1. Record Nr. UNINA9911006574403321

Autore	Di Jasio Lucio
Titolo	Programming 16-bit PIC microcontrollers in C : learning to fly the PIC 24 / / Lucio Di Jasio
Pubbl/distr/stampa	Oxford, : Elsevier, 2012
ISBN	9786613348234 9781283348232 1283348233 9781856178716 1856178714 9781856178709 1856178706
Edizione	[2nd ed.]
Descrizione fisica	1 online resource (415 p.)
Collana	Embedded Technology
Disciplina	005.1 629.8/95
Soggetti	C (Computer program language) Microprocessors - Programming
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Front Cover; Programming 16-Bit PIC Microcontrollers in C; Copyright Page; Contents; Preface; Introduction to the Second Edition; Introduction; Who Should Read this Book?; Structure of the Book; What This Book Is Not; Checklists; I: First Flights; 1: The First Flight; Flight Plan; Preflight Checklist; The Flight; Compiling and Linking; Debugging the First Project; Port Initialization; Re-Testing PortA; Configuring the PIC24; Testing PortB; Analog vs Digital Pin Control; Post-Flight Briefing; Notes for the Assembly Experts; Notes for the PIC Microcontroller Experts; Notes for the C Experts Tips & TricksExercises; Books; Links; 2: A Loop in the Pattern; Flight Plan; Preflight Checklist; The Flight; While Loops; Animation; Not So Fast, Please!; Post-Flight Briefing; Notes for the Assembly Experts; Notes for the PIC® Microcontroller Experts; Notes for the C Experts; Tips & Tricks; Exercises; Books; Links; 3: More Pattern Work, More Loops; Flight Plan; Preflight Checklist; The Flight; do Loops; Variable

Declarations; for Loops; More Loop Examples; Arrays; Sending a Message; Post-Flight Briefing; Notes for the Assembly Experts; Notes for the PIC® Microcontroller Experts
Notes for the C ExpertsTips & Tricks; Notes for PIC24 GA1 and GB1 Users; Exercises; Books; Links; 4: Numb3rs; Flight Plan; Preflight Checklist; The Flight; Going Long; Long Long Multiplications; Floating Point; Notes for the C Experts; Measuring Performance; Post-Flight Briefing; Notes for the Assembly Experts; Notes for the PIC® Microcontroller Experts; Tips & Tricks; Math Libraries; Complex Data Types; Exercises; Books; Links; 5: Interrupts; Flight Plan; Preflight Checklist; The Flight; Nesting of Interrupts; Traps; Trap Vector Details; A Template and an Example for a Timer1 Interrupt
A Real Example with Timer1Testing the Timer1 Interrupt; The Secondary Oscillator; The Real-Time Clock Calendar (RTCC); Managing Multiple Interrupts; Post-Flight Briefing; Notes for the C Experts; Notes for the Assembly Experts; Notes for the PIC Microcontroller Experts; Tips & Tricks; Exercises; Books; Links; 6: Taking a Look Under the Hood; Flight Plan; Preflight Checklist; The Flight; Memory Space Allocation; Program Space Visibility; Investigating Memory Allocation; Looking at the Map; Pointers; The Heap; MPLAB C Memory Models; Post-Flight Briefing; Notes for the C Experts
Notes for the Assembly ExpertsNotes for the PIC Microcontroller Experts; Tips & Tricks; Exercises; Books; Links; II: Flying "Solo"; 7: Synchronous Communication; Flight Plan; Preflight Checklist; The Flight; Synchronous Serial Interfaces; Asynchronous Serial Interfaces; Synchronous Communication Using the SPI Modules; Testing the Read Status Register Command; Writing to the EEPROM; Reading the Memory Contents; A Non-Volatile Storage Library; Testing the New SEE Library Module; The I2C Interface; I2C Data Transfer Rules; I2C Serial EEPROMs; Talking to I2C Serial EEPROMs; Forming Commands
The SEE Grammar

Sommario/riassunto

New in the second edition: * MPLAB X support and MPLAB C for the PIC24F v3 and later libraries * I2CTM interface * 100% assembly free solutions * Improved video, PAL/NTSC * Improved audio, RIFF files decoding * PIC24F GA1, GA2, GB1 and GB2 support Most readers will associate Microchip's name with the ubiquitous 8-bit PIC microcontrollers but it is the new 16-bit PIC24F family that is truly stealing the scene. Orders of magnitude increases of performance, memory size and the rich peripheral set make programming these devices in C a