

## CRITERIOS DE SELECCIÓN DE LAS HERRAMIENTAS DE DESARROLLO PARA MICROCONTROLADORES PIC

### Selection criteria of development Tools for PIC microcontrollers

#### RESUMEN

Los altos niveles de integración de los circuitos integrados y la reducción del costo de adquisición de dichos dispositivos, han traído consigo un aumento en uso de estos como parte de todo tipo de quipos electrónicos que van desde electrodomésticos hasta equipo médico.

Existe una importante variedad de fabricantes de microcontroladores dentro de los que se destacan Atmel, Motorola, Texas Instruments, Toshiba y Microship; cada uno trayendo ventajas que deben ser discutidas de acuerdo a la aplicación a desarrollar. Los autores de este trabajo ha seleccionado la tecnología desarrollada por Microchip, más conocida como PIC, debido a su popularidad en muchos desarrollos académicos y comerciales en la búsqueda de las alternativas existentes para su programación principalmente en el entorno de Windows.

Como parte del trabajo de este artículo, se han desarrollado una serie de simples programas contadores desde cero hasta nueve implementados en diferentes niveles de programación.

**PALABRAS CLAVE:** microchip, microcontrolador, ensamblador, lenguaje, programa, desarrollo, herramientas, contador.

#### ABSTRACT

High levels of integration of integrated circuits and the low acquisition cost of these devices, have brought an increased use of these as part of all types of electronic devices ranging from appliances to medical equipment.

There is an important variety of manufacturers of microcontrollers within those highlighted Atmel, Motorola, Texas Instruments, Toshiba and microchip, each bringing benefits to be discussed according to the application being developed. The authors of this paper has selected Microchip technology, better known as PIC, because of its popularity in many academic and commercial developments in the search for alternatives for their programming primarily in the Windows environment for this paper.

As part of the work of this article, a series of simple counting programs have been developed in different programming levels.

**KEYWORDS:** microchip, microcontroller, assembler, language, program development tools, counter.

#### 1. INTRODUCCIÓN

Durante las últimas décadas la electrónica digital ha ocupado un lugar muy importante en el desarrollo tecnológico del mundo moderno. Esta ha permitido el crecimiento desbordado de las comunicaciones, lo que en consecuencia ha traído la desaparición de las fronteras físicas y temporales entre los seres humanos. Las tecnologías de telefonía móvil han estado desde los inicios de los años 80, pero solo su popularización se dio a fines de los años 90 gracias al mejoramiento de los niveles de integración, la posibilidad del tratamiento digital de la señal, su reducido costo y bajo consumo de energía. El teléfono como tal, es uno de los mejores reflejos del alcance logrado mediante las tecnologías microcontroladas.

Pese a las grandes ventajas que traen los microcontroladores, una de sus más importantes desventajas, para el lado diseñador, es la dificultad que presentan en el momento de requerir su programación y su amplio grupo de instrucciones que dependen de cada microcontrolador aunque compartan algunas en común. El lenguaje ensamblador es, por naturaleza, el estándar para la implementación de rutinas dentro de los microcontroladores PIC, pero se han desarrollado una importante variedad de compiladores que facilitan su desarrollo pero traen otras dificultades que deben ser discutidas.

Los microcontroladores PIC son fabricados por Microchip Technology, derivado del microcontrolador PIC1640 originalmente desarrollado por la división de

#### JIMMY ALEXANDER CORTÉS OSORIO

Ingeniero Electricista, MSc  
Profesor Asociado  
Universidad Tecnológica de Pereira  
jacoper@utp.edu.co

#### JAIRO ALBERTO MENDOZA VARGAS

Ingeniero Electricista, MSc  
Profesor Asociado  
Universidad Tecnológica de Pereira  
jam@utp.edu.co

#### JOSÉ A. MURIEL ESCOBAR

Ingeniero Mecánico, MSc  
Profesor SENA Dosquebradas  
jamuriel@sena.edu.co

microelectrónica de General Instruments. La palabra PIC hace referencia a "Controlador de interfaz de periféricos" ("Peripheral Interface Controller").

## 2. DEFINICIONES

### 2.1 MICROPROCESADOR

Un diseño clásico requería de muchos componentes sueltos tales como transistores, resistencias y compuertas, que requerían múltiples cálculos matemáticos cada vez que se deseaba realizar un circuito para una función específica, y no era fácil que este mismo resultara útil para otro propósito sin más adecuaciones electrónicas.

En el año 1971 apareció el primer circuito integrado denominado microprocesador desarrollado por la compañía Intel lo que generó, desde ese momento, un cambio drástico en las técnicas de diseño de la mayoría de los equipos electrónicos. A partir de dicho momento, fue posible construir un sistemas electrónicos con la ayuda de un microprocesador más unos cuantos componentes de circuito adicionales, apoyándose en la capacidad de programación del circuito integrado a través de algunas rutinas de software. Desde este momento, los diseños electrónicos fueron mucho más pequeños y simplificados y se podían realizar más tareas en menor tiempo con equipos considerablemente más pequeños.

### 2.2 EL MICROCONTROLADOR

Posteriormente aparece una nueva tecnología en respuesta natural al ya posicionado microprocesador, llamada microcontrolador que simplifica aún más el diseño electrónico, ya que el un solo circuito integrado se logra incluir el microprocesador, la memoria RAM y los puertos de entrada y salida. La figura 1 muestra el microprocesador 4004 de Intel<sup>1</sup> es cual es el primer microprocesador de la historia.

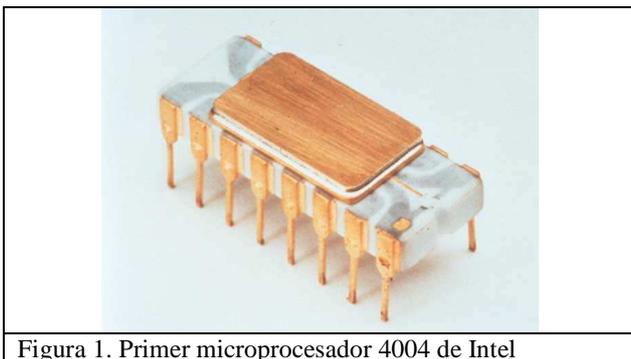


Figura 1. Primer microprocesador 4004 de Intel

Un microcontrolador es un circuito integrado programable para una función específica que incluye en su interior las tres unidades funcionales de una computadora tradicional: CPU, memoria y puertos de entrada y salida, es decir, se trata de una computadora completa en un solo circuito integrado el cual se emplea para como un dispositivo para controlar el funcionamiento de una tarea específica, aunque sus características son limitadas si se comparan con las de un computador personal de escritorio.

Actualmente, un hogar promedio puede llegar a tener muy seguramente distribuido entre los electrodomésticos de la casa más de una decena de microcontroladores, los cuales pueden encontrarse en el televisor, el horno microondas, el control remoto de cada uno de los equipos, la lavadora y por su puesto, los teléfonos tanto fijos como móviles.

### 3.2 EL MICROCONTROLADOR PIC 16F84

En la búsqueda de las opciones de programación de microcontroladores se ha seleccionado el PIC 16F84 como uno de los integrados más populares de la familia 16 por su precio y sus básicas características.[10][13]

#### 3.2.1 Características Generales del microcontrolador PIC 16F84

Repertorio de 35 Instrucciones.

Todas las instrucciones se ejecutan en un solo ciclo excepto las de salto que necesitan dos.

Versiones de 4 MHz (PIC16F84-04) y 10 MHz (PIC16F84-10).

Memoria de programa Flash de 1 K x 14 bits.

Memoria de datos RAM de 68 bytes.

Memoria de datos EEPROM de 64 bytes.

15 registros de funciones especiales.

Pila con 8 niveles de profundidad.

Modos de direccionamiento directo, indirecto y relativo.

4 fuentes de interrupciones:

A través del pin RB0/INT.

Desbordamiento del temporizador TMR0.

Interrupción por cambio de estado de los pines 4:7 del Puerto B.

Completada la escritura de la memoria EEPROM.

1.000.000 de ciclos de borrado/escritura de la memoria EEPROM.

40 años de retención de la memoria EEPROM.

13 pines de E/S con control individual de dirección.

Contador/Temporizador TMR0 de 8 bits con divisor programable.

Power-on Reset (POR).

Power-up Timer (PWRT).

Oscillator Start-up Timer (OST).

Watchdog Timer (WDT).

Protección de código.

Modo de bajo consumo SLEEP.

Puede operar bajo 4 modos diferentes de oscilador.

Programación en serie a través de dos pines.

<sup>1</sup> Imagen tomada de <http://commons.wikimedia.org/wiki/File:Intel4004.jpeg> la cual posee GNU Free Documentation License

Tecnología de baja potencia y alta velocidad CMOS Flash/EEPROM.

Rango de alimentación: 2.0 a 6.0 V.

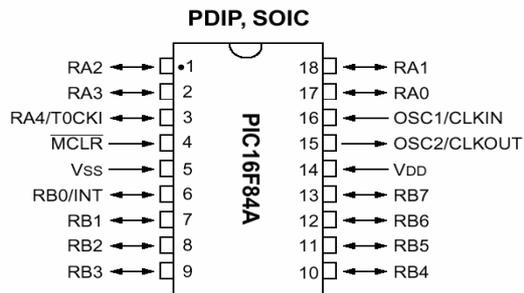


Figura 2. Pines del PIC16F84

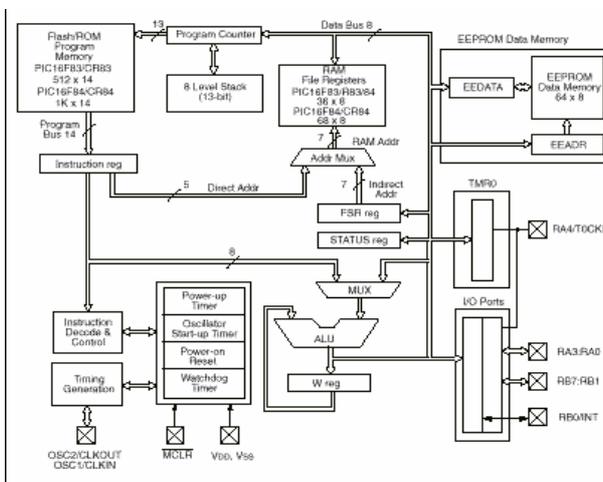


Figura 3. Diagrama de bloques del PIC16F84

Las figuras 2 y 3 muestran los diagramas de pines y bloques del microcontrolador 16F84.

## 5. Herramientas de desarrollo para la programación de PIC

Existe un amplio rango de herramientas para la programación de los microcontroladores PIC. Estas van desde las más puras que recurren al lenguaje de máquina, hasta las herramientas iconográficas que hacen uso de la representación de las instrucciones mediante íconos o diagramas.

### 5.2 Herramientas de lenguaje de bajo nivel

#### 5.2.1 Lenguaje de máquina

Este sería el modelo ideal de programación de los microcontroladores pero dista mucho de ser la forma en que los seres humanos razonan ya que hace uso de ceros y unos para la implementación del programa. Aunque los programas son de tamaño y velocidad óptima, su nivel de programación lo hace poco intuitivo y consume demasiado tiempo para un desarrollo, por básico que sea este.

#### 5.2.2 Lenguaje ensamblador

La dificultad presentada por el lenguaje de máquina que resultaba poco intuitiva, llevó a la aparición del grupo de instrucciones nemotécnicas que facilitaban la recordación de las tareas requeridas tales como sumar, restar, mover etc.

MPLAB v8.4 es el ambiente de desarrollo integrado (IDE) gratuito para escribir código en lenguaje ensamblador para los microcontroladores PIC. MPLAB posee las herramientas necesarias para la realización de proyectos, ya que no solo posee un editor de texto para código, sino que también cuenta con un simulador en el que se puede ejecutar el programa por pasos.

MPLAB resulta ser una de las herramientas más poderosas para el desarrollo de aplicaciones en ensamblador para PIC ya que posee compatibilidad con un amplio rango de aplicaciones tales como HI-TECH, IAR, Byte Craft, B. Knudsen, CCS, Micrium, microEngineering Labs, Proteus Labcenter y MATLAB.[1][2][10][11][12]

El ensamblador que utiliza MPLAB, programa entregado por Microchip para la programación de sus microcontroladores, es *MPASM*. Los elementos básicos del lenguaje ensamblador son las etiquetas, las instrucciones, los operandos, directivas y comentarios.

Como ejemplo, se ha desarrollado el contador con MPLAB v7.4 en ensamblador el cual es simulado en Proteus 7.4 y probado el sistema microcontrolado de desarrollo EasyPic<sup>2</sup> de la compañía MikroElektronika que se aprecia en la figura 4.[3]

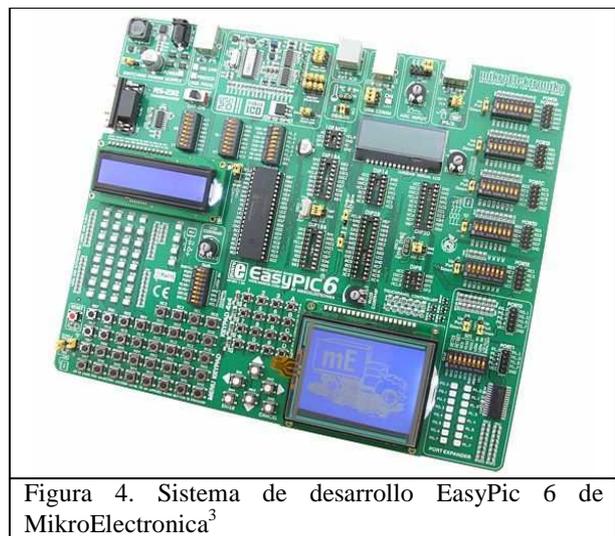


Figura 4. Sistema de desarrollo EasyPic 6 de MikroElektronika<sup>3</sup>

<sup>2</sup> <http://www.mikroe.com/en/tools/easypic6/>

<sup>3</sup> Fotografía tomada de [http://www.mikroe.com/en/Tools/easypic6/gallery/easypic6\\_550\\_1.jpg](http://www.mikroe.com/en/Tools/easypic6/gallery/easypic6_550_1.jpg)

El programa desarrollado simplemente cuenta desde 0 hasta 9 en pasos de un segundo indefinidamente. Este es un ejemplo funcional que presenta la estructura básica del programa.

```

LIST    p=16F84      ; PIC16F84 Este es el PIC usado
#include "P16F84.INC" ; Incluye el archivo

CONTADOR    equ    0ch
REG1        equ    0dh
REG2        equ    0eh
REG3        equ    0fh
COMPARA     equ    10h

reset       org     0          ; Iniciar.

           clrwf        ; Hacer 0 el registro de trabajo.
           movwf       PORTB ; Carga en valor w=0 en el puerto B
           ; para que inicie en Ceros
           goto        configports ; Ir a configports.

;-----inicio configura puertos-----

configports bcf     STATUS,RPO ; Se pasa al banco 1
           clrwf        ; Hacer 0 el registro de trabajo.
           movwf       TRISB ; Carga w=0 en TrisB lo que le hace
           ;de salida
           bcf     STATUS,RPO

;-----fin configura puertos-----

;-----inicio cuerpo programa-----

REINICIO   movlw     0h          ; Se reinicia la cuenta
           movwf     CONTADOR ; Se reinicia el contador
lazo       movf     CONTADOR,0 ; Inicia el contador
           movwf     COMPARA ; guara el valor de w para
           ;compararlo y reiniciar
           call     DISPLAY7 ; Carga w con el numero en el
           ;Display 7 segmentos de la tabla
           movwf     PORTB ; Coloca w en el portB / PortB=w
           call     espere_1s ; Hace una espera de 1 segundo
           incf     CONTADOR,1
           MOVWF    COMPARA,0 ; Regresa el valor de la cuenta a w
           XORLW    b'1001' ; Verifica si el contador llegó a 9 (b'1001')
           BTFSZ   STATUS,Z ; Si llega a 9 se salta la siguiente línea
           goto    lazo ; Vuelve a lazo si no llegó a 9
           goto    REINICIO ; si llegó a 9 lo atiende en reini

;-----fin cuerpo programa-----

;-----inicio rutina 1 segundo-----

espere_1s  ;999990 ciclos
           movlw     0x07
           movwf     REG1
           movlw     0x2F
           movwf     REG2
           movlw     0x03
           movwf     REG3
SEGUNDO_0
           decfsz   REG1, f
           goto    $+2
           decfsz   REG2, f
           goto    $+2
           decfsz   REG3, f
           goto    SEGUNDO_0

           ;6 ciclos
           goto    $+1
           goto    $+1
           goto    $+1

           ;4 ciclos (incluye llamada CALL)
           return

;-----fin rutina 1 segundo-----

;-----inicio rutina Tabla-----

DISPLAY7  ADDWF    PCL,1          ; codificación a 7 segmentos
           RETLW   3Fh          ; 0
           RETLW   06h          ; 1
           RETLW   5Bh          ; 2
           RETLW   4Fh          ; 3
           RETLW   66h          ; 4
           RETLW   6Dh          ; 5
           RETLW   7Dh          ; 6
           RETLW   07h          ; 7
           RETLW   7Fh          ; 8
           RETLW   67h          ; 9

;-----fin rutina Tabla-----

END

```

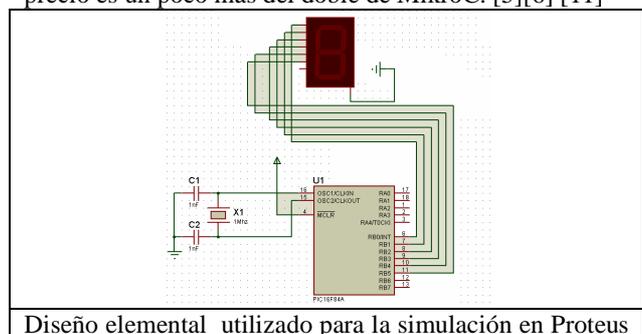
### 5.3 Herramientas de lenguaje de alto nivel

Un lenguaje de alto nivel (high-level language) es aquel que se aproxima más al lenguaje natural. Este utiliza palabras del lenguaje humano las cuales son por general el inglés. Este es más práctico y fácil de manejar para el ya que no se corren tantos riesgos de equivocarse como si lo es en el lenguaje de máquina o en ensamblador; logrando que el programador se concentre en el programa y no el lenguaje reduciendo el tiempo de desarrollo del programa.

Nuevamente, una de las más destacadas herramientas de desarrollo resulta ser MPLAB con su compilador ANSIC el cual es gratuito solo para estudiantes durante 60 días en su versión completa y no optimizado después de este tiempo. El compilador C de MPLAB para uso académico (LITE), se distribuye con una biblioteca completa ANSIC estándar. La biblioteca incluye funciones para la manipulación de cadenas, la asignación dinámica de memoria, conversión de datos, temporización, y funciones matemáticas (trigonométricas, exponenciales e hiperbólicas).[4][9]

Es importante notar que pese a ser esta una excelente alternativa, existe un amplio grupo de alternativas que van desde licenciamiento open source hasta plataformas pagadas.

Los autores del presente trabajo realizaron, una vez más, el mismo ejercicio presentado en ensamblador, que cuenta indefinidamente desde 0 hasta 9 en pasos de un segundo, haciendo uso del entorno de desarrollo de la compañía MikroElektronika con su producto MikroC PRO PIC 2009 v8.2 el cual es un entorno comercial de destacadas prestaciones. Este incluye un editor de código ANSIC, asistente de código, autocorrección, plantillas de código, resaltado de la sintaxis, opciones mejoradas de deshacer, simulación, ventanas de monitoreo, estadísticas de uso de los recursos de PIC, manejo simplificado de gráficos de LCD y amplio soporte de comunicaciones entre otras destacadas características. Algunos usuarios refieren como su mayor desventaja su falta de compatibilidad con el simulador Proteus, ya que no permite la ejecución de su código en C como lo hacen otras plataformas también comerciales tan respetables como PIC C de CCS. Este último sistema posee casi todas las mismas características estándar de MikroC, pero añade un editor de diagramas de flujo. Por otro lado, su precio es un poco más del doble de MikroC. [5][6] [11]



v7.4

```

1 const int Display7[10] = {63,6,91,79,102,109,125,7,127,103};
2 // los numero en decimal que general los segmentos de display
3 int contador;
4
5 void main()
6 {
7
8 TRISB = 0X00; // PORTB es de salida
9 contador=0;
10 do
11 {
12 PORTB =Display7[contador] ;
13 // Hace el puerto igual al valor de vector
14 delay_ms(1000);
15 //Espera 1 segundo
16
17 // Incrementa el contador
18 if (contador==9)
19     contador=0;
20 //Si el contador es 9, se reinicia
21 //la cuenta desde cero
22 else
23     contador++;
24 //Si la cuenta no es nueve, incrementa
25 }
26 while(contador<=9) ;
27 }
    
```

No obstante, es posible ejecutar este programa desde Proteus a través del archivo HEX que se genera desde el ambiente de desarrollo de MikroC. Como puede apreciarse, es mucho más simple el desarrollo en C, pero el precio se paga en el mayor número de líneas generadas de las que se obtienen en ensamblador o en lenguaje de máquina. [10]

**5.4 Lenguajes gráficos**

Como última alternativa de desarrollo evaluada están las herramientas de desarrollo gráfico que no requieren conocimiento específico de ningún lenguaje de programación. Dentro de este grupo se destaca Flowcode<sup>4</sup> y Corechart<sup>5</sup>.

CoreChart es una herramienta de desarrollo se fundamenta en el uso de iconos para la programación de microcontroladores. Este sistema promete simplificar y reducir el número de errores cometidos en lo que de otra manera sería de una gran complejidad, sin comprometer la velocidad de la máquina o la eficiencia. Corechart solo presenta soporte para microcontroladores de la serie 16 y no posee un muy amplio soporte para otros dispositivos. Corechart posee una versión gratuita reducida de desarrollo que puede ser descargada de su propio sitio. [8]

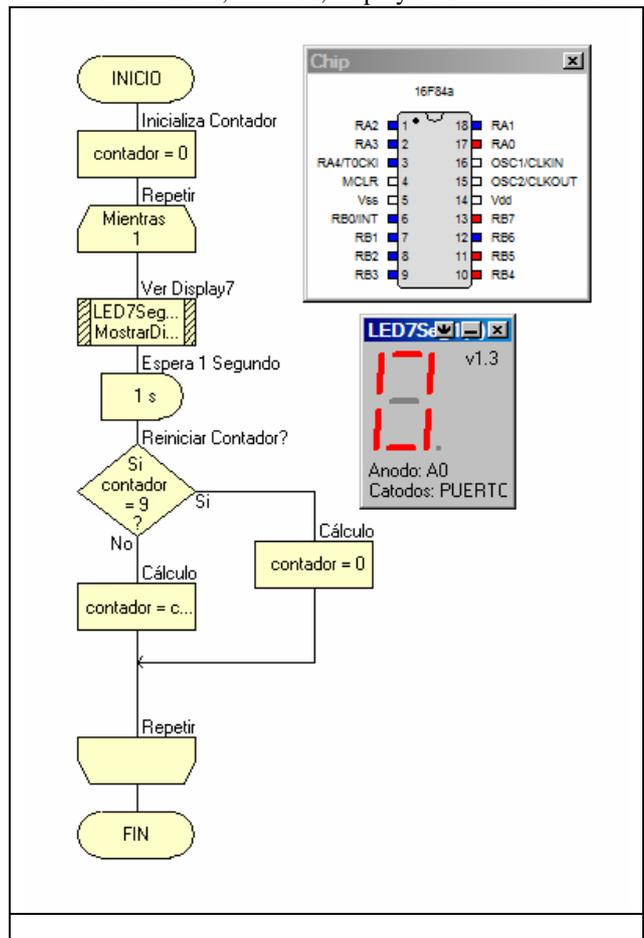
Por otro lado, Flowcode 4 es uno de los más avanzados lenguajes de programación gráfica para microcontroladores del mundo. Una de las grandes ventajas de Flowcode es que permite a aquellos con poca o ninguna experiencia de programación crear complejos sistemas electrónicos en cuestión de minutos. Flowcode está disponible en más de veinte idiomas y actualmente soporta la serie de microcontroladores PICmicro, AVR y ARM.

Flowcode es un poderoso lenguaje que utiliza macros para facilitar el control de dispositivos complejos, como displays de 7 segmentos, los controladores de motor, y pantallas LCD.

La utilización de macros permite a los usuarios controlar dispositivos electrónicos de alta complejidad sin atascarse en la comprensión de la programación en implicada en el desarrollo; esto permite que los usuarios con poca experiencia realicen aplicaciones desde el primer momento siempre que cuente con los conceptos básicos de algoritmos.

La versión 4 incluye panel de control, depuración en el circuito, redes virtuales, personalización de Código C, punto flotante y de nuevos componentes. [7]

Como parte del ejemplo que se ha desarrollado a lo largo de este documento, en el cual se cuenta desde 0 hasta 9 indefinidamente, se presenta la aplicación realizada en Flowcode 4. Esta fue mucho más simple de desarrollar y más intuitiva. Pese a su aparente superficialidad, es una herramienta muy robusta que permite manejar un amplio número de microcontroladores desde la serie 12 hasta la 18. Este soporta el desarrollo de aplicaciones con comunicación serial, USB, TCP/IP, Zigbee manejo de motores con PWM, teclados, displays LCD entre otros.



<sup>4</sup> <http://www.matrixmultimedia.com/>

<sup>5</sup> <http://www.elabtronics.com/>

Flowcode es reconocida internacionalmente como uno de los líderes del mercado en el desarrollo de microcontroladores para la educación. Flowcode se utiliza en más de 700 escuelas, colegios y universidades alrededor del mundo, pese a esto no es muy conocido en Colombia donde apoyaría de forma importante la introducción de las tecnologías microcontroladas en las escuelas, colegios y universidades de manera rápida, logrando desarrollos importantes en muy corto tiempo.

Flowcode posee versión gratuita que limita el tamaño del archivo a 2k y el número de dispositivos a usar. El Flowcode gratuito produce código hexadecimal sólo para los microcontroladores: 16F628A, 16F690, 16F88, 16F84A, 16F877, 18F2220 y 18F4431. Su precio comercial base es de aproximadamente 275 dólares para la versión Flowcode 4 home. Dentro de sus principales desventajas resulta las limitaciones en la arquitectura en el hardware que restringe, en algunas instancias, el diseño criterios de conexión preestablecidos por el programa. Por otro lado, es claro que el tamaño de los archivos HEX y ASM generados son de mucho mayor tamaño que los encontrados en los anteriores productos evaluados en este documento.[7]

	MPLAB	PIC C	MIKROC	CORECHART	FLOWCODE
Posee versión gratuita	Sí	Sí	Sí	Sí	Sí
Intuitivo	No	No	No	Sí	Sí
Precio	Excelente	Bueno	Bueno	Bueno	Bueno
Tiempo Desarrollo	Regular	Sobre	Sobre	Excelente	Excelente
Tiempo de Capacitación	prolongado	medio	medio	Rápido	Rápido
Compatibilidad con software terceros	Excelente	Bueno	Bueno	Bajo	Bueno
Tamaño archivos HEX	Excelente	Bueno	Bueno	Regular	Regular
Gama de PIC Soportados	Excelente	Excelente	Excelente	Regular	Bueno

Tabla 1. Comparativo cualitativo de los ambientes de desarrollo evaluados para Windows

Los autores de presente artículo realizaron una evaluación cualitativa de cada una de las plataformas de desarrollo la cual se puede apreciar en la tabla 1.

## 6. CONCLUSIONES

Existen un amplio grupo de herramientas que van desde aquellas que recurren a lo más puro del lenguaje ensamblador hasta entornos gráficos. Cada diseñador debe definir cual es la herramienta más adecuada de acuerdo a sus requerimientos, por lo que se puede concluir que todos los ambientes evaluados son idóneos dependiendo del espacio donde se deseen usar. Se deja al lector la selección final del nivel de la herramienta más adecuada para sus requerimientos de acuerdo al perfil del diseñador y los requerimientos específicos en tiempo de desarrollo, tamaño de los archivos o el precio de la plataforma a usar entre otros criterios.

## REFERENCIAS

- [1] MPLAB, Integrated Development Environment [En línea] disponible en: [http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=en019469&part=SW007002](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=en019469&part=SW007002) (consultado el 15 de Octubre de 2009)
- [2] Proteus, [En línea] disponible en: <http://www.labcenter.co.uk/index.cfm> (consultado el 15 de Octubre de 2009)
- [3] EasyPIC6, circuit Debugger [En línea] disponible en: <http://www.mikroe.com/en/tools/easypic6/> (consultado el 15 de Octubre de 2009)
- [4] MPLAB C Compiler for Academic Use [En línea] disponible en: [http://www.microchip.com/stellent/idcplg?IdcService=SS\\_GET\\_PAGE&nodeId=1406&dDocName=en536656](http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en536656) (consultado el 15 de Octubre de 2009)
- [5] Program PIC Esily [En línea] disponible en: <http://www.mikroe.com/en/compilers/mikroc/pro/pic/> (consultado el 15 de Octubre de 2009)
- [6] Custom Computer Services, [En línea] disponible en: <http://www.ccsinfo.com> (consultado el 15 de Octubre de 2009)
- [7] [En línea] disponible en: <http://www.matrixmultimedia.com/product.php?Prod=Flowcode%20V4%20for%20PIC&PHPSESSID=> (consultado el 15 de Octubre de 2009)
- [8] [En línea] disponible en: [http://www.elabtronics.com/products\\_cat\\_CoreChart.htm](http://www.elabtronics.com/products_cat_CoreChart.htm) (consultado el 15 de Octubre de 2009)
- [9] Palacios E., Remiro F., Lopez L., Microcontrolador PIC 16F84 Desarrollo de proyectos, Segunda Edición, Alfaomega, Mexico 2008, 623 paginas
- [10] Garcia Breijo E., compilador C CCS y simulador Proteus para Microcontroladores, Primero Edición, alfaomega, Mexico, 2008, 263 paginas.
- [11] Verle M., PIC microcontrollers Programming in C, Primera Edición, Belgrado 2009, 336 paginas.
- [12] Angulo J., angulo I., Microncontroladores PIC Diseno practico con aplicaciones, Primera Edición, Peru 1997, 221 paginas.
- [13] Mandado E., Menendez L., Ferrerira L., Matos E., Microcontrladores PIC-sistema integrado para autoaprendizaje, Espana 2007, 345 paginas.