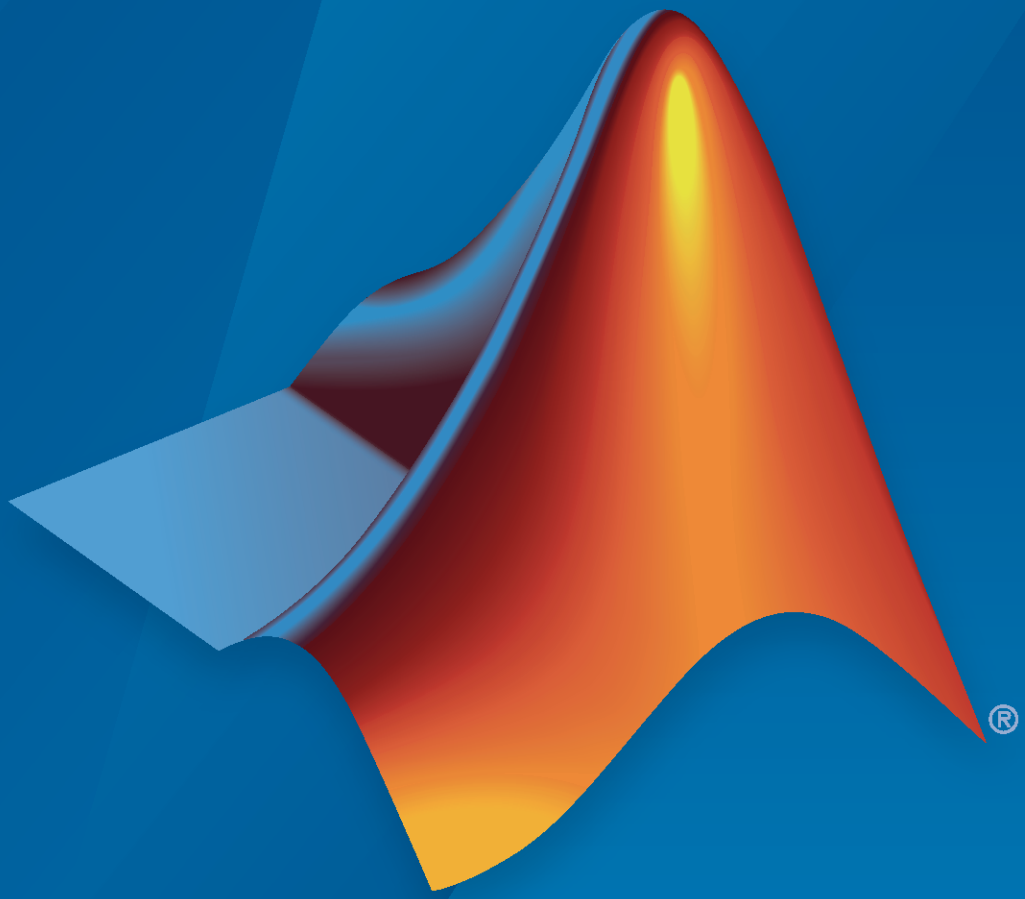
SoC Blockset™ Release Notes



MATLAB® & SIMULINK®

MathWorks®

How to Contact MathWorks

Latest news: `www.mathworks.com`

Sales and services: `www.mathworks.com/sales_and_services`

User community: `www.mathworks.com/matlabcentral`

Technical support: `www.mathworks.com/support/contact_us`

Phone: 508-647-7000

The MathWorks, Inc.
1 Apple Hill Drive
Natick, MA 01760-2098

SoC Blockset™ Release Notes

Contents

R2021a

R2020b

R2020a

R2019b

R2019a

R2021a

Version: 1.4

New Features

Bug Fixes

Memory Channel block supports vectors up to 512 bits

The Memory Channel block now supports vectors up to 512 bits. This feature enables you to create:

- A multichannel frame
- A multichannel sample
- A video frame with multiple pixels
- A video sample with multiple pixels
- A vectorized frame
- A vectorized pixel

To enable sample packing, open the Memory Channel block mask, and on the **Signal Attributes** tab, select the **Enable sample packing (last signal dimension as channel)** parameter.

When you select this parameter, the Memory Channel block packs data for the last dimension of the signal. For example, if the channel data-type is `uint32`, the dimensions are `[1024,4]`, and you select **Enable sample packing (last signal dimension as channel)**, then the memory channel generates 1024 read or write transactions of 128 bits. For this example, if you clear this sample packing parameter, the memory channel generates 4096 transactions of 32 bits each.

The combined width of the flattened signal must not exceed 512 bits. For more information, see "Enable sample packing (last signal dimension as channel)".

Memory Channel block simulates pipelined read and write operations

The Memory Channel block now simulates pipelined random access read and write operations when you set the **Channel type** to `AXI4 Random Access`. This feature enables a higher simulation fidelity of the memory random access throughput. For more information about the AXI4 interface see "Simplified AXI4 Master Interface".

Memory Channel block supports extended data types

The Memory Channel block now supports 16-bit and 128-bit integer data types when you set the **Channel type** parameter to `Software to AXI4 Stream via DMA` or `AXI4 Stream to Software via DMA`.

Audio blocks simulate capture and playback of hardware audio

The blockset now supports the simulation of hardware audio capture and playback. The Audio Playback Interface and Audio Playback blocks simulate the output of audio signals to one of the specified audio hardware peripherals in supported SoC devices. Similarly, the Audio Capture Interface and Audio Capture blocks simulate the capture of audio signals from one of the specified audio hardware peripherals in supported SoC devices. When you use the Audio Playback and Audio Capture blocks with the SoC Blockset™ Support Package for Embedded Linux® Devices, the blocks generate driver code that works with Linux audio devices that comply with **Advanced Linux Sound Architecture** (**ALSA**) .

Video blocks simulate capture and display of hardware video

The blockset now supports the simulation of hardware video capture and display. The Video Display Interface and Video Display blocks simulate the output of video signals to one of the specified video

outputs in supported SoC devices. Similarly, the Video Capture Interface and Video Capture blocks simulate the capture of video signals from one of the specified video peripherals in supported SoC devices. When you use the Video Display and Video Capture blocks with the SoC Blockset Support Package for Embedded Linux Devices, the blocks generate driver code that works with Linux video devices that comply with **Video4Linux (V4L2)** and **Simple Directmedia Layer (SDL)**.

PWM blocks support multiple comparators, variable frequency and variable phase inputs

The PWM Write and PWM Interface blocks now support up to two independent comparators for refined PWM output generation along with events. Comparator-generated events, when connected to the Task Manager block, provide more fine grained control over one PWM output cycle. The PWM blocks now supports explicit phase and frequency input to provide offset synchronization between multiple PWM modules and changing PWM frequency at runtime.

Updates to supported software

SoC Blockset now supports Xilinx® Vivado® 2020.1. For a full list of supported software, see “Supported Third-Party Tools and Hardware”.

Updates to supported hardware

SoC Blockset now supports the Zynq® UltraScale+™ RFSoC ZCU216 Evaluation Kit. For a full list of supported devices, see “Supported Third-Party Tools and Hardware”.

R2020b

Version: 1.3

New Features

Bug Fixes

AXI Stream: Stream data from software to a hardware IP

SoC Blockset now supports a memory protocol to stream data from software to an FPGA. Use the new Stream Write block to initiate and drive a stream from the software subsystem to the FPGA algorithm. To configure the memory channel for this mode, in the Memory Channel block set the **Channel type** parameter to `Software to AXI4-Stream via DMA`.

FPGA IP Core interrupt: Trigger an event-driven software task via interrupt

You can now issue an interrupt from an FPGA or a peripheral IP to software. Use the Interrupt Channel block to arbitrate between interrupt requests, process them one at a time, and send the requests to the Task Manager block for software processing.

ADC Interface Block: Simulate hardware ADC of signals

The ADC Interface and ADC Read blocks simulate the analog-to-digital conversion (ADC) of signals of hardware peripherals in supported SoC devices.

PWM Interface Block: Simulate hardware PWM output signals

The PWM Interface and PWM Write blocks simulate the pulse-width-modulation (PWM) output signals of hardware peripherals in supported SoC devices.

Multiprocessor Simulation: Simulate SoC models with multiple processors

The SoC Blockset software now supports the simulation of multiple processors contained in a single SoC. To simulate a specific processor in the SoC, you can uniquely configure the hardware settings of each Reference block mode. When you use the reference models in combination with a hardware support package, you can deploy the reference models to the specific processors on the SoC hardware.

Interprocessor Communication (IPC): Simulate communication between multiple processors

You can simulate IPC by using the Interprocess Data Read, Interprocess Data Channel, and Interprocess Data Write blocks. The Interprocess Data Channel block simulates hardware behavior for messages sent from one reference processor and received by another processor. The Interprocess Data Write block sends the messages, and the Interprocess Data Read block reads the received messages.

Task Manager Block Enhancements: Deploy tasks as ISRs on bare metal devices

The Task Manager block now supports the simulation and code generation of interrupt service routines (ISR) for bare metal devices. To simulate and manage tasks triggered by hardware interrupts

on systems without an operating system (OS), use the Task Manager block. Interrupts can be produced by the ADC Interface block and from FPGA through the ISR block.

Model Templates: Use added templates or enhanced existing template to design required SoC model

These added model templates are now available.

- RFSoC Template – Use this template to design an RFSoC-enabled wireless communication SoC model.
- Stream from Processor to FPGA Template – Use this template to design a data path from software (Processor) to hardware (FPGA).

Also, the existing Frame Buffer with HDMI Template model template has been enhanced. For more information on model templates, see Use Template to Create SoC Model.

Visualize logged Stateflow states in the Logic Analyzer

When you log signals in Stateflow® charts, you can use the **Logic Analyzer** to visualize the state changes. To log Stateflow states, in the **Simulation** tab, under **Prepare**, select a state logging option. In the Logic Analyzer, you'll see your states marked for logging in the left column.

SoC Blockset Support Package for Texas Instruments C2000 Processors: Generate, build, and deploy reference designs on TI’s C2000 processors

The SoC Blockset Support Package for Texas Instruments™ C2000™ Processors with Embedded Coder® can export reference designs to TI’s C2000™ devices. These reference designs can be used with Texas Instruments design tools.

This support package helps to automate the integration, execution, and verification of reference designs for the hardware boards using TI’s C2000 processors. Supported hardware boards include the Texas Instruments C2000Delfino MCU F28379D LaunchPad and Texas Instruments C2000 Delfino MCU F2837xD Control Card. For more information on these hardware boards, see https://www.ti.com/tool/LAUNCHXL-F28379D and https://www.ti.com/tool/TMDSCNCD28379D, respectively.

SoC Blockset Support Package for Embedded Linux Devices: Design, analyze, and prototype for embedded Linux devices

SoC Blockset Support Package for Embedded Linux Devices enables you to model, simulate, analyze, and prototype software on embedded Linux platforms using SoC Blockset. The support package features key capabilities including embedded C code generation with POSIX® threads and rate-monotonic scheduling (RMS), device driver integration, and device tree and Linux image customization.

Use IP core generation workflow for RFSoC devices (November 2020, Version 20.2.1)

Generate an RFSoC model template using **SoC Template Builder**, modify it to create an RFSoC IP, and then use IP core generation workflow to deploy the IP on a Xilinx RFSoC device. This workflow is

supported for Zynq UltraScale+ RFSoC ZCU111 boards. For additional information, see “Configure RFSoC Design Using SoC Template Builder” (SoC Blockset Support Package for Xilinx Devices).

Updates to supported software

SoC Blockset now supports Xilinx Vivado 2019.2. For a full list of supported software, see https://www.mathworks.com/help/releases/R2020b/soc/ug/supported-third-party-tools-and-hardware.html.

R2020a

Version: 1.2

New Features

Bug Fixes

Compatibility Considerations

socExportReferenceDesign Function: Automate the creation of custom reference design

Use the `socExportReferenceDesign` function to export a custom reference design from your SoC Blockset Simulink® model. You can then use IP Core Generation workflow (requires HDL Coder™) to generate a custom IP core and integrate it into your SoC reference design. For more information about the `socExportReferenceDesign` function, see `socExportReferenceDesign`.

SoC Algorithm Analysis Functions: Analyze MATLAB functions or Simulink models early in the design process

Use the `socFunctionAnalyzer` and `socModelAnalyzer` functions to analyze and compare different algorithms and models early in the design process. The `socFunctionAnalyzer` and `socModelAnalyzer` functions create a report detailing the number of operations in a MATLAB® function or Simulink model, respectively.

IP Core Register Read and Register Write Blocks: Model write operation from processor to hardware logic

The Register Write block and IP Core Register Read block model a write operation from a processor to hardware logic. The IP Core Register Read block receives data sent by a Register Write block from the processor. You can replace the Register Channel block with the IP Core Register Read block when writing from the processor to the FPGA, eliminating the need to wire signals through the model hierarchy. For an example using IP Core Register Read and Register Write blocks, see Packet-Based ADS-B Transceiver.

Memory Channel Block Enhancements: Extended data-type support

The Memory Channel block now supports 64-bit integer data types and up to 128-bit fixed-point data types.

I2C Master Block Enhancements: Extend configuration support to additional slave devices and support for clock stretching feature

The I2C Master block now provides configuration support to I2C slave devices with a 10-bit address. Using the clock stretching feature, the block can now synchronize with slow-running I2C slave devices. The block also supports a burst mode with a maximum of 256 bytes of data per transaction and provides a **statusReg** port to read the status of an I2C bus.

For compatibility considerations, see “I2C Master block has one data register input port and one data register output port” on page 3-3.

socBuilder Tool Enhancements: Improve validation checks and status messages

The **socBuilder** tool generates an FPGA validation report in HTML format. This report lists the warnings and errors, if any, encountered during model parsing. The tool now displays improved

status messages on the **SoC Builder > Build Model** wizard while performing synthesis, implementation, and bit-stream generation processes.

IO Data Source Block: Read data from timeseries object in MATLAB workspace

The IO Data Source block now supports `timeseries` object in the MATLAB workspace as an input source. This feature enables you to provide custom data along with time of occurrence as an input to the block.

Proxy Task Block: Support event-driven tasks

The Proxy Task block now supports event driven task triggers from the Task Manager block. For more information of event driven tasks, see Event-Driven Tasks.

Kernel Profiler: Perform kernel instrumentation profiling for unlimited duration

Previously, the kernel profiler performed kernel instrumentation profiling on the hardware for only a limited duration (based on the free disk storage available on the hardware). Now, the kernel profiler provides an option to perform kernel instrumentation profiling for unlimited time duration.

Updates to supported software

SoC Blockset now supports Xilinx Vivado 2019.1. For a full list of supported software, see Supported Third-Party Tools and Hardware.

Updates to supported hardware

SoC Blockset now supports Zynq UltraScale+ RFSoC ZCU111 Evaluation Kit. To simulate and prototype radio frequency (RF) effects with the RF Data Converter block, use the SoC Blockset Support Package for Xilinx Devices features. For details about this support package, see SoC Blockset Support Package for Xilinx Devices. For a full list of supported devices, see Supported Third-Party Tools and Hardware.

Functionality being removed or changed

I2C Master block has one data register input port and one data register output port

Behavior change

In R2019a, the I2C Master block has input data register ports **dataReg**, **dataReg1**, **dataReg2**, and **dataReg3** and output response data register ports **respData**, **respData1**, **respData2**, and **respData3**. These data register ports support a maximum of 16 bytes per transaction. In R2020a, these register ports are replaced with single data register ports: input port **dataReg** and output port **respData**. Each of these single data register ports support a maximum of 256 bytes per transaction.

In R2020a, Simulink errors if you open a model that was created in an earlier release and that contains an I2C Master block. In this case, connections to ports **dataReg1**, **dataReg2**, **dataReg3**, **respData1**, **respData2**, and **respData3** are either missing or reconnected to empty ports on the

block. Manually check and update the port connections in your model to proceed further. For more information, see I2C Master block documentation.

Signal data no longer streams to the Logic Analyzer when signal logging is disabled
Behavior change

Previously, signals marked for logging have streamed to the **Logic Analyzer**, regardless of the setting for **Signal logging** in the model configuration parameters. Starting in R2020a, signals marked for logging stream to the Logic Analyzer *only* when signal logging is enabled for a model.

To view data in the Logic Analyzer, you must enable signal logging for the model. (Logging is on by default.) To enable signal logging, open **Model Settings** from the toolstrip, navigate to the **Data Import/Export** pane, and select **Signal logging**.

R2019b

Version: 1.1

New Features

Bug Fixes

Proxy Task Block: Model the effect of a task in your application without an explicit task implementation

The Proxy Task block models the effect of a task on your application without defining the implementation of the task. You can configure the block execution timing using the Task Manager block. The Proxy Task block serves as a placeholder for a task you are currently developing or plan to develop.

Testbench Task Block: Model the effect of your external task competing for resources with an application

The Testbench Task block models the effect of a task in an external application competing for execution resources with the tasks in your application. You can configure the block execution timing by using the Task Manager block.

Playback control behavior changed for Logic Analyzer in referenced models

When you use the **Logic Analyzer** in a referenced model, the playback controls in the Logic Analyzer now match the playback controls of the last model you interacted with that logs data to the scope. For example, if you opened the Logic Analyzer from a model referenced by another model with the Model block, the run button  in the scope runs the top level model. If the referenced model is opened as a top model, the run button runs the referenced model in isolation.

Updates to supported software

SoC Blockset now supports to these software versions. For a full list of supported software, see Supported Third-Party Tools and Hardware.

- Xilinx Vivado 2018.3
- Intel® Quartus® Prime 18.1

IO Data Source Block: Read data from a recorded data file at the same time interval at which it was recorded on the hardware board

You can now use the IO Data Source block to read data from a recorded data file at the same time interval as it was recorded on the hardware board.

When you enable the event port of the IO Data Source block and connect it to the Task Manager block, the IO Data Source block reads event signals and generates them on the event port as they were recorded in the recorded data file. Previously, the IO Data Source block generated events based on the sample time specified on the block mask.

This feature enables you to see the real behavior of the data from hardware I/O peripherals in Simulation.

Kernel Profiler: Monitor and record execution times of tasks with LTTng

You can use a Kernel profiler to monitor and record model-related processes and threads running on the Linux of your hardware board without instrumenting code.

Hardware Memory Diagnostics: View additional latencies and data overflow information from FPGA execution

In addition to viewing **Burst request to first transfer complete** latency information from a design running on field programmable gate array (FPGA), you can now view **Burst execution** and **Burst last transfer to complete** latencies information. This feature is similar to that of Simulation Performance Plots.

You can also collect and view data overflow that occurred in bandwidth, burst, and latency memory diagnostics metrics.

R2019a

Version: 1.0

New Features

Introducing SoC Blockset: Design, evaluate, and implement SoC hardware and software architectures

SoC Blockset provides Simulink blocks and visualization tools for modeling, simulating, and analyzing hardware and software architectures for ASICs, FPGAs, and systems on a chip (SoC). You can build your system architecture using memory models, bus models, and I/O models, and simulate the architecture together with the algorithms.

SoC Blockset lets you simulate memory and internal and external connectivity, as well as scheduling and OS effects, using generated test traffic or real I/O data. You can quickly explore different system architectures, estimate interface complexity for hardware and software partitioning, and evaluate software performance and hardware utilization.

SoC Blockset exports reference designs for Xilinx and Intel FPGA devices and SoC platforms, including Zynq-7000, UltraScale+, and Intel SoC FPGAs. These reference designs can be used with Xilinx and Intel design tools.

SoC Blockset Support Package for Xilinx Devices: Generate, build, and deploy reference designs on Xilinx devices

The SoC Blockset Support Package for Xilinx Devices with Embedded Coder or HDL Coder can export reference designs for Xilinx FPGA devices and SoC platforms. These reference designs can be used with Xilinx design tools.

The support package helps to automate integration, execution, and verification of reference designs for the SoC platforms, including Xilinx Artix®-7, Xilinx Kintex®-7, XilinxZynq, and XilinxZynqUltraScale+.

SoC Blockset Support Package for Intel Devices: Generate, build, and deploy reference designs on Intel devices

The SoC Blockset Support Package for Intel Devices with Embedded Coder or HDL Coder can export reference designs for Intel FPGA devices and SoC platforms. These reference designs can be used with Intel design tools.

The support package helps to automate integration, execution, and verification of reference designs for the SoC platforms, including Intel Arria® 10 and Intel Cyclone® V.