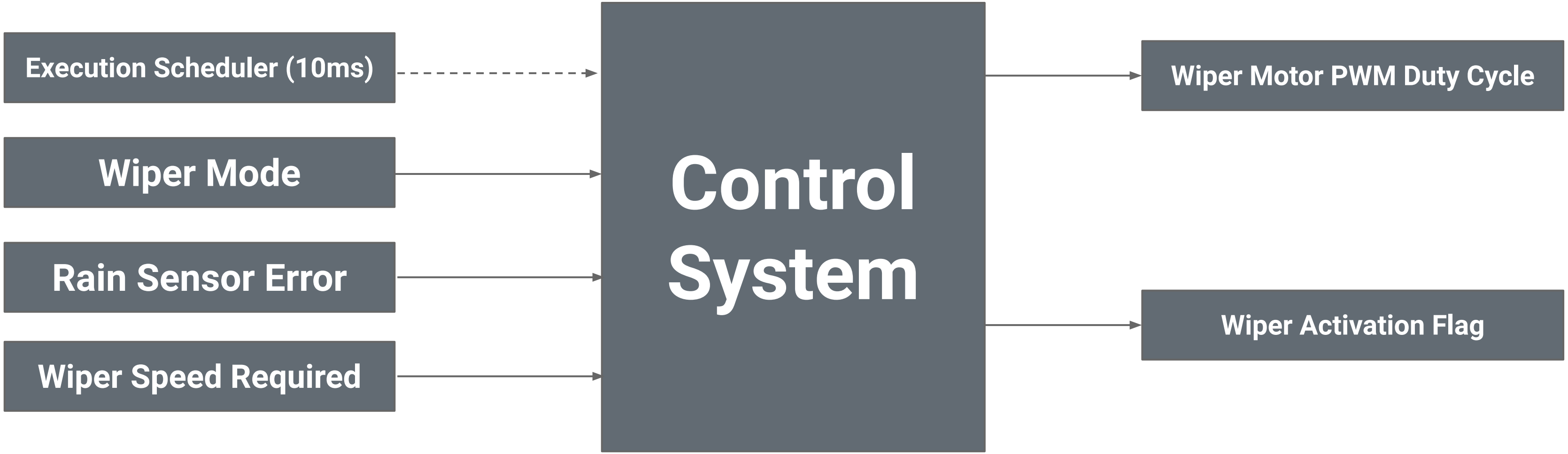
Vehicle Wiper control system using Simulink

By: Mohamed Magdy

Project Requirements : Root Level



Execution Scheduler (10ms)

Wiper Mode

Rain Sensor Error

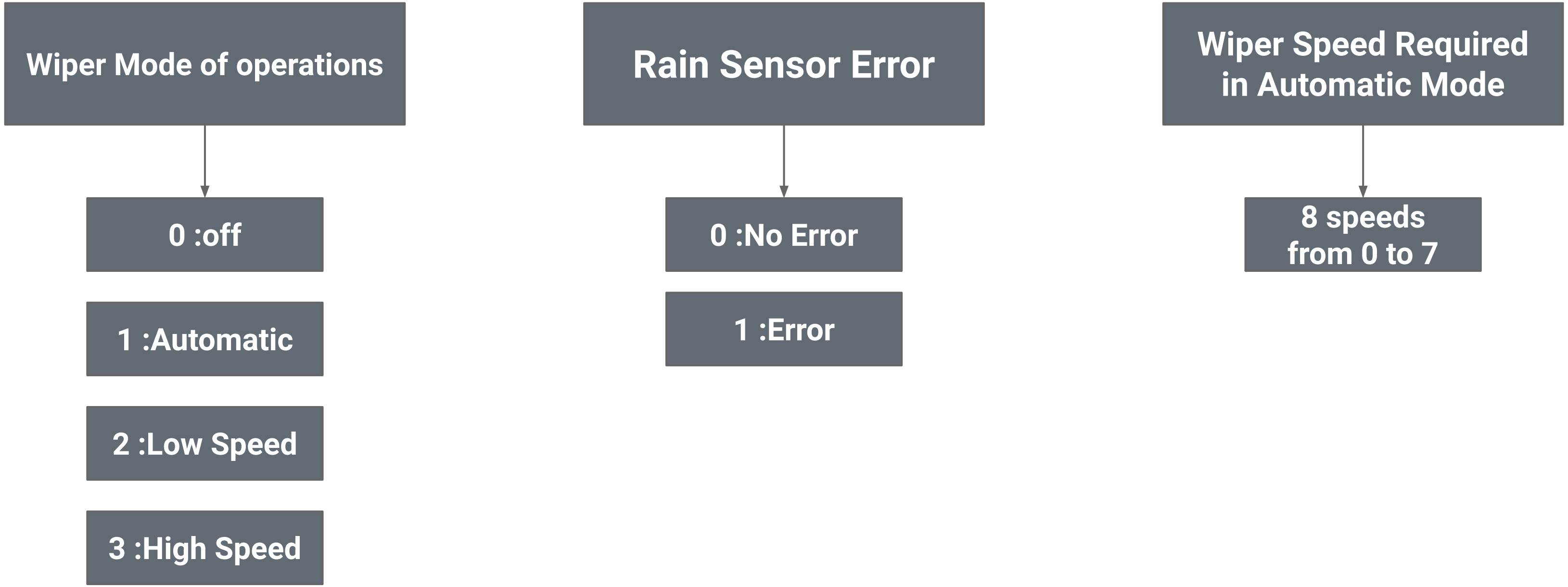
Wiper Speed Required

Control System

Wiper Motor PWM Duty Cycle

Wiper Activation Flag

Project Requirements : System Inputs



Wiper Mode of operations

- 0 :off
- 1 :Automatic
- 2 :Low Speed
- 3 :High Speed

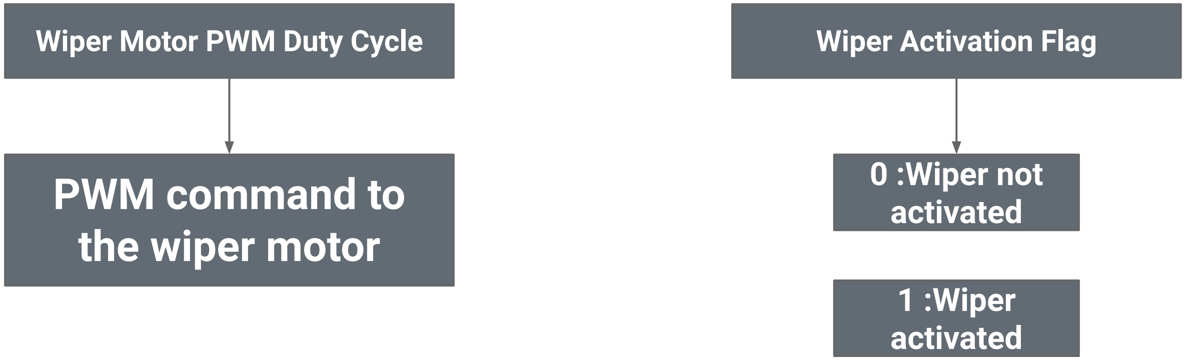
Rain Sensor Error

- 0 :No Error
- 1 :Error

Wiper Speed Required in Automatic Mode

- 8 speeds from 0 to 7

Project Requirements : System Outputs



Wiper Motor PWM Duty Cycle

PWM command to the wiper motor

Wiper Activation Flag

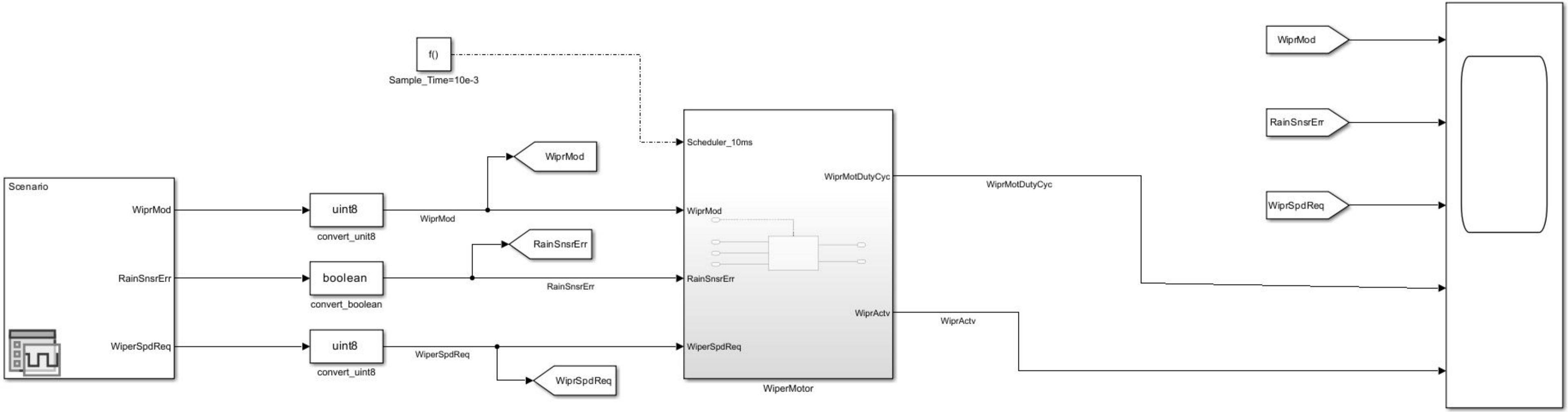
0 :Wiper not activated

1 :Wiper activated

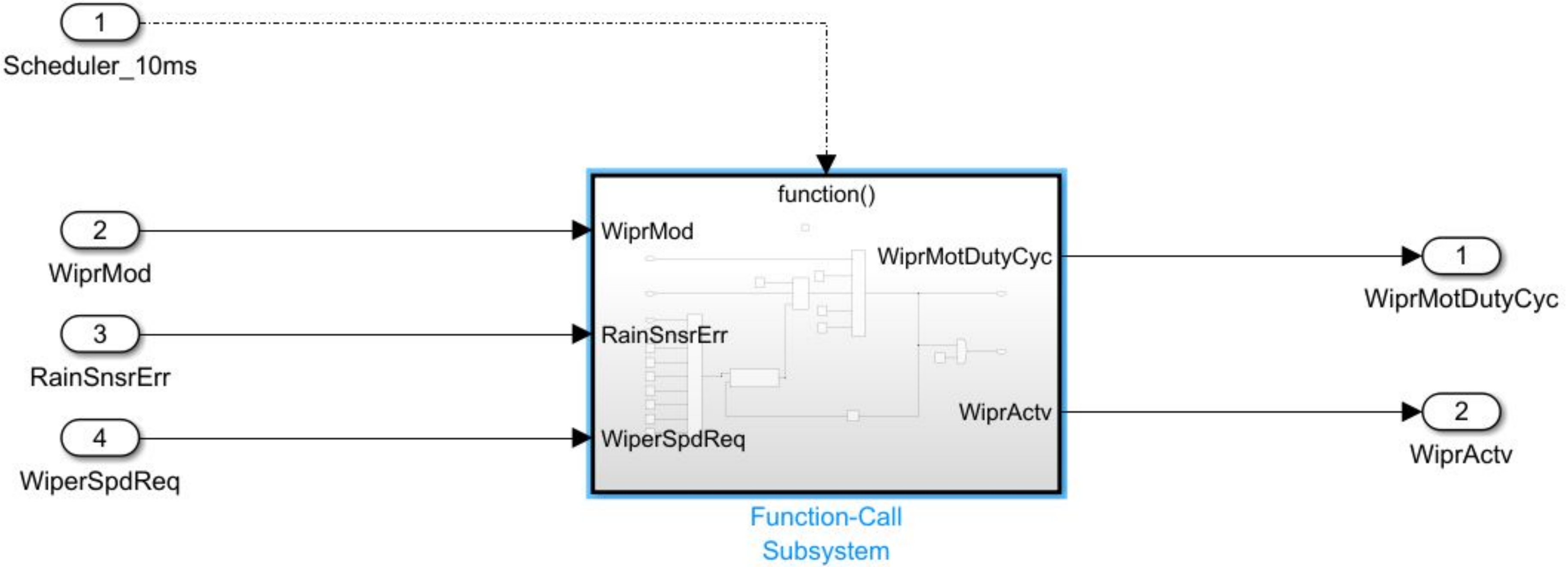
Project Requirements : control system logic

```
If (Wiper Mode = off)  then
    Wiper Motor PWM Duty Cycle = 0%
Else If (Wiper Mode = Low Speed)  then
    Wiper Motor PWM Duty Cycle = 40%
Else If (Wiper Mode = High Speed)  then
    Wiper Motor PWM Duty Cycle = 70%
Else If (Wiper Mode = Auto)  then
    If (Rain Sensor Error = true)  then
        Wiper Motor PWM Duty Cycle = 0%
    Else
        Select a PWM value from [0% 40% 50% 55% 60% 65% 70%] based on speed required [0 1 2 3 4 5 6 7]
        Smooth motor PWM transitions between different speeds
```

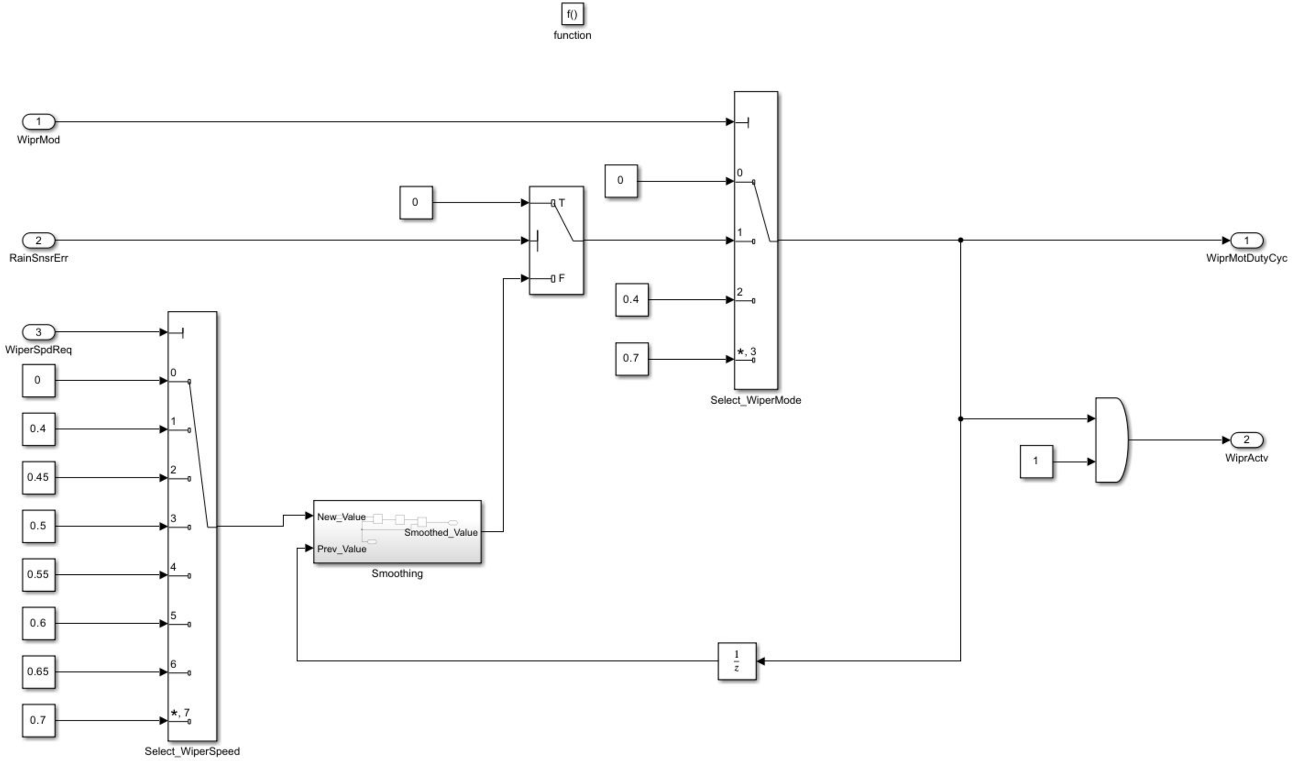
Simulink System Modelling: Root level



Wiper Motor Subsystem

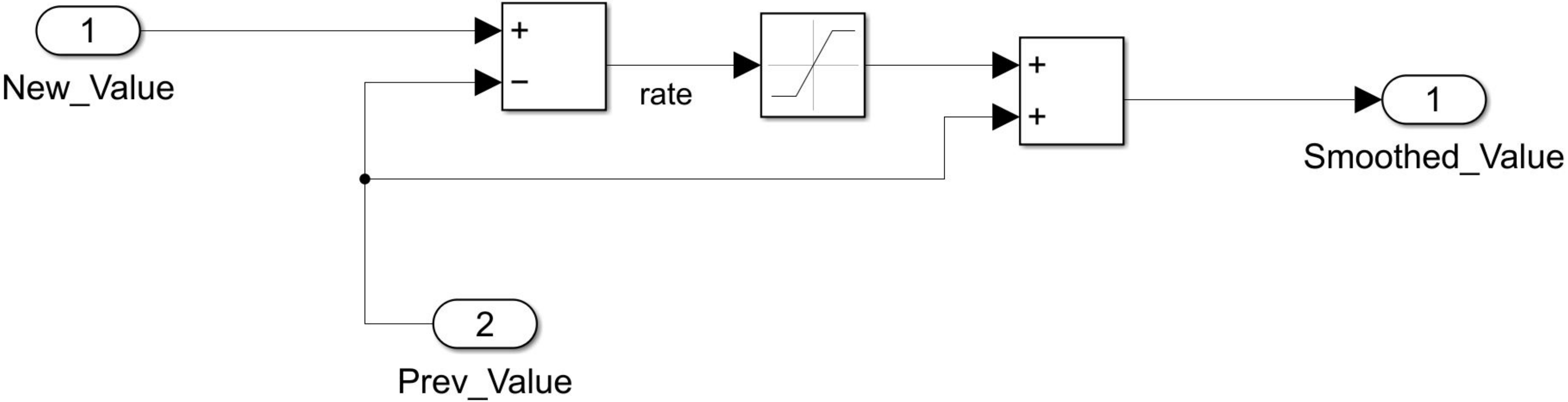


Wiper Motor Subsystem : Modelling the logic

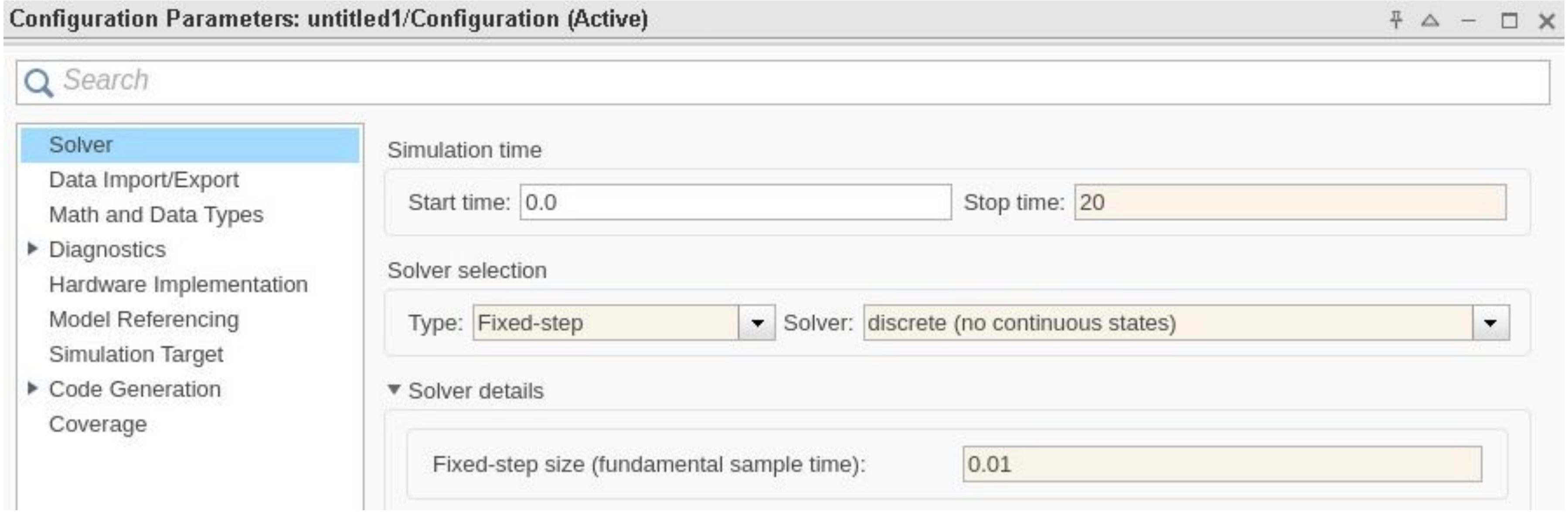


Wiper Motor Subsystem : Automatic Signal Smoothing:

Rate limiter



Model Testing: Setting up the Solver



Configuration Parameters: untitled1/Configuration (Active)

Simulation time

Start time: 0.0 Stop time: 20

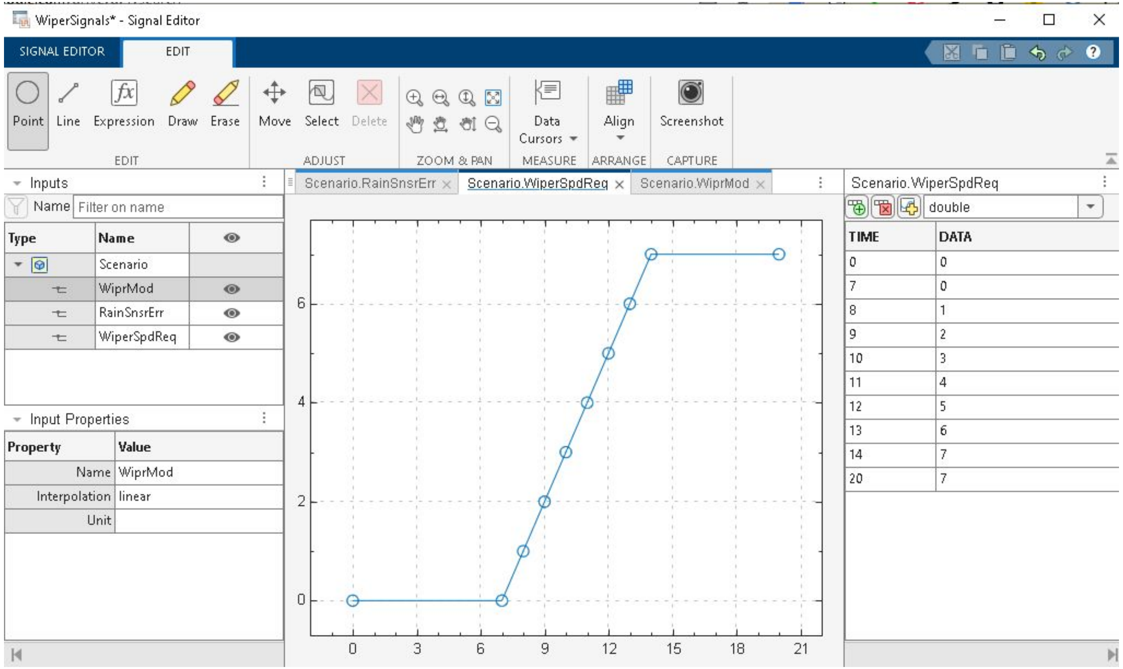
Solver selection

Type: Fixed-step Solver: discrete (no continuous states)

Solver details

Fixed-step size (fundamental sample time): 0.01

Model Testing: Creating Input signals scenario using Signal Editor



WiperSignals* - Signal Editor

Inputs

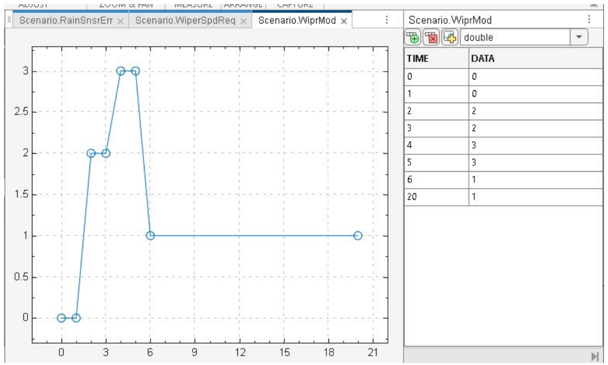
Type	Name	
	Scenario	
	WiprMod	
	RainSnsrErr	
	WiperSpdReq	

Input Properties

Property	Value
Name	WiprMod
Interpolation	linear
Unit	

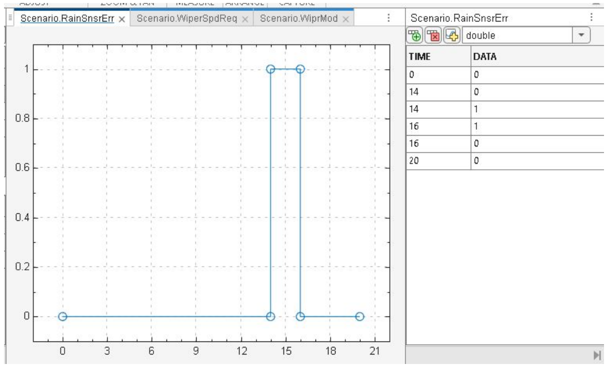
Scenario.WiperSpdReq

TIME	DATA
0	0
7	0
8	1
9	2
10	3
11	4
12	5
13	6
14	7
20	7



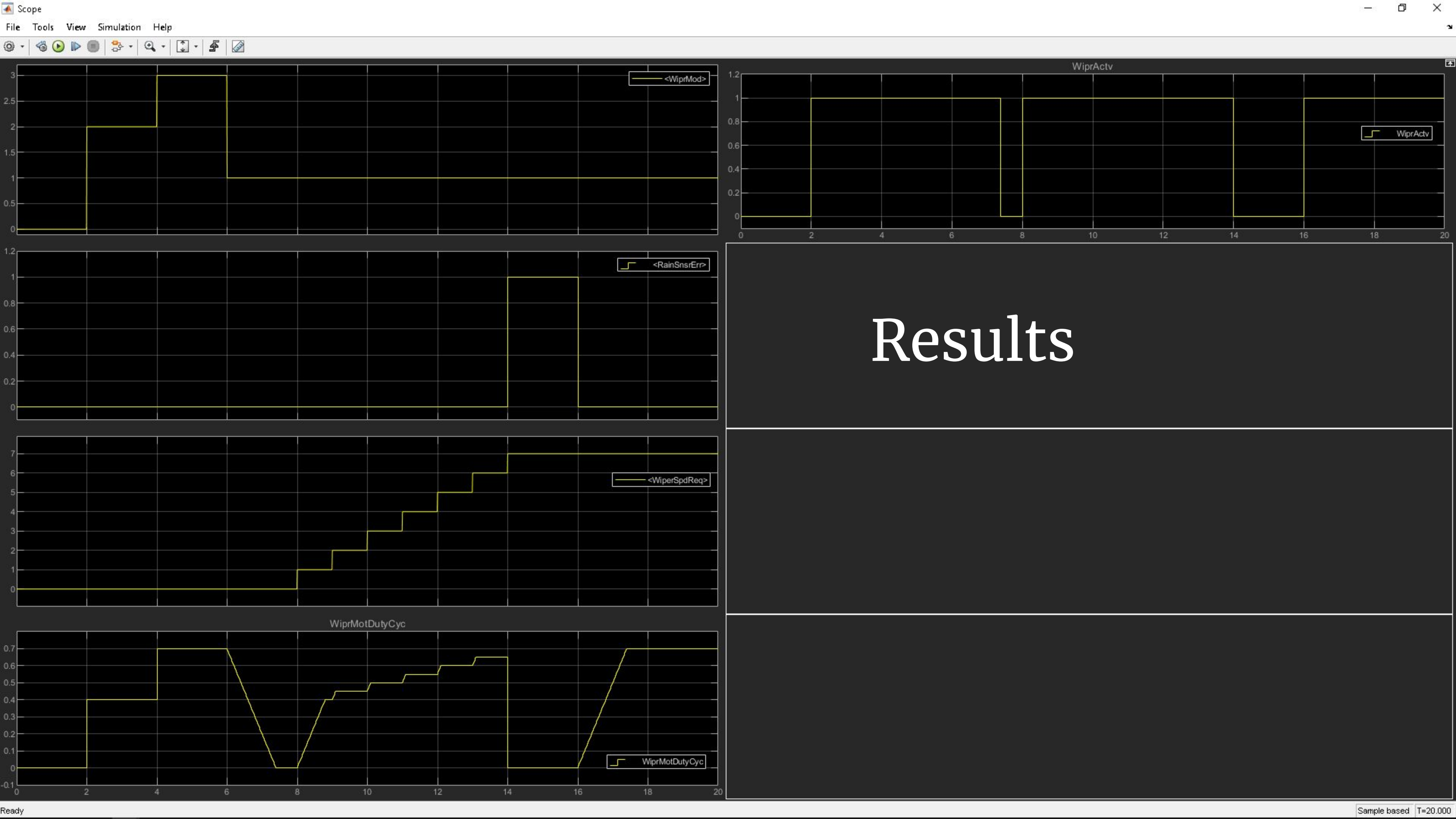
Scenario.WiprMod

TIME	DATA
0	0
1	0
2	2
3	2
4	3
5	3
6	1
20	1



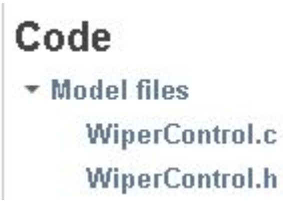
Scenario.RainSnsrErr

TIME	DATA
0	0
14	0
14	1
16	1
16	0
20	0


Scope
File
Tools
View
Simulation
Help
<WiprMod>
0
0.5
1
1.5
2
2.5
3
<RainSnsrErr>
0
0.2
0.4
0.6
0.8
1
1.2
<WiperSpdReq>
0
1
2
3
4
5
6
7
WiprMotDutyCyc
WiprMotDutyCyc
-0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0
2
4
6
8
10
12
14
16
18
20
WiprActv
WiprActv
0
0.2
0.4
0.6
0.8
1
1.2
0
2
4
6
8
10
12
14
16
18
20
Ready
Sample based
T=20.000

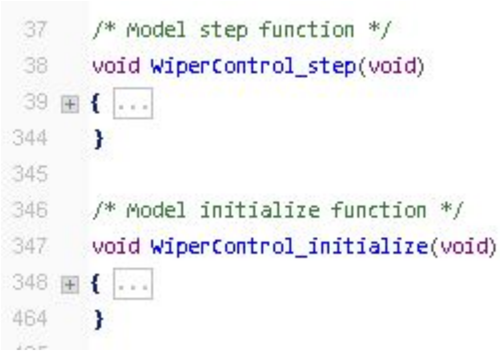
Results

C Code Generation



Code

- Model files
  - WiperControl.c
  - WiperControl.h



```
/* Model step function */
void WiperControl_step(void)
{ ...
}

/* Model initialize function */
void WiperControl_initialize(void)
{ ...
}
```