

Contents :

Part 1: Exploring

- **Day 1: And the Adventure Begins** – Introduction to the PIC32, setting up the MPLAB® IDE, and the "Hello World" of embedded systems (toggling an I/O pin).
- **Day 2: Walking in Circles** – Basics of loops in C (while and for), timing, and using the simulator (MPLAB SIM).
- **Day 3: Message in a Bottle** – Working with strings, character arrays, and basic serial output.
- **Day 4: NUMB3RS** – Data types, integer math, floating-point performance, and the C32 compiler's efficiency.
- **Day 5: Interrupts** – Understanding the multi-vectored interrupt controller, exception handling, and basic timer interrupts.
- **Day 6: Memory** – Deep dive into the PIC32 memory map, pointers, the heap, and the virtual memory system (KSEG0/KSEG1).

Part 2: Experimenting

- **Day 7: Running** – Configuration bits, oscillator settings (PLL), and optimizing performance with Flash wait states and the instruction cache.
- **Day 8: Communication** – Synchronous serial interfaces using the SPI modules.
- **Day 9: Links** – Asynchronous serial communication (UART) and building a simple console library.
- **Day 10: Glass = Bliss** – Interfacing with LCD displays (Parallel Master Port - PMP) and creating custom characters.
- **Day 11: It's an Analog World** – Using the Analog-to-Digital Converter (ADC) for sensors and joysticks.

Part 3: Expansion

- **Day 12: Capturing User Inputs** – Debouncing buttons, using Rotary Encoders, and the PS/2 keyboard protocol.
- **Day 13: UTube** – Generating video signals (Composite/VGA) using the SPI and DMA modules.
- **Day 14: Mass Storage** – Interfacing with SD/MMC cards via SPI.
- **Day 15: File I/O** – Implementing a FAT file system to read and write files on an SD card.
- **Day 16: Musica, Maestro!** – Generating sound and music using Pulse Width Modulation (PWM) and the Output Compare modules.

Index