

INTRODUCCIÓN A LOS MICROCONTROLADORES Y LABORATORIO

Bolilla I

INTRODUCCION

Sistemas de Numeración limitados en cifras.

Binario limitado en 8, decimal y hexadecimal limitados en 2.

Operaciones Lógico-aritméticas en sistemas limitados en cifras.

Definiciones y términos

Nociones de Memorias

Bolilla II

MICROCONTROLADORES

Generalidades

Descripción. Características generales

Bloques principales. Funcionamiento básico

Clasificación

PIC16F84A

Características y patillaje (Hoja de Datos)

Diagrama interno.

Tipos de fuente y oscilador

Memoria interna. Set de instrucciones

Bolilla III

PROGRAMACION

Conceptos básicos

Noción de variable

Formas de Direccionamientos. Representación

Modos de direccionamiento del PIC16F84A

Configuración del PIC16F84A. Palabra de configuración.

Herramientas a utilizar

Editor

Compilador

Simulador

Grabador (Michael Covington)

Circuitos del programador y los módulos de prueba

Bolilla IV

USO BÁSICO DEL PIC

Practicas con puertos y temporizadores

Funcionamiento y programación del TIMERO

Practicas con TMR0 como temporizado y contador

Interrupciones

Interrupciones funcionamiento.

Practicas con Interrupción por TMR0, RB0 y RB4-RB7

Multiplexión de elementos de entrada o salida

Practicas con matriz de swicht y matriz de led

Control de mecanismos

Motor paso a paso de 4 o 3 hilos.

Decodificador de posición (Shift decoder o encoder)

Bolilla V

WachtDog Timer (Vigilante)

Funcionamiento, uso y seteo.

Memoria EEPROM

Programación interna y externa. Limitaciones de uso.

Bolilla VI

COMUNICACIONES

Introducción a comunicación de microprocesadores.

Tipos de comunicación
Comunicación Paralela. Características
Unidireccional.
Practica de envío y recepción
Unidireccional con aviso de dato valido (STB y ACK).
Practica de envío y recepción
Bi-direccional con Handshaking (Entrega en mano propia).
Practica de envío y recepción
Comunicación Serie. Características
Elementos (paridad, stop y arranque)
Métodos RZ, NRZ, Ancho de pulso
Comunicación maestro-esclavo y sin jerarquía
Protocolo I2C, Rs232 y RS485
Memorias seriales.

Bolilla VII

CONVERSION A/D y D/A

Principios

Métodos

Practicas

I) INTRODUCCIÓN

1- Sistemas de Numeración

Se usan en este texto varios sistemas de numeración distintos. Los sistemas digitales usan el sistema binario, pero los operadores, programadores y usuarios están acostumbrados a usar el sistema decimal. Por lo tanto se buscan sistemas numéricos que sirvan de nexo entre el binario y el decimal.

Sistemas de numeración limitados en cifras

Los micros trabajan con un sistema numérico limitado en cifras. Usan el sistema binario limitado a 8 cifras o algún múltiplo de 4. En este texto se estudiara el sistema binario limitado a 8 cifras, el decimal y el hexa limitados en 2 cifras.

Para operar con otros conjuntos de números se debe acordar una representación con los 256 números disponibles.

Además del resultado, los micros, generan indicadores (banderas o flag) sobre que clase de resultado dio. No dependen del valor en si, sino de las características del resultado. Se generan elegidos por convención.

Si hubo desborde en la suma binaria $CY=1$

Si no hubo desborde en la suma binaria $CY=0$

Si pido uno prestado (borrow) en la resta binaria $CY=0$

Si no pido prestado uno (borrow) en la resta binaria $CY=1$

El CY según se realice una suma o resta significa el acarreo o el borrow.

Si el resultado de la operación lógico-matemática es cero $Z=1$

Si el resultado de la operación lógico-matemática no es cero $Z=0$

Si el bit 7 del resultado de la operación lógico-matemática es igual a uno $S=1$

Si el bit 7 del resultado de la operación lógico-matemática es igual a cero $S=0$

El indicador S es llamado bandera de signo.

Si hubo acarreo entre el bit 3 y el bit 4 $AC=1$ o $DC=1$

Si no hubo acarreo entre el bit 3 y el bit 4 $AC=0$ o $DC=0$

Se llama semiacarreo o acarreo digital.

Si el número de bit iguales a uno del resultado es par $P=0$

Si el número de bit iguales a uno del resultado es impar $P=1$

El indicador P es llamado bandera de paridad

Se define el **máximo valor representable (MVR)** como el valor máximo que se puede representar en un sistema limitado en cifras. Es el número donde cada cifra es el símbolo de mayor valor.

Decimal de dos cifras 99

Hexadecimal de dos cifras FF

Binario de ocho cifras 1111 1111

Las sumas cuyo resultado es mayor al MVR ponen a 1 la bandera (indicador o flag) CY .

Como condición de funcionamiento si al máximo representable le sumamos 1, dará como resultado 0 y el indicador CY valdrá 1.

Decimal $99 + 01 = 00$ y $CY=1$

Hexadecimal $FF + 01 = 00$ y $CY=1$

Binario $1111 1111 + 0000 0001 = 0000 0000$ y $CY=1$

En estos casos decimos que hubo desborde o acarreo.

Se observa también un semiacarreo AC , entre:

las unidades y decenas en el sistema decimal

la cifra de orden 0 ($\times 16^0$) y la cifra de orden 1 ($\times 16^1$) del sistema hexadecimal.

el nibble bajo y el alto en el sistema binario, o del bit 3 al bit 4

Suma en Sistemas Limitados en Cifras

Decimal

$$\begin{aligned}
 23 + 45 &= 68 \text{ y } CY=0 \text{ AC}=0 \\
 84 + 53 &= 37 \text{ y } CY=1 \text{ AC}=0 \\
 38 + 25 &= 63 \text{ y } CY=0 \text{ AC}=1 \\
 84 + 47 &= 31 \text{ y } CY=1 \text{ AC}=1 \\
 \text{Suma de máximos} \quad & 99 + 99 = 98 \text{ y } CY=1 \text{ AC}=1
 \end{aligned}$$

Hexadecimal

$$\begin{aligned}
 C5 + 2A &= EF \text{ y } CY=0 \text{ AC}=0 \\
 B8 + 19 &= D1 \text{ y } CY=0 \text{ AC}=1 \\
 72 + D8 &= 4A \text{ y } CY=1 \text{ AC}=0 \\
 4F + B9 &= 08 \text{ y } CY=1 \text{ AC}=1 \\
 \text{Suma de máximos} \quad & FF + FF = FE \text{ y } CY=1 \text{ AC}=1
 \end{aligned}$$

Binario

$$\begin{array}{r}
 1101\ 1001 \\
 +0010\ 0001 \\
 \hline
 1111\ 1010 \quad \text{y } CY=0 \text{ AC}=0
 \end{array}
 \qquad
 \begin{array}{r}
 1010\ 0111 \\
 +0100\ 1010 \\
 \hline
 1111\ 0001 \quad \text{y } CY=0 \text{ AC}=1
 \end{array}$$

$$\begin{array}{r}
 1101\ 1011 \\
 +0110\ 0001 \\
 \hline
 0011\ 1100 \quad \text{y } CY=1 \text{ AC}=0
 \end{array}
 \qquad
 \begin{array}{r}
 1001\ 0111 \\
 +1010\ 1001 \\
 \hline
 0100\ 0000 \quad \text{y } CY=1 \text{ AC}=1
 \end{array}$$

$$\begin{array}{r}
 \text{Suma de máximo} \\
 1111\ 1111 \\
 +1111\ 1111 \\
 \hline
 1111\ 1110 \quad \text{y } CY=1
 \end{array}$$

Resta en Sistemas Limitados en Cifras

Se estudia el sistema decimal limitado a dos cifras y se extenderá el razonamiento al binario y al hexadecimal.

Cuando el minuendo es menor que el sustraendo al final de la resta es necesario pedir una unidad a la cifra de la izquierda o un acarreo negativo, al cual llamamos **borrow**. El borrow se representa en la bandera CY.

Decimal

$$43 - 25 = 18 \quad \text{y } CY=1$$

La resta es una operación compleja de implementar. Lo importante es que la resta siempre reporte la diferencia entre los operandos.

La resta se define:

$$a - b = c \Leftrightarrow a = b + c$$

Se plantea desde otra forma equivalente

$$a + (-b) = c$$

Se debe buscar como representar el valor $(-b)$

$$b - b = 00$$

Entonces

$$b + (-b) = 00$$

Observación:

En sistemas limitados en cifras sumar el valor 100 en forma teórica es lo mismo que sumar 00 en forma práctica. No podemos sumar 100, pero si podemos sumar 99 + 01.

$$00 = 99 + 1$$

Considerando la observación anterior

$$b + (-b) = 99 + 1$$

Se deduce

$$-b = 99 + 1 - b$$

lo cual es

$$-b = 99 - b + 1$$

Se define el **complemento a 10** de un valor x como la diferencia de x con el **máximo representable**

más uno.

$$Ca10(x) = 99 - x + 1$$

También se define **Ca10(x)** como el valor que sumado a x da como resultado 0.

Con el Ca10 podemos transformar la resta en una suma.

$$a - b = a + Ca10(b) = c$$

Ejemplo:

$$43 - 25 = 18 \text{ y } CY = 1$$

Se convierte en suma

$$Ca10(25) = 99 - 25 + 01 = 75$$

Queda

$$43 + 75 = 18 \text{ y } CY=1$$

Se observa que en esta suma hay un acarreo, indicando que en la resta original no hubo borrow. El borrow y el acarreo son de diferente valor. Ambos se representan con CY

Se estudiara un caso donde el minuendo es menor que el sustraendo.

$$43 - 57 = \text{¿?}$$

Y como

$$Ca10(57) = 99 - 57 + 01 = 43$$

Queda

$$43 + 43 = 86 \text{ y } CY=0$$

Este resultado seria el mismo que ejecutar la resta.

$$43 - 53 = 86 \text{ y } CY = 0$$

El borrow significa que se pidió prestado uno, pero no importa de donde. Indica que el sustraendo es mayor que el minuendo.

El carry = 0 indica que en la resta original hay borrow. Y nos indica que el minuendo es menor que el sustraendo.

El valor del resultado a primera vista no representa la diferencia entre los dos valores. Pero se estudia como se relaciona con la diferencia.

$$Ca10(-86) = 99 - 86 + 01 = 14$$

Se comprueba

$$57 - 43 = 14$$

Se deduce que cuando la resta tiene borrow el resultado representa el Ca10 de la diferencia entre los operandos.

Binario

El máximo representable es en este caso 1111 1111. El valor 10 binario representa 2 en decimal.

Ejemplo

$$\text{Calculo } Ca2(1001 \ 1011)$$

En la practica no se puede representar 1 0000 0000, así que se hace la resta con MVR y después le sumamos 0000 0001. Cualquier valor será menor que el MVR.

$$\begin{array}{r}
 1111 \ 1111 \\
 - 1001 \ 1011 \\
 \hline
 0110 \ 0100
 \end{array}
 \quad \text{y} \quad
 \begin{array}{r}
 0110 \ 0100 \\
 + 0000 \ 0001 \\
 \hline
 0110 \ 0101
 \end{array}$$

Entonces

$$Ca2(1001 \ 1011) = 0110 \ 0101$$

Ejemplo de resta

$$1011 \ 0010 - 1001 \ 0110 = \text{¿?}$$

Se hace el complemento del sustraendo

$$Ca2(1001 \ 0110) = 1111 \ 1111 - 1001 \ 0110 + 0000 \ 0001 = 0110 \ 1010$$

Se suman ambos valores

$$\begin{array}{r}
 1011 \ 0010 \\
 + 0110 \ 1010 \\
 \hline
 0001 \ 1100 \quad \text{y } CY=1
 \end{array}$$

La resta no tiene borrow y su resultado es 00011100

Otro ejemplo de resta

$$0111 \ 0101 - 1001 \ 1000 = \text{¿?}$$

Se convierte en suma

$$Ca_2(1001\ 1000) = 1111\ 1111 - 1001\ 1000 + 0000\ 0001$$

$$Ca_2(1001\ 1000) = 0110\ 1000$$

$$\begin{array}{r} 0111\ 0101 \\ +0110\ 1000 \\ \hline 1101\ 1101 \quad \text{y } CY=0 \end{array}$$

La resta tiene borrow y su resultado es 11011101

Al tener borrow se debe complementar el resultado para saber la diferencia.

$$Ca_2(1101\ 1101) = 1111\ 1111 - 1101\ 1101 + 0000\ 0001$$

$$Ca_2(1101\ 1101) = 0010\ 0011 \quad \text{Siendo esta la diferencia entre los operandos.}$$

Hexadecimal

Máximo representable FF

Por lo tanto el Ca_{10} del hexadecimal corresponde al Ca_{16} del decimal.

Ejemplo

Calculo $Ca_{16}(A_2)$

$$\begin{array}{r} FF \quad \quad \quad 5D \\ -A_2 \quad \quad \quad \text{y} \quad \quad +01 \\ \hline 5D \quad \quad \quad \quad \quad \quad 5E \end{array}$$

Ejemplo de resta:

$$E4 - 3F = ?$$

$$Ca_{16}(3F) = FF - 3F + 01 = C1$$

$$\begin{array}{r} E4 \\ +C1 \\ \hline A5 \quad \text{y } CY=1 \end{array}$$

La resta no tiene borrow y su resultado es A5

Definición de Negado o Complemento a 10-1 ($Ca_{[10-1]}$)

En un sistema con límite de cifras se define el negado de x (diferente a opuesto y a inverso) como el valor que sumado a x da el máximo representable.

$$x + Ca_{[10-1]}(x) = \text{máximo representable}$$

Sería el complemento al mayor símbolo del sistema numérico.

Ejemplos:

Decimal

En este sistema sería Ca_9

$$Ca_9(12) = 99 - 12 = 87$$

$$Ca_9(83) = 99 - 83 = 17$$

Hexadecimal

En este sistema sería Ca_{15}

$$Ca_{15}(A_4) = FF - A_4 = 5B$$

$$Ca_{15}(15) = FF - 15 = EA$$

Binario

En este sistema sería Ca_1

$$Ca_1(0111\ 0101) = 1111\ 1111 - 0111\ 0101 = 1000\ 1010$$

$$Ca_1(1001\ 1100) = 1111\ 1111 - 1001\ 1100 = 0110\ 0011$$

Notas

Se deduce que

$$Ca_{10}(x) = Ca_{[10-1]}(x) + 1$$

Se observa que

$$Ca_{10}[Ca_{10}(x)] = x$$

Ejemplo decimal limitado a 2 cifras

$$\text{Ca}_{10}(73) = 99 - 73 + 1$$

$$\text{Ca}_{10}[\text{Ca}_{10}(73)] = 99 - (99 - 73 + 1) + 1$$

$$\text{Ca}_{10}[\text{Ca}_{10}(73)] = 99 - 99 + 73 - 1 + 1$$

$$\text{Ca}_{10}[\text{Ca}_{10}(73)] = 73$$

Se observa que

$$\text{Ca}[10-1]\{\text{Ca}[10-1](x)\} = x$$

Ejemplo decimal limitado a 2 cifras

$$\text{Ca}_9(38) = 99 - 38$$

$$\text{Ca}_9[\text{Ca}_9(38)] = 99 - (99 - 38)$$

$$\text{Ca}_9[\text{Ca}_9(38)] = 99 - 99 + 38$$

$$\text{Ca}_9[\text{Ca}_9(38)] = 38$$

Operaciones Lógicas Básicas

El símbolo => en una ecuación indica operación lógicas.

Operación AND (&) o (*) o (.)

$$0 \& 0 \Rightarrow 0$$

$$0 \& 1 \Rightarrow 0$$

$$1 \& 0 \Rightarrow 0$$

$$1 \& 1 \Rightarrow 1$$

Para números de varias cifras se realizan operaciones individuales entre bit de igual peso, sin generar ninguna clase de acarreo entre posiciones de diferente ponderación.

$$1110\ 1110 \ * \ 0110\ 0011 \Rightarrow 0110\ 0010$$

Operación OR (+)

$$0 + 0 \Rightarrow 0$$

$$0 + 1 \Rightarrow 1$$

$$1 + 0 \Rightarrow 1$$

$$1 + 1 \Rightarrow 1$$

Para números de varias cifras se realizan operaciones individuales entre bit de igual peso, sin generar ninguna clase de acarreo entre posiciones de diferente ponderación.

$$1011\ 0110 \ + \ 0110\ 0111 \Rightarrow 1111\ 0111$$

Operación Exor (#)

$$0 \# 0 \Rightarrow 0$$

$$0 \# 1 \Rightarrow 1$$

$$1 \# 0 \Rightarrow 1$$

$$1 \# 1 \Rightarrow 0$$

$$1001\ 0111 \ \# \ 0100\ 1011 \Rightarrow 1101\ 1100$$

Para números de varias cifras se realizan operaciones individuales entre bit de igual peso.

Operación Complemento a 1

$$\overline{0} \Rightarrow 1$$

$$\overline{1} \Rightarrow 0$$

$$\overline{1100\ 1010} = 0011\ 0101$$

Para números de varias cifras se realizan operaciones individuales entre bit de igual peso.

Hay otra operación lógica que es la **COMPARACIÓN**. En si es una resta en la cual no interesa el resultado. Se recaban datos respecto al resultado y la relación entre los operandos.

Se puede saber cual de los operandos es mayor o si son iguales.

Esta operación activa estos indicadores:

CY Si hubo borrow en la resta vale 1

Z Si el resultado es cero vale 1

AC Si hubo borrow entre los nibles bajos y los altos vale 1

P Según la paridad del resultado es su valor

S Indicador de signo, copia el valor del MSB

Después de comparar x con y o sea (x - y)

Si $x > y$ entonces $CY=1$ y $Z=0$
Si $x < y$ entonces $CY=0$ y $Z=0$
Si $x = y$ entonces $CY=1$ y $Z=1$

4-Definiciones Y Terminos

BIT- Unidad de información.

BIT- Abreviación de Dígito Binario, cada uno de los símbolos del sistema de numeración binario, pero por deformación se les llama BIT.

PALABRA- Conjunto de bits ordenados. También se le identifica con grupo de 8 bits.

BYTE- Conjunto ordenados de 8bit. (Cada bit tiene su ponderación)

NIBLE- Conjunto ordenado de 4 bit. Dos nibles concatenados forman un byte.

DATO- Es una porción de información, se usa como sinónimo de palabra. También se le llama a los bytes no usados para comandar al micro.

PROCESADOR- Dispositivo físico. Conjunto de elementos capaz de ejecutar un programa.

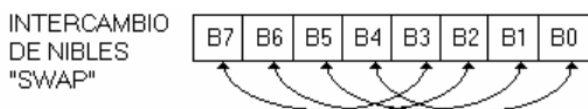
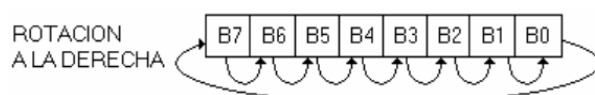
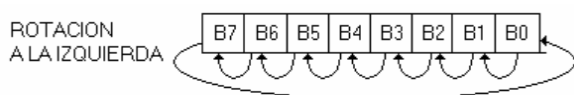
PROCESO- Secuencia de eventos para obtener un resultado o cumplir un objetivo.

µP MICROPROCESADOR- Procesador implementado en un solo chip o circuito integrado.

MICROCONTROLADOR- Conjunto de los elementos de un sistema (procesador, RAM, ROM y periféricos) integrados en un solo chip.

OPERACIÓN MATEMÁTICA- Son la suma, la resta, la multiplicación y la división. Solo los micros más potentes y dedicados a cálculos tienen la multiplicación. La división solo algunos muy escasos.

OPERACIÓN LÓGICA- Son las operaciones EX-OR, NAND, SET, RESET, AND, NOT, OR, Ca1, Ca2, ajuste decimal, cambios de posición de los bit, rotaciones, etc.



DECISIÓN- Es optar entre dos acciones a realizar dependiendo del resultado de alguna operación lógica-matemática.

BUS- Conjunto de líneas físicas que transportan bits asociados entre sí.

BUS DE DATOS- Conjunto de líneas físicas que llevan datos. Cada una de ellas se encarga de transportar un bit de la palabra.

BUS DE DIRECCIONES- Similar al anterior pero cada línea transporta un bit de la dirección a utilizar.

BUS DE CONTROL- Grupo de líneas físicas que transportan a las señales para coordinar acciones dentro y fuera del micro. Es mas difícil distinguirlas por que no siempre corren paralelas entre si.

REGISTRO- Dispositivo físico para almacenar un dato, palabra o byte.

MEMORIA- Dispositivo físico para almacenar información, conformada por registros.

BUFFER- Dispositivo físico que acopla elementos entre si. Permite amplificar en corriente las señales entre componentes.

POSICIÓN DE MEMORIA- Los registros dentro de la memoria están ordenados, por lo cual cada registro tiene su posición de memoria y esta se identifica con una dirección de memoria.

DIRECCIÓN DE MEMORIA- Dato numérico que permite definir un registro o posición de memoria dentro de la misma.

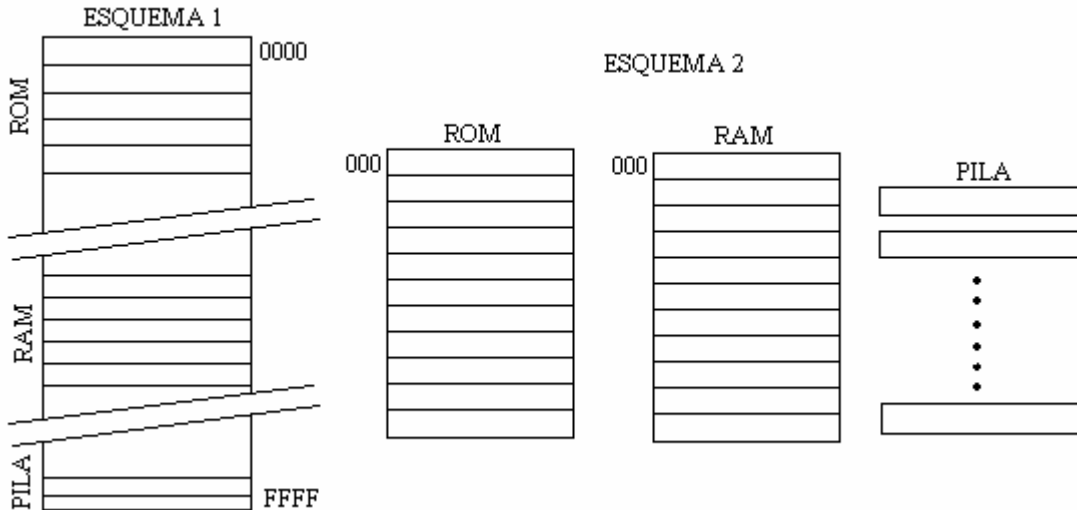
MEMORIA



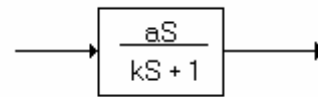
RAM- Porción de la memoria del sistema que puede ser escrita o leída. Generalmente alberga datos.

ROM- Porción de la memoria del sistema que solo puede ser leída. En general alberga comandos y es escrita antes de poner a funcionar el sistema.

MAPA DE MEMORIA- Esquema o representación de todas las posiciones de memoria que visualiza las clases de datos y su uso en el sistema.



BLOQUE- Representación gráfica de cualquier dispositivo o secuencia como un rectángulo con una entrada y una salida, identificado con la ecuación que relaciona estas entre si.



INSTRUCCIONES- Palabras que al ser leídas por el μP desde la memoria le indican acciones a realizar. Cada μP tiene su propio paquete de instrucciones.

MNEMÓNICOS- Nombre de las instrucciones usadas por el programador en la escritura el programa fuente

CÓDIGO OPERATIVO- Valor numérico de la instrucción. A este valor numérico se le llama también Código Máquina.

PROGRAMA- Secuencia de instrucciones u órdenes para conseguir un objetivo o llegar a un resultado.

SUBPROGRAMA- Pequeño programa, parte de otro más extenso o principal.

RUTINA- Similar al programa pero donde se acentúa la idea de ser cíclico o repetitivo.

SUBROUTINA- Rutina o subprograma parte de otro, el cual puede ser invocado a ejecutarse varias veces. Simplifica el programa principal y puede ser representado como un bloque.

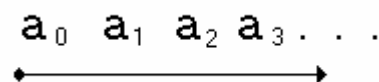
SALTO O JUMP- El micro ejecuta las instrucciones una tras otra, buscándola en posiciones de memoria consecutivas. El salto se produce cuando se busca la siguiente instrucción en una posición de memoria no consecutiva con la ejecutada anteriormente.

BIFURCACIÓN- Equivalente a decision.

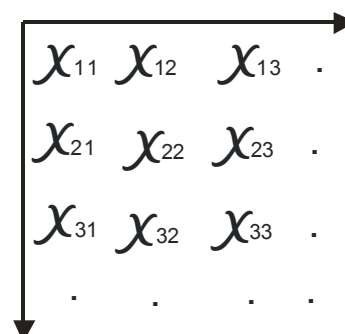
LAZO, LOOP O BUCLE- Sección de un programa que se ejecuta cíclicamente.

ARRAY o MATRIZ- Conjunto de datos los cuales tienen al menos una característica en común. Conjunto de datos ordenados en 1 o más dimensiones:

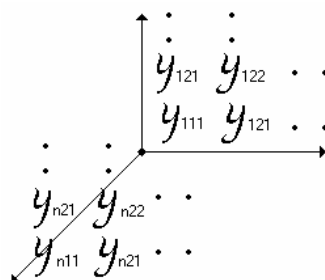
1 dimensión X_a ($a \in a$ los N) Se ordenan en base a un parámetro



2 dimensiones $X_{a,b}$ (a y $b \in a$ los N) Se ordenan en base a 2 parámetros.

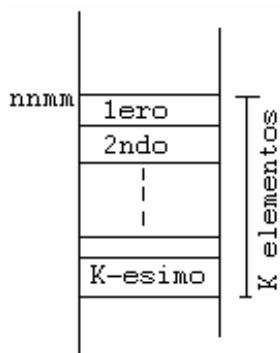


3 dimensiones $X_{a,b,c}$ (a, b y $c \in$ a los N)
 Se ordenan en base a 3 parámetros.



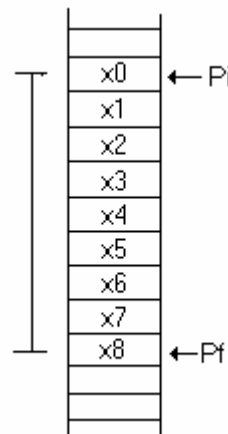
Permite tratar los datos como un objeto. Sin importar el agrupamiento que se haga de los datos estos siempre estarán colocados en posiciones sucesivas de memoria.

a11
a12
a13
a21
a22
a23
a31
a32
a33



SERIE- Grupo de datos almacenados en posiciones de memoria consecutivas. Es una matriz unidimensional. Se necesita definir la dirección de memoria del inicio de la serie y su extensión. Ejemplo inicio en nmm y K elementos de longitud.

TABLA- Grupo de datos almacenados consecutivamente. Se define indicando la posición del primer (P_i) y último dato (P_f).



PUNTERO o VECTOR- Elemento de programación que permite direccionar un dato. Registro que almacena la dirección del inicio de una matriz, o almacena la dirección del dato a ser tratado dentro de la matriz. Dentro de un lazo con solo actualizar su valor se obtiene otro elemento de la serie.

VARIABLE- Elemento de programación que puede modificar su valor. Físicamente se debe asociar en una posición de memoria RAM

ETIQUETA- Elemento de programación que permite sustituir un valor numérico por un texto o una cadena de caracteres, lo cual ayuda a ordenar el programa.

VARIABLE DE LAZO- Variable utilizada en un lazo, la cual es chequeada para determinar si se continua repitiendo o no. Cuando alcanza el valor predeterminado se sale del lazo.

PERIFÉRICO- Dispositivo físico que comunica al uP con el exterior, permitiendo que este tome o entregue datos del mismo.

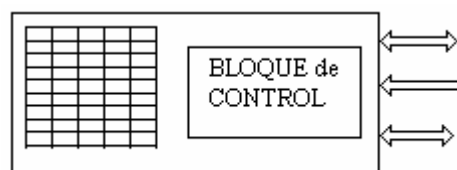
PUERTOS- Datos de dirección para la selección de periféricos. Elemento donde se conectan los periféricos.

DIAGRAMA DE FLUJO- Representación gráfica de cómo la información se modifica en un programa. Representación gráfica de un proceso, se muestran los pasos en bloques tales operaciones, decisiones, bifurcaciones y saltos.

INTERRUPCIÓN- Señal del exterior que indica al procesador la ocurrencia de un evento
MSB- Most Significant Bit - bit más significativo, de mayor peso o ponderación.
LSB- Low Significant Bit - bit menos significativo, de menor peso o ponderación.
PILA- Estructura lógica donde se guardan los datos y el último en ser colocado es el primero en ser sacado. Puede implementarse en la memoria RAM o en un banco de registros dedicados a ese fin.
LIFO- Estructura de PILA (Last IN, First OUT)
FIFO- (First IN, First OUT). Estructura donde el primer dato en entrar es el primero en salir.
PIN, PATILLA, PATA o TERMINAL- Conexión física del dispositivo con el resto del circuito. Puede unirse por soldadura o por contacto a presión.

5- Nociones de Memorias

Son dispositivos electrónicos destinados a conservar información. Están formadas por elementos o células ordenadas llamados registros y un bloque de control. En cada registro se guarda un dato. Este dato puede ser de uno o varios bits de longitud. La memoria tiene líneas de entrada y/o salida de datos (bus de datos) comunes a todos los registros y otras líneas dedicadas a seleccionar el registro con el cual trabajar (bus de direcciones). Las líneas que controlan el proceso de lectura o escritura forman el bus de control.



El bus de control se compone por el indicador de operación a realizar (lectura o escritura), el reloj maestro para sincronizar la operación y alguna señal extra para coordinar el intercambio de datos en el momento que estos son estables y válidos.

Los parámetros más importantes de una memoria son la **capacidad de almacenamiento** de información, el **tiempo de acceso** al dato seleccionado y el **número de bit** de cada dato.

El número de bits de cada dato es la longitud de la palabra que almacena la memoria. Este valor multiplicado por el número de registros internos nos dice la capacidad de almacenamiento de la memoria.

2048 registros de 8 bits cada uno, es una memoria de 16 Kbit o 2Kbyte.

16384 registros de 1 bit cada uno, es una memoria de 16Kbit.

El tiempo de acceso se mide desde que se presentan los valores en los pines de control y hasta el momento en que el dato es válido y estable.

Se clasifican según

- Uso de la memoria
- Método de acceder al registro deseado
- Método físico de almacenamiento de información

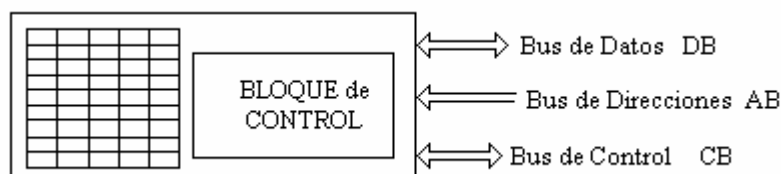
Por el uso de la memoria

Memoria de Programa ROM: Se almacena en ella las instrucciones del programa a ejecutar. Algunos micros tienen esta memoria de mayor número de registros y de bits que la memoria de datos. No pierde su información aunque pierda alimentación. Se graba antes de poner a funcionar el sistema.

Memoria de Datos RAM: Se usan para almacenar información que varía a lo largo del programa. Tiene igual número de bits que la ALU del micro. Tiene que tener suficiente velocidad de acceso como para no entorpecer al procesador.

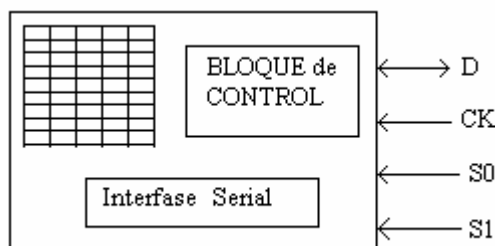
Por la forma de acceder al registro deseado

Acceso Aleatorio: Permite seleccionar el registro con el cual trabajar escribiendo el valor que identifica al registro en los pines del bus de direcciones. Si cada registro está en un estante de un armario, se trata solo de indicar la fila y la columna donde se ubica el cajón deseado. Es el formato con mayor velocidad de acceso, aunque se necesitan muchas líneas de control. El procedimiento puede diferir un poco de un modelo a otro pero básicamente es este.



En el bus de direcciones se selecciona la dirección de la posición de memoria del registro a operar. Con el bus de control se indica que operación se va a ejecutar y en qué momento hacerlo. El dato pasa por el bus de datos hacia o desde la memoria. Pueden ser tanto internas como externas al chip del micro.

Acceso Serial: Para acceder a un registro se necesita acceder antes a los anteriores. O sea para que se lea un dato es necesario leer todos los datos previos. Exteriormente tiene pocas líneas de control y es el formato de más lento acceso. Se selecciona que operación se va a ejecutar (S0-S1), el clock maestro (CK) marca el tiempo de cada bit. Cada palabra sale bit por bit por la misma línea (D). Se



cuentan los bit para saber cuando comienza y termina cada palabra, y así saber en que número de registro se está leyendo. Depende de la memoria en particular si se deben guardar los datos leídos en algún otro sitio.

Se usan comúnmente como memorias externas al chip del microcontrolador. Pueden ser memorias aleatorias con una interfase serial incluida

Método físico de almacenarla

ROM Memorias que no pierden los datos por falta de alimentación. Originalmente se grababan los datos durante el proceso de fabricación de la memoria.

PROM Se graban una vez y después no se pueden alterar. En una matriz de fusibles, grabarla es quemar los fusibles deseados.

UV-EPROM Se graban eléctricamente los datos, se puede borrar con luz ultravioleta y volver a grabar. Se borra toda la memoria junta a través de una ventana en el chip.

EEPROM Se graban y borran eléctricamente. Antes de re-escribirlas hay que borrarlas.

FLASH-EPROM Se graban eléctricamente y se pueden re-escribir sin necesidad de borrarlas previamente. Las tensiones de escritura y borrado de memoria han bajado con el avance de la tecnología. Las primeras necesitaban pulsos de 26 volts, hoy en día la tecnología permite que se graben con 3,3 volts y se sigue bajando. Consume menos y es más pequeña que las EPROM, además de programable en el circuito. Es más rápida, de mayor densidad y tolera más ciclos de escritura/borrado que la EEPROM.

Las memorias EEPROM y FLASH son muy útiles al permitir que los microcontroladores que las incorporan puedan ser reprogramados "en circuito", es decir, sin tener que sacar el circuito integrado de la tarjeta. La reprogramación del microcontrolador puede convertirse en una labor rutinaria dentro de la puesta a punto.

OTP-ROM Son memorias que se pueden grabar una sola vez. Son PROM modernas, tiene su fuerte en el bajo precio para la fabricación en grandes cantidades.

RAM Memorias que pierden los datos almacenados si se pierde la alimentación.

ESTATICAS Después de escritas si no desaparece la alimentación conservan el dato. Los registros están formados por FF o cerrojos.

DINAMICAS Es necesario ir re-escribiendo o refrescando los datos guardados para que no se le borren. La información se guarda en condensadores, los cuales pierden carga con el tiempo. Algunos modelos incluyen circuitería para refrescar los datos si no se le está utilizando. Su ventaja radica en el poco espacio que ocupan, permitiendo grandes densidades.