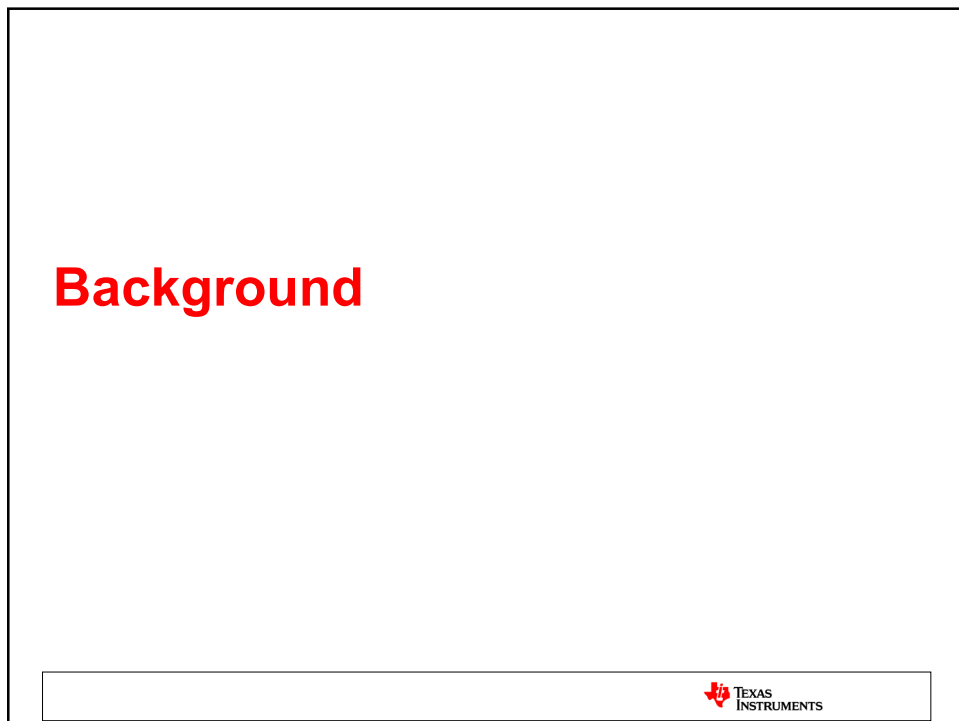




Code Composer Studio v4 Summary

- What is it?
 - Major upgrade to CCS
 - Major architectural changes
 - Based on Eclipse open source software framework
 - New registration/licensing/updating mechanism and model
- Why Eclipse?
 - Quickly becoming a standard for IDEs
 - Excellent software architecture
 - Ability to leverage the work of others
 - Cross-platform support (i.e. Windows & Linux)
 - Wide selection of 3rd party plug-ins available



Eclipse Background

- Eclipse
 - Originally IBM's Java Development Environment
 - Open platform for tool integration built by an open community of tool providers
 - Operates under a open source paradigm
 - Universal platform for tools integration
 - Multi-language, multi-platform and multi-vendor environment
 - Excellent plug-in framework
- Governed by Eclipse Public License (EPL)
 - Royalty free source code
 - World wide redistribution rights
 - Enables you to sell components & products
 - If you modify any source governed by EPL and you need to make the source available
- Eclipse Foundation
 - The Eclipse Foundation is a non-profit corporation formed to advance the creation, evolution, promotion, and support of the Eclipse Platform and to cultivate both an open source community and an ecosystem of complementary products, capabilities, and services.
- Key members
 - IBM/Rational, QNX, WindRiver, Borland, RedHat, MontaVista, [Texas Instruments](#)



CCS Vs. CCE

- Code Composer *Studio*
 - Our main product brand.
- Code Composer *Essentials*
 - Created when we developed a new product for MSP430
 - Has a subset of the features of CCS
 - This brand goes away with CCSv4 when all ISAs are supported by the same product.



Maintenance of Older Products

- Code Composer Studio v3.3
 - Once CCSv4 is released we will stop doing proactive Service Releases. However we will maintain the ability to do Service Releases and will respond to critical customer issues as they arise.
 - Based on past history we expect to do 2-3 Service Releases on 3.3 after 4.0 has shipped
 - Will still be available to purchase into 2010
 - Microcontroller version will be available indefinitely to support F24x
- Code Composer Essentials
 - The move from 3.x to 4.x is much smaller for MSP users than other users as they are already Eclipse based.
 - CCEv3.x will not be maintained after 4.0 is released



Exciting new features

These are some of the reasons you should upgrade!



Licensing Summary

- CCSv4 will incorporate FlexNET licensing
 - Licensing and registration system
- Product branding
 - Platinum: All ISAs
 - Microcontroller: MSP430/C2800 (Stellaris M3 to be added)
- Types of licenses
 - Free time-limited evaluation tools (Platinum)
 - 30 days built-in, can extend for 90 more by registering
 - Free object code size limited tools (Microcontroller)
 - Only restriction on the tool is the size of the object code that can be produced
 - Free bundled tools (Platinum)
 - Free for use with EVM/DSK bundles, simulators and XDS100
 - Node locked (Platinum & Microcontroller)
 - Tied to a computer
 - **Floating** (Platinum & Microcontroller)
 - Share licenses amongst a pool of users
- Enforcement strategy
 - In most cases if you have an issue checking out a license CCS will just tell you that there was a problem and not prevent you from using the tool.

http://tiexpressdsp.com/index.php/CCSv4_Licensing



CCSv4 Pricing Summary

Item	Description	Price
Platinum Eval Tools	Full tools with 120 day limit	FREE
Platinum Bundle	EVM/DSK, sim, XDS100 use	FREE
Platinum Node Locked	Full tools tied to a machine	\$1995
Platinum Floating*	Full tools shared across machines	\$2995
Microcontroller Core	MSP/C2000 code size limited	FREE
Microcontroller Node Locked	MSP/C2000	\$495
Microcontroller Floating*	MSP/C2000	\$795

*3, 5, 10, 25 user bundles available as well

Part #s: <http://tiexpressdsp.com/index.php/Licensing> - CCSv4#Part .23s

**Stellaris support will be added to Microcontroller licenses



Eclipse Plugins! (1/2)

- Problem:
 - More than just an embedded debugger is required during product development
- Solution:
 - CCSv4 is based on Eclipse which has a huge selection of 3rd party plug-ins available (code analysis, source code control, modelling, Perl development...)
 - <http://www.eclipseplugincentral.com>
 - The Eclipse plug-in development environment allows for the creation of your own custom tooling
 - Wizards for creating plug-ins quickly
 - Look for plugins compatible with Eclipse 3.2 (the version CCS 4 is built on)

More info here: http://tiexpressdsp.com/index.php/Extending_CCS



Eclipse Plugins! (2/2)

Validated plugins

- http://tiexpressdsp.com/index.php/Perl_Development
- http://tiexpressdsp.com/index.php/ClearCase_plugin_installation
- http://tiexpressdsp.com/index.php/ClearQuest_plugin_installation
- http://tiexpressdsp.com/index.php/SourceSafe_plugin_installation
- http://tiexpressdsp.com/index.php/SubVersion_plugin_installation

Please contribute more wiki articles if you use/validate other Eclipse 3.2 plugins.



Easier to work with

- Emulation drivers packaged with CCS
- Don't need separate "DSK versions" and "full versions". Anyone can always download the latest free version to work with simulators, DSKs, and XDS100 emulators.
- CCS will automatically check for updates once a week when you start it. Faster file server for delivering updates.



XDS100: Product Brief

- Robust and efficient JTAG emulation controller
 - Plug in and play operation.
- Compatible with Code Composer Studio
- Support for
 - XDS100 v1: c28x, 54x, 55x, 674x processors
 - XDS100 v2: ARM7/9/M3/R4/A8 and 64x+ (4Q09)
- Available from TI E-store: \$79
 - Also available from: Spectrum Digital, Blackhawk, Olimex, Realtime, SEED, ZLG, Bilkon, Pantech, etc.
 - Build your own!
- Additional information:
 - <http://tiexpressdsp.com/wiki/index.php?title=XDS100>



Advanced Editor Features

- Tabs!!!
- Code Completion (Ctrl – Space)
 - Complete word
 - Auto-member information
 - Auto-parameter information
 - ...
- Navigation
 - Back/Forward buttons
 - Back to last edit button
 - Go to definition
 - Go to declaration
- Show line numbers
- Code Folding
 - Collapse functions

 A screenshot of a code editor window. The window has two tabs: 'sinetab.c' and '*modemtx.c'. The active tab is '*modemtx.c', which displays C code. Line numbers are shown in the left margin. The code includes comments and function definitions. A code completion popup is visible, showing a list of variables and functions that match the current cursor position. The editor also shows a 'Go to definition' button (a magnifying glass icon) next to a function call.

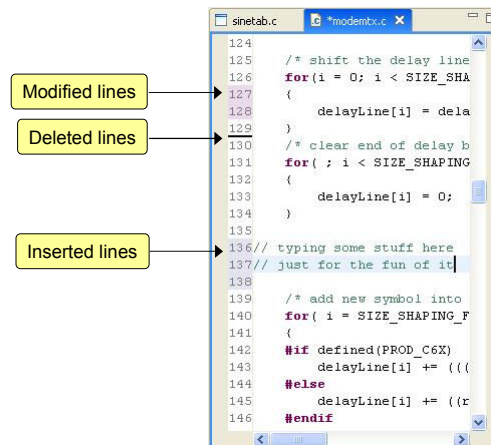
```

53 *****
54 int SineLookup(int sample) {
72
73 *****
  
```



Edit Markers

- If you have the line number column on it also indicates changes in your source file since your last save



Key differences

Be aware of these items to make your migration smoother.



Migration FAQ

- What happens to my CCSv3.3 projects?
 - Import wizard for conversion
 - http://tiexpressdsp.com/index.php?title=Migrating_to_CCSv4#Importing_Projects
- What happens to workspaces?
 - Need to recreate
- What happens to GEL?
 - GEL is present in CCSv4
 - <http://tiexpressdsp.com/index.php?title=GEL>
- How do I automate CCS?
 - Debug Server Scripting
 - <http://tiexpressdsp.com/index.php?title=DSS>



Target Setup

- Change to xml based target configuration
 - Existing target configurations are not compatible
 - Device and connection are separated
 - You can change the emulator type that a configuration uses
 - 3P emulator vendors do not need to create import configurations for every device as you can just choose the emulator and the device
- Setup is integrated into the IDE
- Target configurations can be included in a project so that a project automatically uses that configuration



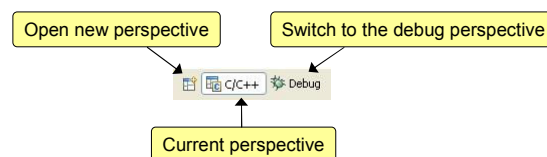
Projects

- CCS .pj1 file are replaced by Eclipse projects
 - An import wizard is provided that does the conversion
 - If someone does not wish to convert due to some users still being on CCSv3.x you can use a standard make project and tell it to build a .pj1 file using timake.exe
 - Create new projects by going to **File** -> **New** -> **CCS Project**
- File system based model
 - Files that are in the project folder and in the project and will be built unless excluded
 - Create subfolders in a project by creating subfolders in the project folder
 - If you don't want a source file to be physically in the project folder you add it to your project using the "link files to project" feature



Perspectives

- Background:
 - Each Workbench window contains one or more perspectives.
 - A perspective defines the initial set and layout of views in the Workbench window.
 - Each perspective provides a set of functionality aimed at accomplishing a specific type of task or works with specific types of resources.
- As you work in the Workbench, you will probably switch perspectives frequently
- Customization
 - Perspectives control what appears in certain menus and toolbars
 - They define visible action sets, which you can change to customize a perspective
 - You can save a perspective that you build in this manner, making your own custom perspective that you can open again later.
- Default perspectives in CCS-Eclipse
 - C/C++: Editing and building
 - Debug: Debugging your programs
- You can use the **General > Perspectives** preference page to open perspectives in the same window or in a new window
 - By default opening a new perspective will switch the perspective but you can change this so that it opens a new workbench window for the perspective



Multi-processor Environment

- Problem:
 - Many devices today include more than one processing core and often reside in a system with many other devices. Displaying debug information from many different cores typically requires many IDE windows.
- Solution:
 - CCSv4 allows you to have a single IDE window and to change the debug context of the IDE to any of the cores in the system.
 - You can also “pin” the context of a debug display to a specific core.
- If desired you can open a top level IDE for any core

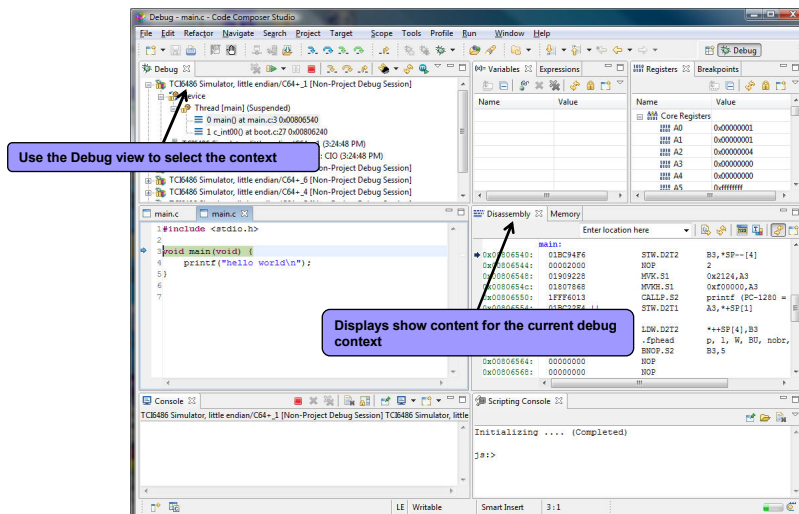
Lots more info at:

http://tiexpressdsp.com/index.php/Multi-Core_Debug_with_CCS

21



CCSv4 Multi-core Environment



BIOS

- CCSv4 includes two versions of DSP/BIOS
 - 5.40
 - Binary compatible with BIOS5.3
 - Updated to support CCSv4 integration
 - Recommended if migrating existing projects to CCSv4
 - Does not support execution graph
 - 6.x
 - New API
 - Supports complete real-time analysis tools including execution graph

23



New Icons (1/2)

“Debug” = Launch debugger and switch to debug perspective

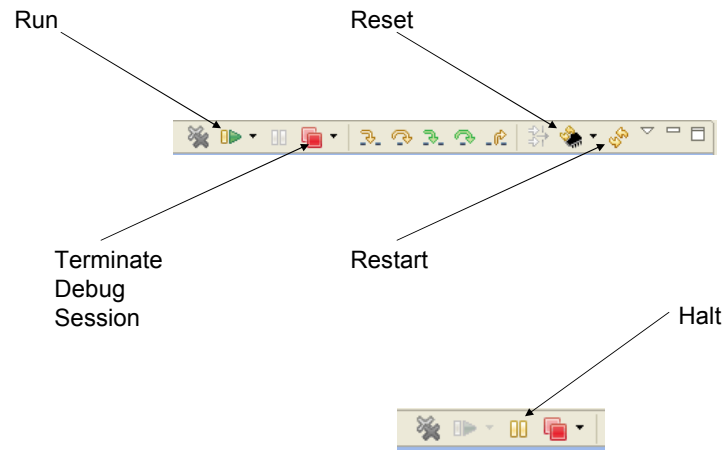


“Rebuild Active Project” = Clean + Build

“Build Active Project” = Incremental build



New Icons (2/2)



Thank you!