

MATLAB EXPO

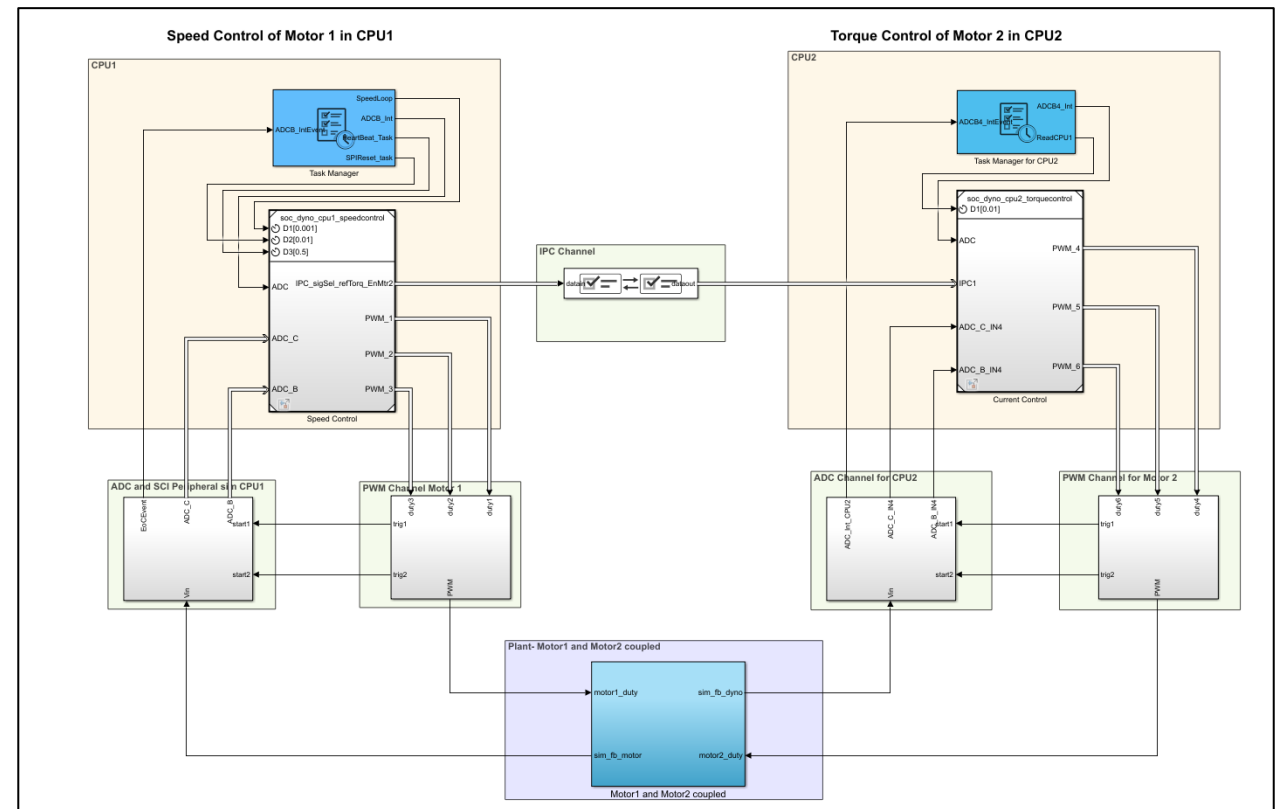
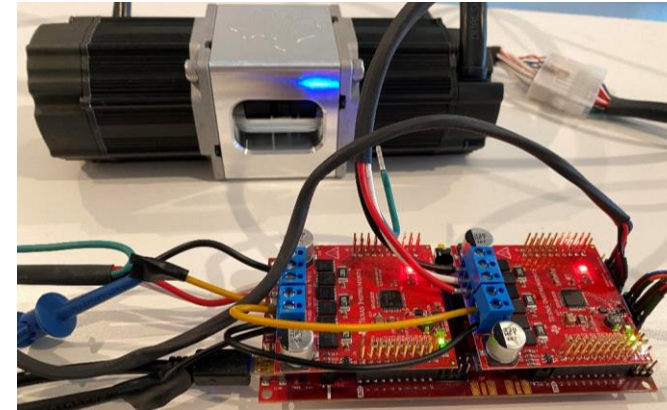
TI C2000™ 듀얼코어 마이크로 컨트롤러에
모터제어 알고리즘을 배포하기 위한 솔루션

정승혁 차장, 매스웍스코리아



Key Takeaways

- Simulate sensorless Field-Oriented Control (FOC) on a dyno setup
- Complete Model-Based Design workflow for multicore microcontroller
 - Hardware component and device driver behavior simulation
 - Enhanced on-device profiling



Dual CPUs PMSM Dyno Testing

al_host_model* - Simulink

DEBUB MODELING FORMAT APPS SWITCH

Library Browser Log Signals Add Viewer Signal Table

Stop Time inf Normal Fast Restart Step Back Pause Step Forward Stop Data Inspector Logic Analyzer

msm_foc_dual_host_model

pc_dual_host_model ▶

PMSM Dual CPU Dyno Control Host

Inputs

SpeedRef.Motor1 Iq_ref.Motor2

Motor1: Current (Ia) Duty Cycle Speed Position Torque Power

Motor2: Current (Ia) Duty Cycle Speed Position Torque Power

Serial Communication: Scope (Per-Unit), Debug1 (SI units), Debug2 (SI units)

HOST Serial Setup

300 Motor 1 - Reference Speed (RPM)

0 Motor 2 - Iq Ref (A)

Stop Start Motor1

Scope (Per-Unit) [600x2] [600x2] Scope

Debug1 (SI units) single [0]

Debug2 (SI units) single [-1.936]

Copyright 2021 The MathWorks, Inc.

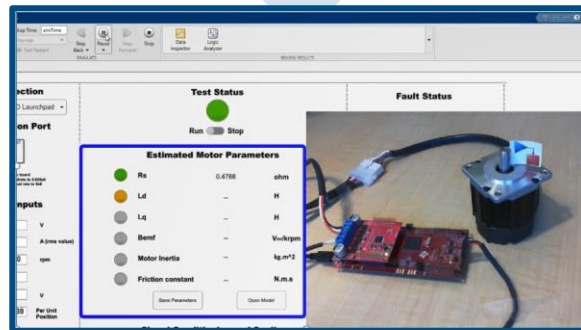
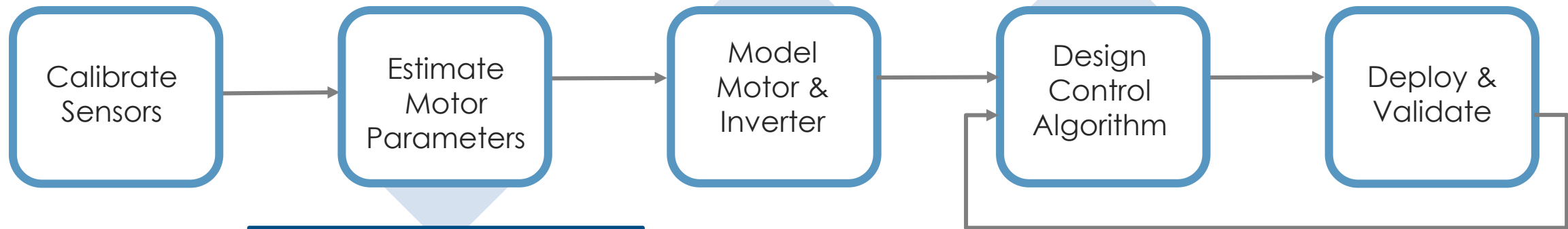
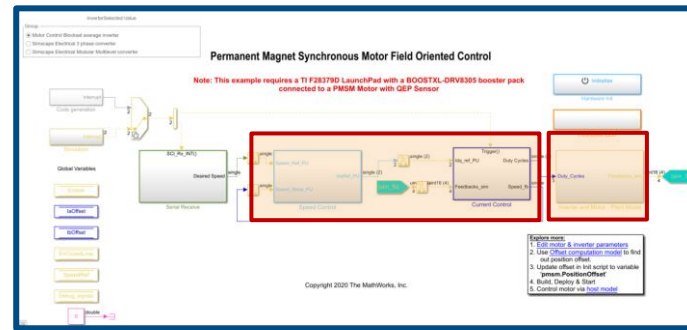
100% T=68.550 FixedS View diagnostics

Scope: Serial Communication/1:1, Serial Communication/1:2

TENNA digital display: 0.056, 24.00

Workflow for Implementing Field-Oriented Control

- Motor Control Blockset



```

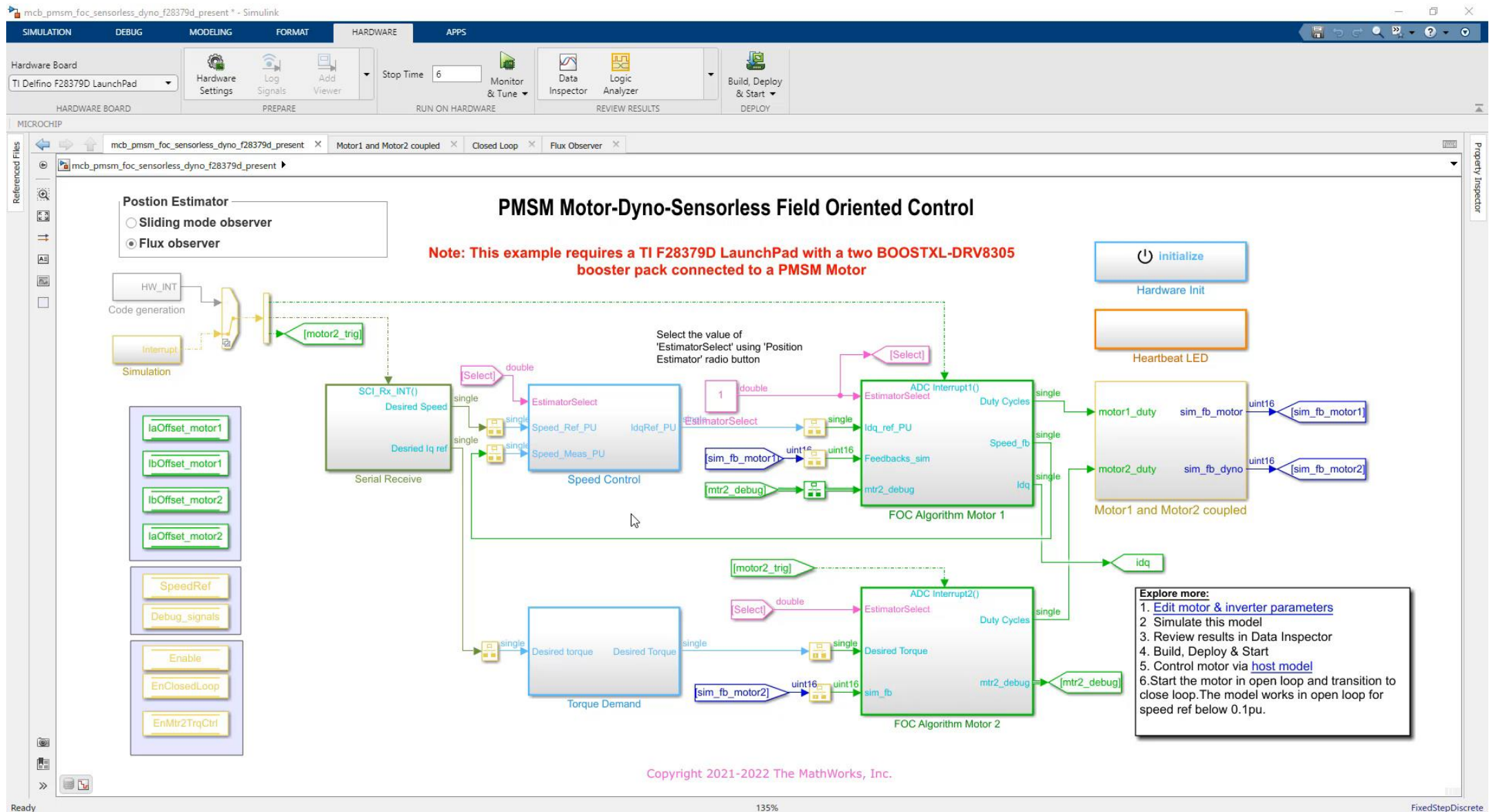
/* Product: '<S26>/Product1' */
mcb_pmsm_foc_hall_f28379d_B.Product1_j =
mcb_pmsm_foc_hall_f28379d_B.Satu
mcb_pmsm_foc_hall_f28379d_B.Satu

/* Sum: '<S26>/Sum2' */
mcb_pmsm_foc_hall_f28379d_B.Sum2_i
+ mcb_pmsm_foc_hall_f28379d_B.Pr

/* Sqrt: '<S26>/Sqrt' */
mcb_pmsm_foc_hall_f28379d_B.Sqrt =
(mcb_pmsm_foc_hall_f28379d_B.Sum

```

Demo – PMSMs Dyno Model in FOC Sensorless Control



Challenges of Deployment on the Embedded Systems..

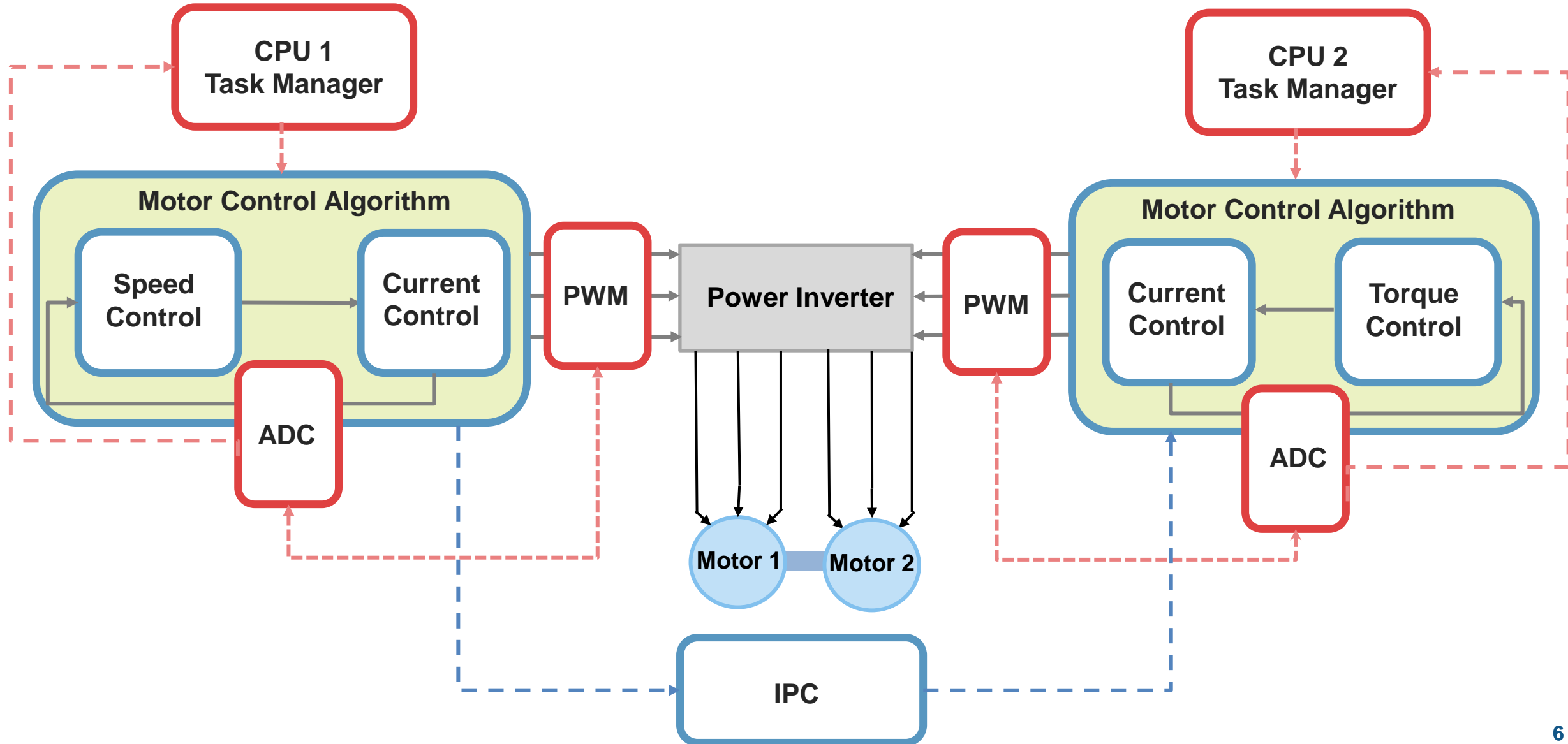
Model

- System requirement
 - TI C2000 dual-core processors
- Controller sample rate is 20kHz
 - Field-oriented control (FOC)
 - Sensorless control
 - Dyno setup (2 motors)
- No sensor delays in my model
- ADC-PWM synchronization

Multicore Processor

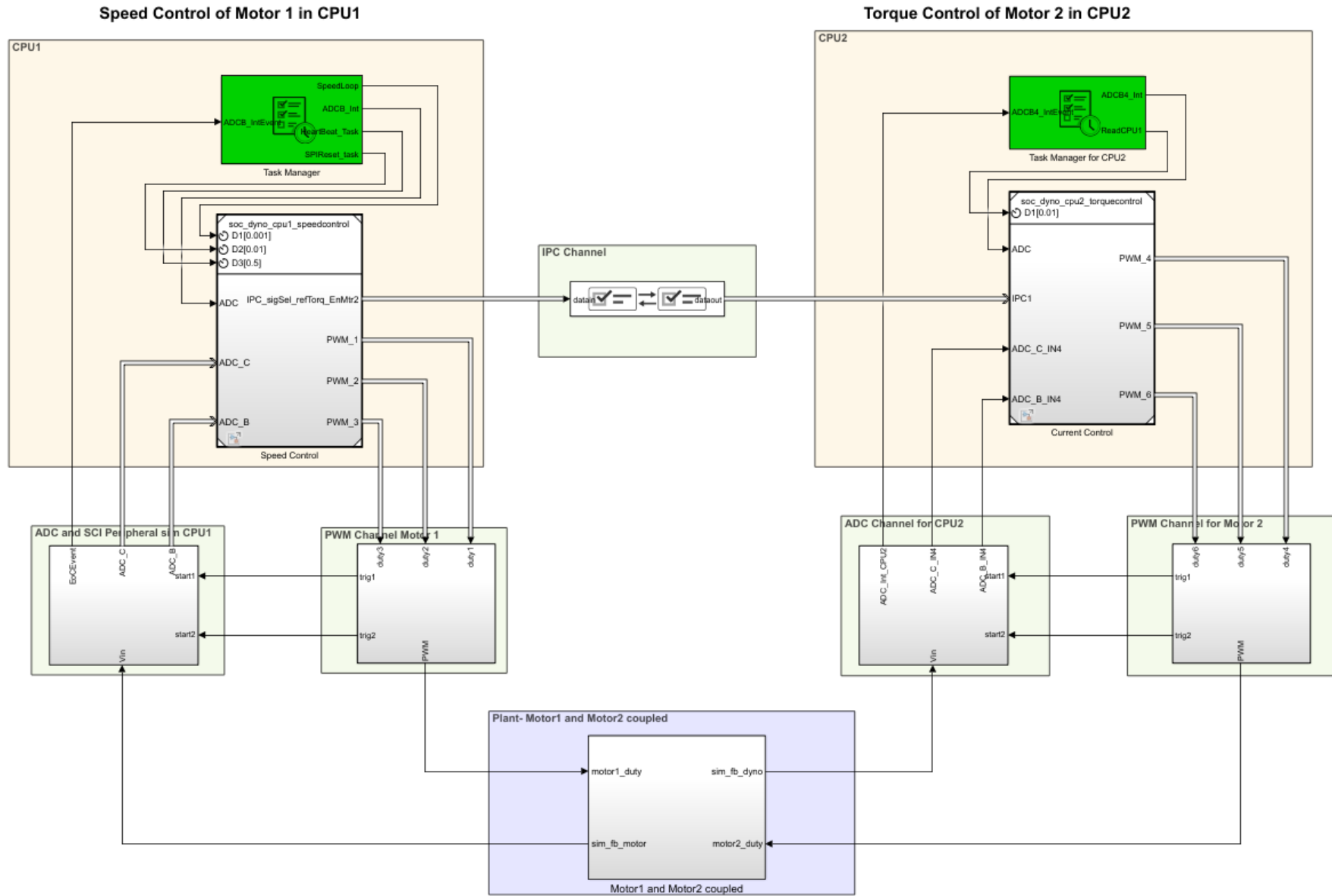
- How to implement and partition controls into two separated cores?
- How to communicate between CPU1 and CPU2?
- How to make sure task execution meets software requirement?

Simulate Motor Control System with Peripherals and Task Execution

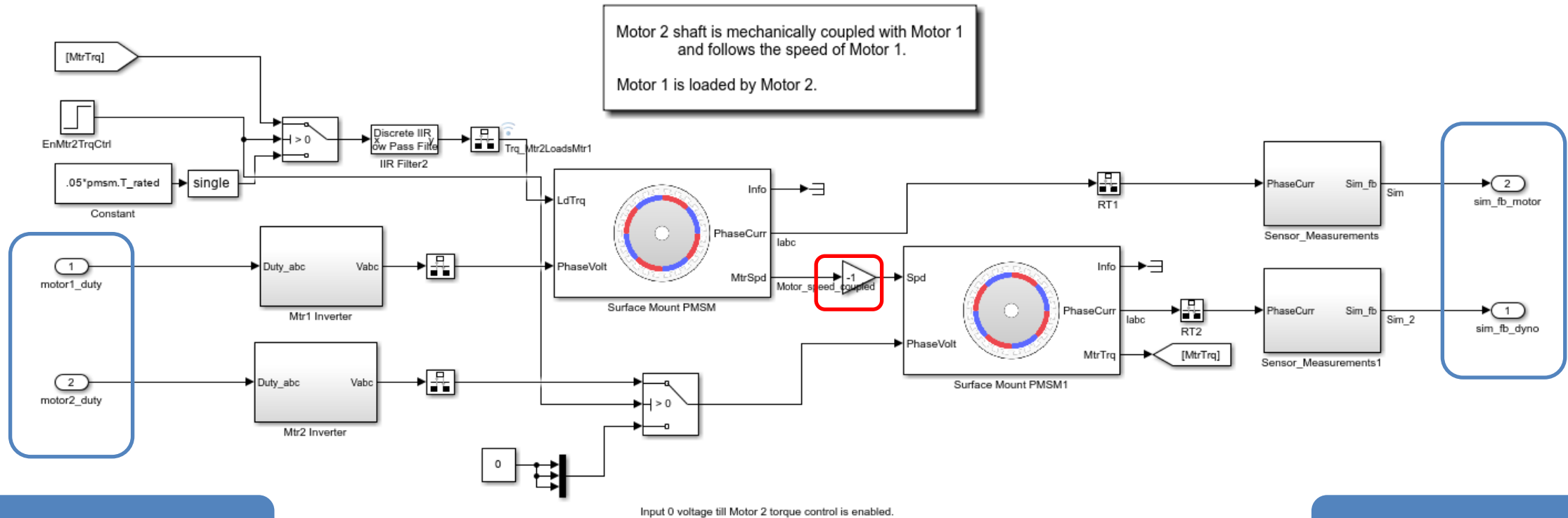


Model Multicore application Using SoC Blockset

Task Manager
IPC
PWM
ADC



Plant Subsystem

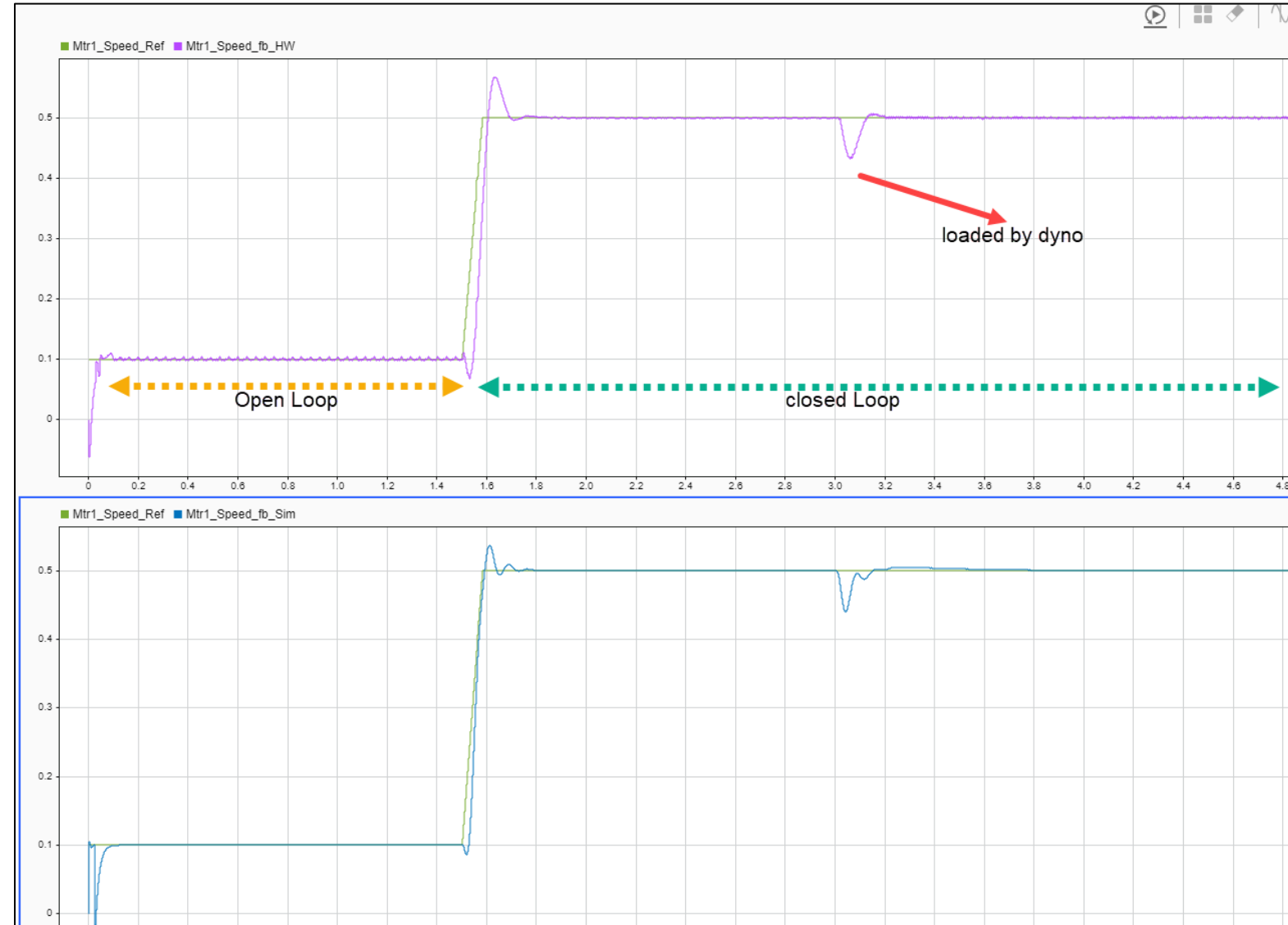


PWM Input

ADC Output

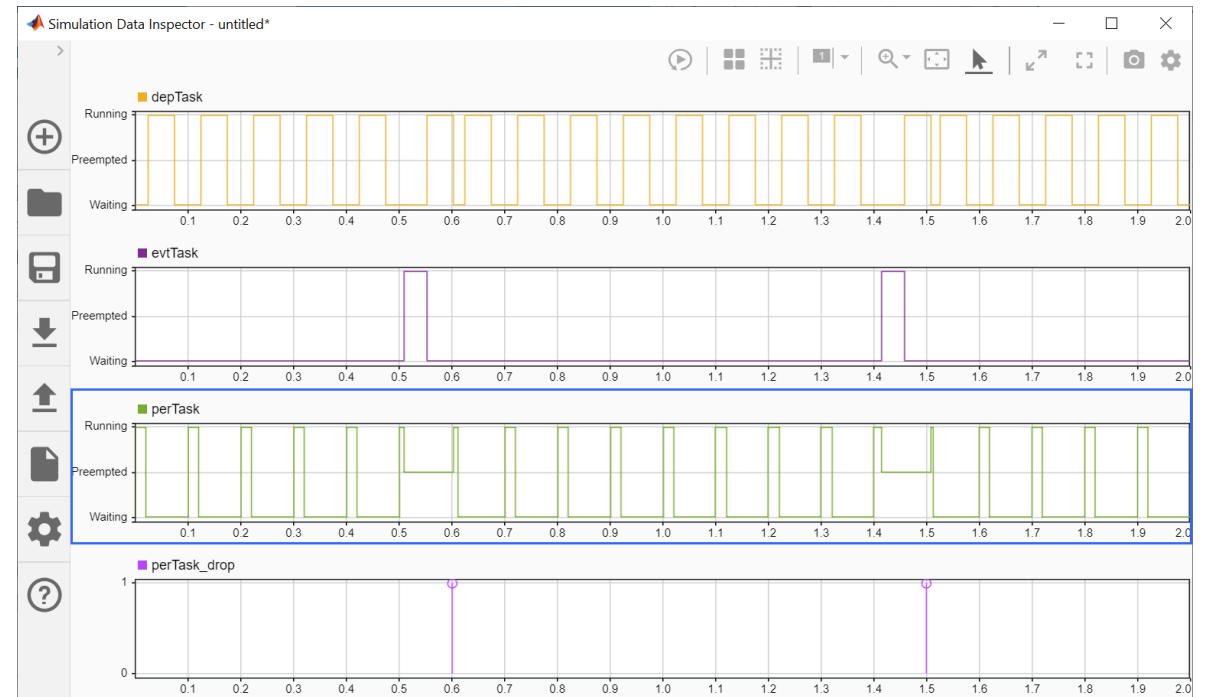
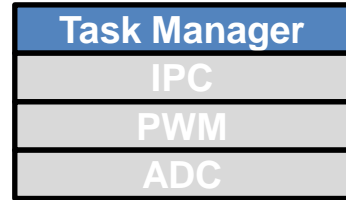
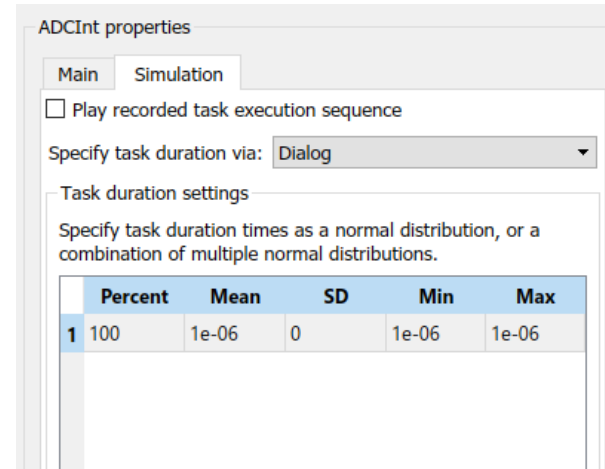
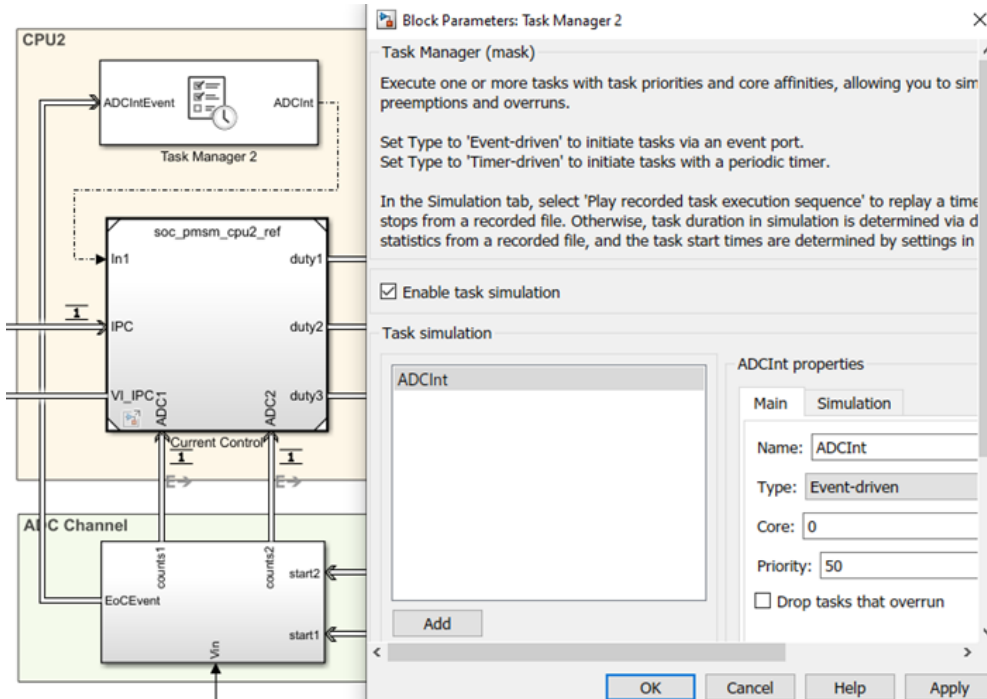
Hardware vs Simulation Analysis

Captured HW
signal using
External mode



Simulation
Result

Task Manager



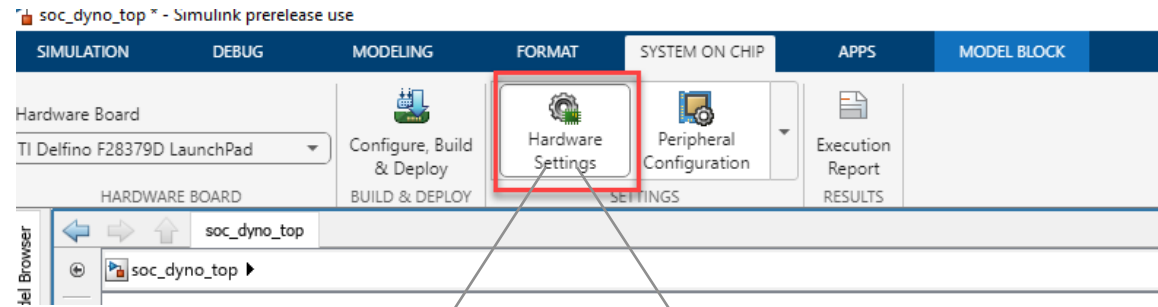
- Model/Simulate
 - Periodic/async tasks
 - Task priorities
 - Latencies
 - Duration as normal distribution

Model Configuration for Multicore Architecture

Top Model

CPU #1 (c28xCPU1)

CPU #2 (c28xCPU2)



Top Model

- Solver
- Data Import/Export
- Math and Data Types
- ▶ Diagnostics
- ▶ Hardware Implementation
- Model Referencing
- Simulation Target
- ▶ Code Generation
- Coverage
- ▶ HDL Code Generation
- Simscape
- ▶ Simscape Multibody

Hardware board: TI Delfino F28379D LaunchPad

Code Generation system target file: [ert.tlc](#)

Device vendor: Texas Instruments Device type: C2000

▶ Device details

Feature set for selected hardware board:

Simulink or Embedded Coder Hardware Support Package

SoC Blockset

Hardware board settings

Processing Unit: None

▶ Design mapping

▶ Task profiling in simulation

▶ Task and memory simulation

Reference Model

- Solver
- Data Import/Export
- Math and Data Types
- ▶ Diagnostics
- ▶ Hardware Implementation
- Model Referencing
- Simulation Target
- ▶ Code Generation
- Coverage
- ▶ HDL Code Generation

Hardware board: TI Delfino F28379D LaunchPad

Code Generation system target file: [ert.tlc](#)

Device vendor: Texas Instruments Device type: C2000

▶ Device details

Feature set for selected hardware board:

Simulink or Embedded Coder Hardware Support Package

SoC Blockset

Hardware board settings

Processing Unit: c28xCPU1

▶ Design mapping

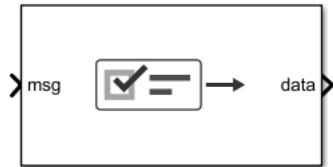
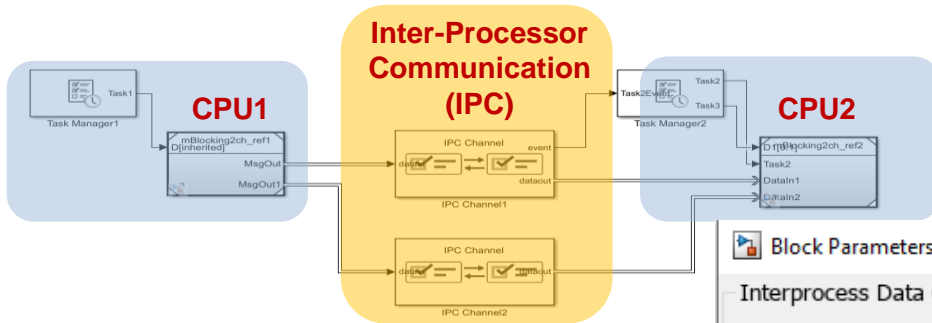
▶ Task profiling on processor

▶ Target hardware resources

Inter-Processor Communication with IPC Blocks

Task Manager
IPC
PWM
ADC

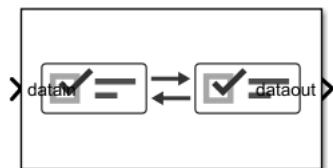
- Model the communication buffering and delay



Interprocess Data Read



Interprocess Data Write



Interprocess Data Channel

Block Parameters: Interprocess Data Channel 1

Interprocess Data Channel
Model data channel between two processes.

Interprocess Data Write block writes the data to the channel from one process.

Interprocess Data Read block reads the data from the channel from another process.

Set Show event port to output a task trigger event when data is written to the channel.

Parameters

Main Statistics

Number of buffers: 1

Propagation delay: 2e-6

Show event port

Define buffer size and timing delay

OK Cancel Help Apply

Block Parameters: Interprocess Data Channel 1

Interprocess Data Channel
Model data channel between two processes.

Interprocess Data Write block writes the data to the channel from one process.

Interprocess Data Read block reads the data from the channel from another process.

Set Show event port to output a task trigger event when data is written to the channel.

Parameters

Main Statistics

Show number of used buffers

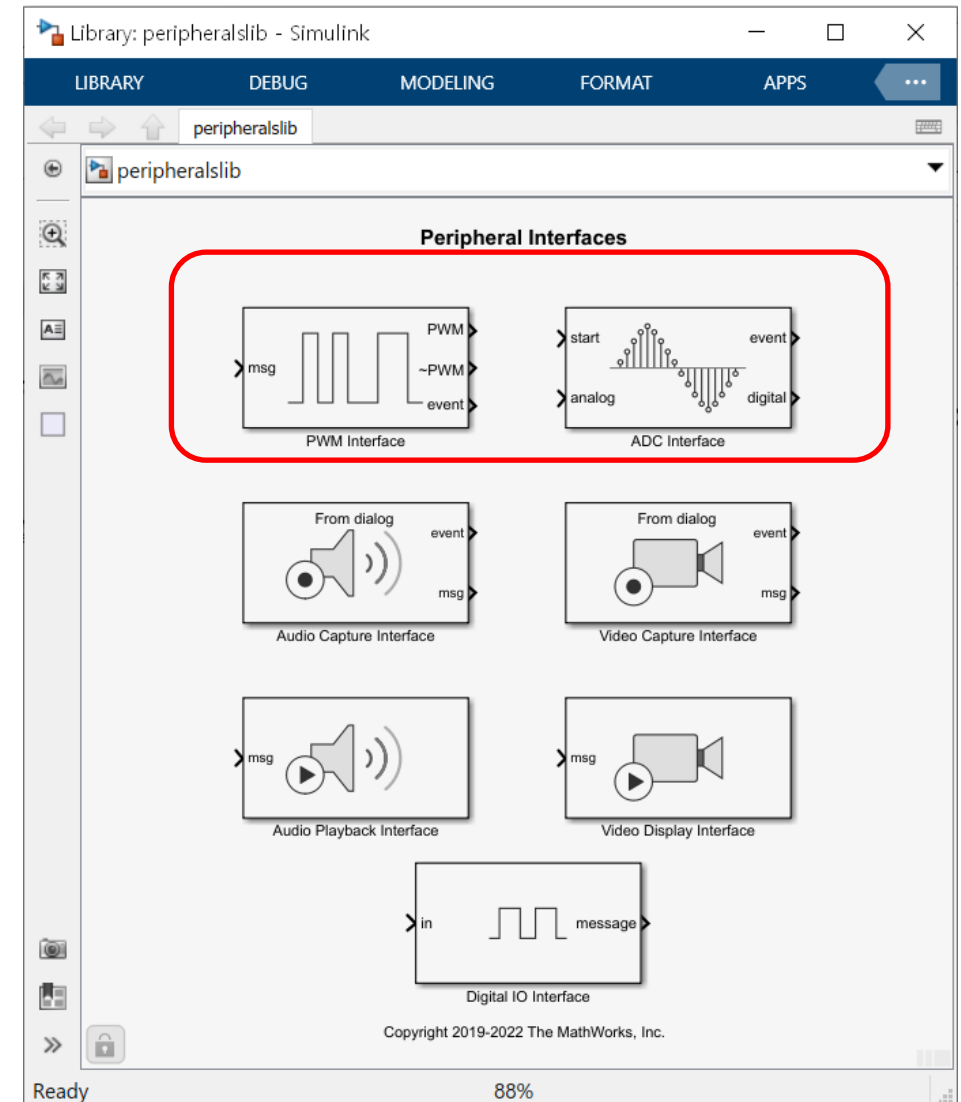
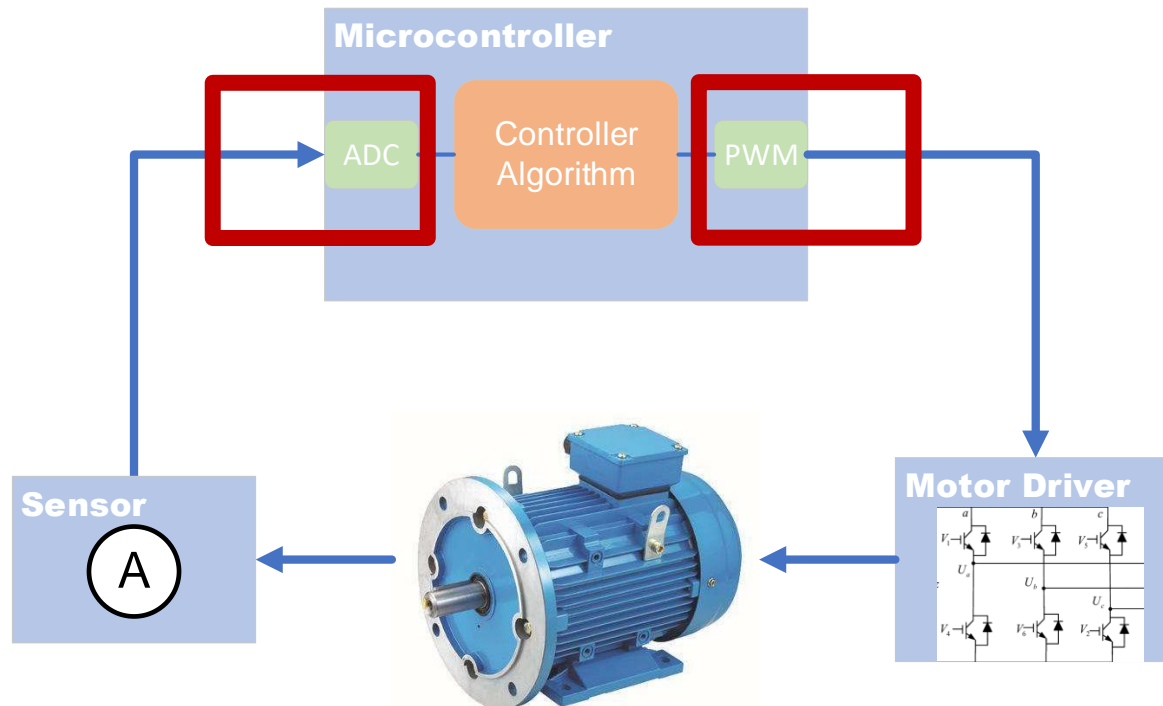
Show when buffer is overwritten

Visualize buffer consumption and overwrites

OK Cancel Help Apply

Motor control system modeling with peripheral interface

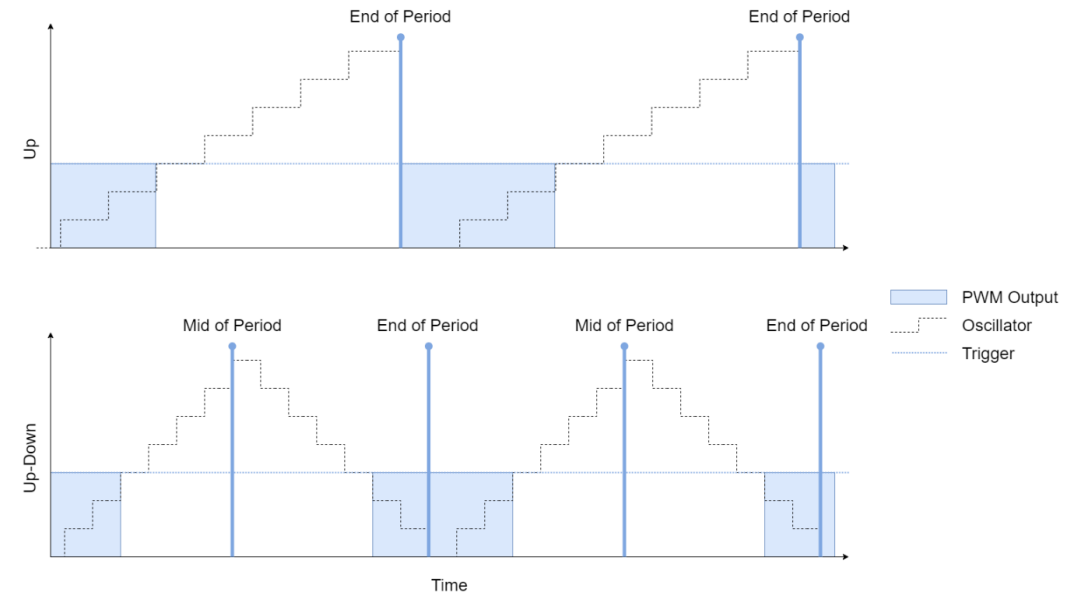
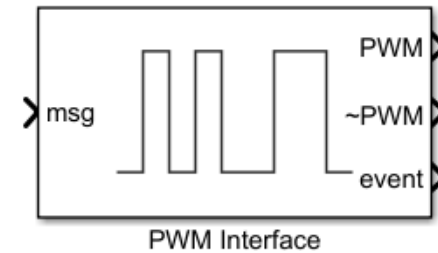
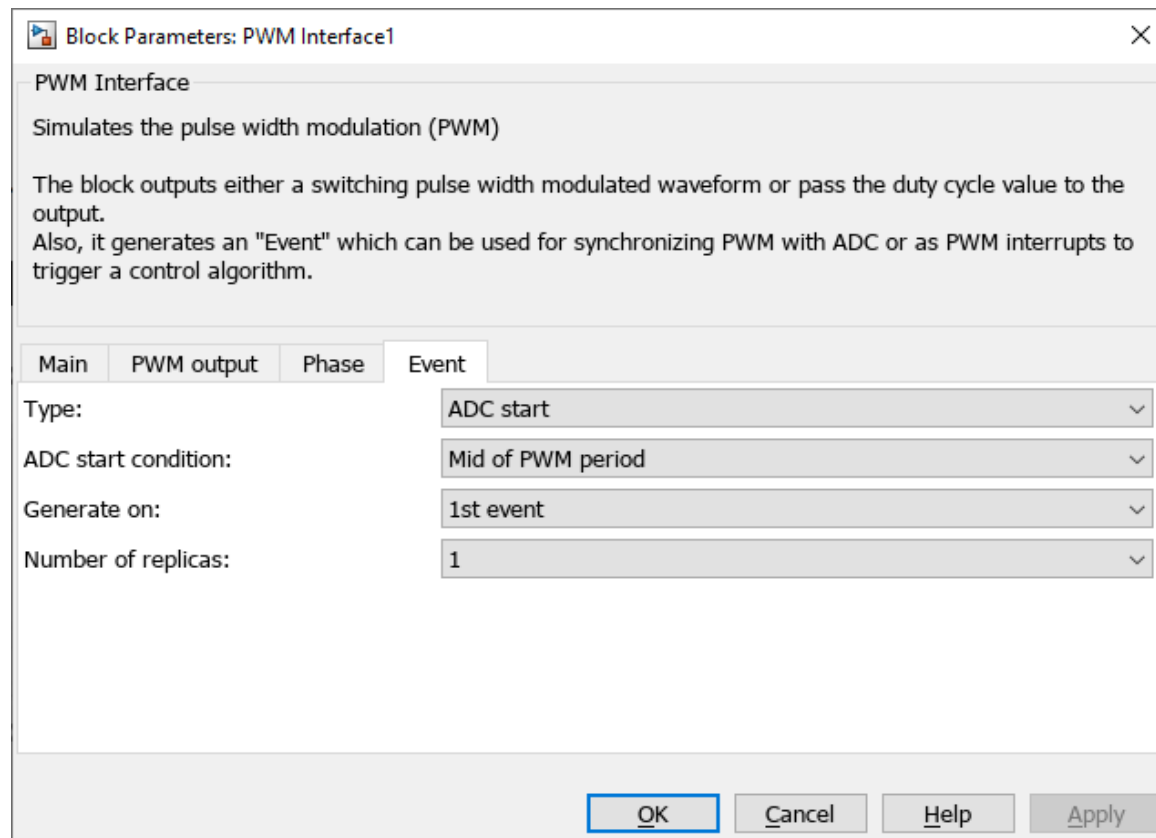
- Motor control peripheral interfaces



PWM Modeling

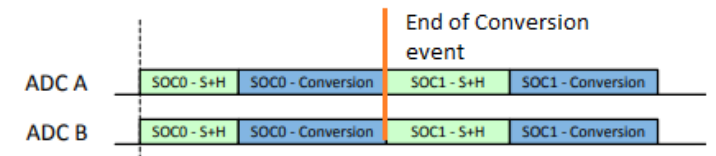
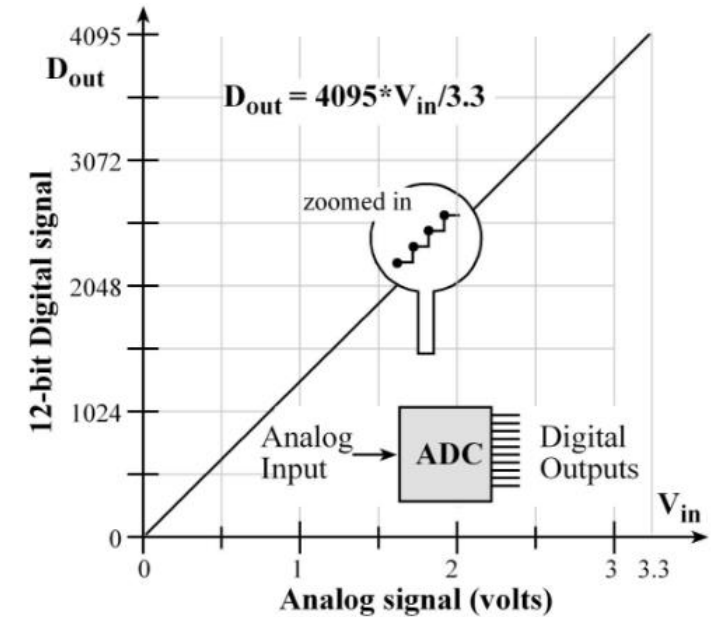
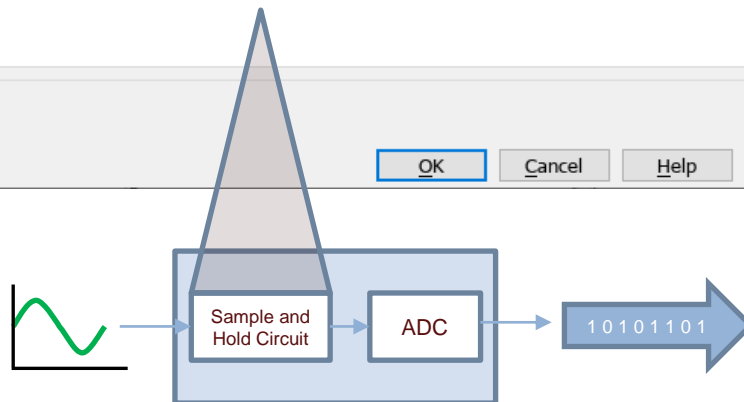
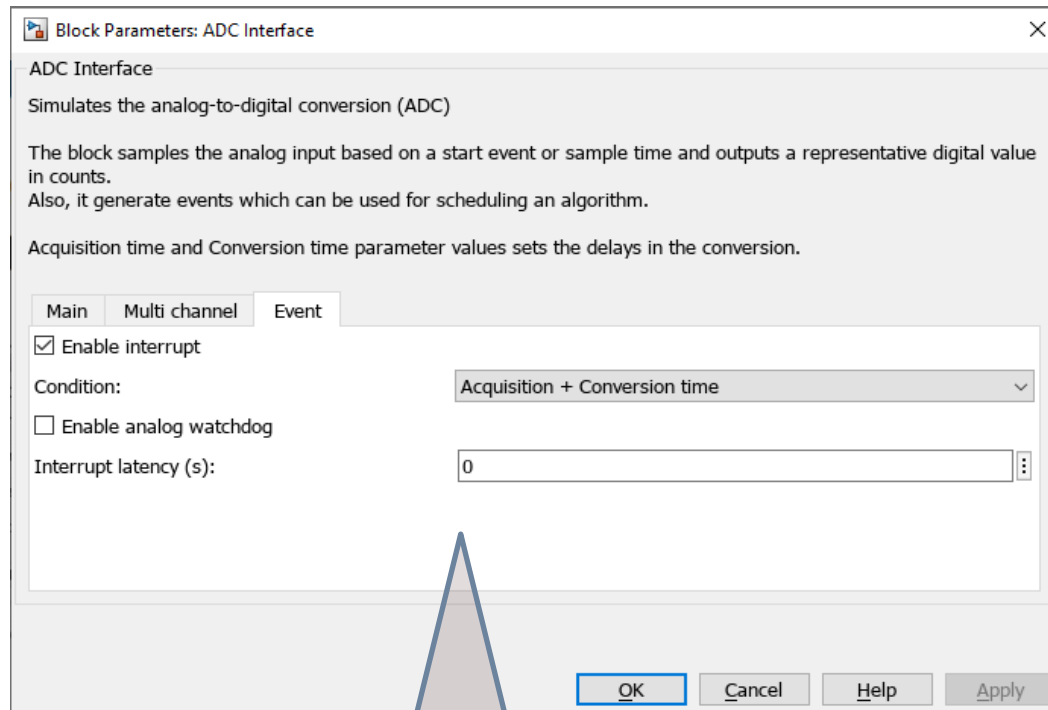
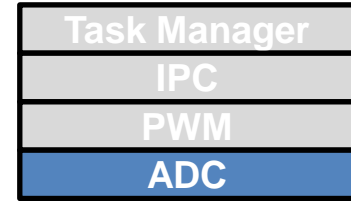
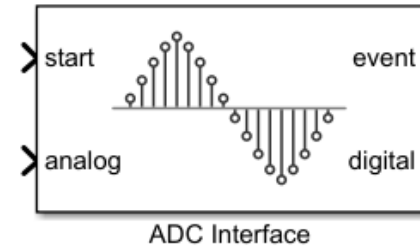
- PWM waveform simulation
- Event to synchronize with ADC or schedule a task

Task Manager
IPC
PWM
ADC

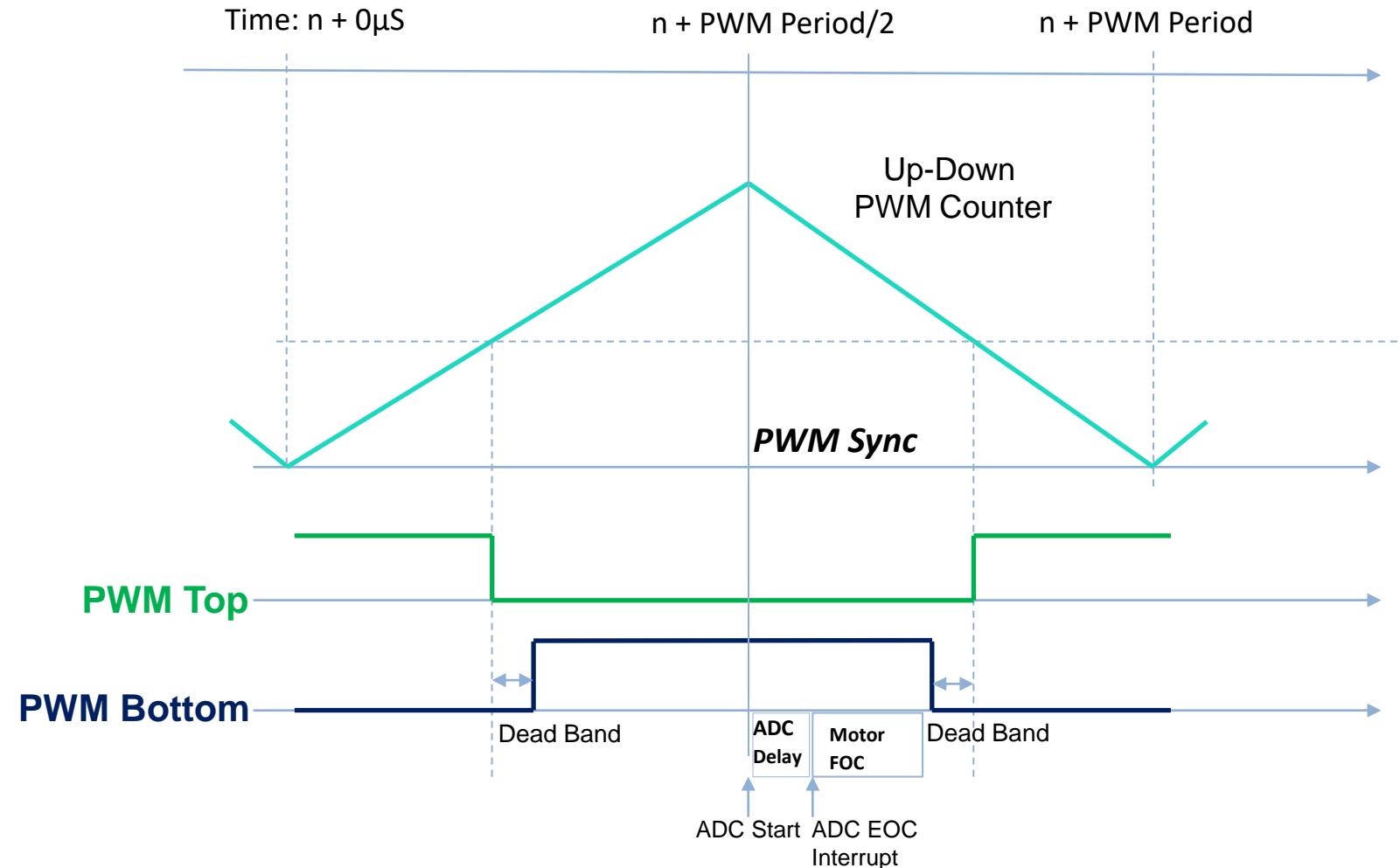
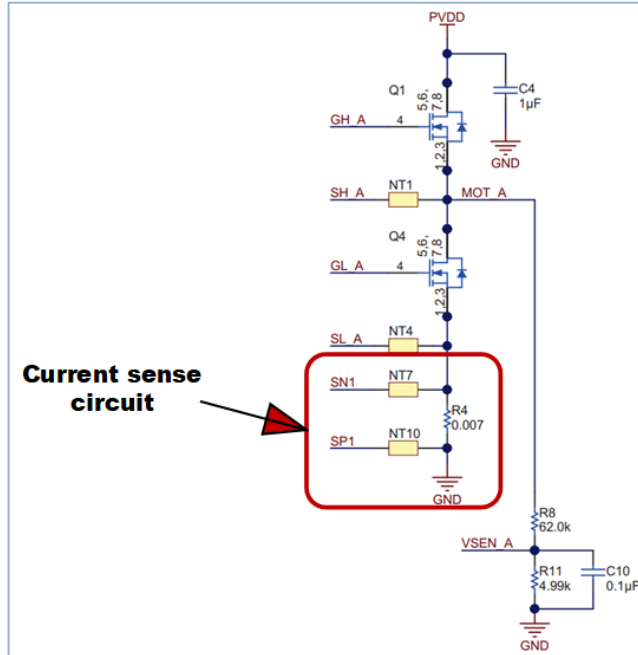


ADC Modeling

- Convert analog values to digital counts
- Model acquisition/conversion delays and trigger events

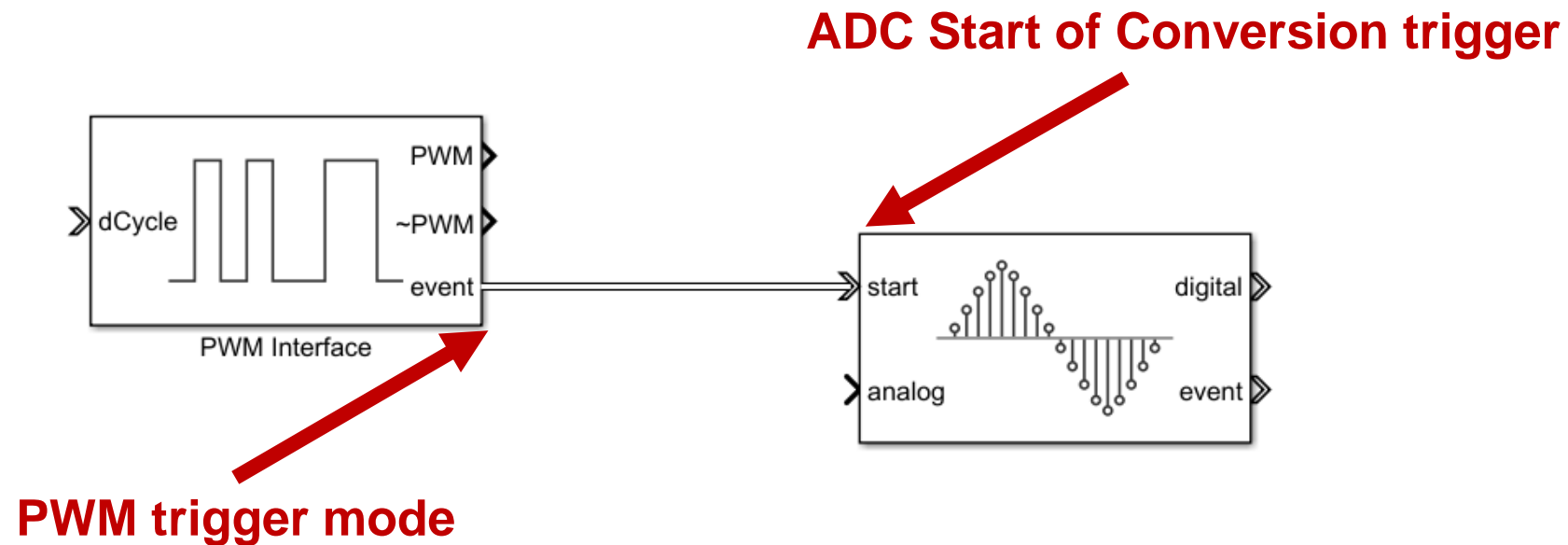


ADC – PWM Synchronization



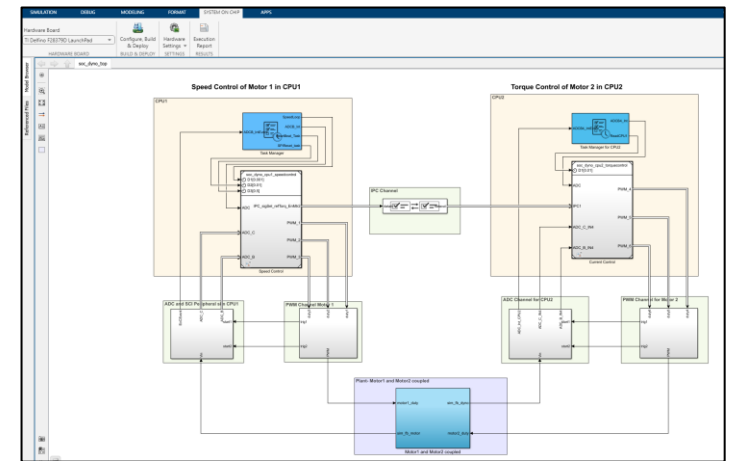
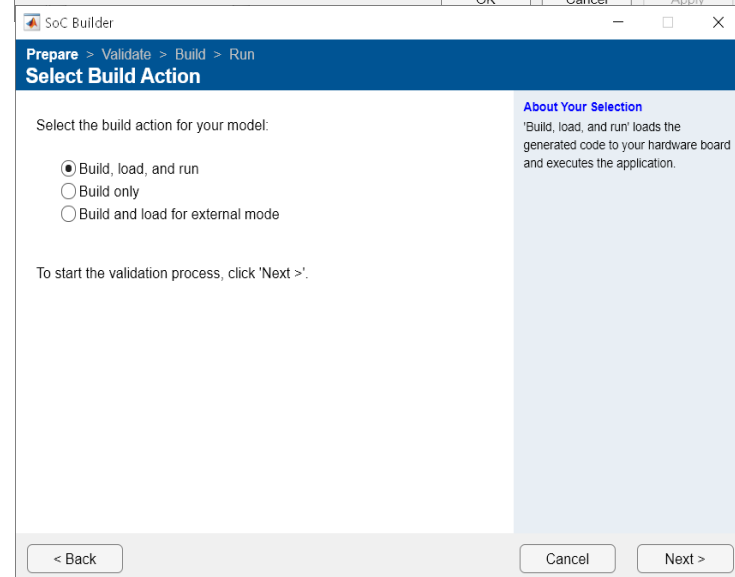
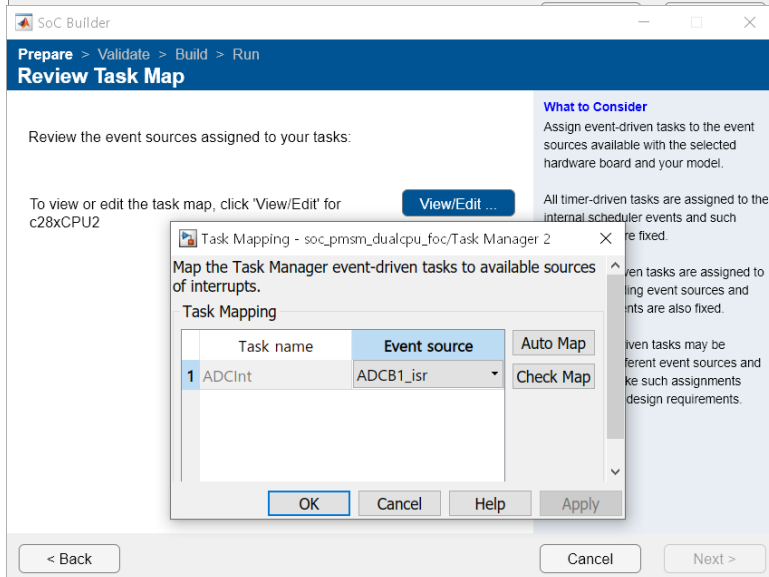
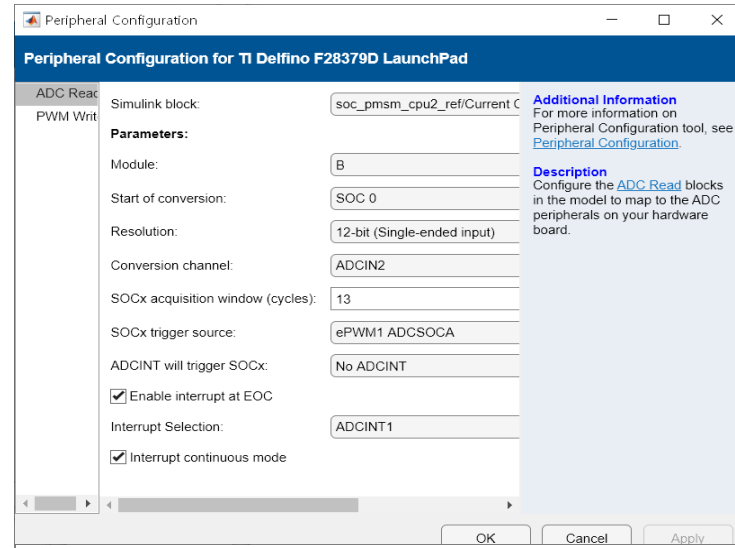
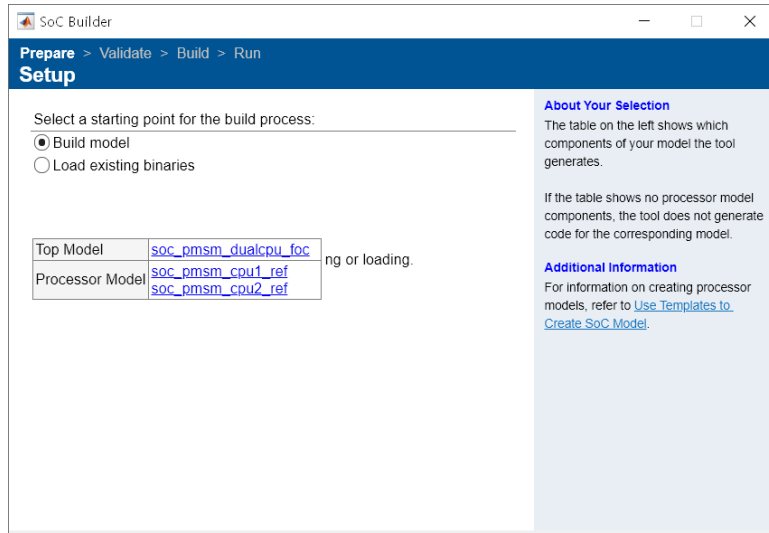
Enable ADC/PWM synchronization simulation

- To provide stable ADC input, the synchronization between ADC and PWM is required for close-loop motor control

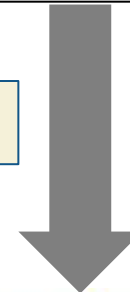


SoC Builder App

Build, load, and execute SoC model on SoC, FPGA, and MCU boards

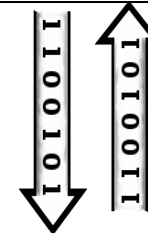
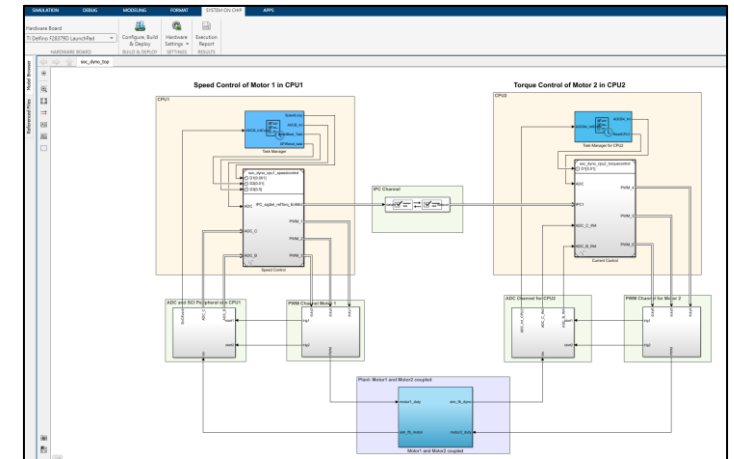
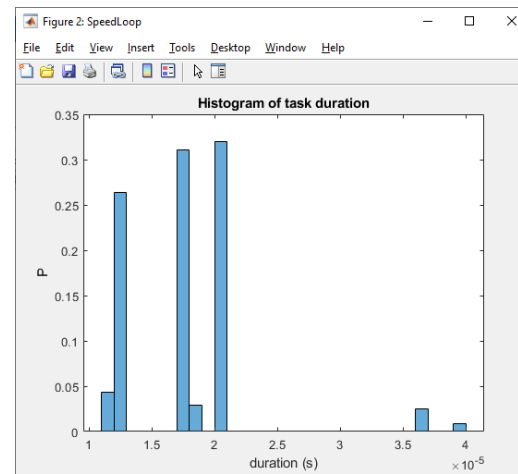
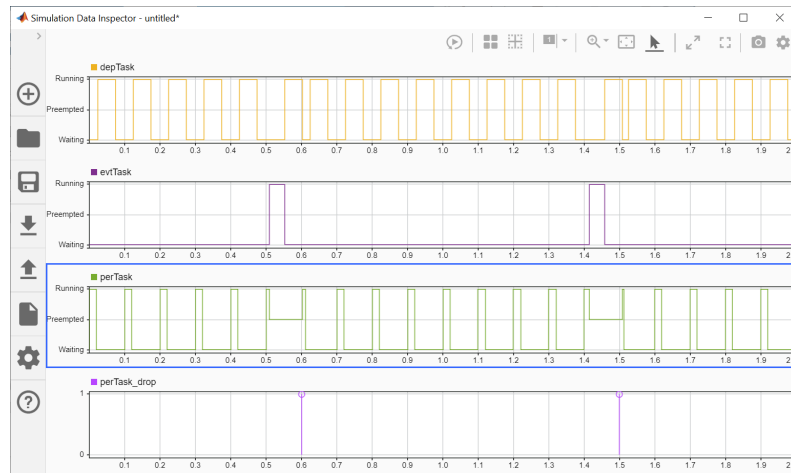


SoC Builder



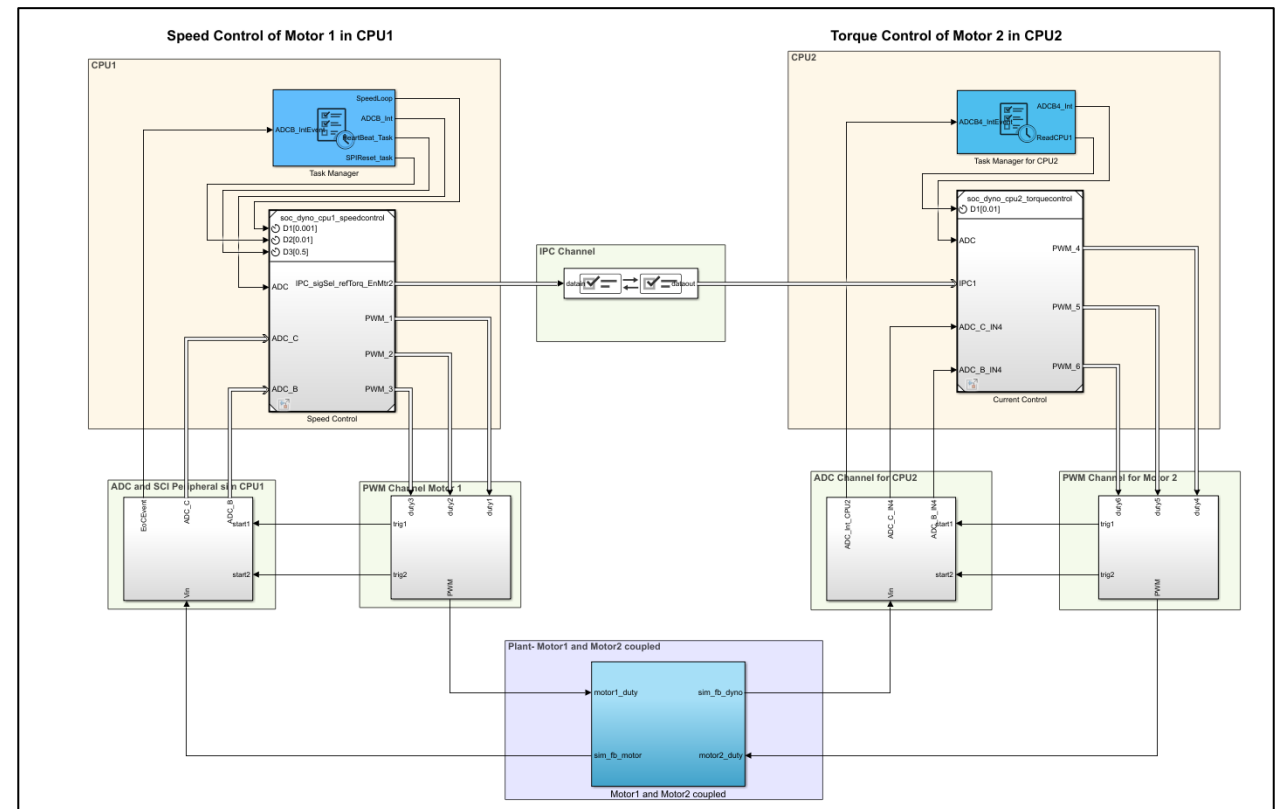
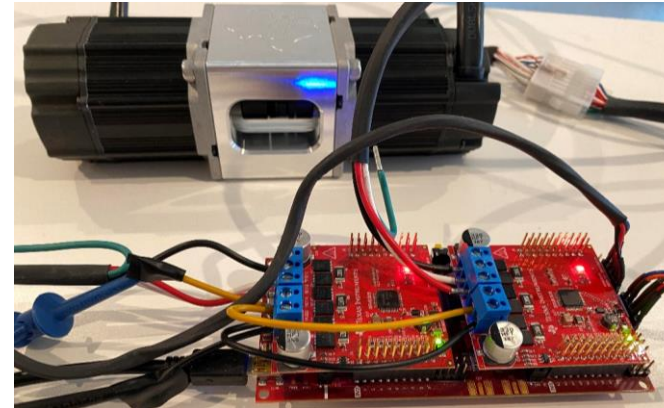
SoC Blockset Key Functionalities – On-Device Profiling

- Real-time performance profiling on hardware, including
 - Task execution
 - CPU utilization
 - Communication buffering and delay
 - Real-time SDI view
 - Analysis report



Wrap Up

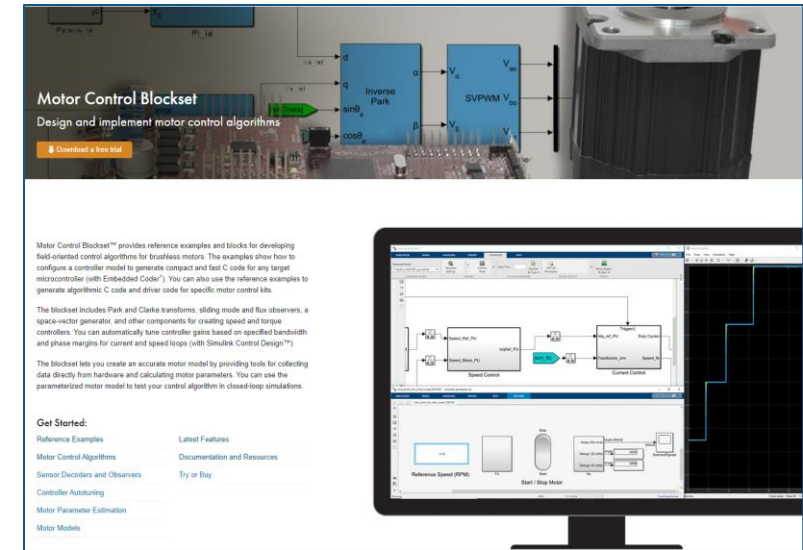
- Simulate sensorless field-oriented control (FOC) on a dyno setup
- Complete Model-Based Design workflow for multicore microcontroller
 - Hardware component and device driver behavior simulation
 - Enhanced on-device profiling



Learn More

- Recording webinars

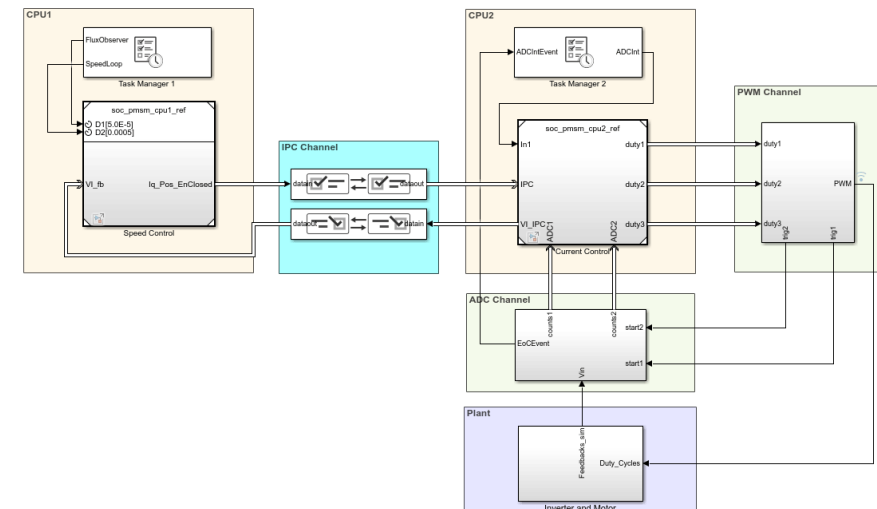
- [Field Oriented Control Made Easy](#)
- [Motor Control with TI Multicore MCUs Using Simulink](#)
- [Implementing Motor and Power Electronics Control on an FPGA-Based SoC](#)



- Shipping Demos

- [Partition Motor Control for Multiprocessor MCUs](#)
- [Control PMSM Loaded with Dual Motor \(Dyno\)](#)
- [Integrate MCU Scheduling and Peripherals in Motor Control Application](#)

Field-Oriented Control on Dual CPU Processor



Copyright 2020 The MathWorks, Inc.

MATLAB EXPO

감사합니다

