ACS Simulator

03-Sep-2002

Report editor: Thomas Bak. Input from Dan Bhanderi (AAU), Mogens Blanke (DTU), Henrik Niemann (DTU), Ron Noteborn (Terma), Michael Quottrup (AAU), Bent Ziegler (DTU)

This is the detailed design document, for the Rømer Attitude Control System (ACS) Algorithm development. This document provides the detailed design for the ACS algorithms, and is intended primarily for the control systems and software engineers involved with the ACS detailed design phase. This document does not necessarily replace the software detailed design for the ACS system, as it works on a system/simulation level. It deals with the design as Matlab/Simulink simulations and does not discuss specific software implementation problems that may arise when implementing the ACS system on the OBC.

This report constitutes the deliverable 3 from Aalborg University for the work described under work package 3620.

Table of Contents

List of Tables

List of Figures

Next
Introduction

Chapter 1. Introduction

Table of Contents

1.1. Section Title

This is the detailed design document, for the Rømer Attitude Control System (ACS) Algorithm development. This document provides the detailed design for the ACS algorithms in terms of a detailed design of a simulation application. It is intended primarily for the control systems and software engineers involved with the ACS detailed design phase. This document does not necessarily replace the software detailed design for the ACS system, as it works on a system/simulation level. It deals with the design as Matlab/Simulink simulations and does not discuss specific software implementation problems that may arise when implementing the ACS system on the OBC. The basis for this document is the architectural design as described in [1].

1.2. Section Title

This is the detailed design document, for the Rømer Attitude Control System (ACS) Algorithm development. This document provides the detailed design for the ACS algorithms in terms of a detailed design of a simulation application. It is intended primarily for the control systems and software engineers involved with the ACS detailed design phase. This document does not necessarily replace the software detailed design for the ACS system, as it works on a system/simulation level. It deals with the design as Matlab/Simulink simulations and does not discuss specific software implementation problems that may arise when implementing the ACS system on the OBC. The basis for this document is the architectural design as described in [1].

1.3. Abbreviations and Acronyms

```
AAD  Attitude Anomaly Detection
ACS  Attitude Control Subsystem
ADD  Architectural Design Document
CAN  Controller Area Network serial bus
CDH  Command and Data Handling unit
CHU  Camera Head Unit
DHS  Data Handling Software
COG  Center of Gravity
COP  Center of Pressure
DDD  Detailed Design Document
DPU  Data Processing Unit
ECEF Earth Centered Earth Fixed
ECI  Earth Centered Inertial
FDI  Failure Detection and Isolation
HK   Housekeeping
I/F  InterFace
IGRF International Geomagnetic Reference Field
JD   Julian Date
LEOP Launch and Early Operations Phase
LVLH Local Vertical Local Horizontal
MAG  Magnetometer
MOI  Moments of Inertia
MONS Measuring Oscillations in Nearby Stars
MTQ  Magnetorquer
MTA  Magnetorquer Assembly (the three magnetorquers)
N/A  Not Applicable
OBC  Onboard Computer
PCDU Power Control and Distribution Unit
P/L  Payload
PUS  Packet Utilization Standard
RMS  Root Mean Square
S/C  Spacecraft
SCB  Spacecraft Body frame
STR  Star Tracker
RGA  Rate Gyro Assembly
RAAN Right Ascension of the Ascending Node
RSR  Rømer Standard Reference
RWA  Reaction Wheel Assembly
SAS  Solar Aspect Sensor
```

```
SSA  Sun Sensor Assembly (all 6 SSA)
S/W  Software
TBC  To Be Confirmed
TBD  To Be Determined
TLE  Two-Line Elements
TM   Telemetry
TC   Telecommand
```

1.4. References

1. Peter Hoffmeyer. RØMER Project Management Plan RØMER/TEB/MAN/PLN/0001(1). Technical report, Terma A/S, April 2001.
2. Thomas Bak; Michael Quottrup; Mogens Blanke (DTU); and Henrik Niemann (DTU). RØMER System Definition Phase Attitude Control Subsystem Algorithms. Technical report, Aalborg University, June 2001.
3. Lars Hedemann Jensen. RØMER Spacecraft ICD. Interface Control Drawing Rev. 1, Terma A/S, 2001.
4. James R. Wertz. Spacecraft Attitude Determination and Control, volume 73. Kluwer Academic Publishers, 1987.
5. CDH S/W ICD. Interface Control Document, Terma A/S, 2001.
6. Billingsley Magnetics. Rømer Billingsley TFM100G2 Magnetometer ICD. Interface Control Document Issue 1, Terma A/S, 2001.
7. Kieran A. Carroll. RØMER Technical Memorandum No. 2001-3-1 - MW-300 Microwheel Configuration Options. Technical report, Dynacon Enterprises Limited, March 2001.
8. Kieran A. Carroll. RØMER Technical Memorandum No. 2001-3-3 - Microwheel Dynamics Model Block Diagram. Technical report, Dynacon Enterprises Limited, March 2001.
9. Kieran A. Carroll. RØMER Technical Memorandum No. 2001-3-2 - MW-200 and MW-300 Interface Information. Technical report, Dynacon Enterprises Limited, March 2001.
10. Dynacon Enterprises Limited. MicroWheel User Manual, July 2001. Document Version 1.0.
11. Gene F. Franklin; J. David Powell; and Abbas Emami-Naeini. Feedback Control of Dynamic Systems. Addison-Wesley Publishing Company, Inc., third edition, 1994.
12. Rebecca A. Masterson; David W. Miller; and Robert L. Grogan. Development of empirical and analytical reaction wheel disturbance models. American Institute of Aeronautics and Astronautics, 1999.
13. Hans-Christian B. Jensen and Rafal Wisniewski. Nonlinear observer based fault detection and isolation for a momentum wheel. August 2001.

Chapter 2. Model - acs_documentation

Table of Contents

2.1. acs_documentation

Table 2-1. acs_documentation System Information

Name	acs_documentation
Depth	0
Type	block_diagram
Blocks	Rømer ACS

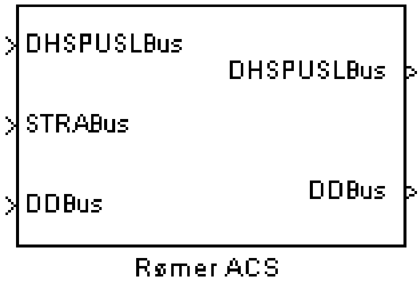
Table 2-2. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.1.1. Description

The aim of this model is to facilitate automatic generation of documation within Simulink.

Figure 2-1. acs_documentation



2.1.1.1. Signals

Table 2-3. acs_documentation Signal Information

InputSignalNames	N/A
OutputSignalNames	N/A

2.2. Rømer ACS

Table 2-4. Rømer ACS System Information

Name	Rømer ACS
Depth	1
Type	block
Blocks	DHSPUSLBus STRABus DDBus Around Bus Selector Bus Selector1 Bus Selector2 Bus Selector3 Bus Creator ProcesLayer RuleLayer SystemInterfaceLayer Work DHSPUSLBus DDBus

Table 2-5. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.2.1. Description

```
Rømer ACS
```

Description from system mask help.

2.2.1.1. Signals

Table 2-6. Rømer ACS Signal Information

InputSignalNames	
OutputSignalNames	

2.3. Bus Creator

Table 2-7. Bus Creator System Information

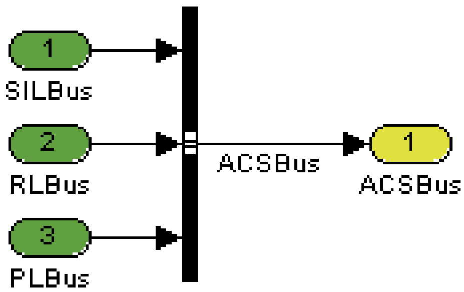
Name	Bus Creator
Depth	2
Type	block
Blocks	SILBus RLBus PLBus Bus Creator2 ACSBus

Table 2-8. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.3.1. Description

Figure 2-2. Bus Creator



2.3.1.1. Signals

Table 2-9. Bus Creator Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-10. Input Signal Information

Name	<5.001>
ParentBlock	acs_documentation/Rømer ACS/Bus Creator/SILBus
Description	

Table 2-11. Input Signal Information

Name	<6.001>
ParentBlock	acs_documentation/Rømer ACS/Bus Creator/RLBus
Description	

Table 2-12. Input Signal Information

Name	<7.001>
ParentBlock	acs_documentation/Rømer ACS/Bus Creator/PLBus
Description	

Table 2-13. Output Signal Information

Name	ACSBus
ParentBlock	acs_documentation/Rømer ACS/Bus Creator/Bus Creator2
Description	

2.4. ProcesLayer

Table 2-14. ProcesLayer System Information

Name	ProcesLayer
Depth	2
Type	block
Blocks	SILBus RLBus DDBus STRABus ActuatorManagement Around Bus Creator Bus Creator1 Bus Creator2 Bus Creator3 Bus Creator4 Bus Creator5 Bus Creator6 Bus Selector Bus Selector1 Bus Selector2 Bus Selector3 Bus Selector4 Control FaultDetection Navigation SensorManagement Subsystem Work PLBus

Table 2-15. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002

LastModifiedBy	tb

2.4.1. Description

2.4.1.1. Signals

Table 2-16. ProcesLayer Signal Information

InputSignalNames	<SILBus> <RLBus>
OutputSignalNames	

2.5. RuleLayer

Table 2-17. RuleLayer System Information

Name	RuleLayer
Depth	2
Type	block
Blocks	ACS/Bus Bus Creator1 Bus Creator2 Bus Creator3 Bus Creator4 Bus Creator6 Bus Selector1 Commander Constant2 Constant3 Constant7 Constant8 Ground Ground1 Ground2 Ground3 Ground4 Ground5 Ground6 Ground7 Guidance Terminator Terminator1 Terminator2 Terminator3 Terminator4 Terminator5 Bus

Table 2-18. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.5.1. Description

2.5.1.1. Signals

Table 2-19. RuleLayer Signal Information

InputSignalNames	<SILBus>
OutputSignalNames	

2.6. SystemInterfaceLayer

Table 2-20. SystemInterfaceLayer System Information

Name	SystemInterfaceLayer
Depth	2
Type	block
Blocks	DHSPUSLBus RLBus PLBus Bus Creator3 Command Interface Data Interface Subsystem Terminator Terminator1 Terminator2 SILBus

Table 2-21. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.6.1. Description

```
SystemInterfaceLayer: This layer provide the interface to the
ground station, and other onboard intelligence (applications). On
Rømer the SystemInterfaceLayer supports ground based planning and
execution by providing the necessary interface for such activities.
```

Description from system mask help.

2.6.1.1. Signals

Table 2-22. SystemInterfaceLayer Signal Information

InputSignalNames	<RLBus> <PLBus>
OutputSignalNames	

2.6.2. Validation

%<TestReports(RPTGEN_LOOP).name>

%<TestReports(RPTGEN_LOOP).name>

2.7. ActuatorManagement

Table 2-23. ActuatorManagement System Information

Name	ActuatorManagement
Depth	3
Type	block
Blocks	GetParameter (A10) SetParameter (A10) SetState (A11) SetTorque (A12) MTAReport (A13) RWAReport (A13) EnableMTA (A14) DisableMTA (A14) Actuator Management Report Actuator Report Actuator Requests Bus Creator Bus Selector Bus Selector1 Bus Selector2 Bus Selector3 Bus Selector4 Bus Selector5 Bus Selector6 Bus Selector7 Bus Selector8 Bus Selector9 Command Normalization Constant1 Enable Magnetic Torquer Assembly Fault Anomaly Detection Ground Magnetic Moment Conversion Momentum Management Terminator ActuatorReport (A34) ActuatorManReport (A44)

	ParameterReport (C01) MTARequest (E04) RWARequest (E06)

Table 2-24. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.7.1. Description

Navigation module

```
Handles interface functions to the actuators.
```

Description from system mask help.

2.7.1.1. Signals

Table 2-25. ActuatorManagement Signal Information

InputSignalNames	<GetParameter (A10)> <SetParameter (A10)> <SetState (A11)> <SetTorque (A12)> <MTAReport (A13)> <RWAReport (A13)> <EnableMTA (A14)> <DisableMTA (A14)>
OutputSignalNames	

2.7.2. Validation

Test001

2.8. Control

Table 2-26. Control System Information

Name	Control
Depth	3
Type	block
Blocks	GetParameter (A30) SetParameter (A30) SetState (A31) EstimatedState (A33) ActuatorReport (A34) Bus Creator1 Bus Creator2 Bus Creator5 Bus Selector1 Bus Selector2 Bus Selector3 Constant Constant11 Constant12 Constant13 Constant2 Control Algorithms Ground Terminator1 Terminator2 SetTorque (A12) ActuatorReport (A24) ControlReport (A43) ParameterReport (C03)

Table 2-27. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.8.1. Description

Navigation module

```
Covers the management and implementation of various control
algorithms for the control loops operated by the ACS.
```

Description from system mask help.

2.8.1.1. Signals

Table 2-28. Control Signal Information

InputSignalNames	<GetParameter (A30)> <SetParameter (A30)> <SetState (A31)> <EstimatedState (A33)> <ActuatorReport (A34)>
OutputSignalNames	

2.8.2. Validation

Test001

2.9. FaultDetection

Table 2-29. FaultDetection System Information

Name	FaultDetection
Depth	3
Type	block
Blocks	A40/41 A42 A43 A44 A45 Bus Creator3 Bus Creator5 Constant1 Constant2 Constant3 Terminator Terminator1 Terminator2 Terminator3 Terminator4 SystemInterface FaultAlarm

Table 2-30. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.9.1. Description

Navigation module

2.9.1.1. Signals

Table 2-31. FaultDetection Signal Information

InputSignalNames	<GetParameter (A40)> <SetParameter (A40)> <SetState (A41)> <NavigationReport (A42)> <ActuatorManReport (A44)>
OutputSignalNames	

2.10. Navigation

Table 2-32. Navigation System Information

Name	Navigation
Depth	3
Type	block
Blocks	GetParameter (A20) SetParameter (A20) SetState (A21) ReferenceState (A22) SensorReport (A23) ActuatorReport (A24) SetOrbitalElements (A25) AD Bus Creator Bus Creator1 Bus Selector3 Ground Ground10 Ground11 Ground12 Ground3 Ground8 Ground9 Qmult1 Terminator1 Terminator2 Terminator3 Terminator4 Terminator7 EstimatedState (A33) NavigationReport (A42) SetSlewStartAttRate (B11) ParameterReport (C02)

Table 2-33. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.10.1. Description

Navigation module

2.10.1.1. Signals

Table 2-34. Navigation Signal Information

InputSignalNames	<SetParameter (A20)> <GetParameter (A30)> <SetState (A21)> <ReferenceState (A22)> <SensorReport (A23)> <ActuatorReport (A24)> <SetOrbitalElements (A25)>
OutputSignalNames	

2.11. SensorManagement

Table 2-35. SensorManagement System Information

Name	SensorManagement
Depth	3
Type	block
Blocks	A00/01 A02/03 Bus Creator Bus Creator1 Constant Constant1 Constant10 Constant11 Constant12 Constant13 Constant2 Constant3 Constant4 Constant5 Constant6 Constant7 Constant8 Constant9 Ground Terminator Terminator1 SystemInterface SensorRequest SensorManReport SensorReport

Table 2-36. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002

LastModifiedBy	tb

2.11.1. Description

SensorManagement, encapsulates sensor management functionality.

2.11.1.1. Signals

Table 2-37. SensorManagement Signal Information

InputSignalNames	<GetParameter (A00)> <SetParameter (A00)>
OutputSignalNames	

2.12. Subsystem

Table 2-38. Subsystem System Information

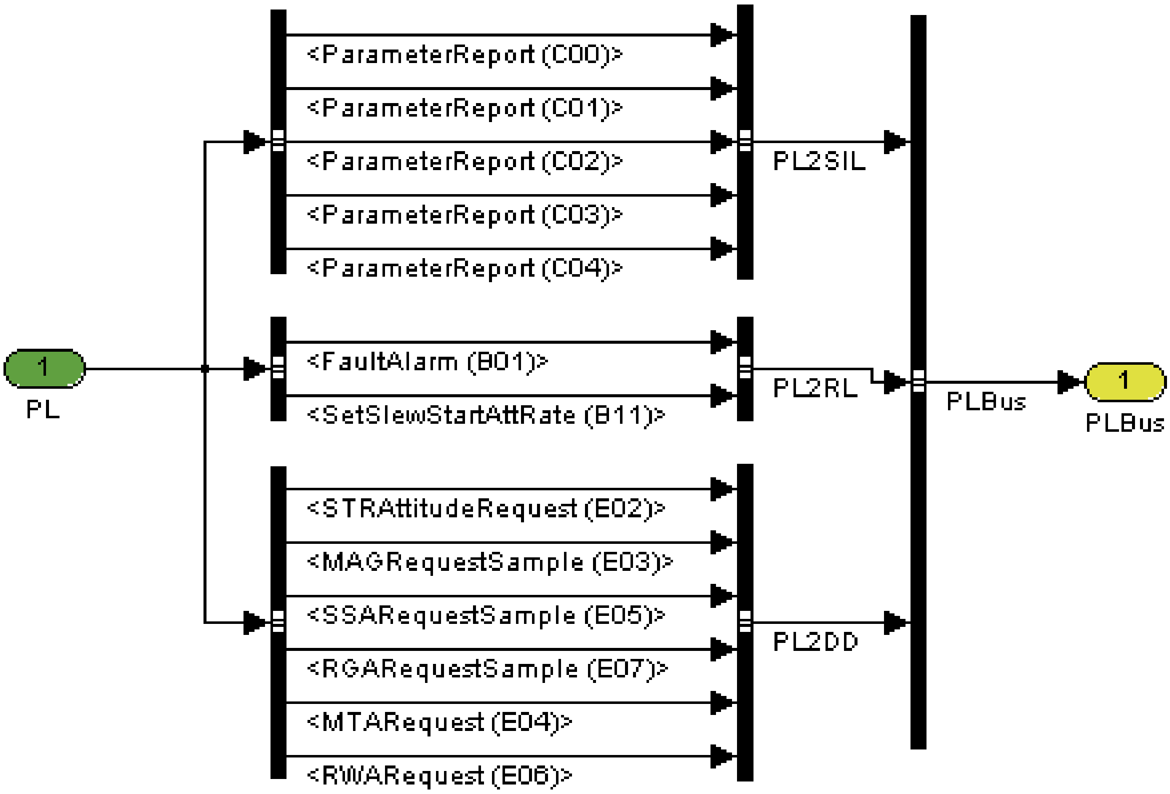
Name	Subsystem
Depth	3
Type	block
Blocks	PL Bus Creator Bus Creator1 Bus Creator2 Bus Creator3 Bus Selector1 Bus Selector2 Bus Selector5 PLBus

Table 2-39. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.12.1. Description

Figure 2-3. Subsystem



2.12.1.1. Signals

Table 2-40. Subsystem Signal Information

InputSignalNames	PL
OutputSignalNames	

Table 2-41. Input Signal Information

Name	<3997.002>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Subsystem/PL
Description	

Table 2-42. Output Signal Information

Name	PLBus
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Subsystem/Bus Creator1
Description	

2.13. Commander

Table 2-43. Commander System Information

Name	Commander
Depth	3
Type	block
Blocks	B00 B01 B02 Ground5 Ground6 Ground7 SetState ActuatorManagement SetState Control SetState FaultDetection SetState Guidance SetState Navigation SetState SensorManagement Terminator Terminator1 Terminator2 A01 A11 A21 A31 A41 C10 C20 C06 B12

Table 2-44. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.13.1. Description

2.13.1.1. Signals

Table 2-45. Commander Signal Information

InputSignalNames	
OutputSignalNames	ActManState

2.14. Guidance

Table 2-46. Guidance System Information

Name	Guidance
Depth	3
Type	block
Blocks	B10 B12 B13 Ground Ground1 Terminator Terminator1 Terminator2 A22 C05

Table 2-47. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.14.1. Description

2.14.1.1. Signals

Table 2-48. Guidance Signal Information

InputSignalNames	

OutputSignalNames	

2.15. Command Interface

Table 2-49. Command Interface System Information

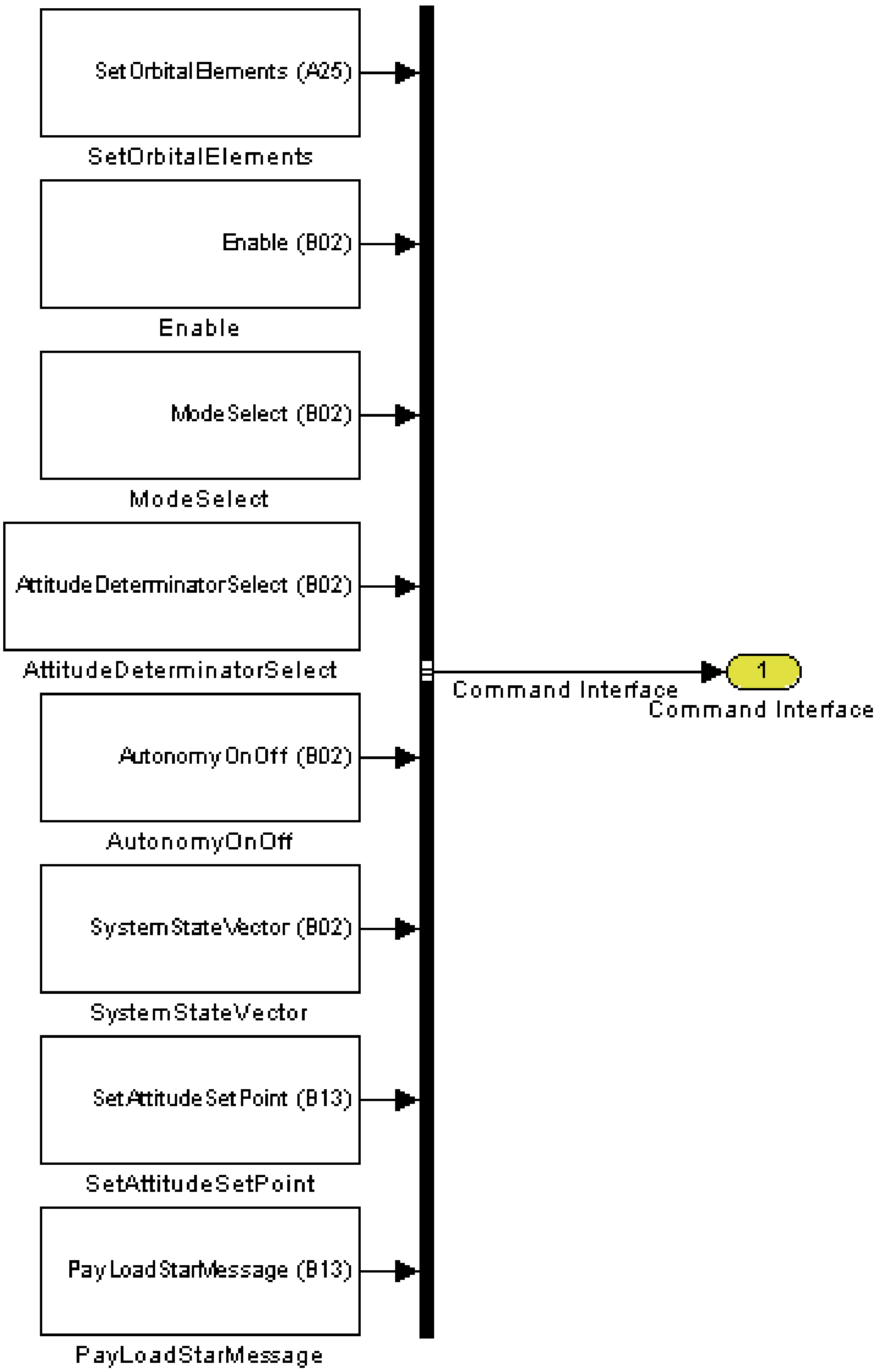
Name	Command Interface
Depth	3
Type	block
Blocks	AttitudeDeterminatorSelect AutonomyOnOff Bus Creator4 Enable ModeSelect PayLoadStarMessage SetAttitudeSetPoint SetOrbitalElements SystemStateVector Command Interface

Table 2-50. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.15.1. Description

Figure 2-4. Command Interface



2.15.1.1. Signals

Table 2-51. Command Interface Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-52. Output Signal Information

Name	Command Interface
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Command Interface/Bus Creator4
Description	

2.16. Data Interface

Table 2-53. Data Interface System Information

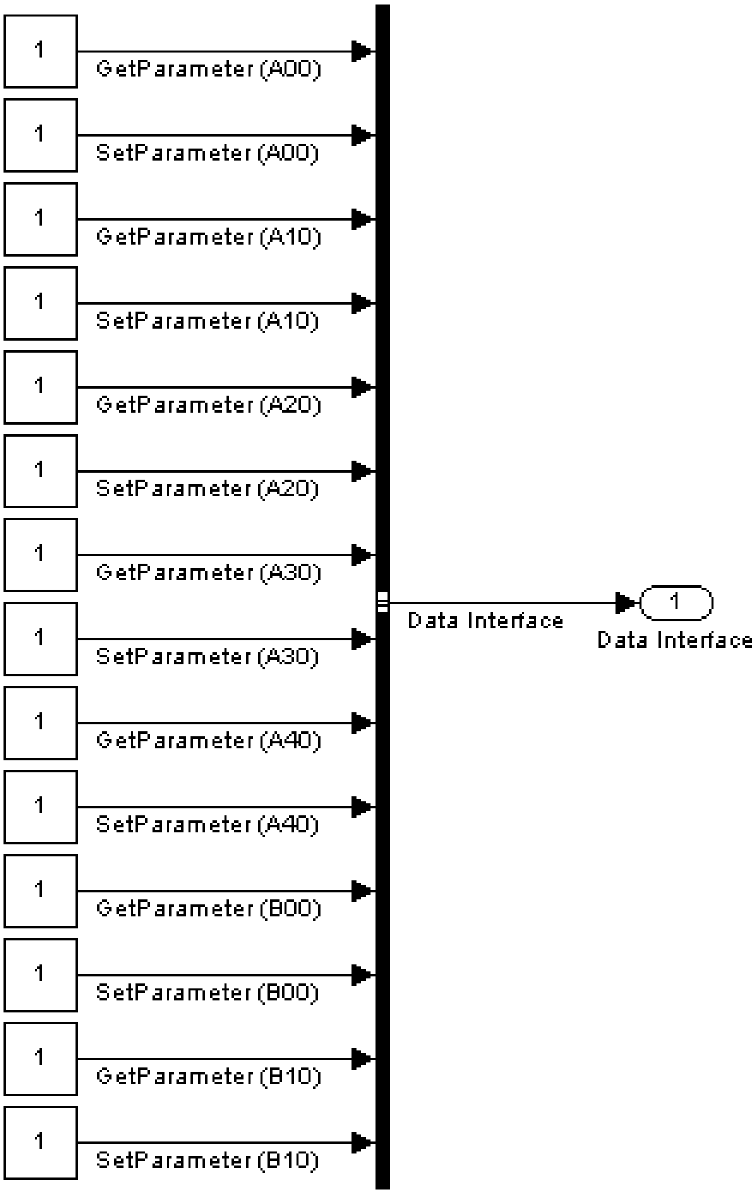
Name	Data Interface
Depth	3
Type	block
Blocks	Bus Creator4 Constant Constant1 Constant10 Constant11 Constant12 Constant13 Constant2 Constant3 Constant4 Constant5 Constant6 Constant7 Constant8 Constant9 Data Interface

Table 2-54. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.16.1. Description

Figure 2-5. Data Interface



2.16.1.1. Signals

Table 2-55. Data Interface Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-56. Output Signal Information

Name	Data Interface
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Data Interface/Bus Creator4
Description	

2.17. Subsystem

Table 2-57. Subsystem System Information

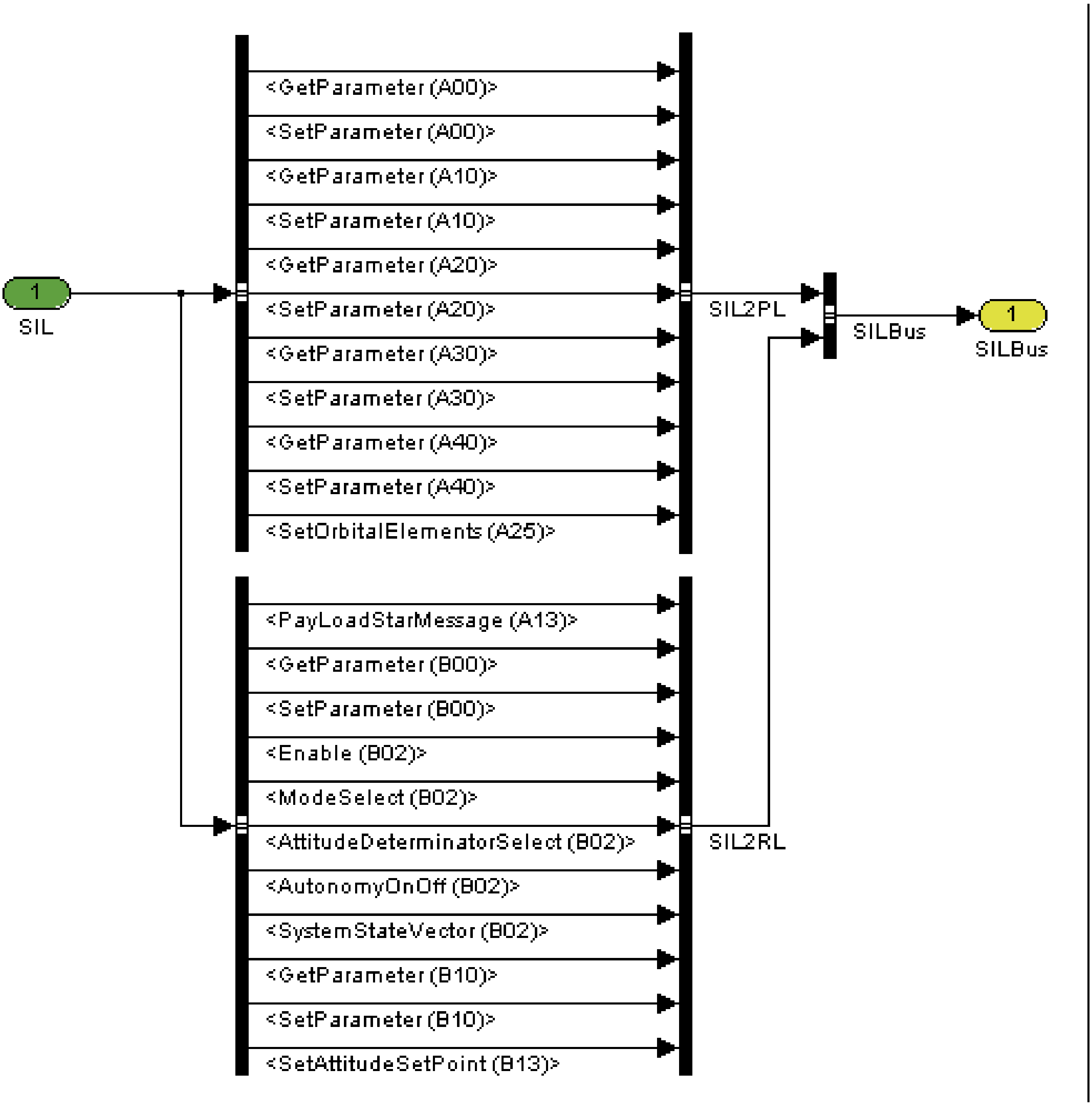
Name	Subsystem
Depth	3
Type	block
Blocks	SIL Bus Creator Bus Creator1 Bus Creator2 Bus Selector Bus Selector1 SILBus

Table 2-58. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.17.1. Description

Figure 2-6. Subsystem



2.17.1.1. Signals

Table 2-59. Subsystem Signal Information

InputSignalNames	SIL
OutputSignalNames	

Table 2-60. Input Signal Information

Name	<4206.0035>
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Subsystem/SIL
Description	

Table 2-61. Output Signal Information

Name	SILBus
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Subsystem/Bus Creator1
Description	

2.18. Actuator Management Report

Table 2-62. Actuator Management Report System Information

Name	Actuator Management Report
Depth	4
Type	block
Blocks	RWATorqueApp RWAMeasSpeed MTAMagMomentApp ActManState ActManDet Bus Creator2 ActuatorManReport (A44)

Table 2-63. acs_documentation Information

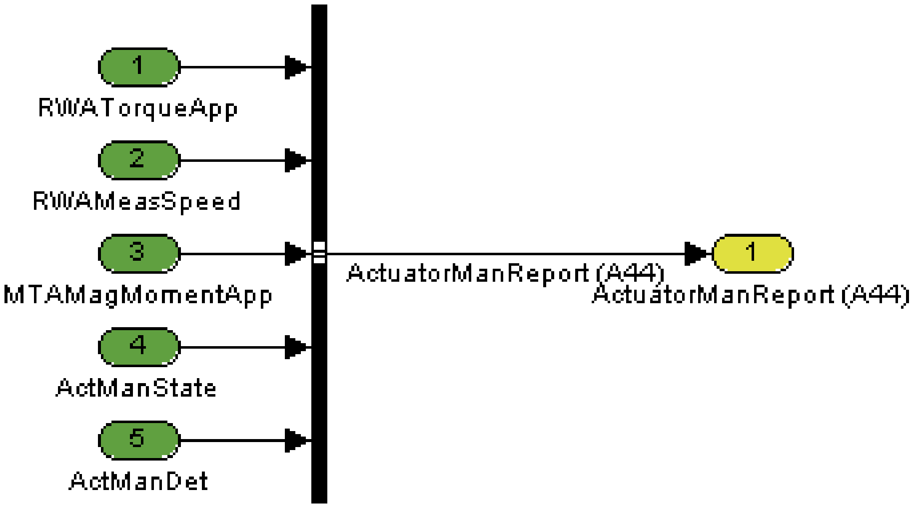
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.18.1. Description

```
Generates the ActuatorManReport to the FaultDetection. The report
contains information about the the actuators, the internal symbolic
state of the ActuatorManagement, and the detection of actuator
failures.
```

Description from system mask help.

Figure 2-7. Actuator Management Report



2.18.1.1. Signals

Table 2-64. Actuator Management Report Signal Information

InputSignalNames	<RWATorqueApp> <RWAMeasSpeed> <MTAMagMomentApp> <ActManState>
OutputSignalNames	

2.18.2. Validation

Test001

2.19. Actuator Report

Table 2-65. Actuator Report System Information

Name	Actuator Report
Depth	4
Type	block
Blocks	RWAReport MTAReport Bus Creator5 Bus Selector Bus Selector1 MTA Actuator Report RWA Actuator Report ActuatorReport (A34)

Table 2-66. acs_documentation Information

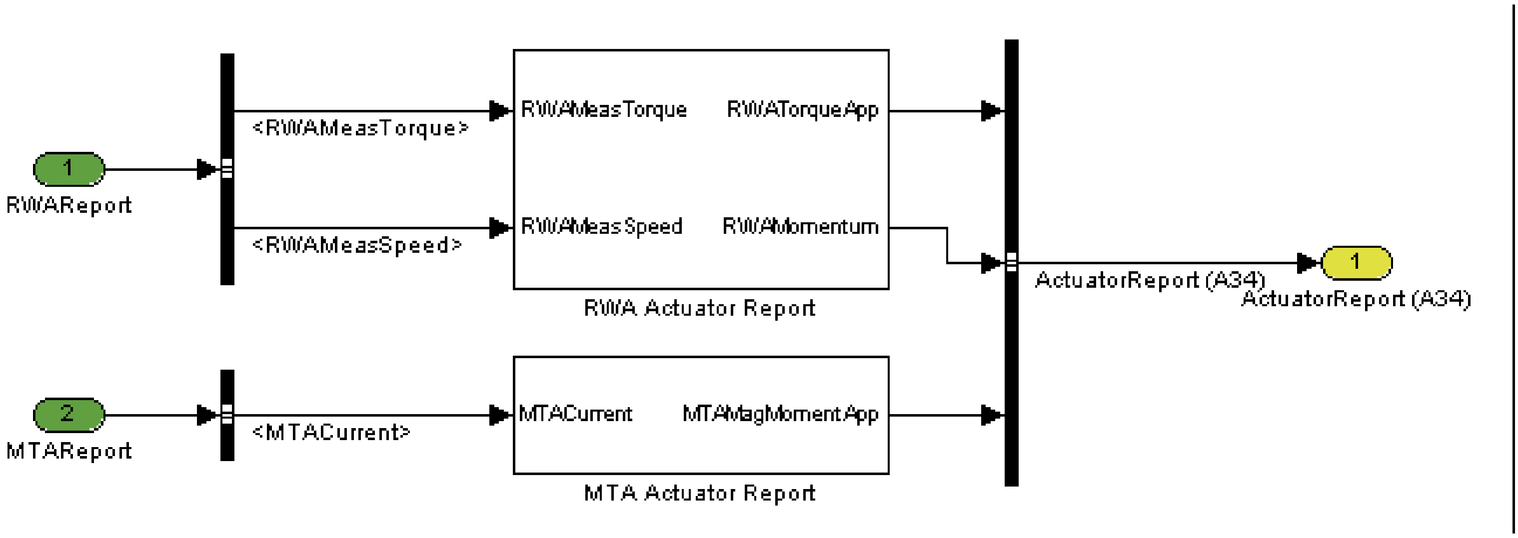
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.19.1. Description

```
Generates the ActuatorReport to the Control. The report contains information
about the actuators, needed for the execution of the control algorithms.
```

Description from system mask help.

Figure 2-8. Actuator Report



2.19.1.1. Signals

Table 2-67. Actuator Report Signal Information

InputSignalNames	
OutputSignalNames	

2.19.2. Validation

[Test001](#)

2.20. Actuator Requests

Table 2-68. Actuator Requests System Information

Name	Actuator Requests
Depth	4
Type	block
Blocks	RWAEnable MTACurrentCmdN RWATorqueCmdN MTAEnableFinal Bus Creator1 Bus Creator8 Constant1 Constant7 MTARequest (E04) RWARequest (E06)

Table 2-69. acs_documentation Information

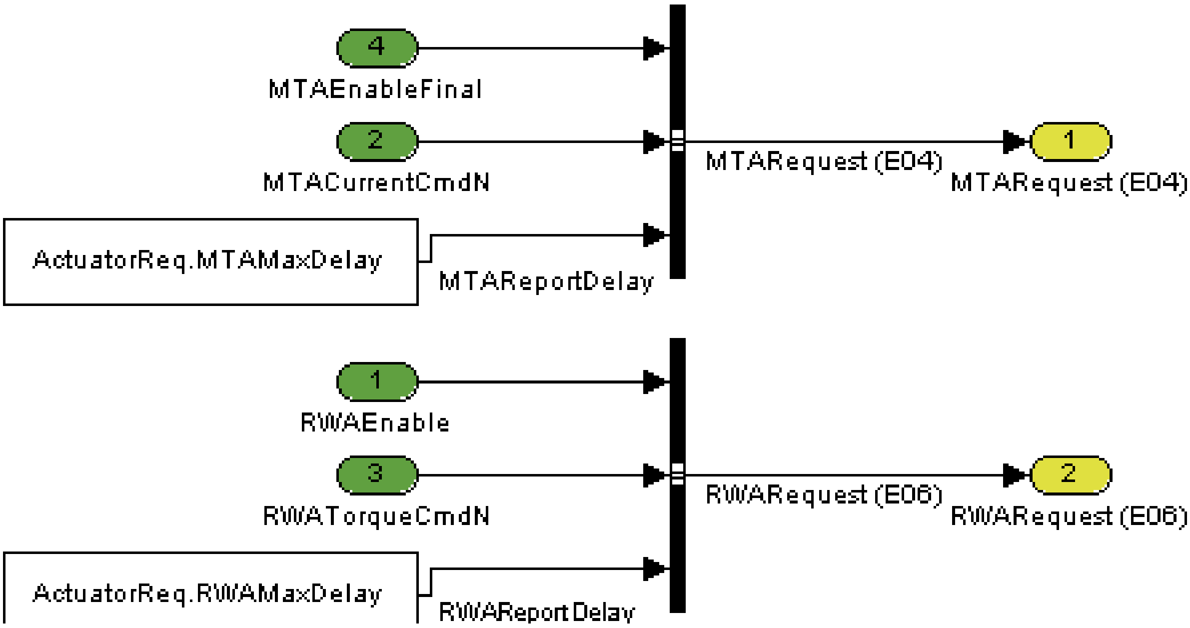
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.20.1. Description

```
Handles the request to the actuators.
```

Description from system mask help.

Figure 2-9. Actuator Requests



2.20.1.1. Signals

Table 2-70. Actuator Requests Signal Information

InputSignalNames	<RWAEnable>
OutputSignalNames	

2.20.2. Validation

Test001

2.21. Command Normalization

Table 2-71. Command Normalization System Information

Name	Command Normalization
Depth	4
Type	block
Blocks	RWACmdTorque MTACmdCurrent MTA Command Current Normalization RWA Command Torque Normalization RWATorqueCmdN MTACurrentCmdN

Table 2-72. acs_documentation Information

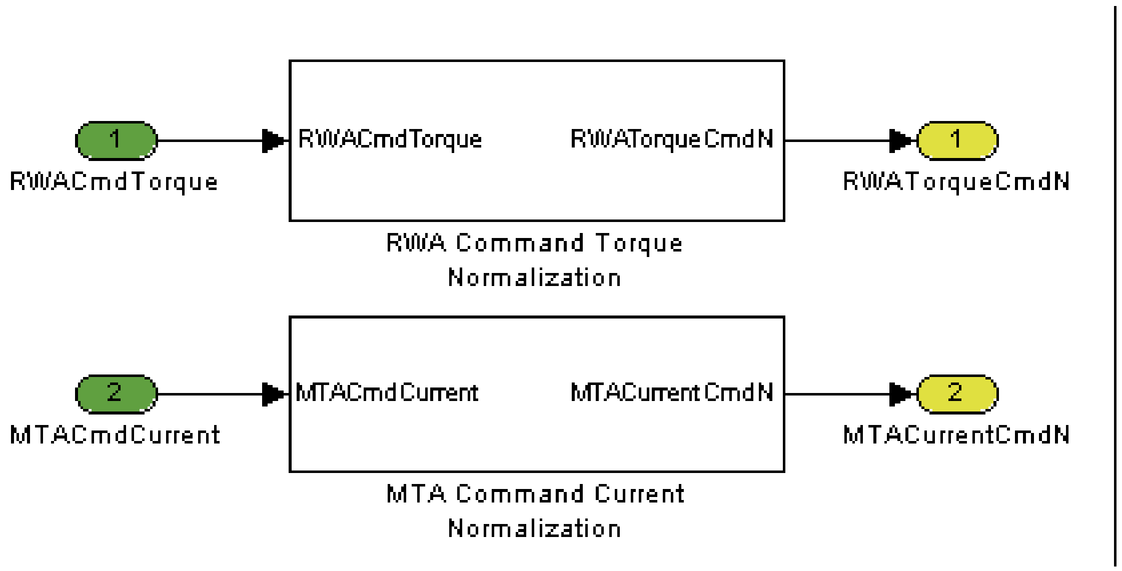
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.21.1. Description

```
Handles the normalization of command torque and current to the
actuators.
```

Description from system mask help.

Figure 2-10. Command Normalization



2.21.1.1. Signals

Table 2-73. Command Normalization Signal Information

InputSignalNames	
OutputSignalNames	

2.21.2. Validation

Test001

2.22. Enable Magnetic Torquer Assembly

Table 2-74. Enable Magnetic Torquer Assembly System Information

Name	Enable Magnetic Torquer Assembly
Depth	4
Type	block
Blocks	MTAStatus MTAEnable EnableMTA DisableMTA Bus Selector Logical Operator1 Logical Operator2 MTAEnableFinal

Table 2-75. acs_documentation Information

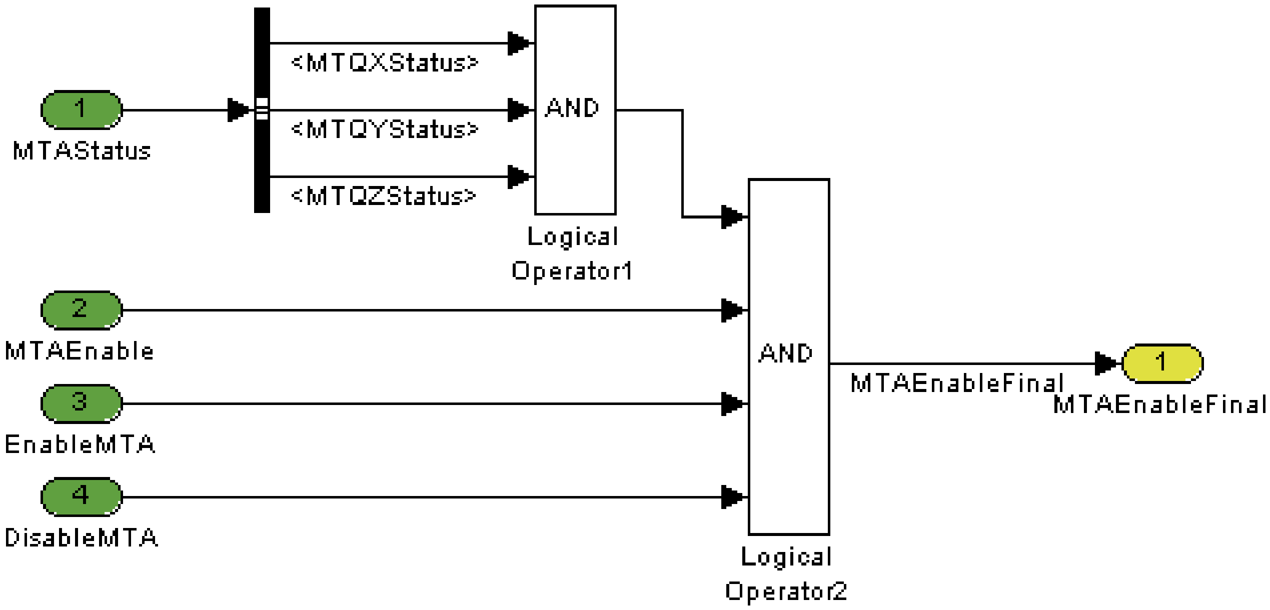
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.22.1. Description

```
Handles the enableing/disableing of the magnetic torquer assembly.
```

Description from system mask help.

Figure 2-11. Enable Magnetic Torquer Assembly



2.22.1.1. Signals

Table 2-76. Enable Magnetic Torquer Assembly Signal Information

InputSignalNames	<MTAStatus> <MTAEnable>
OutputSignalNames	

2.22.2. Validation

Test001

2.23. Fault Anomaly Detection

Table 2-77. Fault Anomaly Detection System Information

Name	Fault Anomaly Detection
Depth	4
Type	block
Blocks	MTAReport (A13) RWAReport (A13) ActuatorCmdN SetParameter (A10) Actuator Command Range Check Actuator Outlier Check Actuator Range Check Actuator Status Check Bus Creator Bus Creator8 Bus Selector Bus Selector1 Bus Selector10 Bus Selector11 Bus Selector12 Bus Selector2 Bus Selector3 Bus Selector4 Bus Selector5 Bus Selector6 Bus Selector7 Bus Selector8 Bus Selector9 Ground Ground1 ActManDet

Table 2-78. acs_documentation Information

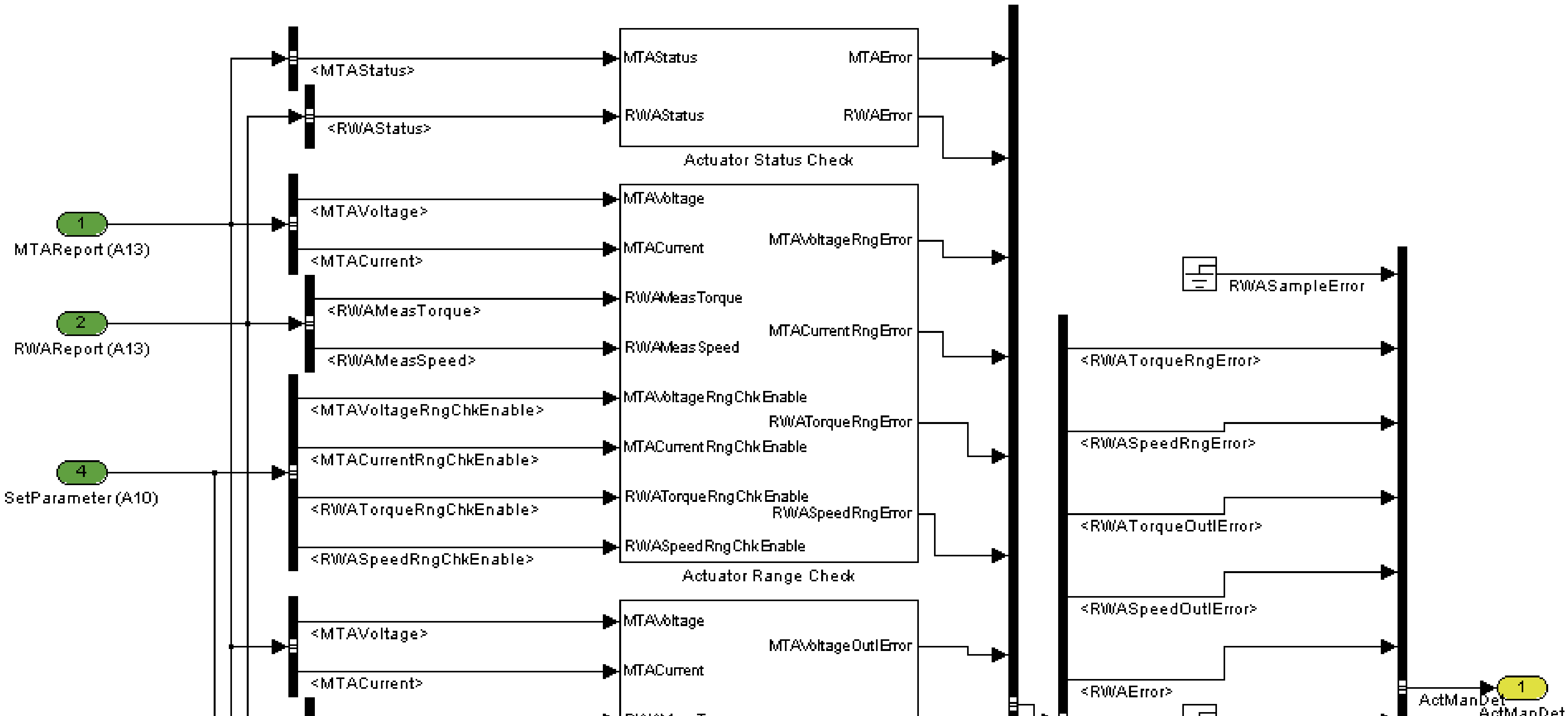
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

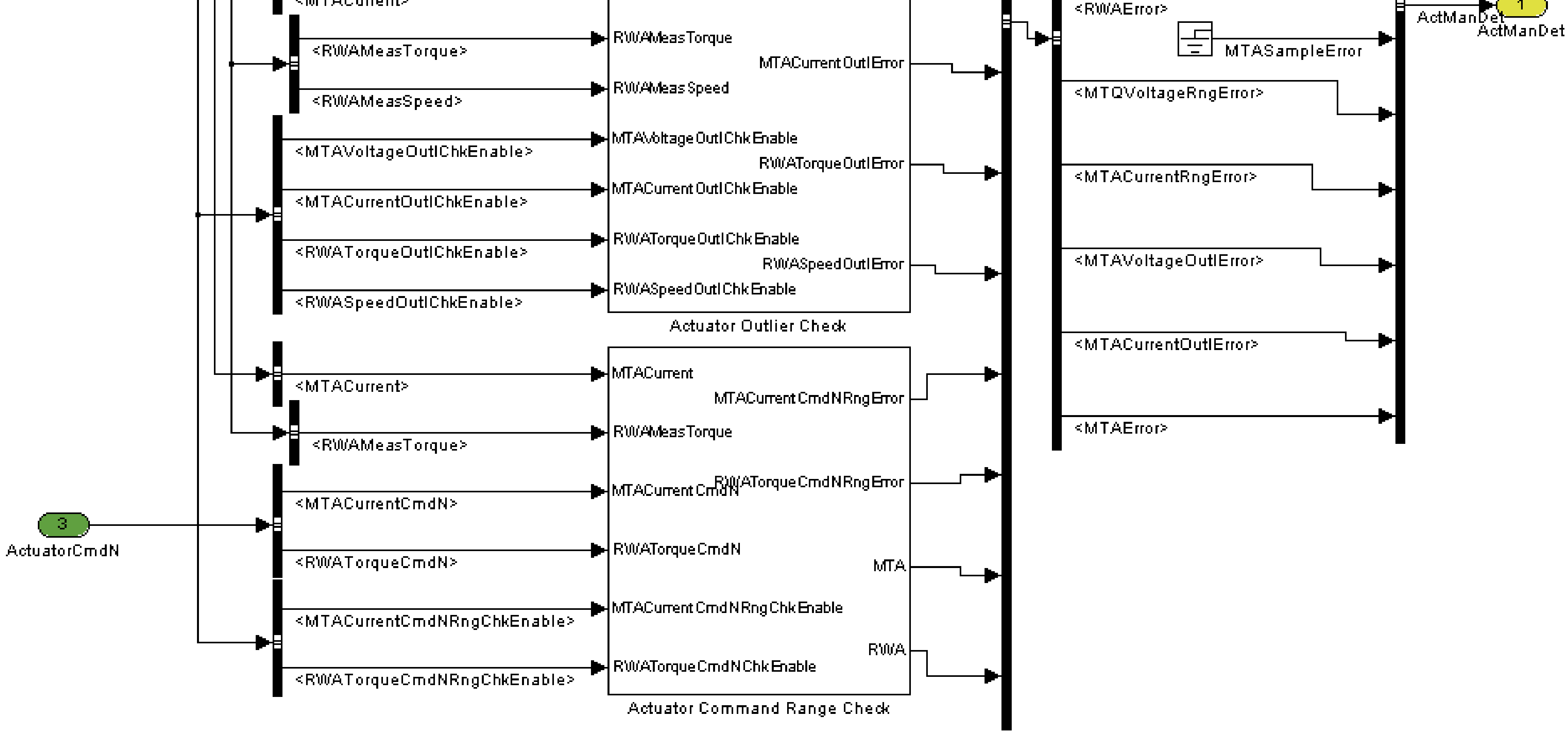
2.23.1. Description

```
Handles fault detection on the single actuator level by checks of range, outlier and consistency with commanded
current/torque commands based on feedback from the actuators.
```

Description from system mask help.

Figure 2-12. Fault Anomaly Detection





2.23.1.1. Signals

Table 2-79. Fault Anomaly Detection Signal Information

InputSignalNames	
OutputSignalNames	

2.23.2. Validation

Test001

2.24. Magnetic Moment Conversion

Table 2-80. Magnetic Moment Conversion System Information

Name	Magnetic Moment Conversion
Depth	4
Type	block
Blocks	MTACtrlMagMoment Enable Demux1 Demux2 Demux3 Matrix Gain MagMomConv.Arsr2mtqx Matrix Gain MagMomConv.Arsr2mtqy Matrix Gain MagMomConv.Arsr2mtqz Matrix Gain MagMomConv.Ascb2rsr Matrix Gain MagMomConv.MTAScaleFactor Mux Terminator1 Terminator2 Terminator3 Terminator4 Terminator5 Terminator6 MTACmdCurrent

Table 2-81. acs_documentation Information

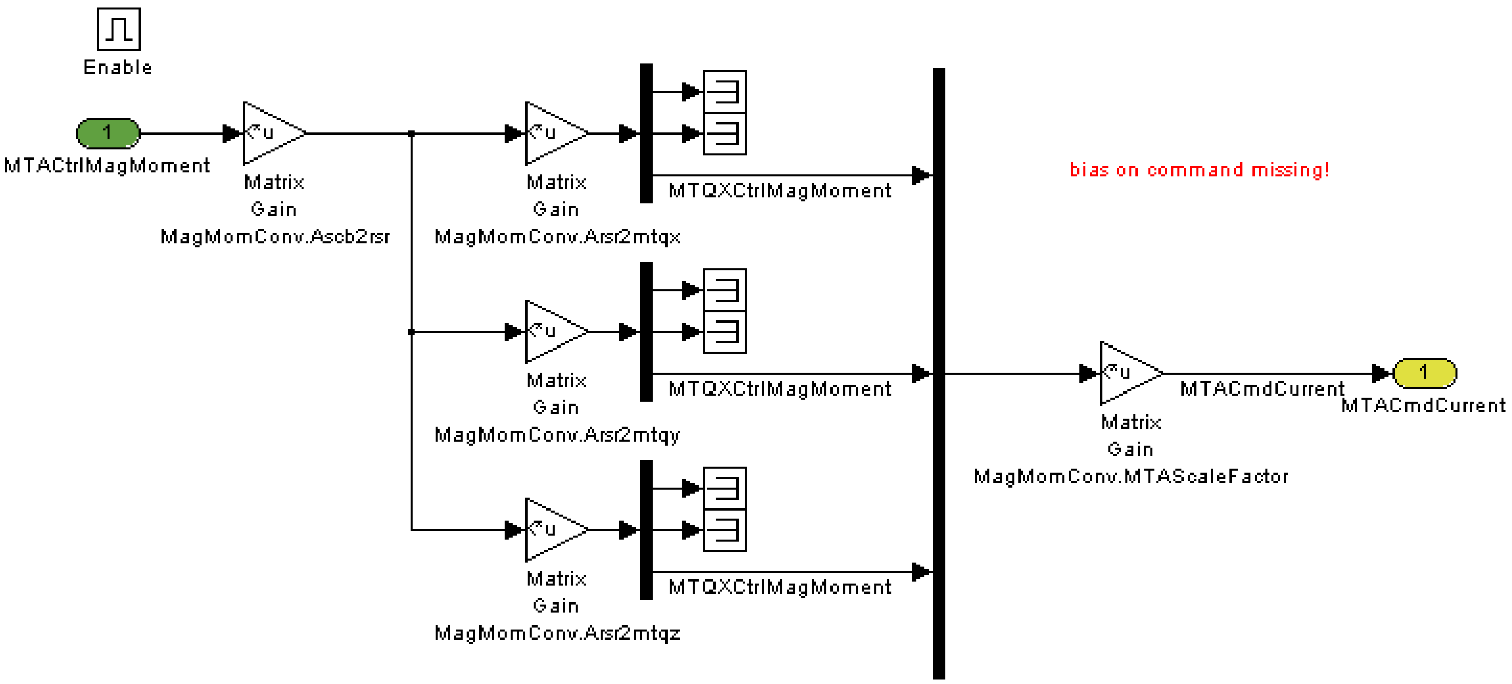
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.24.1. Description

```
Handles the conversion of the MTA control magnetic moment into command currents to the
magnetic torquers in the MTA.
```

Description from system mask help.

Figure 2-13. Magnetic Moment Conversion



2.24.1.1. Signals

Table 2-82. Magnetic Moment Conversion Signal Information

InputSignalNames	<MTACtrlMagMoment>
OutputSignalNames	

2.24.2. Validation

Test001

2.25. Momentum Management

Table 2-83. Momentum Management System Information

Name	Momentum Management
Depth	4
Type	block
Blocks	MomCtrlEnable RWAEnable RWACtrlTorque RWANomSpeed RWAMeasSpeed RWAStatus Control Torque Distribution Momentum Control RWA Valid Logic Sum RWACmdTorque

Table 2-84. acs_documentation Information

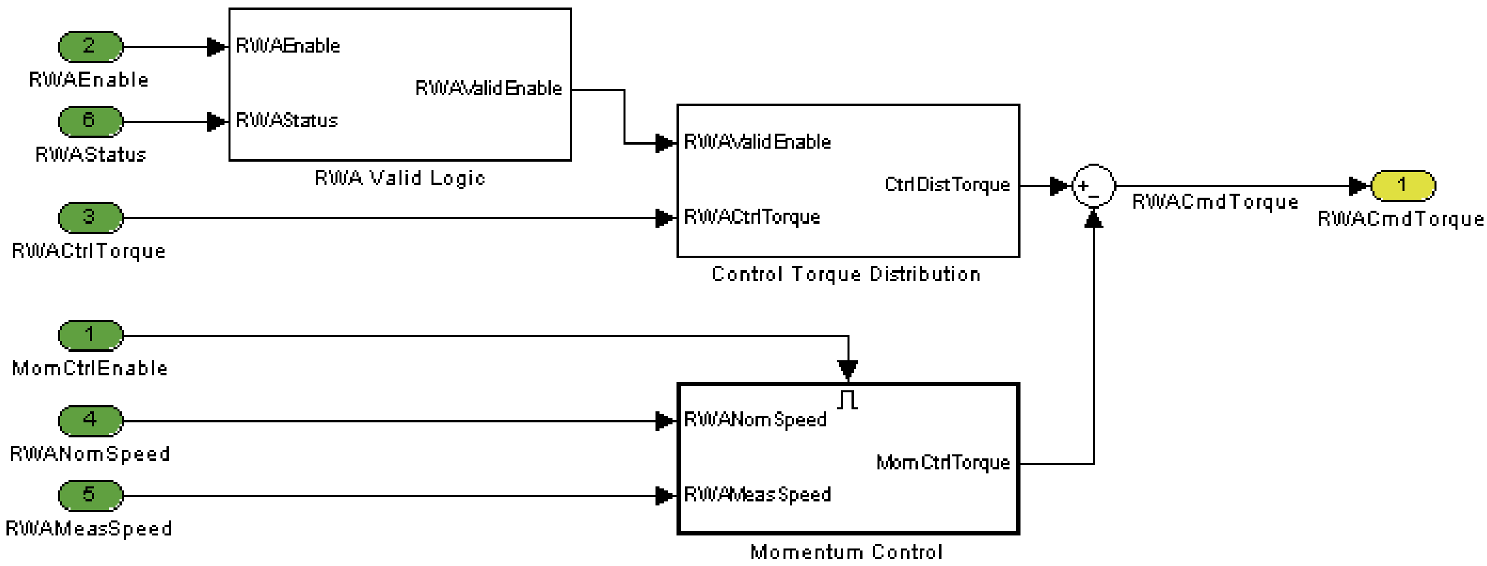
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.25.1. Description

```
Handles the momentum managament of the reaction wheels in the RWA.
```

Description from system mask help.

Figure 2-14. Momentum Management



2.25.1.1. Signals

Table 2-85. Momentum Management Signal Information

InputSignalNames	<MomCtrlEnable> <RWAEnable> <RWACtrlTorque> RWANomSpeed <RWAMeasSpeed> <RWAStatus>
OutputSignalNames	

2.25.2. Validation

Test001

2.26. Control Algorithms

Table 2-86. Control Algorithms System Information

Name	Control Algorithms
Depth	4
Type	block
Blocks	CtrlMode IntFineEnable EstimatedState RWNominalSpeed RWAMomentum SunAngles Enable Bus Selector Bus Selector1 Bus Selector2 Bus Selector3 Bus Selector4 Coarse/Fine Pointing Control Mode Action Magnetic Safe Standby Sum MTACtrlMagMoment RWACtrlTorque

Table 2-87. acs_documentation Information

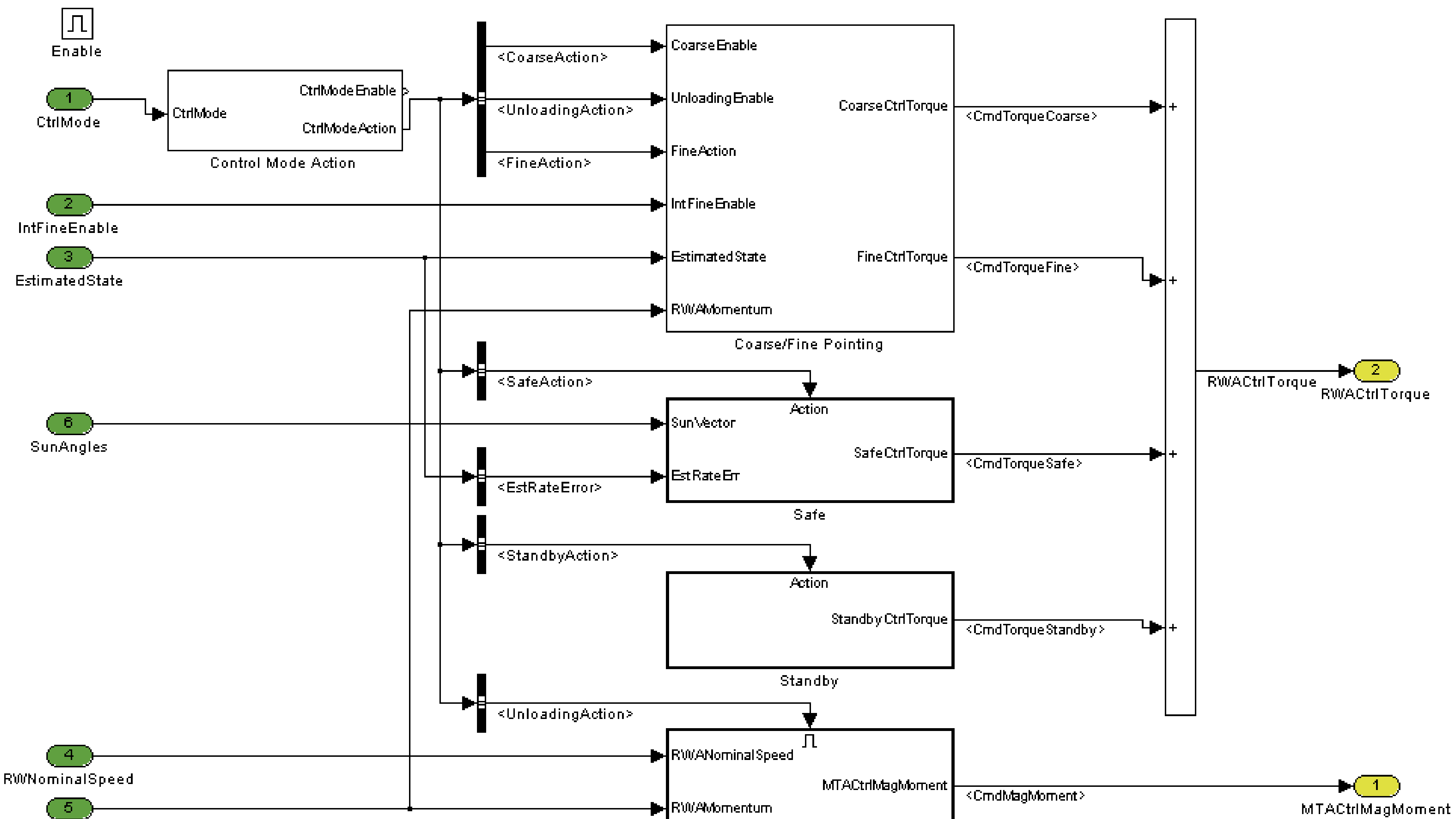
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.26.1. Description

```
Contains the control alogorithms operated by the ACS.
```

[Description from system mask help.](#)

Figure 2-15. Control Algorithms



2.26.1.1. Signals

Table 2-88. Control Algorithms Signal Information

InputSignalNames	<CtrlMode> <FineIntEnable> RWNominalSpeed <RWAMomentum> SunAngles <CtrlEnable>
OutputSignalNames	

2.26.2. Validation

Test001

2.27. Qmult1

Table 2-89. Qmult1 System Information

Name	Qmult1
Depth	4
Type	block
Blocks	q2 q1 Fcn3 Fcn4 Fcn5 Fcn6 Mux Mux3 q3

Table 2-90. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.27.1. Description

```
This block computes the product of two quaternions.
```

Description from system mask help.

2.27.1.1. Signals

Table 2-91. Qmult1 Signal Information

InputSignalNames	<AttitudeRef> EstAttitudeError
OutputSignalNames	EstAttitude

2.27.2. Validation

Test001

2.28. SetState ActuatorManagement

Table 2-92. SetState ActuatorManagement System Information

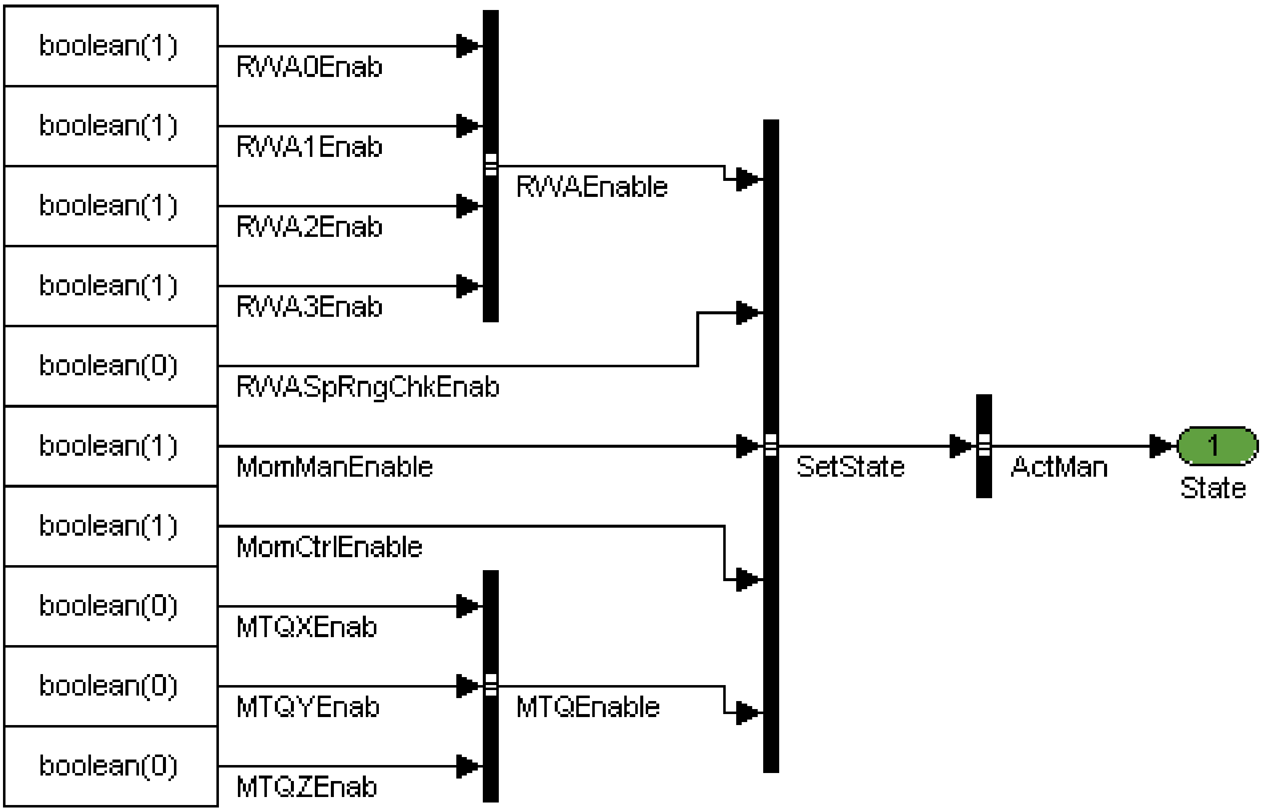
Name	SetState ActuatorManagement
Depth	4
Type	block
Blocks	Bus Creator1 Bus Creator10 Bus Creator2 Bus Creator9 Constant1 Constant10 Constant2 Constant3 Constant4 Constant5 Constant6 Constant7 Constant8 Constant9 State

Table 2-93. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.28.1. Description

Figure 2-16. SetState ActuatorManagement



2.28.1.1. Signals

Table 2-94. SetState ActuatorManagement Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-95. Output Signal Information

Name	ActMan
ParentBlock	acs_documentation/Rømer ACS/RuleLayer/Commander/SetState ActuatorManagement/Bus Creator10
Description	

2.29. SetState Control

Table 2-96. SetState Control System Information

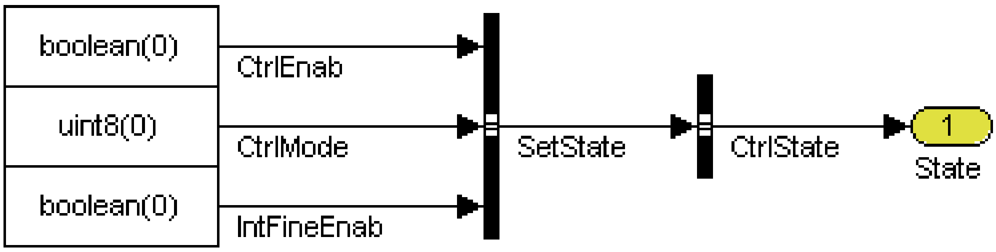
Name	SetState Control
Depth	4
Type	block
Blocks	Bus Creator10 Bus Creator8 Constant1 Constant2 Constant3 State

Table 2-97. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.29.1. Description

Figure 2-17. SetState Control



2.29.1.1. Signals

Table 2-98. SetState Control Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-99. Output Signal Information

Name	CtrlState
ParentBlock	acs_documentation/Rømer ACS/RuleLayer/Commander/SetState Control/Bus Creator10
Description	

2.30. SetState FaultDetection

Table 2-100. SetState FaultDetection System Information

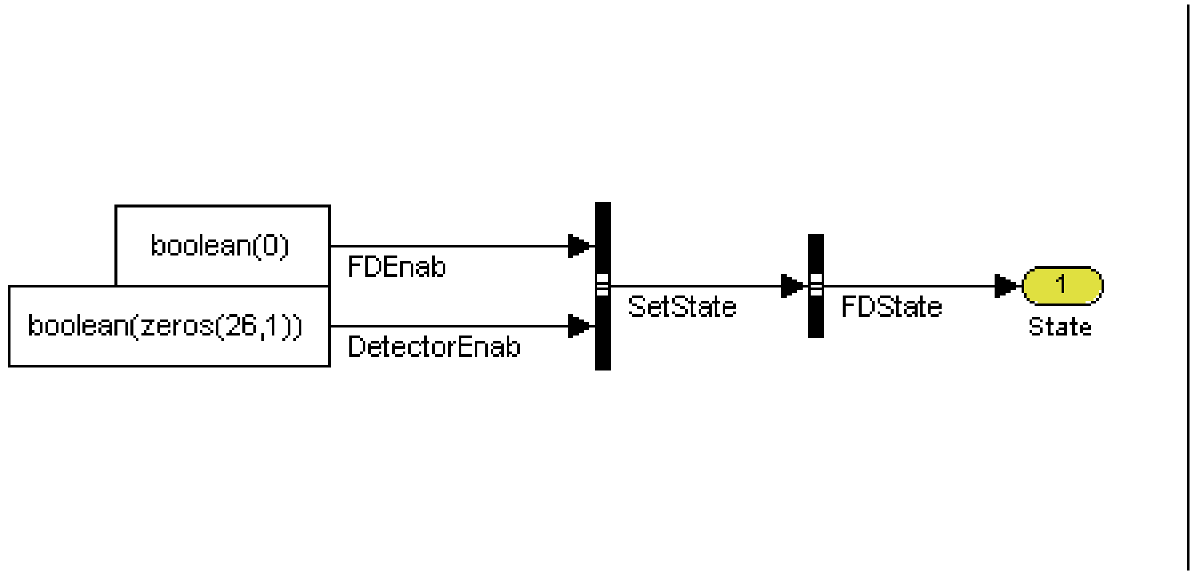
Name	SetState FaultDetection
Depth	4
Type	block
Blocks	Bus Creator10 Bus Creator9 Constant4 Constant5 State

Table 2-101. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.30.1. Description

Figure 2-18. SetState FaultDetection



2.30.1.1. Signals

Table 2-102. SetState FaultDetection Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-103. Output Signal Information

Name	FDState
ParentBlock	acs_documentation/Rømer ACS/RuleLayer/Commander/SetState FaultDetection/Bus Creator10
Description	

2.31. SetState Guidance

Table 2-104. SetState Guidance System Information

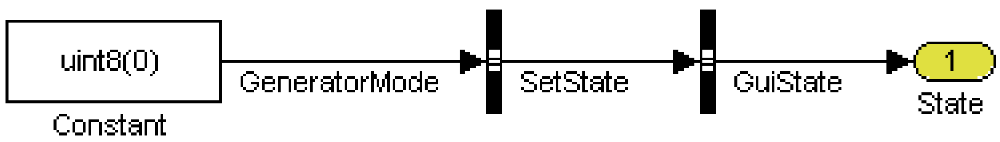
Name	SetState Guidance
Depth	4
Type	block
Blocks	Bus Creator1 Bus Creator10 Constant State

Table 2-105. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.31.1. Description

Figure 2-19. SetState Guidance



2.31.1.1. Signals

Table 2-106. SetState Guidance Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-107. Output Signal Information

Name	GuiState
ParentBlock	acs_documentation/Rømer ACS/RuleLayer/Commander/SetState Guidance/Bus Creator1
Description	

2.32. SetState Navigation

Table 2-108. SetState Navigation System Information

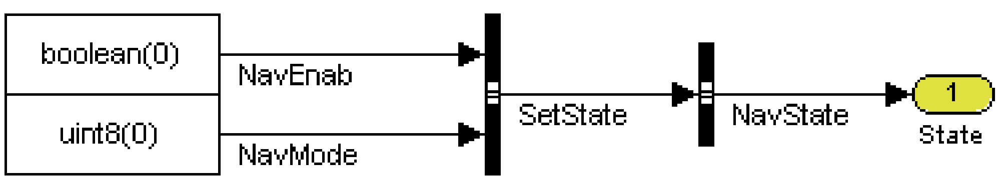
Name	SetState Navigation
Depth	4
Type	block
Blocks	Bus Creator10 Bus Creator5 Constant7 Constant8 State

Table 2-109. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.32.1. Description

Figure 2-20. SetState Navigation



2.32.1.1. Signals

Table 2-110. SetState Navigation Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-111. Output Signal Information

Name	NavState
ParentBlock	acs_documentation/Rømer ACS/RuleLayer/Commander/SetState Navigation/Bus Creator10
Description	

2.33. SetState SensorManagement

Table 2-112. SetState SensorManagement System Information

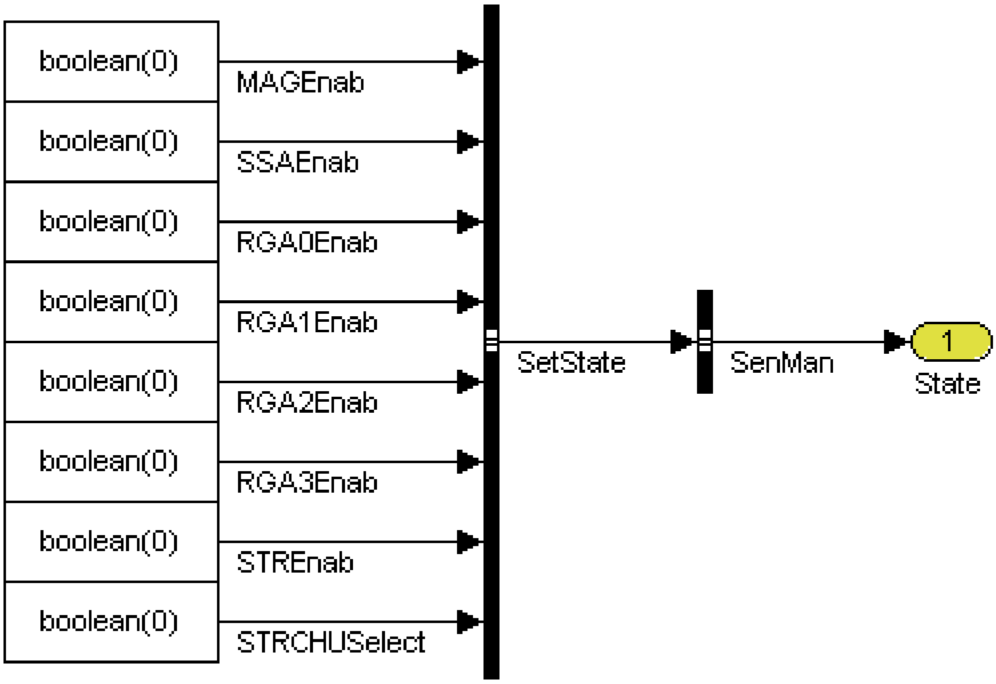
Name	SetState SensorManagement
Depth	4
Type	block
Blocks	Bus Creator10 Bus Creator9 Constant1 Constant2 Constant3 Constant4 Constant5 Constant6 Constant7 Constant8 State

Table 2-113. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.33.1. Description

Figure 2-21. SetState SensorManagement



2.33.1.1. Signals

Table 2-114. SetState SensorManagement Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-115. Output Signal Information

Name	SenMan
ParentBlock	acs_documentation/Rømer ACS/RuleLayer/Commander/SetState SensorManagement/Bus Creator10
Description	

2.34. AttitudeDeterminatorSelect

Table 2-116. AttitudeDeterminatorSelect System Information

Name	AttitudeDeterminatorSelect
Depth	4
Type	block
Blocks	Bus Creator Constant AttitudeDeterminatorSelect (B02)

Table 2-117. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.34.1. Description

Figure 2-22. AttitudeDeterminatorSelect

2.34.1.1. Signals

Table 2-118. AttitudeDeterminatorSelect Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-119. Output Signal Information

Name	AttitudeDeterminatorSelect (B02)
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Command Interface/AttitudeDeterminatorSelect/Bus Creator
Description	

2.35. AutonomyOnOff

Table 2-120. AutonomyOnOff System Information

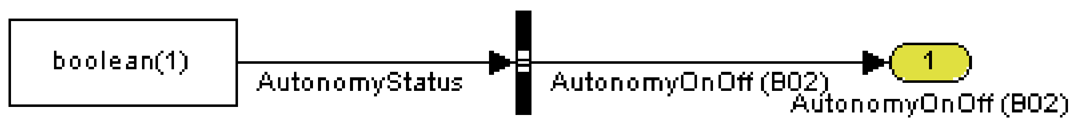
Name	AutonomyOnOff
Depth	4
Type	block
Blocks	Bus Creator Constant AutonomyOnOff (B02)

Table 2-121. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.35.1. Description

Figure 2-23. AutonomyOnOff



2.35.1.1. Signals

Table 2-122. AutonomyOnOff Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-123. Output Signal Information

Name	AutonomyOnOff (B02)
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Command Interface/AutonomyOnOff/Bus Creator
Description	

2.36. Enable

Table 2-124. Enable System Information

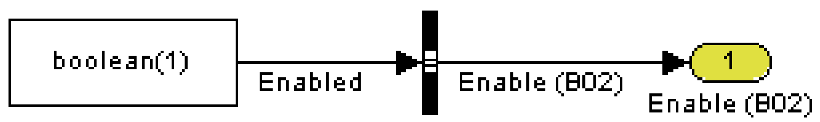
Name	Enable
Depth	4
Type	block
Blocks	Bus Creator Constant Enable (B02)

Table 2-125. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.36.1. Description

Figure 2-24. Enable



2.36.1.1. Signals

Table 2-126. Enable Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-127. Output Signal Information

Name	Enable (B02)
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Command Interface/Enable/Bus Creator
Description	

2.37. ModeSelect

Table 2-128. ModeSelect System Information

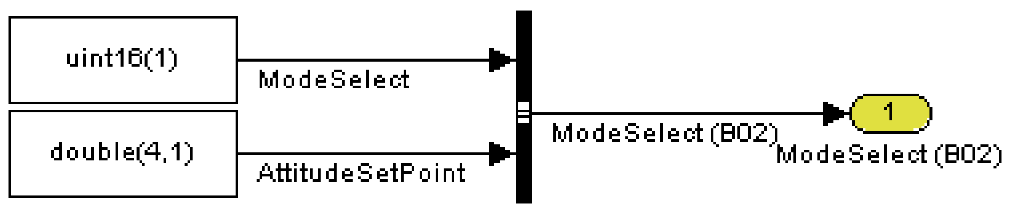
Name	ModeSelect
Depth	4
Type	block
Blocks	Bus Creator5 Constant Constant1 ModeSelect (B02)

Table 2-129. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.37.1. Description

Figure 2-25. ModeSelect



2.37.1.1. Signals

Table 2-130. ModeSelect Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-131. Output Signal Information

Name	ModeSelect (B02)
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Command Interface/ModeSelect/Bus Creator5
Description	Sets a reference point for UTC time with respect to onboard time. This allows Navigation to calculate reference vectors. Sets the orbital elements for the onboard orbit propagation.

2.38. PayLoadStarMessage

Table 2-132. PayLoadStarMessage System Information

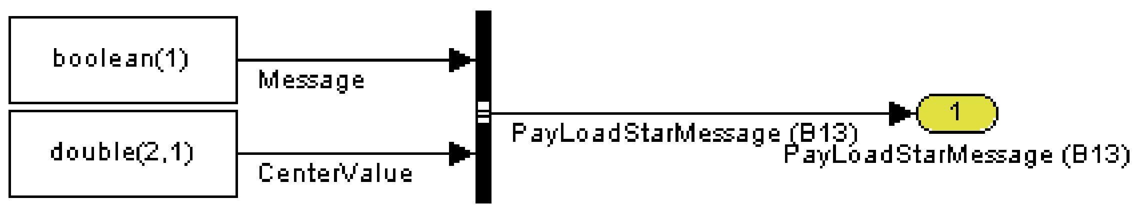
Name	PayLoadStarMessage
Depth	4
Type	block
Blocks	Bus Creator5 Constant Constant1 PayLoadStarMessage (B13)

Table 2-133. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.38.1. Description

Figure 2-26. PayLoadStarMessage



2.38.1.1. Signals

Table 2-134. PayLoadStarMessage Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-135. Output Signal Information

Name	PayLoadStarMessage (B13)
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Command Interface/PayLoadStarMessage/Bus Creator5
Description	Sets a reference point for UTC time with respect to onboard time. This allows Navigation to calculate reference vectors. Sets the orbital elements for the onboard orbit propagation.

2.39. SetAttitudeSetPoint

Table 2-136. SetAttitudeSetPoint System Information

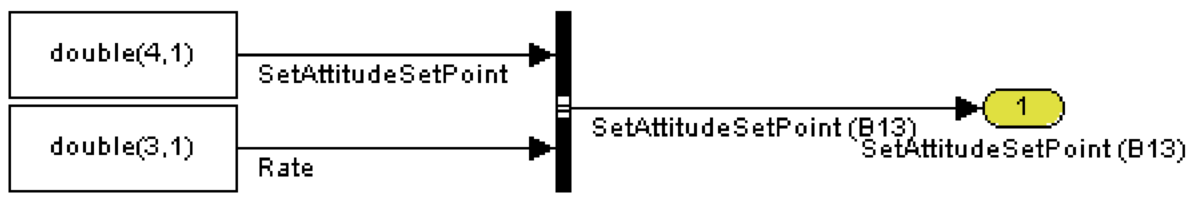
Name	SetAttitudeSetPoint
Depth	4
Type	block
Blocks	Bus Creator Constant Constant1 SetAttitudeSetPoint (B13)

Table 2-137. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.39.1. Description

Figure 2-27. SetAttitudeSetPoint



2.39.1.1. Signals

Table 2-138. SetAttitudeSetPoint Signal Information

InputSignalNames	

OutputSignalNames	

Table 2-139. Output Signal Information

Name	SetAttitudeSetPoint (B13)
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Command Interface/SetAttitudeSetPoint/Bus Creator
Description	

2.40. SetOrbitalElements

Table 2-140. SetOrbitalElements System Information

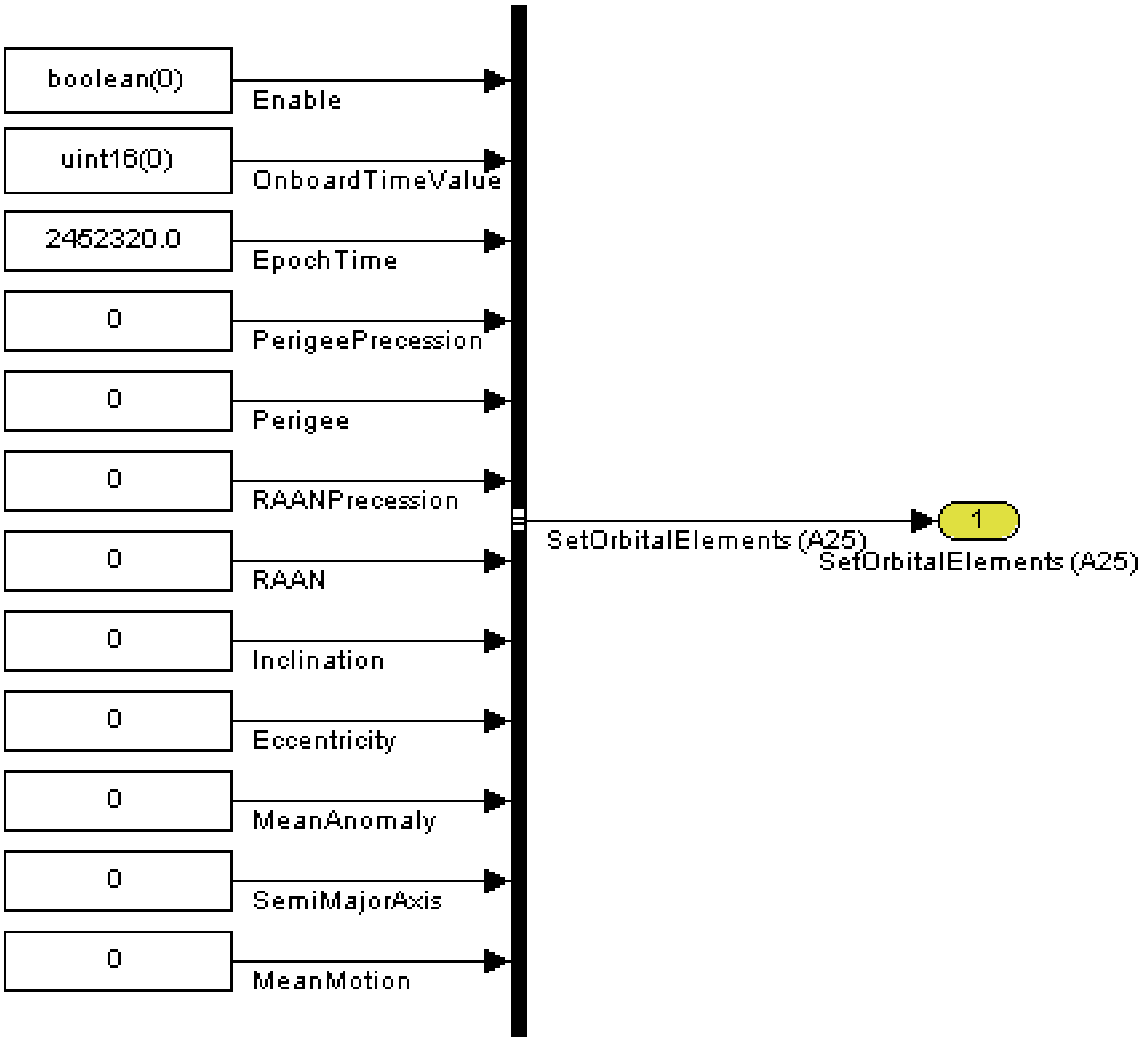
Name	SetOrbitalElements
Depth	4
Type	block
Blocks	Bus Creator5 Constant1 Constant10 Constant11 Constant12 Constant2 Constant3 Constant4 Constant5 Constant6 Constant7 Constant8 Constant9 SetOrbitalElements (A25)

Table 2-141. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.40.1. Description

Figure 2-28. SetOrbitalElements



2.40.1.1. Signals

Table 2-142. SetOrbitalElements Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-143. Output Signal Information

Name	SetOrbitalElements (A25)
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Command Interface/SetOrbitalElements /Bus Creator5
Description	Sets a reference point for UTC time with respect to onboard time. This allows Navigation to calculate reference vectors. Sets the orbital elements for the onboard orbit propagation.

2.41. SystemStateVector

Table 2-144. SystemStateVector System Information

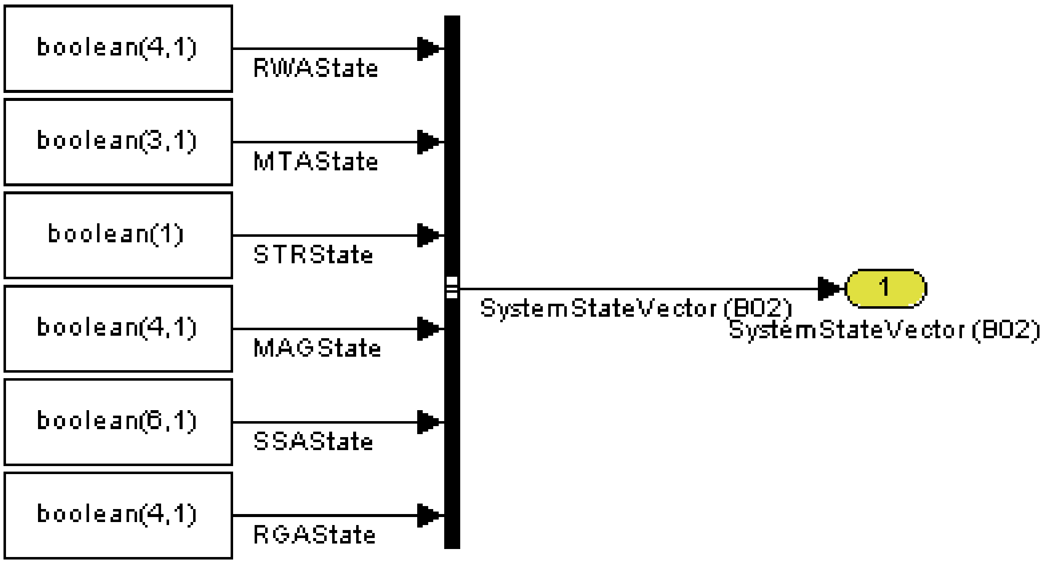
Name	SystemStateVector
Depth	4
Type	block
Blocks	Bus Creator Constant Constant1 Constant2 Constant3 Constant4 Constant5 SystemStateVector (B02)

Table 2-145. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.41.1. Description

Figure 2-29. SystemStateVector



2.41.1.1. Signals

Table 2-146. SystemStateVector Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-147. Output Signal Information

Name	SystemStateVector (B02)
ParentBlock	acs_documentation/Rømer ACS/SystemInterfaceLayer/Command Interface/SystemStateVector/Bus Creator
Description	

2.42. MTA Actuator Report

Table 2-148. MTA Actuator Report System Information

Name	MTA Actuator Report
Depth	5
Type	block
Blocks	MTACurrent Bus Selector1 Constant11 Constant2 Constant4 Constant6 Constant7 Constant8 Ground1 Ground2 Ground4 Ground5 Ground6 Ground7 Matrix Gain ActReport.Amtqx2rsr Matrix Gain ActReport.Amtqy2rsr Matrix Gain ActReport.Amtqz2rsr Matrix Gain ActReport.Arsr2scb Mux Mux1 Mux2 Product1 Product2

	Product3 Sum MTAMagMomentApp

Table 2-149. acs_documentation Information

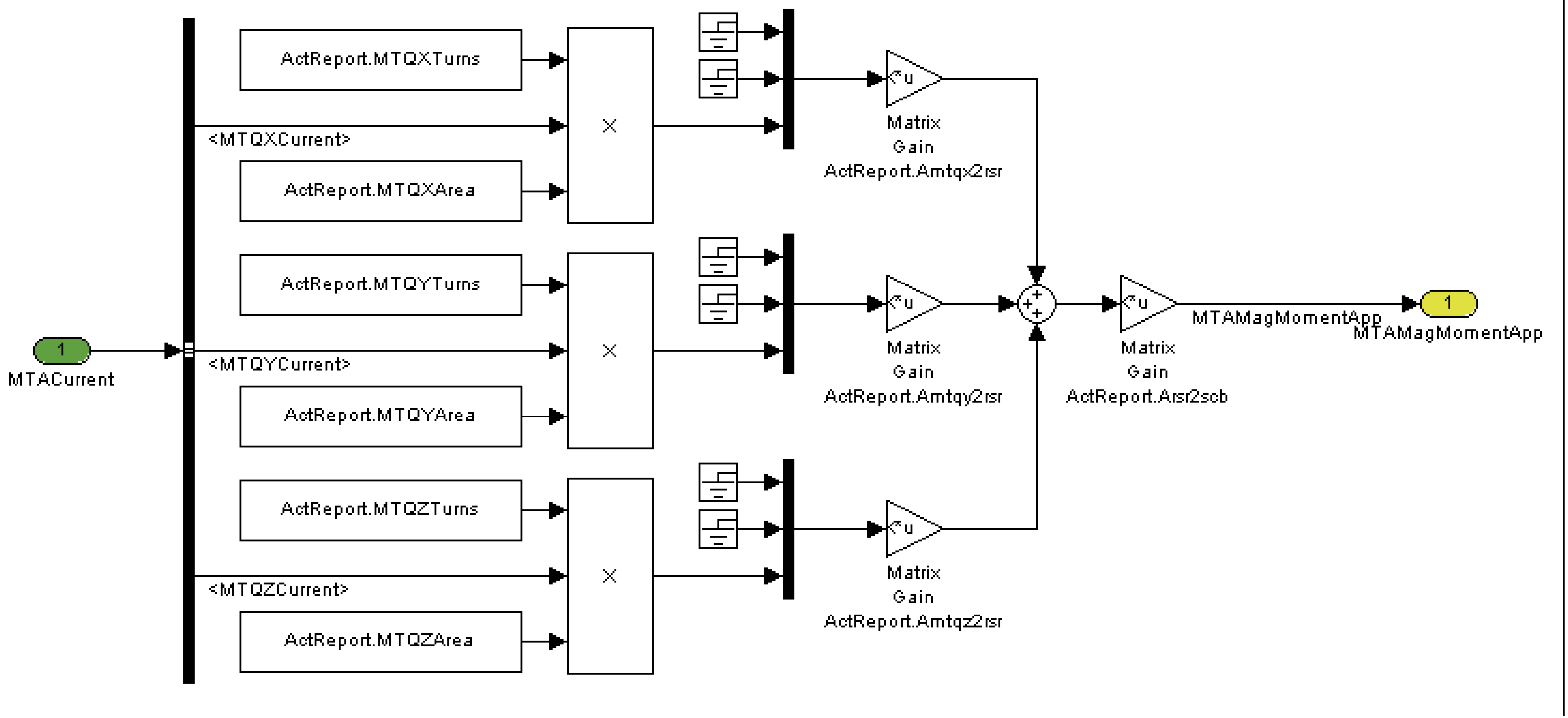
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.42.1. Description

```
Handles the generation of the MTA actuator report.
```

Description from system mask help.

Figure 2-30. MTA Actuator Report



2.42.1.1. Signals

Table 2-150. MTA Actuator Report Signal Information

InputSignalNames	<MTACurrent>
OutputSignalNames	

2.42.2. Validation

Test001

2.43. RWA Actuator Report

Table 2-151. RWA Actuator Report System Information

Name	RWA Actuator Report
Depth	5
Type	block
Blocks	RWAMeasTorque RWAMeasSpeed Gain RPM -> rad/s Matrix Gain ActReport.Arwrz2scb Matrix Gain ActReport.Arwrz2scb Matrix Gain ActReport.RWAInertia RWATorqueApp RWAMomentum

Table 2-152. acs_documentation Information

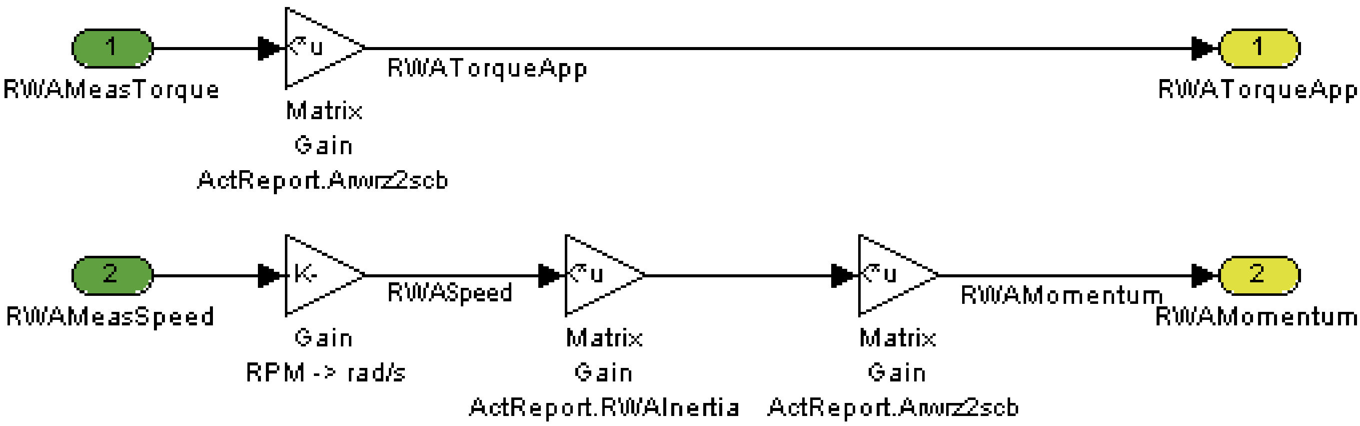
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.43.1. Description

```
Handles the generation of the RWA actuator report.
```

Description from system mask help.

Figure 2-31. RWA Actuator Report



2.43.1.1. Signals

Table 2-153. RWA Actuator Report Signal Information

InputSignalNames	<RWAMeasTorque> <RWAMeasSpeed>
OutputSignalNames	

2.43.2. Validation

Test001

2.44. MTA Command Current Normalization

Table 2-154. MTA Command Current Normalization System Information

Name	MTA Command Current Normalization
Depth	5
Type	block
Blocks	MTACmdCurrent Abs Bus Creator Demux If If Action Subsystem 1 If Action Subsystem 2 Matrix Gain MTACmdNorm.CmdCurrentMaxInv MinMax Product Sum Terminator MTACurrentCmdN

Table 2-155. acs_documentation Information

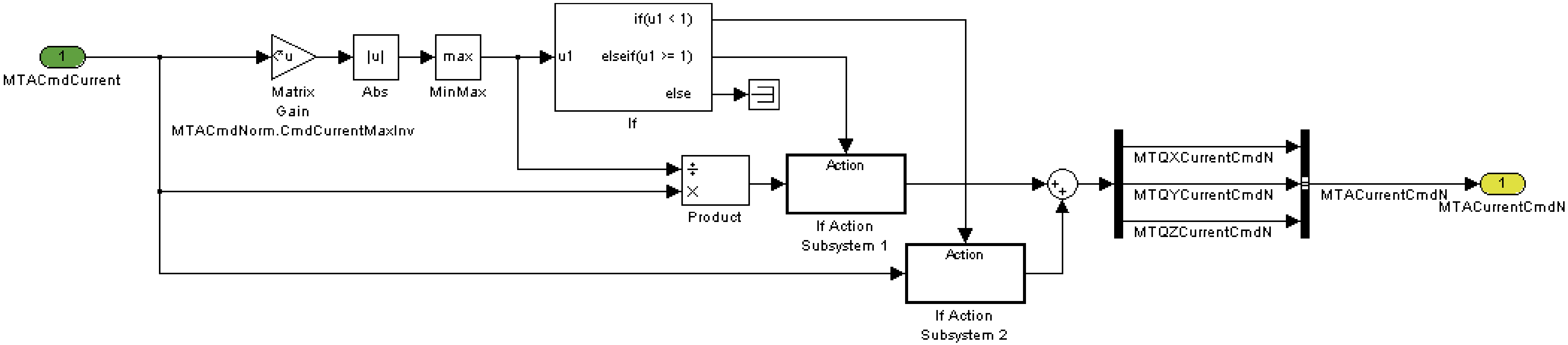
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.44.1. Description

```
Handles the normalization of command current to the magnetic torquers in the MTA.
```

Description from system mask help.

Figure 2-32. MTA Command Current Normalization



2.44.1.1. Signals

Table 2-156. MTA Command Current Normalization Signal Information

InputSignalNames	
OutputSignalNames	

2.44.2. Validation

Test001

2.45. RWA Command Torque Normalization

Table 2-157. RWA Command Torque Normalization System Information

Name	RWA Command Torque Normalization
Depth	5
Type	block
Blocks	RWACmdTorque Abs Bus Creator Demux If If Action Subsystem 1 If Action Subsystem 2 Matrix Gain RWACmdNorm.CmdTorqueMaxInv MinMax Product Sum Terminator RWATorqueCmdN

Table 2-158. acs_documentation Information

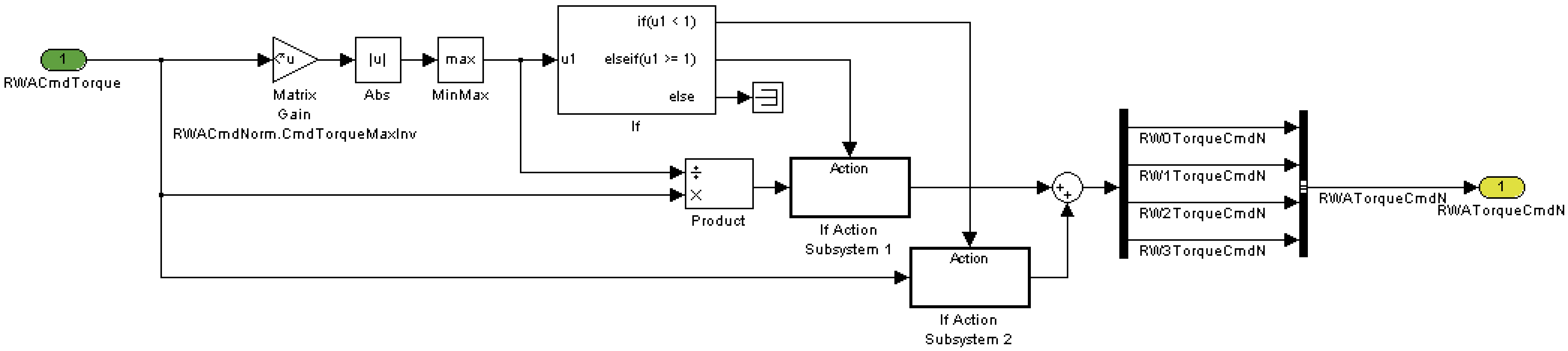
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.45.1. Description

```
Handles the normalization of command torque to the reaction wheels in the RWA.
```

Description from system mask help.

Figure 2-33. RWA Command Torque Normalization



2.45.1.1. Signals

Table 2-159. RWA Command Torque Normalization Signal Information

InputSignalNames	
OutputSignalNames	

2.45.2. Validation

Test001

2.46. Actuator Command Range Check

Table 2-160. Actuator Command Range Check System Information

Name	Actuator Command Range Check
Depth	5
Type	block
Blocks	MTACurrent RWAMeasTorque MTACurrentCmdN RWATorqueCmdN MTACurrentCmdNRngChkEnable RWATorqueCmdNChkEnable Ground1 Ground2 Ground3 Ground4 Terminator1 Terminator2 Terminator3 Terminator4 Terminator5 Terminator6 MTACurrentCmdNRngError RWATorqueCmdNRngError MTA RWA

Table 2-161. acs_documentation Information

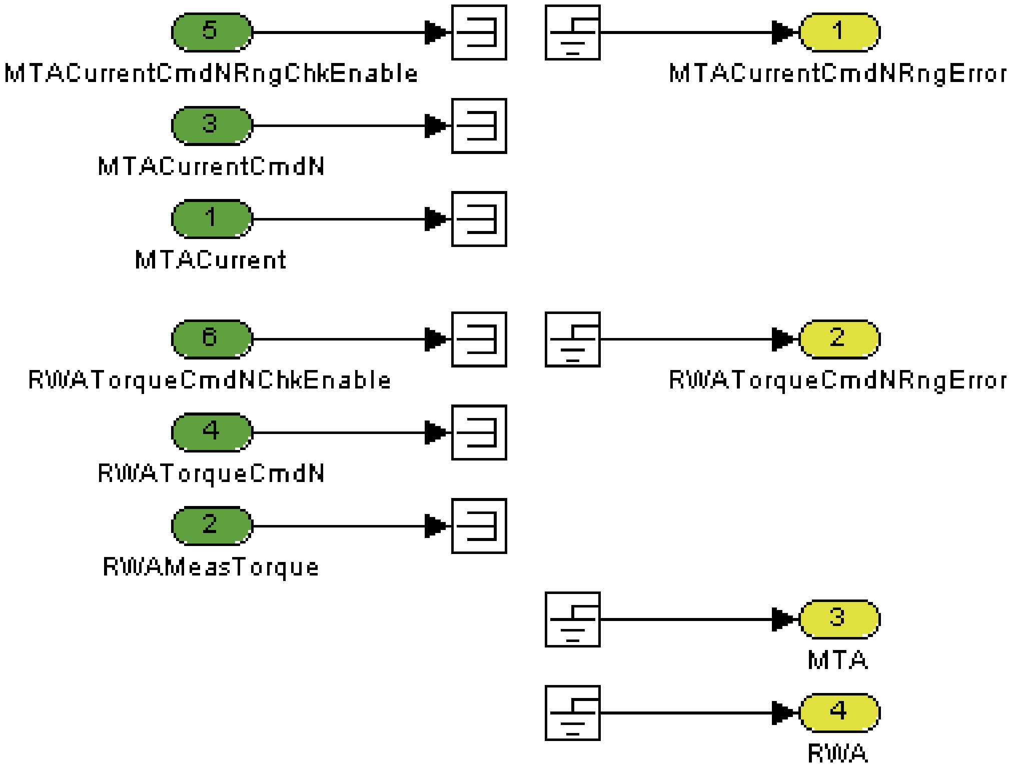
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.46.1. Description

This block performs range check on report data from actuators.

Description from system mask help.

Figure 2-34. Actuator Command Range Check



2.46.1.1. Signals

Table 2-162. Actuator Command Range Check Signal Information

InputSignalNames	<MTACurrent> <RWAMeasTorque> <MTACurrentCmdN> <RWATorqueCmdN> <MTACurrentCmdNRngChkEnable> <RWATorqueCmdNRngChkEnable>

OutputSignalNames	

2.46.2. Validation

Test001

2.47. Actuator Outlier Check

Table 2-163. Actuator Outlier Check System Information

Name	Actuator Outlier Check
Depth	5
Type	block
Blocks	MTAVoltage MTACurrent RWAMeasTorque RWAMeasSpeed MTAVoltageOutlChkEnable MTACurrentOutlChkEnable RWATorqueOutlChkEnable RWASpeedOutlChkEnable Ground Ground1 Ground2 Ground3 Terminator Terminator1 Terminator2 Terminator3 Terminator4 Terminator5 Terminator6 Terminator7 MTAVoltageOutlError MTACurrentOutlError RWATorqueOutlError RWASpeedOutlError

Table 2-164. acs_documentation Information

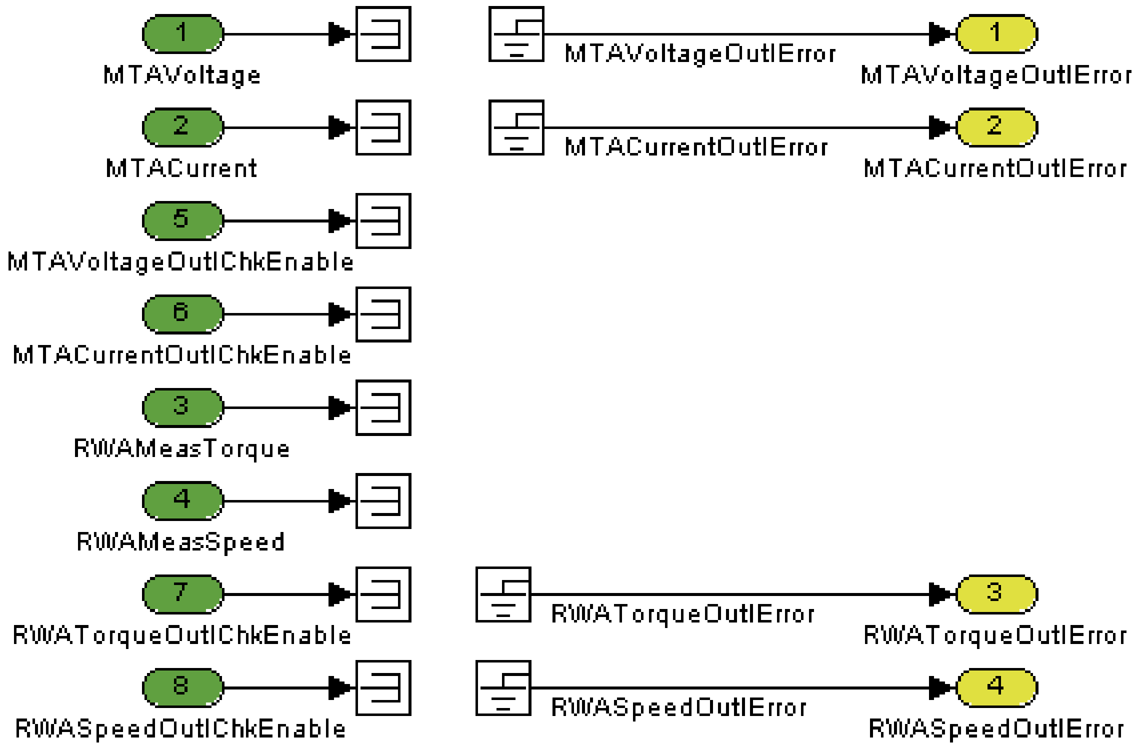
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.47.1. Description

```
This block performs range check on report data from actuators.
```

[Description from system mask help.](#)

Figure 2-35. Actuator Outlier Check



2.47.1.1. Signals

Table 2-165. Actuator Outlier Check Signal Information

InputSignalNames	<MTAVoltage> <MTACurrent> <RWAMeasTorque> <RWAMeasSpeed> <MTAVoltageOutlChkEnable> <MTACurrentOutlChkEnable> <RWATorqueOutlChkEnable> <RWASpeedOutlChkEnable>
OutputSignalNames	

2.47.2. Validation

Test001

2.48. Actuator Range Check

Table 2-166. Actuator Range Check System Information

Name	Actuator Range Check
Depth	5
Type	block
Blocks	MTAVoltage MTACurrent RWAMeasTorque RWAMeasSpeed MTAVoltageRngChkEnable MTACurrentRngChkEnable RWATorqueRngChkEnable RWASpeedRngChkEnable MTA Range Check RWA Range Check MTAVoltageRngError MTACurrentRngError RWATorqueRngError RWASpeedRngError

Table 2-167. acs_documentation Information

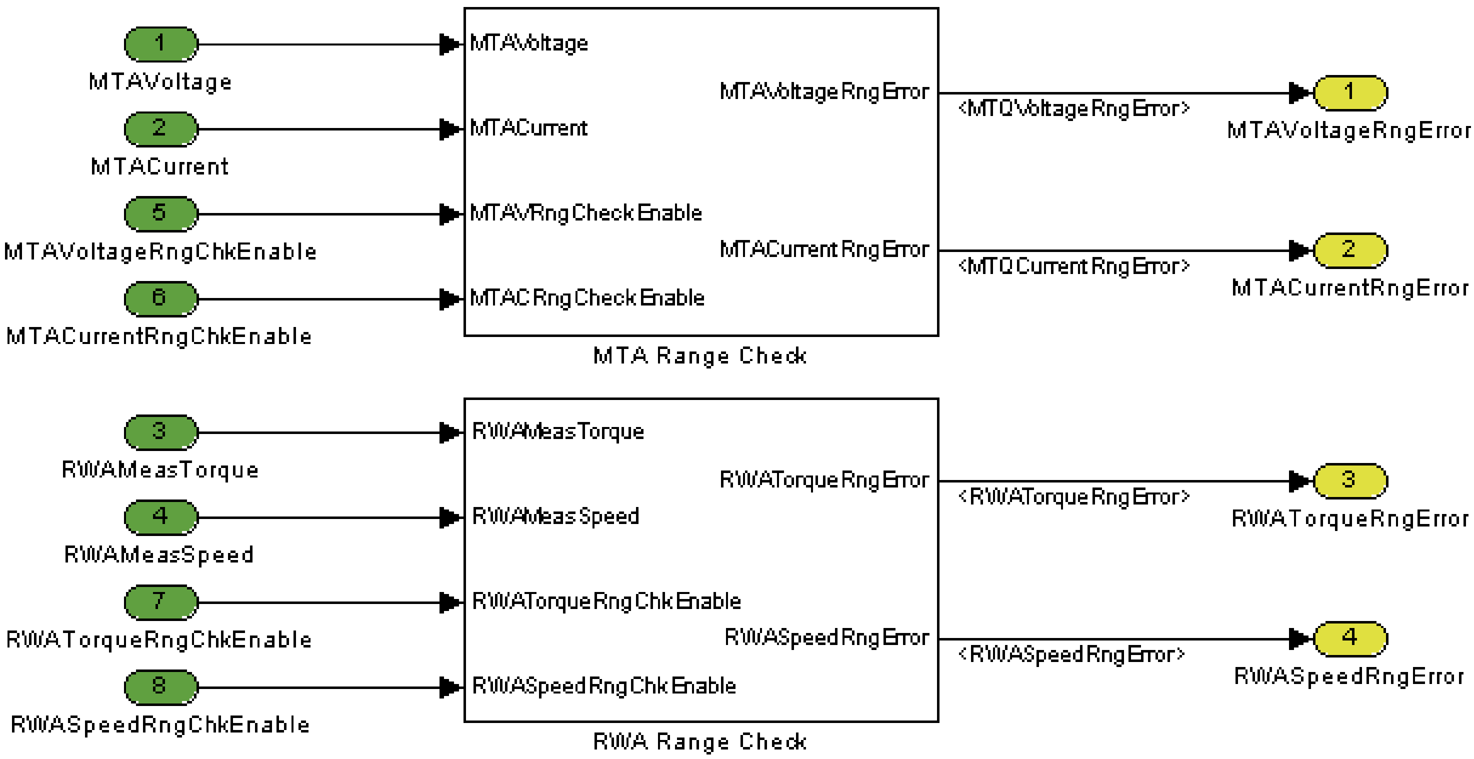
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.48.1. Description

```
Handles range check on report data from actuators.
```

Description from system mask help.

Figure 2-36. Actuator Range Check



2.48.1.1. Signals

Table 2-168. Actuator Range Check Signal Information

InputSignalNames	<MTAVoltage> <MTACurrent> <RWAMeasTorque> <RWAMeasSpeed> <MTAVoltageRngChkEnable> <MTACurrentRngChkEnable> <RWATorqueRngChkEnable> <RWASpeedRngChkEnable>
OutputSignalNames	

2.48.2. Validation

Test001

2.49. Actuator Status Check

Table 2-169. Actuator Status Check System Information

Name	Actuator Status Check
Depth	5
Type	block
Blocks	MTAStatus RWAStatus Bus Creator Bus Creator1 Demux Demux1 Logical Operator1 Logical Operator2 MTAError RWAError

Table 2-170. acs_documentation Information

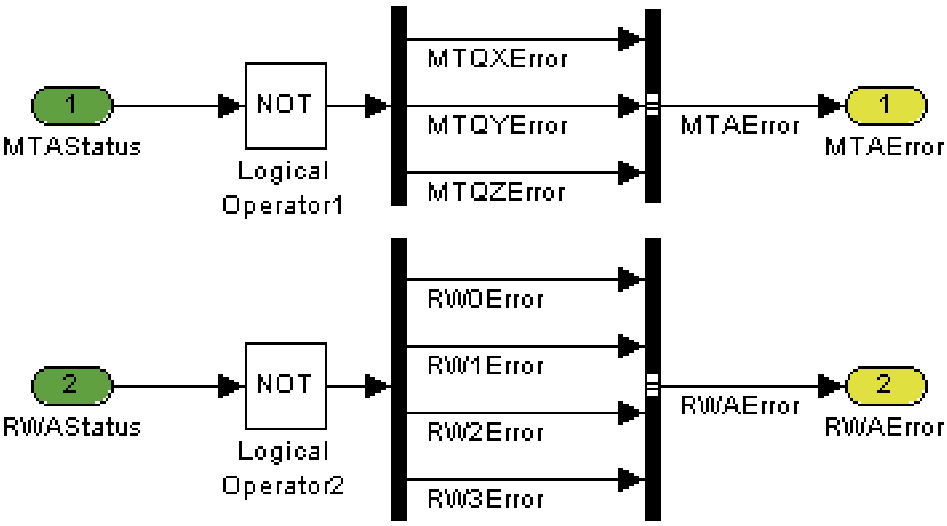
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.49.1. Description

```
This block performs range check on report data from actuators.
```

Description from system mask help.

Figure 2-37. Actuator Status Check



2.49.1.1. Signals

Table 2-171. Actuator Status Check Signal Information

InputSignalNames	<MTAStatus> <RWAStatus>
OutputSignalNames	

2.49.2. Validation

Test001

2.50. Control Torque Distribution

Table 2-172. Control Torque Distribution System Information

Name	Control Torque Distribution
Depth	5
Type	block
Blocks	RWAValidEnable RWACtrlTorque Bus Selector Bus Selector1 Bus Selector2 Bus Selector3 Bus Selector4 Distribution (RW0 Not Valid) Distribution (RW1 Not Valid) Distribution (RW2 Not Valid) Distribution (RW3 Not Valid) Nominal Distribution (RWA Valid) Sum CtrlDistTorque

Table 2-173. acs_documentation Information

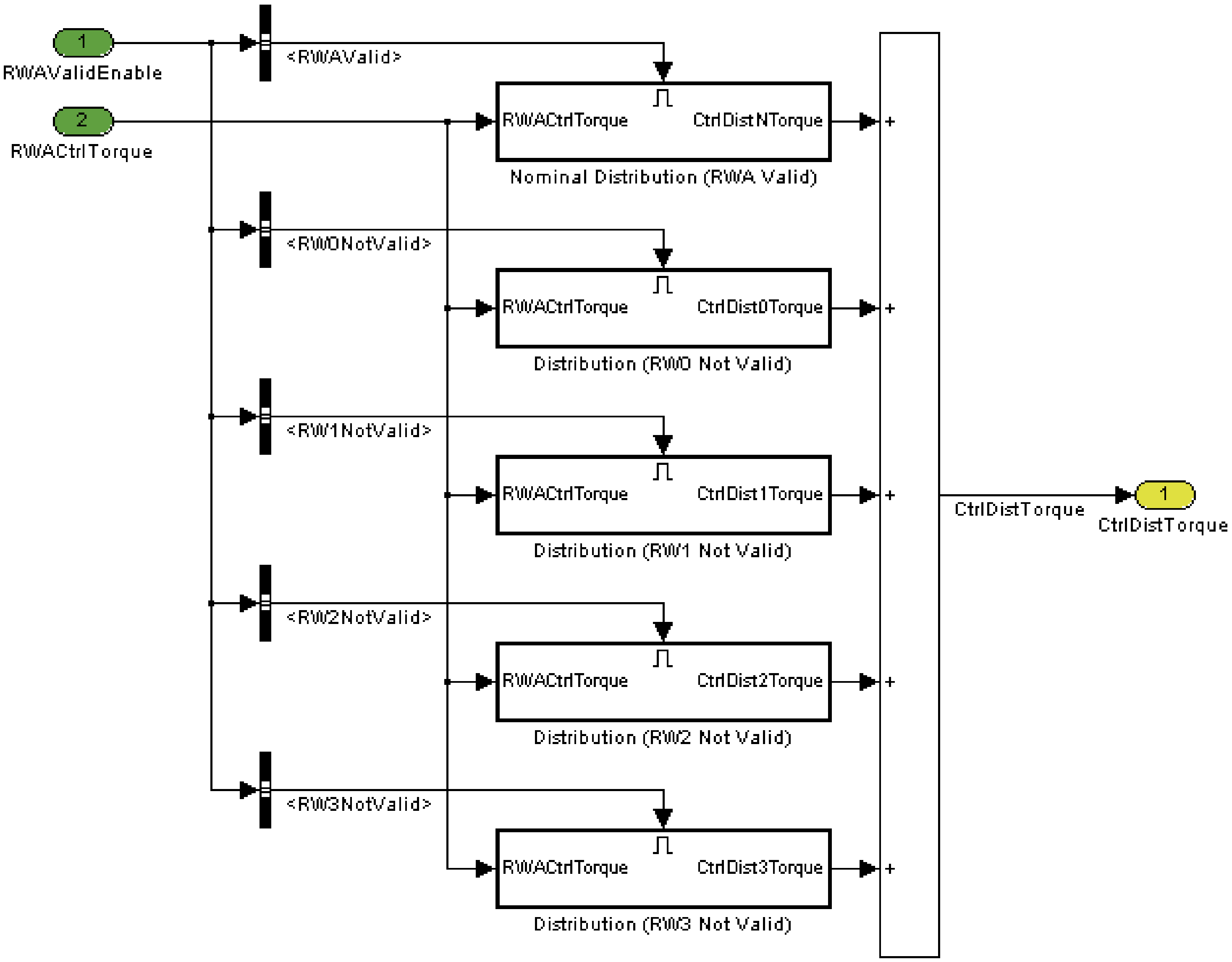
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.50.1. Description

```
Handles the distribution of the control torque among the operating reaction
wheels in the RWA.
```

Description from system mask help.

Figure 2-38. Control Torque Distribution



2.50.1.1. Signals

Table 2-174. Control Torque Distribution Signal Information

InputSignalNames	
OutputSignalNames	

2.50.2. Validation

[Test001](#)

2.51. Momentum Control

Table 2-175. Momentum Control System Information

Name	Momentum Control
Depth	5
Type	block
Blocks	RWANomSpeed RWAMeasSpeed Enable Bus Creator Demux1 Gain RPM -> rad/s Matrix Gain MomCtrl.E4CRCRWA Matrix Gain MomCtrl.GainMatrix Matrix Gain MomCtrl.RWAInertia1 Matrix Gain MomCtrl.RWAInertia2 Sum MomCtrlTorque

Table 2-176. acs_documentation Information

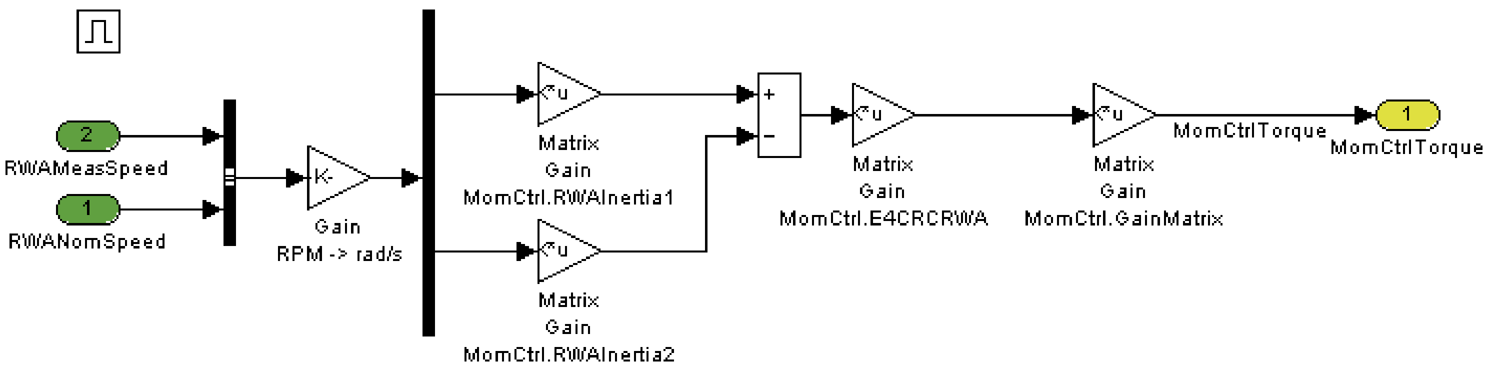
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.51.1. Description

```
Momentum Control of the reaction wheels in the RWA.
```

Description from system mask help.

Figure 2-39. Momentum Control



2.51.1.1. Signals

Table 2-177. Momentum Control Signal Information

InputSignalNames	
OutputSignalNames	

2.51.2. Validation

Test001

2.52. RWA Valid Logic

Table 2-178. RWA Valid Logic System Information

Name	RWA Valid Logic
Depth	5
Type	block
Blocks	RWAEnable RWAStatus Bus Creator Bus Selector Bus Selector1 If Logical Operator1 Logical Operator2 Logical Operator3 Logical Operator4 RW0 Not Valid RW1 Not Valid RW2 Not Valid RW3 Not Valid RWA Valid RWAValidEnable

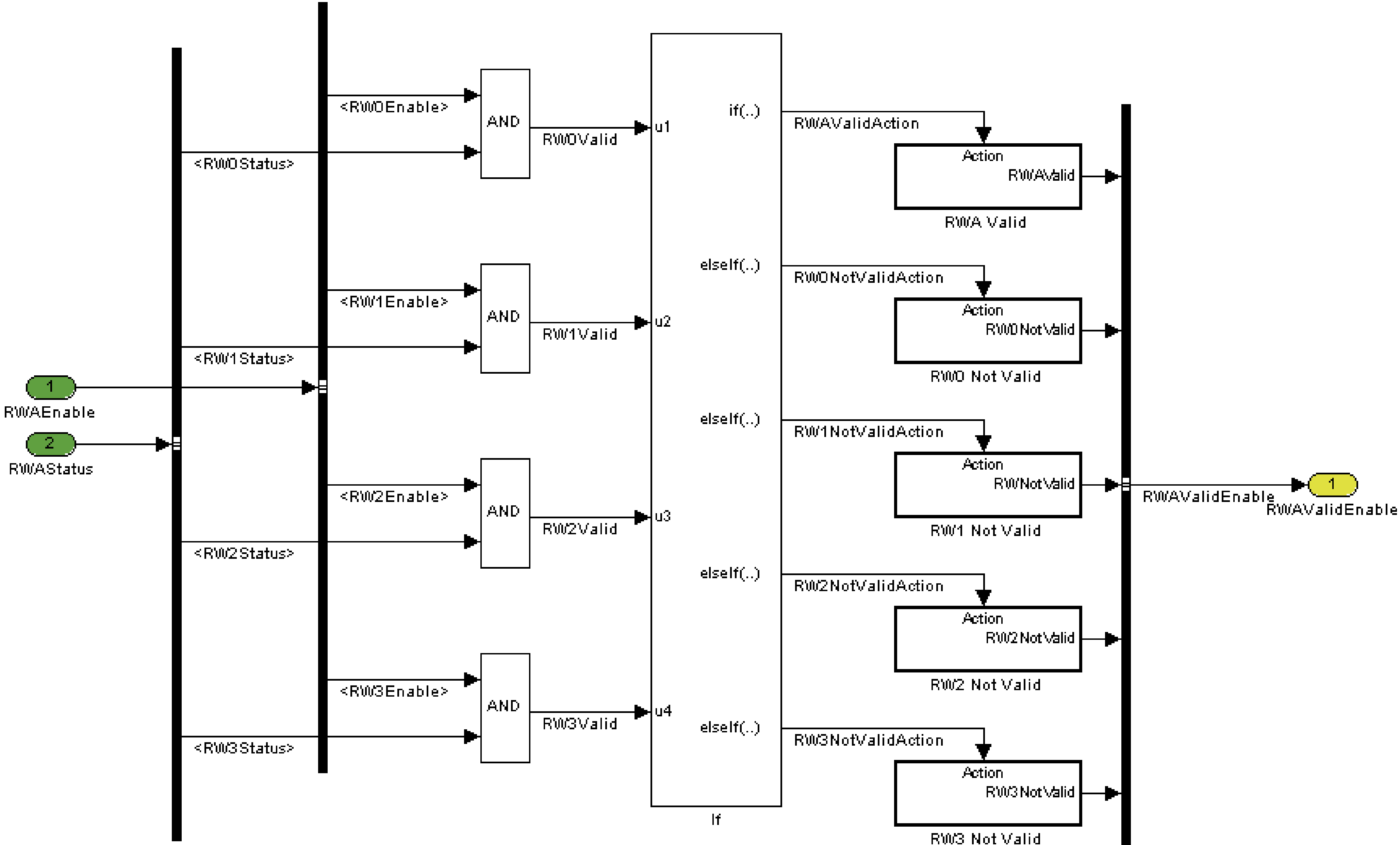
Table 2-179. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002

LastModifiedBy	tb

2.52.1. Description

Figure 2-40. RWA Valid Logic



2.52.1.1. Signals

Table 2-180. RWA Valid Logic Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-181. Input Signal Information

Name	<5335.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Momentum Management/RWA Valid Logic/ RWAEnable
Description	

Table 2-182. Input Signal Information

Name	<5358.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Momentum Management/RWA Valid Logic/ RWAStatus
Description	

Table 2-183. Output Signal Information

Name	RWAValidEnable
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Momentum Management/RWA Valid Logic/ Bus Creator

Description	

2.53. Coarse/Fine Pointing

Table 2-184. Coarse/Fine Pointing System Information

Name	Coarse/Fine Pointing
Depth	5
Type	block
Blocks	CoarseEnable UnloadingEnable FineAction IntFineEnable EstimatedState RWAMomentum Bus Selector2 Coarse Control Mode Enable Coarse Pointing Fine Pointing CoarseCtrlTorque FineCtrlTorque

Table 2-185. acs_documentation Information

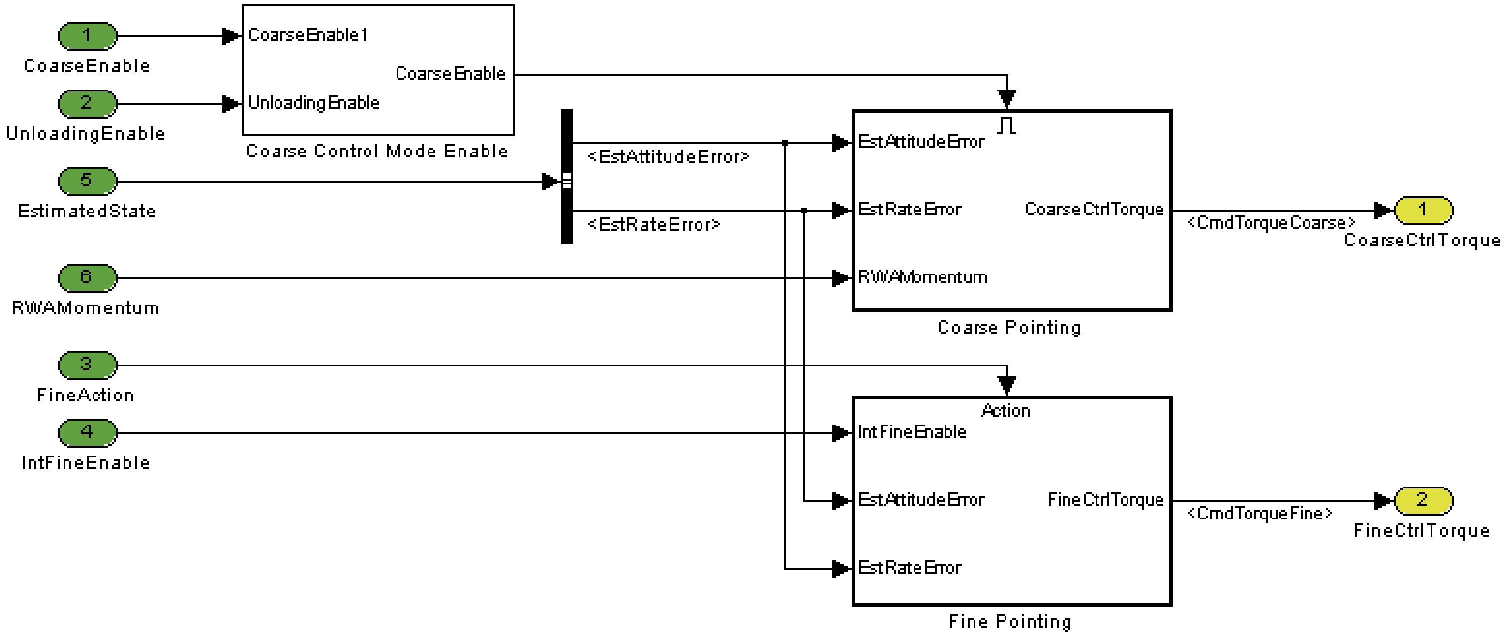
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.53.1. Description

```
Covers the Coarse Pointing and Fine Pointing control algorithms and the logic that enables the
Coarse Pointing control algorithm.
```

Description from system mask help.

Figure 2-41. Coarse/Fine Pointing



2.53.1.1. Signals

Table 2-186. Coarse/Fine Pointing Signal Information

InputSignalNames	<CoarseAction> <UnloadingAction> <FineAction>
OutputSignalNames	

2.53.2. Validation

Test001

2.54. Control Mode Action

Table 2-187. Control Mode Action System Information

Name	Control Mode Action
Depth	5
Type	block
Blocks	CtrlMode Action -> Enable Bus Creator Switch Case CtrlModeEnable CtrlModeAction

Table 2-188. acs_documentation Information

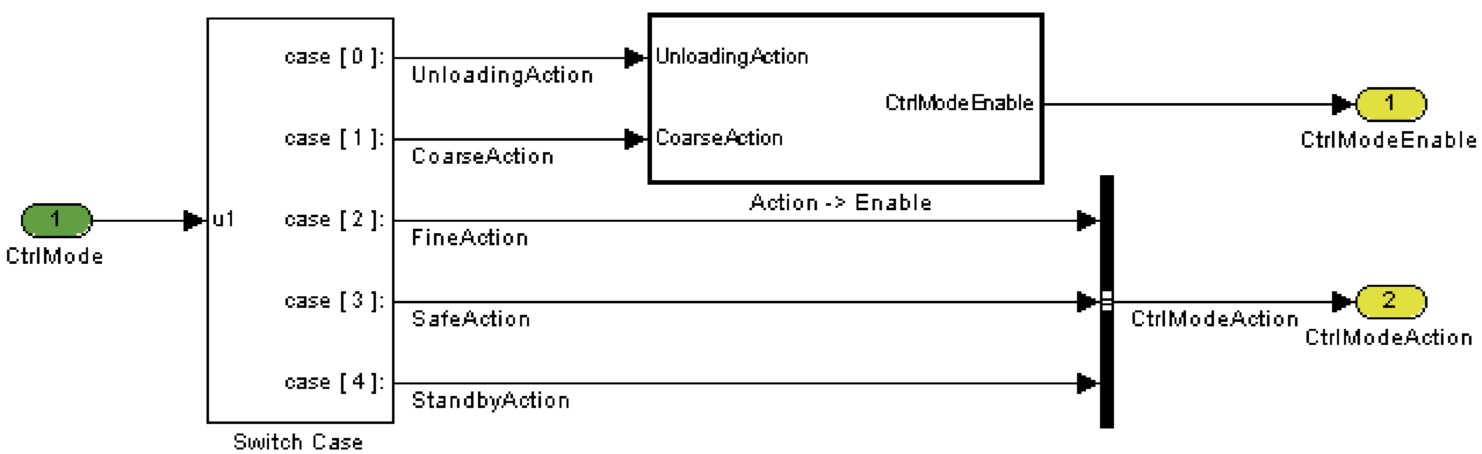
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.54.1. Description

```
Handles the enabling of the control modes as specified by the CtrlMode.
```

Description from system mask help.

Figure 2-42. Control Mode Action



2.54.1.1. Signals

Table 2-189. Control Mode Action Signal Information

InputSignalNames	
OutputSignalNames	

2.54.2. Validation

Test001

2.55. Magnetic

Table 2-190. Magnetic System Information

Name	Magnetic
Depth	5
Type	block
Blocks	RWANominalSpeed RWAMomentum Enable Constant Terminator Terminator1 MTACtrlMagMoment

Table 2-191. acs_documentation Information

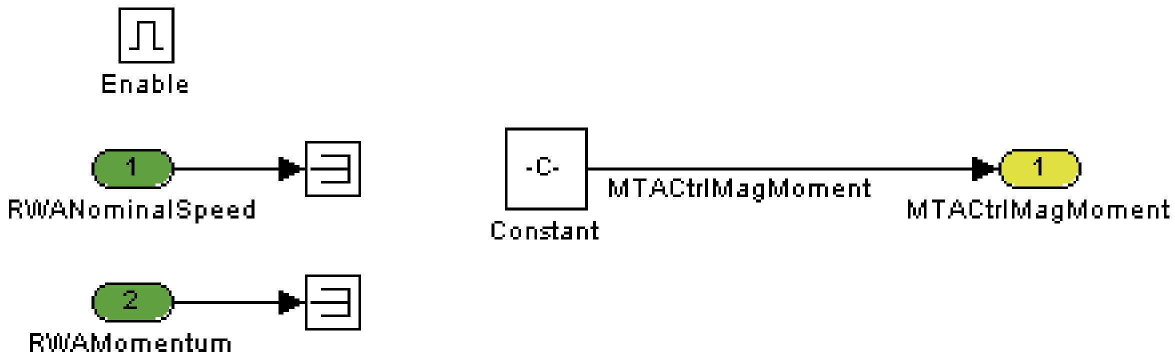
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.55.1. Description

```
Implements the Magnetic control algorithm used in the Unloading
control mode..
```

Description from system mask help.

Figure 2-43. Magnetic



2.55.1.1. Signals

Table 2-192. Magnetic Signal Information

InputSignalNames	<UnloadingAction>
OutputSignalNames	

2.55.2. Validation

Test001

2.56. Safe

Table 2-193. Safe System Information

Name	Safe
Depth	5
Type	block
Blocks	SunVector EstRateErr Action Port Constant Terminator Terminator1 SafeCtrlTorque

Table 2-194. acs_documentation Information

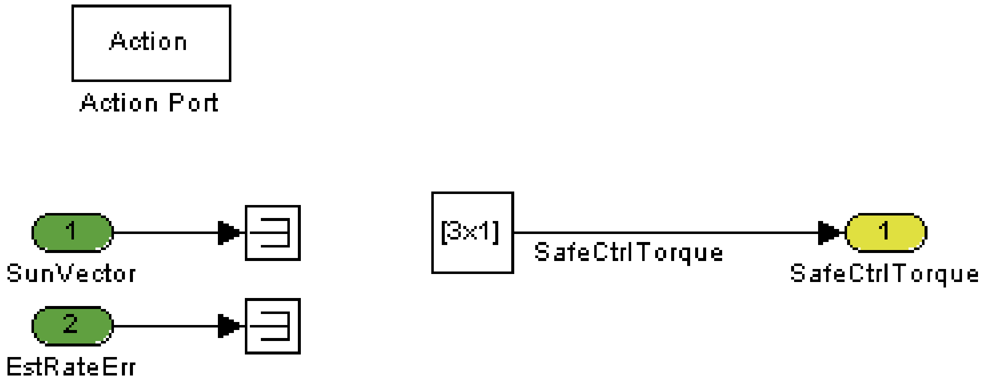
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.56.1. Description

```
Implements the Safe control algorithm used in the Safe control mode.
```

Description from system mask help.

Figure 2-44. Safe



2.56.1.1. Signals

Table 2-195. Safe Signal Information

InputSignalNames	<EstRateError> <SafeAction>
OutputSignalNames	

2.56.2. Validation

Test001

2.57. Standby

Table 2-196. Standby System Information

Name	Standby
Depth	5
Type	block
Blocks	Action Port Constant StandbyCtrlTorque

Table 2-197. acs_documentation Information

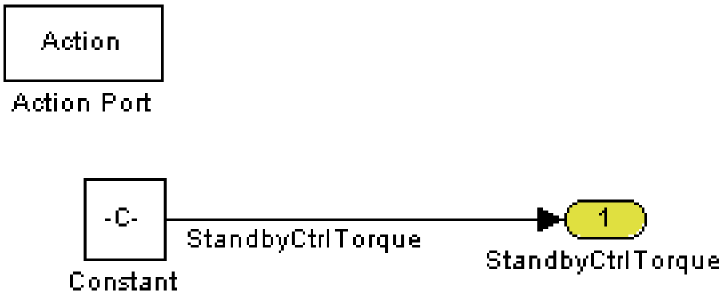
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.57.1. Description

```
Implements the algorithm used in the Standby control mode.
```

Description from system mask help.

Figure 2-45. Standby



2.57.1.1. Signals

Table 2-198. Standby Signal Information

InputSignalNames	<StandbyAction>
OutputSignalNames	

2.57.2. Validation

Test001

2.58. If Action Subsystem 1

Table 2-199. If Action Subsystem 1 System Information

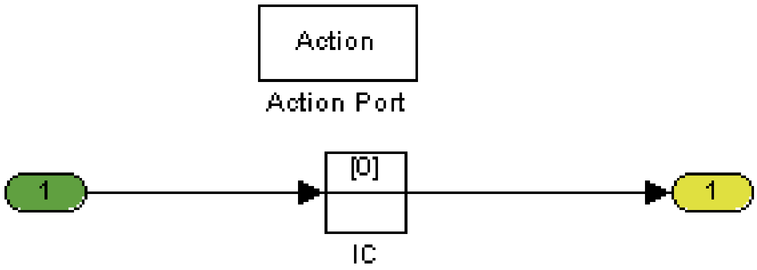
Name	If Action Subsystem 1
Depth	6
Type	block
Blocks	In 1 Action Port IC Out 1

Table 2-200. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.58.1. Description

Figure 2-46. If Action Subsystem 1



2.58.1.1. Signals

Table 2-201. If Action Subsystem 1 Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-202. Input Signal Information

Name	<5428.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Command Normalization/MTA Command Current Normalization/If Action Subsystem 1/In 1
Description	

Table 2-203. Output Signal Information

Name	<5430.0016>

ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Command Normalization/MTA Command Current Normalization/If Action Subsystem 1/IC
Description	

2.59. If Action Subsystem 2

Table 2-204. If Action Subsystem 2 System Information

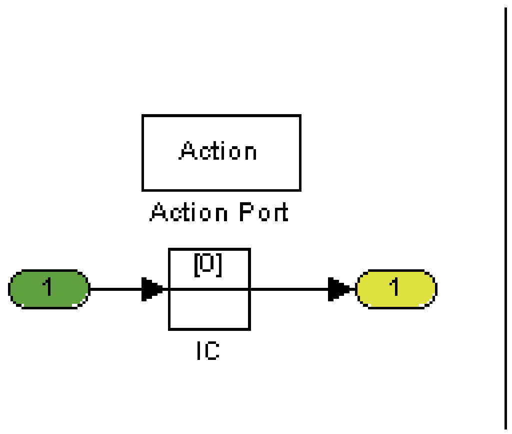
Name	If Action Subsystem 2
Depth	6
Type	block
Blocks	In 1 Action Port IC Out 1

Table 2-205. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.59.1. Description

Figure 2-47. If Action Subsystem 2



2.59.1.1. Signals

Table 2-206. If Action Subsystem 2 Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-207. Input Signal Information

Name	<5432.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Command Normalization/MTA Command Current Normalization/If Action Subsystem 2/In 1
Description	

Table 2-208. Output Signal Information

Name	<5434.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Command Normalization/MTA Command Current Normalization/If Action Subsystem 2/IC
Description	

2.60. If Action Subsystem 1

Table 2-209. If Action Subsystem 1 System Information

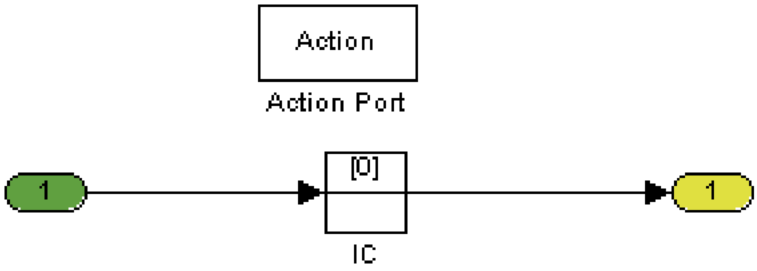
Name	If Action Subsystem 1
Depth	6
Type	block
Blocks	In 1 Action Port IC Out 1

Table 2-210. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.60.1. Description

Figure 2-48. If Action Subsystem 1



2.60.1.1. Signals

Table 2-211. If Action Subsystem 1 Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-212. Input Signal Information

Name	<5438.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Command Normalization/RWA Command Torque Normalization/If Action Subsystem 1/In 1
Description	

Table 2-213. Output Signal Information

Name	<5437.0016>

ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Command Normalization/RWA Command Torque Normalization/If Action Subsystem 1/IC
Description	

2.61. If Action Subsystem 2

Table 2-214. If Action Subsystem 2 System Information

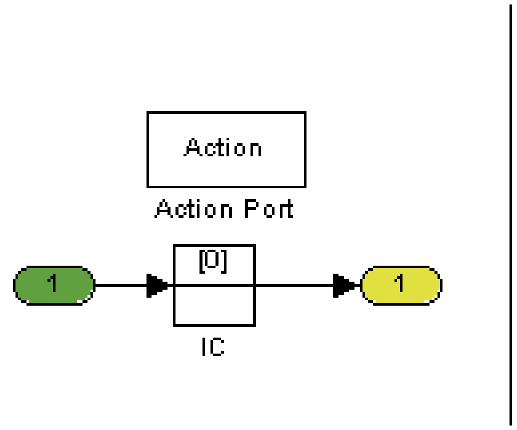
Name	If Action Subsystem 2
Depth	6
Type	block
Blocks	In 1 Action Port IC Out 1

Table 2-215. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.61.1. Description

Figure 2-49. If Action Subsystem 2



2.61.1.1. Signals

Table 2-216. If Action Subsystem 2 Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-217. Input Signal Information

Name	<5442.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Command Normalization/RWA Command Torque Normalization/If Action Subsystem 2/In 1
Description	

Table 2-218. Output Signal Information

Name	<5441.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Command Normalization/RWA Command Torque Normalization/If Action Subsystem 2/IC
Description	

2.62. MTA Range Check

Table 2-219. MTA Range Check System Information

Name	MTA Range Check
Depth	6
Type	block
Blocks	MTAVoltage MTACurrent MTAVRngCheckEnable MTACRngCheckEnable MTACurrent Range Check MTAVoltage Range Check MTAVoltageRngError MTACurrentRngError

Table 2-220. acs_documentation Information

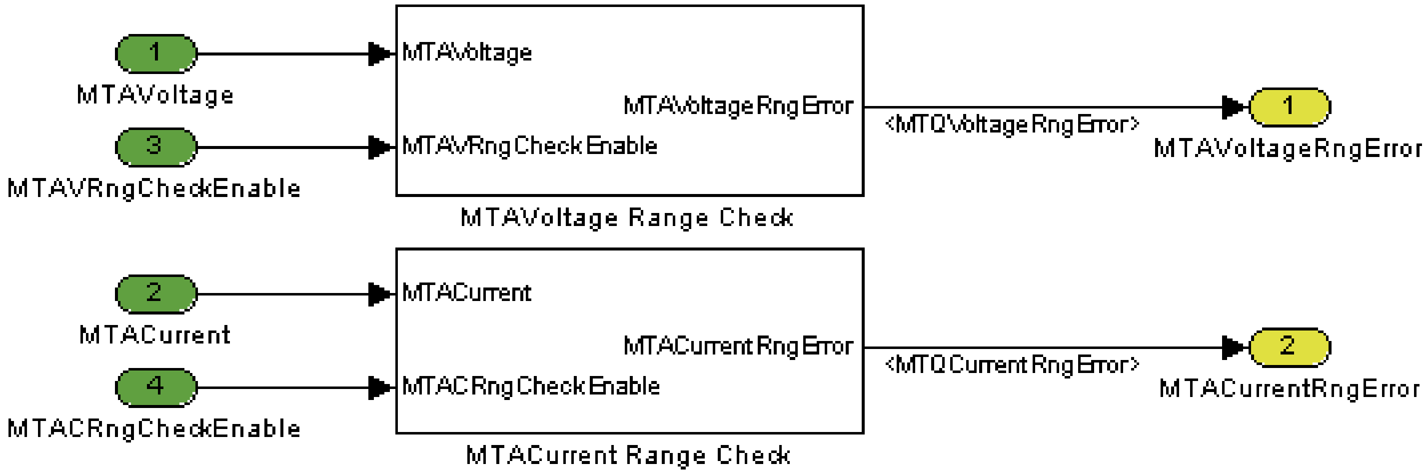
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.62.1. Description

```
This block performs range check on report data from the MTQ. Range
check is performed on reported voltages and currents.
```

Description from system mask help.

Figure 2-50. MTA Range Check



2.62.1.1. Signals

Table 2-221. MTA Range Check Signal Information

InputSignalNames	
OutputSignalNames	

2.62.2. Validation

Test001

2.63. RWA Range Check

Table 2-222. RWA Range Check System Information

Name	RWA Range Check
Depth	6
Type	block
Blocks	RWAMeasTorque RWAMeasSpeed RWATorqueRngChkEnable RWASpeedRngChkEnable RWAMeasSpeed Range Check RWAMeasTorque Range Check RWATorqueRngError RWASpeedRngError

Table 2-223. acs_documentation Information

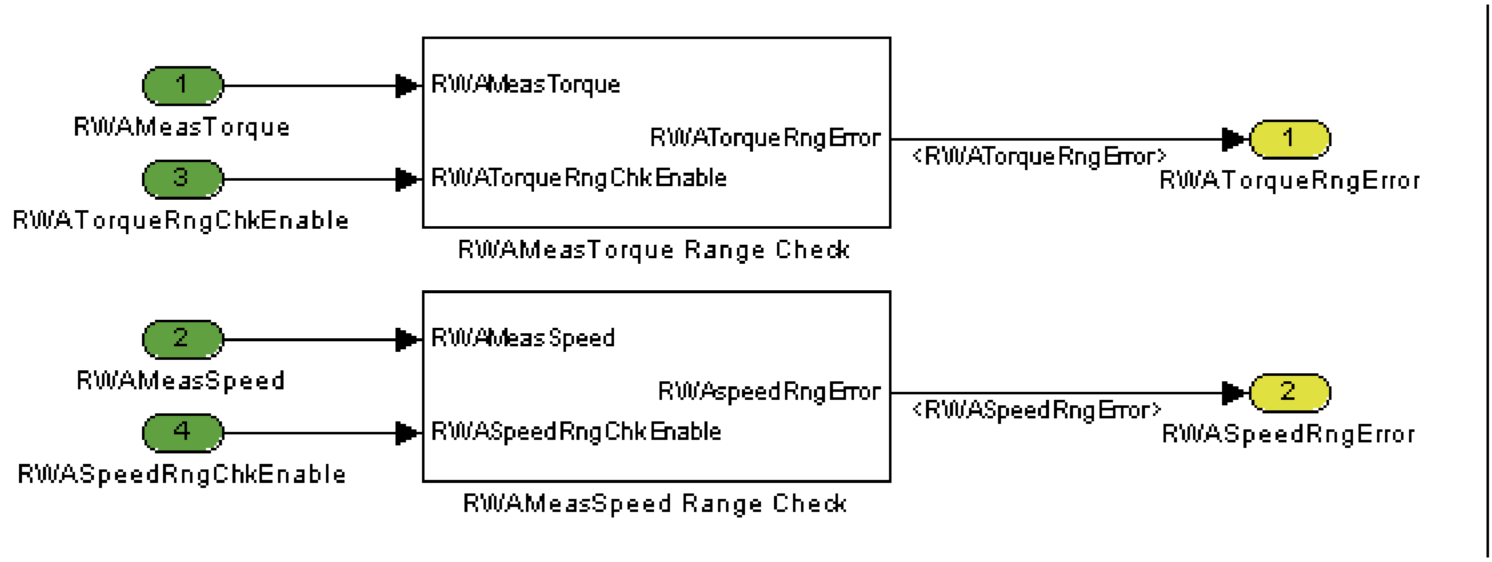
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.63.1. Description

```
This block performs range check on report data from the RWA. Range
check is performed on reported wheel torque and speed.
```

Description from system mask help.

Figure 2-51. RWA Range Check



2.63.1.1. Signals

Table 2-224. RWA Range Check Signal Information

InputSignalNames	
OutputSignalNames	

2.63.2. Validation

Test001

2.64. Distribution (RW0 Not Valid)

Table 2-225. Distribution (RW0 Not Valid) System Information

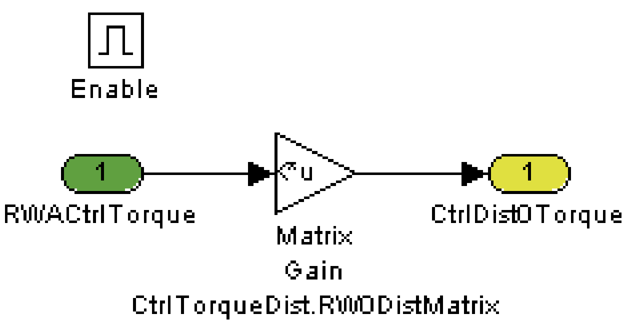
Name	Distribution (RW0 Not Valid)
Depth	6
Type	block
Blocks	RWACtrlTorque Enable Matrix Gain CtrlTorqueDist.RW0DistMatrix CtrlDist0Torque

Table 2-226. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.64.1. Description

Figure 2-52. Distribution (RW0 Not Valid)



2.64.1.1. Signals

Table 2-227. Distribution (RW0 Not Valid) Signal Information

InputSignalNames	<RW0NotValid>
OutputSignalNames	

Table 2-228. Input Signal Information

Name	<5464.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Momentum Management/Control Torque Distribution/Distribution (RW0 Not Valid)/RWACtrlTorque
Description	

Table 2-229. Output Signal Information

Name	<5466.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Momentum Management/Control Torque Distribution/Distribution (RW0 Not Valid)/Matrix Gain CtrlTorqueDist.RW0DistMatrix
Description	

2.65. Distribution (RW1 Not Valid)

Table 2-230. Distribution (RW1 Not Valid) System Information

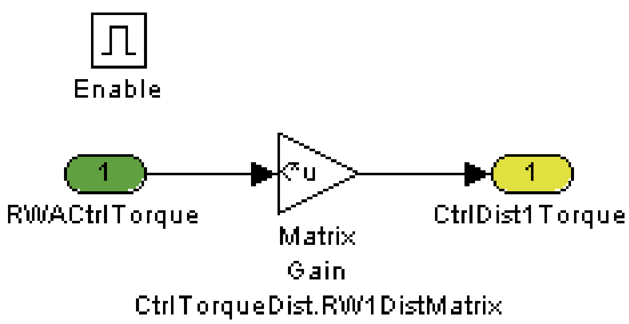
Name	Distribution (RW1 Not Valid)
Depth	6
Type	block
Blocks	RWACtrlTorque Enable Matrix Gain CtrlTorqueDist.RW1DistMatrix CtrlDist1Torque

Table 2-231. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.65.1. Description

Figure 2-53. Distribution (RW1 Not Valid)



2.65.1.1. Signals

Table 2-232. Distribution (RW1 Not Valid) Signal Information

InputSignalNames	<RW1NotValid>
OutputSignalNames	

Table 2-233. Input Signal Information

Name	<5468.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Momentum Management/Control Torque Distribution/Distribution (RW1 Not Valid)/RWACtrlTorque
Description	

Table 2-234. Output Signal Information

Name	<5470.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Momentum Management/Control Torque Distribution/Distribution (RW1 Not Valid)/Matrix Gain CtrlTorqueDist.RW1DistMatrix
Description	

2.66. Distribution (RW2 Not Valid)

Table 2-235. Distribution (RW2 Not Valid) System Information

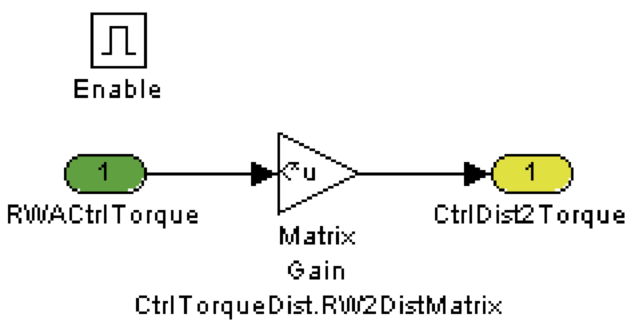
Name	Distribution (RW2 Not Valid)
Depth	6
Type	block
Blocks	RWACtrlTorque Enable Matrix Gain CtrlTorqueDist.RW2DistMatrix CtrlDist2Torque

Table 2-236. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.66.1. Description

Figure 2-54. Distribution (RW2 Not Valid)



2.66.1.1. Signals

Table 2-237. Distribution (RW2 Not Valid) Signal Information

InputSignalNames	<RW2NotValid>
OutputSignalNames	

Table 2-238. Input Signal Information

Name	<5472.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Momentum Management/Control Torque Distribution/Distribution (RW2 Not Valid)/RWACtrlTorque
Description	

Table 2-239. Output Signal Information

Name	<5474.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Momentum Management/Control Torque Distribution/Distribution (RW2 Not Valid)/Matrix Gain CtrlTorqueDist.RW2DistMatrix
Description	

2.67. Distribution (RW3 Not Valid)

Table 2-240. Distribution (RW3 Not Valid) System Information

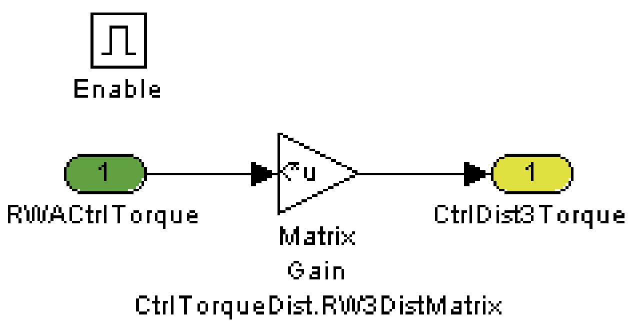
Name	Distribution (RW3 Not Valid)
Depth	6
Type	block
Blocks	RWACtrlTorque Enable Matrix Gain CtrlTorqueDist.RW3DistMatrix CtrlDist3Torque

Table 2-241. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.67.1. Description

Figure 2-55. Distribution (RW3 Not Valid)



2.67.1.1. Signals

Table 2-242. Distribution (RW3 Not Valid) Signal Information

InputSignalNames	<RW3NotValid>
OutputSignalNames	

Table 2-243. Input Signal Information

Name	<5476.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Momentum Management/Control Torque Distribution/Distribution (RW3 Not Valid)/RWACtrlTorque
Description	

Table 2-244. Output Signal Information

Name	<5478.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/Momentum Management/Control Torque Distribution/Distribution (RW3 Not Valid)/Matrix Gain CtrlTorqueDist.RW3DistMatrix
Description	

2.68. Nominal Distribution (RWA Valid)

Table 2-245. Nominal Distribution (RWA Valid) System Information

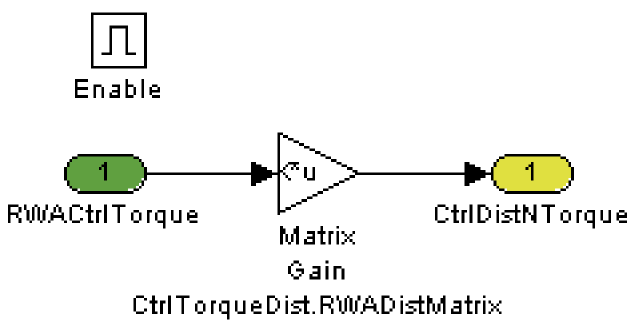
Name	Nominal Distribution (RWA Valid)
Depth	6
Type	block
Blocks	RWACtrlTorque Enable Matrix Gain CtrlTorqueDist.RWADistMatrix CtrlDistNTorque

Table 2-246. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.68.1. Description

Figure 2-56. Nominal Distribution (RWA Valid)



2.68.1.1. Signals

Table 2-247. Nominal Distribution (RWA Valid) Signal Information

InputSignalNames	<RWAValid>
OutputSignalNames	

Table 2-248. Input Signal Information

Name	<5480.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Momentum Management/Control Torque Distribution/Nominal Distribution (RWA Valid)/RWACtrlTorque
Description	

Table 2-249. Output Signal Information

Name	<5482.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Momentum Management/Control Torque Distribution/Nominal Distribution (RWA Valid)/Matrix Gain CtrlTorqueDist.RWADistMatrix
Description	

2.69. RW0 Not Valid

Table 2-250. RW0 Not Valid System Information

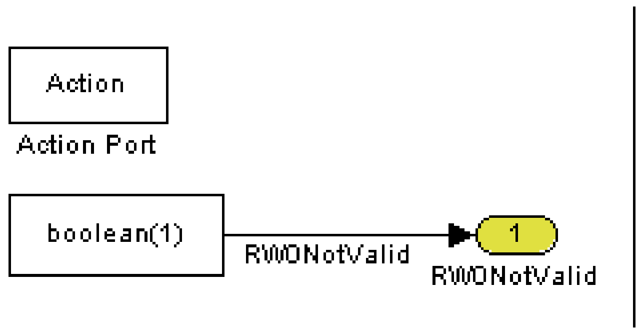
Name	RW0 Not Valid
Depth	6
Type	block
Blocks	Action Port Constant RW0NotValid

Table 2-251. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.69.1. Description

Figure 2-57. RW0 Not Valid



2.69.1.1. Signals

Table 2-252. RW0 Not Valid Signal Information

InputSignalNames	RW0NotValidAction

OutputSignalNames	

Table 2-253. Output Signal Information

Name	RW0NotValid
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Momentum Management/RWA Valid Logic/RW0 Not Valid/Constant
Description	

2.70. RW1 Not Valid

Table 2-254. RW1 Not Valid System Information

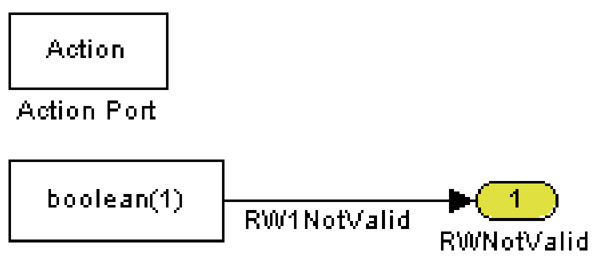
Name	RW1 Not Valid
Depth	6
Type	block
Blocks	Action Port Constant RWNotValid

Table 2-255. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.70.1. Description

Figure 2-58. RW1 Not Valid



2.70.1.1. Signals

Table 2-256. RW1 Not Valid Signal Information

InputSignalNames	RW1NotValidAction

OutputSignalNames	

Table 2-257. Output Signal Information

Name	RW1NotValid
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Momentum Management/RWA Valid Logic/RW1 Not Valid/Constant
Description	

2.71. RW2 Not Valid

Table 2-258. RW2 Not Valid System Information

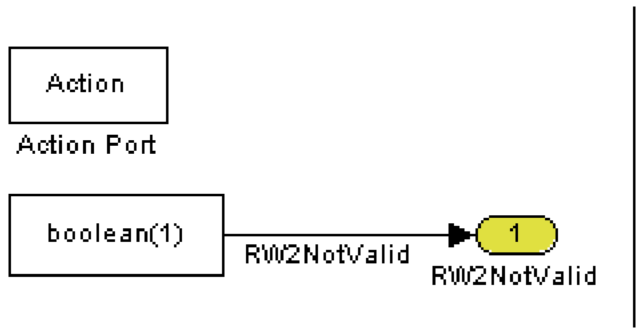
Name	RW2 Not Valid
Depth	6
Type	block
Blocks	Action Port Constant RW2NotValid

Table 2-259. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.71.1. Description

Figure 2-59. RW2 Not Valid



2.71.1.1. Signals

Table 2-260. RW2 Not Valid Signal Information

InputSignalNames	RW2NotValidAction

OutputSignalNames	

Table 2-261. Output Signal Information

Name	RW2NotValid
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Momentum Management/RWA Valid Logic/RW2 Not Valid/Constant
Description	

2.72. RW3 Not Valid

Table 2-262. RW3 Not Valid System Information

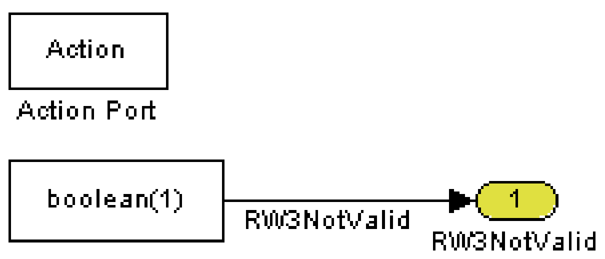
Name	RW3 Not Valid
Depth	6
Type	block
Blocks	Action Port Constant RW3NotValid

Table 2-263. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.72.1. Description

Figure 2-60. RW3 Not Valid



2.72.1.1. Signals

Table 2-264. RW3 Not Valid Signal Information

InputSignalNames	RW3NotValidAction

OutputSignalNames	

Table 2-265. Output Signal Information

Name	RW3NotValid
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Momentum Management/RWA Valid Logic/RW3 Not Valid/Constant
Description	

2.73. RWA Valid

Table 2-266. RWA Valid System Information

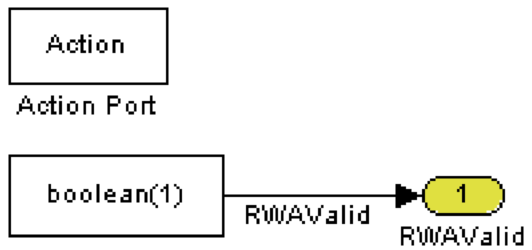
Name	RWA Valid
Depth	6
Type	block
Blocks	Action Port Constant RWAValid

Table 2-267. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.73.1. Description

Figure 2-61. RWA Valid



2.73.1.1. Signals

Table 2-268. RWA Valid Signal Information

InputSignalNames	RWAValidAction

OutputSignalNames	

Table 2-269. Output Signal Information

Name	RWAValid
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/ActuatorManagement/ Momentum Management/RWA Valid Logic/RWA Valid/Constant
Description	

2.74. Coarse Control Mode Enable

Table 2-270. Coarse Control Mode Enable System Information

Name	Coarse Control Mode Enable
Depth	6
Type	block
Blocks	CoarseEnable1 UnloadingEnable Logical Operator CoarseEnable

Table 2-271. acs_documentation Information

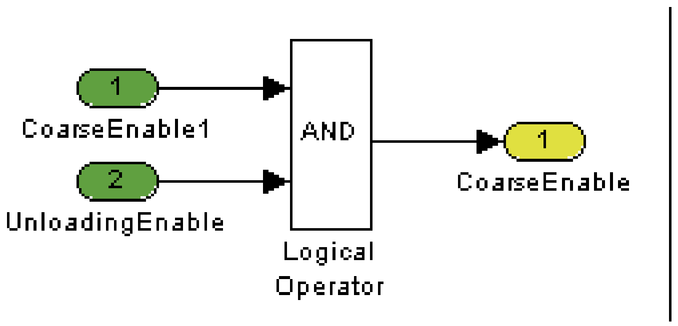
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.74.1. Description

```
Handles the enabling of the Coarse Pointing control mode.
```

Description from system mask help.

Figure 2-62. Coarse Control Mode Enable



2.74.1.1. Signals

Table 2-272. Coarse Control Mode Enable Signal Information

InputSignalNames	
OutputSignalNames	

2.74.2. Validation

Test001

2.75. Coarse Pointing

Table 2-273. Coarse Pointing System Information

Name	Coarse Pointing
Depth	6
Type	block
Blocks	EstAttitudeError EstRateError RWAMomentum Enable 3x3 cross product Bus Creator Bus Selector Bus Selector1 Bus Selector2 Determine Controller Gain Matrices Memory Product1 Product2 Product3 Saturation Sum2 Vector Part Zero-Order Hold Coarse.SampleTime CoarseCtrlTorque

Table 2-274. acs_documentation Information

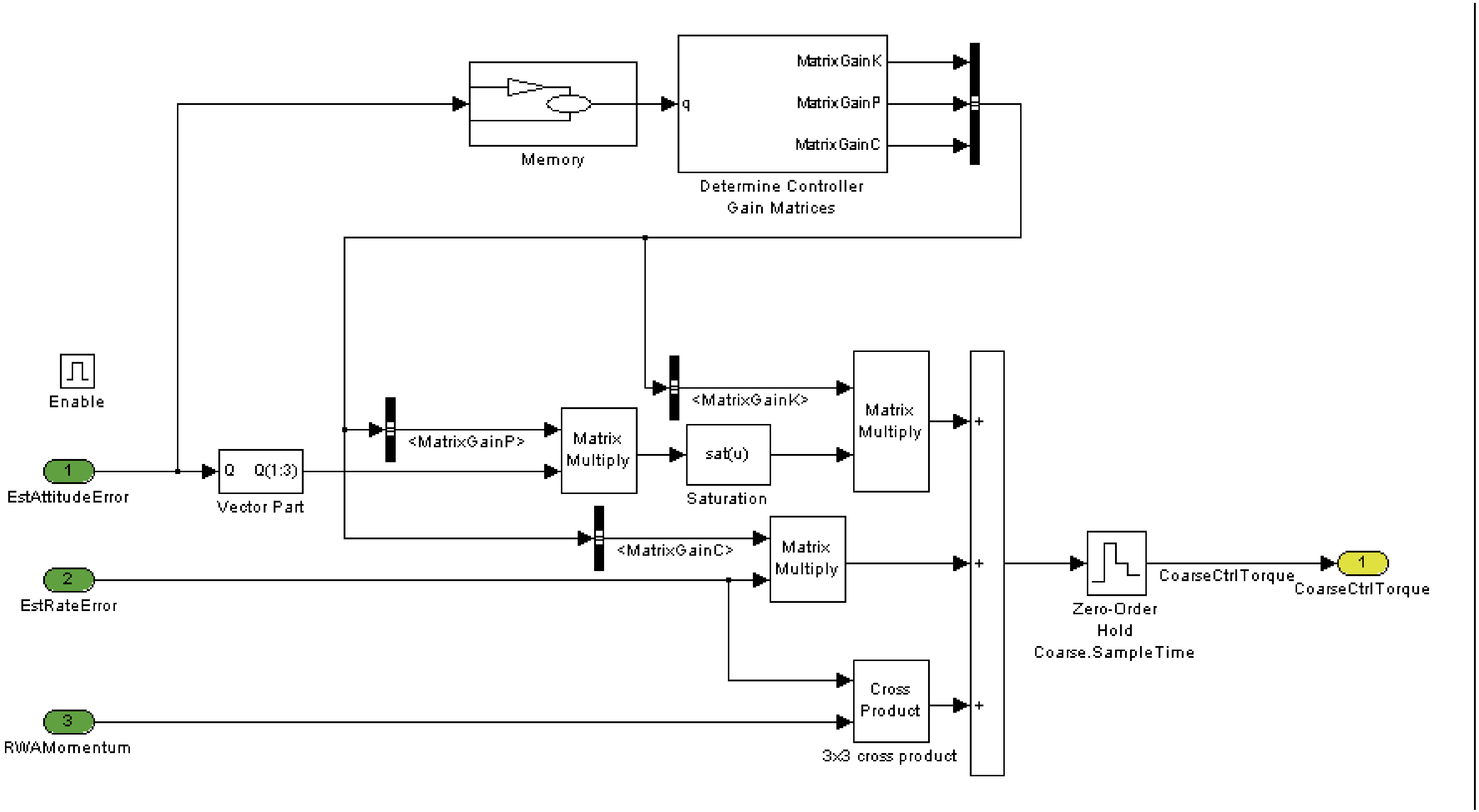
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.75.1. Description

```
Implements the Coarse Pointing control algorithm used in the Coarse and Unloading control mode.
```

Description from system mask help.

Figure 2-63. Coarse Pointing



2.75.1.1. Signals

Table 2-275. Coarse Pointing Signal Information

InputSignalNames	<EstAttitudeError> <EstRateError>
OutputSignalNames	

2.75.2. Validation

Test001

2.76. Fine Pointing

Table 2-276. Fine Pointing System Information

Name	Fine Pointing
Depth	6
Type	block
Blocks	IntFineEnable EstAttitudeError EstRateError Action Port Constant1 Ground Terminator Terminator1 Terminator2 Terminator3 FineCtrlTorque

Table 2-277. acs_documentation Information

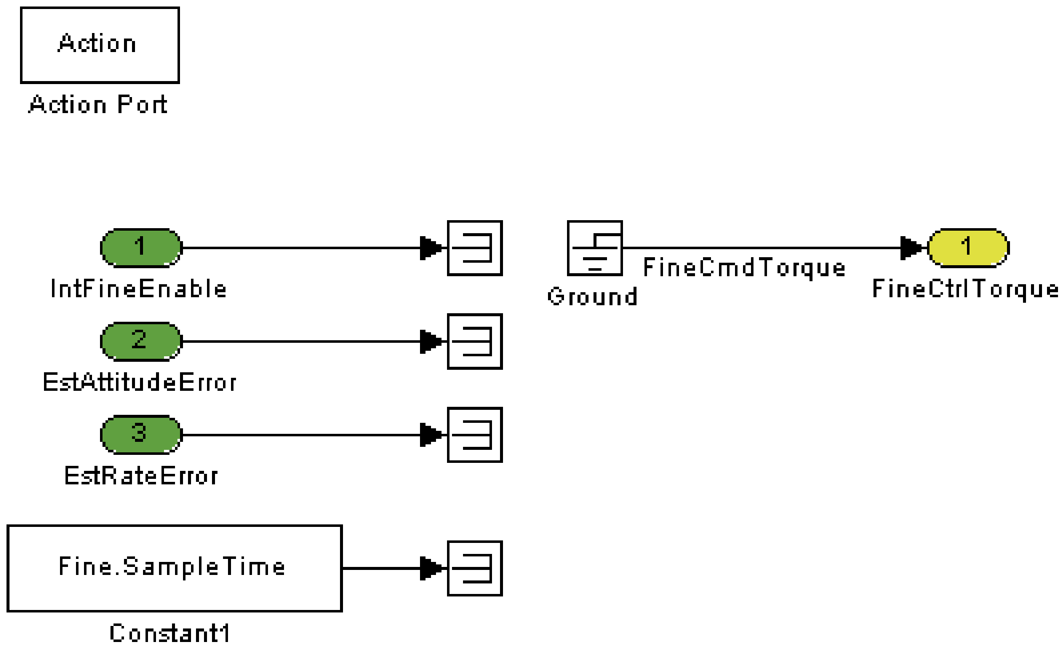
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.76.1. Description

```
Implements the Fine Pointing control algorithm used in the Fine
Pointing control mode.
```

Description from system mask help.

Figure 2-64. Fine Pointing



2.76.1.1. Signals

Table 2-278. Fine Pointing Signal Information

InputSignalNames	<EstRateError> <EstAttitudeError>
OutputSignalNames	

2.76.2. Validation

Test001

2.77. Action -> Enable

Table 2-279. Action -> Enable System Information

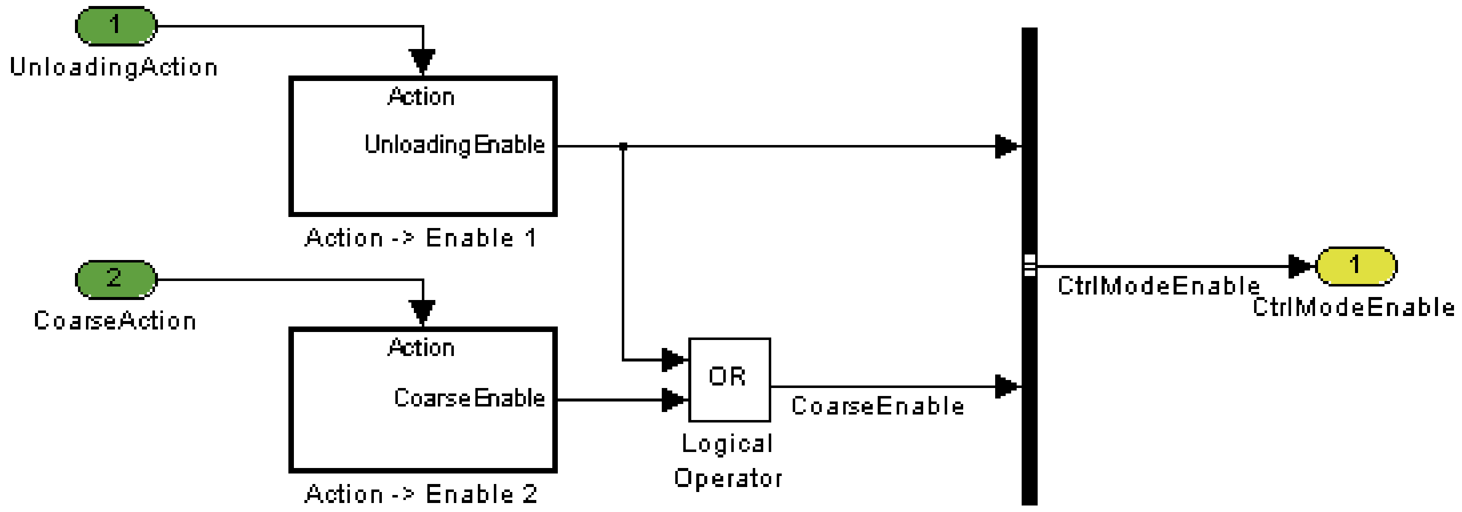
Name	Action -> Enable
Depth	6
Type	block
Blocks	UnloadingAction CoarseAction Action -> Enable 1 Action -> Enable 2 Bus Creator Logical Operator CtrlModeEnable

Table 2-280. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.77.1. Description

Figure 2-65. Action -> Enable



2.77.1.1. Signals

Table 2-281. Action -> Enable Signal Information

InputSignalNames	UnloadingAction CoarseAction
OutputSignalNames	

Table 2-282. Input Signal Information

Name	<5591.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Control Mode Action/Action -> Enable /UnloadingAction
Description	

Table 2-283. Input Signal Information

Name	<5592.0016>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Control Mode Action/Action -> Enable /CoarseAction
Description	

Table 2-284. Output Signal Information

Name	CtrlModeEnable

ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Control Mode Action/Action -> Enable /Bus Creator
Description	

2.78. MTACurrent Range Check

Table 2-285. MTACurrent Range Check System Information

Name	MTACurrent Range Check
Depth	7
Type	block
Blocks	MTACurrent MTACRngCheckEnable Bus Creator Bus Selector1 Bus Selector2 Bus Selector3 Bus Selector4 Bus Selector5 Bus Selector6 Range Check1 Range Check2 Range Check3 MTACurrentRngError

Table 2-286. acs_documentation Information

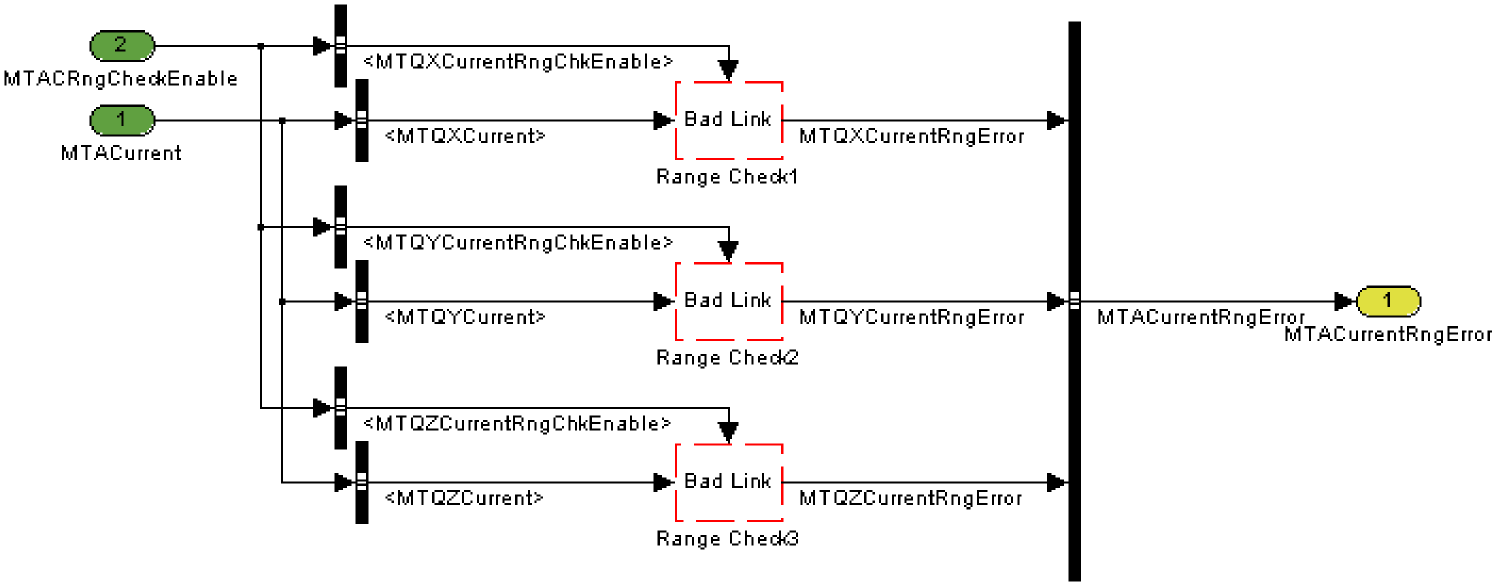
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.78.1. Description

```
Handles range check on reported currents from the magnetic torquers in the MTA.
```

Description from system mask help.

Figure 2-66. MTACurrent Range Check



2.78.1.1. Signals

Table 2-287. MTACurrent Range Check Signal Information

InputSignalNames	
OutputSignalNames	

2.78.2. Validation

Test001

2.79. MTAVoltage Range Check

Table 2-288. MTAVoltage Range Check System Information

Name	MTAVoltage Range Check
Depth	7
Type	block
Blocks	MTAVoltage MTAVRngCheckEnable Bus Creator Bus Selector1 Bus Selector2 Bus Selector3 Bus Selector4 Bus Selector5 Bus Selector6 Range Check1 Range Check2 Range Check3 MTAVoltageRngError

Table 2-289. acs_documentation Information

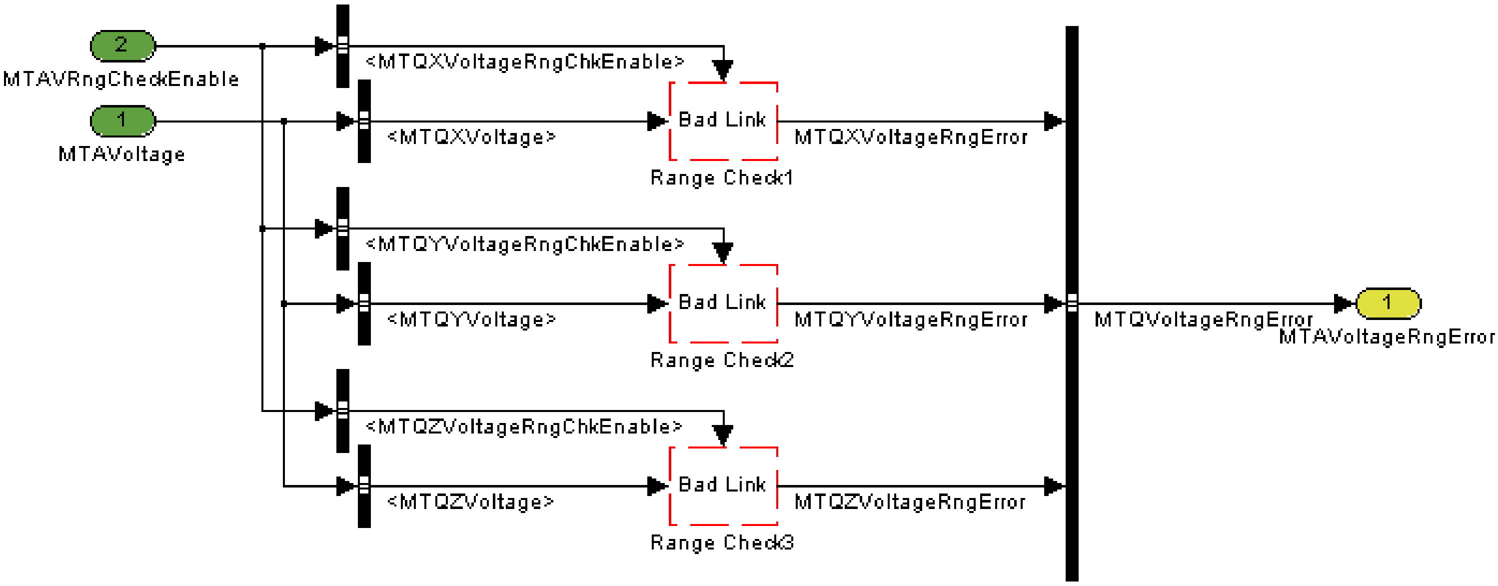
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.79.1. Description

```
Handles range check on reported voltages from the magnetic torquers in the MTA.
```

Description from system mask help.

Figure 2-67. MTAVoltage Range Check



2.79.1.1. Signals

Table 2-290. MTAVoltage Range Check Signal Information

InputSignalNames	
OutputSignalNames	

2.79.2. Validation

Test001

2.80. RWAMeasSpeed Range Check

Table 2-291. RWAMeasSpeed Range Check System Information

Name	RWAMeasSpeed Range Check
Depth	7
Type	block
Blocks	RWAMeasSpeed RWASpeedRngChkEnable Bus Creator Bus Selector1 Bus Selector2 Bus Selector3 Bus Selector4 Bus Selector5 Bus Selector6 Bus Selector7 Bus Selector8 Range Check0 Range Check1 Range Check2 Range Check3 RWAspeedRngError

Table 2-292. acs_documentation Information

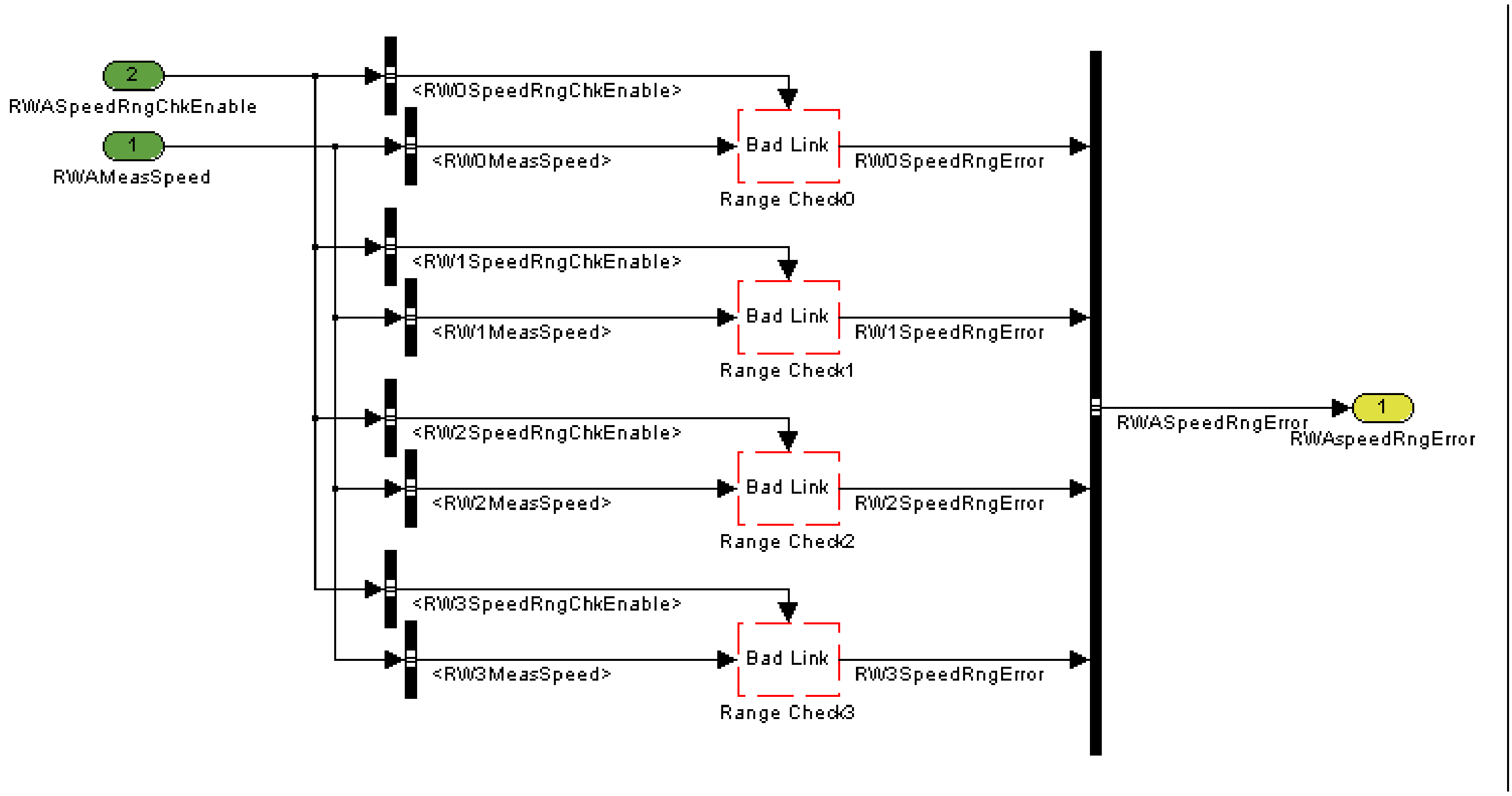
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.80.1. Description

```
This block performs range check on reported torque from the RWA.
```

Description from system mask help.

Figure 2-68. RWAMeasSpeed Range Check



2.80.1.1. Signals

Table 2-293. RWAMeasSpeed Range Check Signal Information

InputSignalNames	
OutputSignalNames	

2.80.2. Validation

Test001

2.81. RWAMeasTorque Range Check

Table 2-294. RWAMeasTorque Range Check System Information

Name	RWAMeasTorque Range Check
Depth	7
Type	block
Blocks	RWAMeasTorque RWATorqueRngChkEnable Bus Creator Bus Selector1 Bus Selector2 Bus Selector3 Bus Selector4 Bus Selector5 Bus Selector6 Bus Selector7 Bus Selector8 Range Check0 Range Check1 Range Check2 Range Check3 RWATorqueRngError

Table 2-295. acs_documentation Information

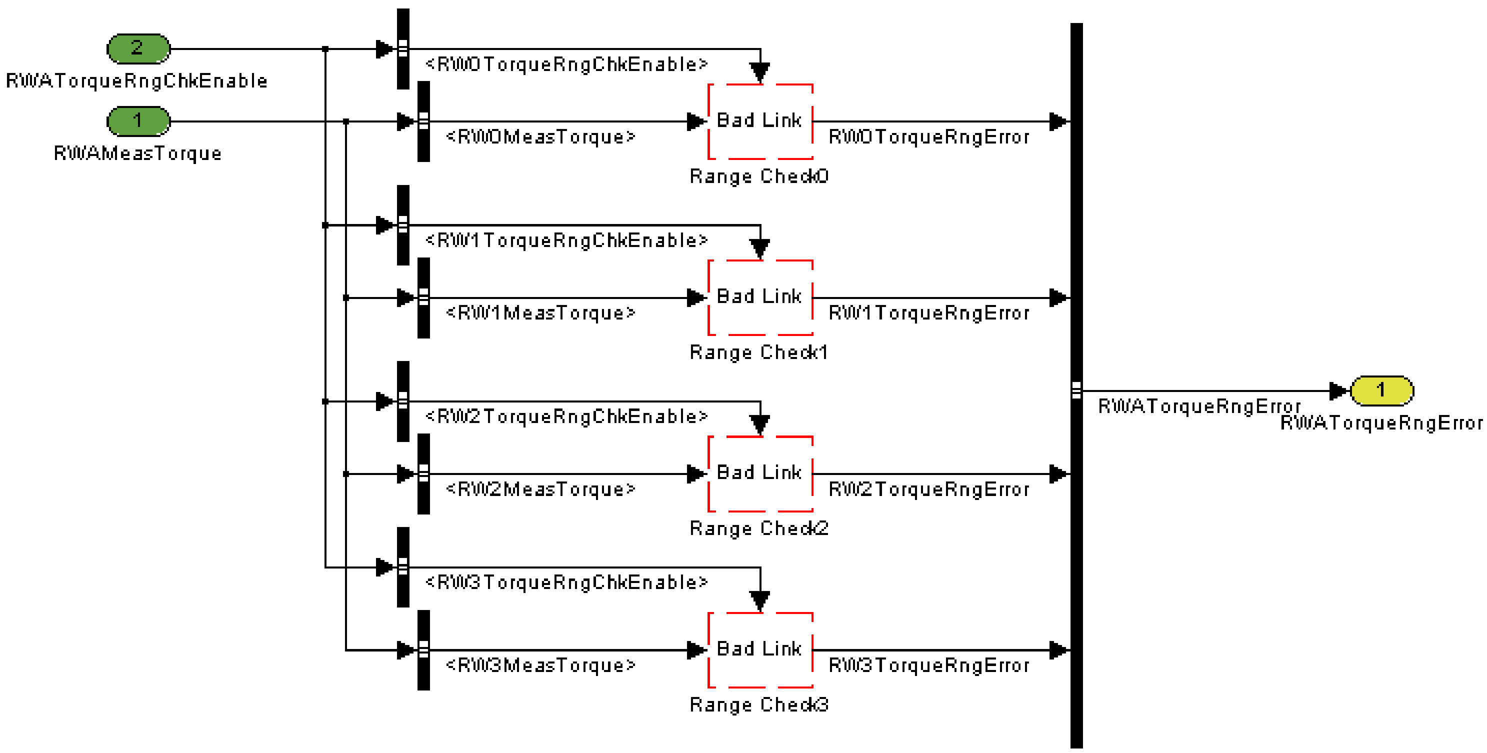
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.81.1. Description

```
This block performs range check on reported torque from the RWA.
```

Description from system mask help.

Figure 2-69. RWAMeasTorque Range Check



2.81.1.1. Signals

Table 2-296. RWAMeasTorque Range Check Signal Information

InputSignalNames	
OutputSignalNames	

2.81.2. Validation

Test001

2.82. 3x3 cross product

Table 2-297. 3x3 cross product System Information

Name	3x3 cross product
Depth	7
Type	block
Blocks	a(1) a(2) a(3) b(1) b(2) b(3) Product Product1 Selector Selector1 Selector2 Selector3 Sum c(1) c(2) c(3)

Table 2-298. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.82.1. Description

```
Calculates the cross product of two 3x1 vectors.
```

Description from system mask help.

2.82.1.1. Signals

Table 2-299. 3x3 cross product Signal Information

InputSignalNames	
OutputSignalNames	

2.82.2. Validation

Test001

2.83. Determine Controller Gain Matrices

Table 2-300. Determine Controller Gain Matrices System Information

Name	Determine Controller Gain Matrices
Depth	7
Type	block
Blocks	q LU Inverse 2*(Coarse.NatFreq*2*pi)^2 Constant1 Constant2 Create Diagonal Matrix Fcn1 Fcn2 Fcn3 IC1 Matrix Multiply Mux MatrixGainK MatrixGainP MatrixGainC

Table 2-301. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.83.1. Description

Description from system mask help.

Figure 2-70. Determine Controller Gain Matrices

2.83.1.1. Signals

Table 2-302. Determine Controller Gain Matrices Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-303. Input Signal Information

Name	<6835.0035>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Coarse//Fine Pointing/Coarse Pointing/Determine Controller Gain Matrices/q
Description	

Table 2-304. Output Signal Information

Name	MatrixGainC
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Coarse//Fine Pointing/Coarse Pointing/Determine Controller Gain Matrices/Constant2
Description	

Table 2-305. Output Signal Information

Name	MatrixGainK

ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Coarse//Fine Pointing/Coarse Pointing/Determine Controller Gain Matrices/Create Diagonal Matrix
Description	

Table 2-306. Output Signal Information

Name	MatrixGainP
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Coarse//Fine Pointing/Coarse Pointing/Determine Controller Gain Matrices/2*(Coarse.NatFreq*2*pi)^2
Description	

2.84. Memory

Table 2-307. Memory System Information

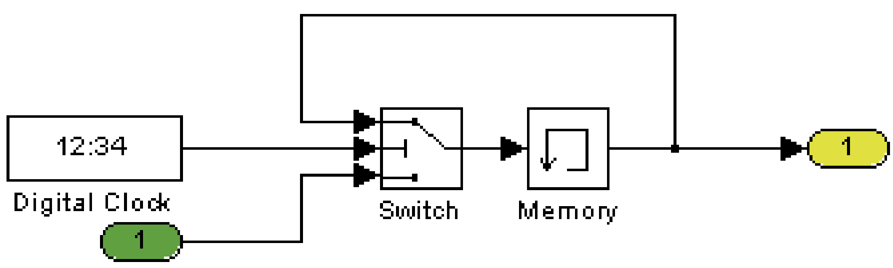
Name	Memory
Depth	7
Type	block
Blocks	In1 Digital Clock Memory Switch Out1

Table 2-308. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.84.1. Description

Figure 2-71. Memory



2.84.1.1. Signals

Table 2-309. Memory Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-310. Input Signal Information

Name	<6846.0035>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Coarse//Fine Pointing/Coarse Pointing/Memory/In1
Description	

Table 2-311. Output Signal Information

Name	<6840.0035>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Coarse//Fine Pointing/Coarse Pointing/Memory/Memory
Description	

2.85. Saturation

Table 2-312. Saturation System Information

Name	Saturation
Depth	7
Type	block
Blocks	u Saturation sat(u)

Table 2-313. acs_documentation Information

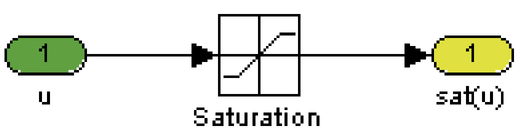
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.85.1. Description

```
Saturate elements of n-dimensional vector with specified upper and
lower bounds.
```

Description from system mask help.

Figure 2-72. Saturation



2.85.1.1. Signals

Table 2-314. Saturation Signal Information

InputSignalNames	
OutputSignalNames	

2.85.2. Validation

2.86. Vector Part

Table 2-315. Vector Part System Information

Name	Vector Part
Depth	7
Type	block
Blocks	Q Demux Mux Terminator Q(1:3)

Table 2-316. acs_documentation Information

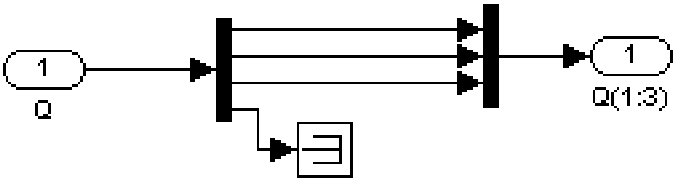
LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.86.1. Description

```
Extracts the vector part of a quaternion, assuming Q(4) is the
scalar element.
```

Description from system mask help.

Figure 2-73. Vector Part



2.86.1.1. Signals

Table 2-317. Vector Part Signal Information

InputSignalNames	
OutputSignalNames	

2.86.2. Validation

2.87. Action -> Enable 1

Table 2-318. Action -> Enable 1 System Information

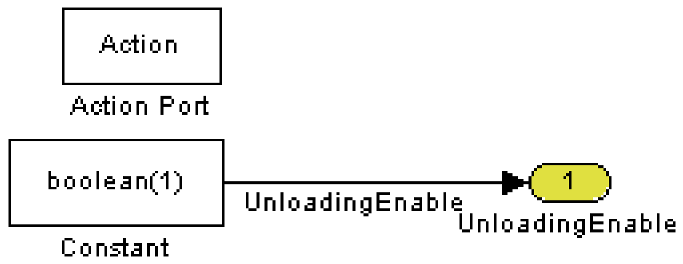
Name	Action -> Enable 1
Depth	7
Type	block
Blocks	Action Port Constant UnloadingEnable

Table 2-319. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.87.1. Description

Figure 2-74. Action -> Enable 1



2.87.1.1. Signals

Table 2-320. Action -> Enable 1 Signal Information

InputSignalNames	

OutputSignalNames	

Table 2-321. Output Signal Information

Name	UnloadingEnable
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Control Mode Action/Action -> Enable /Action -> Enable 1/ Constant
Description	

2.88. Action -> Enable 2

Table 2-322. Action -> Enable 2 System Information

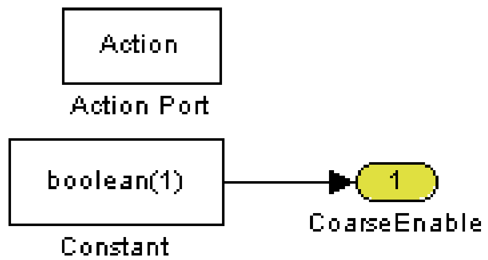
Name	Action -> Enable 2
Depth	7
Type	block
Blocks	Action Port Constant CoarseEnable

Table 2-323. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.88.1. Description

Figure 2-75. Action -> Enable 2



2.88.1.1. Signals

Table 2-324. Action -> Enable 2 Signal Information

InputSignalNames	
OutputSignalNames	

Table 2-325. Output Signal Information

Name	<6863.0035>
ParentBlock	acs_documentation/Rømer ACS/ProcesLayer/Control/Control Algorithms/Control Mode Action/Action -> Enable /Action -> Enable 2/ Constant
Description	

2.89. LU Inverse

Table 2-326. LU Inverse System Information

Name	LU Inverse
Depth	8
Type	block
Blocks	A Identity Matrix LU Solver inv(A)

Table 2-327. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.89.1. Description

```
Matrix inverse using LU factorization.
```

Description from system mask help.

2.89.1.1. Signals

Table 2-328. LU Inverse Signal Information

InputSignalNames	MatrixGainK
OutputSignalNames	

2.89.2. Validation

2.90. LU Solver

Table 2-329. LU Solver System Information

Name	LU Solver
Depth	9
Type	block
Blocks	A B Backward Substitution Forward Substitution LU Factorization Permute Matrix X

Table 2-330. acs_documentation Information

LastModifiedDate	Mon Aug 19 16:18:52 2002
LastModifiedBy	tb

2.90.1. Description

```
Solve AX=B using LU decomposition.  A must be square.  B must have
the same number of rows as A.
```

Description from system mask help.

2.90.1.1. Signals

Table 2-331. LU Solver Signal Information

InputSignalNames	
OutputSignalNames	

2.90.2. Validation

Chapter 3. Other contributions

Chapter 4. Revisions

LU Inverse.mdl: Revision No revision found

Cross Product.mdl: Revision No revision found

Actuator report data range check.mdl: Revision No revision found

ActuatorManagement_lib.mdl: Revision not found

Actuator report data range check.mdl: Revision No revision found

ActuatorManagement_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

Actuator report data range check.mdl: Revision No revision found

ActuatorManagement_lib.mdl: Revision not found

Control_lib.mdl: Revision not found

Control_lib.mdl: Revision not found

Control_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

Control_lib.mdl: Revision not found

Control_lib.mdl: Revision not found

Control_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

.mdl: Revision No revision found

ActuatorManagement_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

Control_lib.mdl: Revision not found

LU Solver.mdl: Revision No revision found

Control_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

ActautorManagement_lib.mdl: Revision No revision found

MTQ report data range check.mdl: Revision No revision found

ActuatorManagement_lib.mdl: Revision not found

ActuatorManagement_lib.mdl: Revision not found

Qmult.mdl: Revision No revision found

ActuatorManagement_lib.mdl: Revision not found

```
ActuatorManagement_lib.mdl: Revision not found

RWA report data range check.mdl: Revision No revision found

Actuator report data range check.mdl: Revision No revision found

Actuator report data range check.mdl: Revision No revision found

.mdl: Revision No revision found

Control_lib.mdl: Revision not found

Math_lib.mdl: Revision not found

Control_lib.mdl: Revision not found

SystemInterfaceLayer.mdl: Revision No revision found

Math_lib.mdl: Revision not found
```

Detailed Design Guidance Component

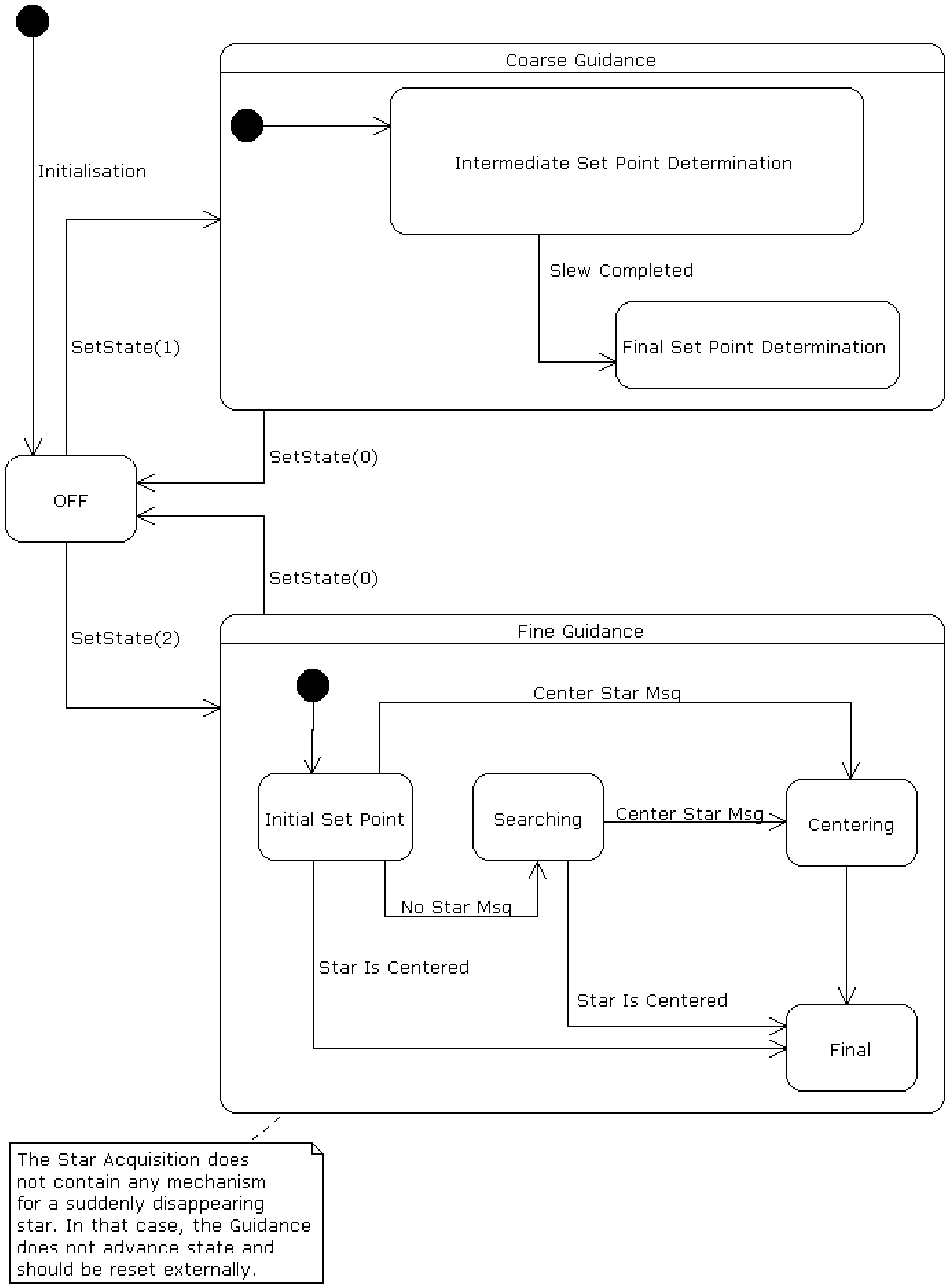
Description
The Guidance component delivers setpoints to the Control and Navigation units on the basis of setpoint commands from the Commander and from the Command Interface, as well as on the basis of PayloadStarMessages.

Input
(**SetParameter**)
(**GetParameter**)
SetState [1x1 uint8] Selection of the GeneratorMode [Off(0), CoarseGuidance(1), FineGuidance(2)]
SetAttitudeSetpoint (
  AttitudeSetpoint [4x1 double] Spacecraft commanded attitude quaternion relative to ECI [N/A]
  RateSetpoint [3x1 double] Spacecraft commanded angular velocity in SCB frame [rad/s]
)
SetSlewStartAttRate(
  StartAttitude [4x1 double] Spacecraft referenceattitude quaternion relative to ECI at start of slew [N/A]
  StartRate [3x1 double] Spacecraft angular velocity in SCB frame at start of slew [rad/s]
)
PayloadStarMessage(
  Message [1x1 uint8] Indication of Payload Star Sighting [Missing (0), Center (1)]
  CenterValue [2x1 double] Star Coordinate in P/L field of view [arcsec TBC]
)

Masked Parameters
MaxSlewRate [1x1 double] Maximum Total Spacecraft Angular Rate relative to ECI [rad/sec]
MaxSlewAcc [1x1 double] Maximum Total Spacecraft Angular Acceleration relative to ECI [rad/sec^2]
SlewTimeSlice [1x1 double] Amount of time per reference update [sec]
AcquisitionGridTable [4x100 double] Table of Quaternions indicating relative attitudes w.r.t. initial fine pointing set point. Table has size 100, but has MaxStarAcqTrials entries. The rest is filled up with identity attitude [N/A]
StarAcqIndex [1x1 uint8] Counter indicating which entry in the AcquisitionGridTable is being used at the moment [N/A]
MaxStarAcqTrials [1x1 uint8] Number of valid entries in the AcquisitionGridTable [N/A]
StarCenterDeadZone [1x1 double] Maximum allowed off axis position of star in telescope field of view without recentering [arcsec].

Output
ReferenceState: (A22) (
  AttitudeRef [4x1 double]: Spacecraft reference attitude quaternion relative to ECI frame [N/A]
  RateRef [3x1 double]: Spacecraft commanded angular velocity in SCB frame [rad/s]
)

State Diagram



Component User Manual

The Guidance component is supposed to be used in the following sequence:

1. SetState(0), resets the internal state to OFF.
2. SetAttitudeSetPoint, to set the desired reference attitude and rate for the spacecraft
3. For Coarse Guidance: SetState(1)
4. For Fine Guidance: SetState(2)

The Coarse Guidance can be used in combination with Coarse and Unloading Control/Navigation.
The Fine Guidance can be used in combination with Fine Control/Navigation.

All other control modes (Safe) require the Guidance OFF.

In Coarse Guidance, the input SetSlewStartAttRate is required to be active.
The PayloadStarMessage input is only used when in Fine Guidance, otherwise it is ignored.

Operations

This section defines the operations to be performed by the component, on the basis of events. These operations are different per state.

Operations in OFF State

Entry:
The output ReferenceState is set equal to the input SetAttitudeSetPoint.

Exit:
None.

SetState (OFF)
The state is transited to OFF state (this means the entry event is called).

SetState (COARSE_GUIDANCE)
The state is changed to Coarse Guidance.

SetState (FINE_GUIDANCE)
The state is changed to Fine Guidance.

Operations in Coarse Guidance State

Entry:

1. ComputeDeltaQuaternion
2. ComputeDeltaAngleAndVector, the result of this is fieflux, the vector n, and fiemax.
3. Set fie=0
4. A judgment is needed on the size of the fiemax. If below a threshold then transit to Coarse/Final Set Point Determination, else transit to Coarse/Intermediate Set Point Determination.

Exit:
None.

SetState (OFF)
The state is transited to OFF state.

SetState (COARSE_GUIDANCE)
The state is changed to Coarse Guidance (entry event)

SetState (FINE_GUIDANCE)
Ignored.

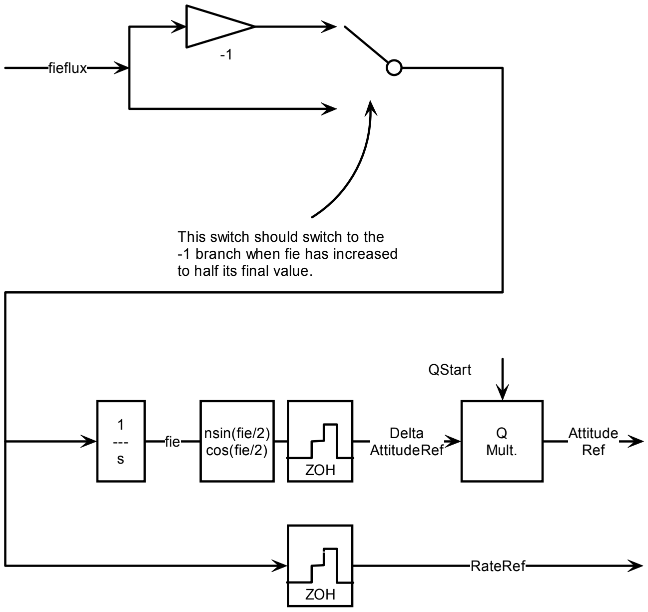
Operations in Coarse/Intermediate Set Point Determination

Entry:
None.

Exit:
None.

Every control cycle:

The following schema computes the references for the output of the Guidance module.
When fie reaches its final value, transit to Coarse/Final Set Point Determination



Operations in Coarse/Final Set Point Determination

Entry:
Set the AttitudeRef and the RateRef to the commanded values.

Exit:
None.

Operations in Fine Guidance State

Entry:
Transit to Fine/Initial Set Point

Exit:
None.

SetState (OFF)
The state is transited to OFF state.

SetState (COARSE_GUIDANCE)
Ignored.

SetState (FINE_GUIDANCE)
The state is changed to Fine Guidance (this means the entry event is called).

Operations in Fine/Initial Set Point

Entry:
1. Set the AttitudeRef and the RateRef to the commanded values.
2. Set StarAcqIndex to 0

Exit:
None.

PayloadStarMessage(Center)
1. If Delta is lower than StarCenterDeadZone, transit to Fine/Final,
2. else transit to Fine/Centering.

PayloadStarMessage(No Star)
Transit to Fine/Searching.

Operations in Fine/Searching

Entry:

Exit:
None.

PayloadStarMessage(Center)
1. If the Delta is lower than StarCenterDeadZone then transit to Fine/Final,
2. else transit to Fine/Centering.

PayloadStarMessage(No Star)
1. Increase StarAcqIndex with 1.
2. If StarAcqIndex<=MaxStarAcqTrials then set the AttitudeRef output to the corresponding entry in the StarAcqGridTable, by multiplying the commanded set point with the table entry,
3. else do nothing (TBC).

Operations in Fine/Centering

Entry:
1. Compute the final set point with help of the Delta and the current setpoint.
2. Set the AttitudeRef to this final setpoint.
3. Transit to Final.

Exit:
None.

PayloadStarMessage(Center)
Ignored.

PayloadStarMessage(No Star)
Ignored.

Operations in Fine/Final

Entry:
Keep the output as it is.

Exit:
None

PayloadStarMessage(Center)
Ignored.

PayloadStarMessage(No Star)
Ignored.

Detailed Design Data Interface Component

Description
The Data Interface is called by the PUS Service Layer to request values of parameters, and to change values of parameters in modules of the ACS. Every parameter has a unique identifier, while the Data Interface has a table which maps ranges of identifiers to the components in the ACS.
Parameters can only be changed while the ACS is disabled. A parameter change is responded to with a ACS_Parameter_Report. If the change has not been performed, a flag in this report is set to false.

Input
(SetParameter)
(GetParameter)
SetState (
    ParameterChangeEnable [1x1 boolean] Allows parameters to be changed or not.
)

ParameterReport (
    ByteSize [1x1 uint8] The ID of the parameter/variable that is requested.
    ParameterId [1x1 unit16] Id of reported parameter (TBD)
    ParameterValue [ByteSizex1 unit8] Binary structure depending on requested parameter type with a length of <ByteSize> octets.
)

ACS_Get_Parameter (
    ParameterId [1x1 uint8] The ID of the parameter/variable that is set.
)

ACS_Set_Paremeter (
    ParameterId [1x1 uint8] The ID of the parameter/variable that is set.
    ParameterValue [variable] The value of the parameter that is set.
)

Masked Parameters
ParameterMap [TBDx3 uint8] Map with [lower range, upper range, component identifier].

Component Identifiers:
1: Data Interface
2: Event Interface
3: Command Interface
4: Guidance
5: Commander
6: Navigation
7: Fault Detection
8: Control
9: Sensor Management
10: Actuator Management

Output
ACS_ParameterReport (
    ByteSize [1x1 uint8] The ID of the parameter/variable that is requested.
    ParameterId [1x1 unit16] Id of reported parameter (TBD)
    ParameterValue [ByteSizex1 unit8] Binary structure depending on requested parameter type with a length of <ByteSize> octets.
    CheckFlag [1x1 boolean] Is False when there is a problem.
)

Possible causes for FALSE CheckFlag in ACS_ParameterReport:
- call to ACS_SetParameter while ACS is Enabled,
- call to ACS_SetParameter with invalid parameter ID,
- TBC: call to ACS_SetParameter with out of range parameter,

- call to ACS_GetParameter with invalid parameter ID.

Operations
N/A

State Diagram
N/A

Component User Manual
Parameter Request: A parameter is requested by the PUS Service Layer by ACS_GetParameter with the appropriate ID. Expected response is an ACS_ParameterReport, with the value and a checkflag. The component itself calls the GetParameter interface of the related component and expects a ParameterReport back from that component.

Parameter Change: A parameter is changed by the PUS Service Layer by ACS_SetParameter with the appropriate ID. Expected response is an ACS_ParameterReport, with the value and a checkflag. This is only possible when the ACS is disabled, otherwise a report with a false flag is returned. The component itself calls the SetParameter interface of the related component but does not expect a ParameterReport back from that component.

In order to inform the component on that parameter changes are allowed, the SetState interface must be used (by Commander) to enable or disable it.

Detailed Design Command Interface Component

Description

The Command Interface is called by the PUS Service Layer to pass on commands, which are mainly the enabling and disabling, switching of modes and setpoints, but also to report on enabling/disabling of equipment. Commands are acknowledged with a TC verification report, which serves as information source for the PUS command verification.

Input

(SetParameter)
(GetParameter)

ACS_Enable (Enable [1x1 boolean] to enable/disable ACS)

ACS_ModeSelect (
    Desired Mode [1x1 uint8] selects mode [Unloading(0), Coarse(1), Fine(2), safe(3), Standby (4)]
    Attitude Set Point [4x1 double] Spacecraft commanded attitude quaternion relative to ECI [N/A]
)

ACS_SetAutonomyOnOff [1x1 boolean] Enables/Disables the onboard autonomy [N/A]

ACS_ChangeAttitudeDeterminator [1x1 uint8] Selects a Navigation Mode [Raw (0), Coarse(1), Fine(2), Safe(3)]

ACS_SetAttitudeSetpoint (
    AttitudeSetpoint [4x1 double] Spacecraft commanded attitude quaternion relative to ECI [N/A]
    RateSetpoint [3x1 double] Spacecraft commanded angular velocity in SCB frame [rad/s]
)

ACS_SetUTCAndOrbitalElements (
    Onboard Time Value [1x1 TBD] Reading of the CDH clock [TBD]
    Orbital Elements [TBD] Description of orbit in inertial space [various]
)

ACS_CenterStar [2x1 double] Star Coordinate in P/L field of view [arcsec TBC]

ACS_MissingStar

ACS_STRStatusReport [1x1 uint8] Indication of STR State [Disabled (0), EnabledCHUPLUSZ (1), EnabledCHUMINZ (2)]

ACS_GYRStatusReport [4x1 uint8] Indication of each rate sensor state [Off (0), Off_Passed(1), On_Failed(2), On_Passed(3)]

ACS_SSAStatusReport [6x1 boolean] Indication of each sunsensor status [N/A]

ACS_MAGStatusReport [1x1 boolean] Indication of magnetometer status [N/A]

ACS_WheelStatus [4x1 boolean] Indication of each wheel status [N/A]

ACS_TorquerStatus [3x1 boolean] Indication of each torquer status [N/A]

TCVerificationReport(
    Acceptance [1x1 boolean] Indication of Command Acceptance [REJECTED (0), ACCEPTED(1)]
    AcceptErrorCode [1x1 uint8] Indicates reason of failure for acceptance [INVALID-CMD (0), OK (1)]
    Execution [1x1 boolean] Indicates completion of the command execution [N/A]
    ExeErrorCode [1x1 uint8] Indicates reason of failure for execution. If no failure, then this parameter is set to OK. [TBD]
)

ConfigureStarTracker (
Enable [1x1 boolean] Enabling Request of Star Tracker Operation [N/A]
CHUSelect [1x1 uint8] Selection of Camera [PLUSZ (0), MINZ(1)]
)

ConfigureRateSensors (
RGA0Enable [1x1 boolean] Enabling Request of RGA in wheel 0 [N/A]
RGA1Enable [1x1 boolean] Enabling Request of RGA in wheel 1 [N/A]
RGA2Enable [1x1 boolean] Enabling Request of RGA in wheel 2 [N/A]
RGA3Enable [1x1 boolean] Enabling Request of RGA in wheel 3 [N/A]
)

ConfigureSunSensors (
SSAPlusXEnable [1x1 boolean] Enabling Request of SSA [N/A]
SSAMinXEnable [1x1 boolean] Enabling Request of SSA [N/A]
SSAPlusYEnable [1x1 boolean] Enabling Request of SSA [N/A]
SSAMinYEnable [1x1 boolean] Enabling Request of SSA [N/A]
SSAPlusZEnable [1x1 boolean] Enabling Request of SSA [N/A]
SSAMinZEnable [1x1 boolean] Enabling Request of SSA [N/A]
)

ConfigureMagnetometer [1x1 boolean] Enabling Request of Magnetometer [N/A]

ConfigureWheels (
RWA0Enable [1x1 boolean] Enabling Request for wheel 0 [N/A]
RWA1Enable [1x1 boolean] Enabling Request for wheel 1 [N/A]
RWA2Enable [1x1 boolean] Enabling Request for wheel 2 [N/A]
RWA3Enable [1x1 boolean] Enabling Request for wheel 3 [N/A]
)

ConfigureTorquers (
TRQ0Enable [1x1 boolean] Enabling Request for torquer 0 [N/A]
TRQ1Enable [1x1 boolean] Enabling Request for torquer 1 [N/A]
TRQ2Enable [1x1 boolean] Enabling Request for torquer 2 [N/A]
)

Masked Parameters
None.

Output
ACS_EnableWheels [4x1 boolean] Enable/Disable wheels [N/A]
ACS_EnableTorquers [3x1 boolean] Enable/Disable torquers [N/A]
ACS_EnableSTR [1x1 uint8] Select CHU [Off (0), PLUSZ (1), MINZ(2)]
ACS_EnableGYR [4x1 boolean] Enable/Disable rate sensors [N/A]
ACS_EnableSAS [6x1 boolean] Enable/Disable sun sensors [N/A]
ACS_EnableMAG [1x1 boolean] Enable/Disable magnetometer [N/A]

Enable [1x1 boolean] Enables the ACS, thereby activating the ACS RuleBase Logic and initiating configurations.
ModeSelect (
ModeSelect [1x1 uint8] Which Control Mode is desired [Unloading(0), Coarse(1), Fine(2), Safe(3), Standby (4)]
AttitudeSetpoint [4x1 double] Spacecraft commanded attitude quaternion relative to ECI [N/A]
)
AttitudeDeterminatorSelect [1x1 uint8] Selects a Navigation Mode [Raw (0), Coarse(1), Fine(2), Safe(3)]
AutonomyOnOff [1x1 uint8] Commands the autonomy level [Off(0), On(1)]
SystemStateVector (
RWAState [4x1 boolean] State of the wheels [N/A]
MTQState [3x1 boolean] State of the torquers[N/A]

STRState [1x1 boolean] State of the star tracker [N/A]
MAGState [1x1 boolean] State of the magnetometer [N/A]
SSAState [6x1 boolean] State of the sun sensors [N/A]
RGAState [4x1 boolean] State of the rate sensors [N/A]
)
SetAttitudeSetpoint (
AttitudeSetpoint [4x1 double] Spacecraft commanded attitude quaternion relative to ECI [N/A]
RateSetpoint [3x1 double] Spacecraft commanded angular velocity in SCB frame [rad/s]
)

PayloadStarMessage(
Message [1x1 integer] Indication of Payload Star Sighting [Missing (0), Center (1)]
CenterValue [2x1 double] Star Coordinate in P/L field of view [arcsec TBC]
)

SetUTCAndOrbitalElements (
OnboardTimeValue [1x1 TBD] Onboard time corresponding to the epoch time for elements given below [TBD]
EpochTime [1x1 double] Epoch time for elements. Synchronized with OnboardTimeValue [JD]
PerigeePrecession [1x1 double] Precession of perigee at epoch [rad/sec]
Perigee [1x1 double] Perigee at epoch [rad]
RAANPrecession [1x1 double] Precession of right ascending node at epoch [rad/sec]
RAAN [1x1 double] Right ascending node at epoch [rad]
Inclination [1x1 double] Inclination at epoch [rad]
Eccentricity [1x1 double] Eccentricity [N/A]
MeanAnomaly [1x1 double] Mean anomaly at epoch [rad]
SemiMajorAxis [1x1 double] Semi major axis [km]
MeanMotion [1x1 double] Mean motion [deg/day]
)

Operations
There is a direct connection between an input and a response on the output:
Incoming Interfaces

Input	Output	Remark
ACS_Enable	Commander::Enable	
ACS_ModeSelect	Commander::ModeSelect	
ACS_ChangeAttitudeDeterminator	Commander:: AttitudeDeterminatorSelect	
ACS_SetAttitudeSetpoint	Guidance::SetAttitudeSetpoint	
ACS_SetUTCAndOrbitalElements	Navigation::SetOrbitalElements	
ACS_CenterStar	Guidance::PayloadStarMessage	
ACS_MissingStar	Guidance::PayloadStarMessage	
ACS_STRStatusReport	Commander::SystemStateVector	
ACS_GYRStatusReport	Commander::SystemStateVector	
ACS_SSAStatusReport	Commander::SystemStateVector	
ACS_MAGStatusReport	Commander::SystemStateVector	
ACS_WheelStatus	Commander::SystemStateVector	
ACS_TorquerStatus	Commander::SystemStateVector	
ACS_SetAutonomyOnOff	Commander::AutonomyOnOff	

Outgoing Interfaces

Input	Output	Remark
TCVerificationReport	ACS_TCVerificationReport	
ConfigureStarTracker	ACS_EnableSTR	
ConfigureRateSensors	ACS_EnableGYR	
ConfigureSunSensors	ACS_EnableSAS	
ConfigureMagnetometer	ACS_EnableMAG	
ConfigureWheels	ACS_EnableWheels	
ConfigureTorquers	ACS_EnableTorquers	

State Diagram

N/A

Component User Manual

N/A

Detailed Design Event Interface Component

Description

The EventInterface is responsible for output of event based information to the PUS Service Layer. These events are a result of fault detection, autonomous reconfigurations and other state changes. The information is output as a set of values, which will be converted into a PUS Event Report by the PUS Service Layer. Origin of the events is always the Commander.

Note: *at the moment the event interface passes on information from the Commander to the PUS Service Layer. There is no processing or altering of the data. This means that this component is more or less conceptual, and might be completely invisible in the actual flight software.*

Input

ConfigurationReport(
CurrentMode [1x1 uint8] Indicates current mode of ACS [UNLOADING (0), COARSE (1), FINE (2), SAFE (3), STANDBY (4)]
CurrentCtrl [1x1 uint8] Control mode [UNLOADING (0), COARSE (1), FINE (2), SAFE (3), STANDBY (4)]
CurrentAD [1x1 uint8] Navigation mode [RAW (0), COARSE (1), FINE (2), SAFE (3)]
CurrentFD [1x1 uint8] Indicates current fault detector, [TBD]
AutonomyStatus [1x1 uint8] Indicating autonomy level [Off(0), On(1)]
GuidanceStatus [1x1 uint8] Indicating which mode Guidance is in [Off(0), CoarseGuidance(1), FineGuidance(2)]
RWAStatus [4x1 boolean] Indicates which wheels are enabled. [N/A]
TRQStatus [3x1 boolean] Indicates which torquers are enabled. [N/A]
MAGStatus [1x1 boolean] Indicates if magnetometer is enabled. [N/A]
SSAStatus [6x1 boolean] Indicates which sun sensors are enabled. [N/A]
RGAStatus [4x1 boolean] Indicates which rate sensors are enabled. [N/A]
ConfigStatus [1x1 uint8] Indicates what the reason was for this report [Command (0), Internal Event(1)]
Alarms [TBDx1 boolean] alarm indicators [N/A]
)

FaultDetectionReport (
OldAlarm [TBDx1 boolean] alarm indicators before event [N/A]
NewAlarm [TBDx1 boolean] alarm indicators after event [N/A]
)

LostAttitudeReport [event] Indicates that attitude is not considered stable around setpoint [N/A]
ReachedSetPointReport [event] Indicates that attitude is considered stable around setpoint [N/A]
EndofUnloadingReport [1x1 uint8] Reason for ending Unloading [Finished (0), Command(1), HighAltitude (2)]

Masked Parameters

None.

Output

ConfigurationReport(
CurrentMode [1x1 uint8] Indicates current mode of ACS [UNLOADING (0), COARSE (1), FINE (2), SAFE (3), STANDBY (4)]
CurrentCtrl [1x1 uint8] Control mode [UNLOADING (0), COARSE (1), FINE (2), SAFE (3), STANDBY (4)]
CurrentAD [1x1 uint8] Navigation mode [RAW (0), COARSE (1), FINE (2), SAFE (3)]
CurrentFD [1x1 uint8] Indicates current fault detector, [TBD]
AutonomyStatus [1x1 uint8] Indicating autonomy level [Off(0), On(1)]
GuidanceStatus [1x1 uint8] Indicating which mode Guidance is in [Off(0), CoarseGuidance(1), FineGuidance(2)]
RWAStatus [4x1 boolean] Indicates which wheels are enabled. [N/A]
TRQStatus [3x1 boolean] Indicates which torquers are enabled. [N/A]
MAGStatus [1x1 boolean] Indicates if magnetometer is enabled. [N/A]
SSAStatus [6x1 boolean] Indicates which sun sensors are enabled. [N/A]

RGAStatus [4x1 boolean] Indicates which rate sensors are enabled. [N/A]
ConfigStatus [1x1 uint8] Indicates what the reason was for this report [Command (0), Internal Event(1)]
Alarms [TBDx1 boolean] alarm indicators [N/A]
)

FaultDetectionReport (
OldAlarm [TBDx1 boolean] alarm indicators before event [N/A]
NewAlarm [TBDx1 boolean] alarm indicators after event [N/A]
)

LostAttitudeReport [event] Indicates that attitude is not considered stable around setpoint [N/A]
ReachedSetPointReport [event] Indicates that attitude is considered stable around setpoint [N/A]
EndofUnloadingReport [1x1 uint8] Reason for ending Unloading [Finished (0), Command(1), HighAltitude (2)]

Operations
On the events defined above, information is passed on to the external PUS Service Layer.

State Diagram
N/A

Component User Manual
All inputs can be offered to the component at any time in the form of an event.

Abstract

Fine-pointing control performance is crucial for the ability of the Rømer Satellite to maintain accurate observation of faint stars over longer periods of time. Pointing accuracy requirements are high on this mission, and several sources could contribute to derate performance. Measurement noise, bias, drift and temperature driven misalignments between satellite instruments could seriously limit the quality of scientific data. Main sources analysed in the report include sensor noise, reaction wheel imperfections and satellite parameter uncertainty.

The control system is analysed in a discrete time implementation, the performance of the system is analysed and the sensitivity to disturbances and measurement related noise are scrutinized, based on quoted performance of reaction wheels and rate gyros. Effects of sampling time fluctuation, satellite inertia uncertainty and alignments contribute to the build requirements of the satellite. The overall finding is that the fine-pointing requirements of the MONS experiment can be met with adequate margin when reasonable design requirements are met. The major uncertainty that remains is the magnitude of wheel torque disturbance.

Fine Pointing Control for the Rømer Satellite System Definition Phase 1

Bent Ziegler, Mogens Blanke and Henrik Niemann

August 19, 2002

About this document

Purpose

The accuracy and quality in fine-pointing of the Rømer satellite's main telescope towards a selected star under observation is crucial to the success of the mission. This study was undertaken to design a fine-pointing controller of the required high performance, to analyze the effects of various sources of noise and torque disturbances, and to validate the design over an envelope of uncertainty in satellite parameters.

Scope

The report derives a general model for an inertial pointing satellite at an arbitrary point of operation. The point of operation includes stored momentum, quaternion and angular velocity about any axis. A fine-pointing controller is suggested based on quaternion arithmetic, that is robust to large initial offset in pointing, and in fact, maps into the preferred maneuvering controller when gains are reduced to maneuvering values. An architecture for integral control is suggested that effectively deals with possible alteration in misalignment between the main telescope and the ACS star tracker. Performance of the controller is analyzed against measurement noise from star imager, rate gyro sensors, attitude determination subsystem and from reaction wheel torque disturbances. Parameter uncertainty in satellite inertia, wheel alignment and from OBC sampling time fluctuation are analyzed.

Accurate data from wheels have not been included as these data have not been made available from the supplier. Measurements of wheel disturbance torque, demand quantization, and time-responses over the envelope of operation, were requested.

Structural flexibility has not been analysed in this report but must be included in a final validation exercise before the controller is left to care for the main experiment of the satellite.

Contractual aspects

This document is a formal deliverable of WP 3620 (DTU) of the Rømer Satellite Definition phase 1.

Copyright and other rights reside with DTU and the authors, as stated in the Rømer Definition Phase Agreement of Collaboration between Aalborg University and Technical University of Denmark.

Maintenance

The present document is maintained by DTU.

Inclusion of these results of this document into a larger ACS design document is envisaged. The latter document will be maintained by its originating organization (AAU).

Copyright

1 Dynamics for Satellite with Wheel Actuation

This section discuss the modelling of wheels and the various imperfections that need to be included for assessment of high precision pointing accuracy. The reaction wheel assembly is the largest anticipated disturbance source on the satellite, so a detailed modelling is required of disturbance torques generated by the wheels and other imperfections associated with control of the wheels.

1.1 Rotational dynamics with static unbalanced wheel

The usual formulation of Euler's moment equation relates external moment $\mathbf{N}_{ext}$ on the satellite to the derivative in body frame of total angular momentum $\mathbf{L}$,

$$\left.\frac{\partial \mathbf{L}}{\partial t}\right]_b = \mathbf{N}_{ext} - \boldsymbol{\omega} \times \mathbf{L} \tag{1}$$

where $\boldsymbol{\omega}$ is the angular velocity of a (rotating) coordinate system fixed in the satellite's body.

Assessment of wheel imperfections is done considering a rigid body and some moving mass, say a particle of mass m_i located in the body by r_i , which is measured relative to the centre of the chosen coordinate system. The absolute velocity of the particle is

$$v_i = v_o + \omega \times r_i + \frac{\partial r_i}{\partial t} \tag{2}$$

where v_o is the velocity of the centre of the coordinate system and ω the angular velocity of the coordinate system. The angular momentum over an envelope of such masses become

$$\mathbf{h} = \int_w r_i \times (v_o + \omega \times r_i + \frac{\partial r_i}{\partial t}) dm \tag{3}$$

$$\mathbf{h} = \int_w r_i \times (\omega \times r_i) dm - v_o \times \int_w r_i dm + \int_w r_i \times \frac{\partial r_i}{\partial t} dm \tag{4}$$

The first term is by definition the inertia tensor

$$\mathbf{J}_w = \int_w (r_i \times \omega \times r_i)\, dm \tag{5}$$

The second term is zero if the coordinate system is taken in the centre of mass. If this is not the case, we get the term

$$-v_o \times \int_w r_i dm = -v_o \times \tilde{r} M_w \tag{6}$$

where M_w is the mass of the wheel being analysed.

An ideal wheel rotating about its z axis, having with inertia radius R_w has the inertia tensor

$$\mathbf{J}_w = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \frac{M_w R^2}{4} \tag{7}$$

A reaction wheel rotating about its z axis has the angular momentum $\mathbf{h}_w = (0, 0, h_w)^T$ in its own axis system. Being aligned according to the mounting of the wheel, in a direction that has a direction cosine matrix $\mathbf{A}_{bw}$, the wheel orientation in satellite coordinates is $\mathbf{e}_b = \mathbf{A}_{bw}\ (0, 0, 1)^T$. The angular momentum of the wheel, seen in body coordinates, is $\mathbf{h}_b = \mathbf{A}_{bw}\ \mathbf{h}_w = \mathbf{A}_{bw}\ (0, 0, h_w)^T$.

1.2 Wheel with static imbalance

If the wheel has a static imbalance, characterised by a small mass dm placed at a radius r_s, then $\mathbf{r}_i = r_s(\sin(nt), \cos(nt), 0)$ and the total mass M_w, then

$$\frac{\partial r_i}{\partial t} dm = r_s(n\cos(nt), -\mathbf{n}\sin(nt), 0) dm \tag{8}$$

and

$$\cos(\omega t) \tag{9}$$

$$\int_w r_i \times \frac{\partial r_i}{\partial t} dm = r_s{}^2 \left(\int_w \begin{bmatrix} 0 & 0 & \cos(nt) \\ 0 & 0 & -\sin(nt) \\ -\cos(nt) & \sin(nt) & 0 \end{bmatrix} \begin{bmatrix} \cos(nt) \\ -\sin(nt) \\ 0 \end{bmatrix} dm \right) \omega \tag{10}$$

$$= -r_s^2 M_w \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} n \tag{11}$$

This term is equivalent to a change of inertia by $-r_s^2 M_w$.

The force generated by a wheel imbalance of mass m_s then amounts to

$$\mathbf{F}_w = m_s \frac{\partial^2 \mathbf{r}_i}{\partial t^2} = m_s r_s n^2 \begin{bmatrix} -\sin(nt) \\ -\cos(nt) \\ 0 \end{bmatrix} \tag{12}$$

The torque generated on the satellite is, when the wheel is located in the satellite by $\mathbf{r}_w$, measured from the satellite coordinate system's centre to the centre of the wheel,

$$\mathbf{N}_{dist} = \mathbf{r}_w \times \mathbf{F}_w = \mathbf{r}_w \times \mathbf{A}_{bw} m_s r_s n^2 \begin{bmatrix} -\sin(nt) \\ -\cos(nt) \\ 0 \end{bmatrix} \tag{13}$$

This term shows the resulting torque variation on the satellite due to wheel imperfection, which is seen to act like an external torque disturbance at the wheel rotational speed.

Now consider the angular momentum of a wheel with speed n_w relative to the satellite (about the wheels' third axis), then the absolute angular velocity of the wheel is

$$\mathbf{\Omega}_{wi} = \mathbf{A}_{wb}\ \omega_{bi} + (0,0,n)^T \tag{14}$$

The nominal wheel inertia is

$$\mathbf{J}_w = \begin{bmatrix} J_{rr} & 0 & 0 \\ 0 & J_{rr} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \frac{M_w R^2}{4}$$

With satellite inertia $\mathbf{I}_{so}$(without wheels) and angular velocity in inertial space ω_{ib}, we have

$$\mathbf{L}{=}\mathbf{I}_{so}\ \omega_{ib}+\sum_{i=1}^{k}\ \mathbf{A}_{bw}\ \mathbf{J}_w \mathbf{\Omega}_{wi} \Longleftrightarrow \tag{15}$$

$$\mathbf{L} = \mathbf{I}_{so}\ \omega_{ib}+\sum_{i=1}^{k}\ \mathbf{A}_{bw}\ \mathbf{J}_w \mathbf{A}_{wb}\ \omega_{bi} + \sum_{i=1}^{k} \mathbf{A}_{bw}(0,0,J_{zz}n)^T \tag{16}$$

The rigid body dynamics of the satellite with non-spinning wheels is by definition

$$\mathbf{I}_s = \mathbf{I}_{so}+\sum_{i=1}^{k}\ \mathbf{A}_{bw}\ \mathbf{J}_w \mathbf{A}_{wb}$$

hence

$$\mathbf{L} = \mathbf{I}_s\ \omega_{ib}+\sum_{i=1}^{k} \mathbf{A}_{bw}(0,0,J_{zz}n)^T$$

1.3 Dynamic wheel imbalance

Dynamic imbalance of a wheel means the inertia tensor is not diagonal. This means $\mathbf{J}_w = \tilde{A}_w \mathbf{J}_{wo} \tilde{A}_w^T$ where $\tilde{A}_w$ is a rotation matrix representing the tilt of the inertia tensor caused by dynamic imbalance. Let the dynamic imbalance be specified as contributions from two (small) masses m_d located with an axial offset of plus and minus $0.5z_d$ from the centre of gravity of the wheel. The dynamic imbalance of the wheel will then give rise to angular accelerations perpendicular to the wheel's z axis such that the angular momentum of the wheel is [1]

$$\dot{\mathbf{h}}_w = \begin{bmatrix} 0 \\ 0 \\ J_{zz}\dot{n} \end{bmatrix} + \begin{bmatrix} 2m_d z_z n^2 \cos(nt) \\ 2m_d z_z n^2 \sin(n_w t) \\ 0 \end{bmatrix} \tag{17}$$

The last term is seen to be equivalent to a contribution to the disturbance torque on the satellite

$$\mathbf{N}_{w,dwd} = 2m_d z_z n^2 \begin{bmatrix} \cos(nt) \\ \sin(nt) \\ 0 \end{bmatrix} \tag{18}$$

which is again a disturbance torque proportional in amplitude to wheel speed squared and varying in time as the wheel rotational speed. The torque acting on the satellite is

$$\mathbf{N}_{s,dwd} = \mathbf{A}_{bw} 2 m_d z_z n^2 \begin{bmatrix} \cos(nt) \\ \sin(nt) \\ 0 \end{bmatrix} \tag{19}$$

1.4 Wheel suspension eigenmodes

The above torques assumes a rigid suspension of the wheel. Any eigenmodes of the shaft within the range of rotational wheel speed will amplify the wheel motion associated with the static and dynamic disturbance torques, and generate a resulting disturbance torque on the satellite that is a factor larger than the values calculated from Eq. 19.

This effect is noted but not further considered in this phase of the study. Barred ranges for wheels in the torque allocation algorithm could be utilized to avoid the ranges of wheel speed where wheel shaft resonance amplify the wheel disturbance torques.

1.5 Wheel torque

Driving of the wheel is accomplished by an electrical torque motor, to give a nominal behaviour

$$\dot{h}_z = -N_{wcmd} \tag{20}$$

However, rotor torque generation wary with position, friction torque can depend on wheel position θ and generation of electrical torque has several sources of imperfection. Wheel speed is hence

$$\dot{h}_z = -\eta(\theta_w) N_{wcmd} - f_w(\theta_w, n_w) \simeq -N_{wcmd}(t) - N_{dist1}(t) \tag{21}$$

The approximation implies that the torque and friction imperfections, which do indeed couple to the angular velocity dynamics of the satellite, are considered to only act as time-varying disturbance, not correlated with the satellites motion. This is considered adequate for linear analysis. A high fidelity simulation should confirm this assumption.

In summary, the dynamic equations of a satellite with k wheels is described by

$$\frac{d(\mathbf{I}_s\omega_{ib})}{dt} = \mathbf{N}_{ext} - \omega_{ib} \times (\mathbf{I}_s\omega_{ib}) - \omega_{ib} \times \sum_{i=1}^{k} \left[\mathbf{A}_{bwi}\tilde{A}_{wi}(0,0,h_{zi})^T \tilde{A}_{wi}^T\right] \tag{22}$$

$$+ \left(\sum_{i=1}^{k} \left[\mathbf{A}_{bw}\tilde{A}_w \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tilde{A}_w^T N_{wcmd} \right]_i \right)$$

$$+ \left(\sum_{i=1}^{k} \left[\mathbf{A}_{bw} \left(2m_d z_d\right) n^2 \begin{bmatrix} \cos(nt) \\ \sin(nt) \\ 0 \end{bmatrix} \right]_i \right)$$

$$- \left(\sum_{i=1}^{k} \left[r_w \times \mathbf{A}_{bw} m r_s n^2 \begin{bmatrix} \sin(nt) \\ \cos(nt) \\ 0 \end{bmatrix} \right]_i \right)$$

1.6 Dynamic equation of motion with disturbances

We abbreviate this to the standard form, where $\mathbf{h}$ is the total angular momentum of wheels seen in the satellite's coordinate system, $\mathbf{N}_{dist}$ the disturbance torque, and again, we consider the imbalance as part of the time-varying function $\mathbf{N}_{dist}(t)$, and consider the coupling with the $\mathbf{h}$ vector of the wheels to be a second order effect that is insignificant. With the abbreviated notation the dynamic equation reads,

$$\dot{\omega} = -\mathbf{I}_s^{-1}\left(\omega \times \mathbf{I}_s\omega\right) - \mathbf{I}_s^{-1}\omega \times \mathbf{h} - \mathbf{I}_s^{-1}\dot{\mathbf{h}} + \mathbf{I}_s^{-1}\mathbf{N}_{ext} + \mathbf{I}_s^{-1}\mathbf{N}_{dist} \tag{23}$$

Writing the cross product as a matrix operation using $\mathbf{S}(\omega)$,

$$\mathbf{S}(\omega) = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{24}$$

gives

$$\dot{\omega} = -\mathbf{I}_s^{-1}\mathbf{S}(\omega)\mathbf{I}_s\omega - \mathbf{I}_s^{-1}\mathbf{S}(\omega)\mathbf{h} - \mathbf{I}_s^{-1}\dot{\mathbf{h}} + \mathbf{I}_s^{-1}\mathbf{N}_{ext} + \mathbf{I}_s^{-1}\mathbf{N}_{dist} \tag{25}$$

The combined nonlinear dynamic model of the satellite is completed by noting that control torque in the body coordinate system is $\mathbf{N}_{ctrl}$ and gives the rate of change of the total angular momentum from wheels,

$$\dot{\mathbf{h}} = -\mathbf{N}_{ctrl} \tag{26}$$

1.7 Kinematics

The kinematics of the satellite, using quaternion parameters for attitude representation, is

$$\frac{d}{dt}\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 & \omega_3 & -\omega_2 & \omega_1 \\ -\omega_3 & 0 & \omega_1 & \omega_2 \\ \omega_2 & -\omega_1 & 0 & \omega_3 \\ -\omega_1 & -\omega_2 & -\omega_3 & 0 \end{bmatrix}\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{27}$$

or, in re-written form

$$\frac{d}{dt}\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = -\frac{1}{2}\begin{bmatrix} 0 & -\omega_3 & +\omega_2 \\ +\omega_3 & 0 & -\omega_1 \\ -\omega_2 & +\omega_1 & 0 \\ \omega_1 & \omega_2 & \omega_3 \end{bmatrix}\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} q_4 & 0 & 0 \\ 0 & q_4 & 0 \\ 0 & 0 & q_4 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \tag{28}$$

we can write the kinematic equation using $q = \left[\mathbf{g}^T, q_4\right]$ where $\mathbf{g} = [q_1, q_2, q_3]^T$ is the first three components of the quaternion,

$$\frac{d}{dt}\begin{bmatrix} \mathbf{g} \\ q_4 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} -\mathbf{S}(\omega) \\ -\omega^T \end{bmatrix}\mathbf{g} + \frac{1}{2}q_4\begin{bmatrix} \mathbf{I}_{3\times 3} \\ 0 \end{bmatrix}\omega \tag{29}$$

1.8 Nonlinear model

The combined dynamic and kinematic equations Eqs. 23 and 29 give, in partly vectorised form

$$\frac{d}{dt}\begin{bmatrix} \omega \\ \mathbf{g} \\ q_4 \\ \mathbf{h} \end{bmatrix} = \begin{bmatrix} -\mathbf{I}_s^{-1}\mathbf{S}(\omega)\mathbf{I}_s\omega - \mathbf{I}_s^{-1}\mathbf{S}(\omega)\mathbf{h} + \mathbf{I}_s^{-1}\mathbf{N}_{ctrl} + \mathbf{I}_\mathbf{s}^{-1}\mathbf{N}_{dist} + \mathbf{I}_s^{-1}\mathbf{N}_{ext} \\ -\frac{1}{2}\mathbf{S}(\omega)\mathbf{g} + \frac{1}{2}q_4\mathbf{I}_{3\times 3}\omega \\ -\frac{1}{2}\omega^T\mathbf{g} \\ -\mathbf{N}_{ctrl} \end{bmatrix} \tag{30}$$

1.9 Limitations of the model

It is noted that the effects of flexibility of the satellite's structure or appendages is not included in this model.

1.10 Effects of rotating reference coordinate system

The effects caused by a rotation of both measured values and reference coordinates for the inertia tensor to a satellite geometry defined coordinate system is achieved by rotating the various quantities in the dynamic equation,

$$\frac{d}{dt}\left(\mathbf{I}_s\omega_p\right) = -\omega_p \times \mathbf{I}_s\omega_p - \omega_p \times \mathbf{h}_p - \dot{\mathbf{h}}_p + \mathbf{N}_{ext,p} + \mathbf{N}_{dist,p} \tag{31}$$

then, rotating to the body system using $\omega_p = \mathbf{A}_{pb}\omega_b$, $\mathbf{h}_p = \mathbf{A}_{pb}\mathbf{h}_b$, $\mathbf{N}_p = \mathbf{A}_{pb}\mathbf{N}_b$, $\mathbf{I}_s = \mathbf{A}_{pb}\mathbf{I}_{sb}\mathbf{A}_{pb}^T$ gives

$$\begin{aligned}\frac{d}{dt}\left(\mathbf{A}_{pb}\mathbf{I}_{sb}\mathbf{A}_{pb}^T\mathbf{A}_{pb}\omega_b\right) = &-\mathbf{A}_{pb}\omega_b \times \mathbf{A}_{pb}\mathbf{I}_{sb}\mathbf{A}_{pb}^T\mathbf{A}_{pb}\omega_b \\ &- \mathbf{A}_{pb}\omega_b \times \mathbf{A}_{pb}\mathbf{h}_b - \mathbf{A}_{pb}\dot{\mathbf{h}}_b + \mathbf{A}_{pb}\mathbf{N}_{ext,b} + \mathbf{A}_{pb}\mathbf{N}_{dist,b}\end{aligned} \tag{32}$$

since $\mathbf{A}\omega \times \mathbf{A}\mathbf{h} = \mathbf{A}(\omega \times \mathbf{h})$, this simplifies to

$$\frac{d}{dt}\left(\mathbf{I}_{sb}\omega_b\right) = -\omega_b \times \mathbf{I}_{sb}\omega_b - \omega_b \times \mathbf{h}_b - \dot{\mathbf{h}}_b + \mathbf{N}_{ext,b} + \mathbf{N}_{dist,b} \tag{33}$$

1.11 Linearized Model

This non-linear equation of motion is linearized in an arbitrary point of operation $(\bar{\omega}, \bar{\mathbf{g}}, \bar{q}_4, \bar{h})$ in order to arrive at a set of linear state space equations. The deviation from steady state (point of linearization) is denoted by a tilde above the variables, $\omega = \bar{\omega} + \tilde{\omega}$, but the quaternion representation of attitude poses a specific problem. With $d\mathbf{g}$ denoting the orientation at time $t + dt$ relative to the attitude at time t, then, since

$$\begin{bmatrix} \tilde{g} \\ \tilde{q}_4 \end{bmatrix} = \begin{bmatrix} dg_1 \\ dg_2 \\ dg_3 \\ dq_4 \end{bmatrix} = \begin{bmatrix} e_1 \sin\left(\frac{1}{2}\omega dt\right) \\ e_2 \sin\left(\frac{1}{2}\omega dt\right) \\ e_3 \sin\left(\frac{1}{2}\omega dt\right) \\ \cos\left(\frac{1}{2}\omega dt\right) \end{bmatrix} \simeq \begin{bmatrix} \frac{1}{2}\omega_1 dt \\ \frac{1}{2}\omega_2 dt \\ \frac{1}{2}\omega_3 dt \\ 1 \end{bmatrix} \tag{34}$$

then $\frac{d}{dt}q_4 = 0$ and $\mathbf{S}(\omega)d\mathbf{g} = \mathbf{0}$ hence

$$\frac{d}{dt}\tilde{\mathbf{g}} = -\frac{1}{2}\mathbf{S}(\omega)\tilde{\mathbf{g}} + \frac{1}{2}\tilde{q}_4 \mathbf{I}_{3\times 3}\omega = \frac{1}{2}\mathbf{I}_{3\times 3}\omega \tag{35}$$

$$\mathbf{h} = \bar{\mathbf{h}} + \tilde{\mathbf{h}}; \quad \frac{d}{dt}\mathbf{h} = \frac{d}{dt}\tilde{\mathbf{h}} \tag{36}$$

The desired form of the linear equation of motion has a state vector $\mathbf{x} = (\tilde{\omega}_1, \tilde{\omega}_2, \tilde{\omega}_3, \tilde{g}_1, \tilde{g}_2, \tilde{g}_3, h_1, h_2, h_3)^T$ and has control input $\mathbf{u} = \mathbf{N}_{ctrl}$. The external and disturbance torques have zero mean. The state space equation is then

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}_u(t)\mathbf{u}(t) + \mathbf{B}_e(t)\mathbf{N}_{ext}(t) + \mathbf{B}_d(t)\mathbf{N}_{dist}(t) \tag{37}$$

where

$$A_{ij} = \frac{\partial f_i}{\partial x_j}; \quad B_{ij} = \frac{\partial f_i}{\partial u_j} \tag{38}$$

and

$$f = \begin{bmatrix} -\mathbf{I}_s^{-1}\mathbf{S}(\omega)\mathbf{I}_s\omega - \mathbf{I}_s^{-1}\mathbf{S}(\omega)\mathbf{h} + \mathbf{I}_s^{-1}\left(\mathbf{N}_{ctrl} + \mathbf{N}_{dist} + \mathbf{N}_{ext}\right) \\ -\frac{1}{2}\mathbf{S}(\omega)\mathbf{g} + \frac{1}{2}q_4\mathbf{I}_{3\times 3}\omega \\ -\frac{1}{2}\omega^T\mathbf{g} \\ -\mathbf{N}_{ctrl} \end{bmatrix} \tag{39}$$

Using symbolic manipulation to calculate the Jacobians Eq. 38, we obtain

$$\mathbf{A} = \begin{bmatrix} \mathbf{I}_s^{-1}\mathbf{A}_{\omega,\omega} & \mathbf{0} & \mathbf{I}_s^{-1}\mathbf{A}_{\omega,h} \\ \frac{1}{2}\mathbf{I}_{3\times 3} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}; \ \mathbf{B}_u = \begin{bmatrix} \mathbf{I}_s^{-1} \\ \mathbf{0} \\ -\mathbf{I}_{3\times 3} \end{bmatrix}; \ \mathbf{B}_d = \begin{bmatrix} \mathbf{I}_s^{-1} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}; \ \mathbf{B}_e = \begin{bmatrix} \mathbf{I}_s^{-1} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{40}$$

where

$$\mathbf{A}_{\omega,\omega} = [\mathbf{A}_{\omega,1}, \mathbf{A}_{\omega,2}, \mathbf{A}_{\omega,3}] \tag{41}$$

where the three columns of $\mathbf{A}_{\omega,\omega}$ are

$$\mathbf{A}_{\omega,1} = \begin{bmatrix} \omega_2 I_{31} - \omega_3 I_{21} \\ -2I_{31}\omega_1 - I_{32}\omega_2 - \omega_3 I_{33} + \omega_3 I_{11} + h_3 \\ 2I_{21}\omega_1 + \omega_2 I_{22} + I_{23}\omega_3 - \omega_2 I_{11} - h_2 \end{bmatrix} \tag{42}$$

$$\mathbf{A}_{\omega,2} = \begin{bmatrix} I_{31}\omega_1 + 2I_{32}\omega_2 + \omega_3 I_{33} - \omega_3 I_{22} - h_3 \\ \omega_3 I_{12} - \omega_1 I_{32} \\ -\omega_1 I_{11} - 2I_{12}\omega_2 - I_{13}\omega_3 + \omega_1 I_{22} + h_1 \end{bmatrix} \tag{43}$$

$$\mathbf{A}_{\omega,3} = \begin{bmatrix} -I_{21}\omega_1 - \omega_2 I_{22} - 2I_{23}\omega_3 + \omega_2 I_{33} + h_2 \\ \omega_1 I_{11} + I_{12}\omega_2 + 2I_{13}\omega_3 - \omega_1 I_{33} - h_1 \\ \omega_1 I_{23} - \omega_2 I_{13} \end{bmatrix} \tag{44}$$

$$\mathbf{A}_{\omega,h} = \begin{bmatrix} 0 & \omega_3 & -\omega_2 \\ -\omega_3 & 0 & \omega_1 \\ \omega_2 & -\omega_1 & 0 \end{bmatrix} \tag{45}$$

and the nominal condition is expressed through the parameters are $\omega = \bar{\omega}$ and $h = \bar{h}$. This linear model accept an arbitrary moment of inertia tensor, which enables subsequent use for both controller design and analysis of sensitivity (robustness) properties. Uncertainties include magnitude and rotation of the inertia tensor and alignment of wheels. The basic dynamic properties change with the resulting angular momentum of the wheels. The changes in linear dynamics could be analysed should the satellite be demanded to rotate along one of its axes, e.g. during manoeuvres. The linear model is then available should quantitative analysis be desired. The nonlinear analysis is limited to stating the more elementary questions of stability.

1.12 Nominal linear model

The following parameters apply for the nominal satellite, in inertial pointing condition, $\omega = \mathbf{0}$

$$\omega = (0,0,0); \quad \mathbf{h} \in [0,1]\, Nms; \quad \mathbf{I}_s = \begin{bmatrix} 14.79 & 0 & 0 \\ 0 & 14.33 & 0 \\ 0 & 0 & 3.53 \end{bmatrix} kgm^2 \tag{46}$$

The nominal model for fine-pointing design is hence

$$\mathbf{A} = \begin{bmatrix} \mathbf{I}_s^{-1}\mathbf{A}_{\omega,\omega} & \mathbf{0} & \mathbf{0} \\ \frac{1}{2}\mathbf{I}_{3\times3} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}; \tag{47}$$

$$\mathbf{B}_u = \begin{bmatrix} \mathbf{I}_s^{-1} \\ \mathbf{0} \\ -\mathbf{I}_{3\times3} \end{bmatrix}; \ \mathbf{B}_d = \begin{bmatrix} \mathbf{I}_s^{-1} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}; \ \mathbf{B}_e = \begin{bmatrix} \mathbf{I}_s^{-1} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{48}$$

with

$$\mathbf{A}_{\omega,\omega} = \begin{bmatrix} 0 & -h_3 & h_2 \\ h_3 & 0 & -h_1 \\ -h_2 & h_1 & 0 \end{bmatrix} \tag{49}$$

All states in the state-space equation are measured on the satellite in normal condition.

2 Fine-pointing controller design

Having attitude parameterised as the attitude quaternion, and angular body rates are further measured, it is advantageous to employ a controller that can use this information directly. The error in pointing direction is expressed as elements of the direction cosine error matrix. The satellite has reached the desired attitude when this matrix becomes the unit matrix. Denote the coordinate system for the target orientation as the unit vectors $(\mathbf{1}_{xT}, \mathbf{1}_{yT}, \mathbf{1}_{zT})$ and the satellite's present direction through $(\mathbf{1}_{xS}, \mathbf{1}_{yS}, \mathbf{1}_{zS})$. The direction cosine matrix for the error, the rotation it takes for the satellite to rotate from the present to the target direction, is

$$\mathbf{A}_E \equiv \mathbf{A}_{T,S} = \mathbf{A}_{T,I}\mathbf{A}_{I,S} = \mathbf{A}_{T,I}\mathbf{A}_{S,I}^T \tag{50}$$

The directional error is seen in the off-diagonal elements of the $\mathbf{A}_E$, matrix, for example $a_{23} = \mathbf{1}_{yS} \cdot \mathbf{1}_{zT}$. The z axis of the satellite can only be aligned into the target direction if this scalar product is zero. The control law shall hence provide a torque that rotates the satellite about is x-axis to give $a_{23} = \mathbf{1}_{yS} \cdot \mathbf{1}_{zT} = 0$.

This control idea is exploited as

$$\mathbf{N}_{ctrl} = \begin{bmatrix} -k'_{xp}(a_{32E} - a_{23E}) + k_{xd}\omega_1 \\ -k'_{yp}(a_{13E} - a_{31E}) + k_{yd}\omega_1 \\ -k'_{zp}(a_{21E} - a_{12E}) + k_{zd}\omega_1 \end{bmatrix} \tag{51}$$

where damping is obtained by feedback from the angular velocity estimate (measurement). Since the pointing error, expressed as the quaternion error is

$$\mathbf{q}_E = \mathbf{q}_S^{-1}\mathbf{q}_T = \begin{bmatrix} q_{T4} & g_{T3} & -g_{T2} & g_{T1} \\ -g_{T3} & q_{T4} & g_{T1} & g_{T2} \\ g_{T2} & -g_{T1} & q_{T4} & g_{T3} \\ -g_{T1} & -g_{T2} & -g_{T3} & q_{T4} \end{bmatrix} \begin{bmatrix} -g_{S1} \\ -g_{S2} \\ -g_{S3} \\ q_{S4} \end{bmatrix} \tag{52}$$

and the direction cosine matrix $\mathbf{A}_E$ is related to this quaternion, (Wertz, 1991) the relationship between the rotation matrix and quaternion gives

$$\mathbf{N}_{ctrl} = \begin{bmatrix} N_{xctrl} \\ N_{yctrl} \\ N_{zctrl} \end{bmatrix} = \begin{bmatrix} k_{xp}g_{E1}q_{E4} + k_{xd}\omega_1 \\ k_{yp}g_{E2}q_{E4} + k_{yd}\omega_1 \\ k_{zp}g_{E3}q_{E4} + k_{zd}\omega_1 \end{bmatrix} \tag{53}$$

Linear quadratic optimal control can be achieved by designing controller gains such that an index is minimized, which is a trade-off between control effort and control error

$$J_{fpc} = \int_0^\infty \left(\begin{bmatrix} \boldsymbol{\omega} & \mathbf{g}_e \end{bmatrix} \mathbf{R}_{xx} \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{g}_e \end{bmatrix} + \mathbf{N}_{ctrl}^T \mathbf{R}_{\mathrm{u}} \mathbf{N}_{ctrl} \right) dt \tag{54}$$

This leads to the general controller

$$\mathbf{N}_{ctrl} = \mathbf{K}_p \hat{\mathbf{g}} \hat{q}_4 + \mathbf{K}_d \hat{\omega} \tag{55}$$

When analysing this multivariable controller, a minimal number of control parameters can be obtained , and simultaneously, the effective linear range of the controller can be enhanced, if system cross couplings are reduced through controller design. Therefore, it is an option to select off-diagonal elements in the control law such that the $\boldsymbol{\omega} \times \mathbf{h}$ term is compensated by the generation of control torque: Therefore the multi-variable design of the fine-pointing control law will be based on a controller structure

$$\mathbf{N}_{ctrl} = \mathbf{K}_p \hat{\mathbf{g}} \hat{q}_4 + \mathbf{K}_p \tau_d \hat{\omega} + \hat{\omega} \times \hat{\mathbf{h}} \tag{56}$$

Finally, pointing errors should be kept zero despite external torque, and we therefore introduce integral action as

$$\mathbf{N}_{ctrl} = \mathbf{K}_p \left(\hat{\mathbf{g}} \hat{q}_4 + \frac{1}{\tau_I} \int_0^t \hat{\mathbf{g}} \hat{q}_4 d\tau \right) + \left(\mathbf{K}_p \tau_d - \hat{\mathbf{h}} \times \right) \hat{\omega} \tag{57}$$

2.1 Controller model for linear analysis

In the linear analysis $q_{4E} = \tilde{q}_4 = 1$, and the controller to be analysed takes the linear form

$$\begin{bmatrix} N_{xctrl} \\ N_{yctrl} \\ N_{zctrl} \end{bmatrix} = \begin{bmatrix} k_{xp} \tilde{g}_1 + k_{xd} \omega_1 \\ k_{yp} \tilde{g}_2 + k_{yd} \omega_1 \\ k_{zp} \tilde{g}_3 + k_{zd} \omega_1 \end{bmatrix} \tag{58}$$

With integral action

$$\begin{bmatrix} N_{xctrl} \\ N_{yctrl} \\ N_{zctrl} \end{bmatrix} = \begin{bmatrix} k_{xp} \left(\tilde{g}_1 + \tau_{xd} \omega_1 + \dfrac{1}{\tau_{xI}} \int_0^t \tilde{g}_1 d\tau \right) \\ k_{yp} \left(\tilde{g}_2 + \tau_{yd} \omega_2 + \dfrac{1}{\tau_{yI}} \int_0^t \tilde{g}_2 d\tau \right) \\ k_{zp} \left(\tilde{g}_3 + \tau_{zd} \omega_3 + \dfrac{1}{\tau_{zI}} \int_0^t \tilde{g}_3 d\tau \right) \end{bmatrix} \tag{59}$$

A state-space model of the controller and system is as follows where the integral state within the controller is $\mathbf{z}$

$$\dot{\mathbf{z}} = [\mathbf{0}]\, \mathbf{z} + \mathbf{I} \tilde{\mathbf{g}} \tag{60}$$

$$N_{ctrl} = \left[\frac{1}{\tau_I} K_p \right] [\mathbf{z}] + \begin{bmatrix} K_p & K_p \tau_d \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{g}} \\ \omega \end{bmatrix} \tag{61}$$

$$\mathbf{K}_p = \begin{bmatrix} 8.5944 & 0 & 0 \\ 0 & 8.5944 & 0 \\ 0 & 0 & 2.0544 \end{bmatrix} \tag{62}$$

$$\tau_I = \begin{bmatrix} 172.0340 & 0 & 0 \\ 0 & 166.7439 & 0 \\ 0 & 0 & 172.6387 \end{bmatrix} \tag{63}$$

$$\tau_d = \begin{bmatrix} 1.2903 & 0 & 0 \\ 0 & 1.2506 & 0 \\ 0 & 0 & 1.2902 \end{bmatrix} \tag{64}$$

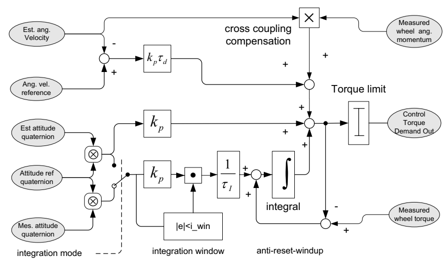


Figure 1: Structure of one axis of the fine-pointing controller with integration window, integrator anti-windup and possible selection integral feedback from measured error quaternion.

2.2 Implementation

The implementation should be made robust to varying sample period and in fluctuations therein, and integrator wind-up must be taken care of. Further, bias is commonly encountered in a position estimate when linear filters (extended Kalman filter) are used on a nonlinear system. This bias can either come from non-perfect compensation in the estimator of external disturbance torques or just from parameter uncertainty problems. Integration of the raw (measured) attitude quaternion error could overcome such bias difficulties, while the filtering of measurement noise is retained on the proportional and derivative parts of the controller. Finally, target attitude (reference) and angular velocity reference are input to the controller. The inclusion of a reference angular velocity in the controller is motivated by generality.

The structure of the fine pointing controller is shown in Figure 1.

3 Fine Pointing Controller Performance

In this section the performance of the continuous fine pointing controller will be analysed. The performance specification is given in the frequency domain, and the analysis is most informative if presented in this domain.

The transfer functions from the various sources of noise and disturbance can be obtained as Laplace domain operators. The pointing accuracy is determined

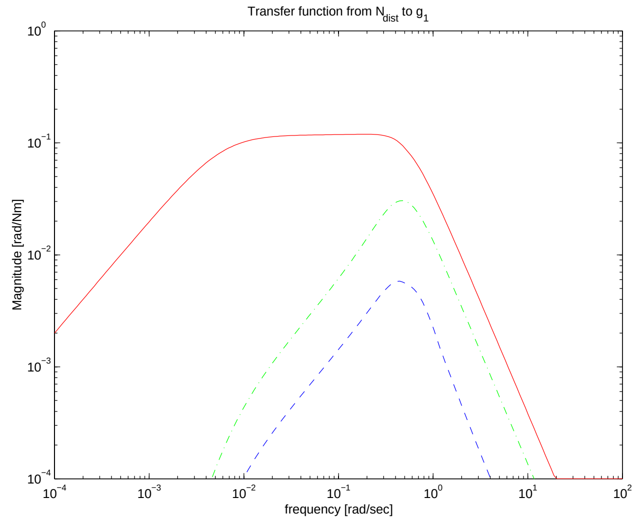


Figure 2: Transfer function (gain) from torque disturbances to pointing error g_1. Solid line is for noise on satellite x-axis, dashed for y-axis and dash-dotted for z-axis

from

$$\tilde{g}(s) = H_{gg}(s)g_n(s) + H_{g\omega}(s)\omega_n(s) + H_{gd}(s)N_{dist}(s) \tag{65}$$

These transfer functions show the effects on the satellite's pointing when subject to measurement or estimation noise g_n and ω_n, and to disturbance torques from wheels and satellite flexibility. The flexibility aspects were not considered in this part of the study.

The transfer functions in Eq. 65 are functions of the satellite equations of motion, Eq. 47 and of the controller Eq. 57. Using nominal parameters for Rømer, we obtain a set of nominal transfer functions from the various disturbances to the deviation from ideal pointing, $\tilde{g}$. Uncertainty of the satellite's inertia tensor will further cause cross-coupling between disturbance in one axis to motion in others. Change of wheel angular momentum within the nominal range for the wheels has a similar effect. Figures 2, 3 and 4 show the maximal transfer functions that exist from noise and torque disturbances to g_1. The maximal cross couplings are obtained using a worst case combination of parameters.

3.1 Disturbance Analysis

3.1.1 Wheel loop analysis

An assessment was made of a number of imperfections within the wheel control loop. These included

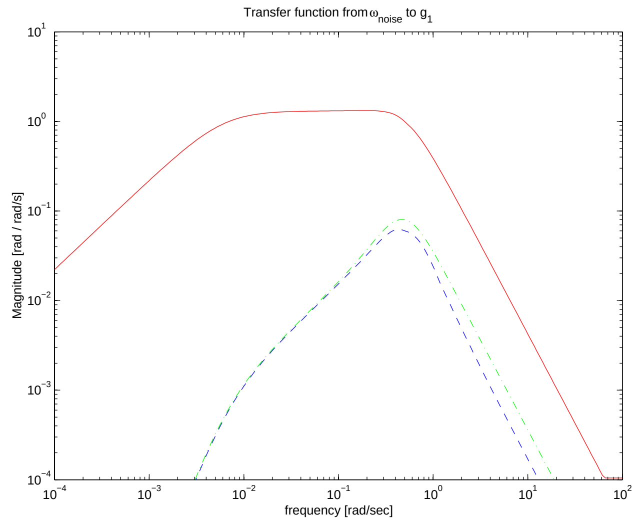


Figure 3: Transfer function (gain) from disturbances in angular velocity estimation to pointing error g_1. Solid line is for noise on satellite x-axis, dashed for y-axis and dash-dotted for z-axis

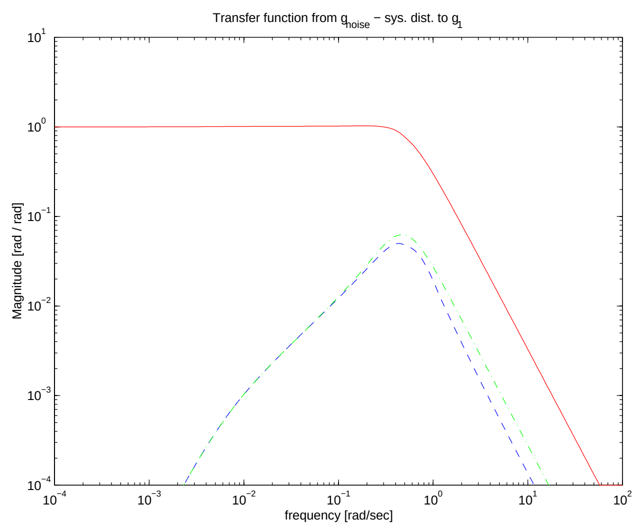


Figure 4: Transfer function (gain) from disturbances in attitude quaternion estimation to pointing error g_1. Solid line is for noise on satellite x-axis, dashed for y-axis and dash-dotted for z-axis

- torque fluctuation due to variation in transistor PWM modulation (turn off phenomena)

- torque fluctuation due to magnetic field variation over a revolution of the wheel

- bearing noise (torque)

- imbalance - static and dynamic

- speed decoding and possible speed/torque quantization from this noise

- toque quantization due to A/D conversion of command

- torque quantization due to resolution in PWM timing

- noise spectrum of sources of random disturbance

The assessment of the various sources listed was quite uncertain due to lack of detailed information from manufacturer(s). Some general information is, however, available from the literature.

It was concluded to consider the deterministic disturbance separately and combine all stochastic effects in the manufacturers specification. The total RMS value of torque noise is below 50×10^{-6} Nm. The spectral density is assumed to be band-limited white noise giving the intensity

$$N_w(s) = \begin{cases} 1.5708 \cdot 10^{-10} \frac{\mathrm{Nm}^2}{\mathrm{Hz}} \text{ for } \omega < 100 \text{ rad/sec} \\ 0 \text{ for } \omega > 100 \text{ rad/sec} \end{cases} \tag{66}$$

Deterministic disturbances will occur from wheel imbalance. These motions associated with this harmonic varying torque at fairly high frequencies will be aliasing in both STR and MONS instruments. Since integration of the optical instruments is physical, filtering can not be applied.

3.1.2 Estimation noise

Estimation of attitude quaternion and angular velocity is performed by a linear Kalman filter based on measurements from STA and RGA. The spectra for estimation errors of the three imaginary parts of the quaternion and angular velocity are shown on Figure 6 where the small angle approximation $\sin\Phi = \Phi$ has been used. The spectra are approximated by the transfer functions

$$S_{yy}(s) = H_{yx}(s)H_{yx}^*(s)S_{xx}(s) \tag{67}$$

where

$$\begin{aligned} H_{ng1}(s) &= \tfrac{1.3(s+3)}{s+0.3} & H_{ng2}(s) &= \tfrac{1.3(s+3)}{s+0.3} \\ H_{ng3}(s) &= \tfrac{1.5(s+2)}{s+0.05} \\ H_{nw1w2}(s) &= \tfrac{0.4(s+3)}{s+0.2} & H_{nw2}(s) &= \tfrac{0.4(s+3)}{s+0.2} \\ H_{nw3}(s) &= \tfrac{0.06(s+3)}{s+0.03} \end{aligned} \tag{68}$$

See Appendix A for further details for the derivation of spectra.

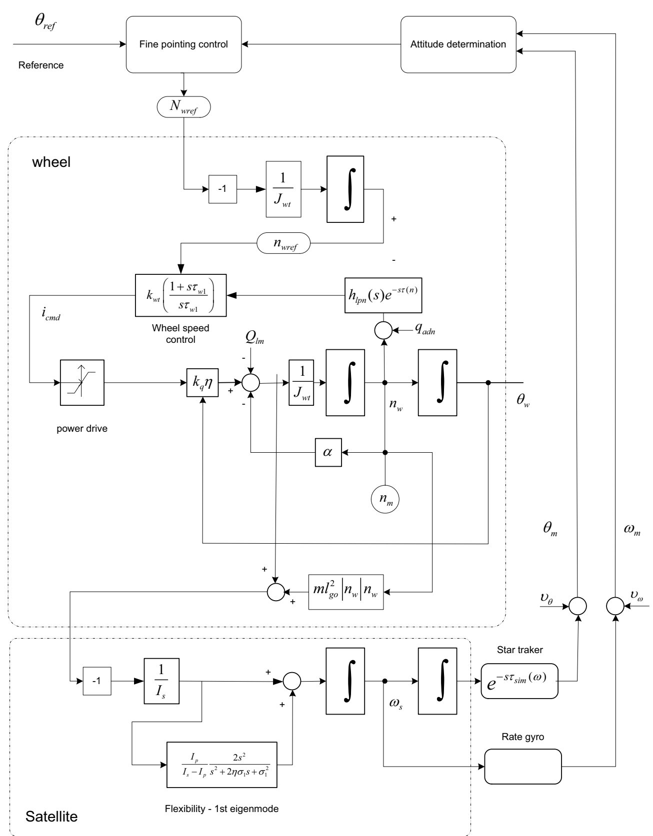


Figure 5: Single degree of freedom illustration of dynamics between wheels and satellite.

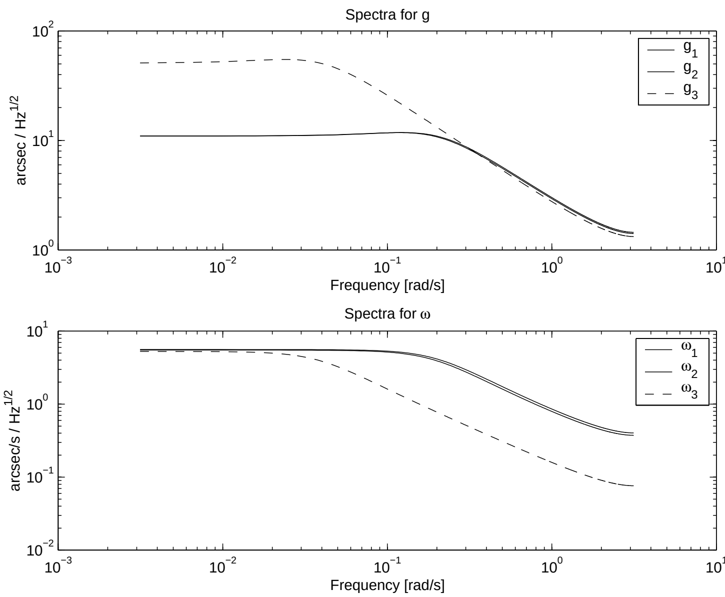


Figure 6: Transfer function used for noise generation for attitude and angular velocity

3.1.3 Results

Deterministic Disturbances from Wheels The effects of wheel imbalance are plotted in Figure 7. Results show a wide margin (about 500) to the specified maximum. Since resonances in wheel suspension or satellite structure can be poorly damped, certain frequencies could be amplified by a factor of 15-100 without reduction on scientific performance. We still have a margin to specify wheels with less minute balancing without consequence for the scientific quality.

The figures specified for the new Dynacon wheels are a factor of 10 smaller than reported from other wheels (Teldix), but even an optimistic specification will be accommodated within the large margin.

Stochastic Disturbances

The effects of the total noise from RWA (Eq. 66), RGA and STA (Eq. 68) on satellite attitude in the continuous system are shown on Figures 8 - 10. All requirements are met for all frequencies and the RMS values lie within the specifications with a factor of at least 14.7 (g_2).

3.2 Uncertainty Analysis

The linear satellite model include a number of parameters that are not constant. Further, some of the parameters are not known exactly. Both things result in that the linear model of the satellite is not known exactly. These parameter variations need to be included in the analysis of the closed-loop performance of

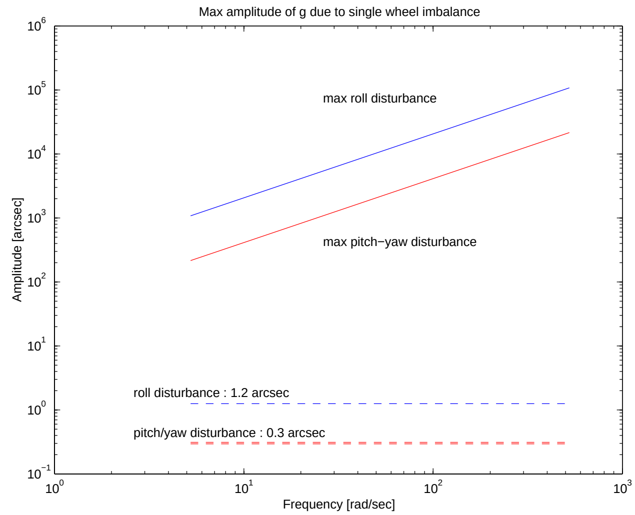


Figure 7: The effects of wheel imbalance

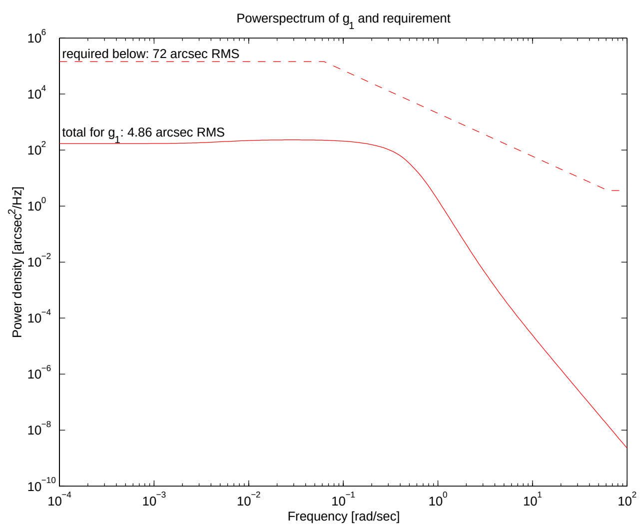


Figure 8: Power spectrum for g_1 and the performance specifications

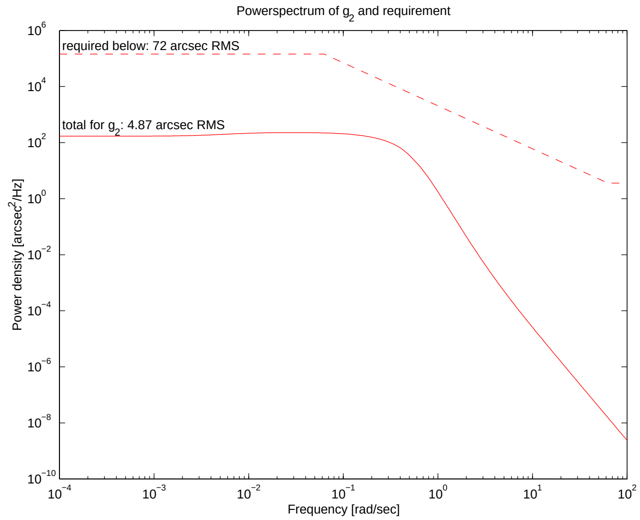


Figure 9: Power spectrum for g_2 and the performance specifications

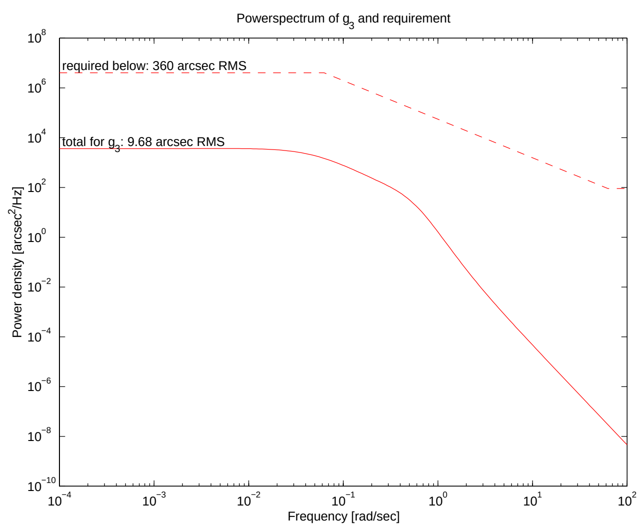


Figure 10: Power spectrum for g_3 and the performance specifications

the control system.

The angular momentum h_i in all three axis are parameters that are not constant, magnitude and principal axis of the inertia tensor I_s are not known exactly.

Due to the spinning up of the wheels, the angular momentum can increase from 0 Nms to maximum at 0.5 Nms until momentum unloading takes place. The nominal value used in the linear model for the angular momentum is 0 Nms for all 3 axes. The A matrix in the linear model Eq. 37 is then given by

$$A = \begin{bmatrix} I_s^{-1} A_{\omega,\omega} & 0 & 0 \\ \frac{1}{2} I_{3\times 3} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

where I_s is the inertia tensor and

$$A_{\omega,\omega} = \begin{bmatrix} 0 & -h_3 & h_2 \\ h_3 & 0 & -h_1 \\ -h_2 & h_1 & 0 \end{bmatrix}$$

with $h_i \in [0\ Nms,\ 0.5\ Nms]$

The variation with h_i could be accommodated in the fine pointing controller, according to Eq. 1, but it is useful to see whether such a compensation is necessary.

The inertia tensor is not known exactly in the linear model. The diagonal elements of the nominal inertia tensor are assumed to be known with an accuracy of $\pm 10\%$. The inertia tensor is then given by

$$I_{diag} = diag(I_{11}(1+\Delta_{11}), I_{22}(1+\Delta_{22}), I_{33}(1+\Delta_{33}))$$

where I_{ii} are the nominal diagonal elements in the inertia tensor and Δ_{ii} are the inertia uncertainties in the 3 directions. Δ_{ii} is bounded by

$$|\Delta_{ii}| \leq 0.1$$

Further, the direction of the inertia tensor is also uncertain. It is assumed that the inertia tensor could be rotated within ± 0.1 rad., around any of the principal axis of the nominal inertia tensor . The true inertia tensor is hence given by

$$I_s = A_{rotation} \times I_{diag} \times A_{rotation}^T$$

where $A_{rotation}$ is the rotation matrix given by

$$A_{rotation} = \begin{bmatrix} 1 & \psi & -\theta \\ -\psi & 1 & \phi \\ \theta & -\phi & 1 \end{bmatrix}$$

for small rotations. It is assumed that the 3 angles are bounded by:

$$\begin{aligned} |\psi| &\leq 0.1\ rad \\ |\theta| &\leq 0.1\ rad \\ |\phi| &\leq 0.1\ rad \end{aligned}$$

All together, the model include 9 parameters that are uncertain, the angular momentum in 3 axis, the 3 diagonal inertia parameters in the inertia tensor I_{diag} and the 3 rotation angles for the rotation of the inertia tensor.

The analysis of the robustness/performance of the closed loop system can be done in two ways. The analysis can be done directly by evaluating the closed loop system in a number of working points or the analysis can be done by using analytic norm based methods. The first approach was taken when calculating the plots shown in Figures 2 - 4 and 8 - 10. The closed loop system was evaluated in 19683 (3^9) points, varying the 9 uncertain parameters to obtain the worst case situation. It was shown that the performance conditions are satisfied at the entire boundary of assumed parameter variation. However, this approach will not guarantee that there does not exist a combination of points of the parameter space that could reduce the closed loop performance.

Using a norm based analysis method, it is possible to give guarantee for robust performance of the closed loop. However, such analysis is usually conservative due to the way uncertainty is modelled. The analysis is applied to indicate if there is any robustness problems in the closed loop system and where these problems could be. The technical means to analyse the robustness is a plot of singular values.

1. Uncertainty model for inertia

In connection with the robustness analysis of the system, it does not make sense to consider the 3 axis separately. Therefore, the analysis is derived for the overall system. This means that all 3 axes are included in the analysis.

A single full uncertain block is selected to describe the model uncertainty in the linear satellite model. We describe the uncertainty as an inverse multiplicative uncertainty block at the output of the system,

$$G(s) = (I + \Delta(s))^{-1} G_0(s)$$

where $G(s)$ is the real system, $G_0(s)$ is the nominal system and $\Delta(s)$ is the uncertain block.

In the case when only the elements in the inertia tensor are uncertain, an upper bound for the Δ block can be calculated to:

$$\bar{\sigma}(\Delta(j\omega)) \leq 0.17, \forall\omega$$

2. Uncertainty from variations in h

Regarding the variations of h_i, it turns out that using a single block for the description is indeed conservative. The variation of the angular momentum h_i in the linear satellite model will introduce two pure imaginary poles. When the variation of h_i is included in the calculation of Δ, the upper bound will increase, in particular at low frequencies. The bound is above 1 at low frequencies, this means there could be more that 100% model error. A more detailed description of the uncertainties in the system is hence needed to obtain non conservative bounds. Therefore, it is not useful to use the bound for h in this robustness analysis.

3. Analysis

The analysis is derived by calculating a suitable norm of the closed transfer function matrix. The setup for the analysis is shown in Figure 11. The performance specifications is given by the transfer function matrix between the input vectors $\tilde{N}_{dist}$, $\tilde{N}_{noise}$ to the output vector y. W_{dist} and W_{noise} are weight matrices that specify the performance requirement. The transfer function matrix between the input vector v_1 and the output vector y gives the robust stability condition. W_Δ is the weighting matrix for the perturbation block Δ_s, where Δ is given by

$$\Delta = W_\Delta \Delta_s, \ \ \|\Delta_s\| \leq 1 \ \forall \omega$$

W_Δ is constant.

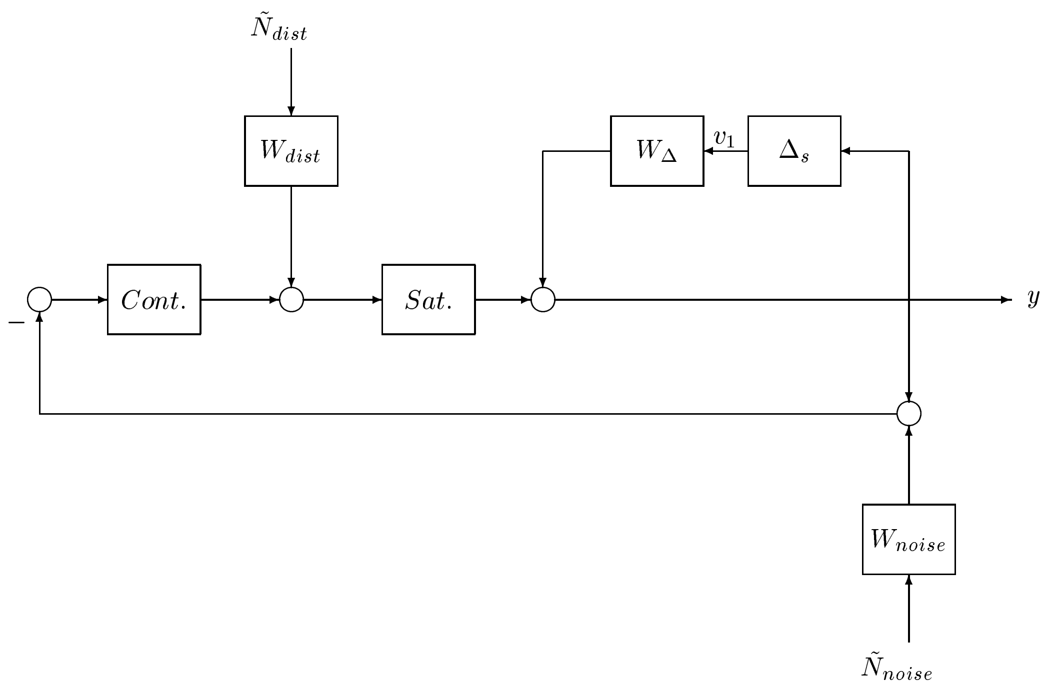


Figure 11: Setup for robust performance analyse

The analysis can now be derived by calculating a suitable norm of the complete closed loop transfer function matrix as function of the frequency. The used norms are the induced 2-norm and the induced infinity norm. The induced 2-norm (also known as the ∞-norm) is normally used in connection with robustness analysis. The induced infinity norm is closely related with the spectral analysis shown earlier in this report. It is defined as the maximal row sum of a matrix.

The system satisfy robust performance within the requirement if the norm of closed loop transfer function matrix is smaller than 1 for all frequencies.

An analysis of the nominal performance ($W_\Delta = 0$) is shown in Figure 12. It is seen that both the induced 2-norm and the induced infinity-norm (as function of the frequency) of the closed loop system are below 1 at all frequencies, as desired. The maximal value of the curves is found around 0.4 rad/sec. This is in accordance with the power spectrum in Figures 8 - 10.

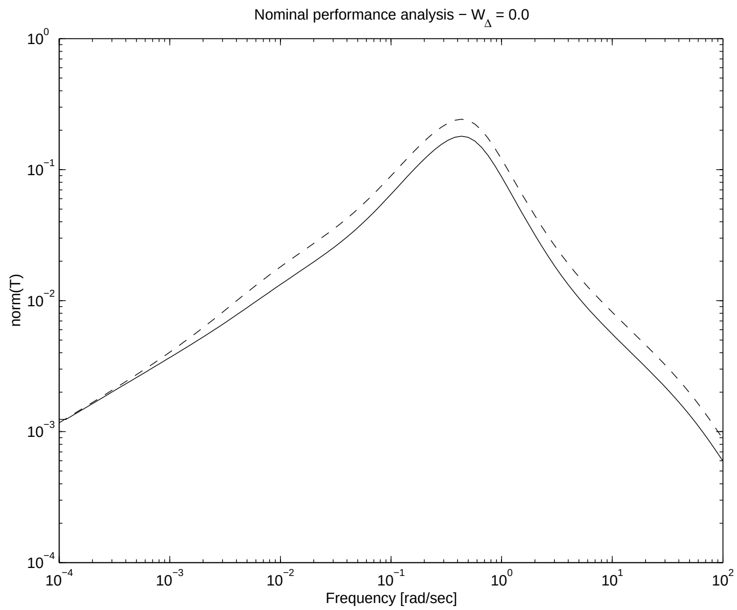


Figure 12: Nominal performance analysis of the closed loop system. The induced 2-norm (solid line) and the induced infinity norm (dashed line) are shown for the system.

An analysis of robust performance is shown in Figure 13 for

$$W_\Delta = 0.2$$

Both curves are again below 1 which indicates that robust performance is obtained for an inverse multiplicative perturbation at the output with magnitude of 0.2. A larger uncertainty will cause the curve for induced infinity norm to exceed 1 so $W_\Delta = 0.2$ is the maximum uncertainty for which robust performance is obtained.

In conclusion, this robustness analysis shows that we can guarantee robust performance for all variations in the inertia tensor.

Regarding the variations in the angular momentum, it is not possible to guarantee robust performance from results of this analysis. However, it is possible to remove this effect entirely by adding the suggested compensation term in the controller, Eq. 1. It is hence necessary to include the compensation term $-\omega \times h$ in the fine-pointing controller.

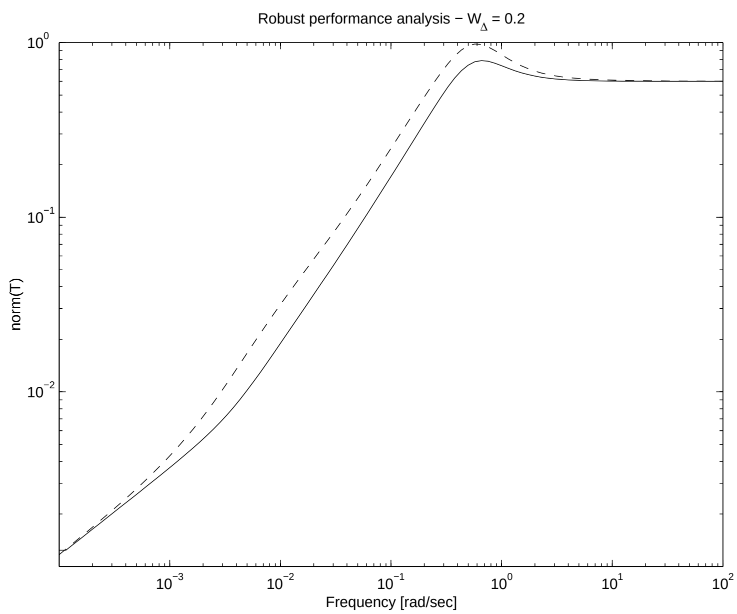


Figure 13: Robust performance analysis of the closed loop system. The induced 2-norm (solid line) and the induced infinity norm (dashed line) are shown for the system.

4 Analysis of Discrete Fine Pointing Controller

The discretisation is based on the continuous control system designed in section 2 and the linear plant model from section 1. The linear plant model is discretised using a zero order hold network and the controller is discretised using Tustin approximation. The sampling time is $T = 1sec$.

The analysis will be performed with respect to

- Nominal stability
- Nominal performance
- Robust stability
- Robust performance

The functional block diagram used for the analysis is shown on Figure 14

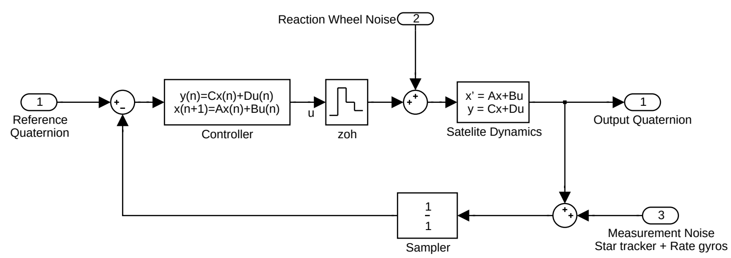


Figure 14: Functional block diagram for discrete controller analysis. measurement noise is from the attitude determination.

4.1 Nominal plant

Nominal stability is examined by observing the poles in the discrete closed loop transfer function

$$\boldsymbol{T}(z) = \boldsymbol{G_o}(z)\boldsymbol{K}(z)\left(\boldsymbol{I}_{6\times 6} + \boldsymbol{G_o}(z)\boldsymbol{K}(z)\right)^{-1} \tag{69}$$

where

$$\boldsymbol{G_o}(z) = z\{\boldsymbol{G}(s)\boldsymbol{G}_h(s)\} \tag{70}$$

$$\boldsymbol{G}_h(s) = \frac{1 - e^{-sT}}{s}\boldsymbol{I}_{3\times 3} \tag{71}$$

and $\boldsymbol{G}$ is the plant, $\boldsymbol{G_h}$ is a 'zoh'-network, $\boldsymbol{K}$ is the controller and $z\{\boldsymbol{G}(s)\boldsymbol{G}_h(s)\}$ denotes the z-transform of $\boldsymbol{G}(s)\boldsymbol{G}_h(s)$. All poles for this closed loop system lie within the unit circle.

4.2 Estimation noise

Estimation noise propagates through a system with both continuous and discrete states. An expression for the influence from estimation noise to continuous output $g(s)$ is

$$\boldsymbol{g}(s) = \boldsymbol{G}_o(s)\left(\boldsymbol{I} + \boldsymbol{K}(z)\boldsymbol{G}_o(z)\right)^{-1}\boldsymbol{K}(z)\boldsymbol{N}_v(z) \tag{72}$$

$$= \boldsymbol{G}_o(s)\boldsymbol{S}(z)\boldsymbol{K}(z)\boldsymbol{N}_v(z) \tag{73}$$

where $\boldsymbol{S}(z) = (\boldsymbol{I} + \boldsymbol{K}(z)\boldsymbol{G}_o(z))^{-1}$ and $\boldsymbol{N}_v(z)$ denotes discrete estimation noise. The spectrum for this transfer function is found by first finding the spectrum for the discrete part of the system, which is then multiplied by the spectrum for the continuous part of the system.

$$S_{tot}(f) = S_{G(s)}(f) \cdot S_{zoh(s)}(f) \cdot S_{SK(z)}(f) \tag{74}$$

Applying the estimation noise spectra from Figure 6 to the total spectrum yields the noise on three imaginary components of the output quaternion in Figure 15.

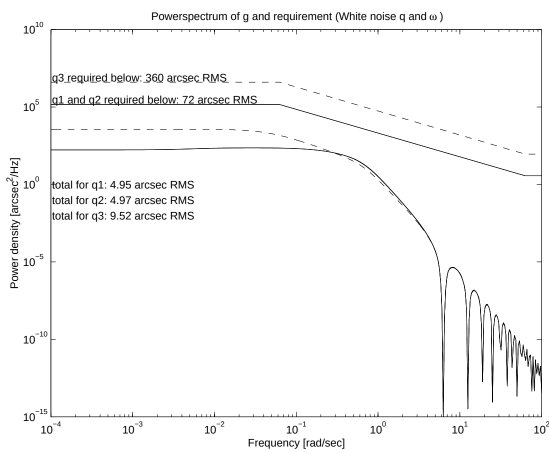


Figure 15: Spectra for g(s) due to estimation noise

4.3 Wheel noise

The output spectrum caused by the reaction wheel noise, $\boldsymbol{N}_w$ is now analysed. For a start the system will be considering the system as an entirely discrete system so the output noise can be found from the expression

$$\boldsymbol{g}(z) = \boldsymbol{S}(z)\boldsymbol{G}_o(z)\boldsymbol{N}_w(z) = \left(\boldsymbol{I} + \boldsymbol{G}_o(z)\boldsymbol{K}(z)\right)^{-1}\boldsymbol{G}_o(z)\boldsymbol{N}_w(z) \tag{75}$$

Applying the wheel noise (Eq. 66) to Equation 75 yields the output spectra in Figure 16. The requirements are met for the three axes with a good margin.

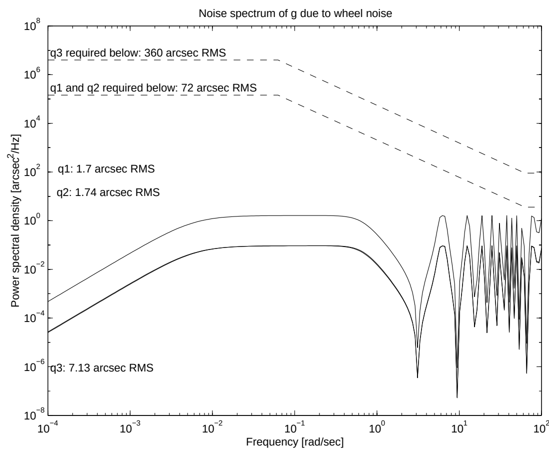


Figure 16: Spectra for g(z) due to wheel noise

However, the noise source is continuous and so is the output. Therefore the relation between a continuous input and continuous output through a discrete system should be considered. This is expressed as

$$\boldsymbol{g}(s) = \boldsymbol{G}(s)\left[\boldsymbol{N}_w(s) - \boldsymbol{G}_h(s)\boldsymbol{K}^*(s)\left(\boldsymbol{I} + \boldsymbol{G}_o(z)\boldsymbol{K}(z)\right)^{-1} z\{\boldsymbol{G}(s)\boldsymbol{N}_w(s)\}\right] \quad (76)$$

where $z\{\boldsymbol{G}(s)\boldsymbol{N}_w(s)\}$ means the z-transform of $\boldsymbol{G}(s)\boldsymbol{N}_w(s)$. This expression yields the Laplace transform of the output signal, but what we need is the output spectrum which is found from the expression

$$S_{gg}(s) = H_{gw}(s)H^*_{gw}(s)S_{ww}(s) \quad (77)$$

where $H(s)$ is the transfer function from wheel noise to output and S_{ww} is the spectrum for wheel noise. It is, however, not possible to extract a transfer function from Eq. 76 that can be complex conjugated and multiplied to the wheel noise spectrum. Instead the square root of the actuator noise is used in Eq. 76 and the output spectrum becomes

$$\begin{aligned} S_{gg}(s) \quad = \quad & \Big|\boldsymbol{G}(s)\Big[\sqrt{\boldsymbol{N}_w(s)} - \boldsymbol{G}_h(s)\boldsymbol{K}^*(s)\left(\boldsymbol{I} + \boldsymbol{G}_o(z)\boldsymbol{K}(z)\right)^{-1} \\ & z\{\boldsymbol{G}(s)\sqrt{\boldsymbol{N}_w(s)}\}\Big]\Big|^2 \end{aligned} \quad (78)$$

The output spectra for the three quaternion components due to wheel noise in a continuous/discrete system are shown in Figure 17. The high values of low frequency noise on the plots are due to numerical limitations in Matlab and should not cause any concern.

A plot showing performance of the nominal system including all disturbances from reaction wheels and estimation noise can be seen on Figure 18. The margins towards the specifications are now reduced to a factor 14.5 compared to the 14.8

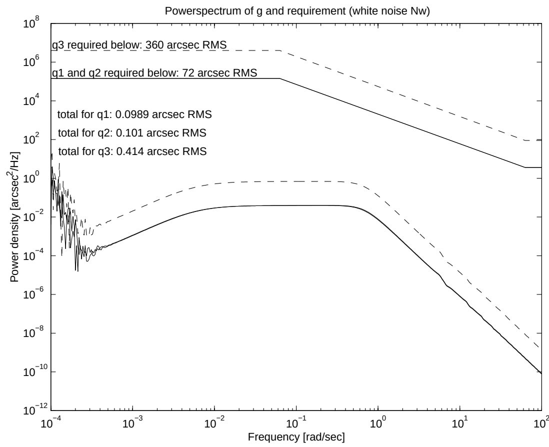


Figure 17: Spectra for y(s) due to wheel noise in a continuous/discrete system

for the continuous system but performance is still obtained with a large margin. Discretisation of the system is thus seen not to have a significant effect on performance for the nominal plant.

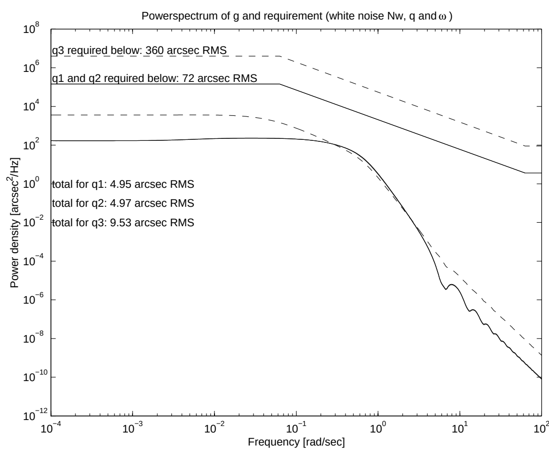


Figure 18: Spectra for g(s) due to wheel and estimation noise in a continuous/discrete system

5 Perturbed Plant

Robust stability is analysed by observing the closed loop pole positions as the inertia matrix for the satellite is varied as described in section 3.2. It is assumed that the term $-\omega \times h$ from the angular momentum of the wheels are counteracted by the control system and can be neglected in the stability analysis. The poles can be seen in Figure 19 where it is seen that they all stay within the unit circle. The same analysis has also been derived by calculating the sampled-data

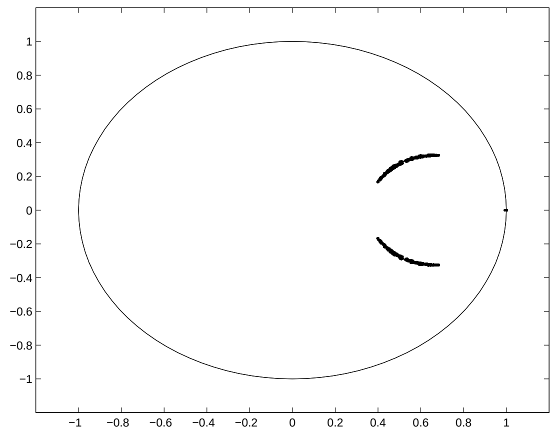


Figure 19: Pole positions as parameters are varied

$\mathcal{H}_\infty$ norm of the controlled satellite (the sampled-data $\mathcal{H}_\infty$ does not exist for unstable sampled-data systems). This analysis verified this result.

For analysis of robust performance the same variations of the inertia matrix as for robust stability are applied. The power spectra for the imaginary components of the output quaternion driven by the noise sources can be seen in Figure 20. The performance requirements for the perturbed plant are met with a margin of a factor 13.4.

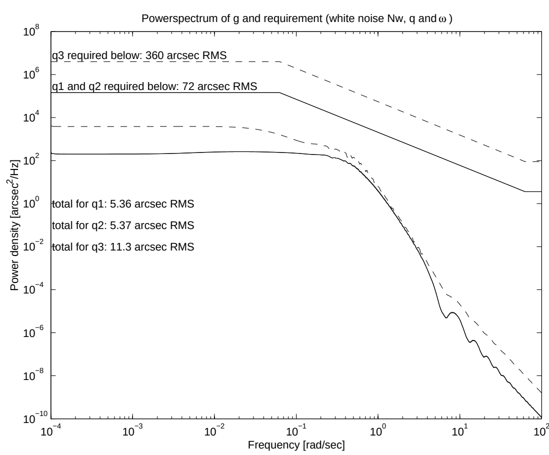


Figure 20: Spectra for g(s) due to actuator and estimation noise in a perturbed continuous/discrete system

The fine pointing controller from section 2 has been discretised and analysed with respect to nominal and robust stability and performance. For analysis the system has been considered as consisting of both continuous and discrete states. Effects from estimation noise and wheel noise have been included. Estimation noise has been considered as discrete noise while the wheel noise has been treated as continuous noise. Robustness has been analysed by varying parameters of the satellite inertia matrix while assuming that uncertainties in the wheel angular momentum has been counteracted by the control system. The closed loop system is stable and meets all performance requirements with a large margin.

6 Variation of the Sampling Period

The analysis of the influence from variation of the sampling period will be performed with respect to

- Nominal stability
- Nominal performance
- Robust stability
- Robust performance

The analysis will be derived with a variation of the sampling period of $\pm 0.25 sec.$ with respect to stability anf for a sampling variation of of $\pm 0.1 sec.$ with respect to performance.

6.1 Nominal Stability

The analysis of the system gives that the nominal system is closed loop stable for a variation of the sampling perion of $\pm 0.25 sec.$ Again the analysis has been derived by a calculation of the closed loop poles for the sampled-data system with a variation of the sampling period. The sampled-data $\mathcal{H}_\infty$ norm has been used.

6.2 Nominal Performance

The nominal performance analysis for a variation of the sampling period of $\pm 0.1 sec.$

6.3 Robust Stability

The robust stability analysis has again been derived for a variation of the sampled-data period of $\pm 0.25 sec.$ The sampled-data system is still stable for a variation of the sampling perion with $\pm 0.25 sec.$ together with a variation of the system parameters as described above.

6.4 Robust Performance

The closed loop performance of the satellite is very sensitive to variations in the sampling period. The analysis of the robust performance is derived for a sampling period at 0.9*sec.* and 1.1*sec.* respectively.

7 Main telescope pointing

The ultimate objective of the finepointing controller is to maintain the image of an observed star within the desired area of the MONS telescope CCD.

ACS available measurements are are processed by the attitude determination subsystem, based on star tracker assembly STA with two heads for observation, STA1 and STA2. Rate gyro units are available as angular rate feedback, but are not essential for the steady state pointing accuracy.

The reference coordinates for the star under observation are sent by ground command.

The generation of reference given to the FPC, or the ADD subsystem must provide proper handling of

- initial misalignment between STA1, STA2, MONS,
- thermal or otherwise generated offset between boresight of STA1, STA2 and MONS,
- bias of the ADD estimate

This section deals with the reference system architecture used for the FPC controller.

The present design does not include MONS information in the attitude determination. The FPC should therefore acquire this through the reference generation.

Different ideas have been proposed on how to generate the reference:

1. Update the reference whenever the MONS reports the image outside a define region (deadband)
2. Investigate slowly, say once per minute whether a reference update is required
3. Use the MONS information whenever available, i.e. every 4 seconds when an image is processed.

Each of these possibilities were analysed and simulation results illustrate the principal behaviour.

Figure 21 shows the architecture of the reference subsystem. The input shown to the FPC + satellite block is the reference. The reference is generated by incrementing a counter (integrator) by an adsustable increment proportional

Parameter	**Fig. 22**	**Fig. 23**	**Fig. 23**	**Fig. 23**	Unit
deadzone	+/-50	+/-50	+/-50	0	arcsec
ref. countergain	-0.005	-0.001	0.01	0.01	s^{-1}
ref. prop. gain	5	5	5	5	s^{-1}
update interval	60	60	0	0	s

Table 1: Parameter values in reference system simulation

to the offset measured by MONS. The MONS subsystem must provide a direction indication of where to find the star when outside the MONS main telescope field of view. A deadzone and an update interval are simulated - the parameters in these blocks are adjustble in the simulation.

To show the properties of the different elements in the architecture, an offset of MONS is simulated. When FPC is activated at time 50 sec, the offset has a magnitude of 600 arcsec. The window of MONS is 360 arcsec.

Three cases are shown in Figure 22 and 23. The reference gain is the same in all plots except the first, where a large limit cycle is generated.

It is seen that the combination of integration in the loop, a limiter and a deadzone give the possibility of instability or a constant limit cycle of several times the MONS window with nonzero values of deadzone and reference update timer. The reference counter gain determines which of these conditions the loop will have.

A smooth and stable response is achievable when the deadzone is avoided and the reference update timer is set to zero. Filtering of any MONS noise is done in the integration provided by the counter. Appropriate gain is to be abblied.

The algorithm is as follows:

$$\begin{aligned} q_{ref} &= q_{ref_{cmd}} \otimes \tilde{q}_{mons} \\ \tilde{q}_{mons} &= K_{pref}\breve{q}_{mons} + \int_0^t \breve{q}_{mons} dt \end{aligned} \tag{79}$$

The integral part should be protected against wind-up.

$\breve{q}_{mons}$ is raw output from MONS, the error quaternion with saturation. The combined MONS and field monitor observation/soft ware provide this information.

The conclusion is that the reference generator with only a reference update gain and a counter works satisfactory. Neither deadzone nor update dely should be used in the reference generator. It is also concluded that MONS or the reference generator must output a signed error (error signal with saturation) when the desired star is outside the MONS field of view.

8 Conclusion

The control system was analysed in a discrete time implementation, the performance of the system was analysed and the sensitivity to disturbances and

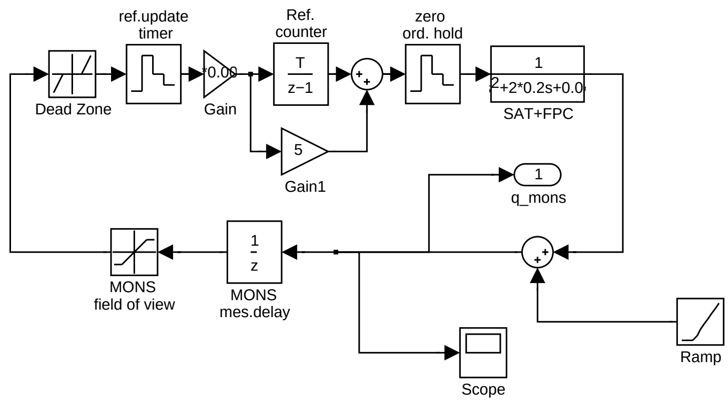


Figure 21: The architecture for the reference subsystem

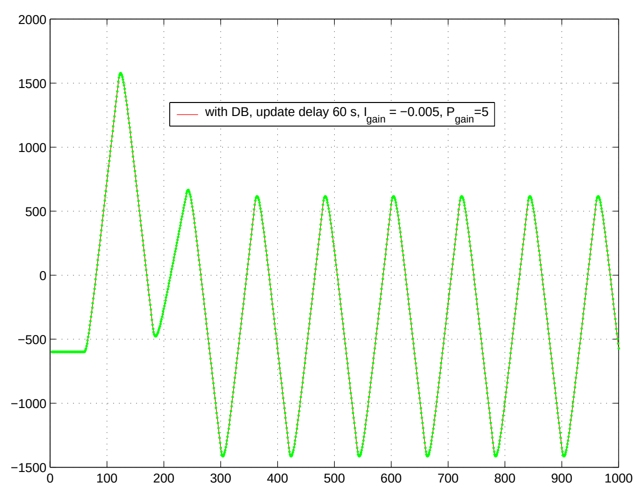


Figure 22:

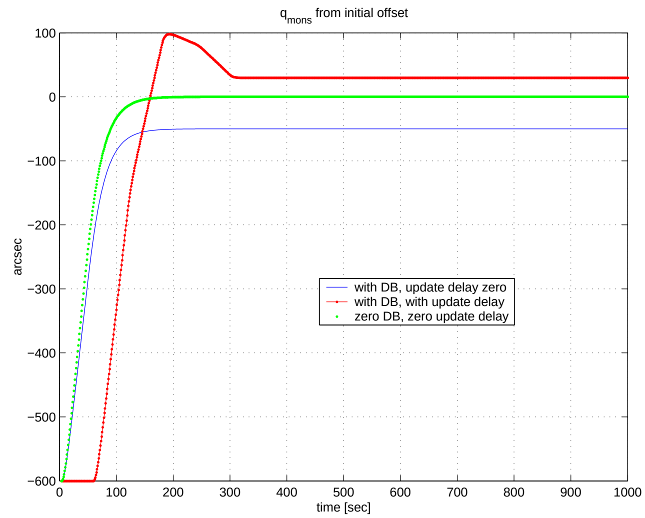


Figure 23:

measurement related noise were scrutinized, based on quoted performance of reaction wheels and rate gyros. Effects of sampling time fluctuation, satellite inertia uncertainty and alignments contribute to the build requirements of the satellite. The overall finding is that the fine-pointing requirements of the MONS experiment can be met with adequate margin when reasonable design requirements are met. The major uncertainty that remains is the magnitude of wheel torque disturbance.

An analysis of reference generator for the system has also been derived.

A Random variables and stochastic processes

In the analysis of pointing accuracy, we need to model the noise sources. The noise form the star imager is a discrete-time process, but the noise from other sources, including rate gyros and momentum wheel torque disturbances are continuous by nature. This appendix deals with the characterisation of stochastic process in continuous and discrete time.

A.1 About stochastic signals

Stochastic signals have a random variation and are described by two main features:

- The amplitude distribution
- The time and frequency domain properties.

By random we mean that there is no way to predict an exact value at a future instant of time.

A.2 Amplitude distribution.

A random process can be characterized through the amplitude of measurements taken as a time sequence. The properties can be fully determined by calculation of the moments of the process:

$$P_n = \int_{-\infty}^{\infty} x^n p(x) dx \tag{80}$$

where p(x) is the probability density function of the signal, and x the amplitude. The first moment is the mean value,

$$\mu = \int_{-\infty}^{\infty} x p(x) dx \tag{81}$$

The mean value is thus the weighted linear sum of x(t) over all values of x. Similarly, the second moment, which is also referred to as the variance is

$$\sigma^2 = \int_{-\infty}^{\infty} x^2 p(x) dx \tag{82}$$

A Gaussian process is one with a probability density function with mean value μ and variance σ^2 , given by

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

A.2.1 Continuous-time processes

An important property of a stationary random $v(t)$ process is its autocorrelation function,

$$R_{vv}(\tau) = E\{v(t)v(t+\tau)\} \tag{83}$$

A periodogram is the Fourier transform of the autocorrelation function of the process taken over an interval $[-T, T]$. The ensamble mean of periodograms is the power spectrum of the process,

$$S_{vv}(j\omega) = \lim_{T\to\infty} \int_{-T}^{T} \left(1 - \frac{|\tau|}{T}\right) R_{vv}(\tau) e^{-j\omega\tau} d\tau = \int_{-\infty}^{\infty} R_{vv}(\tau) e^{-j\omega\tau} d\tau \tag{84}$$

Likewise, the autocorrelation is the inverse Fourier transform of the mean spectrum

$$R_{vv}(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{vv}(j\omega) e^{j\omega\tau} d\omega$$

The variance of the stationary process is the area of the power spectral density function, hence total power of the signal,

$$\sigma_{vv}^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty} S_{vv}(j\omega)d\omega \equiv R_{vv}(0) \tag{85}$$

Example 1 *Consider a process with an exponential autocorrelation function*

$$R_{vv}(\tau) = a^2 e^{-\beta|\tau|} \tag{86}$$

where the constant β *is positive. The power spectrum of this process is*

$$S_{vv}(j\omega) = \int_{-\infty}^{\infty} a^2 e^{-\beta|\tau|} e^{-j\omega\tau} d\tau$$

$$= a^2 \int_0^{\infty} e^{-(\beta+j\omega)\tau} d\tau + a^2 \int_{-\infty}^{0} e^{-(-\beta+j\omega)\tau} d\tau$$

$$= \frac{a^2}{\beta + j\omega} - \frac{a^2}{-\beta + j\omega} = \frac{2a^2\beta}{\beta^2 + \omega^2}$$

The variance of the signal is

$$\sigma_{vv}^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty} \frac{2a^2\beta}{\beta^2 + \omega^2} d\omega = a^2 \tag{87}$$

Remark 2 *A stationary Gaussian process with exponential autocorrelation function is called a Gauss-Marcov process in the literature.*

Remark 3 *It is emphasised that all calculations in this appendix are showh for the two-sided power spectrum (extending over negarive and positive frequencies).*

A.2.2 Response of a linear system

Let a linear system have the transfer function (Laplace transform) $H(s)$,

$$y(s) = H(s)x(s) \tag{88}$$

the, the power spectrum S_{yy} of the output y, is related to that of the input S_{xx} by

$$S_{yy}(j\omega) = H(j\omega)H(-j\omega)S_{xx}(j\omega) = |H(j\omega)|^2 S_{xx}(j\omega) \tag{89}$$

Note that the amplitude transfer function is squared when the relation between input and output spectra is calculated.

A.2.3 The white noise abstraction

White noise in continuous time is an abstraction where the autocorrelation function approach the dirac delta function

$$R_{vv}(\tau) = \lim_{\beta\to\infty} Q_i e^{-\beta|\tau|} = Q_i \delta(\tau) \tag{90}$$

White noise has a constant spectral density function (flat spectrum),

$$S_{vv}(\omega) = Q_i \tag{91}$$

If the white noise signal has constant spectral density for all frequencies, we have defined a signal with infinite variance.

To avoid mathematical difficulties, a stochastic process that is formally the integral of a white noise process is introduced

$$w(t) = \int_0^t v(\tau)d\tau \tag{92}$$

$w(t)$ is called a Wiener process, and the infinitesimal increment $dw(t) = w(t + dt) - w(t)$.

The mean of $w(t)$ is zero because

$$E\{w(t)\} = E\{\int_0^t v(u)du\} = \int_0^t E\{v(u)\}du = 0 \tag{93}$$

The variance of $w(t)$ is

$$\begin{aligned} E\{w(t)w^T(t)\} &= E\{\int_0^t v(t_1)dt_1 \int_0^t v(t_2)dt_2\} \\ &= \int_0^t \int_0^t E\{v(t_1)v(t_2)\}dt_1dt_2 \\ &= \int_0^t \int_0^t Q_i\delta(t_1 - t_2)dt_1dt_2 = Q_i t \end{aligned} \tag{94}$$

A.2.4 Bandwidth limited white noise

When evaluating given characteristics, e.g. for the rate sensor, it is useful to introduce bandlimited white noise is a random process whose spectral amplitude is constant over a finite range of frequencies, and zero outside this range. A noise source that has a finite spectral density down to zero frequency, and a bandwidth of ω_B [rad/s] is defined by

$$S_{vv}(\omega) = \begin{cases} q_i & |\omega| \leq \omega_B \\ 0 & |\omega| > \omega_B \end{cases} \tag{95}$$

The corresponding autocorrelation function is

$$R(\tau) = 4\frac{\omega_B}{2\pi}q_i\frac{\sin(\omega_B\tau)}{\omega_B\tau}$$

If a signal is characterised by having a power (variance) of σ^2 and the bandwidth ω_B, [rad/s] then

$$q_i = \frac{1}{2}\frac{2\pi}{\omega_B}\sigma^2 \tag{96}$$

If the bandwidth is $2\pi f_B$ where f_B is in [Hz], the intensity is

$$q_i = \frac{1}{2}\frac{1}{f_B}\sigma^2 \tag{97}$$

The unit of the spectral density q_i is $[amplitude^2 * s]$.

A.3 Discrete time

Simulation and discrete-time computations require a representation of a stochastic sighal sampled at time intervals T, the sampling time.

A.3.1 Sampling a linear stochastic differential equation

Consider the differential equation

$$\dot{x}(t) = \mathbf{A}x(t) + \mathbf{B}v(t) \tag{98}$$

where $v(t)$ is a zero mean white noise process with intensity Q_i. The autocorrelation of $v(t)$ is $R_{vv}(t_1 - t_2) = E\{v(t_1)v^T(t_2)\} = Q_i\delta(t_1 - t_2)$, and the variance is infinite as discussed above.

With the introduction of the Wiener process above, and the infinitesimal increment $dw(t) = w(t+dt) - w(t)$, the differential equation (98) can be written

$$dx = \mathbf{A}xdt + \mathbf{B}dw \tag{99}$$

Integration of (99) over one sampling period, T, yields

$$x((k+1)T) = e^{\mathbf{A}T}x(kT) + \int_{kT}^{(k+1)T} e^{A((k+1)T-s)}\mathbf{B}dw(s) \tag{100}$$

Define the random variable

$$e(kT) = \int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-s)}dw(s).$$

It has zero mean because $w(s)$ has zero mean. The random variables $e(kT)$ and $e(iT)$ are uncorrelated for $k \neq i$, because the increments of w over disjoint intervals are uncorrelated. The variance of $e(kT)$ is given by

$$\begin{aligned}
\mathbf{Q}_{ee} &= E(\mathbf{e}(kT)\mathbf{e}(kT)^T) \\
&= E(\int_{kT}^{(k+1)T}\int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-s)}\mathbf{B}d\mathbf{w}(s)d\mathbf{w}(t)^T\mathbf{B}^T e^{\mathbf{A}^T((k+1)T-t)} \\
&= \int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-s)}\mathbf{B}\mathbf{Q}_i\mathbf{B}^T e^{\mathbf{A}^T((k+1)T-s)}ds \\
&= \int_0^T e^{\mathbf{A}\tau}\mathbf{B}\mathbf{Q}_i\mathbf{B}^T e^{\mathbf{A}^T\tau}d\tau
\end{aligned} \tag{101}$$

Thus the sampled data model associated to (98) can be written

$$x((k+1)T) = e^{AT}x(kT) + e(kT)$$

where $e(kT)$ is a sample of a discrete time white noise sequence with variance given by Eq.101.

Calculation of the variance $\mathbf{Q}_{ee}$ when sampling the continuous signal with intensity $\mathbf{Q}_i$ follows from integrating Eq. 101. With the usual approximation of the matrix exponential

$$e^{\mathbf{A}T} = \mathbf{I} - \mathbf{A}T + \frac{1}{2!}\mathbf{A}^2T^2 - ... \tag{102}$$

A rapid approximation is to include only one term in the series expansion. This gives the first order approximation

$$\mathbf{Q}_{ee} \simeq \mathbf{B}\mathbf{Q}_i\mathbf{B}^T T \tag{103}$$

for the variance of the variance of the discrete time noise source. This result is used in simulations.

References

[1] R.A. Masterton, D.W. Miller, and R.L. Grogan. Development of empirical and analytical reaction wheel disturbance models. In *AIAA/ASME/ASCE/ASC Structures, Structural Dynamics and Materials Conference*, St. Louis, MO, USA, 1999. paper no. AIAA-99-1204.