

RAD51 Reference
Document Revision 0.2

■ *A Complete
Reference to Using
the Systronix RAD51
Rapid Application
Development Environment
and 8051 Cross Assembler*

RAD51

LIMITED WARRANTY

The information in this manual is subject to change without notice and does not represent a commitment on the part of Systronix, Inc. Systronix, Inc. makes no warranty, express or implied, for the use or misuse of its products, which are provided with the understanding that you, the user, will determine fitness for a particular application. Systronix assumes no responsibility for any errors which may appear in this manual. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose other than the purchaser's personal use without the written permission of Systronix, Inc.

Systronix reserves the right to revise this documentation and the software and hardware described herein or make any changes to the specifications of the product described herein at any time without obligation to notify any person of such revision or change.

TRADEMARKS

Systronix is a registered trademark of Systronix Inc, INT_EL and Intel are registered trademarks of Intel Corporation, Microsoft and MS-DOS are registered trademarks of Microsoft Corporation.

Systronix[®], Inc.
555 South 300 East #21
Salt Lake City, UT 84111
TEL: 801-534-1017
FAX: 801-534-1019
Internet: www.systronix.com
email: info@systronix.com

Copyright © 1999 by Systronix[®], Inc.
All rights reserved.

Revision 0.1 - printed July 6, 1999

CREDITS and a BIT of RAMBLING FROM the AUTHOR

This manual was created using Corel WordPerfect 8.0 on an NT4 workstation with 256 Mbytes DRAM. I remember our first real Systronix computer - an IBM PC/AT with a 6 MHz 286 and 128 Kbytes of DRAM (wow!), \$3000 plus \$700 for a 30 Mbyte hard disk. It had a 'high-resolution' (720x400) Hercules graphic card with an amber AMDEK monitor. Network Services by Christopher Robin. Review and input by Scott Kendall and Jared Wyckoff. Win32 programming by Steve Hsieh & associates.

- Bruce Boyes, Systronix, Inc.

*RAD51
Technical
Reference*

*Systronix, Inc.
Complete Solutions for Rapid Development
of Embedded Control Systems*

Document Revisions

1. 0.1 Started, to ship with initial public release of RAD51
2. 0.2, added description of project management along with screen captures.

Table of Contents

RAD51 - READ THIS FIRST	1
Errata	1
Beta 1.0.7	1
Beta 1.0.4.4	1
Terminology in This Manual	2
What You Need to Use RAD51	2
Quick Start - Step by Step	2
Install RAD51 on your PC from CDROM	2
Install RAD51 from a single file from the Internet	2
RAD51 DETAILED DESCRIPTION	3
Philosophy and Purpose of RAD51	3
RAD51 Versions	3
RAD51 Features	3
Systronix Web Site & Forum	3
Getting Technical Support	4
Getting the Most out of RAD51	5
Project Management and Folder Layout	5
Folder and Assembly Code File Organization	5
RAD51 Projects	6
Creating a New Project from Scratch	6
Creating a new project by copying an existing .ASM file	7
Controller Selection	7
Relative Output File Paths	8
PRJ and Main ASM Files in Same Folder	8
Shared/Common Code in a Parent Folder	8
Project File Tree Pane	9
Project Symbol Tree Pane	10
Project Resource and Info Panes	10
File Menu: Files vs Projects	10
RAD51 Help, Configuration and Options	10
Assembler and RAD51 Help & Help Files	10
Controller Selection and Controller.cfg file	10
Editor Configuration and Options	11
TROUBLESHOOTING & DEVELOPMENT TIPS	12
Internet FAQ	12
Start Simple	12
Learning Assembly Code and Embedded Programming	12
Exception Handling	12
Quick Diagnosis Table	13

RAD51

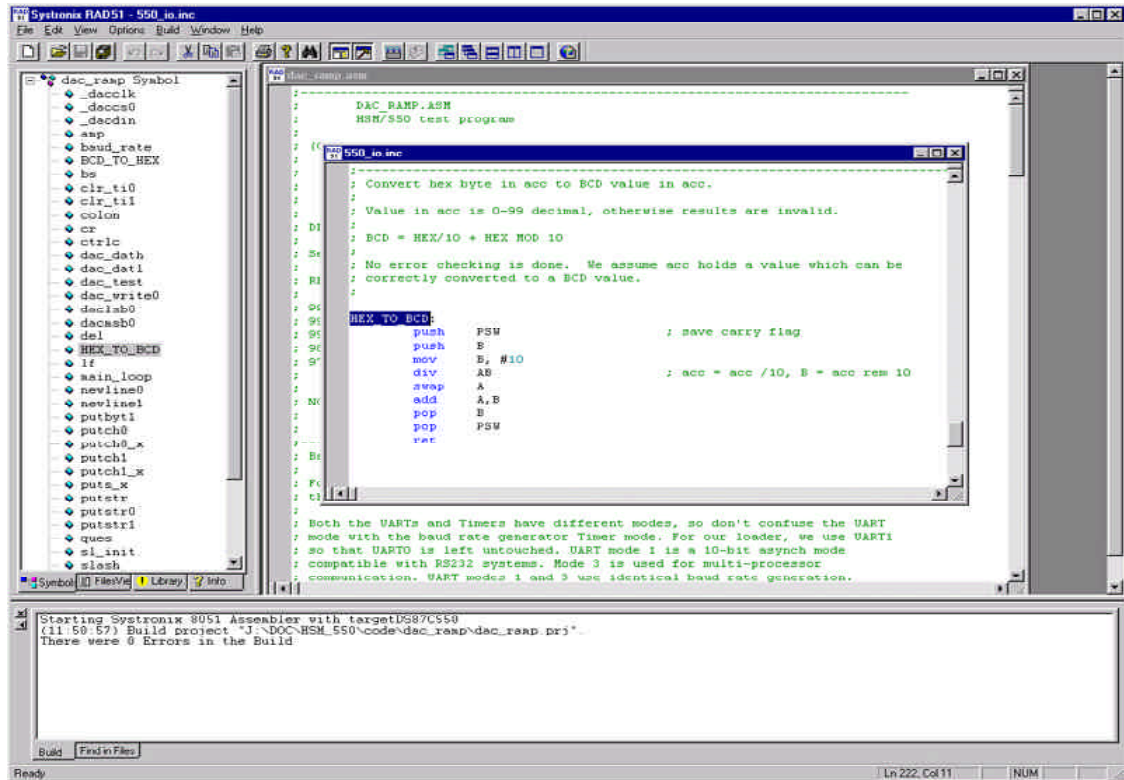
Beta Release
1999 June 30

SYSTRONIX®

Rapid 8051 Assembly Code Development

The new Rad51 IDE and 8051 assembler helps you develop 8051 assembly code more quickly.

Notice the symbol view in the left pane. The highlighted symbol was clicked which moved the source editor highlight to the symbol's definition in one of the project include files.



Write 8051 assembly code in a 32-bit Windows environment.

Text Editor. Symbol table view. Cross assembler. Code management tools.

Only \$299.FREE!

At last there is an 8051 cross-assembler for Windows 95/98/NT. And, at least for now, the base RAD51 product is FREE.

RAD51 is extensible, and we plan to offer application-specific plugin wizards for sale in the future, as well as publish the API.

Rather than spend a lot of money on advertising, we thought that giving RAD51 away might engender good will and help us tell the world about Systronix Rapid Development Systems.

Example programs with source code are included.

- Programmer's text editor with standard features such as search and replace, syntax highlighting, etc.
- Extensible controller definition file - modify it yourself or download the latest for free at www.systronix.com
- Symbol table view lists all project symbols. Double clicking takes you to the symbol definition in any project file.
- File tree view lists all project files. Double clicking opens that file in the editor.
- Clicking on errors in the output window takes you to the offending line in the source editor.
- Real technical support included!
- Latest info & free download at www.systronix.com.

What is RAD51?

RAD51 is an integrated development environment and cross-assembler for the 8051 family.

Why Did Systronix Develop RAD51?

We wanted an extensible software platform on which to build 8051 code development tools. We also saw the lack of a good, free, Windows cross-assembler. In the recent past we flushed \$\$\$ down the drain on postcard mailings and other unproductive advertising. So instead of doing that again, we thought "why not develop our 8051 code platform and then give away the base product?" Maybe we'll just be flushing more \$\$\$ down the drain, but at least we'll get a useful tool out of the deal. So there you have it.

32-bit Windows Required

RAD51 makes use of the 32-bit Windows operating systems and will not run on other platforms. It is a true 32-bit object oriented application. RAD51 is available for download from the Internet or on CD-ROM only.

Standard HEX and LST file output

RAD51 emits standard Intel HEX files and an assembled output listing.

Add other 8051 Controller Definitions

RAD51 uses an ascii file CONTROLLER.CFG, which contains SFR and BIT definitions for controllers which are supersets of the generic 8031 device. You can easily add your own, and download the latest version from the web site.

Project Management

RAD51 uses PRJ project definition files. RAD51 knows which files your project includes and maintains a complete list of all project symbols. With a little bit of planning, RAD51 projects can easily share common files such as I/O drivers, and maintain a hierarchical structure of code folders on your hard disk. This eases maintenance and use of common code. Sorry - RAD51 won't generate PERT charts or schedules.

Easy Installation

RAD51 uses InstallShield for easy installation. It is Windows compatible, meaning all configuration is implemented in the PRJ file or Windows registry (not INI, autoexec or config files), and it includes an uninstaller.

RAD51 IDE is Extensible

RAD51 was developed with an extensible interface to ease plugging in simulators, downloaders, application-specific wizards, etc. We intend to publish the API. We are still working on this so please be patient. It will be released on the web site when ready. We have no specific delivery date for the API at this time.

How do I order?

Download RAD51 at www.systronix.com or [ftp.systronix.com](ftp://ftp.systronix.com). RAD51 also ships with all current Systronix development hardware on a CD-ROM. If you prefer, you can order the CD-ROM for a nominal charge. Check our website to see if on-line ordering is up yet, otherwise call, Fax, or email.

TECHNICAL DETAILS

IDE Programmer's source code editor with standard MDI functions and configurability. Syntax highlighting. Output window and

Assembler 8051 cross-assembler supports 64 KBytes address space. No macros. No linker or librarian. The assembler is implemented as a DLL which plugs into the IDE.

File View File tree of all project files. Double click on a file to open it in the source editor.

Symbol View Building a project generates a complete symbol table list, including symbols in all project files. Double click on a symbol to jump to it's definition in the source code.

Output View Build results are displayed in the output pane. Double click on an error to go to the offending source code line.

ICE symbol & debug tables Currently not generated, but this may change in the near future.

Requirements Windows 95/98/NT, and an Internet connection or CD-ROM drive. RAD51 is available for Intel-compatible hardware platforms only.

Price :

Internet download: Free!

CD-ROM: \$20, including priority mail shipping.

(Buy lots of Systronix hardware and help keep RAD51 free forever.)

All systems include:

- Assembler and RAD51 technical references as indexed PDF files (free Adobe Acrobat Reader 4.X recommended)
- Sample programs
- Tech support (please check the web site FAQ and updates area and try email before calling).

SYSTRONIX®

555 South 300 East #21, Salt Lake City, Utah, USA 84111
Tel:+1-801-534-1017 Fax:+1-801-534-1019 www.systronix.com

revised 1999 jun 30 bab

RAD51 - READ THIS FIRST

If you really hate reading manuals, then install the program and go to work. If you get stuck, come back here and/or check the troubleshooting section or the FAQ at www.systronix.com.

■ *Errata*

Beta 1.0.7

Three digit revision number.

All PRJ file paths can be relative so that you can copy projects to other folders, and we can include sample projects with the RAD51 installation!

RAD51 help file (this document) available from the RAD51 help menu.

Beta 1.0.4.4

Link to the RAD51 help file is grayed out and you cannot access the help file from the RAD51 IDE. It is accessible from the Windows START menu in your RAD51 installation directory.

It's possible to choose a new BASIC Project Type but there is no support at this time for such projects. This item should be grayed out.

Saving to file types not defined in your registry (such as .inc) doesn't work. If there is a registry key for .inc on your computer then it will work.

Controller selection causes problems building if no controller is selected. A build is reported when in fact none occurs.

The InstallShield install program does not create a .PRJ association for RAD51 projects. Therefore, double clicking on a PRJ file will not invoke RAD51 on that project.

Errors in the 80C320 definition of controller.cfg

■ *Terminology in This Manual*

We use a fixed pitch font to represent what you see or type on your PC:

```
C:\RAD51\DIR *.EXE
```



We use the pointing hand to call attention to something worthy of special note, such as a common pitfall or interesting feature of RAD51.

■ *What You Need to Use RAD51*

- ✓ A “Wintel” PC with 32-bit Windows such as Windows 95, Windows 98, or Windows NT 4.X. RAD51 will not run under Windows 3.X.
- ✓ Some knowledge of 8051 assembly code. Dallas Semiconductor, Intel, Philips and other 8051 vendors publish printed and on-line 8051 manuals and references. Personally, we prefer the Intel printed books - they are detailed and concise. Try contacting an Intel literature office and request the “8-Bit Embedded Controllers” data book. Other 8051 references are detailed later in this manual.
- ✓ To understand the High Speed Microcontroller family, you need the Dallas High Speed Microcontroller data sheets and application notes, available from www.systronix.com, www.dalsemi.com, or by calling Dallas Semiconductor at 972-371-4000. We’ve included some Dallas data sheets in PDF format on your RAD51 CD-ROM.

■ *Quick Start - Step by Step*

Install RAD51 on your PC from CDROM

Insert the CD-ROM, and it should auto-launch the installation program. Otherwise, you can execute “launch.exe” in the root menu, from the Start->Run menu or from Explorer. This invokes a master install menu which includes such install options as Adobe Acrobat Reader.

Install RAD51 from a single file from the Internet

Download the file from our web site and save it in a temporary directory on your local hard disk or network disk. The file is called “setupex.exe”. Double-clicking on the file will invoke the Installshield wizard for RAD51. This won’t install Acrobat Reader.

RAD51 DETAILED DESCRIPTION

■ *Philosophy and Purpose of RAD51*

The Systronix RAD51 is intended serve as the base platform for the coming generation of Systronix embedded control development tools.

■ *RAD51 Versions*

RAD51 is initially available in a freeware version. We're doing this as an advertising vehicle. We want more people to find out about Systronix and our other (non-free) products. We plan to keep a free version and eventually sell special options or plug-ins to the base product.

■ *RAD51 Features*

1. 32-bit native Windows application. Takes advantage of the unlimited memory and "flat" memory model in the 32-bit environment. Supports long filenames.
2. Extensible - able to accept tool wizards and plug-ins.
3. Processors defined in an ASCII controller.cfg file which can be easily expanded to accommodate new microcontrollers.
4. Tools to help you write assembly code programs. For example, RAD51 generates a symbol table in one of the project window panes. Click on any symbol and the editor takes you to the place of that symbol's definition.
5. Built-in text editor with syntax highlighting.
6. Project management features. For example, RAD51 supports an environment in which a base code folder holds frequently-reused code such as I/O libraries. Individual projects are subfolders below this folder. The projects include whichever of the common libraries they need. In this manner you have only one place to keep and maintain the common code, and a logically structured layout for each project.

■ *Systronix Web Site & Forum*

Our web site (www.systronix.com) is the main repository for new RAD51 example code and documentation. There are links from the web site to an ftp file area. You can also join a forum of Systronix users. Our web site has information about the forum..

■ *Getting Technical Support*

Our technical support is included with your purchase of RAD51. Actually, we are giving away RAD51 - but we will still support it. We believe good support begins with good written documentation (starting with what you're reading right now). If you can't find the answer in our documentation, then try the FAQ on the web at www.systronix.com, send e-mail to support@systronix.com, call us at 801-534-1017, or Fax us at 801-534-1019. When you contact us, please tell us about any errors or weaknesses in the documentation so that we can improve it in the next revision.

If you can, please contact us by e-mail first. You can attach a file of source code and captured output (use MIME encoding if possible) to your message. If you send us an example of a problem please make the example as simple as possible, and include any necessary I/O driver "include" files if you have modified them. *Please send us ASCII text or PDF files rather than word processor files.* Due to the risk of viruses, we cannot accept word processor files containing macros. We try to answer all e-mail within one or two business days.

Support does not include designing or writing code for you or debugging your application. We'll be happy to give you some general suggestions. Beyond that, if you need an application developed, we offer that service as well.

Please feel free to contact us with any unusual questions about programming RAD51. We can probably help you approach your needs in the most efficient way. That's why we're here! Customers consistently give us high marks for courteous, competent technical support. But, we're only human and sometimes we make mistakes. We listen to our customers (we've gotten some of our best product ideas from them!) so tell us how we're doing for you.

Getting the Most out of RAD51

■ Project Management and Folder Layout

RAD51 includes some simple project management features to help take some of the drudgery out of assembly code maintenance and programming. Once you get the hang of using it, we think you will really like it. We do! We use RAD51 here on a daily basis, so we have to put up with the same things you do! If something annoys us we will fix it. Check back with the web site periodically to get the latest version of RAD51.

RAD51 is designed to help you in partitioning and maintaining shared code such as I/O routines. RAD51 projects start with a *PROJECT FILE*. You can create a new project from scratch, or from an existing project. Here's how to do it, along with suggested project and code folder layout. The examples included with RAD51 demonstrate some of these principles.

Folder and Assembly Code File Organization

First of all, think about your project and what can, or should be, shared code: I/O routines (serial output specific to a processor or board), math routines, peripheral device access (LCD, keypad, ADC, DAC, etc). Here's a screen capture of how we organized the code for the Systronix HSM550 example projects. Shared code is in the "CODE" folder, with subfolders for each project which will use that shared code.

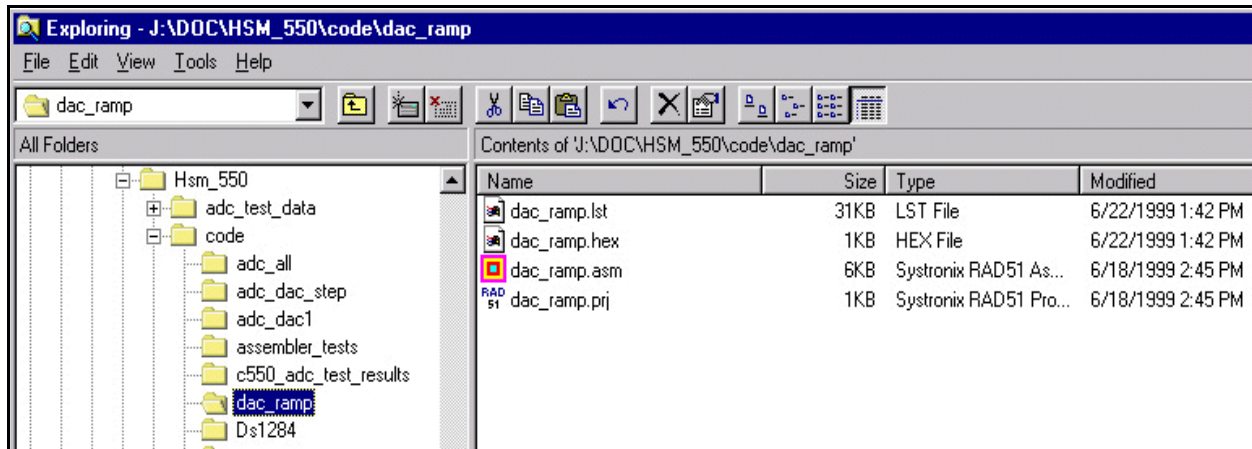


Figure 1. Typical Project Folder Hierarchy

You can see in the right pane that a project's folder contains the project definition file "dac_ramp.prj", the main assembly code file "dac_ramp.asm", and the output files with extensions ".lst" and ".hex". The project PRJ file and the main assembly code file MUST have the same name.

Figure 2 shows the contents of the parent folder, which we called "code". Our names are arbitrary, use whatever you prefer, but try to make the names descriptive. In the code folder you can see a number of files with extension ".inc": 550_io.inc, dac_550.inc, and so forth. These are assembly code "include" files and are "included" by many of the projects in the child folders such as dac_ramp. We'll show you how to

include this shared code in a moment.

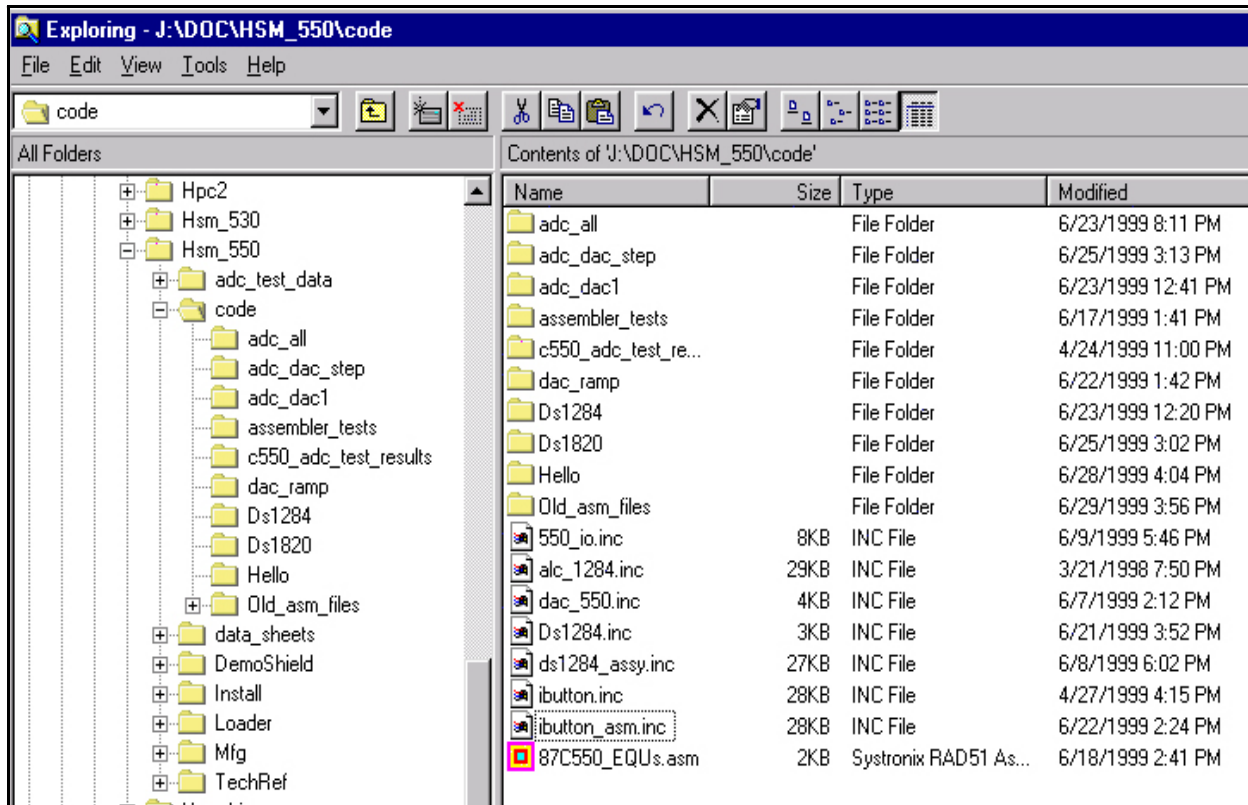


Figure 2. Shared code in the parent folder, in this example it's named "code".

■ RAD51 Projects

Creating a New Project from Scratch

Always start a new assembly code project by creating a PROJECT file. To do this, pull down the File->New->Project menu. This opens the new project dialog box.

In this example, we will create a new project which is in a child folder of J:\DOC\HSM_550\CODE.

Starting at the top of the New Project Configuration box, select Project Type. RAD51 currently supports only 8051 Assembly Projects.

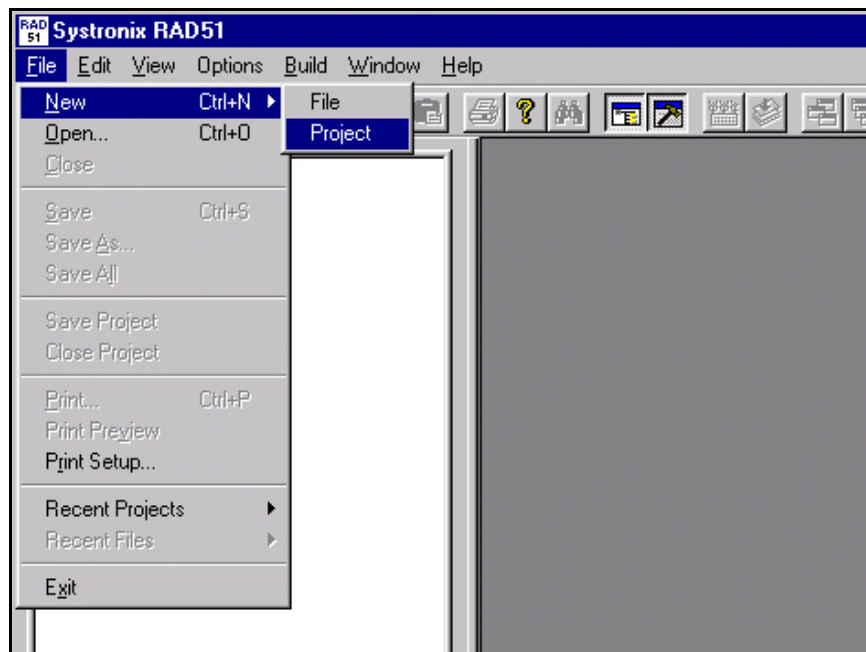


Figure 3. Creating a new project from the File menu.

Project File Options contains three choices and tell you where the main project file will be located. The first choice will create a new project folder in a child folder directly below the “Project Home Folder” (second dialog line from the bottom). You can click on the browse button to select the desired Project Home Folder.

Enter the New Project Name in the dialog area provided. The new name will be echoed along with the “.ASM” extension in the Main File (.asm) output dialog area. In this case the New Project Name is dac_ramp, a program which steps HSM550's DAC output in a ramp wave.

The other two project file options are worth mentioning.

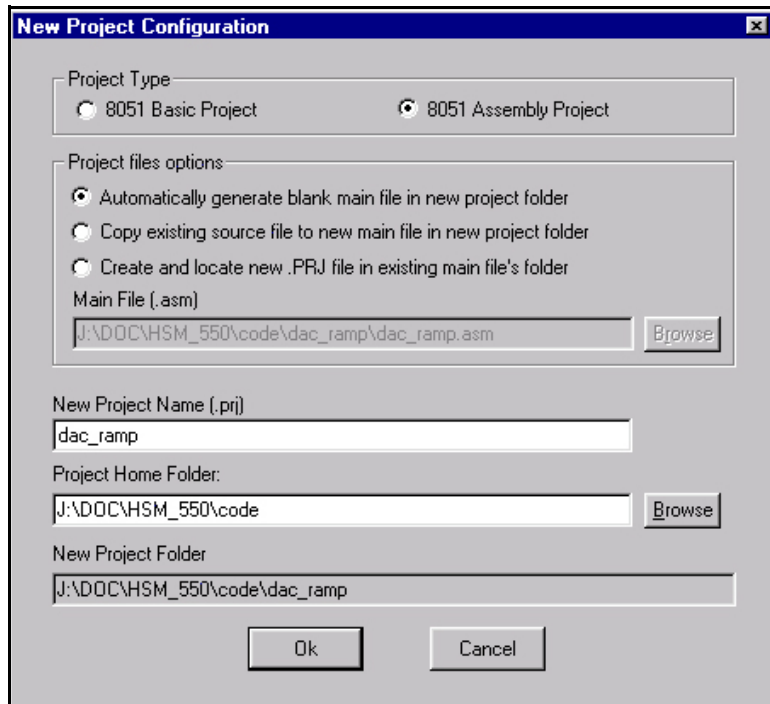


Figure 4. The new project dialog box.

Creating a new project by copying an existing .ASM file

If you wish to base your new project on an existing assembly code file (typically with the .ASM extension), use the second Project files option. This will copy your existing ASM file into a new folder, and a new .ASM file with the same name as the new project folder.

Creating a new project using an existing .ASM file

The third option lets you create a RAD51 project in whichever folder you already have an assembly code file which you want to be the new project. This is a quick and convenient way to create RAD51 projects from existing assembly code. In this case, the main file browse button will become active, and the new project will assume the name of your existing assembly code file.

Controller Selection

The RAD51 Options->Project Settings->Assembler dialog contains a pull down list of microcontrollers. These definitions come from the “controller.cfg” file located in the RAD51 installation directory. More about this file later, including how to add controllers to it. For now, select a controller from the list.

Relative Output File Paths

The RAD51 Options->Project Settings->Assembler dialog contains fields for the .HEX and .LST output file paths. These can be complete, absolute paths or relative paths. The figure shows how to specify output files located in the same folder as the PRJ file.

Relative paths are useful for exchanging project files with others, since they will probably not have the same absolute paths you do. This is how we distribute the examples shipped with RAD51.

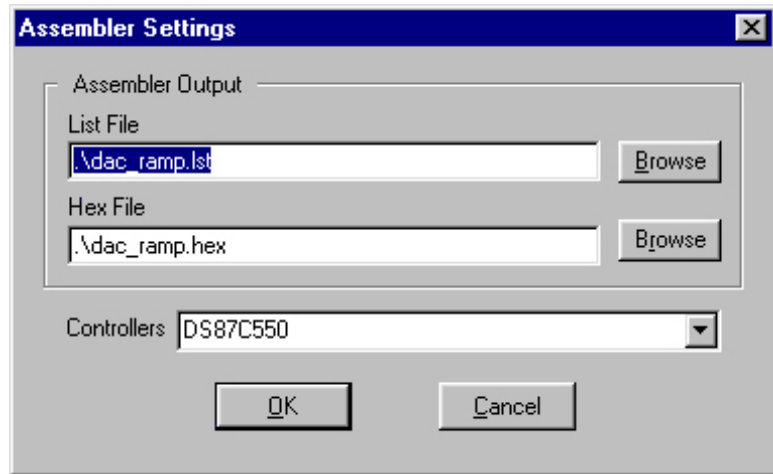


Figure 5. Use of relative paths in project output files.

PRJ and Main ASM Files in Same Folder

RAD51 assumes the PRJ and main ASM files will always be in the same folder. This enables the project paths to be relative to the location of the PRJ file. Relative paths let you copy a project with its common code folders (such as the examples we ship with RAD51) to a different computer. RAD51 will still be able to build the project in its new location since all the folders have the same relative locations.

Shared/Common Code in a Parent Folder

Keep Shared/Common Code in a folder which is the parent of your projects which use the shared code. Your project folders are then children of the code they share. For example:

- Common Code Folder (I/O drivers, math routines, etc)
- Project Folder
- Another Project Folder
- Yet Another Project folder

Project File Tree Pane

The figure shows the open project, with source code entered in the programmer's editor in the right pane. In the left pane we've selected the file view. You can see a file tree for the dac_ramp project. It consists of the main program file, dac_ramp.asm, and two include files: dac_550.inc and 550_io.inc. These files are included by using the "INCL" directive in the main assembly code file. Near the bottom of dac_ramp.asm, we entered these two include directives:

```
INCL "..\dac_550.inc"
INCL "..\550_io.inc"
```

The "..\" is the file path which means the folder one level above the location of the current file. This directs the assembler to look in the CODE folder which is one level above the DAC_RAMP folder.

Note that the file pane tree shows the *LOGICAL* relationship of your project files. It does not show their physical location - use Explorer for that as we did in Figure 1 and Figure 2.

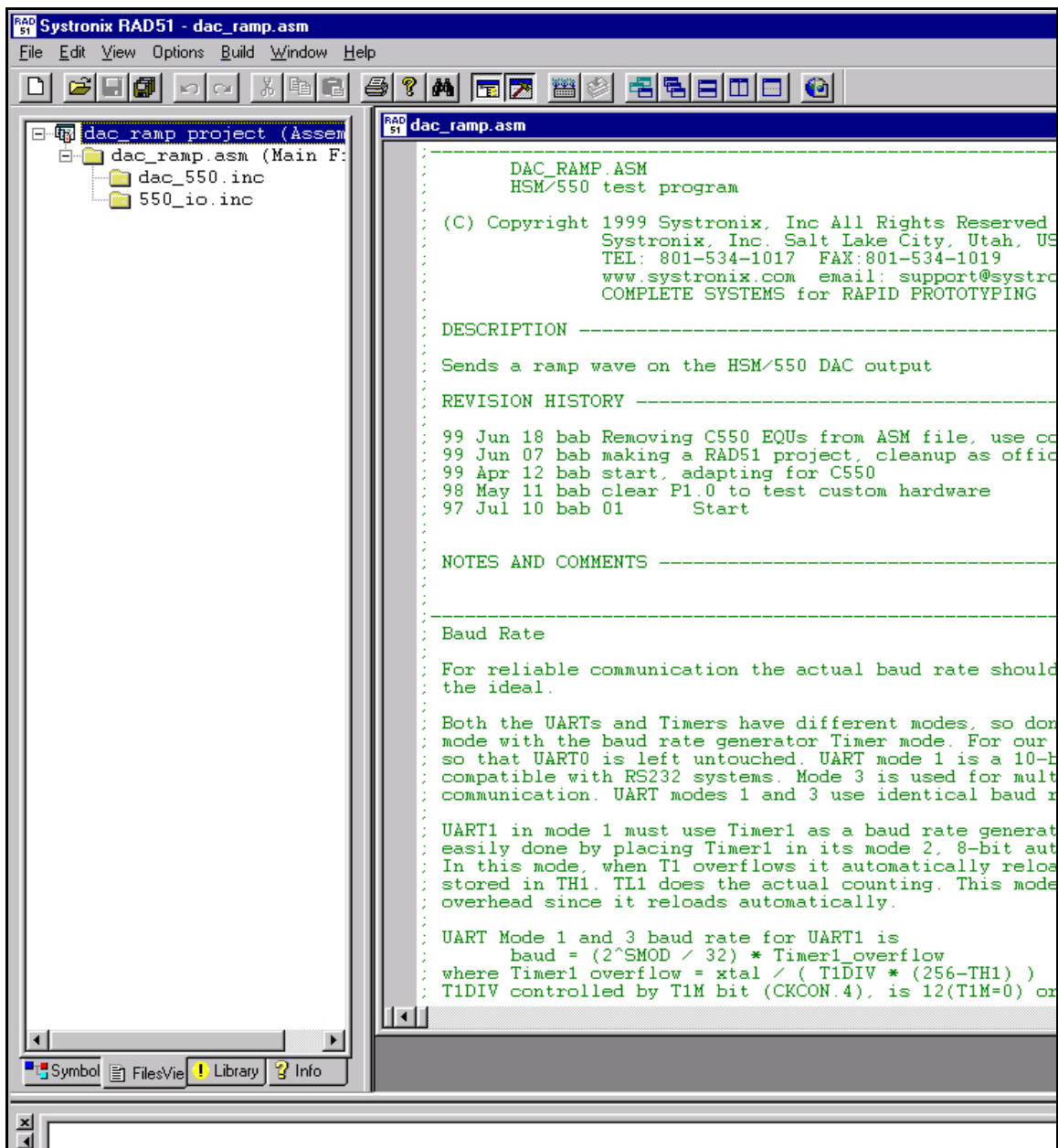


Figure 6. Project open, with project file tree in left pane.

Project Symbol Tree Pane

The left pane also can display a symbol tree, but you must first perform a BUILD in order for the assembler to collect all of your project's symbols. Double-click on the symbol, and the editor will move to the place in which the symbol is defined. You can search for symbols by pressing the first letter of the symbol's name on your keyboard.

Project Resource and Info Panes

These panes are currently unused by RAD51, but will contain output in the future.

File Menu: Files vs Projects

You can also open a file within the RAD51 File menu. Files do not have the PRJ association which Projects do. Therefore files do not have the File Tree and Symbol Tree generation capability.

■ *RAD51 Help, Configuration and Options*

Assembler and RAD51 Help & Help Files

The RAD51 Help menu has links to PDF help files for the assembler and the RAD51 environment. RAD51 also makes some use of tool tips. Holding the cursor stationary over a dialog window or icon will give you a floating "balloon" of help.

Controller Selection and Controller.cfg file

The RAD51 Options->Project Settings->Assembler dialog contains a pull down list of microcontrollers. These definitions come from the "controller.cfg" file located in the RAD51 installation directory. The controller.cfg file is simply ascii text with a section containing a controller name, followed by its SFR and BIT definitions. For example:

```
#DS87C550
;SFR names and locations
P0      SFR 080H
SP      SFR 081H
DPL     SFR 082H
DPH     SFR 083H
DPL1    SFR 084H
DPH1    SFR 085H
DPS     SFR 086H
; now some bits
IT0     BIT 088H
IE0     BIT 089H
IT1     BIT 08AH
IE1     BIT 08BH
```

The SFRs and BITs can occur in any order. You can include comments preceded by a semi-colon. White space is not significant.

The RAD51 assembler contains generic 8031 SFRs and BITs, these are not included in the controller.cfg file. The file is intended to provide definitions of only those SFRs and BITs which are in addition to, or different from, generic 8031 resources.

Feel free to add your own controllers and share them with others on our web site. Also check back from time to time, as we will be adding to the list of controllers and you can download new controller.cfg files from the web site www.systronix.com.

Editor Configuration and Options

The RAD51 Options->Editor dialog includes customization of colors, font, and tab settings. These settings become part of the PRJ file for that project. Some settings are stored in the Windows registry and will apply to all projects.

TROUBLESHOOTING & DEVELOPMENT TIPS

Internet FAQ

There is a FAQ (Frequently Asked Questions) on our web site. It is updated on a regular basis.

Start Simple

Start with a simple program, get it working, and then add complexity in modules. Try to add new functions as subroutines which you can call or not call to easily isolate suspected problems.

Learning Assembly Code and Embedded Programming

If you find a simple way to learn assembly code or embedded programming, let us know! We get asked “how do I learn ...” quite frequently. Maybe in a few years there will really be such a science as Computer Science, but at the moment, starting a discussion about “good design” is a good way to start an argument. This manual is not intended to teach you assembly code or good programming principles. Here are a few good books. Some of these may be hard to find - good luck - we don't stock them for that reason. If your local big chain bookstore won't order them, try a college bookstore or a smaller, independently owned bookstore, or search for the title on the Internet.

The Art of Embedded System Programming by Jack Ganssle, published by Academic Press. This one should be easy to order at your local bookstore. This book focuses on philosophy and general good programming hygiene. It's not specific to any processor. Lots of true and humorous anecdotes, very readable. He also publishes a delightful free e-mail newsletter. To subscribe, send a message to majordomo@ganssle.com, with the words "subscribe embedded your-email-address" in the body. Of course, substitute your actual email address for the text 'your-email-address'.

The Microcontroller Idea Book by Jan Axelson. YES we do stock this one. It's a good overview of microcontroller interfacing ideas, intended for hobbyists or those who are new to embedded systems. Lots of schematics and sample code (available on disk) for BASIC-52 controllers.

C and the 8051 by Thomas Schultz. As the title suggests, Professor Schultz assumes you will be using C, so most of the source code is in C. But there is a lot of good information applicable to any language or assembly code. Available at major bookstores and www.amazon.com.

Exception Handling

All embedded applications should have run-time *exception handling*. Good exception handling is one hallmark of a good programmer. Exception handling is a difficult topic, and highly application dependent, so it is hard to make specific recommendations. Therefore we will pass the buck here,

leaving the details to you, and yet pat ourselves on the back for at least mentioning it.

Quick Diagnosis Table

RAD51 QUICK DIAGNOSIS TABLE		
SYMPTOM	EXPLANATION	SOLUTION
Many undefined symbols when building one of the Systronix sample projects	No controller definition in use, therefore the assembler can't know about the controller-specific symbols	In the menu Options->Project Settings->Assembler select an appropriate controller from the pulldown list.
My controller is not in the pulldown list in the Options->Project Settings->Assembler menu	You need a new controller.cfg file	Add the controller SFR and BIT definitions yourself to the file "controller.cfg", or check our website (www.systronix.com) for a new version which may already have the controller you need. If you elect to add it, please send us the file as mime attachment to an email message (support@systronix.com) so that we can post it on our web site.
Multiply-defined symbol errors	Symbols defined both as EQUs in your source code and in the controller definition file	Multiple symbols can safely be EQUated to the same value - to do so is not an error. But the same symbol can never legitimately be assigned multiple values (an SFR or BIT can only exist in one location). This indicates either an error in the source code or the controller.cfg file.
Can't find files errors during a build of a project copied from the web or another PC	PRJ may contain absolute paths to output files, or source code may contain absolute paths in INCL include directives.	Specify relative output paths in Options->Project Settings->Assembler, this will make your projects portable. Also make INCL directives relative, as we have done in the examples.