

NEW
'Professional'
Edition

Software for TERN C/C++ programmable controllers

Programming TERN controllers — 3 easy steps

**A \$2000
Value**



EV-P C/C++ Evaluation Kit \$249

(Includes: Paradigm C/C++ PRO Lite TERN Edition 32-bit Windows IDE, choice of one Tern controller, hardware accessories/basic options)

DV-P C/C++ Development Kit \$699

(Includes: Paradigm C/C++ PRO TERN Edition 32-bit Windows IDE, choice of one Tern controller, hardware accessories/basic options)

***Upgrade from EV-P to DV-P** \$450

(includes software in CD, Password & one Debug ROM for controller specified by customer) *Pricing may be higher on some high-end controllers

Step 1 Debugging

- Program in C/C++, compile, locate, and remote debug
- Connect controller to a PC via RS-232 serial link

Step 2 Standalone Field Test

Run controller standalone, away from PC, with

- DEBUG ROM installed, and STEP2 jumper set
- application program running in battery-backed SRAM
(Battery lasts 3-5 years under normal conditions.)

Step 3 Production (DV-P Kit only)

- Generate an application HEX or BIN file
- Produce your own ROM/Flash

Developers today face a bewildering range of embedded application development options. Even with project requirements that seem straight-forward, developers often find their first challenge has nothing to do with their actual design. On lower-end systems, developers first have to plan their strategies for squeezing the necessary logic into a few kilobytes worth of obscure machine code. On higher-end, 32-bit PC-derived operating systems, embedded engineers often find themselves evaluating complicated features that have almost no relevance to their immediate application (Plug-n-Play support? Network boot daemons? Windowing systems? Kernel build configuration?).

And there's no source of greater headache than being forced to combine hardware, compiler, runtime, and debugger development tools from different vendors. There's no doubt about it: getting started on a new embedded programming project can be complicated and confusing.

TERN's software kits simplify the entire process. They include all of the software, hardware, accessories, and manuals needed to complete your application development on any controller of your choice. With the TERN development kits, you will be able to run and debug sample programs on your controller within minutes of opening the box, installing the software on your PC, connecting the provided serial cable, and powering up the board.

The EV-P Kit is for the first-time buyer looking to make a prototype stand-alone unit for evaluation and field testing. It supports STEP 1 and STEP 2, but not STEP 3 of the development cycle. With the EV-P, you will still be able to compile, download, and debug your application code. You even have the capability to run your code out of the battery-backed SRAM for initial stand-alone testing. The DV-P Kit will be needed to generate ROM-able application files, necessary when you go into OEM production for your system (STEP3 of the development cycle). If you began with the EV-P

kit for evaluation purposes, a software-only upgrade is available for upgrading your system to the full DV-P.

The TERN kits include:

- ✓ software tools needed for the entire development process from beginning to end,
- ✓ choice of production hardware controller, ready to be debugged,
- ✓ accessories, including DB-9 serial cable, wall transformer,
- ✓ documentation for all TERN products, specifications for peripheral parts/processors used on TERN products.

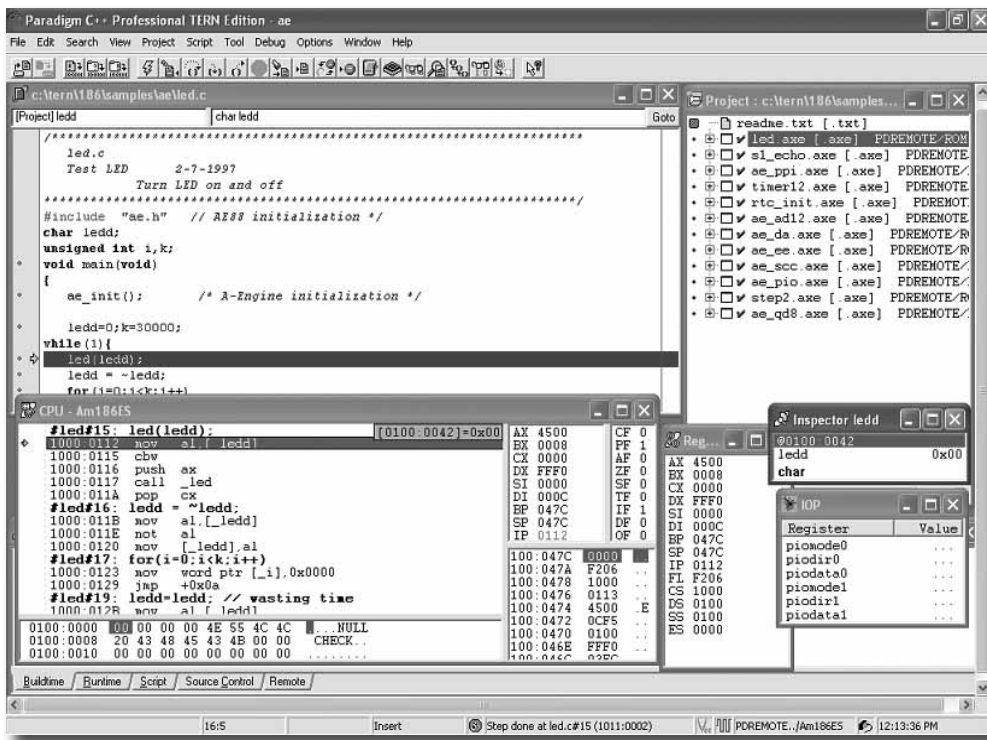
Software

The EV-P and DV-P kits provide comprehensive, professional software tools for building embedded applications. These software kits provide a modern 32-bit Windows-based C/C++ Integrated Development Environment (IDE) that hosts the entire development process in one location.

The newest **Paradigm C++ Professional TERN Edition** is now available for TERN customers. This new edition introduces a number of bug fixes and new features. The Professional edition is compatible with all Windows environments, including 64-bit Windows 7.

The IDE consists of:

- A modern editor with syntax highlighting, header file expansion.
- A Project Manager for tracking build dependencies in large projects, and managing source version control.
- Built-in class/method/struct symbol database.
- Integrated ANSI C and C++ compiler/assembler, executable re-locator.
- Remote debugger integrated with the editor: watch windows, dynamic expression evaluation, conditional breakpoints.



You can see the development system in action on the page 5 screen-shot. This screen-shot shows the sample program *led.c* being debugged. This basic C program, which blinks the LED, is running directly on the TERN controller while the debugger retains full control over the execution.

New in updated Professional edition:

- Plain-text project format for easier configuration.
- Windows compatible HELP files.
- Visual Assist can help correct and highlight problems; case is automatically fixed.
- AutoCompletion works for every symbol in your code. It understands everything necessary to complete symbols accurately.
- Debugger/compiler improvements. Better breakpoints, logging, and more.

The C/C++ kits provide a variety of software solutions to assist you in developing your project. At the lowest level, these kits include a set of hundreds of C library functions (like the *led()* function used

above) that allow you to easily access the features of your TERN controller, such as the serial I/O, ADC, DAC, Flash, graphics LCD, timers, interrupts, etc. TERN also offers higher-level software runtime libraries for more complicated application requirements (see previous page for Software Runtime details).

A wide range of sample programs for every controller demonstrates the use of these library functions and hardware features, allowing you to quickly and easily develop your application. Even those with basic C/C++ knowledge will find development to be straight-forward.

Hardware

The TERN EV-P or DV-P kits include *any* one basic TERN controller of your choice, allowing you to start your implementation immediately. The boards that you're provided are not modified "prototype" boards; these are the same production units that you will be able to purchase indefinitely into the future.

Select one of the controllers that fits your needs most closely, and we will add the basic options needed for a working prototype. You can also add any of the other options available for that controller.

For example, if you selected the AE-P controller, the Kit (*EV-P-AEP40*, \$249 or *DV-P-AEP40*, \$699) includes a 40 MHz AE-P with 128KB SRAM, RS232, 5V regulator, DEBUG ROM, wall transformer, serial cable and software in a CD. If you need it, add the 11 channel 12-bit ADC option for an additional \$20.

International customers must specify 110V or 220V AC power supply for your wall transformer.

Support

TERN takes pride in intimate technical support for all of our development customers. Our customers are our partners, and we will do our best to assist in your project development process. Whether it's a simple compilation error, or a complicated request for new sample code, TERN is always accessible via email at tech@tern.com. You may also find help online at: <http://groups.yahoo.com/group/tern>.

Custom Application Development

TERN's ultimate commitment is in helping customers develop their application successfully. *Our success is intertwined with the success of your project.* The C/C++ software kits make this task easier, but there can still be a substantial learning curve involved in developing embedded applications for a new platform; even quick application prototyping can be a frustrating experience.

For those most interested in getting started quickly in the development of their applications, TERN now offers professional custom software development and board design. Engineers at TERN and our partners have accumulated thousands of hours worth of experience developing applications for use on TERN

controllers, and you can leverage our experience to increase your own productivity.

We will work with you to develop an initial application specification (including controller selection/design). Based on this specification, our engineers will implement a prototype of the application. The fully commented source code and a working hardware demonstration is provided to you once we complete development. Our focus is on making the code reliable and easily modifiable to fit your changing application needs over time. We will provide support to make sure the deliverables can be integrated into your system successfully. *TERN can also help port (re-code) applications implemented in other languages or for other platforms for use on TERN systems.*

Contact us to discuss your application needs and ask for a quote!