

Table of Contents

| | |
|---|----|
| Introduction | 11 |
| Target Audience | 11 |
| What is an Embedded System? | 11 |
| Why use the C Programming Language? | 13 |
| Prerequisites | 13 |
| The Approach Taken in this Book | 19 |
| Some Good Advice | 19 |
| | |
| 1 Your First C Program | 20 |
| 1.1 About the Programming Tools | 20 |
| 1.2 Downloading the DJGPP Compiler | 22 |
| 1.3 Installing the DJGPP Compiler | 23 |
| 1.4 Setting Up the DJGPP Compiler | 25 |
| 1.5 Installing the Compiler from the Elektor Website | 28 |
| 1.6 How C Programs are Created | 29 |
| 1.7 Start Programming | 30 |
| 1.8 Compiling the Program | 31 |
| 1.9 About Your First C Program | 35 |
| 1.10 Analysing the Program | 35 |
| 1.10.1 Characters and Strings | 35 |
| 1.10.2 Functions | 36 |
| 1.10.3 Program Statements | 36 |
| 1.10.4 Preprocessor Directives | 37 |
| 1.10.5 Whitespace Characters | 37 |
| 1.11 Exercises | 37 |
| 1.11.1 Solutions | 38 |
| 1.12 Summary | 38 |
| | |
| 2 C Basics | 40 |
| 2.1 Setting up Programmer's Notepad | 40 |
| 2.1.1 Some General Settings | 40 |
| 2.1.2 A New Program to Compile | 43 |
| 2.1.3 Setting up Programmer's Notepad to Run the DJGPP Compiler | 45 |
| 2.1.4 A Keyboard Shortcut for the Compiler | 47 |
| 2.1.5 Setting Up Programmer's Notepad to Run Programs | 48 |
| 2.2 How the sticky.c Program Works | 50 |
| 2.3 Input, Output and Variables | 51 |
| 2.3.1 Variables | 53 |
| 2.3.2 Getting Input from the User – the <code>scanf ()</code> Function | 55 |
| 2.4 Variable Types | 57 |
| 2.4.1 Floating Point Variables | 57 |
| 2.4.2 Character Variables | 58 |
| 2.5 Arithmetic Operators | 59 |
| 2.6 Field Width Specifiers | 61 |

| | | |
|--------|--|-----|
| 2.7 | Compiling and Linking | 61 |
| 2.7.1 | Some Compile and Link Experiments | 63 |
| 2.8 | Errors and Warnings | 64 |
| 2.8.1 | Compile Errors | 64 |
| 2.9 | Link Errors | 66 |
| 2.10 | Exercises | 67 |
| 2.10.1 | Solutions | 68 |
| 2.11 | Summary | 69 |
| 3 | Comparative Operators and Decisions | 71 |
| 3.1 | Comparative Operators | 71 |
| 3.1.1 | True and False | 71 |
| 3.2 | Decisions | 72 |
| 3.2.1 | Using <code>if</code> to Make a Decision | 73 |
| 3.2.2 | Using <code>else</code> in Conjunction with <code>if</code> | 73 |
| 3.2.3 | The <code>=</code> Operator and the <code>==</code> Operator | 75 |
| 3.2.4 | Using <code>else if</code> in Conjunction with <code>if</code> | 77 |
| 3.3 | Exercises | 79 |
| 3.3.1 | Solutions | 79 |
| 3.4 | Summary | 80 |
| 4 | The <code>while</code> Loop | 82 |
| 4.1 | The <code>while</code> Loop | 82 |
| 4.2 | Using <code>if</code> Inside the <code>while</code> Loop | 84 |
| 4.3 | The Guess My Number Game | 85 |
| 4.4 | Back to the Temperature Controller Example | 86 |
| 4.5 | Commenting Programs | 88 |
| 4.6 | Programming Style | 89 |
| 4.7 | Exercises | 91 |
| 4.7.1 | Solutions | 91 |
| 4.8 | Summary | 93 |
| 5 | Functions | 94 |
| 5.1 | Your Second Function | 95 |
| 5.2 | Passing Data to a Function | 96 |
| 5.3 | Passing More Than One Value to a Function | 97 |
| 5.4 | Passing a Variable to a Function | 98 |
| 5.5 | Getting a Value Back from a Function | 99 |
| 5.6 | Passing Values to a Function and Receiving a Value Back | 100 |
| 5.7 | Flashing LED Simulation Program | 101 |
| 5.8 | Pre-processor Directives | 102 |
| 5.9 | Functions Calling Functions | 106 |
| 5.10 | Using Multiple Source Files | 107 |
| 5.11 | Header Files | 108 |
| 5.12 | The <code>make</code> Program and Make File | 109 |
| 5.12.1 | How the Make File Works | 112 |

| | | |
|--------|--|-----|
| 5.13 | How Functions Relate to Linking and Library Files | 112 |
| 5.14 | Exercises | 113 |
| 5.14.1 | Solutions | 113 |
| 5.15 | Summary | 115 |
| 6 | Number Systems | 117 |
| 6.1 | Binary Basics | 117 |
| 6.2 | The Need for Binary Numbers | 118 |
| 6.3 | Numbering Systems | 119 |
| 6.3.1 | A Quick Look at Decimal Numbers | 119 |
| 6.3.2 | Binary Numbers | 120 |
| 6.3.3 | Hexadecimal Numbers | 124 |
| 6.4 | Working with Hexadecimal Numbers in C | 129 |
| 6.4.1 | Field Width Specifiers Revisited | 131 |
| 6.5 | The ASCII Alphanumeric Code | 133 |
| 6.6 | Exercises | 137 |
| 6.6.1 | Solutions | 137 |
| 6.7 | Summary | 138 |
| 7 | Memory and Microcontrollers | 139 |
| 7.1 | Memory Basics | 139 |
| 7.2 | A Look at a Memory Chip | 140 |
| 7.3 | How Microprocessors Access Memory and Peripherals | 144 |
| 7.4 | Pointers | 145 |
| 7.5 | More on C Data Types | 148 |
| 7.6 | Choosing a Microcontroller and Embedded System | 152 |
| 7.7 | Summary | 155 |
| 8 | Your First Embedded C Program | 157 |
| 8.1 | How Embedded Programming Differs from PC Programming | 157 |
| 8.2 | The Embedded C Programming Tools | 160 |
| 8.3 | The YAGARTO Toolchain | 160 |
| 8.3.1 | Installing YAGARTO | 161 |
| 8.3.2 | Testing the YAGARTO Installation | 163 |
| 8.3.3 | Running Eclipse for the First Time | 163 |
| 8.4 | Writing Your First Embedded Program | 164 |
| 8.4.1 | The <code>while(1)</code> Loop | 164 |
| 8.4.2 | About the Source Code | 165 |
| 8.4.3 | Opening the Program in Eclipse | 165 |
| 8.4.4 | Modifying the Program to Run on Your Embedded System | 169 |
| 8.4.5 | Modifying the Make File | 174 |
| 8.4.6 | Loading the Program to Flash Memory | 174 |
| 8.4.7 | A Brief Explanation of the Project Files | 178 |
| 8.5 | Exercises | 179 |
| 8.6 | Summary | 180 |

| | | |
|---------|--|-----|
| 9 | Embedded I/O & Memory Maps | 181 |
| 9.1 | Loading a Program into SRAM | 182 |
| 9.2 | Writing to More than One LED | 183 |
| 9.2.1 | The Program's New C Language Elements | 188 |
| 9.3 | Reading the Switches and Writing to the LEDs (I/O) | 189 |
| 9.4 | The AT91SAM7S Memory Map | 194 |
| 9.5 | A Closer Look at the PIO Controller | 195 |
| 9.6 | Exercises | 199 |
| 9.6.1 | Solutions | 199 |
| 9.7 | Summary | 199 |
| 10 | The DBGU Serial Port | 201 |
| 10.1 | Hardware Requirements for PC to μ C Serial Communications | 201 |
| 10.1.1 | Serial Cable | 201 |
| 10.1.2 | USB to RS-232 Serial Adapter | 203 |
| 10.2 | Programming the Serial Port | 203 |
| 10.2.1 | About the DBGU Serial Port Hardware | 209 |
| 10.2.2 | How the <code>serial_tx</code> Program Works | 211 |
| 10.3 | Using the <code>sprintf()</code> Function | 213 |
| 10.4 | Receiving Data on the Serial Port | 215 |
| 10.5 | Exercises | 217 |
| 10.5.1 | Solutions | 217 |
| 10.6 | Summary | 218 |
| 11 | Previous C Topics Revisited | 219 |
| 11.1 | Serial Port Driver | 219 |
| 11.2 | Format Specifiers | 220 |
| 11.3 | Escape Sequences | 221 |
| 11.4 | Loops | 224 |
| 11.4.1 | A <code>while</code> Loop that uses <code>break</code> and <code>continue</code> | 224 |
| 11.4.2 | The <code>do while</code> Loop | 225 |
| 11.4.3 | The <code>for</code> Loop | 226 |
| 11.5 | Nested Loops and Decisions | 227 |
| 11.6 | Decision Making with the <code>switch</code> Statement | 229 |
| 11.7 | The Conditional Operator | 231 |
| 11.8 | Functions and Pointers | 233 |
| 11.8.1 | Returning More Than One Value from a Function | 235 |
| 11.9 | Variables and Scope | 236 |
| 11.9.1 | Global Variables | 236 |
| 11.10 | Static Variables | 237 |
| 11.11 | Floating Point Data Types | 238 |
| 11.12 | Casts | 238 |
| 11.13 | Exercises | 239 |
| 11.13.1 | Solutions | 239 |
| 11.14 | Summary | 240 |

| | | |
|---------|---|-----|
| 12 | Arrays and Strings | 242 |
| 12.1 | Arrays | 242 |
| 12.2 | Strings | 246 |
| 12.2.1 | Writing to a String | 248 |
| 12.2.2 | Initialising a String | 250 |
| 12.2.3 | C Library String Functions | 251 |
| 12.3 | Arrays and Addresses | 252 |
| 12.3.1 | Passing an Array to a Function | 254 |
| 12.4 | Strings as Pointers | 256 |
| 12.5 | A Look at the <code>DBGUTxMSG()</code> Function | 256 |
| 12.6 | Multidimensional Arrays | 257 |
| 12.7 | Exercises | 260 |
| 12.7.1 | Solutions | 261 |
| 12.8 | Summary | 261 |
| 13 | Bit Manipulation and Logical Operators | 263 |
| 13.1 | Bit Manipulation with Bitwise Operators | 263 |
| 13.1.1 | Why Do We Need Bitwise Operators? | 266 |
| 13.1.2 | The Left and Right Shift Operators | 269 |
| 13.1.3 | The C Assignment Operators | 271 |
| 13.2 | Logical Operators | 272 |
| 13.3 | Operator Precedence | 275 |
| 13.4 | Exercises | 276 |
| 13.4.1 | Solutions | 276 |
| 13.5 | Summary | 277 |
| 14 | More Hardware Programming | 278 |
| 14.1 | The AT91SAM7S Timer Counter | 278 |
| 14.2 | The Analogue to Digital Converter (ADC) | 281 |
| 14.3 | Using the Timer and Interrupt | 283 |
| 14.4 | The Watchdog Timer | 286 |
| 14.5 | Summary | 288 |
| 15 | Wrapping Up | 290 |
| 15.1 | Structures | 290 |
| 15.1.1 | Pointers to Structures | 293 |
| 15.2 | Unions | 294 |
| 15.3 | Enumerated Type | 296 |
| 15.4 | The <code>typedef</code> Declarator | 299 |
| 15.5 | Storage Class Specifiers | 300 |
| 15.6 | Type Qualifiers | 301 |
| 15.7 | The <code>goto</code> Statement | 301 |
| 15.8 | A List of All C Keywords | 302 |
| 15.9 | More Preprocessor Directives | 302 |
| 15.10 | Debugging | 303 |
| 15.11 | Some Final Example Programs | 303 |
| 15.11.1 | Voltmeter | 303 |

| | |
|---|-----|
| 15.11.2 Voltmeter with ADC Filtering | 305 |
| 15.11.3 Stopwatch | 305 |
| 15.12 Summary | 306 |
| Appendix A: The ASCII Table | 308 |
| Appendix B: Source Code and Software | 310 |
| PC programs Chapter 1 to Chapter 7 | 310 |
| Embedded Programs Chapter 8 to Chapter 15 | 313 |
| Software Programs | 316 |
| Index | 317 |