Laboratorio de microprocesadores 86.07

Guía de Trabajos Prácticos

1er cuatrimestre

Año 2022

Trabajo Práctico Nº 1 - Manejo de Puertos

Objetivo:

Usar los registros de los puertos, ver utilidad de R pullup interna

Practica:

Realizar un programa que prenda un LED cuando se presiona el pulsador 1 y quede parpadeando hasta que se apague cuando se presiona el pulsador 2. Se debe utilizar y calcular los CM de la rutina de retardo dada a continuación.

El LED está conectado a un pin del microcontrolador y los pulsadores a otros dos pines como indica la figura.

Notar que un pulsador al estar accionado pone el pin a GND y el otro a VCC. Uno usa R de pull up interna y el otro no. ¿Cuál de estas 2 conexiones de pulsadores le parece mejor y por qué?

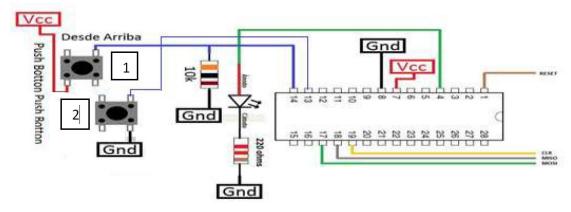
Implementar la lectura de los pulsadores con antirrebote

Materiales

- 1 LED
- 1 Resistencia de 220 Ohms
- 1 Resistencia de 10Kohms
- 2 Pulsador
- 1 Microcontrolador ATmega328p

Programador USBasp V3.0

Diagrama Esquemático



Lectura recomendada

"The AVR microcontroller and embedded systems. Embedded system using Assembly and C". Autores:

MUHAMMAD ALI MAZIDI, SARMAD NAIMI, SEPEHR NAIMI

Ejemplo de generación de un retardo

```
;-----
; Rutina de Retardo de 5ms
                                          ;Cálculo del tiempo de retardo)
;-----
                                          ;------
                                          Llamada 5ms * 1
delay5ms:
      ldi Temp1, 66; para 8mhz; 1 ciclo
                                          Ldi Temp1 * 1
LOOP0:
                                          LDI Temp 2 * 66
      ldi temp2, 200; 1 ciclo
LOOP1:
      dec temp2; 1 ciclo
                                          dec temp2 * 200
      brne LOOP1; 1 si es falso 2 si es
                                          brne loop 2 * 200
verdadero
                                          dec temp1 * 66
      dec Temp1;1
      brne LOOP0; 2
                                          brne Loop2 * 66
                                          ret * 1
      ret
                                          Tiempo=
                                          (5+1+66+(200+400)*66+66+132+4)/8000000=0
                                          ,00498seg ~= 5ms
```

Rutina de retardo a utilizar

```
; Assembly code
; Delay 8 000 000 cycles
; 1s at 8.0 MHz
Idi r19, 150
Idi r20, 128
L1: dec r20
brne L1
dec r19
brne L1
dec r18
brne L1
```

Resumen teórico

LED

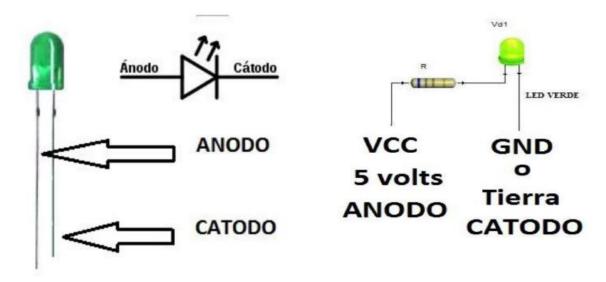
LED acrónimo de Light Emitting Diode o Diodo Emisor de Luz, es un dispositivo semiconductor que emite luz al

circular a través de él una corriente eléctrica.

Los LEDs como los diodos normales tienen un ánodo y un cátodo, los cuales se pueden identificar de manera fácil,

siendo el ánodo la pata más larga como se observa en la figura.

Parpadeo de un LED



Como se puede apreciar en la figura superior, al LED se le coloca una

resistencia en serie para limitar la corriente del mismo, en este caso que se alimentará con 5v se sugieren usar

resistencias de 220 Ohms.

Para calcular las resistencias exactas y obtener un desempeño óptimo se realiza a través de la siguiente fórmula:

$$R = \frac{Vcc - V_{LED}}{I_{LED}}$$

Color	Caida de tensión (VLED) V	Intensidad máxima (ILED) mA	Intensidad media (ILED)mA		
Rojo	1.6	20	5 – 10		
Verde	2.4	20	5 – 10		
Amarillo	2.4	20	5-10		
Naranja	1.7	20	5 – 10		

Caída de tensión e intensidad.

Tensión de alimentación, VCC, es el voltaje aplicado al circuito Caída de tensión del LED, Vled es el voltaje necesario para el funcionamiento del LED, generalmente está entre 1.7 y 3.3 voltios, depende del color del diodo y de la composición de metales.

El microcontrolador tiene varios puertos de los cuales podemos hacer uso, estos puertos los podemos configurar como nosotros queramos, como entrada o como salida, para poder hacer esto es necesario escribir en los registros del puerto para darle las instrucciones necesarias. Existen tres principales formas de controlar los puertos, tomamos como ejemplo el puerto B. DDR

Para hacer que el puerto B se comporte como entrada o salida, es necesario setear el DDR, este registro no activará ni desactivará ningún pin del microcontrolador, simplemente le indicará al puerto si este será entrada o salida.

The Port B Data Direction Register – DDRB

Bit	7	6	5	4	3	2	1	0	
	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
Read/Write	R/W	-							
Initial Value	0	0	0	0	0	0	0	0	

Para indicarle al DDR si el puerto será de entrada o salida, el 1 indica salida, y el 0 entrada, se le puede escribir como hexadecimal, decimal, o binario, por ejemplo, si queremos que todos los bits del puerto sean salidas lo podemos escribir como sigue:

DDRB=0xFF; //Como Hexadecimal, DDRB=255; //Como Decimal, DDRB=0b11111111; //Como Binario

Si queremos que algunos bits funcionen como entradas:

DDRB=0x8C; //Como Hexadecimal, DDRB=140; //Como Decimal, DDRB=0b10001100; //Como Binario

Se puede escribir en cualquiera de los tres tipos (binario hexadecimal y decimal), en todos los registros.

PORT

El PORT controla la salida del puerto, este se usa en caso de que el DDR haya sido seleccionado como salida, un 1 en el PORT indica un nivel alto en el puerto como salida, un 0 indica que el pin estará en nivel bajo. Veamos algunas configuraciones de ejemplo para el PORT:

PORTB=0xFF; Todos los pines están en 1 lógico (high), PORTB=0x00; Todos los pines están en 0 lógico (low)

PORTB=0x03; //Solo los primeros dos bits del puerto están high

The Port B Data Register – PORTB

Bit	7	6	5	4	3	2	1	0	
	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
Read/Write	R/W								
Initial Value	0	0	0	0	0	0	0	0	

PIN

El PIN es un registro de lectura (notar en la imagen del registro donde dice Read/Write, todos son R), este registro nos da un 1 si el pin del microcontrolador tiene una tensión mayor a Vih (limite de tensión de 8

entrada por arriba del cual se considera 1 lógico), y un cero si el pin presenta una tensión menor a ViL (la tensión de entrada low por debajo de la cual se considera 0 lógico. En este caso el valor del PIN se le puede asignar a una variable la cual guardará el valor del mismo, al momento de ejecutar la instrucción. ej. valor=PINB; //El valor de PINB

The Port B Input Pins Address – PINB	Bit	7	6	5	4	3	2	1	0	
		PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
	Read/Write	R	R	R	R	R	R	R	R	•
	Initial Value	N/A								

Aclaración: Cuando el micro resetea, es decir empieza a funcionar, los puertos están seteados como entrada(como valor inicial). DDR valor inicial = 0