



## Laboratorio de Microprocesadores - 86.07

# Trabajo Práctico N°2:

## Manejo de Puertos

Profesor:			Ing. Guillermo Campiglio									
Cuatrimestre/Año:			1º/2020									
Turno de las clases prácticas			Miércoles									
Jefe de trabajos prácticos:			Ing. Pedro Ignacio Martos									
Docentes guía:			Ing. Fabricio Baglivo, Ing. Fernando Pucci									
Fecha de presentación:			27/05/2020									
Alumno			Seguimiento del proyecto									
Nombre	Apellido	Padrón										
Maximiliano Daniel	Reigada	100565										

### Observaciones:

.....

.....

.....

.....

.....

.....

.....

.....

Fecha de aprobación			Firma J.T.P	

# Índice

<b>1. Objetivo</b>	<b>1</b>
<b>2. Descripción</b>	<b>1</b>
<b>3. Diagrama de conexiones en bloques</b>	<b>1</b>
<b>4. Circuito esquemático</b>	<b>2</b>
<b>5. Listado de componentes</b>	<b>2</b>
<b>6. Diagrama de flujo del Software</b>	<b>3</b>
<b>7. Código de programa</b>	<b>4</b>
7.1. Uso de resistor pull-down externo . . . . .	4
7.2. Uso de resistor pull-up interno . . . . .	4
<b>8. Resultados</b>	<b>5</b>
<b>9. Conclusiones</b>	<b>6</b>

## 1. Objetivo

El presente trabajo práctico tiene como objetivo poner en práctica el uso de los registros de los puertos de entrada/salida en un microcontrolador, y ver la utilidad de las resistencias pull-up internas que se encuentran conectadas a los pines de este.

## 2. Descripción

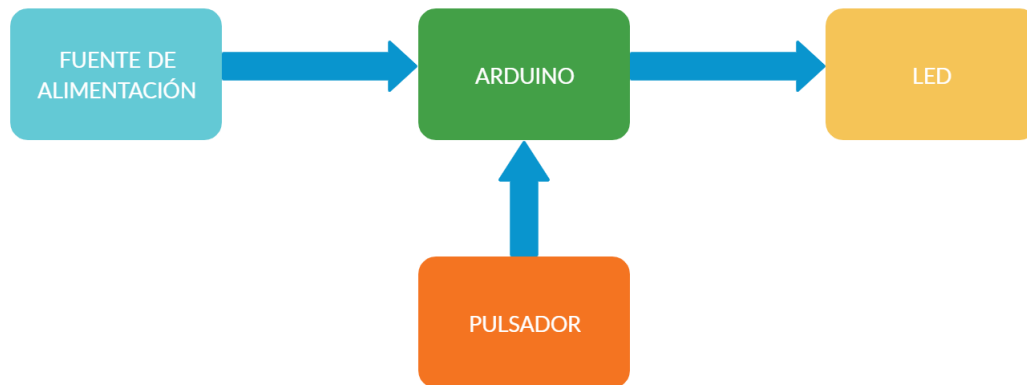
Mediante la implementación de una placa de microcontrolador de código abierto basado en el microchip ATmega328P, denominada Arduino Uno, se busca controlar el encendido/apagado de un LED a partir del uso de un pulsador.

Se procede a conectar el LED al pin 2 del puerto B del microcontrolador, colocando un resistor en serie para limitar la corriente de este. A su vez, se conecta el pulsador entre  $V_{CC}$  y el pin 0 de dicho puerto, empleando un resistor a modo de pull-down, para establecer un estado lógico bajo en el pin cuando este se encuentre en reposo.

La idea principal de este trabajo radica en que el LED se mantenga encendido mientras el pulsador se encuentra presionado, y permanezca apagado cuando esto no suceda. Como práctica adicional, se pretende también analizar las modificaciones a implementar tanto en programación como en circuito, para hacer uso de la resistencia pull-up interna de los puertos del microcontrolador.

## 3. Diagrama de conexiones en bloques

A continuación, puede verse el diagrama de conexiones en bloques correspondiente a este trabajo práctico:



**Figura 1:** Diagrama de conexiones en bloques.

## 4. Circuito esquemático

Se procede a realizar el esquemático del circuito a implementar. Cabe destacar que, si bien en la realidad los componentes se encuentran conectado al Arduino Uno y este a su vez a una fuente de tensión externa, resulta más útil representar en el esquema solo los componentes electrónicos esenciales del sistema, conectados a referencias de tensión de un determinado valor:

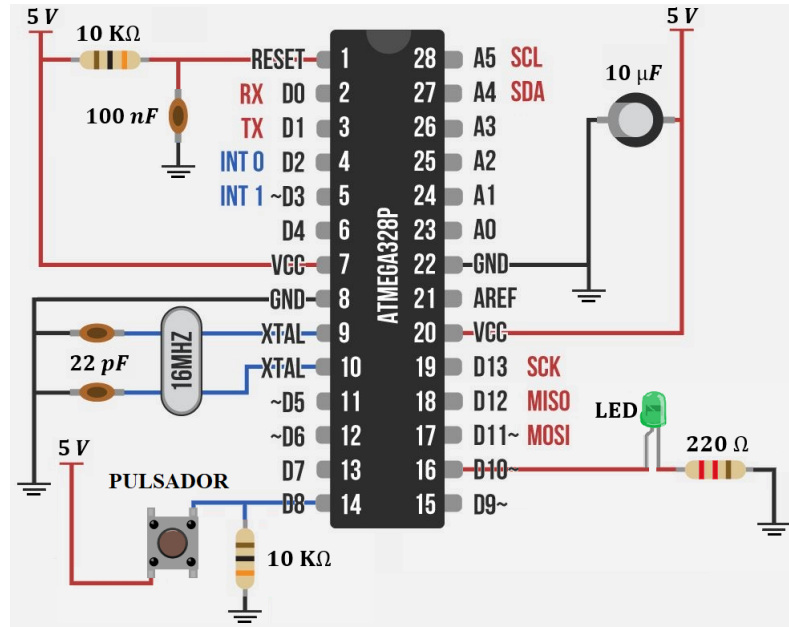


Figura 2: Circuito esquemático.

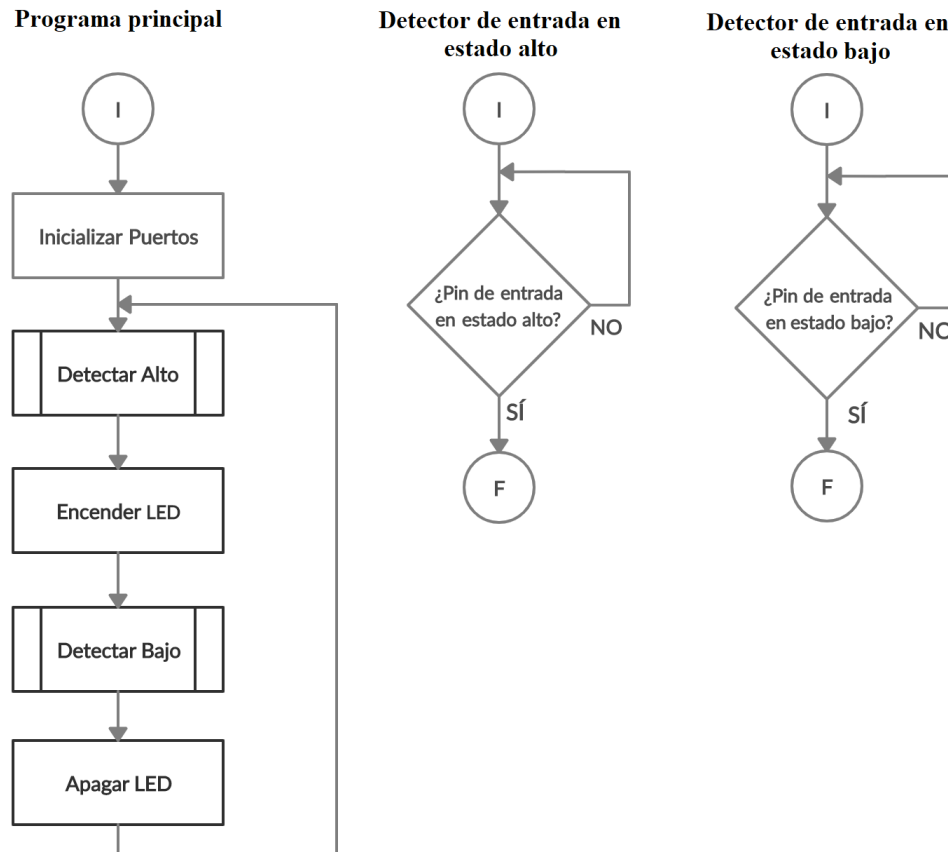
## 5. Listado de componentes

En el desarrollo de este trabajo práctico se utilizan los siguientes componentes:

- 1 LED verde de 5mm
- 1 Pulsador Dip Tact Switch 6x6x5mm
- 1 Resistor de 220  $\Omega$
- 1 Resistor de 10  $k\Omega$
- 4 Cables de conexión
- 1 Protoboard
- 1 Arduino Uno (con microcontrolador ATmega328P incluido)

## 6. Diagrama de flujo del Software

A continuación, se explicita el diagrama de flujo correspondiente al código desarrollado:



**Figura 3:** Diagrama de flujo.

## 7. Código de programa

A continuación, pueden verse los códigos desarrollados para este proyecto:

### 7.1. Uso de resistor pull-down externo

```

1  .INCLUDE "M328PDEF.INC"
2
3  .EQU PUERTO_SALIDA = PORTB
4  .EQU PUERTO_ENTRADA = PINB
5  .EQU CONF_PUERTO = DDRB
6  .EQU BOTON = 0
7  .EQU LED = 2
8
9  .CSEG
10 .ORG 0x0000
11     JMP MAIN
12
13 .ORG INT_VECTORS_SIZE
14
15 MAIN:
16
17     LDI R16, HIGH(RAMEND)      ;Se inicializa el Stack Pointer al final de la RAM
18     OUT SPH, R16
19     LDI R16, LOW(RAMEND)
20     OUT SPL, R16
21
22     ; Se inicializan puertos
23     LDI R18, 0x24
24     LDI R19, 0x00
25     OUT CONF_PUERTO, R18
26     OUT PUERTO_SALIDA, R19
27
28     ; Se controla el LED
29 CONTROLAR_LED:
30     CALL DETECTAR_ALTO        ; Espera hasta que se presione el pulsador
31     SBI PUERTO_SALIDA, LED
32     CALL DETECTAR_BAJO        ; Espera hasta que se deje de presionar el pulsador
33     CBI PUERTO_SALIDA, LED
34     JMP CONTROLAR_LED        ; Se reinicia el ciclo de control
35
36
37
38     ; Detecta flancos bajos en pin conectado al botón
39 DETECTAR_BAJO:
40     SBIC PUERTO_ENTRADA, BOTON
41     JMP DETECTAR_BAJO
42     RET
43
44     ; Detecta flancos altos en pin conectado al botón
45 DETECTAR_ALTO:
46     SBIS PUERTO_ENTRADA, BOTON
47     JMP DETECTAR_ALTO
48     RET

```

### 7.2. Uso de resistor pull-up interno

```

1  .INCLUDE "M328PDEF.INC"
2
3  .EQU PUERTO_SALIDA = PORTB
4  .EQU PUERTO_ENTRADA = PINB
5  .EQU CONF_PUERTO = DDRB
6  .EQU BOTON = 0
7  .EQU LED = 2
8
9  .CSEG
10 .ORG 0x0000

```

```

11      JMP MAIN
12
13      .ORG INT_VECTORS_SIZE
14
15      MAIN:
16          ;Se inicializa el Stack Pointer al final de la RAM
17          LDI R16, HIGH(RAMEND)
18          OUT SPH, R16
19          LDI R16, LOW(RAMEND)
20          OUT SPL, R16
21          ; Se inicializan puertos
22          LDI R17, 0x01
23          LDI R18, 0x24
24          OUT CONF_PUERTO, R18
25          OUT PUERTO_SALIDA, R17
26          ; Se controla el LED
27      CONTROLAR_LED:
28          CALL DETECTAR_ALTO          ; Espera hasta que se deje de presionar el pulsador
29          CBI PUERTO_SALIDA, LED
30          CALL DETECTAR_BAJO          ; Espera hasta que se presione el pulsador
31          SBI PUERTO_SALIDA, LED
32          JMP CONTROLAR_LED          ; Se reinicia el ciclo de control
33
34
35          ; Detecta flancos bajos en pin conectado al botón
36      DETECTAR_BAJO:
37          SBIC PUERTO_ENTRADA, BOTON
38          JMP DETECTAR_BAJO
39          RET
40
41          ; Detecta flancos altos en pin conectado al botón
42      DETECTAR_ALTO:
43          SBIS PUERTO_ENTRADA, BOTON
44          JMP DETECTAR_ALTO
45          RET
46

```

## 8. Resultados

Luego de desarrollar el código descripto en las secciones anteriores y armar el circuito dispuesto, se logra controlar el encendido y apagado del led mediante el accionar y reposo del pulsador dispuesto en uno de los puertos de entrada del microcontrolador.

Por otra parte, se consigue analizar que para utilizar la resistencia pull-up interna del puerto de entrada, se debe modificar el circuito de la siguiente manera:

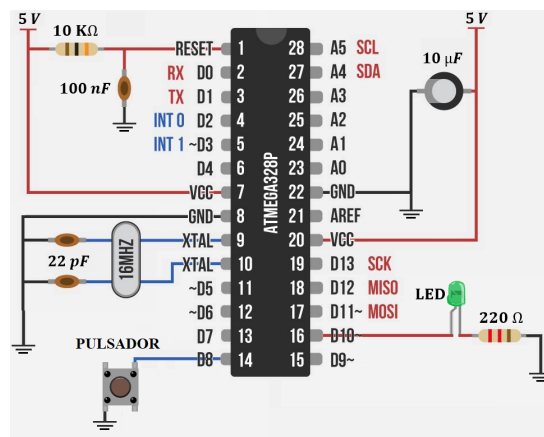


Figura 4: Circuito esquemático usando resistencia interna pull-up.

De esta forma ya no es necesario utilizar el resistor de  $10k\Omega$  externo, sin embargo, se debe modificar el programa. Para empezar, luego de indicar que pines se utilizan como entradas y salidas, se debe poner un 1 en el bit 0 del registro PORTB para así poder habilitar la resistencia interna. Por otro lado, como ahora se trata con una conexión pull-up, se debe invertir la lógica utilizada en el ejercicio anterior, ya que cuando se presiona el botón, en el pin de entrada se tiene un nivel lógico bajo, y de lo contrario se tiene un nivel lógico alto.

Finalmente, se procede a realizar un listado de los componentes utilizados y sus costos:

Componentes	Costos
Led verde de 5mm	\$7,00
Pulsador Dip Tact Switch	\$22,00
Resistor de $220\ \Omega$	\$5,50
Resistor de $10\ k\Omega$	\$5,50
Cables de conexión	\$8,00
Protoboard	\$232,00
Arduino Uno	\$720,00
TOTAL	\$1000,00

## 9. Conclusiones

Tras haber realizado todos los pasos pedidos en el enunciado de este trabajo práctico, resta destacar las conclusiones que la experiencia ha aportado.

Si bien en este caso de aplicación, se utilizan solo los pines de un determinado puerto, tanto para encender el led, como para detectar el estado del pulsador, el desarrollo del programa da una idea bastante completa de que cosas deberían modificarse en este si fuera necesario utilizar varios puertos. Por otra parte, hacer uso de las resistencias pull-up internas es una práctica cuya utilidad se extiende más allá de este trabajo, y puede ser útil a futuro.