

La excelente accesibilidad a la programación de microcontroladores

Por Aritza Etxebarria, Ignacio Angulo y José M^a Angulo

Figura 2. Fotografía del módulo Basic Stamp 2 de Parallax, que contiene los recursos necesarios para ser programado desde un PC por el lenguaje PBASIC, que es uno de los más fáciles.

Figura 1. Fotografía de un popular microcontrolador. Se trata del PIC16F84. Tiene 18 patitas, mide unos 3 centímetros y cuesta menos de 6 euros. Muchos aparatos que manejan a diario incorporan un μ C para gobernar su funcionamiento.

En ocasiones, consideramos que ciertas actividades son muy complejas y por eso las evitamos. Suele suceder con la programación de microcontroladores. Y nada más lejos de la realidad. Existen muchas técnicas de programación muy diferentes, con niveles de dificultad muy distintos a su vez. Desde programar a golpe de ratón y de forma gráfica, hasta programar en el lenguaje más complejo, el ensamblador, que es el lenguaje que usa internamente la máquina. Todas estas técnicas tienen sus ventajas y desventajas, que trataremos de desvelar en el siguiente artículo.

Con este artículo, queremos dirigirnos muy especialmente a dos importantes colectivos.

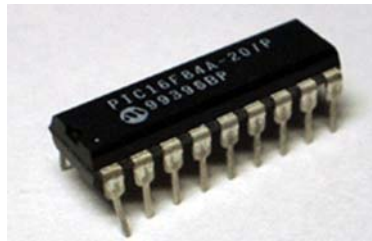
El primero está formado por todos los usuarios del PC que lo emplean como una herramienta habitual para su trabajo y que incluso confeccionan programas de gestión. En este grupo hay cada vez más curiosos que les gustaría programar computadores enanos, como son los microcontroladores, para gobernar dispositivos y aparatos que utilizan.

El segundo grupo está formado por los "últimos electrónicos", que aún siguen diseñando con Electrónica cableada y no han dado el paso a sustituir todos los componentes activos y pasivos que conectan por un diminuto computador programable. También se encuentran en este colectivo muchos avezados profesionales que no han saltado del soldador a la tecla, educadores que vislumbran magníficas posibilidades para sus alumnos y estudiantes que aún desconocen las posibilidades de esta nueva estrategia de diseño y sus pequeñas dificultades.

Si se acierta en la selección del lenguaje de programación, es tan fácil diseñar sistemas basados en microcontrolador como manejar un moderno teléfono móvil o un sofisticado electrodoméstico.

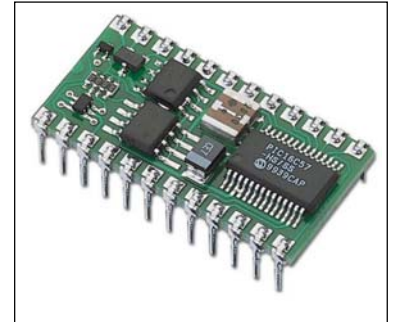
¿Qué es un microcontrolador?

"Un microcontrolador es un computador construido sobre el chip que contiene un circuito integrado". Está fabricado con millones de diminutos transistores, resistencias y otros componentes electrónicos interconectados entre sí. Cabe en la punta del dedo, pesa unos pocos gramos y cuesta unos pocos euros (ver fig.1).



Hoy en día, estamos rodeados de microcontroladores; están en el coche, en la cocina, en el ascensor, en el teléfono móvil, en el ordenador, etc.. Millones de microcontroladores realizan pequeñas tareas programadas para controlar el programa del microondas, calcular el consumo de nuestro vehículo, programar la calefacción de nuestro hogar, etc. Regular el comportamiento de los dispositivos y aparatos que nos rodean mediante un computador supone una precisión, seguridad, velocidad y potencia insospechadas. Además con un computador tan pequeño y tan barato apenas se modifica el tamaño o el coste del producto en el que se incorpora.

Pero la operatividad de un computador depende de las instrucciones que ejecuta y en consecuencia del programa. Por tanto, es el programa el que dota de utilidad al microcontrolador. Es el programa quien define las características del dispositivo gobernado, o controlado. Pero, ¿cómo se programa el microcontrolador? De la correcta contestación a esta pregunta depende la decisión que muchos interesados pueden tomar.



Como hay muchos modelos de microcontroladores también existen muchos lenguajes y formas de programarlos. En la figura 2 se muestra un pequeño circuito que incluye entre sus componentes a un microcontrolador PIC similar al mostrado en la figura 1, pero con esta implementación se ha intentado facilitar su programación. El módulo Basic Stamp 2 mostrado en la figura 2 soporta recursos lógicos y físicos para poder ser programado por un lenguaje muy sencillo como es el PBASIC.

Iniciaremos el recorrido de la programación de microcontroladores, presentando la forma más básica: La programación gráfica, a golpe de ratón.

Programando a golpe de ratón

El módulo microcontrolador Basic Stamp, de Parallax, está orientado, en gran medida, a la iniciación en la programación de microcontroladores.

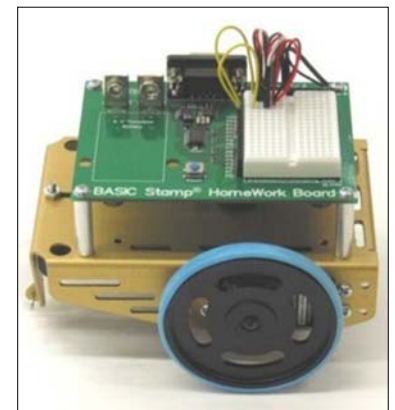


Figura 3.- Fotografía del robot Home Boe-Bot basado en un módulo microcontrolador BS2-IC y que puede programarse de forma gráfica y mediante el lenguaje PBASIC.

dores. Aunque también hay modelos orientados al control industrial, el más conocido y comercializado, el Basic Stamp 2, está orientado al sector educativo. Por eso una de las propuestas más fascinantes de aplicación de este elemento es la construcción de un robot para la posterior programación del mismo con el objetivo que realice diversas tareas. El robot Home Boe-Bot que se muestra en la fotografía de la figura 3, está basado en el Basic Stamp 2, y lo pueden montar y programar los alumnos de la ESO. En este robot puede emplearse un método de programación gráfico e intuitivo que simplifica el primer contacto con el software.

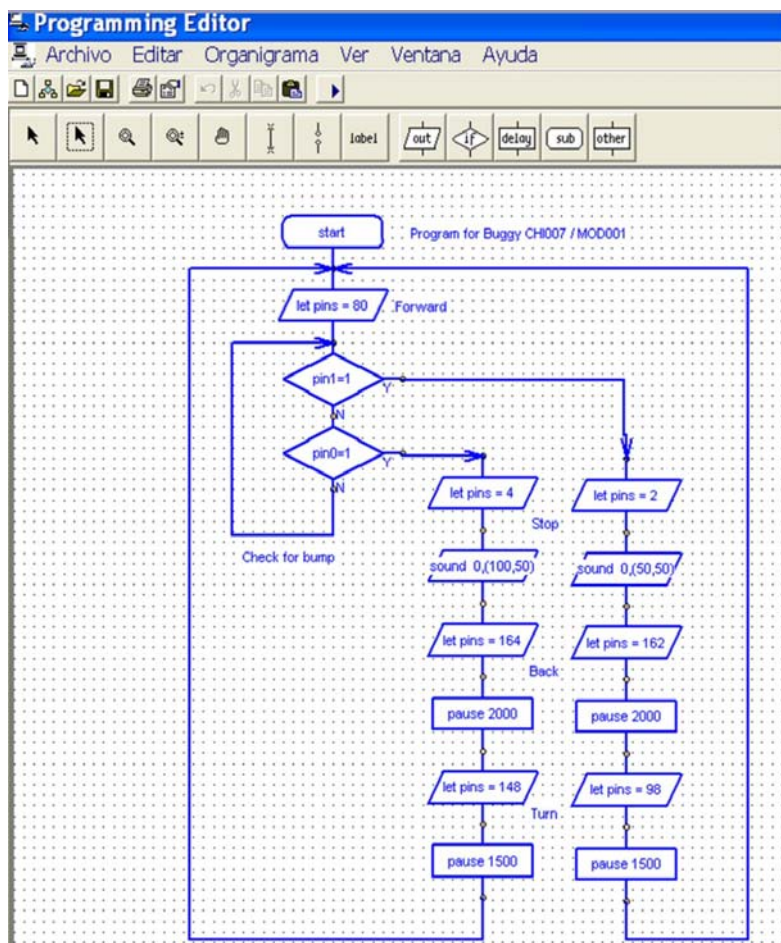
Parallax ha diseñado un software que podemos descargar gratuitamente de su página web

(www.parallax.com) denominado GUI Bot, que nos permite programar un robot. Puede descargarse libremente este programa, así como un tutorial en castellano, desde la página www.microcontroladores.com que corresponde a la del distribuidor en España Ingeniería de Microsistemas Programados S.L. En la ventana mostrada en la figura 4 se muestran las posibilidades de programación del robot. Se puede controlar sus movimientos en todos los sentidos y luego irlos ordenando para configurar el programa de actuación de la máquina gobernada por el microcontrolador.

En la figura 4 se ha programado el robot para que avance durante 2 segundos, gire a la derecha durante segundo y medio, y por último, se



Figura 4.- Ventana de trabajo del programa GUI Bot para programar todas las acciones y movimientos del robot Home Boe Bot.



detenga durante un segundo. Para programarlo, simplemente se arrastran las acciones mostradas a la izquierda a la lista de acciones a ejecutar del centro, y se coloca el tiempo durante el que se desea que se realice cada una de las acciones programadas.

Otro ejemplo de programación gráfica, lo encontramos en el microprocesador Picaxe. En la figura 5, podemos observar un programa realizado de forma gráfica mediante organigramas y en el Listado 1 tenemos el programa equivalente a dicho organigrama.

```

main:
label_6:  let pins = 80
label_D:  if pin1=1 then label_2A
         if pin0=1 then label_23
         goto label_D

label_23: let pins = 4
         sound 0,(100,50)
         let pins = 164
         pause 2000
         let pins = 148
         pause 1500
         goto label_6

label_2A: let pins = 2
         sound 0,(50,50)
         let pins = 162
         pause 2000
         let pins = 98
         pause 1500
         goto label_6
    
```

Listado 1. Programa en lenguaje BASIC.

Figura 5. Un ejemplo de programación gráfica con el sistema microprocesador Picaxe.

Es fácil comprobar, que la ventaja de este método de programación es su sencillez. Pero en contrapartida, tiene la desventaja de ser un sistema poco flexible.

El lenguaje PBASIC

El siguiente nivel de programación, es el que corresponde al lenguaje PBASIC. El lenguaje PBASIC es un lenguaje de alto nivel basado en el clásico BASIC. Se utiliza en la familia de microcontroladores Basic Stamp de la empresa Parallax. Es un lenguaje muy sencillo, de hecho, existen autores que califican este lenguaje como "el más fácil del mundo", equiparable a escribir en inglés. Las instrucciones coinciden con el nombre en inglés de la operación que llevan a cabo.

Por otro lado, y para facilitar la programación, Parallax, distribuye gratuitamente a través de su página web, el programa Basic Stamp Editor, que facilita la edición de los programas en el PC gracias a la utilización de resaltado de sintaxis automático y una ayuda excelente con multitud de ejemplos.

Nada mejor que un ejemplo para entender la facilidad de uso de este lenguaje.

En un microcontrolador, las entradas y las salidas, se corresponden con ciertas patitas del circuito

integrado. Y éste, interactúa con el medio, a través de estas patitas, sacando por ellas un determinado voltaje o leyendo el voltaje que se aplica desde el exterior a una determinada patita.

El Basic Stamp 2 dispone de 16 patitas, numeradas de 0 a 15, para poder realizar dichas operaciones de entrada y salida. Supongamos que hay un LED conectado en una de las patitas del microcontrolador, por ejemplo, la número 4.

Si queremos hacer un programa en PBASIC que haga parpadear el LED cada medio segundo, una posible solución sería la de la figura 6, en la que se muestra la ventana del editor BASIC STAMP utilizado en la confección de los programas en el PC.

Las 3 primeras líneas del programa PBASIC mostrado en la figura 6, son comentarios aclaratorios.

La línea 5 es una etiqueta para marcar una posición a la que queremos ir más adelante.

La línea 6 hace que el microcontrolador saque por la patita 4 un nivel lógico alto (high) y así el LED recibe +5 voltios para poder encenderse.

Después, gracias a la línea 7, el programa hace una pausa (pause) durante 500 milisegundos (medio segundo).

En la línea 8 se saca un nivel lógico bajo (low) por la patita 4 y con la línea nueve se vuelve a esperar 500 milisegundos.

Por último, en la línea 10 se indica (GOTO) que la siguiente instrucción a ejecutar sea la que lleva la etiqueta salto, es decir, que vuelva a la línea 5 para estar repitiendo el programa de forma indefinida.

El PBASIC es el nivel de programación idóneo para empezar partiendo de cero. Es un nivel que permite dar solución a la mayoría de los programas. Pero, no permite hacer ciertas tareas que a veces son necesarias, como es el caso de la atención a las interrupciones.

Programación orientada a objetos. Programando en Java

Hasta el lenguaje más moderno de programación, puede utilizarse a la hora de programar microcontroladores. Parallax dispone de un módulo microcontrolador, el Javelinstamp, programable en Java, lenguaje de última generación orientado a objetos.

La programación orientada a objetos es una técnica de programación moderna que actualmente tiene mucho éxito en la comunidad de programadores. Entre otras ventajas, esta técnica de programación permite la reutilización del código, al abstraer los problemas en entidades y objetos distintos. Así no hay que reinventar la rueda cada vez que afrontemos un nuevo proyecto.

Un ejemplo de programación en JAVA, podría ser el siguiente, donde se realiza una cuenta atrás.

```
import stamp.core.*
public class CuentaAtras {
    static void main() {
        System.out.println("Comienza la cuenta atrás");
        CPU.delay(10000);
        for(myVar = 10; myVar >= 1; myVar--){
            CPU.delay(2000);
            System.out.println(myVar);
        }
        System.out.println("¡Paseo!");
    }
}
```

Este lenguaje tiene muchas similitudes con el lenguaje C.

Un nivel intermedio. El lenguaje C.

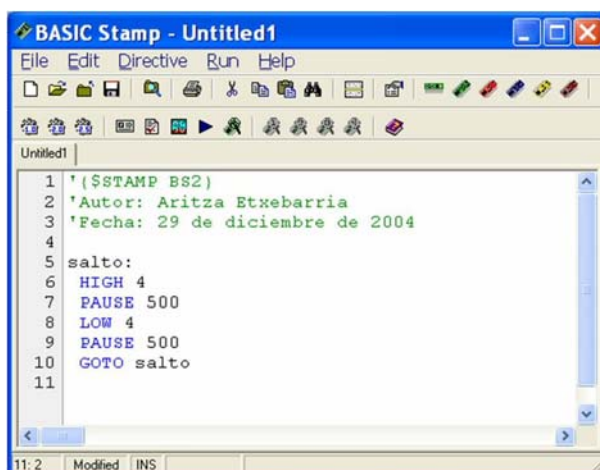
C es un lenguaje de programación de nivel medio ya que combina los elementos del lenguaje de alto nivel, como por ejemplo el lenguaje PBASIC analizado con anterioridad, con la funcionalidad del ensamblador, que es el lenguaje de bajo nivel o nivel máquina.

Naturalmente, es un poco más complicado que los lenguajes de alto nivel, pero aporta una mayor flexibilidad y potencia.

Se propone un programa que suma 2 unidades al valor que se introduce por las patitas de la puerta

Listado 2. Un ejemplo en Java

Figura 6.- Ventana del editor en la que se muestra un programa en PBASIC que hace parpadear al LED conectado en la patita 4 cada medio segundo.



A de un microcontrolador PIC 16F84, y muestra el resultado por la las patitas de la puerta B del mismo. En la puerta A podrían existir conectados unos interruptores (entradas binarias) y por la puerta B unos LED (salidas binarias) para poder comprobar el resultado.

duda, el lenguaje con el que se puede depurar hasta el máximo el rendimiento de nuestro programa. La mayoría de las soluciones industriales y de fabricación masiva utilizan este lenguaje como solución definitiva de los proyectos.

Al lenguaje ensamblador se le

```
#include <16f84.h> /* Dispositivo a utilizar */
#bytepuertaa=05 /* Posición de la puerta A */
#bytepuertab=06 /* Posición de la puerta B */
void main(void)
{
    int valor; /* Variable que contiene el valor
leído */
    int *puertaa = bytepuertaa; /*Puerta A del
microcontrolador */
    int *puertab = bytepuertab; /*Puerta B del
microcontrolador */
    set_tris_a(0xff) ; /* Declara la puerta A como entrada */
    set_tris_b(0x00) ; /* Declara la puerta B como salida */
    do {
        valor = *puertaa; /* Valor de Puerta A -> valor */
        valor = valor + 2; /* Suma 2 a valor */
        *puertab = valor; /* Muestra el valor en la puerta
B */
    }
    while (TRUE); /* Repite el programa de forma indefinida
*/
}
```

Como se observa en el programa en lenguaje C, en primer lugar se declaran variables y puertas de entrada y salida, luego se opera sumando dos unidades y, finalmente, se repite indefinidamente el programa, para que el valor binario introducido por los interruptores de la Puerta A de entrada se le sumen dos unidades y el resultado se refleje en el valor binario correspondiente al encendido de los LED.

El lenguaje Ensamblador

Este es el lenguaje mas complicado. Pero a la vez, es el lenguaje más específicamente preparado para trabajar con el microcontrolador. Por tanto, con el lenguaje ensamblador podremos hacer uso de todas las características y recursos del microcontrolador. Así mismo, si el tiempo es un factor crítico a la hora de la solución de un proyecto, es sin

suele llamar lenguaje de bajo nivel o lenguaje máquina. Su manejo es muy difícil porque cada instrucción realiza una operación elemental sobre los elementos del microcontrolador, por lo cual es necesario conocer en profundidad la arquitectura y el funcionamiento de la máquina para poder aplicar correctamente el lenguaje ensamblador. A continuación vamos a programar en ensamblador el ejemplo de programación anterior en lenguaje C, que queda tal y como se muestra en el listado nº 4.

El código utilizado en ensamblador es mucho menos amigable que el empleado en C, y todavía mucho menos que en PBASIC. En la Tabla 1 se muestra una comparativa entre los lenguajes descritos con referencia al nivel de dificultad y a la potencia.

Conclusiones

Como hemos podido observar, existe un amplísimo abanico de

```
LIST P=16F84 ;Modelo de microcontrolador a emplear
RADIX HEX ; Medidas en hexadecimal.
PUERTAA EQU 0x05 ;Dirección de la puerta A
PUERTAB EQU 0x06 ;Dirección de la puerta B
ESTADO EQU 0x03 ;Dirección de la palabra de estado
W EQU 0 ;Dirección del registro acumulador
ORG 0 ;Dirección de inicio del programa
bsf ESTADO,5 ;Accedemos al banco de
configuración
movlw 0xff ;para configurar la puerta A
movwf PUERTAA ;como entrada
movlw 0x00 ; Y la puerta B
movwf PUERTAB ;como salida
bcf ESTADO,5 ;Volvemos al banco de datos
puertas
inicio movf PUERTAA,W ;El contenido de A a W
addlw 2 ;Sumamos 2 al contenido de W
movwf PUERTAB ;Lo mostramos por la puerta B
goto inicio ;Bucle infinito
end ;Fin del programa
```

posibilidades a la hora de programar un µC. Desde el método de programación más tradicional al más moderno orientado a objetos. La elección, naturalmente, corresponde al lector en función de sus conocimientos y los objetivos de su proyecto. Aprender a programar microcontroladores, sólo se consigue programando proyectos reales. Por eso animamos los interesados a que se sumen en las variadas formas de programación para que busquen el campo de aplicación de cada una de ellas y la que mejor se adapte a sus necesidades. o

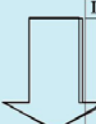
Listab 4. Un ejemplo en ensamblador

Bibliografía

- 1- "Diseño práctico con µCs ", Angulo, Romero, Angulo, Thomson, 2004.
- 2.-"µCs PIC. Diseño práctico de aplicaciones. 1ª Parte", Angulo, Angulo, Editorial Mc Graw Hill, 2003.
- 3.- "Laboratorio de Prácticas de Microelectrónica", Volúmenes 1 y 2, Angulo, Editorial Mc Graw-Hill
4. -www.parallax.com
5. -www.didatec.es
- 6.- www.microcontroladores.com

Listab 3. Un ejemplo en C

Tabla 1. Comparativa entre dificultad y potencia de los lenguajes más comúnmente utilizados.

	Dificultad	Lenguaje de programación	Potencia
	Menor	Programación grafica: GUI Bot, Picaxe, ...	Menor
		PBasic	
		Java	
	Mayor	Lenguaje C Ensamblador	Mayor