

Práctica #2

Br. Brayan J. Ceballos A.

C.I. 29.569.937

Universidad de Carabobo

Arquitectura del Computador

Profesor José Canache

05/12/2023

Práctica #2

El presente informe tiene como finalidad la presentación y descripción de los microcontroladores pertenecientes a la familia 8031, 8051 y 8052. Así como las PCL, Arduinos y Raspberry Pi. Adicionalmente se adjuntará en archivos aparte dentro del repositorio los código necesarios para la programación de alguno de estos elementos.

Microcontroladores

Estos son dispositivos electrónicos que combinan en un solo elemento llamado chip: un procesador, memoria, puertos E/S y otros periféricos. Son empleados en una gran variedad de aplicaciones, desde dispositivos domésticos hasta equipos de carácter industrial. La familia 8031, 8051 y 8052 son microcontroladores de 8 bits desarrollados por la empresa Intel en el año 1980, están basados en la arquitectura MCS-51, la cuál se encarga de separar la memoria de un programa y la de datos, así como de incluir un procesador lógico que permite las operaciones de este tipo a nivel de bit.

Microcontroladores 8031

El microcontrolador 8031 contiene la siguientes características:

- Memoria de programa de 4 KB.
- Procesador de 8 bits.
- Memoria de datos de 128 bytes.
- 4 puertos de E/S de propósito general.
- 2 temporizadores.
- 1 puerto serie.
- 1 interruptor.

Para su programación es necesario un lenguaje de alto nivel, en los cuáles se incluyen el lenguaje C o C++, y también se puede usar como opción válida algún lenguaje de ensamblador.

Para la programación de estos microcontroladores se necesita un chip, una placa de desarrollo, un programador, un cable serial o USB, y un software de desarrollo que soporte el lenguaje C y ensamblador. El costo de estos componentes pueden rondar en un rango de \$10 US/\$50 US.

A su vez se necesitan librerías específicas para el acceso a los registros y periféricos del dispositivo

Para el manejo de puertos, los puertos de E/S del microcontrolador 8031 se pueden utilizar para la lectura de datos de dispositivos externos o para escribir datos en dispositivos externos. Para esta lectura se utiliza la instrucción *in.* , para escribir un dato se utiliza la instrucción *out.* Acá se muestra un ejemplo de ello:

```
C Lectura_Escritura_de_Puertos.c X
C: > Users > braya > OneDrive > Documentos > UC > Semestre 4 > Arquitectura del Computador > Práctica 2 > C Lectura_Escritura_de_Puertos.c
1  /*Lectura y Escritura de datos en puertos en C*/
2
3  #include <stdio.h>
4  #include <stdlib.h>
5  #include <8051.h>
6
7  int main()
8  {
9      uint8_t dato;
10
11      // Lee el dato del puerto P1.0
12      dato = inp(P1.0);
13
14      // Escribe el valor 1 en el puerto P1.0
15      outp(P1.0, 1);
16
17      while (1)
18      {
19          // Instrucción a ejecutar
20      }
21
22      return 0;
23  }
24  /*Brayan Ceballos*
```

Ahora bien para el manejo de temporizadores se utilizan comandos distintos, los cuales sirven para generar eventos periódicos o para medir el tiempo. Para esta configuración se necesitan las siguientes instrucciones:

1. *set_timer_mode()*: Que establece el modo de funcionamiento del temporizador.
2. *set_timer_reload*: Que se encarga de establecer el valor de recarga del temporizador.
3. *set_timer_prescaler()*: Que inicia el preescalador del temporizador.
4. *start_timer()*: Inicia el temporizador.
5. *stop_timer()*: Detiene el temporizador.

Acá se muestra un ejemplo de configuración de un temporizador en 0, a modo CTC, con un valor de recarga de 255 y un preescalador de 1, iniciando el temporizador posteriormente y así generando un evento en cada 255 ciclos de reloj:

```
C Lectura_Escritura_de_Puertos.c  C Uso_de_temporizadores.c X
C: > Users > braya > OneDrive > Documentos > UC > Semestre 4 > Arquitectura del Computador > Práctica 2 > C Uso_de_temporizadores.c
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <8051.h>
4
5  int main()
6  {
7      // Configura el temporizador 0 en modo CTC
8      set_timer_mode(0, T0_MODE_CTC);
9
10     // Establece el valor de recarga del temporizador 0 en 255
11     set_timer_reload(0, 255);
12
13     // Establece el preescalador del temporizador 0 en 1
14     set_timer_prescaler(0, 1);
15
16     // Inicia el temporizador 0
17     start_timer(0);
18
19     while (1)
20     {
21         // Instrucción a ejecutar
22     }
23
24     return 0;
25 }
26 /*Brayan Ceballos*
```

Microprocesador 8051

Fueron desarrollados por la Intel Corporation, con diferentes finalidades, cada una de las empresas que recibieron licencia de “proveedores secundarios” desarrollaron distintos MCS-51.

La mayoría de las características del hardware se muestran mediante secuencias cortas de instrucciones, permitiendo una manipulación un poco más sencilla. Este circuito integrado más genérico de la familia 8051 fue el primero lanzado comercialmente, y posee las siguientes características:

- 4K bytes de memoria ROM.
- 128 bytes de memoria RAM.
- 4 puertos de E/S de 8 bits.
- 2 temporizadores de 16 bits.
- Una interfaz serial.
- 64K de espacio para código en memoria externa.
- 64K de espacio para datos en memoria externa.
- Procesador booleano.

- 210 ubicaciones direccionables para bits.
- Multiplicación y división en 4 ms.

Los diseños que requieren de muy poca memoria externa o de otros componentes externos utilizan terminales como puertos de E/S de propósito general. Donde cada línea opera de manera independiente como interfaz para dispositivos de un solo bit, como interruptores, LEDs, transistores, solenoides, motores y altavoces. A continuación se mencionan los puertos empleados.

- **Puerto 0:** Este es un puerto de propósito dual en las terminales 32-39 del circuito integrado 8051. Es usado como un puerto de E/S de propósito general en diseños que requieren un mínimo de componentes. Este puerto puede convertir se en un bus de direcciones y datos multiplexados en diseños más complejos que requieran de memoria externa.
- **Puerto 1:** Este es exclusivo a E/S en las terminales 1-8. Las terminales, designadas como P1.0, P1.1, P1.2, etc., están disponibles para emplearse como interfaces para dispositivos externos, en caso de ser necesario. Ninguna de las terminales del puerto 1 tiene otra función asignada, por lo tanto solo se utilizan como interfaces para dispositivos externos.
- **Puerto 2:** (terminales 21-28) es un puerto de propósito dual, que sirve como E/S de propósito general, o como el byte superior del bus de direcciones en diseños que utilizan memoria externa para código o más de 256 bytes de memoria externa para datos.
- **Puerto 3:** Este se caracteriza por tener un propósito dual en las terminales 10-17. Se puede utilizar como E/S de propósito general, pero también cumple múltiples funciones ya que sus terminales tienen un propósito alterno relacionado con las características especiales del 8051.

Memoria de datos

La memoria de datos interna del 8051 tiene un tamaño de 256 bytes, divididos en tres secciones:

- **Bando de registros (00H-1FH):** Esta sección contiene los registros de propósito general del 8051, como los registros de datos, los registros de estado y los registros de control.
- **RAM direccionable por bits (20H-2FH):** Esta parte permite acceder a los bits individuales de una palabra de memoria.

- **RAM de propósito general (30H-7FH):** Esta sección se puede utilizar para almacenar datos de uso general.

Memoria de programa

La memoria de programa interna del microcontrolador 8051 tiene un tamaño de 4096 bytes (4 KB). Esta se utiliza para el almacenamiento del código del programa que se ejecuta en el 8051. Entre sus características más importantes resaltan las siguientes:

1. Los registros y los puertos de E/S están “mapeados” a memoria, lo que quiere decir que se puede acceder a ellos como a cualquier otra localidad en memoria.
2. La pila reside dentro de la RAM interna, en lugar de en la RAM externa.

Para finalizar, se puede decir que la memoria interna del 8051 es un recurso importante para diseñadores de sistemas basados en este microcontrolador. Es importante comprender la organización de la memoria interna para aprovecharla al máximo.

Microcontroladores 8052

La familia de microcontroladores 8052 es una familia de microcontroladores de 8 bits fabricados por Intel. Los microcontroladores 8052 se caracterizan por su bajo costo, su facilidad de uso y su amplia gama de funciones.

Las principales características de los microcontroladores 8052 incluyen:

- CPU de 8 bits
- Memoria de programa de 4 KB
- Memoria de datos de 256 bytes
- Dos puertos de entrada/salida de 8 bits
- Dos temporizadores/contadores de 8 bits
- Interrupciones
- Bus de datos de 8 bits
- Bus de direcciones de 16 bits

Costo

El costo de los microcontroladores 8052 varía según el fabricante y el número de piezas que se compren. En general, el costo de un microcontrolador 8052 es de alrededor de 1 dólar estadounidense.

Requerimientos de programación

Además del chip 8052, se necesitan los siguientes elementos para programar un microcontrolador 8052:

- Un programador de microcontroladores
- Un software de programación de microcontroladores

El programador de microcontroladores es un dispositivo que se utiliza para transferir el código del programa al microcontrolador. El software de programación de microcontroladores es un programa que se utiliza para crear y transferir el código del programa al microcontrolador.

Programación en lenguaje C

Los microcontroladores 8052 se pueden programar en varios lenguajes de programación, incluyendo C, ensamblador y lenguaje máquina. El lenguaje C es un lenguaje de programación de alto nivel que es relativamente fácil de aprender y usar.

Para programar un microcontrolador 8052 en C, se necesita un compilador de C que sea compatible con los microcontroladores 8052. El compilador convertirá el código fuente de C en código máquina que se puede transferir al microcontrolador.

Cuadro comparativo de pines y características de microcontroladores

Microcontroladores	Pines	ROM	RAM	Puertos E/S	Temporizadores	Interrupciones	Frecuencia máxima
8031	40	0 KB	128 B	4 (32 Líneas)	2 de 16 bits	5 (2 externas)	12 MHz
8051	40	4 KB	128 B	4 (32 líneas)	2 de 16 bits	5 (2 externas)	12MHz
8052	40	8 KB	128 B	4 (32 líneas)	3 de 16 bits	6 (2 externas)	12MHz

PLC

Los PLC (Controladores Lógicos Programables) son dispositivos electrónicos que se utilizan para controlar sistemas automatizados. Los PLC se caracterizan por su robustez, fiabilidad y facilidad de programación.

Las principales características de los PLC incluyen:

- **Entradas y salidas digitales y analógicas:** Los PLC pueden controlar entradas y salidas digitales y analógicas. Las entradas digitales pueden utilizarse para leer señales binarias, como interruptores o sensores. Las salidas digitales pueden utilizarse para controlar dispositivos binarios, como actuadores o luces. Las entradas analógicas pueden utilizarse para leer señales analógicas, como sensores de temperatura o presión. Las salidas analógicas pueden utilizarse para controlar dispositivos analógicos, como motores o válvulas.
- **Memoria:** Los PLC tienen memoria para almacenar el código del programa y los datos de configuración.
- **Relés:** Los PLC suelen utilizar relés para controlar los dispositivos externos. Los relés son dispositivos electromecánicos que pueden conmutar circuitos eléctricos.

Costos de los PLC

El costo de los PLC varía según el fabricante, las características y la cantidad de piezas que se compran. En general, los PLC son más caros que los microcontroladores, pero ofrecen una mayor fiabilidad y facilidad de programación.

Requerimientos de programación

Para programar un PLC, se necesitan los siguientes elementos:

- Un software de programación de PLC
- Un cable de conexión entre el PLC y el ordenador

El software de programación de PLC es un programa que se utiliza para crear y transferir el código del programa al PLC. El cable de conexión se utiliza para conectar el PLC al ordenador.

Programación en lenguaje C

Los PLC se pueden programar en varios lenguajes de programación, incluyendo C, Ladder, FBD y SFC. Para programar un PLC en C, se necesita un compilador de C que sea compatible con el PLC. El compilador convertirá el código fuente de C en código máquina que se puede transferir al PLC.

Arduinos

El Arduino es una plataforma de electrónica de código abierto basada en un microcontrolador Atmel AVR. Arduino se caracteriza por su bajo costo, su facilidad de uso y su amplia comunidad de desarrolladores. Entre sus principales características se encuentran:

- **Microcontrolador Atmel AVR:** El microcontrolador Atmel AVR es un microcontrolador de 8 bits que se utiliza para controlar las entradas y salidas de Arduino.

- **Entradas y salidas digitales:** Arduino tiene una variedad de entradas y salidas digitales que se pueden utilizar para controlar dispositivos externos.
- **Entradas y salidas analógicas:** Arduino tiene una o dos entradas analógicas que se pueden utilizar para leer señales analógicas.
- **Programación en lenguaje C:** Arduino se puede programar en el lenguaje de programación C.

Costos de Arduino

El costo de Arduino varía según el modelo y el fabricante. En general, Arduino es una plataforma de electrónica de bajo costo.

Requerimientos de programación

Para programar Arduino, se necesitan los siguientes elementos:

- Un IDE de Arduino
- Un cable de conexión entre el ordenador y Arduino

El IDE de Arduino es un entorno de desarrollo integrado que se utiliza para crear y transferir el código del programa a Arduino. El cable de conexión se utiliza para conectar el ordenador a Arduino.

Para hacer más ilustrativa estas características se muestra un cuadro comparativo entre PLC y Arduino:

Características	PLC	Arduino
Rango de Aplicación	Sistemas industriales complejos	Sistemas de control de baja a media complejidad
Requisitos de progra.	Lenguajes de programación específicos de PLC	Lenguaje C
Costo	US\$ 250- US\$ 17.000.000	US\$ 32

Raspberry Pi

Raspberry Pi es una placa computadora de bajo costo desarrollada por la Fundación Raspberry Pi. Raspberry Pi se caracteriza por su versatilidad, su potencia y su facilidad de uso.

Las principales características de Raspberry Pi incluyen:

- **Procesador:** Raspberry Pi utiliza un procesador ARM de 64 bits.
- **Memoria:** Raspberry Pi tiene una memoria RAM de 1 GB a 8 GB.
- **Almacenamiento:** Raspberry Pi tiene una memoria de almacenamiento de 16 GB a 256 GB.
- **Entradas y salidas:** Raspberry Pi tiene una variedad de entradas y salidas que se pueden utilizar para conectar dispositivos externos.
- **Conectividad:** Raspberry Pi tiene una conectividad Ethernet, Wi-Fi y Bluetooth.

Costos de Raspberry Pi

El costo de Raspberry Pi varía según el modelo, en internet su precio ronda desde los US \$8-US \$ 42. En general, Raspberry Pi es una placa computadora de bajo costo.

Requerimientos de programación

Raspberry Pi se puede programar en una variedad de lenguajes de programación, incluyendo Python, C, C++ y Java.

Aplicaciones de Raspberry Pi

Raspberry Pi se puede utilizar para una variedad de aplicaciones, incluyendo:

- **Automatización del hogar:** Raspberry Pi se puede utilizar para automatizar tareas domésticas, como controlar el encendido y apagado de luces, el control de la temperatura y el acceso a la puerta.

- **Educación:** Raspberry Pi se puede utilizar como herramienta de enseñanza para aprender sobre programación, electrónica y robótica.
- **Ingeniería:** Raspberry Pi se puede utilizar para crear prototipos de dispositivos electrónicos y sistemas de control.
- **Ciencia:** Raspberry Pi se puede utilizar para realizar experimentos científicos y recopilar datos.

A manera de conclusión podemos decir que, la familia de microprocesadores 8031, 8051, 8052, Arduino, Raspberry Pi y PLC son dispositivos electrónicos que se utilizan para controlar sistemas automatizados. Cada uno de estos dispositivos tiene sus propias características y ventajas, lo que los hace adecuados para diferentes aplicaciones.

En general, la elección del dispositivo adecuado para una aplicación determinada dependerá de una serie de factores, incluyendo el alcance de la aplicación, el presupuesto y los requisitos de rendimiento.