 **Universidad Autónoma de Nuevo León**

**Facultad de Ingeniería Mecánica y Eléctrica**

#LAB. CONTROLADORES Y MICROCONTROLADORES PROGRAMABLES

#Práctica P3

"Microcontroladores"

\*Nombre o nombres de los integrantes junto con su matrícula:

#Verónica Yazmín Gómez Cruz #1884224

#Nahaliel Gamaliel Ríos Martínez #1884244

#Ing. Jesus Daniel Garza Camarena

Semestre Febrero 2021 – Junio 2021

# MN1N2

San Nicolás de los Garza, N.L. #11.03.2021

# Objetivo

Conocer cómo se utilizan las macros del Microcontrolador y ver el cómo se leen los estados de las entradas de un MCU.

# Introducción.

Los rebotes son las falsas pulsaciones (ruido) que se producen al hacer falsos contactos en el interruptor, para eliminarlos se utiliza el proceso "Debounce".

Los dispositivos electrónicos al cambiar de estado generan una señal que, sin ser perfectamente cuadrada, en general es más o menos "recta", en el entorno real muchos dispositivos físicos habitualmente generan ruido en los flancos de señal, por ejemplo, en la variación de tensión que ocurre cuando el cambio de estado se genera por un pulsador, en el rango de unos micro-segundos la señal es puro ruido, esos picos pueden provocar disparos múltiples de una interrupción.

**Diagrama de bloques**

Diagrama

Descripción generada automáticamente

# Diagrama de flujo.

Diagrama

Descripción generada automáticamente

# Materiales utilizados

1 ATmega328p

2 LED’s rojos

1 LED’s Amarillo

2 Push button

4 resistencias 220

1 Arduino Uno

1 Protoboard

Jumpers

# Código en Atmel.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* LLENAR ESTE ESPACIO CON LOS SIGUIENTES DATOS: \*

\* Nombre: Verónica Yazmín Gómez Cruz \*

\* Nahaliel Gamaliel Rios Martinez \*

\* Hora clase: N1-N2 \*

\* Día: M \*

\* N° de lista: 17, 18 \*

\* N° de Equipo: 7 \*

\* Dispositivo: ATMEGA328P \*

\* Rev: 1.0 \*

\* Propósito de la actividad: Con dos botones a la \*

\* entrada ver a la salida 3 leds con el siguiente \*

\* funcionamiento: \*

\* 1- Salida\_1 = lógica AND \*

\* 2- Salida\_2 = lógica OR \*

\* 3- Salida\_3 = lógica EXOR \*

\* Fecha: 5.3.2021 \*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*atmega328P PIN - OUT\*/

/\* PIN - OUT

atmega328P

-------

PC6 |1 28| PC5

PD0 |2 27| PC4

PD1 |3 26| PC3

PD2 |4 25| PC2

PD3 |5 24| PC1

PD4 |6 23| PC0

VCC |7 22| GND

GND |8 21| AREF

PB6 |9 20| AVCC

PB7 |10 19| PB5

PD5 |11 18| PB4

PD6 |12 17| PB3

PD7 |13 16| PB2

PB0 |14 15| PB1

--------

\*/

/\*atmega328P PIN FUNCTIONS\*/

/\*

atmega328P PIN FUNCTIONS

pin function name pin function name

1 !RESET/PCINT14 PC6 15 PCINT1/OC1A PB1

2 RxD/PCINT16 PD0 16 PCINT2/OC1B/SS PB2

3 TxD/PCINT17 PD1 17 PCINT3/OC2A/MOSI PB3

4 INT0/PCINT18 PD2 18 PCINT4/MISO PB4

5 INT1/PCINT19/OC2B PD3 19 PCINT5/SCK PB5

6 PCINT20 PD4 20 ANALOG VCC AVCC

7 +5v VCC 21 ANALOG REFERENCE AREF

8 GND GND 22 GND GND

9 XTAL1/PCINT6 PB6 23 PCINT8/ADC0 PC0

10 XTAL2/PCINT7 PB7 24 PCINT9/ADC1 PC1

11 PCINT21/OC0B PD5 25 PCINT10/ADC2 PC2

12 PCINT22/OC0A/AIN0 PD6 26 PCINT11/ADC3 PC3

13 PCINT23/AIN1 PD7 27 PCINT12/ADC4/SDA PC4

14 PCINT0/AIN1 PB0 28 PCINT13/ADC5/SCL PC5

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Bibliotecas\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <avr/io.h>//se incluyen las Bibliotecas de E/S del AVR atmega328P

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Macros y constantes\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define F\_CPU 1000000UL //1 Mhz

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Variables globales\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//--Espacio para declarar variables globales

//Entradas

#define boton\_1 PIND7 // PIN 13

#define boton\_2 PIND0 // PIN 2

//Salidas

#define LED\_1 PORTB0 // PIN 14

#define LED\_2 PORTB4 // PIN 18

#define LED\_3 PORTC0 // PIN 23

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Funciones\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//--Espacio para Establecer funciones

void initialize\_ports(void);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Declaración de Funciones\*\*\*\*\*\*\*\*\*\*\*\*\*/

//--Espacio para declarar funciones

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Programa principal\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

int main(void)

{

//--Inicialización

initialize\_ports(); // va hacía la inicialización de puertos

//--Ejecución

while (1)

{

//Boton 1 - set = push esta en 0

//Boton 2 - Clear = push esta en 1

//Logica AND

if (bit\_is\_clear(PIND,boton\_1) && bit\_is\_set(PIND,boton\_2)) {

PORTB |=\_BV(LED\_1); //Encender

}else{

PORTB &=~\_BV(LED\_1);//Apagar

}

//Logica OR

if (bit\_is\_clear(PIND,boton\_1) || bit\_is\_set(PIND,boton\_2)) {

PORTB |=\_BV(LED\_2); //Encender

}else{

PORTB &=~\_BV(LED\_2); //Apagar

}

//Logica XOR

if ( bit\_is\_clear(PIND,boton\_1) == bit\_is\_set(PIND,boton\_2) ) {

PORTC &=~\_BV(LED\_3); //Apagar

}else{

PORTC |=\_BV(LED\_3); //Encender

}

}

} // END MAIN

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Definición de funciones\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//Descripcion de lo que hace la funcion: \*

//initialize\_ports : inicializa los puertos de entrada o \*

//salida \*

// \*

// \*

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void initialize\_ports(void)

{

//Entradas

DDRD &=~(1<<boton\_1);

PORTD |=(1<<boton\_1); //pull-up

DDRD &=~(1<<boton\_2);

PORTD &=~(1<<boton\_2); // pull-down al poner la

//Salidas

//Declaramos puertos de salida

DDRB |=\_BV(LED\_1);

DDRB |=\_BV(LED\_2);

DDRC |=\_BV(LED\_3);

//Por seguridad se inicializa en 0

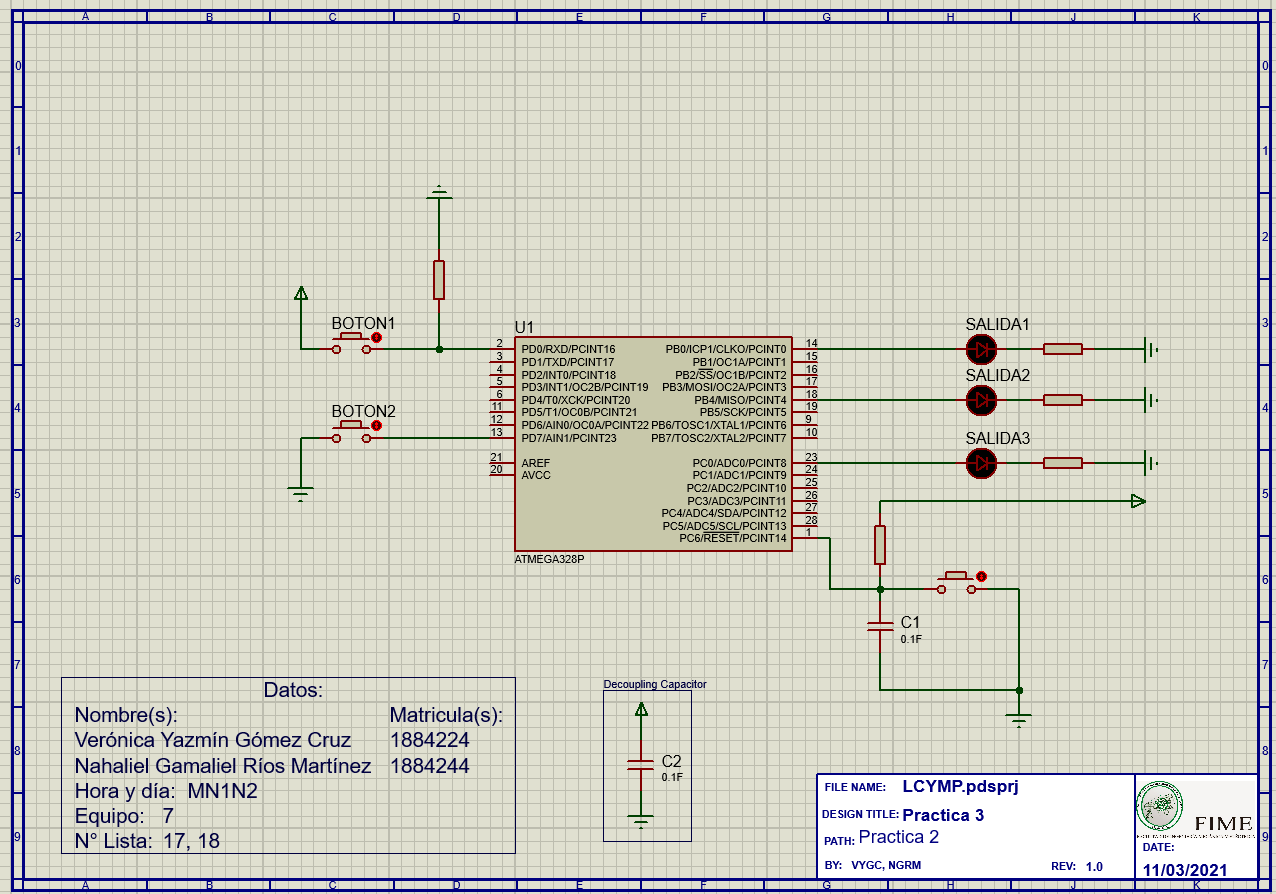
PORTB &=~\_BV(LED\_1);

PORTB &=~\_BV(LED\_2);

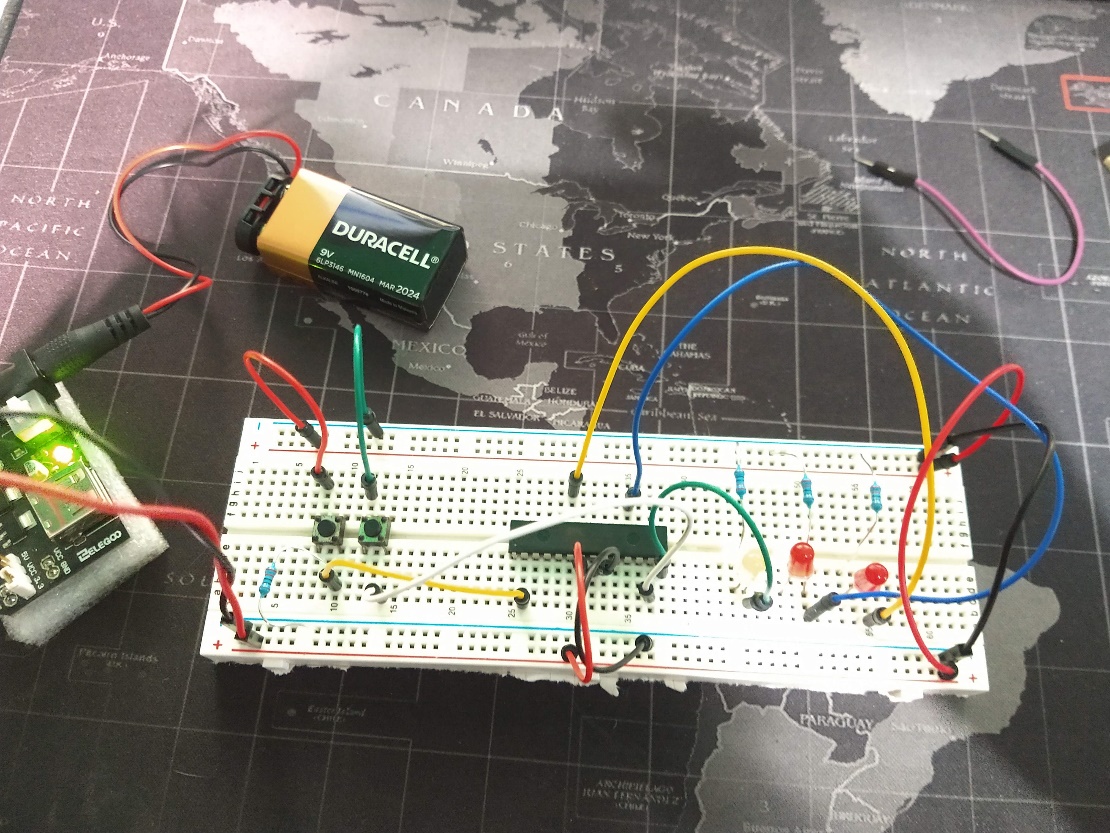
PORTC &=~\_BV(LED\_3);

}

# Diagrama del circuito en PROTEUS.



**Fotografía del Protoboard armado**



**Conclusión**

En esta práctica se utilizaron 2 botones y 3 leds para realizar la simulación de las lógicas de las compuertas AND OR Y EXOR a través de Proteus y MICROCHIP STUDIO para su codificación, durante el desarrollo se logró comprender el uso de macros del microcontrolador, así como el uso y teoría del pull-up y pull-down, lo cual dependiendo de cual se utilice se activará la salida de corriente con “1” o con “0”.

**Bibliografía**

Prof. Bolaños D. (2019). Electrónica. 2021, de NA Sitio web: <http://www.bolanosdj.com.ar/MOVIL/ARDUINO2/Pull_Up_Down.pdf>