8051/251 Development Tools

Optimum Device Support

The Keil Cx51 ANSI C Compiler supports all classic and extended 8051 device variants. Compiler extensions provide full access to all CPU resources and support up to 16MB memory. Keil Cx51 generates code with the efficiency and speed of hand-optimized assembly. New compiler and linker optimizations shrink programs into the smallest single chip devices.

The Keil µVision2 IDE fully integrates Cx51 Version 7 and provides control of the Compiler, Assembler, Real-Time OS, Project Manager, and Debugger in a single, intelligent environment. With support for all 8051 derivatives and full compatibility with emulators and third party tools, Cx51 Version 7 is clearly the best choice for your 8051 projects.

This brochure provides details about all Keil Cx51 Compilers including:
- C51 C Compiler for Classic 8051 Devices
- CX51 C Compiler for Philips 51MX Devices
- C251 C Compiler for 251 Devices

µVision2 integrates project management, development, and debugging.
Development Tools Overview

8051 Device Support

The Keil Cx51 Compiler is the de facto industry standard tool set for the 8051 microcontroller family. It is available in three different packages:

- The **CA51 Compiler/Assembler Kit** is designed for classic 8051 projects and includes the µVision2 IDE, C Compiler, Macro Assembler, and Code Banking Linker.
- The **DK51 Developers Kit** includes all components of the CA51 plus the µVision2 Debugger, complete device simulation, and target monitor.
- The **PK51 Professional Developers Kit** includes all components of the DK51 plus extended memory support (up to 16MB), variable banking, real-time operating system, linker optimizations, ISD51 in-system debugger, Dallas 390 contiguous mode support, and Philips MX support.

251 Device Support

The advanced 251 architecture adds 16-bit and 32-bit instructions as well as stack-based addressing to the 8051. Standard devices are available from Atmel, Intel, and Sanyo. IP cores are also available.

The Keil C251 Compiler is the most efficient 251 tool chain available. 8051/251 combo kits are available for developers who want to use both architectures.

Device Database™

The Device Database™ contains detailed information about the devices supported by the Keil development tools.

- The web-based Device Database™ provides parametric searching and helps locate chips that meet specific requirements.
- In µVision2 projects, all required options are automatically set when you select the device from the Device Database™. Dialogs are customized to display only relevant options and prevent you from selecting incompatible directives.

<table>
<thead>
<tr>
<th>Keil 8051 Development Tool Kits</th>
<th>PK51</th>
<th>DK51</th>
<th>CA51</th>
</tr>
</thead>
<tbody>
<tr>
<td>µVision2 IDE</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>C51 Compiler</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>MACRO Assembler</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>BL51 Code Banking Linker</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>CX51 Compiler</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>LX51 Linker/Locater</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Support for Extended Devices</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Analog Devices MicroConverters</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Dallas 390, and the Philips 51MX</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>µVision2 Debugger/Simulator</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>8051 Target Monitor</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>ISD51 In-System Debugger</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>RtxTiny2 Real-Time Kernel</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>


Device Database™ contains details used to configure the tools and links to other sources of information (like data sheets, board, emulators). A complete list of all supported chips along with technical information is available at [www.keil.com/dd](http://www.keil.com/dd).
Project Management

A project is composed of all the source files, development tool options, and directions necessary to create a program.

A single µVision2 project can generate one or more Target programs, each with its own set of options. The source files used to create a target are organized into Groups. Options may be set at the target, group, or file level.

Tool setup is simplified with intelligent options dialog pages. For example, the linker configuration is derived from the off-chip memory address specifications and the on-chip memory sizes from the Device Database™.

Source Code Editor

The µVision2 editor includes all the editing features to which you are accustomed. Color syntax highlighting and text indentation are optimized for C and assembler source code. Editing is even available while debugging so you can quickly test and make changes to your application.

Integrated Utilities

µVision2 contains powerful features that help you complete your projects on time:

- The integrated Source Browser maintains a database of your program’s symbols for quick navigation through your source files. You may use the detailed information to optimize the memory types of variables.
- Find in Files performs a global text search in all specified files.
- The Tools Menu allows you to start third-party utilities from within the µVision2 IDE.
- A configurable SVCS interface provides access to Version Control Systems.
- The PC-Lint interface gives you advanced syntax analysis of your application code.
- The Flash tool interface integrates third party flash download utilities.

The Edit, Build, and Debug toolbar provides quick access to important commands.

Attributes Column

Bookmark
No Executable Code
Enabled Breakpoint
Code already executed
Current PC, next statement
Disabled Breakpoint
Selected line from Output Window
Code is not executed

The Attributes Column in the Editor Window provides status of each source code line.

The Integrated Source Browser provides fast navigation to definitions and references.

Right-click to open context-sensitive (local) menus in local windows. For example, in the editor windows the local menu allows you to access browser information and debugger commands.
The Keil 8051 and 251 development kits contain different C compilers for optimum support of the various device variants. C51 is used as generic term for these compilers:

- C51 Compiler for classic 8051 devices
- CX51 Compiler for Philips MX extensions
- C251 Compiler for the 251 architecture

**New Code Optimizations**

C51 Version 7 adds new optimizations that generate smaller programs and help you put more features in less code space (so you can fit larger programs in single-chip devices).

- **Dynamic Register Allocation** puts more variables in registers and reduces code (fewer MOV instructions are generated) and data space (less overlaid data is required).
- **Common Tail Optimization** combines identical code in conditional statement blocks and further reduces code space.

**Memory and SFRs**

The C51 compiler provides complete access to all hardware components.

- Access to special function registers is provided by the `sfr` and `sbit` keywords.
- Variables may be assigned to any address space.
- The `at` keyword allows you to locate variables at fixed memory addresses.

The Linker supports code banking and the µVision2 Debugger supports up to 16MB of code and xdata space for program testing.

**Flexible Pointers**

C51 provides two kinds of pointers.

- **Generic pointers** store the memory type and the address of an object in three bytes and may access any variable regardless of its location in the 8051 memory space.
- **Memory-specific pointers** are declared with a memory type and refer to a specific memory area. Addresses are 1-2 bytes so pointer accesses are fast.

**C51 Highlights**

- Support for all 8051 derivatives and variants
- Fast 32-bit IEEE floating-point math
- Efficient interrupt code and direct register bank control
- Bit-addressable objects
- Sophisticated syntax checking and detailed warnings
- Use of AJMP and ACALL instructions
- Memory banking support to expand the 64KB limit
- C-level access to all 8051 SFRs
- Register parameters and dynamic register variables
- Global program wide register optimization
- Common code block subroutine optimization
- Integrated variable banking support †
- Use of multiple data pointers
- Use of on-chip arithmetic units
- Support for extended device variants
- Generic and memory-specific pointers
- Reentrant functions and register bank independent code
- Debug information support for all emulator vendors
- Simple assembly language interface

**Memory Models and Memory Selectors**

Three well-defined memory models determine the default memory selector used for variables. However, you may always explicitly specify a memory selector for any variable to optimize your application.

<table>
<thead>
<tr>
<th>Selector</th>
<th>Address Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>128 Bytes directly addressable on-chip RAM.</td>
</tr>
<tr>
<td>bdata</td>
<td>16 Bytes bit-addressable RAM; mixed bit/byte access.</td>
</tr>
<tr>
<td>idata</td>
<td>256 Bytes indirectly addressable on-chip RAM.</td>
</tr>
<tr>
<td>pdata</td>
<td>256 Bytes <em>paged</em> external RAM.</td>
</tr>
<tr>
<td>xdata</td>
<td>64KB extended RAM space.</td>
</tr>
<tr>
<td>code</td>
<td>64KB program memory.</td>
</tr>
<tr>
<td>far †</td>
<td>16MB data/const memory on extended devices or variable banking on classic devices; 64KB object size.</td>
</tr>
<tr>
<td>near ‡</td>
<td>64KB direct addressable memory on 251 devices</td>
</tr>
<tr>
<td>huge ‡</td>
<td>16MB indirectly addressable memory; any size object.</td>
</tr>
<tr>
<td>edata ‡</td>
<td>96 Bytes extended bit-addressable RAM on 251</td>
</tr>
</tbody>
</table>

† Only available with the LX51 Extended Linker.
‡ Only available in the C251 Compiler.
LX51 Extended Linker

The LX51 Extended Linker adds functionality to the Cx51 Compiler. For example:

- **Linker Code Packing** analyzes the entire application and generates common code blocks even on code banking applications.
- **AJMP** and **ACALL** instructions replace longer **LJMP** and **LCALL** instructions when possible.
- **Far memory support** allows access to 16MB memory even on classic 8051 devices.
- **Detailed data type checking** is performed across all public symbol definitions.
- **Incremental linkage** is available for Flash download and multi-application support.

ISD51 In-System Debugger

ISD51 is a new debug monitor technology which works in all standard 8051 target systems and which requires no hardware modification. ISD51 is a small module (only 530 bytes) that links with your application.

Using a standard 8051 serial port, ISD51 allows real-time program execution, single-stepping, multiple software breakpoints, and access to all memory spaces including CPU registers and SFRs.

RtxTiny2 Real Time Kernel

The RtxTiny2 multitasking real-time kernel makes implementing complex, time-critical software projects easy. RtxTiny2 is royalty-free and is fully integrated into the Keil Cx51 tool chain. RtxTiny2 works on all classic 8051 device variants and supports even multiple DPTR and arithmetic units.

RtxTiny2 is the successor of the popular RTX51 operating system and provides:

- Single chip and code banking support.
- Round robin and cooperative task switching.
- Task management with create and delete.
- Timeout, Signal, and Ready events.
- Interrupt support for sending signals.

Extended 8051 Device Overview

- **Philips 51MX** provides 8MB code and 8MB xdata space and adds CPU instructions for generic pointer access.
- **Dallas 390** provides 16MB code and 16MB xdata space and adds math accelerator and 24-bit JMP instructions.
- **Analog Devices MicroConverter** provides 16MB xdata space and a high-precision A/D converter.
- **Atmel** provides several 8051 derivatives with on-chip EEPROM that may be accessed using the far memory types.
- **Mentor** provides the M8051EW IP core with integrated code banking and extended variable space.

The PK51 Professional Developers Kit fully supports all extended 8051 device variants and advanced CPU features.
The µVision2 Debugger provides source-level, symbolic simulation and target debugging. The complete instruction set and on-chip peripherals of the selected device are simulated. Dialogs allow you to view and change peripheral settings.

For high-fidelity simulation of your hardware, Keil Software offers the Advanced Generic Simulation Interface (AGSI). AGSI is an API specification that extends the simulation capabilities of the µVision2 Debugger with application-specific dialogs and peripheral support. Numerous third-party AGSI DLLs are already available.

**Breakpoints**

µVision2 allows you to set program breakpoints while editing (even before the source code is compiled or assembled). Breakpoints may be set on program code, conditional expressions, or variable/memory accesses. They are enabled when you start the µVision2 Debugger. Triggered breakpoints can halt program execution or run a debugger command or function script.

**Debug Function Scripts**

µVision2 incorporates a C-like function language that allows you to generate and use:
- Built-in functions like printf and memset.
- Signal functions to simulate analog and digital inputs to the CPU.
- User functions to extend the command scope and combine repetitive operations.

**Variables and Memory**

µVision2 offers several ways to view and change variables and memory.
- Move the mouse pointer over a variable to view its value.
- Use Watch Window to view and change local variables and user-specified variables.
- Use the Memory Window to view and change up to four memory areas.
The Keil µVision2 Debugger interfaces to target hardware via Advanced Generic Debugger Interface (AGDI) drivers. Keil provides the following drivers:

- **Monitor-51** configurable target monitor. This monitor is integrated into user hardware and is provided on many third-party evaluation boards.
- **Monitor-390** configurable target monitor for Dallas contiguous mode.
- **ISD51** in-system debugger. ISD51 runs on standard 8051 hardware and is linked with the user application.
- **EPM900** emulator/programmer for Philips LPC900 device series.
- **SmartMX DBox** hardware for Philips SmartCards.

Several third-party products interface via AGDI drivers to the µVision2 Debugger:

- ChipCon **CC1010** RF Transceiver.
- JTAG interface for **Cygnal 51Fxxx** series.
- Cypress **USB development kits**.
- Infineon **SLE66** SmartCard ROM Monitor.
- **SST SoftICE** for FlashFlex51 devices.
- JTAG interface for **Triscend E5** series.

In addition, many emulator vendors offer µVision2 AGDI drivers for their emulators.

### MCBx51 and MCB517

The MCBx51 and MCB517 Starter Kits are a great way to get started with your 8051 or 251 project. Each Starter Kit includes 2KB-limited development tools and numerous sample programs which are ready-to-run. You can easily test the 8051 or 251 device performance and the flexibility of our development tools at the same time.

- MCBx51 supports numerous 44-pin PLCC 8051 and 251 devices including many high-speed versions.
- MCB517AC includes the high-performance Infineon C517A, the 81C90 CAN controller, and code banking support.
Advantages of Cx51 and Related Tools

- New optimizations in Cx51 generate code that approaches the efficiency of hand-tuned assembly.
- Cx51 supports all 8051 variants and provides access to all hardware components.
- Cx51 generates very fast interrupt code with or without 8051 register bank switching.
- Cx51 supports multiple DPTR and arithmetic units.
- Cx51 assigns register variables and performs global register optimization on an entire application.
- Linker optimizations maximize AJMP and ACALL instructions and generate common code subroutines.
- All tools generate detailed warning and error messages to help you locate hard-to-find problems.
- Cx51 supports reentrant functions and register bank independent code for interrupt service routines and multitasking applications.
- Efficient memory banking and debugging let you easily expand the 64KB space limit.

Advantages of µVision2

- Easy-to-use integrated development environment provides intelligent setup for tool options.
- High-speed CPU and peripheral simulator with drivers for numerous 8051 derivatives.
- C-like language for creating user and signal functions.
- Integrated performance analyzer and code coverage.
- Interfaces to target monitor and in-system debuggers.

15 Years of Keil Cx51

Since its market introduction in 1988, the Keil Software Cx51 Compiler has become the leading C compiler for the 8051 microcontroller family. No other 8051 compiler generates more efficient code for your embedded applications. That’s why more than 40,000 users worldwide trust the Keil advantage for their 8051 and 251 projects.

Cx51 Version 7 sets new standards for code density. When compared to C51 Version 6, the new optimizations of Cx51 Version 7 reduce the code size of your applications by 5% to 10%.

Keil Elektronik GmbH and Keil Software, Inc., have representatives in the following countries:

Australia, Austria, Belgium, Brazil, Canada, China, Czech, Denmark, Finland, France, Germany, India, Ireland, Israel, Italy, Japan, Korea, Malaysia, Mexico, Netherlands, New Zealand, Norway, Poland, Portugal, Russia, Romania, Singapore, Slovenia, Slovakia, Spain, South Africa, Sweden, Switzerland, Taiwan, Turkey, Thailand, United Kingdom, and United States.

Contact Keil Elektronik GmbH or Keil Software, Inc., for the distributor nearest you.