

CONTENIDO

AGRADECIMIENTO	lv
MENSAJE DEL EDITOR.....	v
EL AUTOR	vi
LISTA DE EJEMPLOS.....	xiv
CONTENIDO PÁGINA WEB	xv
CÓDIGO WEB	xvii
PRÓLOGO	xix

CAPÍTULO 1

Conceptos básicos

Introducción	2
1.1 ¿Qué es un Sistema Embebido?.....	3
1.2 Conceptos básicos de programación en Alto Nivel.....	6
1.2.1 ¿Qué es un compilador? ¿Qué es un interpretador?	6
1.2.2 Estructura y pasadas de un compilador	8
1.2.3 Tiempo de compilación vs. tiempo de ejecución	11
1.3 ¿De dónde viene el ANSI C?.....	12
1.3.1 Forma general de un proyecto en C.....	13
1.4 Periféricos más comunes en Sistemas Embebidos.....	16
1.4.1 Puertos de entrada/salida y Función KBI.....	17
1.4.2 Conversor analógico a digital (ADC).....	19
1.4.3 Computador operando apropiadamente (COP).....	20
1.4.4 Detector de bajo nivel de voltaje (LVI).....	22
1.4.5 Temporizador (TIMER)	23
1.4.6 Comunicación serial asincrónica (SCI)	27
1.4.7 Comunicación serial sincrónica (SPI)	29
1.4.8 Comunicación serial I2C.....	31
1.5 Interrupciones en Sistemas Embebidos Microcontrolados.....	33
1.5.1 El Concepto de Interrupción.....	33
1.5.2 ¿Cómo trabaja el procesador ante una interrupción?	35
1.6 Cambio de contexto	40

1.7 Latencia de interrupción	41
1.8 Zonas críticas de software.....	42
1.9 Herramientas para Diseño Embebido en el mercado.....	43
1.9.1 Herramientas para Microchip™.....	43
1.9.2 Herramientas para Renesas™.....	45
1.9.3 Herramientas para Texas Instrumens.....	46
1.9.4 Herramientas para Freescale™.....	46
1.9.5 Herramientas utilizadas en los ejemplos.....	48
RESUMEN DEL CAPÍTULO.....	62
PREGUNTAS DEL CAPÍTULO.....	63

CAPÍTULO 2

Arquitectura básica de microcontroladores para sistema embebidos

Introducción	66
2.1 Arquitectura RISC (Harvard) Microchip de 8 bits.....	67
2.1.1 Componentes básicos de la arquitectura Microchip™.....	69
2.1.2 Modos de direccionamiento Microchip™.....	72
2.1.3 Mapa de memoria Microchip™ del PIC16F877A.....	75
2.1.4 Características del microcontrolador PIC16F877A.....	77
2.1.5 Fuentes de interrupción del microcontrolador PIC16F877A.....	80
2.2 Arquitectura CISC (Von Newman) Freescale™ de 8 bits.....	84
2.2.1 Modelo de programación Freescale™ HC(S)08.....	84
2.2.2 Modos de direccionamiento Freescale™ HC(S)08.....	87
2.2.3 Mapa de memoria del microcontrolador AP16A.....	92
2.2.4 Características del microcontrolador AP16A.....	93
2.2.5 Fuentes de interrupción del microcontrolador AP16A.....	96
RESUMEN DEL CAPÍTULO.....	102
PREGUNTAS Y EJEMPLOS SUGERIDOS.....	103

CAPÍTULO 3

Ambiente típico de compiladores para sistemas embebidos

Introducción	106
3.1 El manejador de Proyectos (IDE).....	107
3.2 Introducción al compilador CCS para Microchip™.....	109
3.2.1 Creación de proyectos embebidos usando el ambiente CCS.....	109
3.2.2 Familiarización con el IDE del CCS.....	113
3.3 Introducción al compilador Codewarrior® de Freescale™.....	126

3.3.1 Creación de proyectos embebidos en C usando Codewarrior®	126
3.3.2 Familiarización con el IDE de Codewarrior®	134
3.4 El primer programa embebido en C usando Codewarrior®	142
RESUMEN DEL CAPÍTULO	146
PREGUNTAS Y EJEMPLOS SUGERIDOS	147

CAPÍTULO 4

Conceptos de programación embebida en C

Introducción	150
4.1 Ventajas y desventajas del C usado en Sistemas Embebidos	151
4.2 Programación embebida vs. programación para PC	161
4.3 Constantes y variables	162
4.3.1 Constantes	163
4.3.2 Variables	169
4.4 Tipos de datos	170
4.4.1 char en Codewarrior ® (o int8 en CCS)	170
4.4.2 int en Codewarrior ® (o int16 para CCS)	170
4.4.3 long en Codewarrior ® (o int32 para CCS)	170
4.4.4 float (en Codewarrior ® y en CCS)	170
4.5 ¿Cómo elegir un tipo de variable en un sistema embebido?	172
4.6 Modificadores a tipos de datos comunes en Sistemas Embebidos	173
4.6.1 El modificador “unsigned”	174
4.6.2 El modificador “signed”	174
4.6.3 El modificador “volatile”	175
4.6.4 El modificador “near”	179
4.6.5 El modificador “for”	181
4.6.6 El modificador “register”	184
4.6.7 El modificador “const”	187
4.6.8 El modificador “static”	188
4.6.9 El modificador “extern”	192
4.7 Moldes o “Casting” para variables y constantes	195
RESUMEN DEL CAPÍTULO	198
PREGUNTAS DEL CAPÍTULO	199

CAPÍTULO 5

Directivas de compilación y operadores

Introducción	202
--------------------	-----

5.1	Directivas más comunes del preprocesador.....	203
5.1.1	Macro o equivalencia #define	203
5.1.2	Inclusión de archivo de cabecera #include.....	205
5.1.3	Notificación de error al compilar #error.....	205
5.1.4	Notificación de precaución al compilar #warning	206
5.1.5	Directiva #line	207
5.1.6	Compilación condicional #if, #elif, #else, #endif, #ifdef y #ifndef	207
5.1.7	Directiva #undef	209
5.1.8	Directiva #pragma	210
5.2	Medida en bytes de expresiones “sizeof”	213
5.3	Definiciones de tipo “typedef”	214
5.4	Operadores aritméticos.....	217
5.5	Operadores relacionales.....	220
5.5.1	Operadores de Comparación cuantitativa: <, >, <=, >=	220
5.5.2	Operador de igualdad: ==, !=	221
5.6	Operadores lógicos booleanos.....	221
5.6.1	El operador (OR).....	222
5.6.2	El operador && (AND).....	222
5.6.3	El operador ! (NOT).....	223
5.7	Operadores orientados a BIT.....	224
5.7.1	El operador (OR Bit a Bit).....	225
5.7.2	El operador & (AND bit a bit).....	226
5.7.3	El operador ~ (NOT bit a bit)	228
5.7.4	El operador ^ (OR exclusiva o XOR).....	228
5.8	Los operadores >> y << (Desplazamiento).....	232
5.9	Los operadores de APUNTAOR & Y *	236
5.9.1	Declaración de variables tipo apuntador.....	238
5.10	Precedencia y asociatividad	240
	RESUMEN DEL CAPÍTULO.....	243
	PREGUNTAS DEL CAPÍTULO.....	244

CAPÍTULO 6

Funciones y sentencias de control

	Introducción	246
6.1	Funciones en C para Sistemas Embebidos.....	247
6.1.1	Prototipo de una función.....	247
6.1.2	El concepto de paso de argumentos a función.....	251

6.1.3 Paso de argumentos por valor	252
6.1.4 Archivos de cabecera (.H)	255
6.1.5 Paso de argumentos por referencia	256
6.2 Sentencias de control.....	261
6.2.1 La sentencia “if .. else”	261
6.2.2 La sentencia “do... while”	267
6.2.3 La sentencia “while”	270
6.2.4 La sentencia “for”	273
6.2.5 La sentencia “switch”	278
6.2.6 La sentencia “break”	285
6.2.7 La sentencia “continue”	285
6.2.8 La sentencia “goto”	286
6.3 Mezcla de C con lenguaje ensamblador en Sistemas Embebidos	287
6.4 Recursividad en Sistemas Embebidos	293
6.5 Arreglos (ARRAYS) de datos	295
6.5.1 Arreglos unidimensionales	295
6.5.2 Cadenas de datos “strings”	301
6.5.3 Arreglos multidimensionales	303
6.5.4 Estructuras “struct” y uniones “union”	304
6.5.5 Estructuras de bits.....	310
6.6 Apuntadores a funciones	315
6.7 Manejo de interrupciones en Sistemas Embebidos desde C.....	320
6.7.1 Configuración de interrupciones en Codewarrior ®	323
6.8 Convenciones útiles de programación embebida en C.....	329
6.8.1 Sobre la distribución del código fuente.....	330
6.8.2 Sobre los nombres de funciones.....	331
6.8.3 Sobre las variables y apuntadores.....	332
6.8.4 Sobre las constantes	333
6.8.5 Sobre los paréntesis y corchetes.....	333
RESUMEN DEL CAPÍTULO	335
PREGUNTAS Y EJERCICIOS PROPUESTOS	336

CAPÍTULO 7

Librerías estándar en C para sistemas embebidos

Introducción	338
7.1 ¿Qué es una librería ANSI C?	339
7.2 Librería matemática <MATH.H>.....	340

7.3 Librería estándar <STDLIB.H>	349
7.4 Librería estándar de entrada/salida <STDIO.H>	352
7.5 Librería de manejo de cadenas <STRING.H>	359
7.6 Librería de tipos <CTYPE.H>	370
7.7 Librería de manejo de tiempo <TIME.H>	379
RESUMEN DEL CAPÍTULO	382
PREGUNTAS Y EJERCICIOS PROPUESTOS	383

CAPÍTULO 8

Bajo consumo de energía en sistemas embebidos

Introducción	386
8.1 La ecuación de consumo de energía	387
8.2 Modo “WAIT”	389
8.3 Modo “STOP”	390
8.3.1 Modo “Stop3”	391
8.3.2 Modo “Stop2”	392
8.3.3 Modo “Stop1”	394
8.4 Otros modos de bajo consumo	395
8.4.1 Modo “Low Power Run” (<i>LP_{run}</i>)	396
8.4.2 Modo “Low Power Wait” (<i>LP_{wait}</i>)	398
8.5 Consideraciones en el diseño de un Sistema Embebido	399
RESUMEN DEL CAPÍTULO	409
PREGUNTAS Y EJERCICIOS PROPUESTOS	410

CAPÍTULO 9

El “Processor Expert™” para sistemas embebidos Freescale™

Introducción	412
9.1 “PROCESSOR EXPERT™”	413
9.2 El Concepto del “BEAN”	414
9.3 Librería de “BEANS”	415
9.4 Creación de proyectos usando el “PROCESSOR EXPERT™”	416
9.5 Ventanas del proyecto con “PROCESSOR EXPERT™”	418
9.5.1 La ventana “bean inspector”	418
9.5.2 La ventana “bean selector”	420
9.5.3 La ventana “Target CPU”	421

9.6 Creación de <i>BEANS</i> en “ <i>PROCESSOR EXPERT™</i> ”.....	430
9.6.1 Creación de un <i>bean plantilla</i>	430
9.6.2 Uso del <i>Bean Wizard™</i>	432
9.7 Consideraciones sobre el uso del “ <i>PROCESSOR EXPERT™</i> ”	438
RESUMEN DEL CAPÍTULO	452
PREGUNTAS Y EJERCICIOS PROPUESTOS	453

CAPÍTULO 10

Sistemas operativos de tiempo real para sistemas embebidos

Introducción	456
10.1 ¿Qué es un sistema operativo de tiempo real?.....	457
10.2 Terminología básica sobre RTOS.....	459
10.2.1 Tareas (<i>Task</i> ó <i>Thread</i>)	459
10.2.2 Recursos (<i>resources</i>).....	462
10.2.3 Eventos (<i>events</i>).....	462
10.2.4 Semáforos (<i>semaphores</i>)	462
10.2.5 Mensajes (<i>message mailbox</i>).....	462
10.2.6 Bloques de Memoria (<i>buffers</i>).....	463
10.2.7 Reloj y Timers (<i>Clock Tick & Timers</i>).....	463
10.2.8 Kernel.....	463
10.3 Sistema de LOOP consecutivo/interrupción.....	466
10.3.1 Diseño del Loop principal	466
10.3.2 Archivo de cabecera <i>TareaX.H</i>	469
10.3.3 Archivo código fuente <i>TareaX.C</i>	470
10.3.4 Manejo de prioridades	474
10.4 Sistema RTOS UC/OS-II de Micrium	497
10.4.1 Consideración para la implementación del uC/OS-II	498
10.4.2 Manejo de las secciones críticas de software en uC/OS-II	500
10.4.3 Portado del uC/OS-II para el AP16A	501
RESUMEN DEL CAPÍTULO	509
PREGUNTAS Y EJERCICIOS PROPUESTOS	510
ANEXO No. 1 Tabla ASCII	511
ANEXO No. 2 Librerías de funciones estándar del C.....	512
GLOSARIO	517
BIBLIOGRAFÍA	523

LISTA DE EJEMPLOS

EJEMPLO 1. Encendido de Led en lenguaje de máquina	100
EJEMPLO 2. Encendido de led Microchip™ en C.....	122
EJEMPLO 3. Encendido de led Freescale™ en C.....	143
EJEMPLO 4. Declaración y uso de constantes en C.....	167
EJEMPLO 5. Usos y efectos del modificador “volatile”	176
EJEMPLO 6. Usos y efectos del modificador “near”	180
EJEMPLO 7. Usos y efectos del modificador “far”	182
EJEMPLO 8. Usos y efectos del modificador “register”	184
EJEMPLO 9. Uso del modificador “static”	189
EJEMPLO 10. Uso del modificador “extern”	193
EJEMPLO 11. Uso de operadores lógicos AND y OR.....	230
EJEMPLO 12. Uso de operadores desplazamiento.....	234
EJEMPLO 13. Paso de argumentos por valor	252
EJEMPLO 14. Paso de argumentos por referencia.....	257
EJEMPLO 15. Uso de la sentencia if..else.....	262
EJEMPLO 16. Uso de la sentencia “do...while”.....	268
EJEMPLO 17. Uso de la sentencia “while”	271
EJEMPLO 18. Uso de la sentencia “for”.....	276
EJEMPLO 19. Uso de la sentencia “switch”	281
EJEMPLO 20. Mezclando lenguaje C y ensamblador.....	290
EJEMPLO 21. Acceso a arreglos de datos.....	297
EJEMPLO 22. Declaración y manejo de estructuras.....	306
EJEMPLO 23. Estructuras de bits en C.....	312
EJEMPLO 24. Apuntadores a funciones	316
EJEMPLO 25. Base de tiempo real.....	326
EJEMPLO 26. Ángulos de inclinación en grados	343
EJEMPLO 27. Localización dinámica de memoria	350
EJEMPLO 28. La calculadora serial.....	352
EJEMPLO 29. Manejo de display LCD	360
EJEMPLO 30. Manejo de cadenas.....	370
EJEMPLO 31. El reloj de bajo consumo.....	401
EJEMPLO 32. Uso básico del “Processor Expert™”	422
EJEMPLO 33. Creación básica de beans	434
EJEMPLO 34. Escritura y lectura de memoria E2PROM	440
EJEMPLO 35. Alarma de intrusión	476
EJEMPLO 36. Manejo de RTOS uC/OS-II	504

CONTENIDO PÁGINA WEB

Capítulo 1

Información sobre el sistema ICD-U40	45
Información sobre las herramientas para Freescale™	48
Información en formato PDF sobre el AP-Link.....	50
Información en formato PDF sobre el PIC-Link.....	56

Capítulo 2

Hojas de datos del Microcontrolador PIC 16F877A.....	83
Especificación más completa microcontrolados AP16A en el documento CPU08RM Freescale.....	93
Hoja de datos del procesador Freescale MC68HC908AP16A	100
Código fuente: encendido de led en lenguaje maquina, para Freescale™	101

Capítulo 3

Video explicativo sobre creación de proyectos usando compilador CCS.....	109
Información adicional sobre el manejo del compilador CCS.....	113
Código fuente: encendido de led en C, para Microchip	125
Video explicativo sobre encendido de led Microchip.....	125
Video explicativo para crear proyectos en Codewarrior ®.....	126
Video explicativo sobre el manejo del IDE del Codewarrior ®	134
Código fuente: encendido de led en C, para Freescale™	145

Capítulo 4

Código fuente: declaracion y uso de constantes, para Freescale™ y Microchip™	169
Código fuente: uso del modificador “volatile”, para Freescale™ y Microchip™	178
Código fuente: uso del modificador “near”, para Freescale™	180
Código fuente: uso del modificador “far”, para Freescale™	183
Código fuente: uso del modificador “register”, para Freescale™ y Microchip™	187
Código fuente: uso del modificador “static”, para Freescale™ y Microchip™	192
Código fuente: uso del modificador “extern”, para Freescale y Microchip	195

Capítulo 5

Código fuente: uso de operadores AND y OR, para Freescale™ y Microchip™	232
Código fuente: uso de operadores desplazamiento, para Freescale™ y Microchip™	236

Capítulo 6

Código fuente, paso de argumentos por valor, para Freescale™ y Microchip™	254
Código fuente, paso de argumentos por referencia, para Freescale™ y Microchip™	260
Código fuente, uso de la sentencia “if...else”, para Freescale™ y Microchip™	266

Código fuente, uso de la sentencia “do...while”, para Freescale™ y Microchip™	270
Código fuente, uso de la sentencia “while”, para Freescale™ y Microchip™	273
Código fuente, uso de la sentencia “for”, para Freescale™ y Microchip™	277
Código fuente, uso de la sentencia “switch”, para Freescale™ y Microchip™	284
Código fuente, mezclando C y ensamblador, para Freescale™ y Microchip™	293
Código fuente, acceso a arreglos de datos, para Freescale™ y Microchip™	300
Código fuente, manejo de estructuras, para Freescale™ y Microchip™	309
Código fuente, estructuras de bits en C, para Freescale™ y Microchip™	315
Código fuente, apuntadores a funciones, para Freescale™ y Microchip™	319
Código fuente, base de tiempo real, para Freescale™ y Microchip™	329

Capítulo 7

Lista de funciones de la librería <math.h> en el Anexo # 2, y explicación de cada función	341
Información y hoja de datos del acelerómetro de Freescale™ MMA7260Q.....	342
Información y hojas de datos de los componentes del acelerómetro MMA7260Q de Freescale.	343
Código fuente: ángulos de inclinación en grados, para Freescale™	348
Lista de funciones de la librería <stdlib.h> en el Anexo # 2, y explicación de cada función	349
Código fuente: localización dinámica de memoria, para Freescale	351
Lista de funciones de la librería <stdio.h> en el Anexo # 2, y explicación de cada función	352
Código fuente: la calculadora serial, para Freescale™ y Microchip™	359
Lista de funciones de la librería <string.h> en el Anexo # 2, y explicación de cada función.....	359
Código fuente: manejo de display LCD, para Freescale™ y Microchip™	369
Lista de funciones de la librería <ctype.h> en el Anexo # 2, y explicación de cada función	370
Código fuente: manejo de cadenas, para Freescale™ y Microchip™	378
Lista de funciones de la librería <time.h> en el Anexo # 2, y explicación de cada función.....	381

Capítulo 8

Código fuente: el reloj de bajo consumo, para Freescale™	401
Información sobre los modos de bajo consumo de los procesadores HC(S)08 y Flexis™	407

Capítulo 9

Uso básico del Processor Expert: simulación y proyecto completo para Freescale™ (AP-Link).....	430
Escritura y lectura memoria E2PROM: simulación y proyecto completo para Freescale™ (AP-Link)	451

Capítulo 10

Alarma de intrusión: proyecto completo para Freescale™ (AP-Link) y Microchip™ (PIC-Link).	496
Manejo de RTOS uC/OS-11: proyecto completo para Freescale™ (AP-Link) y Flexis™	508

Anexo No. 2

Información adicional y explicación de las funciones de cada librería estándar del C	512
--	-----

CÓDIGO WEB

Para tener acceso al material de la página Web de apoyo del libro “Programación de sistemas embebidos en C, teoría y prácticas aplicadas a cualquier microcontrolador”:

1. Ir a la página
<http://virtual.alfaomega.com.mx>
2. Registrarse como usuario del sitio.
3. Ingresar al apartado de inscripción de libros y registrar la siguiente clave de acceso:

4. Para navegar en la plataforma virtual de recursos del libro, usar los nombres de Usuario y Contraseña definidos en el punto número dos.