

Embedded Target for TI TMS320C6000™ DSP Platform Release Notes

Contents

Summary by Version	4
Version 3.1 (R2006b) Embedded Target for TI TMS320C6000 DSP	7
Version 3.0 (R2006a) Embedded Target for TI TMS320C6000 DSP	12
Version 2.4 (R14SP3) Embedded Target for TI TMS320C6000 DSP	20
Version 2.3 (R14SP2) Embedded Target for TI TMS320C6000 DSP	23
Compatibility Summary for Embedded Target for TI TMS320C6000 DSP	26

Summary by Version

This table provides quick access to what's new in each version. For clarification, see About Release Notes.

Version (Release)	New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Latest Version V3.1 (R2006b)	Yes Details	Yes Summary	Bug Reports	Printable Release Notes: PDF Current product documentation
V3.0 (R2006a)	Yes Details	Yes Summary	Bug Reports	No
V2.4 (R14SP3)	Yes Details	No	Bug Reports	No
V2.3 (R14SP2)	Yes Details	No	Bug Reports	No

About Release Notes

Use release notes when upgrading to a newer version to learn about new features and changes, and the potential impact on your existing files and practices. Release notes are also beneficial if you use or support multiple versions.

If you are not upgrading from the most recent previous version, review release notes for all interim versions, not just for the version you are installing. For example, when upgrading from V1.0 to V1.2, review the New Features and Changes, Version Compatibility Considerations, and Bug Reports for V1.1 and V1.2.

New Features and Changes

These include

- New functionality

- Changes to existing functionality
- Changes to system requirements (complete system requirements for the current version are at the MathWorks Web site)
- Any version compatibility considerations associated with each new feature or change

Version Compatibility Considerations

When a new feature or change introduces a known incompatibility with the previous version, its description includes a **Compatibility Considerations** subsection that details the impact. For a list of all new features and changes that have compatibility impact, see the Compatibility Summary for Embedded Target for TI TMS320C6000 DSP.

Compatibility issues that become known after the product has been released are added to Bug Reports at the MathWorks Web site. Because bug fixes can sometimes result in incompatibilities, also review fixed bugs in Bug Reports for any compatibility impact.

Fixed Bugs and Known Problems

MathWorks Bug Reports is a user-searchable database of known problems, workarounds, and fixes. The MathWorks updates the Bug Reports database as new problems and resolutions become known, so check it as needed for the latest information.

Access Bug Reports at the MathWorks Web site using your MathWorks Account. If you are not logged in to your MathWorks Account when you link to Bug Reports, you are prompted to log in or create an account. You then can view bug fixes and known problems for R14SP2 and more recent releases.

The Bug Reports database was introduced for R14SP2 and does not include information for prior releases. You can access a list of bug fixes made in prior versions via the links in the summary table.

Related Documentation at Web Site

Printable Release Notes (PDF). You can print release notes from the PDF version, located at the MathWorks Web site. The PDF version does not support links to other documents or to the Web site, such as to Bug Reports. Use the browser-based version of release notes for access to all information.

Product Documentation. At the MathWorks Web site, you can access complete product documentation for the current version and some previous versions, as noted in the summary table.

Version 3.1 (R2006b) Embedded Target for TI TMS320C6000 DSP

This table summarizes what's new in V3.1 (R2006b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as Compatibility Considerations , below. See also Summary.	Bug Reports	Printable Release Notes: PDF Current product documentation

New features and changes introduced in this version are

- “Code Composer Studio 3.2 Compatibility”
- “Blocks for UDP Made General Use”
- “TCP/IP Blocks for Targets Added”
- “Host Communication Library Added to C6000lib”
- “New C6000 Memory Blocks”
- “Blocks Relocated from DM642 Library”
- “New Memory Mapping Capability from Real-Time Workshop Embedded Coder”
- “Demo Added”
- “Additional C6000 Processors in Custom C6000 Target Preferences Block”
- “Added Target Preferences blocks for C6455 DSK and PADK6727 Targets”

Code Composer Studio 3.2 Compatibility

Embedded Target for TI TMS320C6000 DSP now works with Code Composer Studio 3.2 for the C6455 DSK and C6727 PADK targets which require CCS 3.2. Note that you must install the following software:

- Code Composer Studio V3.2

- Spectrum Digital Drivers package `setupCCSPlatinum_v30104C.exe` (includes both C2000 and C6000 drivers). This file is available from Spectrum Digital.
- (For DM642 users only) Driver development kit (DDK) that comes on the Texas Instruments software CD included with the DM642 EVM

Compatibility Consideration. Embedded Target for TI TMS320C6000 DSP only works with CCS versions 3.1 and 3.2. The product does not support any earlier CCS versions.

Blocks for UDP Made General Use

We reconfigured the UDP Send and Receive blocks to apply to all C6000 targets. As noted below (here)

TCP/IP Blocks for Targets Added

The C6000 DSP Communication Library includes two new TCP/IP blocks for configuring TCP/IP communications to and from your C6000 target. Note that these blocks only apply on the target.

Host Communication Library Added to C6000lib

The Host Communication library contains five new UDP message-related blocks.

- Byte Pack
- Byte Reversal
- Byte Unpack
- UDP Receive
- UDP Send

These blocks enable you to send and receive UDP messages to and from the target. The byte manipulating blocks configure messages to be used by UDP, and the UDP blocks provide the connection between the transport and communications layers on the target.

New C6000 Memory Blocks

Two new blocks in C6000 DSP Core Support library provide control for memory allocation and copying on the target.

- Memory allocate—Allocate memory section on C6000 target
- Memory copy—Copy to and from memory section on C6000 target

These new blocks replace the functions provided by the From Memory and To Memory blocks. As a result, the older blocks are obsolete and may be removed in the future.

In new models, use the new Memory Allocate and Memory Copy. In older models, we recommend that you replace the To Memory and From Memory blocks with the new blocks.

Compatibility Consideration. The To Memory and From Memory blocks are being deprecated and were removed from the C6000 Core Support Library (c6000dspcorelib) library. They remain in the block reference library memblks. Existing models that use these blocks continue to work until the blocks are removed from the memblks library in a future release.

R13SP2 models that use To Memory and From Memory blocks work because of the Simulink block-forwarding mechanism between the C6000 Core Support (c6000dspcorelib) and memblks libraries. You can avoid invoking the block-forwarding mechanism each time you use the models by resaving your R13SP2 models after you open them with this release of the product .

Blocks Relocated from DM642 Library

We moved the following blocks out of the DM642 Library and into the C6000 DSP Communication Library to indicate they are no longer specific to the DM642 EVM. The blocks still work the same way.

- DM642 EVM IP Config—Renamed C6000 IP Config
- DM642 EVM UDP Receive—Renamed C6000 UDP Receive
- DM642 EVM UDP Send—Renamed C6000 UDP Send

New Memory Mapping Capability from Real-Time Workshop Embedded Coder

When you select a target based on the embedded real-time target (such as the `ti_c6000_ert.tlc`), there is a new configuration option in the **Select** tree—**Memory Sections**.

The **Memory Sections** pane provides an interface for inserting comments and pragmas into the generated code for

- Data defined in custom storage classes
- Internal data not defined in custom storage classes
- Model-level functions
- Atomic subsystem functions with or without separate data

In general, inserting pragmas in your generated code is difficult. This new interface option provides a direct technique that lets you add the pragmas when you generate code from your model.

Pragmas inserted into generated code can surround

- A contiguous block of data or function definitions
- Each data or function definition separately

When pragmas surround each definition separately, the text of each pragma can name the definition to which it applies.

Using the new memory section capability can help you optimize how your code runs on your target, improving performance. These new memory section features work in combination with the memory mapping features in the target preferences blocks for each processor.

For more information about the new **Memory Sections** pane in **Configuration Parameters**, refer to “Introduction to Memory Sections” in your **Real-Time Workshop Embedded Coder** documentation.

Demo Added

New demo “Using the PCI Bus to Exchange Data with xPC.”

Note that “Using the PCI Bus to Exchange Data with xPC” requires xPC to run.

This demo is particularly valuable because it demonstrates how to use some of the new blocks.

Two more demos

- Custom Device Driver via Legacy Code Integration
- User-Controlled Memory Mapping

show you how to use memory allocation and copy blocks to control memory use on the target and to develop your own device drivers.

Additional C6000 Processors in Custom C6000 Target Preferences Block

The Custom C6000 Target Preferences block now supports additional processors in the C6000 DSP family. Adding the block to your model and selecting the appropriate processor from the list lets you target your model to more DSPs. To use the new processors, install CCS 3.2.

Added Target Preferences blocks for C6455 DSK and PADK6727 Targets

New target preferences blocks enable you to configure models to generate code for two new targets:

- C6455 DSP Starter Kit
- Professional Audio Development Kit for the 6727 processor

Compatibility Consideration. To support the new PADK and C6455 processors, all of the target preferences blocks gained a new set of options for board custom code. As a result of the change, if you used earlier versions of the target preferences blocks in models, you should open the block in the model and resave it. This process will update the block and prevent compatibility problems. Users of the DM642 are likely to be most affected by this change to all of the preferences blocks.

Version 3.0 (R2006a) Embedded Target for TI TMS320C6000 DSP

This table summarizes what's new in V3.0 (R2006a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	Yes—Details labeled as Compatibility Considerations , below. See also Summary.	Bug Reports	No

New features and changes introduced in this version are

- “Code Composer Studio 3.1 Compatibility”
- “Asynchronous Scheduler”
- “Most Demos Updated to Use Asynchronous Scheduler”
- “DSP/BIOS Library Added to C6000lib”
- “Communications Blocks in New Host Communication Library added to C6000lib”
- “New C6000 DSP Core Support Blocks Added”
- “New Blocks in DM642 Library”
- “New Memory Mapping Capability from Real-Time Workshop Embedded Coder”
- “‘Host to Target’ and ‘Audio and Video Loopback’ Demos Added”

Code Composer Studio 3.1 Compatibility

Embedded Target for TI TMS320C6000 DSP now works with Code Composer Studio 3.1. Note that you must install the following software:

- Code Composer Studio V3.1

- Spectrum Digital Drivers package setupCCSPlatinum_v30104C.exe (includes both C2000 and C6000 drivers). This file is available from Spectrum Digital.
- (For DM642 users only) Driver development kit (DDK) that comes on the Texas Instruments software CD included with the DM642 EVM
- (For DM642 EVM users only) Network development kit (NDK) that comes on the Texas Instruments software CD included with the DM642 EVM

Compatibility Consideration. Embedded Target for TI TMS320C6000 DSP only works with CCS version 3.1. The product does not support earlier CCS versions.

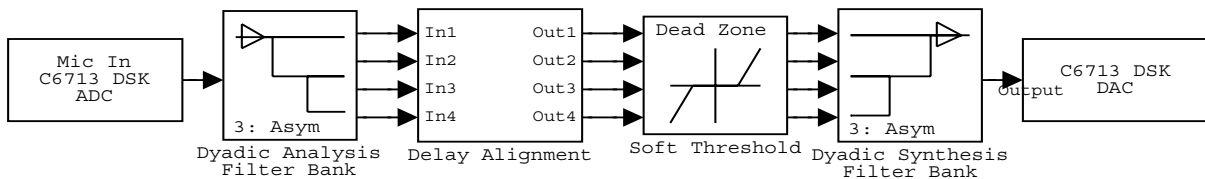
Asynchronous Scheduler

Now you can use an asynchronous (real-time) scheduler for your target application. Earlier versions of the Embedded Target for TI TMS320C6000 DSP used a synchronous CPU timer interrupt-driven scheduler. With the new scheduler you can define interrupts and tasks to occur when you want them to by using new blocks in the following libraries:

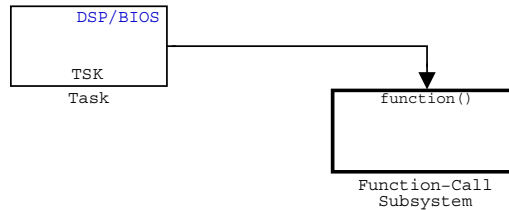
- C6000 DSP Core Support
- DSP/BIOS Library

With the new scheduler, you can schedule multiple tasks asynchronous execution using blocks in the C6000 DSP Core Support and DSP/BIOS Library block libraries. The following figures show a model updated to use the asynchronous scheduler.

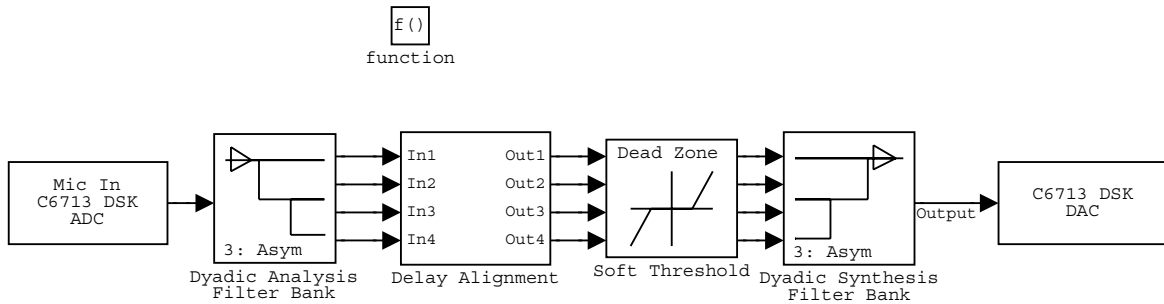
Before



After



Model Inside the Function Call Subsystem Block



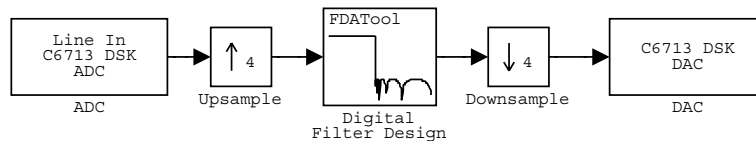
Compatibility Consideration. The V3.0 changes in the real-time scheduler can break some existing multirate models that contain codec blocks such as the ADC and DAC. The models affected contain at least one sample rate that is faster than the codec block rate. You do not run into this problem if all rates in the model are lower than the codec rate.

The new scheduler provides improved control for your processing and improved performance. You should recast all of your models to use the new asynchronous scheduler. To update your models, embed the entire processing algorithm or system in a function-call subsystem driven by a DSP/BIOS Task or Idle Task block from the DSP/BIOS Library library.

An example of such a model contains a combination of an ADC block and a DAC block, with a processing algorithm between them that executes at the higher rate. If you run code generated for such a model in `multitasking` or `auto solver`

mode, you might hear occasional audio glitches or your program may overrun. The exact symptom of the problem depends on the run-time overrun action setting in the TIC6000 Code Generation options.

The following model demonstrates one possible model configuration that can demonstrate the audio problems.



This multirate model uses two interrupts to control real-time execution of the generated code:

- A DMA interrupt to drive the execution of the code for ADC and DAC blocks
- A timer interrupt to drive the execution of the code for the FIR filter at an increased sample rate

In earlier product versions, the generated scheduler constantly synchronized the DMA and timer interrupts to ensure they remained in sync with one another, despite the possible clock drift with interrupts that are clocked by independent clock sources.

With the new real-time scheduler, the product does not synchronize the ADC and timer interrupts.

One interrupt may get out of sync with the other, with the time difference between them (drift) fluctuating with changes in the independent interrupt clocks. When the drift reaches a critical threshold, processing may skip an instance of a lower-priority task.

At that point, the interrupts are back in sync and the process continues. Losing synchronization between the interrupts can corrupt the audio signal or lead to an interrupt overrun.

To avoid the audio problems in an existing model that you cannot update to the new scheduler, set the run-time overrun action for the model to either None or Notify_and_continue to prevent the program from overrunning.

Most Demos Updated to Use Asynchronous Scheduler

To demonstrate using the asynchronous schedule capability, most of the existing demos now use the tasking blocks and interrupts to drive the demos. The processing parts of the demos now reside in function-call subsystem blocks that respond to task blocks of one form or another. Reviewing these demos can help you become familiar with the new blocks and scheduler.

DSP/BIOS Library Added to C6000lib

To enable you to use the new asynchronous scheduler, you must be able to create tasks that respond to interrupts. When you add DSP/BIOS to your configuration for your model, the following new blocks provide the means to add the tasks to your model.

- HWI—Create interrupt service routine on C6000 hardware target
- Task—Create task that runs as separate DSP/BIOS thread
- Triggered Task—Create asynchronously triggered task

Communications Blocks in New Host Communication Library added to C6000lib

The Host Communication library contains five new UDP message-related blocks:

- Byte Pack—Convert input signals into uint8 vector
- Byte Reversal—Reverse order of bytes in input word
- Byte Unpack—Unpack UDP uint8 input vector into Simulink data type values
- UDP Receive—Receive uint8 vector as UDP message
- UDP Send—Send UDP message to host

These blocks enable you to send and receive UDP messages on the target. The Byte related blocks configure messages to be used by UDP, and the UDP blocks provide the connection between the transport and communications layers on the target.

New C6000 DSP Core Support Blocks Added

New blocks in this library enable you to use the new asynchronous scheduler in bare-board code generation mode that does not incorporate the DSP/BIOS features.

- Hardware Interrupt—Generate interrupt service routine. Same as the DSP/BIOS interrupt block
- CPU timer—Generate interrupt service routine
- Idle Task—Create free-running background task

New Blocks in DM642 Library

The new blocks let you configure the audio codec on the DM642 EVM, use the general-purpose I/O pins on the EVM, set up the IP protocol parameters, and use UDP messaging on the target. One block lets you configure the EVM to collect custom video by defining the raw video format.

- DM642 EVM Audio ADC—Configure audio codec and peripherals on the DM642 Evaluation Module
- DM642 EVM Audio DAC—Configure the audio codec to convert digital audio input to analog audio output
- DM642 EVM FPGA GPIO Read—Configure DM642 EVM User GPIO registers to read from selected pins
- DM642 EVM FPGA GPIO Write—Configure DM642 EVM User GPIO registers
- DM642 EVM IP Config—Configure Internet Protocol (IP) parameters for DM642 EVM
- DM642 EVM UDP Receive—Configure Ethernet driver to receive UDP message as uint8 vector
- DM642 EVM UDP Send—Configure Ethernet driver to send UDP message
- DM642 EVM Video Port—Configure video port to receive video data stream from video input port

New Memory Mapping Capability from Real-Time Workshop Embedded Coder

When you select a target based on the embedded real-time target (such as the `ti_c6000_ert.tlc`), there is a new configuration option in the **Select** tree—**Memory Sections**.

The **Memory Sections** pane provides an interface for inserting comments and pragmas into the generated code for

- Data defined in custom storage classes
- Internal data not defined in custom storage classes
- Model-level functions
- Atomic subsystem functions with or without separate data

In general, inserting pragmas in your generated code is difficult. This new interface option provides a direct technique that lets you add the pragmas when you generate code from your model.

Pragmas inserted into generated code can surround

- A contiguous block of data or function definitions
- Each data or function definition separately

When pragmas surround each definition separately, the text of each pragma can name the definition to which it applies.

Using the new memory section capability can help you optimize how your code runs on your target, improving performance. These new memory section features work in combination with the memory mapping features in the target preferences blocks for each processor.

For more information about the new **Memory Sections** pane in **Configuration Parameters**, refer to “Introduction to Memory Sections” in your Real-Time Workshop Embedded Coder documentation.

‘Host to Target’ and ‘Audio and Video Loopback’ Demos Added

Two demos, ‘Host to Target’ and ‘Audio and Video Loopback,’ show you how to use the new asynchronous scheduler with new blocks for defining tasks and

interrupts. You find these new demos in the online Help system, under Simulink->Embedded Target for TIC6000 DSP.

These demos are particularly valuable because they demonstrate how to use some of the new blocks.

Additional C6000 Processors in Custom C6000 Target Preferences Block

The Custom C6000 Target Preferences block now supports additional processors in the C6000 DSP family. Adding the block to your model and selecting the appropriate processor from the list lets you target your model to more DSPs.

Version 2.4 (R14SP3) Embedded Target for TI TMS320C6000 DSP

This table summarizes what's new in V2.4 (R14SP3):

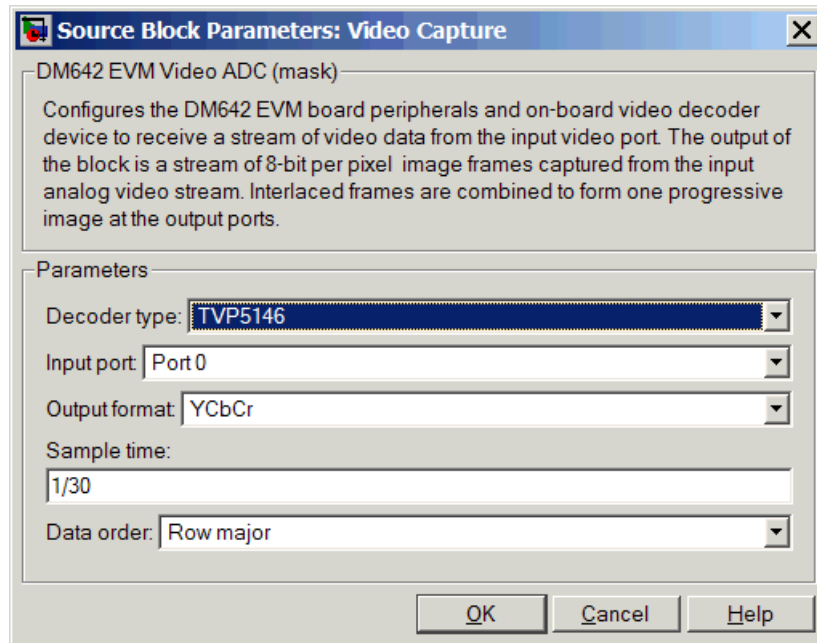
New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	No	Bug Reports	No

New features and changes introduced in this version are described here.

New Video ADC and DAC Block Options for DM642 EVM

Recently, Spectrum Digital released a new version of the DM642 EVM that uses different decoders. To support the new revision, we changed the existing Video ADC and DAC blocks. We added new options for configuring the blocks.

- 1 To the Video ADC block, we added the **Decoder type** option to let you specify whether your EVM uses the Phillips SAA7115 decoder or the newer TI TVP5146/5150 decoder. The new DM642 EVM uses the TVP5146 decoder setting, shown here. This is also the default setting for the block configuration.



- To both the Video ADC and Video DAC block options, we added the **Data order** option. With this, you tell the decoder or encoder whether to output video in row major or column major order.

Most video capture and display systems use row major ordering. MATLAB and Simulink use column major ordering. As a result, some Simulink blocks and MATLAB operations may not produce the output you expect unless you change the ordering for video from the default row major setting to column major.

You can find more information about these new options on the reference pages for the DM642 EVM Video ADC and DM642 Video DAC blocks in the *Embedded Target for TI C6000 DSP User's Guide* documentation in the online Help system.

Model Reference Support

Embedded Target for TI C6000 DSP now supports the model referencing capability provided by Simulink. Model Reference makes working with very

large models that have many blocks more convenient for code generation, maintenance, and operation. For more information about model reference, refer to the Real-Time Workshop documentation in the online Help system.

For information about using model reference with Embedded Target for TI C6000 DSP, refer to the Embedded Target for TMS320C6000 DSP Platform documentation in the online Help system.

Allows Multiple DM642 EVM Video ADC Blocks in a Model

Now you can add up to two Video ADC blocks to Simulink models. Adding two blocks lets you use two inputs to your DM642 EVM and use both video streams. Having two video inputs available lets you do such things as use Picture-in-Picture on your display.

Generated Code Does Not Honor Simulation Stop Time on GRT-based Targets

You can set a simulation stop time in the Configuration Parameters for any model. Models that you target to TI hardware using the GRT-based target do not respond to the simulation stop time in the generated code on the target.

ERT-based targets never respected the stop time setting.

We still support stop requests from the Stop block in code. This is true for both ERT- and GRT-based targets.

Version 2.3 (R14SP2) Embedded Target for TI TMS320C6000 DSP

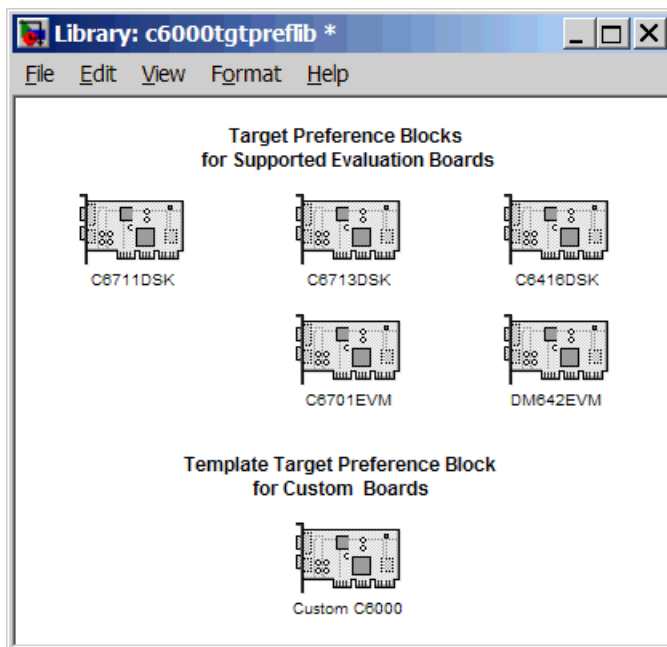
This table summarizes what's new in V2.3 (R14SP2):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems	Related Documentation at Web Site
Yes Details below	No	Bug Reports	No

New features and changes introduced in this version are described here.

New Target Preferences Blocks for Targeting Custom Hardware

Version 2.3 introduces new blocks for targeting both existing supported boards and custom hardware. Located in the C6000 Target Preferences library (enter `c6000tgtpreflib` at the MATLAB prompt) and shown in this figure, the blocks provide configuration details for both supported boards like the C6711 DSK, and your custom hardware based on processors from the C6000 family.



Note the two block categories:

- Blocks for supported boards:
 - C6416DSK—Set the target preferences and memory map for the C6416 DSP Starter Kit
 - C6701EVM—Set the target preferences and memory map for the C6701 Evaluation Module
 - C6711DSK—Set the target preferences and memory map for the C6711 DSP Starter Kit
 - C6713DSK—Set the target preferences and memory map for the C6713 DSP Starter Kit
 - DM642EVM—Set the target preferences and memory map for the DM642 Evaluation Module
- Block for targeting custom hardware—Custom C6000—Set the target preferences and memory map for your custom hardware based on a TI C6000 family processor

These blocks replace the C6000 Target Preferences block released in an earlier version. For details about the blocks, refer to the online Help for the Embedded Target for TI C6000 DSP. Each block has a reference page and you can find tutorial information about using the new blocks in “Targeting C6000 DSP Hardware” in the online Help system as well.

Added Interrupt Threshold Option in Configuration Parameters

To allow you to control how the compiler, linker, and assembler handle loops in your code, Embedded Target for TI C6000 DSP provides options that control interrupts in your code:

- **Interrupt threshold (-mi)**—Enables the interrupt threshold option the project and enables Interrupt threshold value (cycles) so you can set the threshold.
- **Interrupt threshold value (cycles)**—Sets the interrupt threshold for your code.

When you enable and set the threshold options, you are specifying the number of cycles that the compiler can disable interrupts. For more information, refer to the online Help system, and search for the keywords ‘interrupt’ and ‘threshold’.

Embedded Target for TI C6000 Requires CCS 3.0

To use this 2.3 release of Embedded Target for TI C6000 DSP, you must install Code Composer Studio 3.0 from Texas Instruments.

New Target Preferences Blocks Replace Existing C6000 Target Preferences Block

If you have models that use the earlier version of the C6000 Target Preferences block, those models now have a broken link to the block. The C6000 Target Preferences block is obsolete and is not provided in this release.

To update your models, replace the existing C6000 Target Preferences block with the appropriate new block from the C6000 Target Preferences library. Select either the block that is specifically identified for your target, or use the Custom C6000 block if your hardware is not one of the explicitly supported boards.

Compatibility Summary for Embedded Target for TI TMS320C6000 DSP

This table summarizes new features and changes that might cause incompatibilities when you upgrade from an earlier version, or when you use files on multiple versions. Details are provided with the description of the new feature or change.

Version (Release)	New Features and Changes with Version Compatibility Impact
Latest Version V3.1 (R2006b)	See the Compatibility Considerations subheading for this new feature or change: <ul style="list-style-type: none"> • “Code Composer Studio 3.2 Compatibility” • “Added Target Preferences blocks for C6455 DSK and PADK6727 Targets” • “New C6000 Memory Blocks”
Latest Version V3.0 (R2006a)	See the Compatibility Considerations subheading for this new feature or change: <ul style="list-style-type: none"> • “Code Composer Studio 3.1 Compatibility” • “Asynchronous Scheduler”
V2.4 (R14SP3)	None
V2.3 (R14SP2)	None

