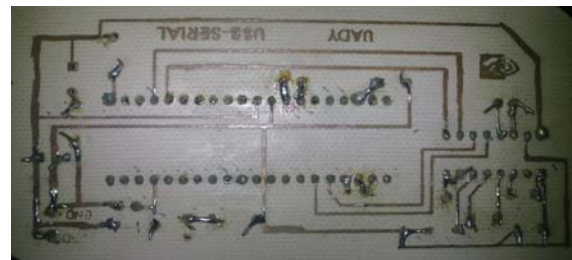
components necessary and welded respectively , figure 12.

Fig. 12. Printed circuit board serial-USB converter

Posteriorly the continuity of each leg of the base PIC18F4550 of the leg of MAX232 was verified and once placed in their respective bases and soldered, all components and cables for RS-232 serial ports and USB to the plate were tested to evaluate the performance of the plate. In figure 13 we see the final hardware.

Fig. 13. serial-USB converter

6. Conclusions

In this paper we presented the design and implementation of a serial-USB converter to transfer data from devices that have RS-232 serial port.

The project had the desired results as it led to the communication from computer to computer and from computer to device, figure 14 shows one of the tests performed on a switch, which is configured from a laptop, with the characteristic that the switch can only be configured through a serial port RS-232 and the laptop does not have the serial port, so the drive through the configuration was done from the laptop's USB port. In this paper we successfully tested the connection from computer to serial port device with a female RS-232 serial port.
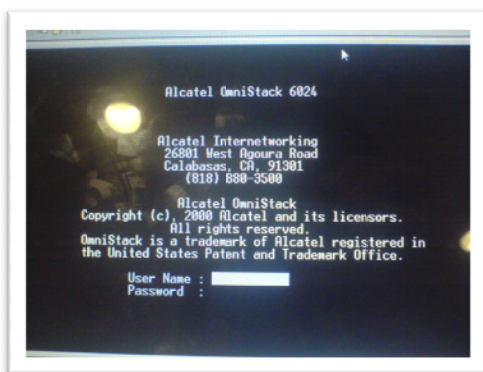

Fig. 14. Switch configuration interface using the serial-USB converter

In future it is planned that the device is hermaphrodite i.e. containing connectors male / female RS-232 serial port, just as it is projected to place on the same device LCD screen so the user can verify that the connection and transmission of information are carried out, other relevant action is also adding buttons and so with the help of the screen give you a list of user options so you can choose the connecting device, the above is achieved by programming macros for the routines of choice and connections.

Referencias

[1] Sistema de video sincronizado.
http://upcommons.upc.edu/pfc/bitstream/2099.1/9544/1/memoria.pdf .
Última consulta Septiembre 10 de 2010

[2] Empleo de Google Earth como navegador GPS.universal.
http://www.uhu.es/carlos.barranco/D_INVEST/Articulo%20revista.pdf
Última consulta Septiembre 10 de 2010

[3] Güzin Özmen y Yüksel Özbay, Monitorización de Señales Electromiográficas con Interfaz USB, *International Journal of Arts and Sciences,* Selcuk University. Turkey. Pag. 61 – 68, 2009

[4] http://rdedatos.tripod.com/rs232.htm (29/06/10)

[5] Dual EIA-232 Driver/Receiver. MAX232 datasheet from Texas Instruments.

[6] http://www.smartsystemselectronics.com.mx/index.php?page=shop.product_details&category_id=2&flypage=flypage.tpl&product_id=2&vmcchk=1&option=com_virtuemart&Itemid=56 (29/06/10)

[7] Bernal Jiménez, Andrés. Desarrollo de un Software de Medición del Estrés para un Dispositivo Foto-pletismógrafo Basado en el Protocolo USB 2.0. Tesis. Centro de Investigación y de Estudios Avanzados del IPN, Unidad Zacatenco. Abril 2010

[8] http://www.i-micro.com/pdf/articulos/usb.pdf (29/06/2010)

[9] http://www.fujitsu.com/downloads/EU/es/soporte/escaneres/cursousb.pdf (29/06/2010)

[10] Valdés, et, al. Microcontroladores Fundamentos y Aplicaciones con PIC. 2007

[11] Microchip. PIC18F2455/2550/4455/4550, Data Sheet. 28/40/44 - Pin High-Performance, Enhanced Flash USB Microcontrollers with nanoWatt Technology.

Erick Alejandro Patrón Coral [A4080876@alumnos.uady.mx] is currently the seventh semester of the degree in computer science Faculty of Mathematics at the Multidisciplinary Unit Tizimin. He has participated in software development projects, robotics, electronics and projects fair in conjunction with Candelario Pinto.

Candelario Pinto Alcocer [A4080891@alumnos.uady.mx] currently is studying a degree in computer science Faculty of Mathematicsat from the Autonomous University of Yucatan (UADY) in the Multidisciplinary Unit Tizimin. He has participated in software development projects, robotics, electronics and projects fair in conjunction with Erick Patron.

Raciel Omar Poot Jiménez [raciel.poot@uady.mx,racielomar@gmail.com] received the degree in Engineer Computer from the Autonomous University of Yucatan

(UADY) in 2008.Currently, he is a full time teacher at the Autonomous University of Yucatan since 2008 at the department of electronic in Tizimin Mexico. He has participated in development projects of electronic engineering.

Erika Rossana Llanes Castro [lcastro@uady.mx] received a degree in Computer Science from the Autonomous University of Yucatan (UADY) in 2002. She is about to receive a Computer Science degree from Monterrey Technological Institute (ITESM), Campus Estado de Mexico.

Currently, she is a full time academic technician at the Autonomous University of Yucatan since 2002 in the department of computer in Tizimin Mexico. She has participated in software engineering development projects.

Lizzie Edmea Narváez Díaz [lendiaz@uady.mx] received a degree in Computer Science from the the Autonomous University of Yucatán (UADY) in 1997. She received a Master of Computer Science degree from Monterrey Technological Institute (ITESM), Campus Cuernavaca, in 2007.

She has been a full time teacher at the Autonomous University of Yucatán since 2000 in the Networking department in Tizimin Mexico. She has participated in software engineering development projects.

Victor Manuel Chi Pech [victor.chi@uady.mx] obtained his degree in Computer Science from the Autonomous University of Yucatan (UADY) in 1996 and his M. Sc. degree in Wireless Network from Monterrey Technological Institute (ITESM), Campus Cuernavaca, in 2007.

Victor Chi has worked since 2000 at the Autonomous University of Yucatan, as a full time professor. He has participated in software engineering development projects.