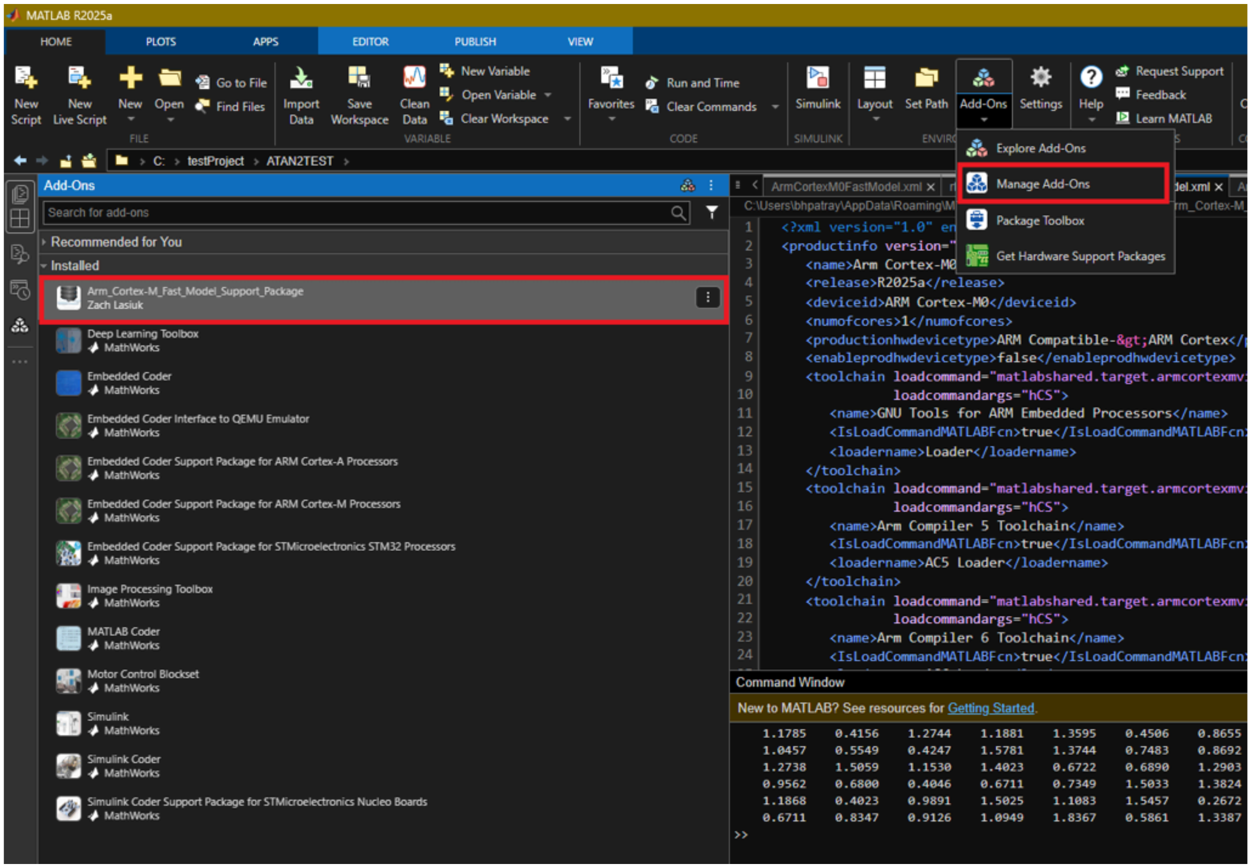
Fast Model SPKG Workflow

This documentation mentions how to run PIL using fast Model Support Package and benchmark in MATLAB 25a using ARM Development Studio latest version.

1) Installation of Fast Model SPKG in 25a

1. Install Fast Model SPKG from Arm_Cortex-M_Fast_Model_Support_Package - File Exchange - MATLAB Central . This is compatible with 2018b, to make it work for 25a, few work arounds were needed.
2. Once Installed, In Add-Ons → Manage Add-Ons, newly installed SPKG is shown.

Snapshot of ARM_Cortex-M_Fast_Model_Support_Package



```
<?xml version="1.0" en
<productinfo version="
   <name>Arm Cortex-M0
   <release>R2025a</release>
   <deviceid>ARM Cortex-M0</deviceid>
   <numofcores>1</numofcores>
   <productionhwdevicetype>ARM Compatible-&gt;ARM Cortex</
   <enableprodhwdevicetype>false</enableprodhwdevicetype>
   <toolchain loadcommand="matlabshared.target.armcortexmv
              loadcommandargs="hCS">
      <name>GNU Tools for ARM Embedded Processors</name>
      <IsLoadCommandMATLABFcn>true</IsLoadCommandMATLABFcn
      <loadername>Loader</loadername>
   </toolchain>
   <toolchain loadcommand="matlabshared.target.armcortexmv
              loadcommandargs="hCS">
      <name>Arm Compiler 5 Toolchain</name>
      <IsLoadCommandMATLABFcn>true</IsLoadCommandMATLABFcn
      <loadername>AC5 Loader</loadername>
   </toolchain>
   <toolchain loadcommand="matlabshared.target.armcortexmv
              loadcommandargs="hCS">
      <name>Arm Compiler 6 Toolchain</name>
      <IsLoadCommandMATLABFcn>true</IsLoadCommandMATLABFcn
```

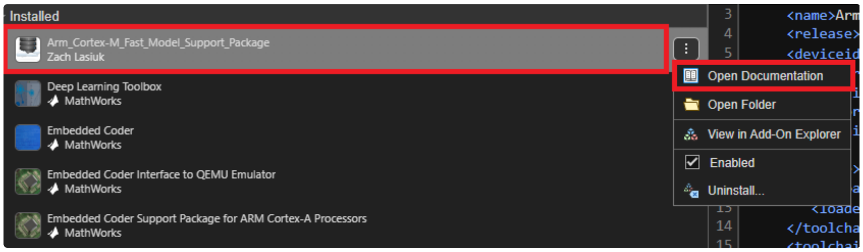
New to MATLAB? See resources for Getting Started.

```
    1.1785    0.4156    1.2744    1.1881    1.3595    0.4506    0.8655
    1.0457    0.5549    0.4247    1.5781    1.3744    0.7483    0.8692
    1.2738    1.5059    1.1530    1.4023    0.6722    0.6890    1.2903
    0.9562    0.6800    0.4046    0.6711    0.7349    1.5033    1.3824
    1.1868    0.4023    0.9891    1.5025    1.1083    1.5457    0.2672
    0.6711    0.8347    0.9126    1.0949    1.8367    0.5861    1.3387
>>
```

ARM_Cortex-M_Fast_Model_Support_Package

3. Documentation for this newly installed SPKG can be obtained as shown in snapshot below -

Open Documentation Snapshot



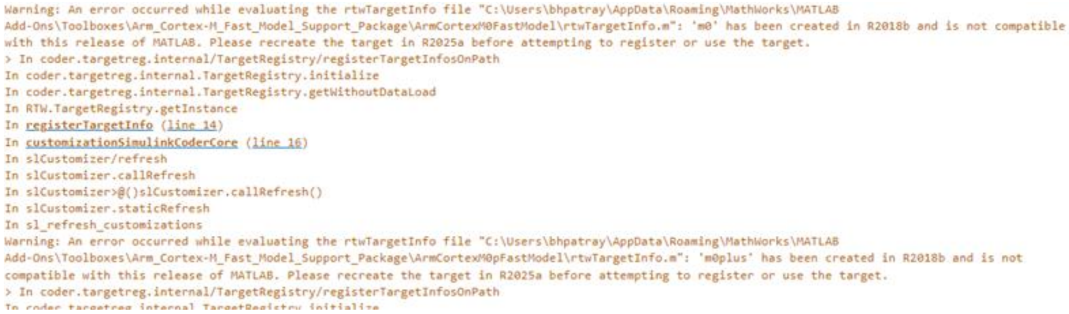
Open Documentation

4. Follow the below steps to make SPKG work in 25a.

Resolution for SPKG incompatibility

Issue - SPKG won't show up and we won't see any Fast Model related hardware in Simulink.

```
sl_refresh_customizations
```

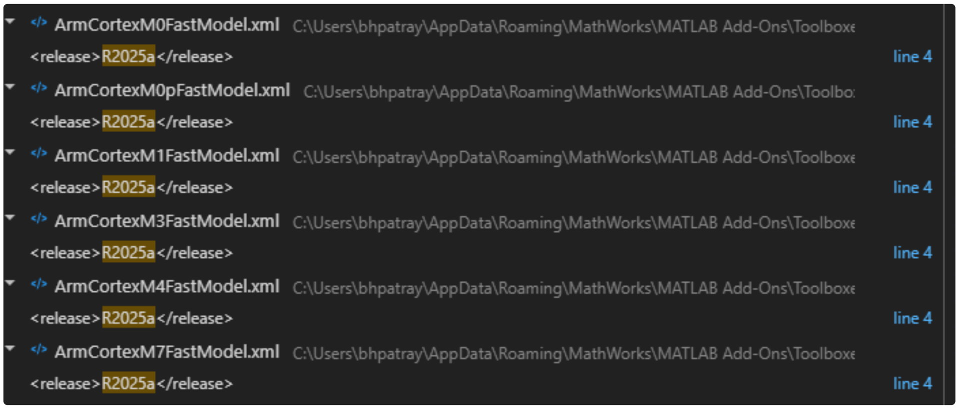


```
Warning: An error occurred while evaluating the rtwTargetInfo file "C:\Users\bhpatray\AppData\Roaming\MathWorks\MATLAB
Add-Ons\Toolboxes\Arm_Cortex-M_Fast_Model_Support_Package\ArmCortexM0FastModel\rtwTargetInfo.m": 'm0' has been created in R2018b and is not compatible
with this release of MATLAB. Please recreate the target in R2025a before attempting to register or use the target.
> In coder.targetreg.internal/TargetRegistry/registerTargetInfosOnPath
In coder.targetreg.internal.TargetRegistry.initialize
In coder.targetreg.internal.TargetRegistry.getWithoutDataLoad
In RTW.TargetRegistry.getInstance
In registerTargetInfo (line 14)
In customizationSimulinkCoderCore (line 16)
In slCustomizer/refresh
In slCustomizer.callRefresh
In slCustomizer>@()slCustomizer.callRefresh()
In slCustomizer.staticRefresh
In sl_refresh_customizations
Warning: An error occurred while evaluating the rtwTargetInfo file "C:\Users\bhpatray\AppData\Roaming\MathWorks\MATLAB
Add-Ons\Toolboxes\Arm_Cortex-M_Fast_Model_Support_Package\ArmCortexM0pFastModel\rtwTargetInfo.m": 'm0plus' has been created in R2018b and is not
compatible with this release of MATLAB. Please recreate the target in R2025a before attempting to register or use the target.
> In coder.targetreg.internal/TargetRegistry/registerTargetInfosOnPath
In coder.targetreg.internal.TargetRegistry.initialize
```

Error in sl_referesh_customizations

Resolution -

- Open Folder (`Manage Add-Ons --> ARM_Cortex-M_Fast_Model_Support_Package --> Open Folder` section in manage add-ons) of this newly installed SPKG
- Search for "R2018b" in the folder. Replace all instances of R2018b with R2025a in all xml files.



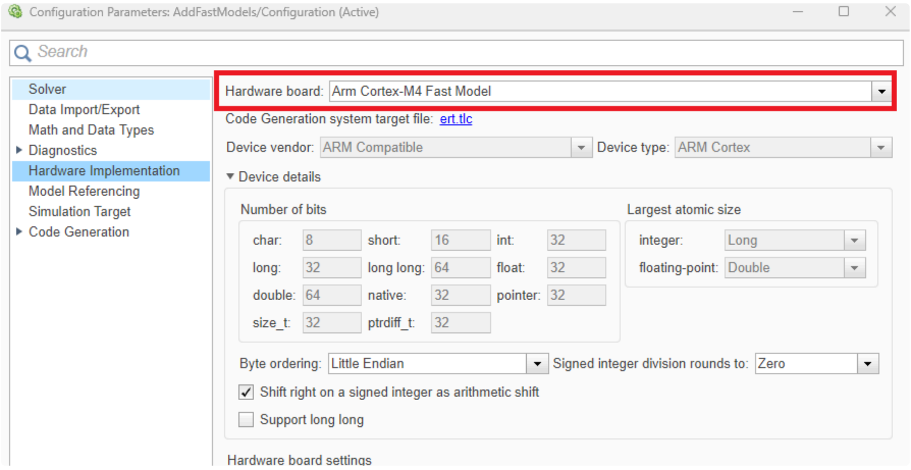
```
ArmCortexM0FastModel.xml C:\Users\bhpatray\AppData\Roaming\MathWorks\MATLAB Add-Ons\Toolboxe
<release>R2025a</release>                                             line 4
ArmCortexM0pFastModel.xml C:\Users\bhpatray\AppData\Roaming\MathWorks\MATLAB Add-Ons\Toolbo:
<release>R2025a</release>                                             line 4
ArmCortexM1FastModel.xml C:\Users\bhpatray\AppData\Roaming\MathWorks\MATLAB Add-Ons\Toolboxe
<release>R2025a</release>                                             line 4
ArmCortexM3FastModel.xml C:\Users\bhpatray\AppData\Roaming\MathWorks\MATLAB Add-Ons\Toolboxe
<release>R2025a</release>                                             line 4
ArmCortexM4FastModel.xml C:\Users\bhpatray\AppData\Roaming\MathWorks\MATLAB Add-Ons\Toolboxe
<release>R2025a</release>                                             line 4
ArmCortexM7FastModel.xml C:\Users\bhpatray\AppData\Roaming\MathWorks\MATLAB Add-Ons\Toolboxe
<release>R2025a</release>                                             line 4
```

Replaced 18b with 25a

- After this run `sl_refresh_customizations` , that will resolve the issue and in hardware section we can see fast Models SPKG working

5. Open Simulink model. Got to 'Hardware Implementation' setting. Under hardware board, any Arm Cortex-M Fast Model can be selected now.

M4 fast Model hardware configure screenshot



Configuration Parameters: AddFastModels/Configuration (Active)

Hardware board: Arm Cortex-M4 Fast Model

Code Generation system target file: ert.tlc

Device vendor: ARM Compatible   Device type: ARM Cortex

Device details

Number of bits

char: 8   short: 16   int: 32
long: 32   long long: 64   float: 32
double: 64   native: 32   pointer: 32
size_t: 32   ptrdiff_t: 32

Largest atomic size

integer: Long
floating-point: Double

Byte ordering: Little Endian   Signed integer division rounds to: Zero

☑ Shift right on a signed integer as arithmetic shift

☐ Support long long

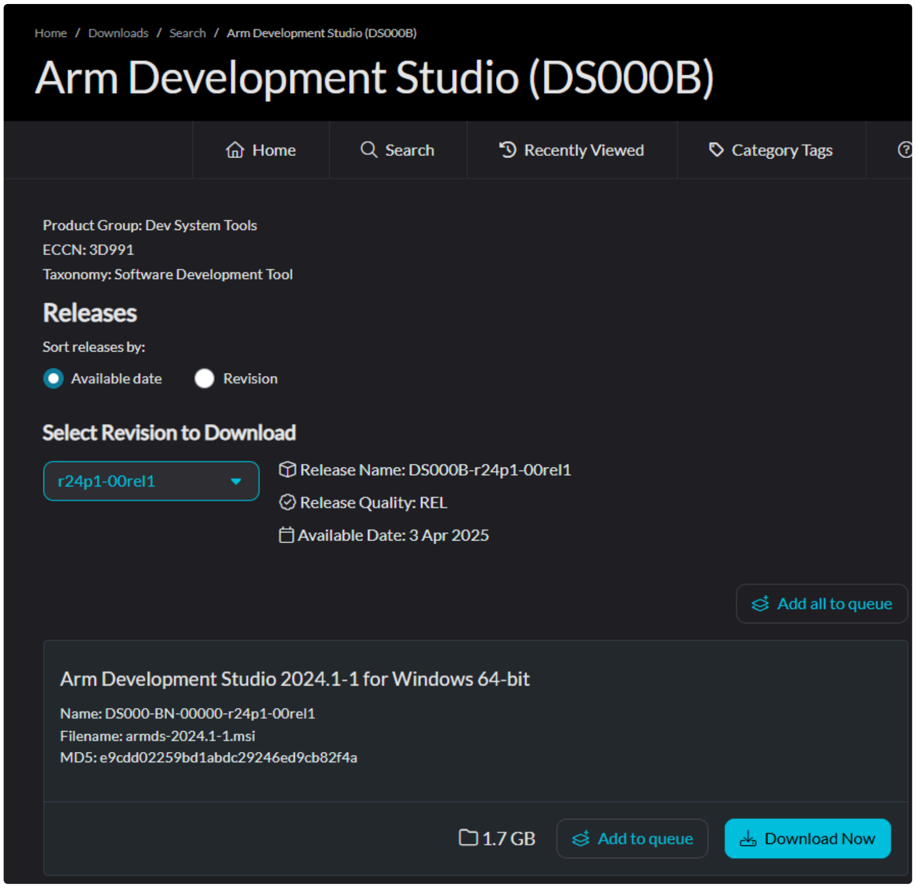
Hardware board settings

M4 fast Model

2) ARM Development Studio installation

1. Install ARM Development Studio. Fast Model support package documentation is pointing DS-5, but this workflow is tested with latest ARM development studio release (2024.1-1 for Windows 64-bit). This can be downloaded from this link.

ARM Development Studio Snapshot



Arm Development Studio (DS000B)

Product Group: Dev System Tools
ECCN: 3D991
Taxonomy: Software Development Tool

Releases

Select Revision to Download

Release Name: DS000B-r24p1-00rel1
Release Quality: REL
Available Date: 3 Apr 2025

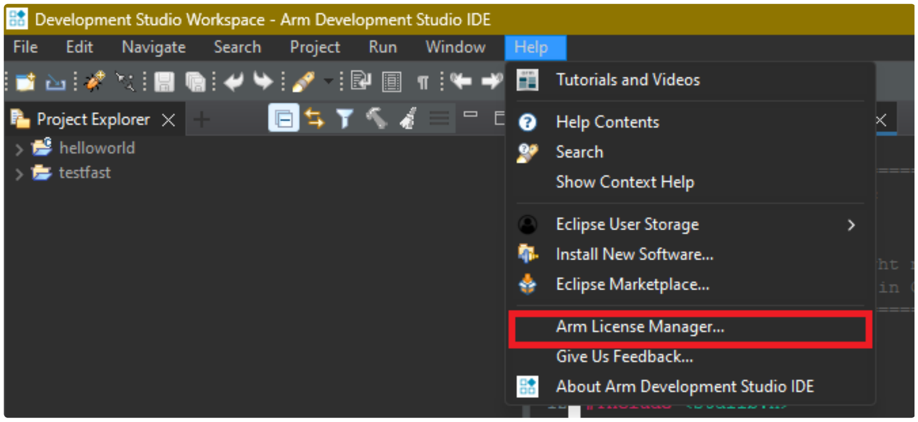
Arm Development Studio 2024.1-1 for Windows 64-bit

Name: DS000-BN-00000-r24p1-00rel1
Filename: armds-2024.1-1.msi
MD5: e9cdd02259bd1abdc29246ed9cb82f4a

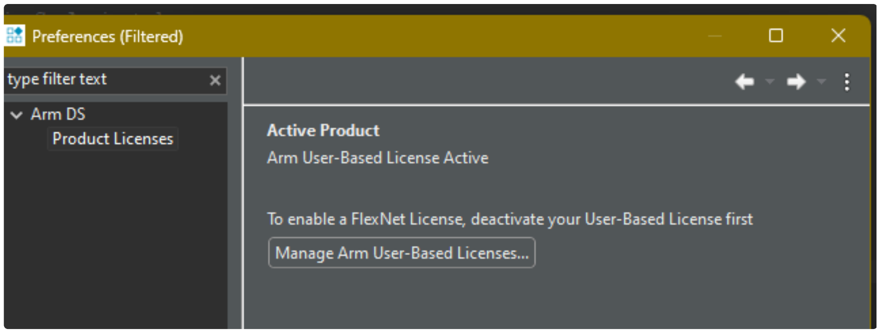
Download ARM Development Studio

2. Install and open ARM development studio. Open Arm DS IDE. User needs to have valid ARM development studio License. Our team had a license which we used for license configuration.

Configure License



Arm License Manager



Preferences (Filtered)

type filter text

Arm DS

Product Licenses

Active Product

Arm User-Based License Active

To enable a FlexNet License, deactivate your User-Based License first

Manage Arm User-Based Licenses...

License Configured

- This license can also be checked in online ARM portal through this link

---

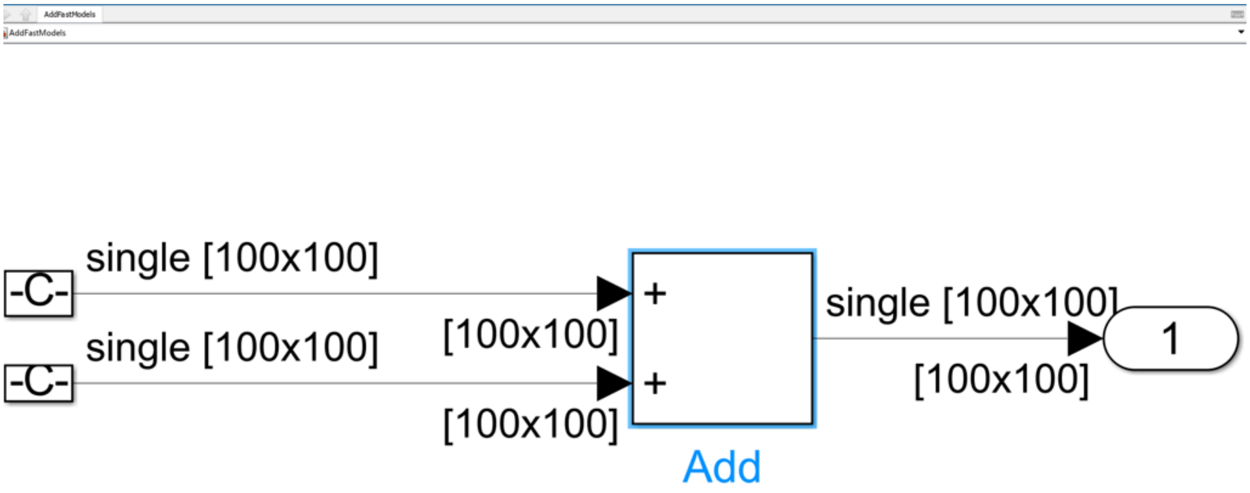
3) Setup In MATLAB

1. Set the relevant environment variables in MATLAB. Documentation for SPKG points to DS-5 paths, But paths are modified to use ARM Development Studio.

```
setenv('ARM_PRODUCT_PATH',"C:\Program Files\Arm\Development Studio 2024.1-1\sw\mappings")
setenv("ARM_TOOL_VARIANT", "ult")
```

2. Create a Model, here simple addition Model is created. AddFastModels.slx

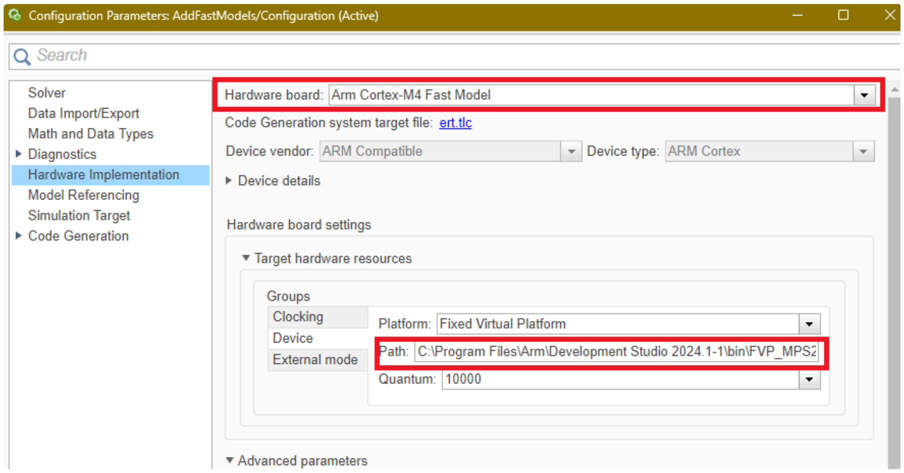
Model Snapshot



Add Block

3. Open Model settings. Set the hardware to M4 fast Model. Under Target hardware resources → Device, Update the Path to `“C:\Program Files\Arm\Development Studio 2024.1-1\bin\FVP_MPS2_Cortex-M4.exe”` (FVP path needs to be updated based on hardware chosen)

M4 fast Model configuration



Configuration Parameters: AddFastModels/Configuration (Active)

Hardware board: Arm Cortex-M4 Fast Model

Code Generation system target file: ert.tlc

Device vendor: ARM Compatible

Device type: ARM Cortex

Platform: Fixed Virtual Platform

Path: C:\Program Files\Arm\Development Studio 2024.1-1\bin\FVP_MPS2

Quantum: 10000

M4 Fast Model

4. We faced compilation errors during build, So we updated Include paths in `Code Generation >> Build configuration >>  C Compiler` as shown below :

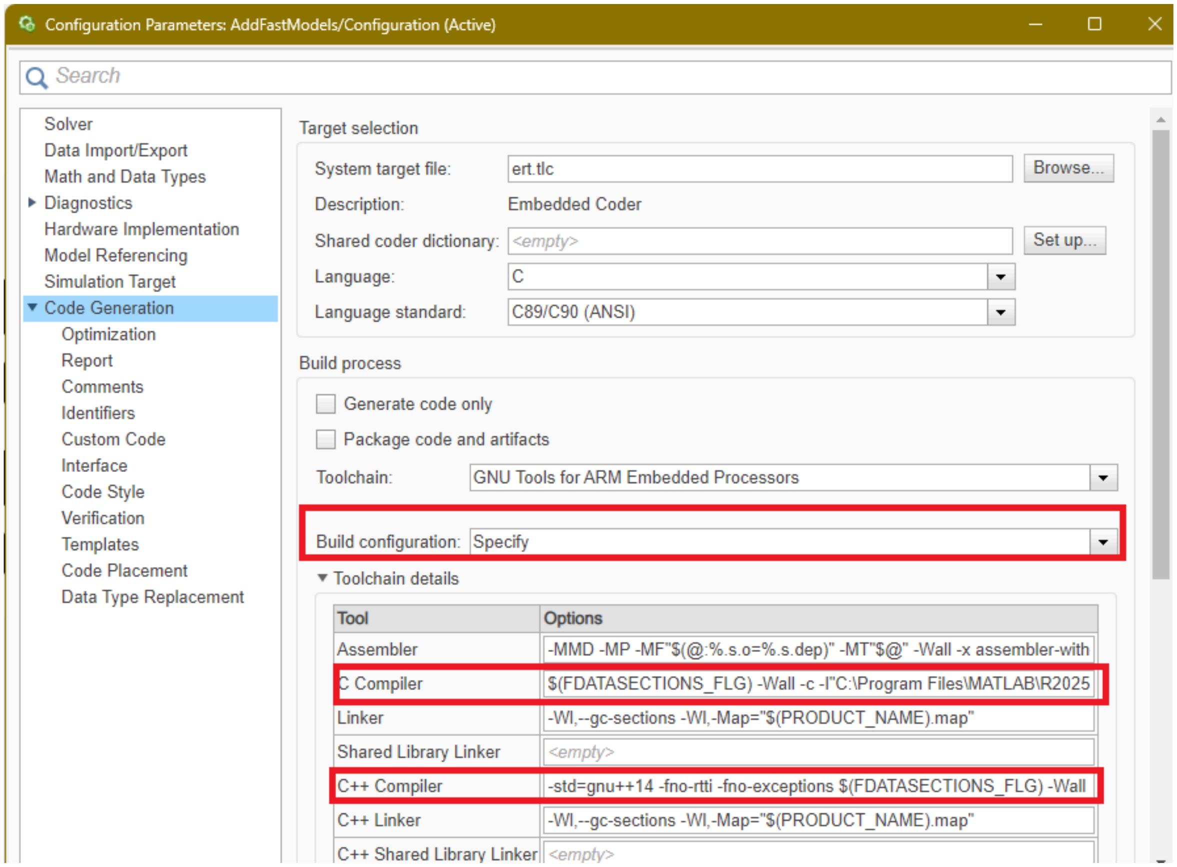
Resolution for compilation errors.

Issue - Once we tried to build , we faced Compilation error

Resolution -

- Append `-I"C:\Program Files\MATLAB\R2025a\toolbox\coder\rtiostream\src"` in C and C++ compiler flags as shown in below snapshot.

Updating C and C++ flags



Target selection

System target file: ert.tlc

Description: Embedded Coder

Shared coder dictionary: <empty>

Language: C

Language standard: C89/C90 (ANSI)

Build process

☐ Generate code only

☐ Package code and artifacts

Toolchain: GNU Tools for ARM Embedded Processors

Build configuration: Specify

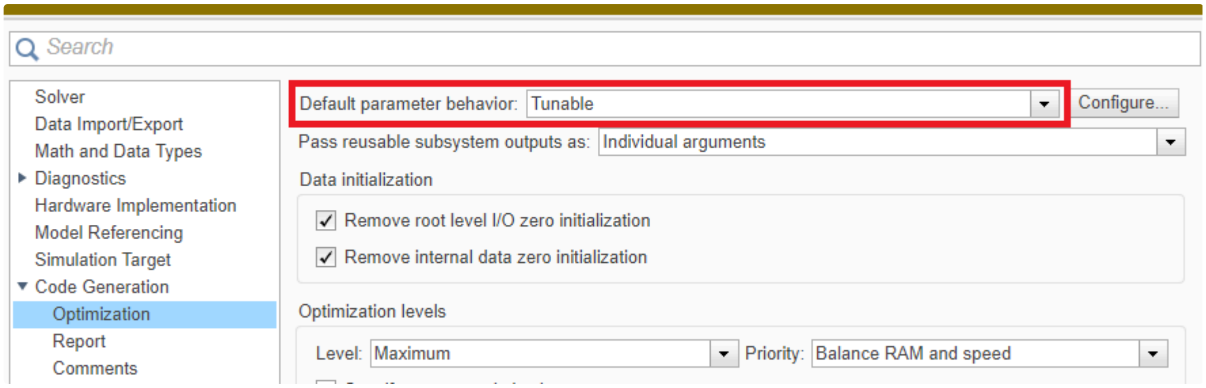
▾ Toolchain details

Tool	Options
Assembler	-MMD -MP -MF"$(@:%.s.o=%.s.dep)" -MT"$@" -Wall -x assembler-with
C Compiler	$(FDATASECTIONS_FLG) -Wall -c -I"C:\Program Files\MATLAB\R2025
Linker	-Wl,--gc-sections -Wl,-Map="$(PRODUCT_NAME).map"
Shared Library Linker	<empty>
C++ Compiler	-std=gnu++14 -fno-rtti -fno-exceptions $(FDATASECTIONS_FLG) -Wall
C++ Linker	-Wl,--gc-sections -Wl,-Map="$(PRODUCT_NAME).map"
C++ Shared Library Linker	<empty>

C Compiler, C++ Compiler

5. Set the default parameter behavior to tunable as shown in snapshot below.

Default parameter behavior setting



Default parameter behavior: Tunable

Pass reusable subsystem outputs as: Individual arguments

Data initialization

☑ Remove root level I/O zero initialization

☑ Remove internal data zero initialization

Optimization levels

Level: Maximum Priority: Balance RAM and speed

Tunable

4) Simulink Model PIL Execution

Run PIL simulation using below script

```
% Fast Model PIL workflow
```

AddFastModels.slx
15 Jul 2025, 09:09 AM

```
rng(0); in1 = rand(10,10,'single'); in2 = in1;
ModelName = 'AddFastModels'; % Model Name needs to be updated when other model is used.
open_system(ModelName);
load_system(ModelName);
set_param(ModelName,'SimulationMode','processor-in-the-loop (pil)');
outFastModel = sim(ModelName);
```

Output Log for Fast Model PIL is shown below -

Output Log

```
### # Output for PIL - outFastModel = sim(ModelName)
### Skipped unpacking from Simulink cache file "AddFastModels.slxc" because the relevant build artifacts on disk are up to date.
### Searching for referenced models in model 'AddFastModels'.
### Total of 1 models to build.
### Starting build procedure for: AddFastModels
### Generating code and artifacts to 'Model specific' folder structure
### Generating code into build folder: C:\testProject\ATAN2TEST\AddFastModels_ert_rtw
### Generated code for 'AddFastModels' is up to date because no structural, parameter or code replacement library changes were found.
### Using toolchain: GNU Tools for ARM Embedded Processors
### 'C:\testProject\ATAN2TEST\AddFastModels_ert_rtw\AddFastModels.mk' is up to date.
### Building 'AddFastModels': "C:\PROGRA~1\MATLAB\R2025a\bin\win64\gmake"  -f AddFastModels.mk buildobj

C:\testProject\ATAN2TEST\AddFastModels_ert_rtw>cd .

C:\testProject\ATAN2TEST\AddFastModels_ert_rtw>if "buildobj" == "" ("C:\PROGRA~1\MATLAB\R2025a\bin\win64\gmake"  -f AddFastModels.mk all )  else ("C:\PROGRA~1\MATLAB\R2025a\bin\win64\gmake"  -f AddFastModels.mk buildobj )
"### Successfully generated all binary outputs."

C:\testProject\ATAN2TEST\AddFastModels_ert_rtw>exit /B 0
### Successful completion of build procedure for: AddFastModels

Build Summary

Top model targets:

Model          Build Reason                                 Status          Build Duration
=========================================================================================
AddFastModels  Compilation artifacts were out of date.  Code compiled.  0h 0m 5.0574s

1 of 1 models built (0 models already up to date)
Build duration: 0h 0m 5.8001s
### Connectivity configuration for component "AddFastModels": m4 ###
### Preparing to start PIL simulation ...
### Using toolchain: GNU Tools for ARM Embedded Processors
### 'C:\testProject\ATAN2TEST\AddFastModels_ert_rtw\pil\AddFastModels.mk' is up to date.
### Building 'AddFastModels': "C:\PROGRA~1\MATLAB\R2025a\bin\win64\gmake"  -f AddFastModels.mk all

C:\testProject\ATAN2TEST\AddFastModels_ert_rtw\pil>cd .

C:\testProject\ATAN2TEST\AddFastModels_ert_rtw\pil>if "all" == "" ("C:\PROGRA~1\MATLAB\R2025a\bin\win64\gmake"  -f AddFastModels.mk all )  else ("C:\PROGRA~1\MATLAB\R2025a\bin\win64\gmake"  -f AddFastModels.mk all )
```

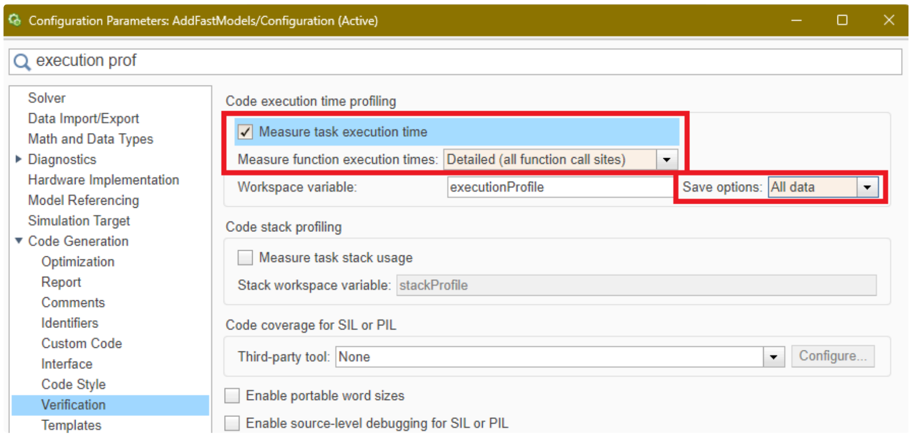
```
"### Invoking postbuild tool "Binary Converter" ..."
"C:/ProgramData/MATLAB/SupportPackages/R2025a/3P.instrset/gnuarm-armcortex.instrset/win/bin/arm-none-eabi-objcopy" -O binary ./AddFastModels.elf ./AddFastModels.bin
"### Done invoking postbuild tool."
"### Invoking postbuild tool "Hex Converter" ..."
"C:/ProgramData/MATLAB/SupportPackages/R2025a/3P.instrset/gnuarm-armcortex.instrset/win/bin/arm-none-eabi-objcopy" -O ihex ./AddFastModels.elf ./AddFastModels.hex
"### Done invoking postbuild tool."
"### Invoking postbuild tool "Executable Size" ..."
"C:/ProgramData/MATLAB/SupportPackages/R2025a/3P.instrset/gnuarm-armcortex.instrset/win/bin/arm-none-eabi-size" ./AddFastModels.elf
   text    data     bss     dec     hex filename
  11299   80104   40560  131963   2037b ./AddFastModels.elf
"### Done invoking postbuild tool."
"### Successfully generated all binary outputs."

C:\testProject\ATAN2TEST\AddFastModels_ert_rtw\pil>exit /B 0
### Updating code generation report with PIL files ...
Warning: Model Web view in code generation report requires Simulink Report Generator, which is not installed.
> In Simulink.report.ReportInfo/emitWebview
In rtw.report.ReportInfo/emitHTML>loc_emitHTML_V2
In rtw.report.ReportInfo/emitHTML
In rtw.report.ReportInfo/generate>locGenerate
In rtw.report.ReportInfo/generate
In rtw.report.generate
In coder.connectivity/SimulinkInterface/updateReport (line 695)
In rtw.pil/SILPILInterface/buildApplication (line 1191)
In rtw.pil.ModelBlockPIL.XILBuildHook (line 366)
In slprivate
In pil_target_sim
In sl_feval
### Starting application: 'AddFastModels_ert_rtw\pil\AddFastModels.elf'
### Downloading application...C:\testProject\ATAN2TEST\AddFastModels_ert_rtw\pil\AddFastModels.elf
Simulation complete. Application will now be terminated safely.

outFastModel = 

  Simulink.SimulationOutput:

                   tout: [51x1 double]
                   yout: [1x1 Simulink.SimulationData.Dataset]

     SimulationMetadata: [1x1 Simulink.SimulationMetadata]
           ErrorMessage: [0x0 char]
```

5) Benchmarking

1. Set profiling to true.

Profiling set to true snapshot

Set `Measure task execution time --> True`, `Measure function execution times --> Detailed`, Workspace `save options → All Data`



Code execution time profiling

☑ Measure task execution time

Measure function execution times: Detailed (all function call sites)

Workspace variable: executionProfile

Save options: All data

Code stack profiling

□ Measure task stack usage

Stack workspace variable: stackProfile

Code coverage for SIL or PIL

Third-party tool: None

□ Enable portable word sizes

□ Enable source-level debugging for SIL or PIL

execution profile

2. Generate `ARM Cortex-M4 MPS2` (M4 QEMU), `STM32 Nucleo F401RE` and `ARM Cortex-M4 Fast Model` execution times.

Script to generate execution timings

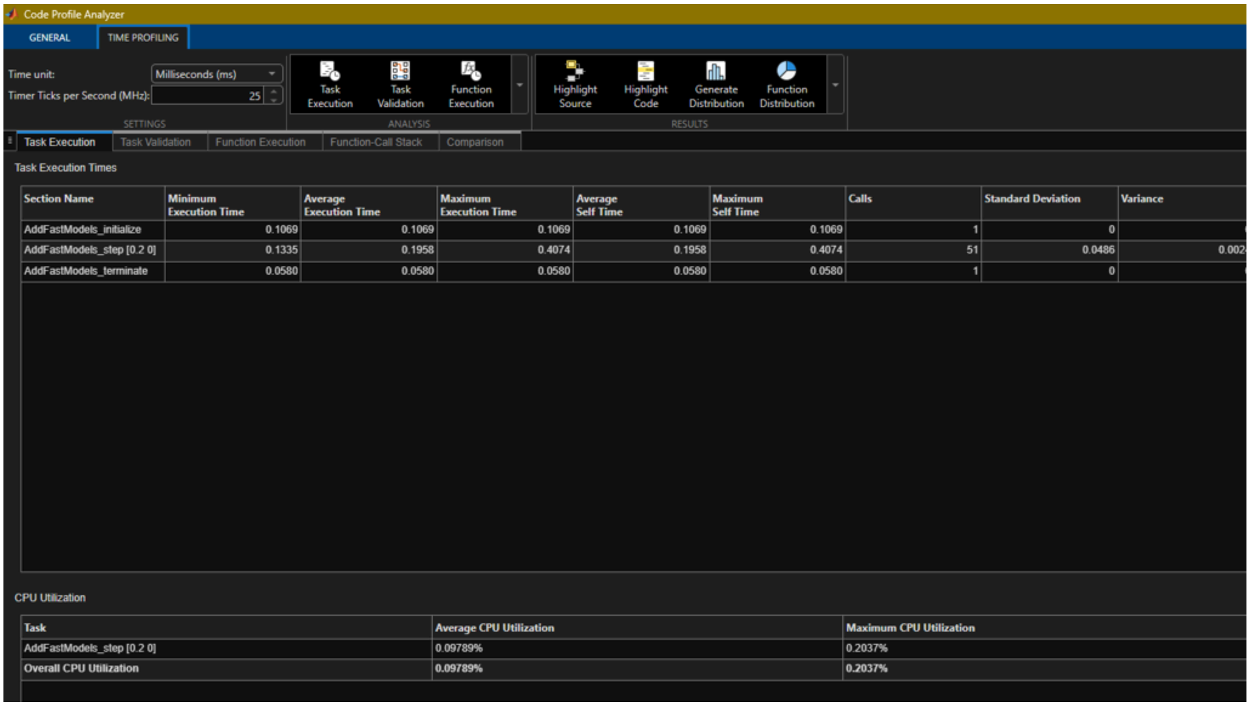
```
%% Compare timings.
in1 = rand(10,10,'single'); in2 = in1;
ModelName = 'AddFastModels';
open_system(ModelName);
load_system(ModelName);
%%
% M4 QEMU
set_param(ModelName,"HardwareBoard",'ARM Cortex-M4 (MPS2)');
set_param(ModelName,'SimulationMode','processor-in-the-loop (pil)');
outM4QEMU = sim(ModelName);

%%
% M4 Fast Model FVP
set_param(ModelName,"HardwareBoard",'Arm Cortex-M4 Fast Model');
% Set device to "C:\Program Files\Arm\Development Studio2024.1-1\bin\FVP_MPS2_Cortex-M4.exe"
set_param(ModelName,'SimulationMode','processor-in-the-loop (pil)');
outM4FM = sim(ModelName);

%%
% M4 Hardware
set_param(ModelName,"HardwareBoard",'STM32 Nucleo F401RE');
set_param(ModelName,'SimulationMode','processor-in-the-loop (pil)');
outM4Hardware = sim(ModelName);
```

3. Visualize benchmark numbers on QEMU, M4 Fast Model and Hardware.

QEMU Benchmarking

Timings of M4QEMU - `coder.profile.show(outM4QEMU.executionProfile)`

Clock - 25MHz



Time unit: Milliseconds (ms)
Timer Ticks per Second (MHz): 25

Task Execution Times

Section Name	Minimum Execution Time	Average Execution Time	Maximum Execution Time	Average Self Time	Maximum Self Time	Calls	Standard Deviation	Variance
AddFastModels_initialize	0.1069	0.1069	0.1069	0.1069	0.1069	1	0	
AddFastModels_step [0.2 0]	0.1335	0.1958	0.4074	0.1958	0.4074	51	0.0486	0.002
AddFastModels_terminate	0.0580	0.0580	0.0580	0.0580	0.0580	1	0	

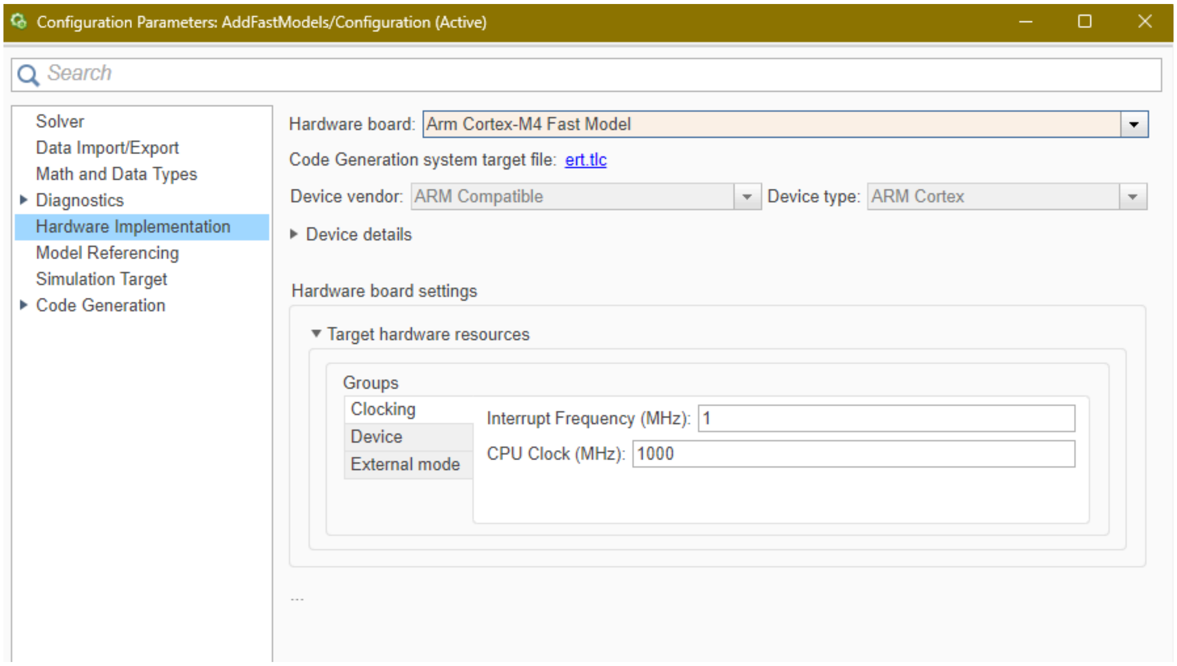
CPU Utilization

Task	Average CPU Utilization	Maximum CPU Utilization
AddFastModels_step [0.2 0]	0.09789%	0.2037%
Overall CPU Utilization	0.09789%	0.2037%

M4 QEMU timings

Fast Model Benchmarking

Default M4 fast Model Values (Interrupt Frequency - 1, CPU Clock - 1000 MHz, Quantum - 10000)



Hardware board: Arm Cortex-M4 Fast Model

Code Generation system target file: ert.tlc

Device vendor: ARM Compatible   Device type: ARM Cortex

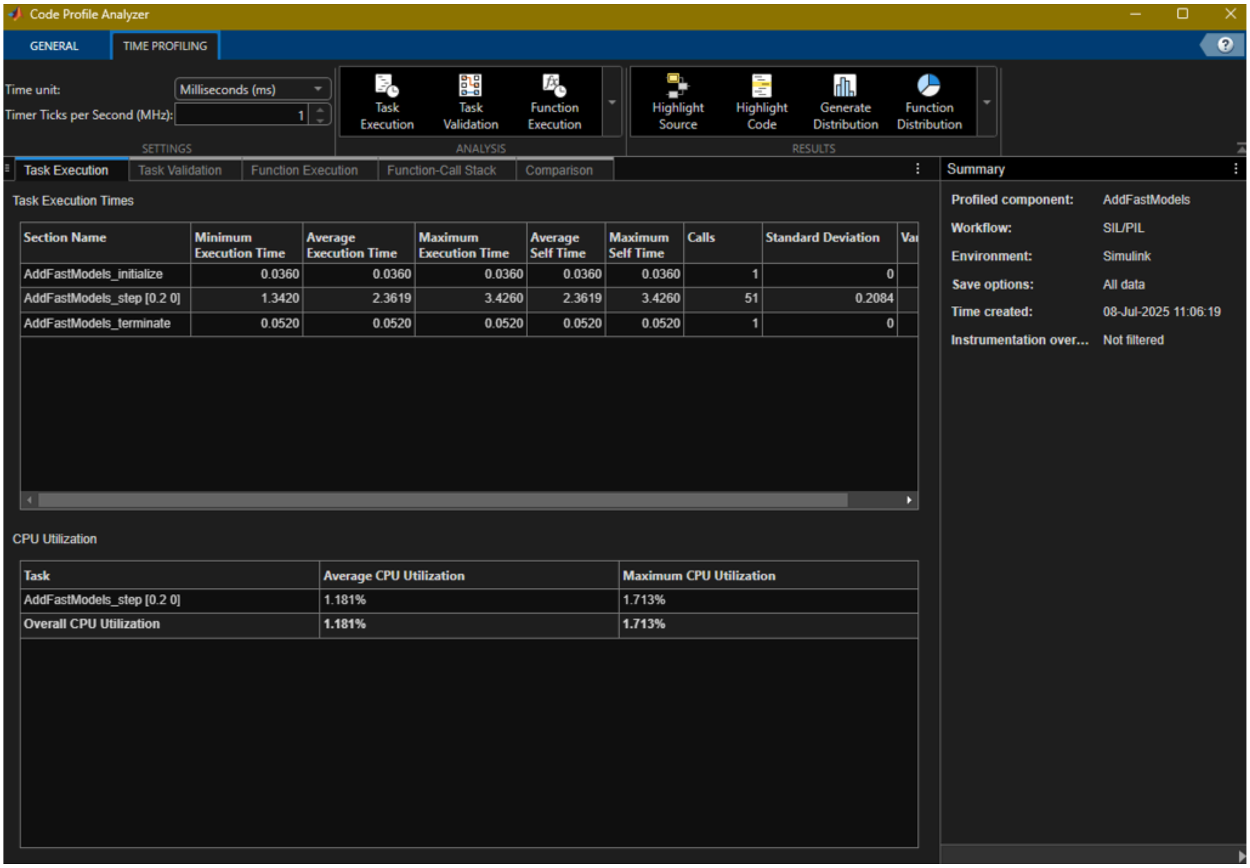
Interrupt Frequency (MHz): 1

CPU Clock (MHz): 1000

M4 Fast Model Defaults

Benchmarking is done with default values - (Interrupt Frequency - 1, CPU Clock - 1000 MHz, Quantum - 10000).

Timings of M4 Fast Model - `coder.profile.show(outM4FM.executionProfile)` , Timer Ticks Per Second(MHz) is different from that of QEMU.



Time unit: Milliseconds (ms)

Timer Ticks per Second (MHz): 1

Task Execution Times

Section Name	Minimum Execution Time	Average Execution Time	Maximum Execution Time	Average Self Time	Maximum Self Time	Calls	Standard Deviation	Var
AddFastModels_initialize	0.0360	0.0360	0.0360	0.0360	0.0360	1	0	
AddFastModels_step [0.2 0]	1.3420	2.3619	3.4260	2.3619	3.4260	51	0.2084	
AddFastModels_terminate	0.0520	0.0520	0.0520	0.0520	0.0520	1	0	

CPU Utilization

Task	Average CPU Utilization	Maximum CPU Utilization
AddFastModels_step [0.2 0]	1.181%	1.713%
Overall CPU Utilization	**1.181%**	**1.713%**

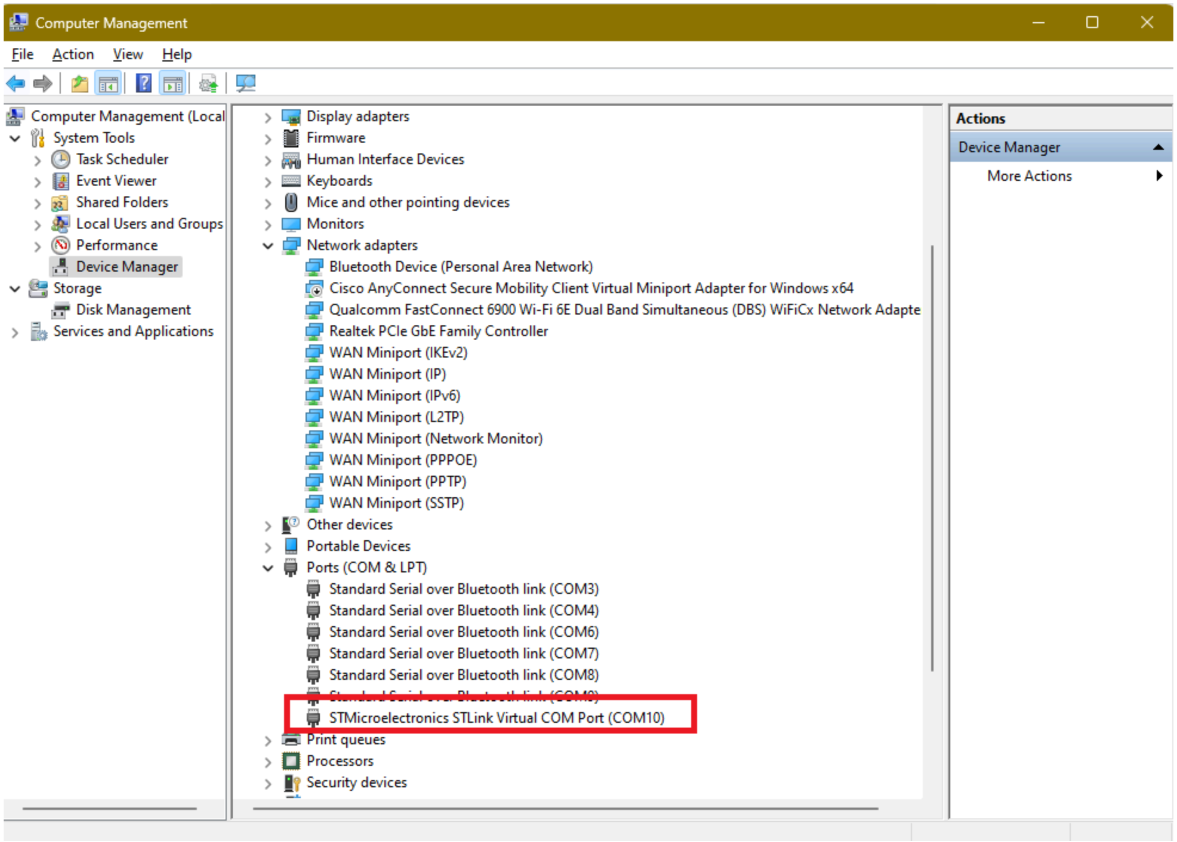
Profiled component:	AddFastModels
Workflow:	SIL/PIL
Environment:	Simulink
Save options:	All data
Time created:	08-Jul-2025 11:06:19
Instrumentation over...	Not filtered

M4 fast Model Benchmarking

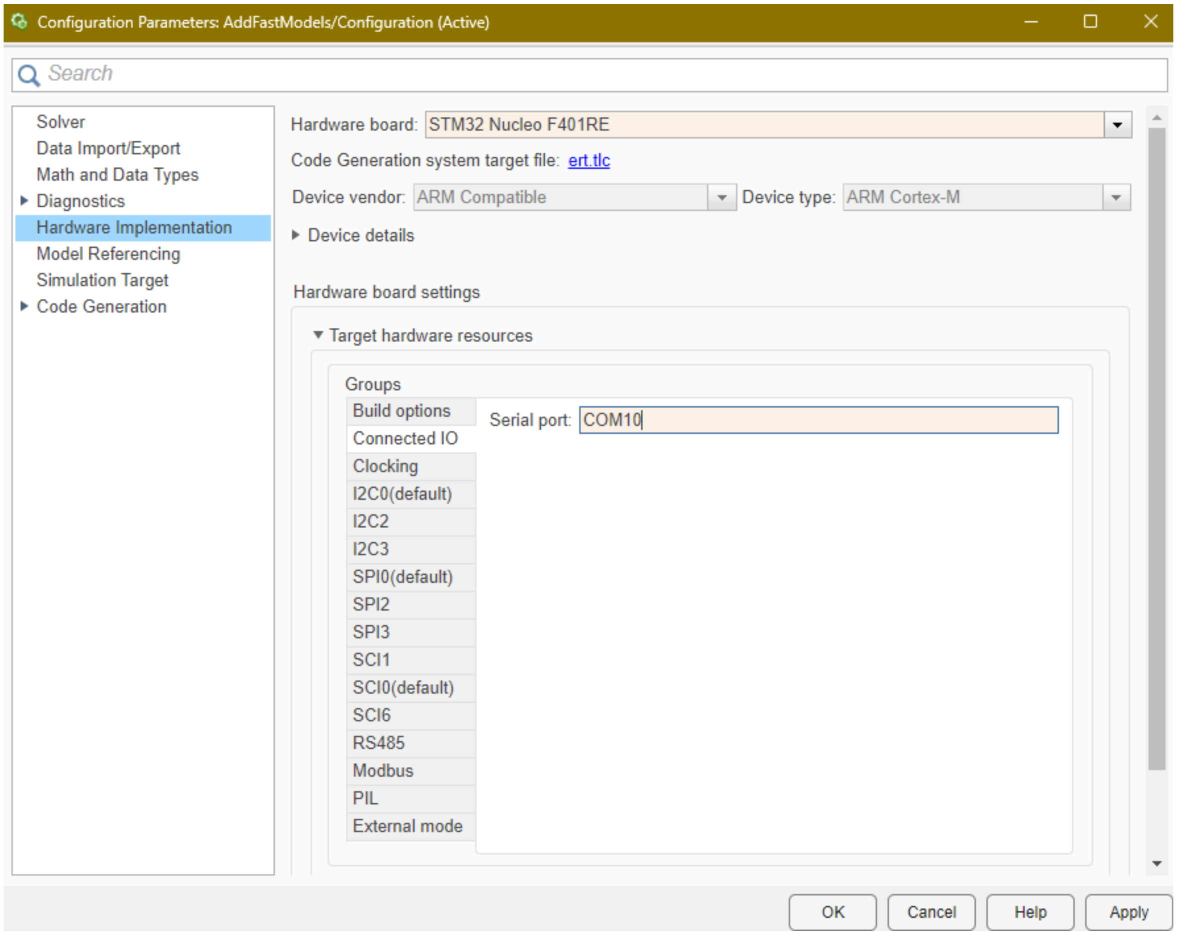
Hardware Benchmarking

- Set COM Port based on hardware COM port-

Configure COM Port


Computer Management
File Action View Help
Computer Management (Local
System Tools
Task Scheduler
Event Viewer
Shared Folders
Local Users and Groups
Performance
Device Manager
Storage
Disk Management
Services and Applications
Display adapters
Firmware
Human Interface Devices
Keyboards
Mice and other pointing devices
Monitors
Network adapters
Bluetooth Device (Personal Area Network)
Cisco AnyConnect Secure Mobility Client Virtual Miniport Adapter for Windows x64
Qualcomm FastConnect 6900 Wi-Fi 6E Dual Band Simultaneous (DBS) WiFiCx Network Adapte
Realtek PCIe GbE Family Controller
WAN Miniport (IKEv2)
WAN Miniport (IP)
WAN Miniport (IPv6)
WAN Miniport (L2TP)
WAN Miniport (Network Monitor)
WAN Miniport (PPPOE)
WAN Miniport (PPTP)
WAN Miniport (SSTP)
Other devices
Portable Devices
Ports (COM & LPT)
Standard Serial over Bluetooth link (COM3)
Standard Serial over Bluetooth link (COM4)
Standard Serial over Bluetooth link (COM6)
Standard Serial over Bluetooth link (COM7)
Standard Serial over Bluetooth link (COM8)
Standard Serial over Bluetooth link (COM9)
STMicroelectronics STLink Virtual COM Port (COM10)
Print queues
Processors
Security devices
Actions
Device Manager
More Actions

STM COM port



Configuration Parameters: AddFastModels/Configuration (Active)

Hardware board: STM32 Nucleo F401RE

Code Generation system target file: ert.tlc

Device vendor: ARM Compatible

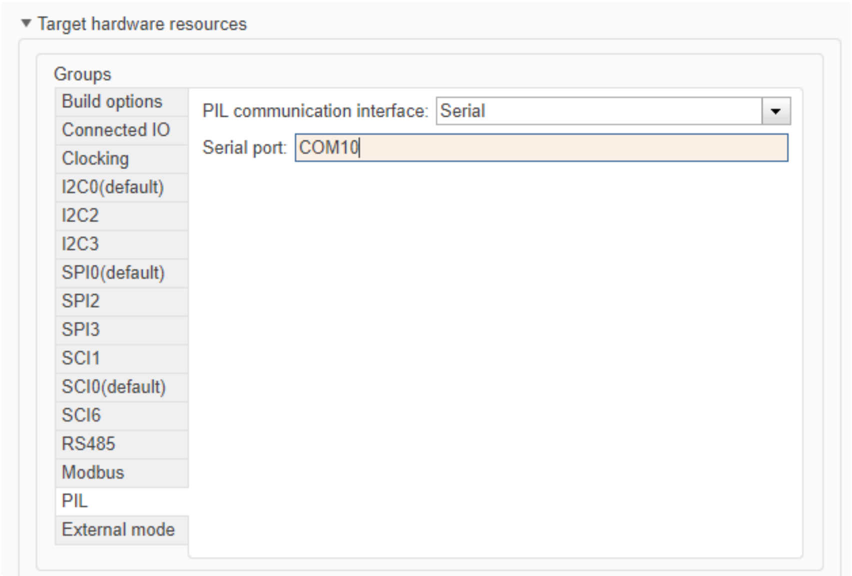
Device type: ARM Cortex-M

Hardware board settings

Target hardware resources

Serial port: COM10

Connected IO COM Port

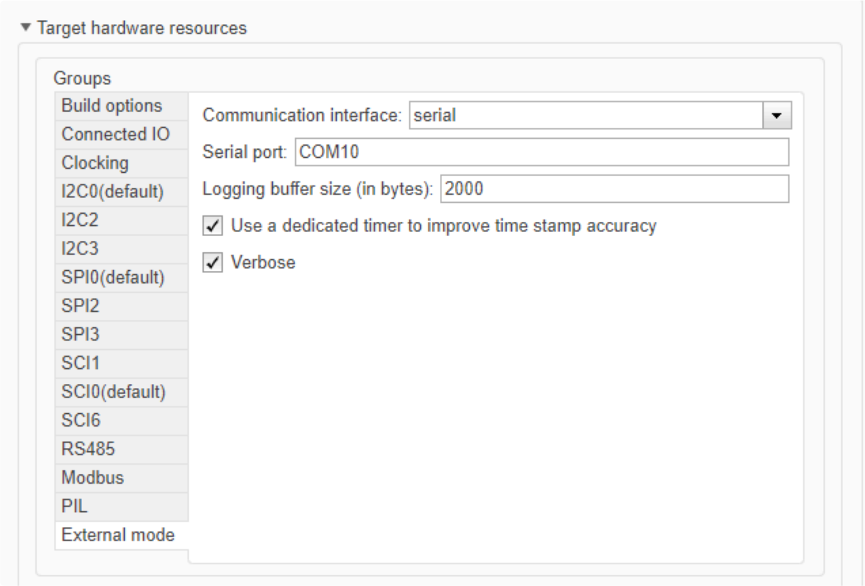


Target hardware resources

PIL communication interface: Serial

Serial port: COM10

PIL Serial COM Port



Communication interface: serial

Serial port: COM10

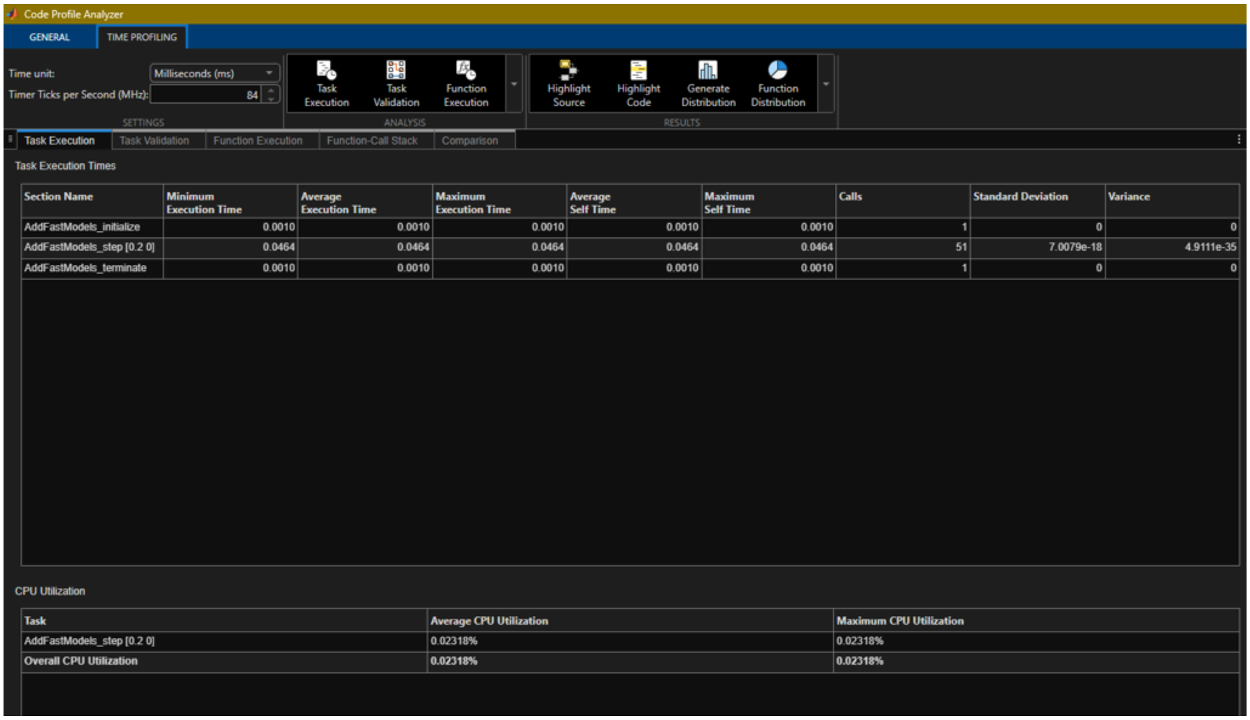
Logging buffer size (in bytes): 2000

☑ Use a dedicated timer to improve time stamp accuracy

☑ Verbose

External Mode COM Port

- Timings of M4 Fast Model - `coder.profile.show(outM4Hardware.executionProfile)`, Clock is 84MHz.



Time unit: Milliseconds (ms)

Timer Ticks per Second (MHz): 84

Task Execution Times

Section Name	Minimum Execution Time	Average Execution Time	Maximum Execution Time	Average Self Time	Maximum Self Time	Calls	Standard Deviation	Variance
AddFastModels_initialize	0.0010	0.0010	0.0010	0.0010	0.0010	1	0	0
AddFastModels_step [0.2 0]	0.0464	0.0464	0.0464	0.0464	0.0464	51	7.0079e-18	4.9111e-35
AddFastModels_terminate	0.0010	0.0010	0.0010	0.0010	0.0010	1	0	0

CPU Utilization

Task	Average CPU Utilization	Maximum CPU Utilization
AddFastModels_step [0.2 0]	0.02318%	0.02318%
Overall CPU Utilization	0.02318%	0.02318%

- Hardware Benchmarking