

Programación de  
**SISTEMAS  
EMBEBIDOS**  
en C

Gustavo Galeano



Alfaomega

## INDICE

### Introducción

- 1.1 ¿Qué es un Sistema Embebido?
- 1.2.1 ¿Qué es un compilador? ¿Qué es un interpretador?
- 1.2.2 Estructura y pasadas de un compilador
- 1.2.3 Tiempo de compilación vs tiempo de ejecución
- 1.3 ¿De dónde viene el ANSI C?
- 1.3.1 Forma general de un proyecto en C
- 1.4 Periféricos más comunes en Sistemas Embebidos
- 1.4.1 Puertos de entrada/salida y Función KBI
- 1.4.2 Conversor analógico a digital (ADC)
- 1.4.3 Computador operando apropiadamente (COP)
- 1.4.4 Detector de bajo nivel de voltaje (LVI)
- 1.4.5 Temporizador (TIMER)
- 1.4.6 Comunicación serial asincrónica (SCI)
- 1.4.7 Comunicación serial sincrónica (SPI)
- 1.4.8 Comunicación serial I2C
- 1.5.1 El Concepto de Interrupción
- 1.5.2 ¿Cómo trabaja el procesador ante una interrupción?
- 1.6 Cambio de contexto
- 1.7 Latencia de interrupción
- 1.8 Zonas críticas de software
- 1.9.1 Herramientas para Microchip™
- 1.9.2 Herramientas para Renesas™
- 1.9.3 Herramientas para Texas Instruments
- 1.9.4 Herramientas para Freescale™
- 1.9.5 Herramientas utilizadas en los ejemplos
- Resumen del capítulo
- Preguntas del capítulo

### Introducción

---

- 2.1 Arquitectura RISC (Harvard) Microchip de 8 bits
- 2.1.1 Componentes básicos de la arquitectura Microchip™
- 2.1.2 Modos de direccionamiento Microchip™
- 2.1.3 Mapa de memoria Microchip™ del PIC16F877A
- 2.1.4 Características del microcontrolador PIC16F877A
- 2.1.5 Fuentes de interrupción del microcontrolador PIC16F877A
- 2.2 Arquitectura CISC (Von Newman) Freescale™ de 8 bits
- 2.2.1 Modelo de programación Freescale™ HC(S) 08
- 2.2.2 Modos de direccionamiento Freescale™ HC(S) 08
- 2.2.3 Mapa de memoria del microcontrolador AP16A