

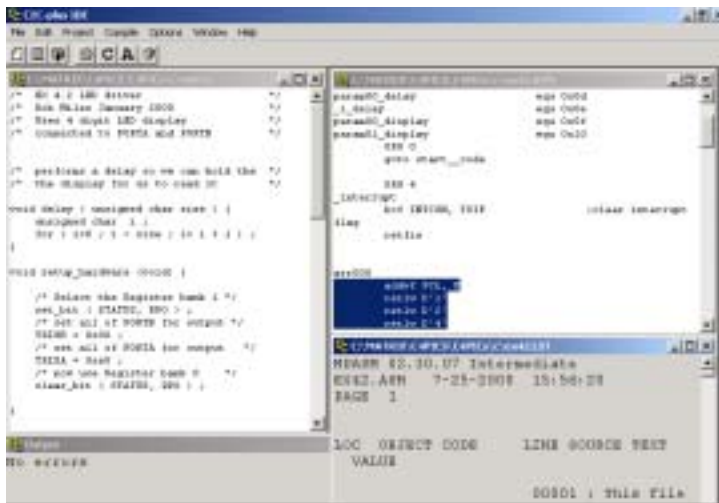
The C for PICmicro® microcontrollers CD-ROM is designed for students and professionals who need to learn how to program embedded microcontrollers in C. The CD contains a course as well as all the software tools needed to create Hex code for a wide range of PICmicro devices - including SourceBoost, a full C compiler and an Integrated Development Environment (IDE).

Although the course focuses on the use of the PICmicro microcontrollers, this CD-ROM will provide a good grounding in C programming for any microcontroller. The C compiler on this product is not licensed for industrial use.



Screen image shows Virtual C PICmicro

- Complete course in C as well as C programming for PICmicro microcontrollers
- Highly interactive course
- Virtual C PICmicro improves understanding
- Includes a C compiler for a wide range of PICmicro devices
- Includes full Integrated Development Environment
- Includes MPLAB software
- Compatible with most PICmicro programmers



Screen image shows IDE and C compiler

This model is available in the following configurations:

48104-00	C for PICmicro MCUs V2.0 CD-ROM	Single User
48104-B0	C for PICmicro MCUs V2.0 CD-ROM	10 User
48104-S0	C for PICmicro MCUs V2.0 CD-ROM	Site License

### What's new in version 3?

This second version differs from version 2.0 in three significant ways:

- The C compiler for V3.0 covers an even greater range of PICmicro devices than in V2.0 - see table below.
- V3.0 includes a tutorial on working with Analogue inputs.
- V3.0 includes the updated and improved SourceBoost IDE – successor to C2C IDE.
- Includes PPP V3
- Includes support for E-blocks



“I was very impressed with the Assembly for PICmicros CD ROM (AKA PICtutor) and have no hesitation in recommending it as a powerful method of delivering the PIC programming element of the A level electronics course. C for PICmicro MCUs is the perfect companion.”

John Verrill  
Electronics teacher & examiner  
Whitby Community College,  
UK

## The contents of the course on the CD-ROM are:

### Introduction

About the course, Navigation, What is a PICmicro?, the Virtual C PICmicro, MPLAB, C2C IDE

### Programming

Conventional programs, Embedded programs, High and low level, Low level PICmicro, Getting y programs into a compiler, writing our programs.

### What is a C program?

Introduction, Comments, The main function header, The main function body, The end of the line, Exercises,

### Variables

Introduction, Types, Floating point, Characters, Integers, Choosing variables, Identifiers, Declara Multiple declarations, Initialization, Names, Assigning statements, Expressions, Operators and Operands, Constant operands, Operators and divide, Working on bits, Shifting bits, Shortcut oper Unary operators, Casting Exercises

### Conditional Statements

Introduction, Logical operators, Equality, Using else, Cunning conditions, Complicated condition Unary operators in conditions, Exercises.

### Statements and Blocks

Introduction , Blocks in Blocks, Global and local variables, Local variable scope, Global variable Global and local in C2C, Exercises.

### Looping the loop

Introduction, While loop, Counting with the while loop, The for loop, Breaking out of loops, Continuing loops, Exercises.

### Functions

Introduction, When to use a function, Functions which return values, Functions which accept parameters, Functions which return values, More than one parameter, Exercises.

### Arrays

Introduction, The need for arrays, Declaring an array, elements in an array, Sorting array element the next dimension, Exercises.

### Switches

Introduction, The switch condition, Switches and breaks, Exercises.

### Pointers

Introduction, Arrays and pointers, Declaring a pointer, Using a pointer, Comparing a pointer, Nul pointers, Functions and pointers, Finding out the size, Strings, Exercises.

### Structures

Introduction, Creating structures, How structures work, Pointers in structures, Structures in struct Exercises.

### The pre-processor

Introduction, The #include directive, magic numbers and #define, Conditional compilation, Exerc

### Software engineering in C

Introduction, Deciding on specification, Deciding how to test, creating the functions, Using the functions, Making a project, External data, And finally...

### PICmicro reference

Introduction, Bits and bytes, Hex and decimal, Ready reckoner, Exercises, The PIC microcontrol PICmicro range, Architecture, Programming the PICmicro, The workign register, File registers, B selection, Special function registers, PORTA and PORTB, TRISA and TRISB, Interrupts, Timer, Watchdog, Sleep, EEPROM memory, Development board, Exercises.

### Compiling and running programs

Introduction, Compiling your program, assembling your program, downloading your program, C2 functions, C2C project files.

### Labs

Lab 1 - Flashing lights, Lab 2 - Switches and torches, Lab 3Sound software, Lab 4 - LEDs and clocks, Lab 5 - LCDs and libraries, Lab 6 - Mystic LCD, Lab 7 - Picing a lock, Lab 8 - reaction timer, Lab 9 - Inspecting your Morse. Lab 10 – Analogue fun, Resource Lab index, Code toolbox, LCD reference.



Alex animated head can be programmed from C for PICmicros using the Actuators training panel

## **This product is shipped with the SourceBoost C2C C compiler. The specification of this is as follows:**

The C2C compiler is a general purpose compiler-cum-IDE. It supports the following functions:

Supported features:

8-bit and 16-bit variables

Pointers

Built in variables

Arrays

Arithmetic expressions

Functions and built in functions

Interrupts

Standard C commands

Libraries and include files supported

Multiple data types

Built in functions:

clear\_wdt, enable\_interrupt, disable\_interrupt, set\_mode, set\_option, set\_tris\_a, set\_tris\_b, set\_tris\_c, output\_port\_a, output\_port\_b, output\_port\_c, output\_high\_port\_a, output\_high\_port\_b, output\_high\_port\_c, output\_low\_port\_a, output\_low\_port\_b, output\_low\_port\_c, input\_port\_a, input\_port\_b, input\_port\_c, input\_pin\_port\_a, input\_pin\_port\_b, input\_pin\_port\_c, sleep, nop, set\_bit, clear\_bit, putchar, getchar, delay\_s, delay\_ms, delay\_us, char\_to\_bcd, bcd\_to\_char

Standard C commands:

- if, else, while, for, return, break, continue, extern, switch, case, default;
- goto and labels;
- char, short, int, long, void;
- ~, ++, --, +, -, <, <=, >, >=, ==, !=, =, !, &, |, ^, &=, |=, ^=, &&, ||, \*, /, %, <<, >>, <<=, >>=;
- one-dimensional arrays; · const char pointers; · const variables and arrays;
- functions with no/one/many parameters and void/char return type; built-in assembler;
- #include
- #define, #undef
- #ifdef, #ifndef, #else, #endif

Data types

Decimal, Octal, Hexadecimal, Binary, ASCII

The following devices are supported:

IC12c508, PIC12c508a, PIC12c509, PIC12c509a, PIC12ce518, PIC12ce519, PIC12cr509a, PIC12c67, PIC12c672, PIC12ce673, PIC12ce674, PIC12f629, PIC12f675, PIC16c505, PIC16c52, PIC16c54, PIC16c55, PIC16c56, PIC16c57, PIC16c58, PIC16c61, PIC16c62, PIC16c62a, PIC16c62b, PIC16c63, PIC16c63a, PIC16c64, PIC16c64a, PIC16c65, PIC16c65a, PIC16c65b, PIC16c66, PIC16c67, PIC16c68, PIC16c620a, PIC16c621, PIC16c621a, PIC16c622, PIC16c622a, PIC16c641, PIC16c642, PIC16c661, PIC16c662, PIC16c71, PIC16c72, PIC16c72a, PIC16c73, PIC16c73a, PIC16c73b, PIC16c74, PIC16c74a, PIC16c74b, PIC16c76, PIC16c77, PIC16c710, PIC16c711, PIC16c712, PIC16c715, PIC16c716, PIC16c717, PIC16c745, PIC16c765, PIC16c770, PIC16c771, PIC16c773, PIC16c774, PIC16c84, PIC16c923, PIC16c924, PIC16ce623, PIC16ce624, PIC16ce625, PIC16cr54, PIC16cr56, PIC16cr57, PIC16cr58, PIC16cr62, PIC16cr63, PIC16cr64, PIC16cr65, PIC16cr620a, PIC16cr72, PIC16cr83, PIC16cr84, PIC16f627, PIC16f627a, PIC16f628, PIC16f628a, PIC16f648a, PIC16f630, PIC16f676, PIC16f72, PIC16f73, PIC16f74, PIC16f76, PIC16f77, PIC16f737, PIC16f747, PIC16f767, PIC16f777, PIC16f83, PIC16f84, PIC16f84a, PIC16f87, PIC16f88, PIC16f818, PIC16f819, PIC16f870, PIC16f877, PIC16f872, PIC16f873, PIC16f873a, PIC16f874, PIC16f874a, PIC16f876, PIC16f876a, PIC16f877, PIC16f877a