mouse_droid

COLLABORATORS

	TITLE : mouse_droid	

ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		August 29, 2021	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1 Mouse Droid **1**

1.1 Requirements and Goals . . . 2
1.2 Constraints . . . 3
1.3 Scope and Context . . . 4
1.4 Solution Strategy . . . 6
1.5 Building Block View . . . 7
1.5.1 Mechanics . . . 8
1.5.2 Electric Parts and Electronics . . . 10
1.5.2.1 Base Logic Board . . . 13
1.5.2.2 Main Board Logic . . . 14
1.5.3 Software . . . 15
1.5.3.1 Requirements and Goals . . . 17
1.5.3.2 Constraints . . . 19
1.5.3.3 Scope and Context . . . 20
1.5.3.4 Solution Strategy . . . 21
1.5.3.5 Building Block View . . . 22
1.5.3.6 Runtime View . . . 23
1.5.3.6.1 Environment Model Composer Sequence . . . 26
1.5.3.7 Deployment View . . . 27
1.5.3.8 Crosscutting Concepts . . . 28
1.5.3.9 Architecture Decisions . . . 29
1.5.3.10 Quality Requirements . . . 30
1.5.3.10.1 Quality Tree . . . 31
1.5.3.10.2 Quality Scenarios . . . 32
1.5.3.11 Risks and Technical Debts . . . 33
1.5.3.12 Glossary . . . 34
1.6 Runtime View . . . 35
1.6.1 Power Modes . . . 36
1.6.1.1 Power Mode Timings . . . 37
1.6.2 Health States . . . 38

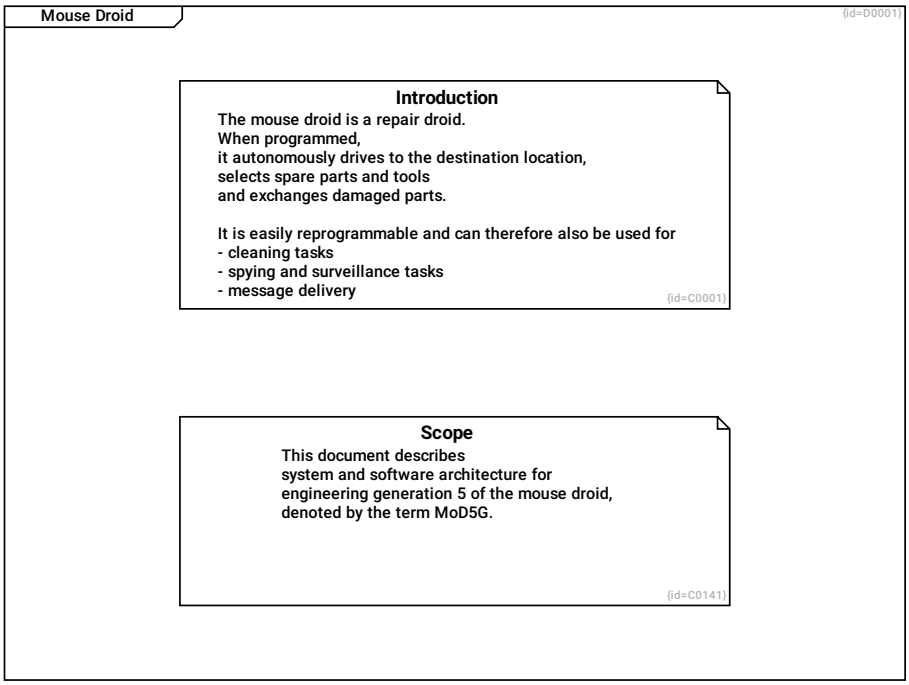
1.7 Deployment View 40
1.8 Crosscutting Concepts 41
1.9 Architecture Decisions 41
1.10 Quality Requirements 43
1.10.1 Quality Tree 43
1.10.1.1 Maintainability 44
1.10.2 Quality Scenarios 46
1.11 Risks and Technical Debts 47
1.12 Glossary 48

Chapter 1

Mouse Droid

This document shows a system architecture and software architecture for a mouse droid. This is a small repair droid similar to the MSE-6 used on death star 1.

The document follows the arc42.org template proposed by Gernot Starke and Peter Hruschka,



Introduction C0001

The mouse droid is a repair droid. When programmed, it autonomously drives to the destination location, selects spare parts and tools and exchanges damaged parts.

It is easily reprogrammable and can therefore also be used for

- cleaning tasks

- spying and surveillance tasks

- message delivery

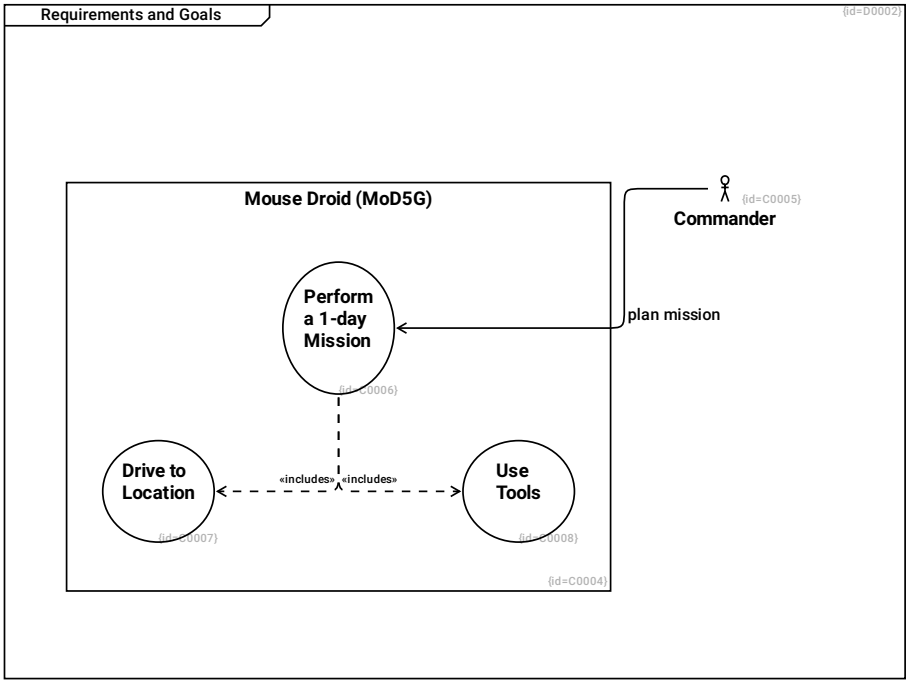
Scope C0141

This document describes system and software architecture for engineering generation 5 of the mouse droid, denoted by the term MoD5G.

1.1 Requirements and Goals

This section gives a short overview on the project goals (Problem Space, System Level L1)

Primary purpose of the MoD5G is to autonomously repair mechanical things.



Commander C0005

The commander instructs the MoD5G on the mission to perform.

plan mission --> Perform a 1-day Mission R0002

provide mission goals and strategy

Perform a 1-day Mission C0006

The mouse droid is able to perform a mission that takes several hours. The energy resources of the MoD5G last for up to one day.

- The commander programs a mission
- The mouse droid drives to the first location
- The mouse droid uses tools to remove a defective part
- The mouse droid installes a spare part
- Above steps are repeated for other goals
- The mouse droid returns to its base location (see Section 1.12: Glossary)

--> Drive to Location R0005

--> Use Tools R0006

Drive to Location C0007

The mouse droid can explore its environment and calculate a route from the current ot the target location.

- The mouse droid explores its environment
- The mouse droid enriches internally memorized map
- The mouse droid calculates a route
- The mouse droid drives along the calculated route
- The mouse droid re-caclulates the route in case of new environment data
- The mouse droid reaches the target location

Use Tools C0008

The mouse droid has a couple of tools inside its chassis.

- The mouse droid uses a screw diver to untighten damaged parts
- The mouse droid uses a gripper to move the damaged part out of the way
- The mouse droid uses a gripper to put a spare part from its internal cargo bin to the target place
- The mouse droid uses a screw diver to tighten replaced parts
- The mouse droid uses a gripper to move the damaged part into its internal cargo bin.

(see Section 1.5.1: Mechanics)

Mouse Droid (MoD5G) C0004

The Mouse Droid (MoD5G) is a repair droid that can be instructed to perform a mission and which autonomously selects tactics to achieve the mission goals

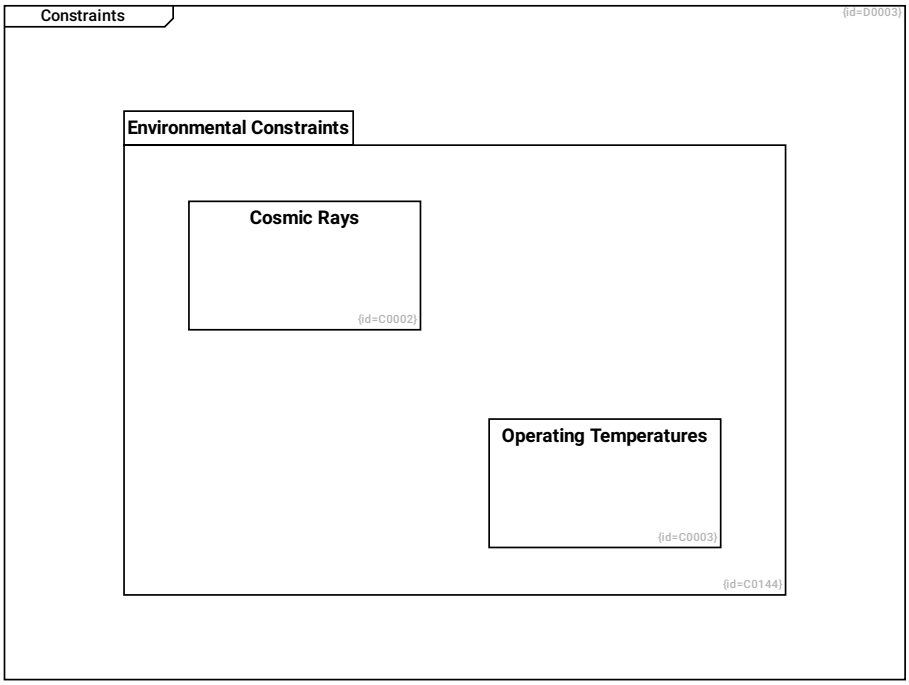
--> Perform a 1-day Mission R0001

--> Drive to Location R0003

--> Use Tools R0004

1.2 Constraints

This section explains the major obstacles, that need to be considered when designing a solution to reach the project goals. (Problem Space, System Level L1)



Cosmic Rays C0002

The droid shall ensure data and program integrity and continue operation after cosmic rays may have interfered with normal operation.

Corrupted data must not be stored permanently.

Environmental Constraints C0144

--> Cosmic Rays R0221

--> Operating Temperatures R0222

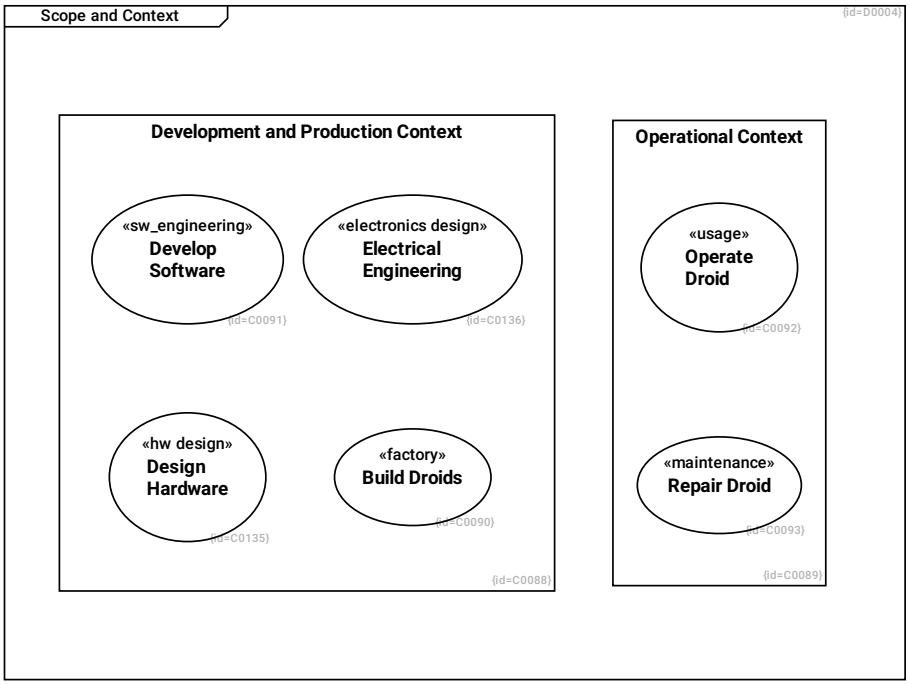
Group of constratins imposed by the operation environment

Operating Temperatures C0003

The droid shall be fully functional in the range 240K..360K, it shall survive temperatures from 200K to 400K.

1.3 Scope and Context

This section shows the organizational contexts of development and operational environments. (Problem Space, System Level L1)



Operational Context C0089

This boundary encompasses the topics that are in scope during operation and maintenance.

--> Repair Droid R0127

--> Operate Droid R0126

Repair Droid C0093

Operate Droid C0092

Electrical Engineering C0136

Design Hardware C0135

Development and Production Context C0088

This boundary encompasses the topics that are in scope during development and production.

--> Electrical Engineering R0188

--> Design Hardware R0187

--> Develop Software R0125

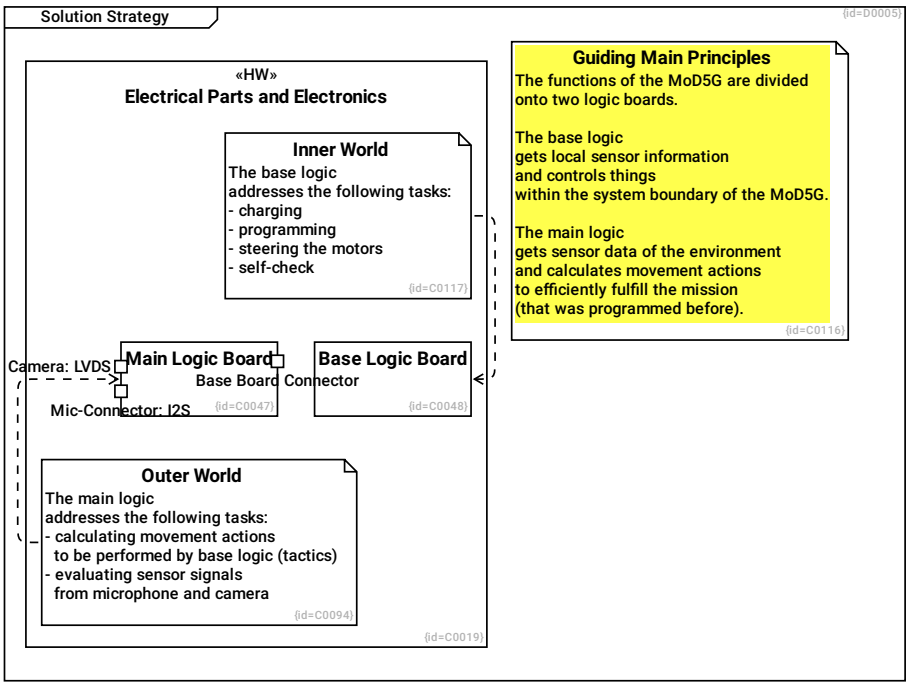
--> Build Droids R0124

Develop Software C0091

Build Droids C0090

1.4 Solution Strategy

This section shows the most fundamental principles of the system design. (Solution Space, System Level L1)



Outer World C0094

The main logic addresses the following tasks:

- calculating movement actions to be performed by base logic (tactics)
- evaluating sensor signals from microphone and camera

--> Main Logic Board R0184

Electrical Parts and Electronics C0019

The main components of the electric parts are

- a set of cables and connectors
- a set of sensors and actuators,
- the energy cell and
- two printed circuit boards (PCB) containing the electronic parts.

--> Base Logic Board R0048

--> Main Logic Board R0047

--> Inner World R0185

--> Outer World R0186

Base Logic Board C0048

The base logic board consists of several electronic parts shown in Section 1.5.2.1: Base Logic Board.

Main Logic Board C0047

The main logic board consists of several electronic parts shown in Section 1.5.2.2: Main Board Logic.

Base Board Connector F0008

Camera: LVDS F0001

Mic-Connector: I2S F0002

Guiding Main Principles C0116

The functions of the MoD5G are divided onto two logic boards.

The base logic gets local sensor information and controls things within the system boundary of the MoD5G.

The main logic gets sensor data of the environment and calculates movement actions to efficiently fulfill the mission (that was programmed before).

Inner World C0117

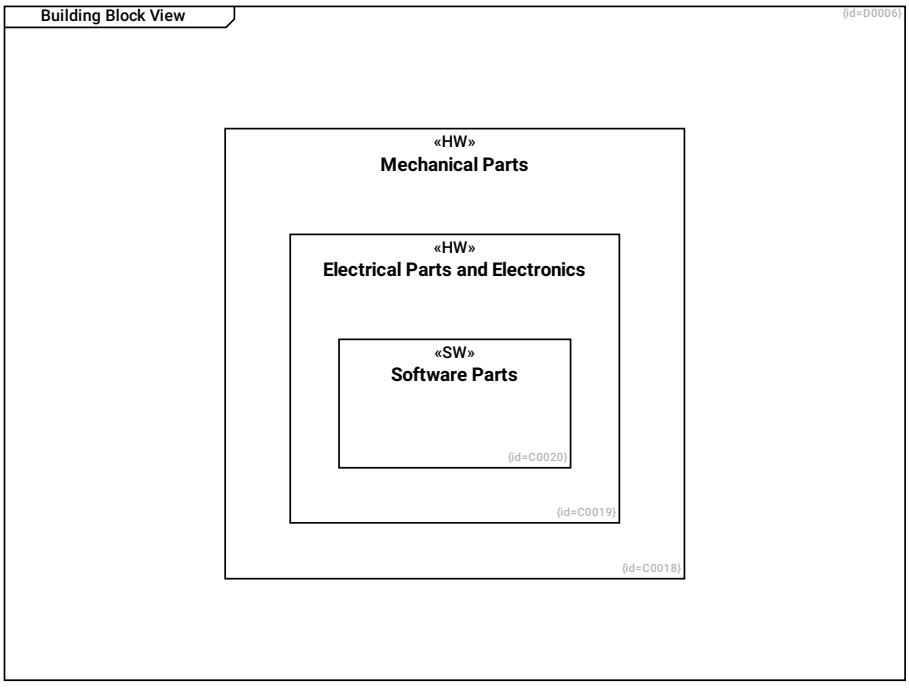
The base logic addresses the following tasks:

- charging
- programming
- steering the motors
- self-check

--> Base Logic Board R0183

1.5 Building Block View

This section shows the parts of the MoD5G system (Solution Space, System Level L1)



Electrical Parts and Electronics C0019

The main components of the electric parts are
- a set of cables and connectors
- a set of sensors and actuators,
- the energy cell and
- two printed circuit boards (PCB) containing the electronic parts.

--> Software Parts R0014

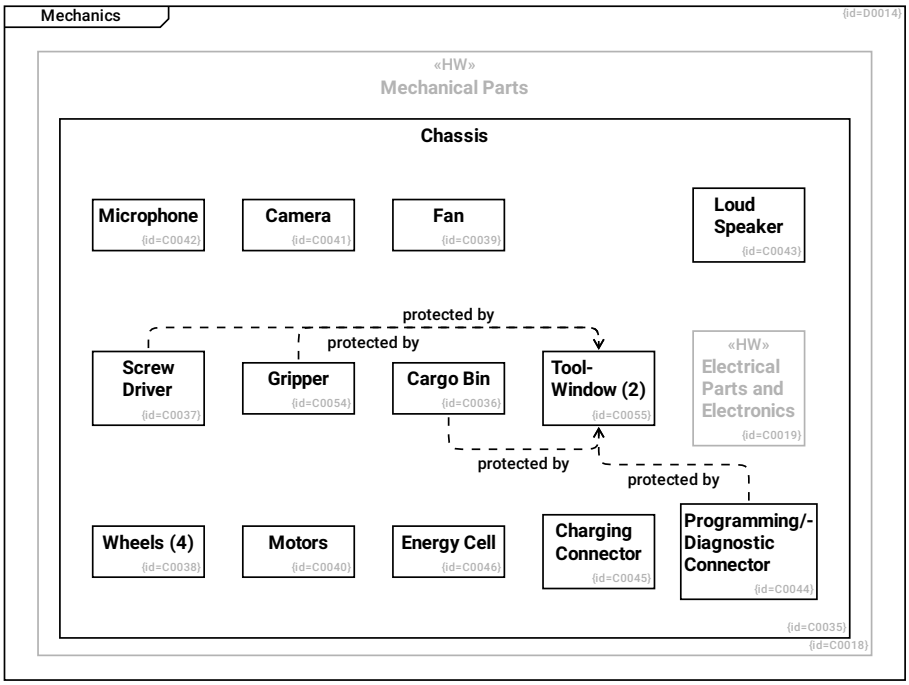
Software Parts C0020

Mechanical Parts C0018

--> Electrical Parts and Electronics R0013

1.5.1 Mechanics

This section shows the mechanical parts of the MoD5G system (Solution Space, System Level L2)



Electrical Parts and Electronics C0019

The main components of the electric parts are
- a set of cables and connectors
- a set of sensors and actuators,
- the energy cell and
- two printed circuit boards (PCB) containing the electronic parts.

Tool-Window (2) C0055

A tool window is a flap in the chassis that protects screw driver, griipper and cargo bin when unused.

Mechanical Parts C0018

--> Electrical Parts and Electronics R0013

--> Chassis R0045

Gripper C0054

A tool that allows to grab objects and move them.

protected by --> Tool-Window (2) R0223

Cargo Bin C0036

protected by --> Tool-Window (2) R0224

Microphone C0042

Energy Cell C0046

Programming/Diagnostic Connector C0044

protected by --> Tool-Window (2) R0226

Motors C0040

Loud Speaker C0043

Charging Connector C0045

Camera C0041

Wheels (4) C0038

Fan C0039

A fan prevents overheating in hot environment conditions.

Screw Driver C0037

protected by --> Tool-Window (2) R0225

Chassis C0035

--> Tool-Window (2) R0065

--> Gripper R0064

--> Cargo Bin R0034

--> Microphone R0040

--> Energy Cell R0044

--> Programming/Diagnostic Connector R0042

--> Motors R0038

--> Loud Speaker R0041

--> Charging Connector R0043

--> Camera R0039

--> Wheels (4) R0036

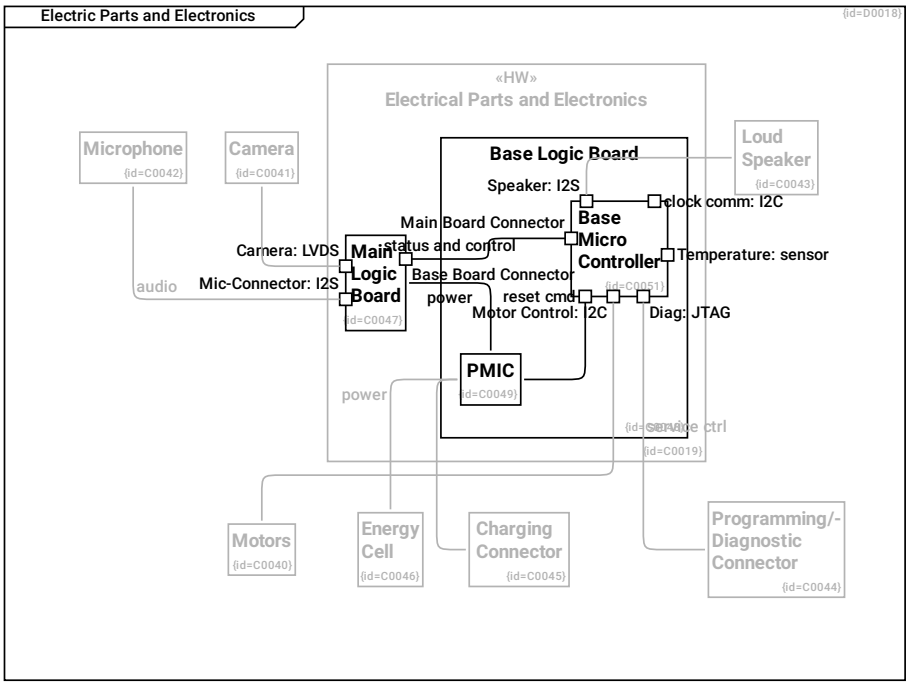
--> Fan R0037

--> Screw Driver R0035

--> Electrical Parts and Electronics R0046

1.5.2 Electric Parts and Electronics

This section shows the electric parts and electronics of the MoD5G system (Solution Space, System Level L2)



Electrical Parts and Electronics C0019

The main components of the electric parts are
- a set of cables and connectors
- a set of sensors and actuators,
- the energy cell and
- two printed circuit boards (PCB) containing the electronic parts.

--> Base Logic Board R0048

--> Main Logic Board R0047

Base Logic Board C0048

The base logic board consists of several electronic parts shown in Section 1.5.2.1: Base Logic Board.

--> Base Micro Controller R0053

--> PMIC R0049

Base Micro Controller C0051

The base micro controller consists of
- logic unit
- data storage
- persistent data+logic storage
- self supervision (by ECC and lockstep-cores)
- HW watchdog
- clock
- temperature sensor
- io ports

Temperature: sensor F0010

clock comm: I2C F0007

Speaker: I2S F0004

Main Board Connector F0009

reset cmd F0006

Motor Control: I2C F0003

Diag: JTAG F0005

--> Motors R0059

--> Loud Speaker R0060

--> PMIC R0062

PMIC C0049

Power Management Integrated Circuit

power --> Main Logic Board R0063

Main Logic Board C0047

The main logic board consists of several electronic parts shown in Section 1.5.2.2: Main Board Logic.

Base Board Connector F0008

Camera: LVDS F0001

Mic-Connector: I2S F0002

status and control --> Base Micro Controller R0068

Microphone C0042

audio --> Main Logic Board R0056

Energy Cell C0046

power --> PMIC R0050

Programming/Diagnostic Connector C0044

service ctrl --> Base Micro Controller R0061

Motors C0040

Loud Speaker C0043

Charging Connector C0045

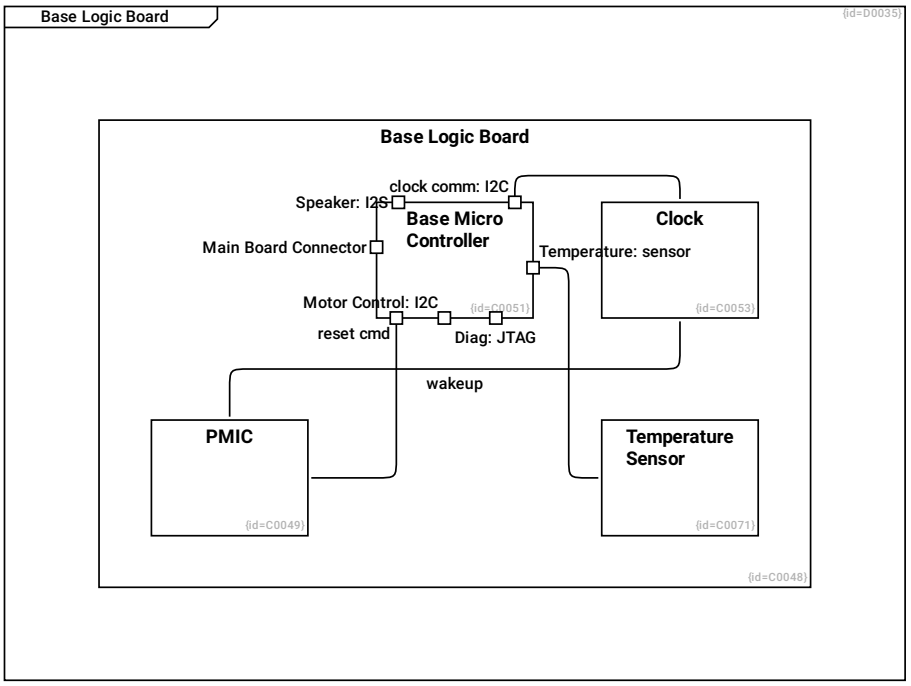
--> PMIC R0057

Camera C0041

--> Main Logic Board R0051

1.5.2.1 Base Logic Board

This section shows the base board logic of the MoD5G system (Solution Space, System Level L3)



Base Logic Board C0048

The base logic board consists of several electronic parts shown in Section 1.5.2.1: Base Logic Board.

--> Base Micro Controller R0053

--> Clock R0055

--> PMIC R0049

--> Temperature Sensor R0095

Base Micro Controller C0051

The base micro controller consists of

- logic unit
- data storage
- persistent data+logic storage
- self supervision (by ECC and lockstep-cores)
- HW watchdog
- clock
- temperature sensor
- io ports

Temperature: sensor F0010

clock comm: I2C F0007

Speaker: I2S F0004

Main Board Connector F0009

reset cmd F0006

Motor Control: I2C F0003

Diag: JTAG F0005

--> PMIC R0062

--> Clock R0066

Clock C0053

wakeup --> PMIC R0067

PMIC C0049

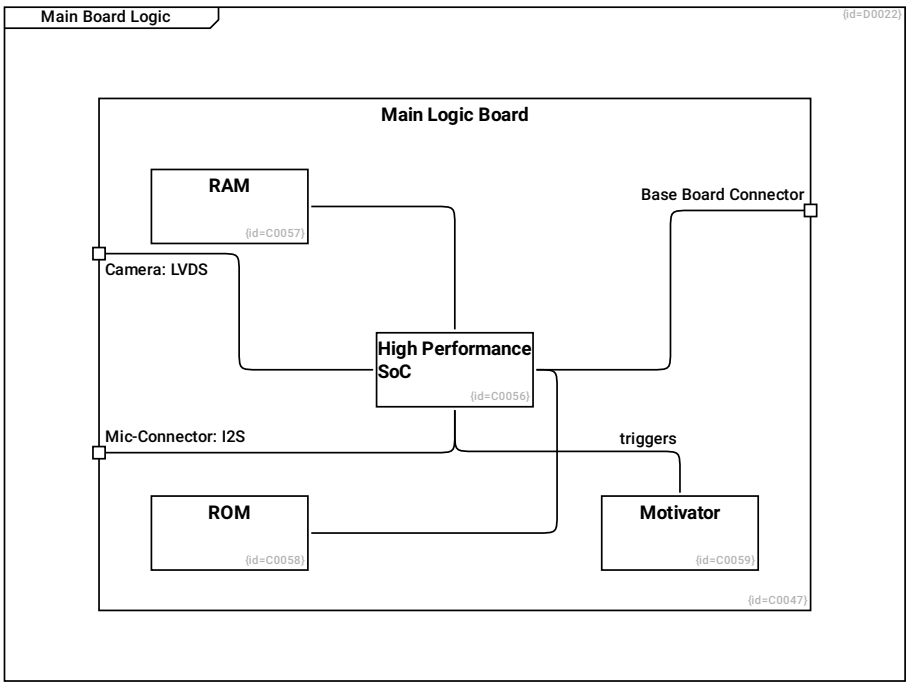
Power Management Integrated Circuit

Temperature Sensor C0071

--> Base Micro Controller R0096

1.5.2.2 Main Board Logic

This section shows the main board logic of the MoD5G system (Solution Space, System Level L3)



RAM C0057

ROM C0058

High Performance SoC C0056

--> RAM R0072

--> ROM R0073

--> Main Logic Board R0075

Main Logic Board C0047

The main logic board consists of several electronic parts shown in Section 1.5.2.2: Main Board Logic.

Base Board Connector F0008

Camera: LVDS F0001

Mic-Connector: I2S F0002

--> RAM R0070

--> ROM R0071

--> High Performance SoC R0069

--> Motivator R0077

--> High Performance SoC R0074

--> High Performance SoC R0076

Motivator C0059

A motivator is a basic component needed to keep going on.

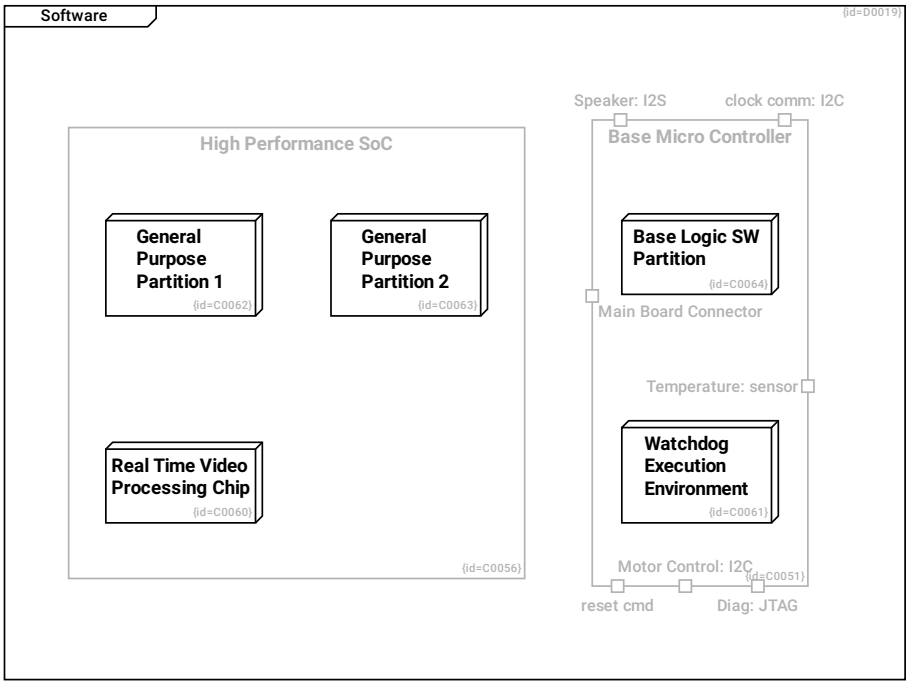
triggers --> High Performance SoC R0078

1.5.3 Software

This diagram shows the virtual machines and specialized (non-versatile) execution environments (Solution Space, System Level L2)

These are deployed onto the logic boards shown in Section 1.5.2: Electric Parts and Electronics.

In this section, this view is further detailed to software elements, their relations and interactions.



Real Time Video Processing Chip C0060

General Purpose Partition 1 C0062

General Purpose Partition 2 C0063

High Performance SoC C0056

--> Real Time Video Processing Chip R0079

--> General Purpose Partition 1 R0081

--> General Purpose Partition 2 R0082

Base Micro Controller C0051

The base micro controller consists of
- logic unit
- data storage
- persistent data+logic storage
- self supervision (by ECC and lockstep-cores)
- HW watchdog
- clock
- temperature sensor
- io ports

Temperature: sensor F0010

clock comm: I2C F0007

Speaker: I2S F0004

Main Board Connector F0009

reset cmd F0006

Motor Control: I2C F0003

Diag: JTAG F0005

--> Base Logic SW Partition R0083

--> Watchdog Execution Environment R0080

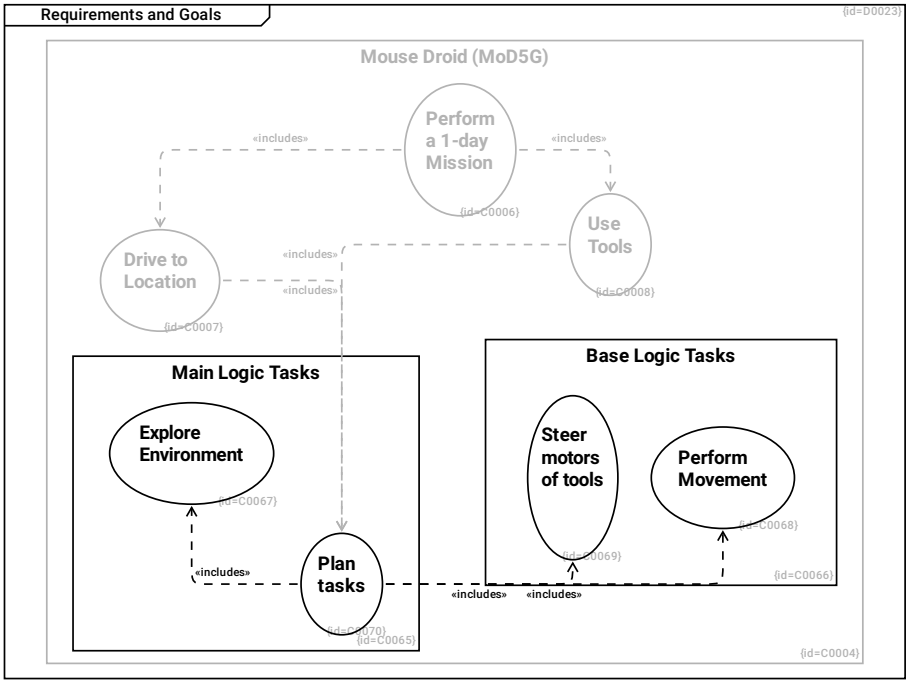
Base Logic SW Partition C0064

Watchdog Execution Environment C0061

1.5.3.1 Requirements and Goals

This section shows the goals of the software development for the MoD5G (Problem Space, Software Level L3)

In gray, the use cases on system level L1 are repeated from Section 1.1: Requirements and Goals to show the refinement to software-only use cases shown in black.



Perform a 1-day Mission C0006

The mouse droid is able to perform a mission that takes several hours. The energy resources of the MoD5G last for up to one day.

- The commander programs a mission

- The mouse droid drives to the first location

- The mouse droid uses tools to remove a defective part
- The mouse droid installes a spare part
- Above steps are repeated for other goals
- The mouse droid returns to its base location (see Section 1.12: Glossary)

--> Drive to Location R0005

--> Use Tools R0006

Drive to Location C0007

The mouse droid can explore its environment and calculate a route from the current ot the target location.
- The mouse droid explores its environment
- The mouse droid enriches internally memorized map
- The mouse droid calculates a route
- The mouse droid drives along the calculated route
- The mouse droid re-caclulates the route in case of new environment data
- The mouse droid reaches the target location

--> Plan tasks R0091

Use Tools C0008

The mouse droid has a couple of tools inside its chassis.
- The mouse droid uses a screw diver to untighten damaged parts
- The mouse droid uses a gripper to move the damaged part out of the way
- The mouse droid uses a gripper to put a spare part from its internal cargo bin to the target place
- The mouse droid uses a screw diver to tighten replaced parts
- The mouse droid uses a gripper to move the damaged part into its internal cargo bin.

(see Section 1.5.1: Mechanics)

--> Plan tasks R0090

Base Logic Tasks C0066

--> Perform Movement R0087

--> Steer motors of tools R0088

Perform Movement C0068

Steer motors of tools C0069

Main Logic Tasks C0065

--> Plan tasks R0089

--> Explore Environment R0086

Plan tasks C0070

The mouse droid creates a list of actions to fulfill the given mission. If data on the environment is missing, it plans an explortion task and re-plans the action list later.

--> Perform Movement R0092

--> Steer motors of tools R0093

--> Explore Environment R0094

Explore Environment C0067

When the mouse droid is missing relevant data on the environment, it plans a list of actions that suits the purpose of gaining the missing knowledge.

Mouse Droid (MoD5G) C0004

The Mouse Droid (MoD5G) is a repair droid that can be instructed to perform a mission and which autonomously selects tactics to achieve the mission goals

--> Perform a 1-day Mission R0001

--> Drive to Location R0003

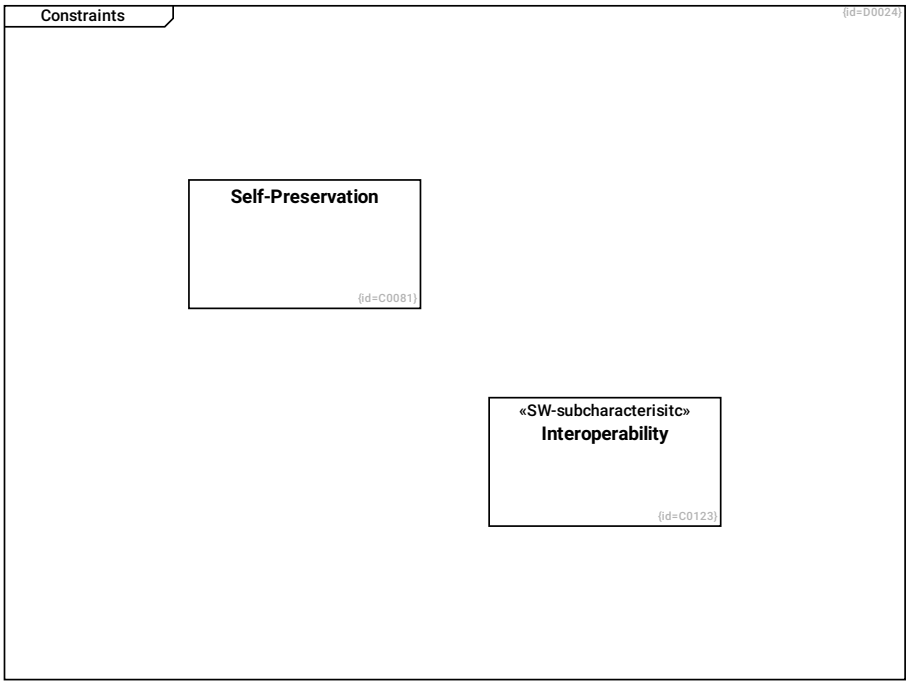
--> Use Tools R0004

--> Base Logic Tasks R0085

--> Main Logic Tasks R0084

1.5.3.2 Constraints

This section explains the major obstacles, that need to be considered when designing a solution to reach the project goals. (Problem Space, Software Level L3)



Self-Preservation C0081

In case a wookiee growls at the MoD5G, it shall flee for self-preservation

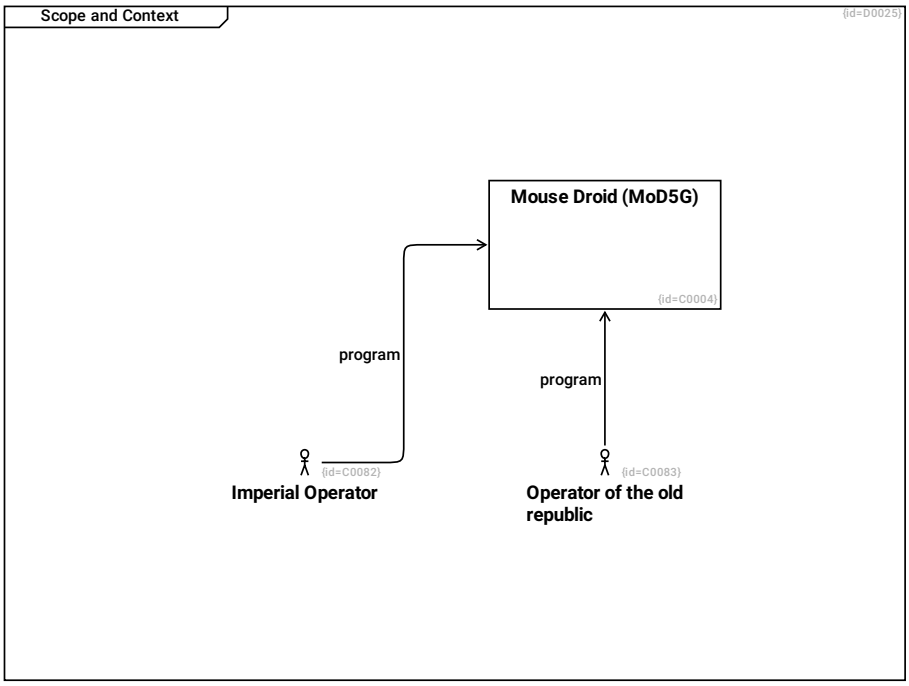
Interoperability C0123

The programming and charging interfaces of the MoD5G shall be compatible to

- old republic terminals
- imperial terminals

1.5.3.3 Scope and Context

This section shows the organizational contexts of development and operational environments. (Problem Space, Software Level L3)



Operator of the old republic C0083

program --> Mouse Droid (MoD5G) R0112

Imperial Operator C0082

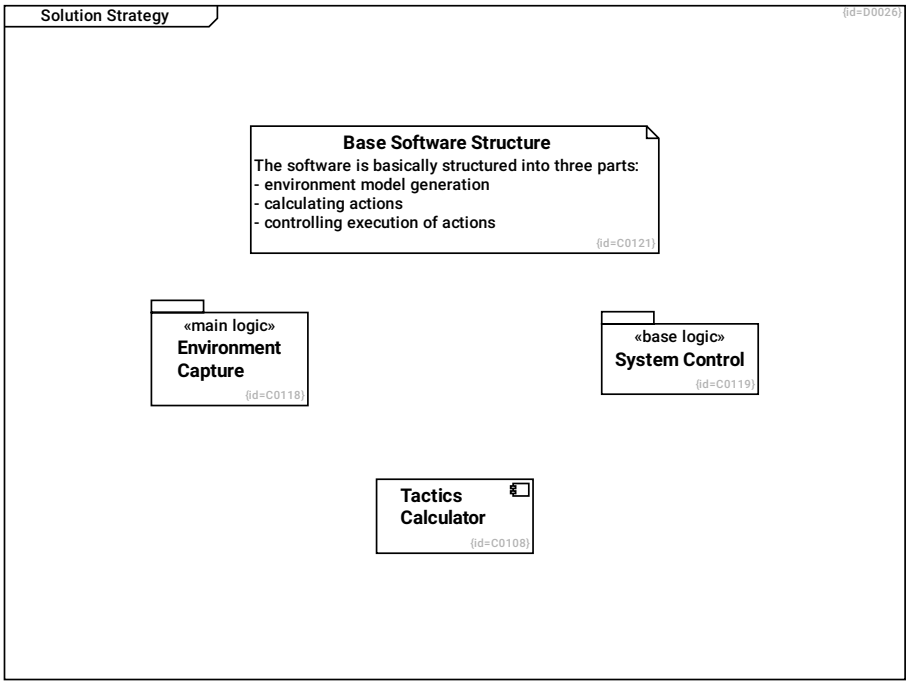
program --> Mouse Droid (MoD5G) R0113

Mouse Droid (MoD5G) C0004

The Mouse Droid (MoD5G) is a repair droid that can be instructed to perform a mission and which autonomously selects tactics to achieve the mission goals

1.5.3.4 Solution Strategy

This section shows the most fundamental principles of the software design. (Solution Space, Software Level L3)



Base Software Structure C0121

The software is basically structured into three parts:

- environment model generation
- calculating actions
- controlling execution of actions

Environment Capture C0118

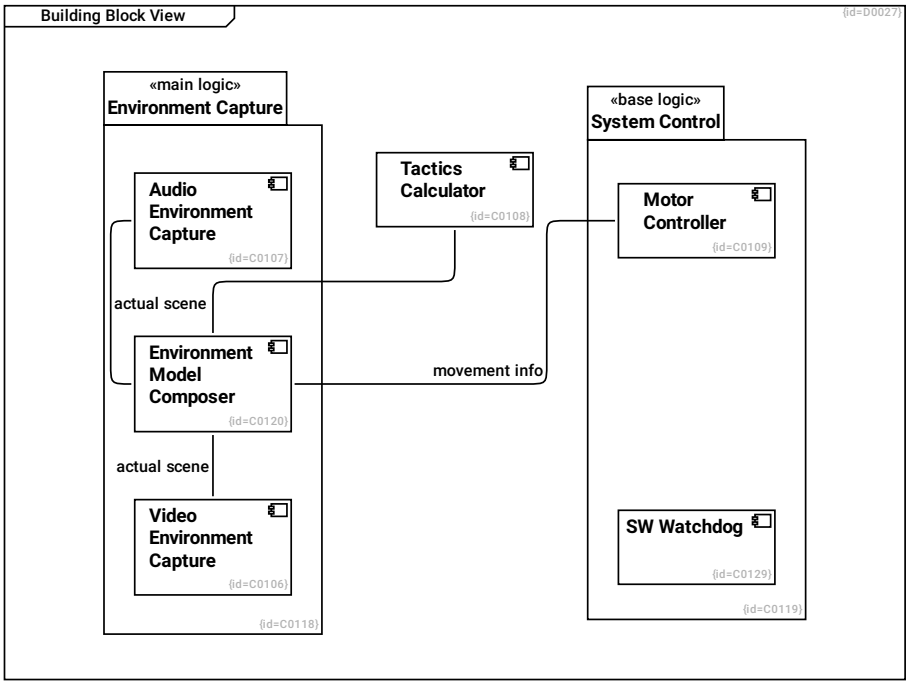
System Control C0119

Tactics Calculator C0108

Calculate tactics based on given strategy and current situation model

1.5.3.5 Building Block View

This section shows the parts of the MoD5G software (Solution Space, Software Level L3)



Audio Environment Capture C0107

actual scene --> Environment Model Composer R0154

Environment Capture C0118

--> Environment Model Composer R0152

--> Video Environment Capture R0148

--> Audio Environment Capture R0149

System Control C0119

--> SW Watchdog R0174

--> Motor Controller R0151

Video Environment Capture C0106

actual scene --> Environment Model Composer R0155

SW Watchdog C0129

The SW Watchdog shall check
- validity of data as well as
- validity of sequence of checkpoints

received from software components on the Main Logic Board.
See also Section 1.5.3.8: Crosscutting Concepts.

Environment Model Composer C0120

--> Tactics Calculator R0156

Tactics Calculator C0108

Calculate tactics based on given strategy and current situation model

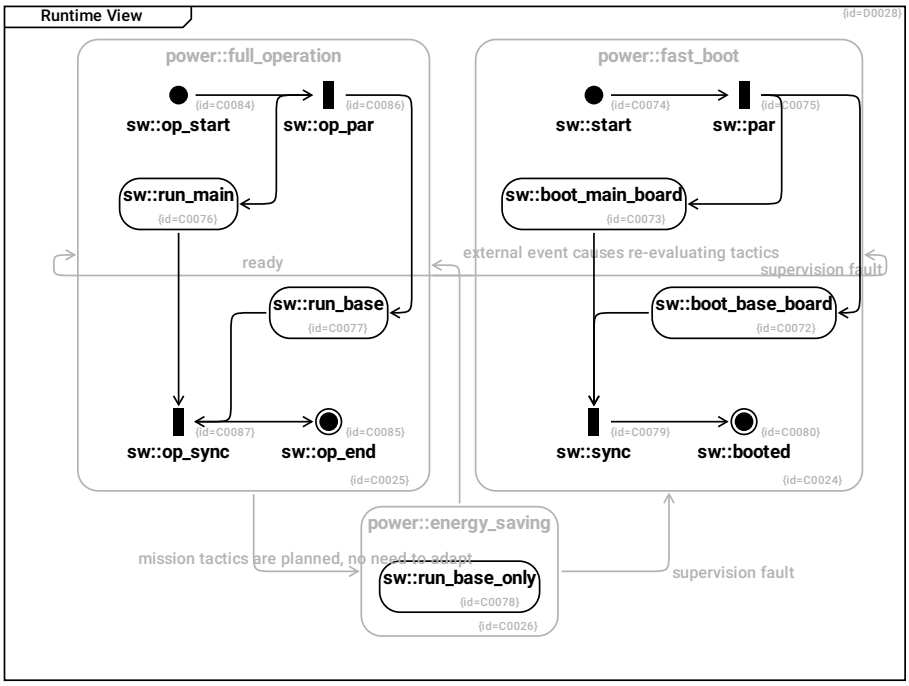
Motor Controller C0109

Move motors according to calculated tactics

movement info --> Environment Model Composer R0153

1.5.3.6 Runtime View

This section shows the dynamic behavior of the software (Solution Space, Software Level L3)
This diagram shows the software states embedded in the system states. See Section 1.6.1: Power Modes.



sw::sync C0079

--> sw::booted R0111

sw::start C0074

--> sw::par R0101

sw::par C0075

--> sw::boot_main_board R0102

--> sw::boot_base_board R0103

sw::boot_main_board C0073

--> sw::sync R0108

sw::booted C0080

sw::boot_base_board C0072

--> sw::sync R0109

sw::op_sync C0087

--> sw::op_end R0118

sw::op_end C0085

sw::op_par C0086

--> sw::run_main R0120

--> sw::run_base R0121

sw::op_start C0084

--> sw::op_par R0119

sw::run_base C0077

--> sw::op_sync R0122

sw::run_main C0076

--> sw::op_sync R0123

power::full_operation C0025

--> sw::op_sync R0117

--> sw::op_end R0115

--> sw::op_par R0116

--> sw::op_start R0114

--> sw::run_base R0105

--> sw::run_main R0104

mission tactics are planned, no need to adapt --> power::energy_saving R0017

supervision fault --> power::fast_boot R0033

power::fast_boot C0024

--> sw::sync R0107

--> sw::start R0099

--> sw::par R0100

--> sw::boot_main_board R0098

--> sw::booted R0110

--> sw::boot_base_board R0097

ready --> power::full_operation R0016

sw::run_base_only C0078

power::energy_saving C0026

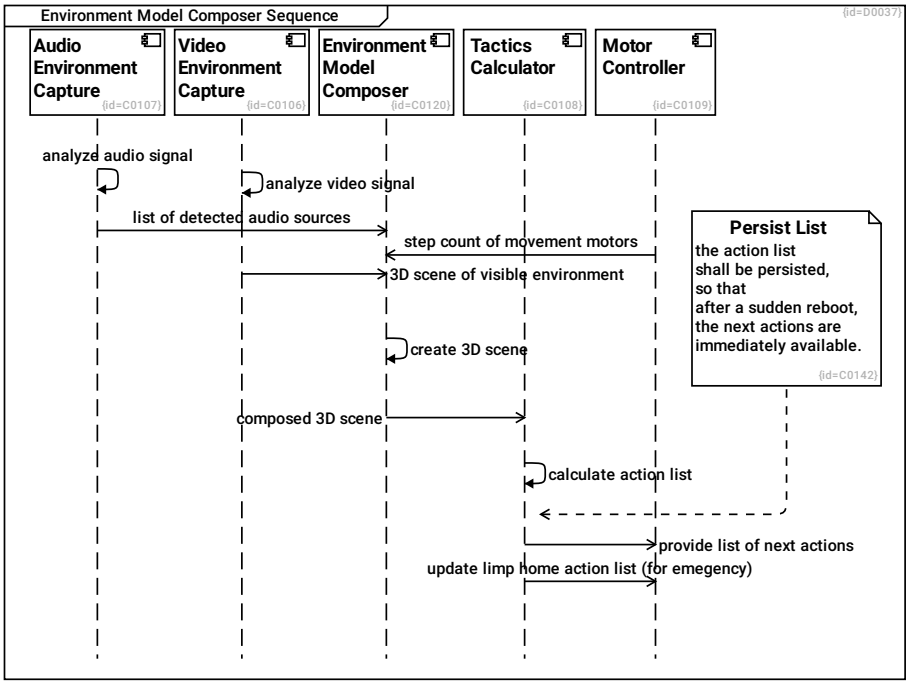
--> sw::run_base_only R0106

external event causes re-evaluating tactics --> power::full_operation R0018

supervision fault --> power::fast_boot R0032

1.5.3.6.1 Environment Model Composer Sequence

This diagram shows the typical communication sequence to compose the environment model.



Audio Environment Capture C0107

analyze audio signal --> Audio Environment Capture R0169

list of detected audio sources --> Environment Model Composer R0164

Video Environment Capture C0106

analyze video signal --> Video Environment Capture R0170

3D scene of visible environment --> Environment Model Composer R0163

Persist List C0142

the action list shall be persisted, so that after a sudden reboot, the next actions are immediately available.

--> Tactics Calculator R0220

Environment Model Composer C0120

create 3D scene --> Environment Model Composer R0171

create 3D scene based on sensors, status and history.

composed 3D scene --> Tactics Calculator R0165

Tactics Calculator C0108

Calculate tactics based on given strategy and current situation model

calculate action list --> Tactics Calculator R0172

calculate action list to follow strategy

provide list of next actions --> Motor Controller R0167

update limp home action list (for emegency) --> Motor Controller R0168

For the emergency case, update the limp home action list

Motor Controller C0109

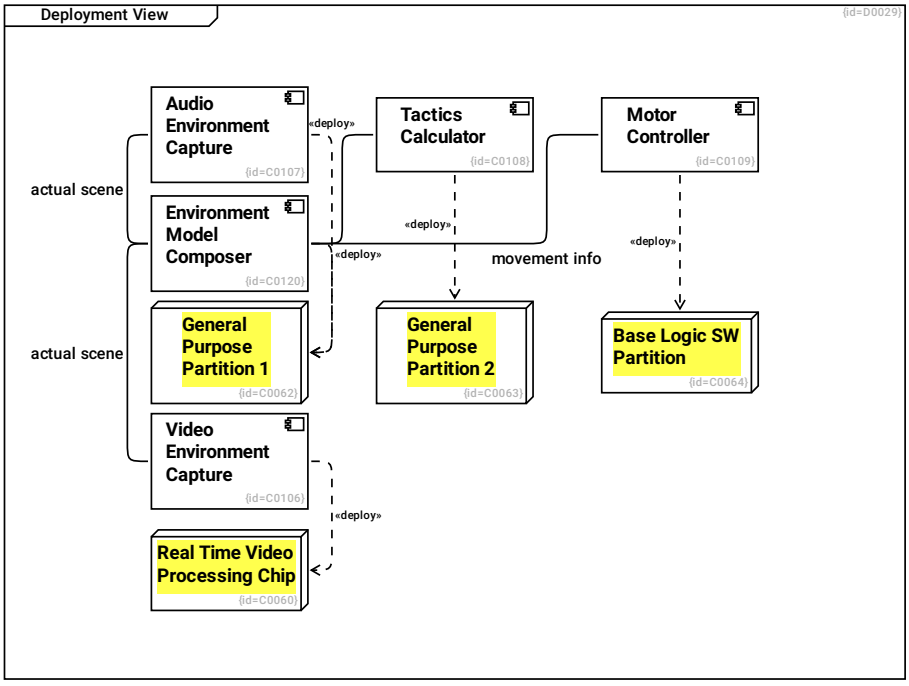
Move motors according to calculated tactics

step count of movement motors --> Environment Model Composer R0166

step count of steering and movement motors

1.5.3.7 Deployment View

This section shows the deployment of the solution into the environment. (Solution Space, Software Level L3)



Audio Environment Capture C0107

actual scene --> Environment Model Composer R0154

--> General Purpose Partition 1 R0204

Real Time Video Processing Chip C0060

General Purpose Partition 1 C0062

General Purpose Partition 2 C0063

Base Logic SW Partition C0064

Video Environment Capture C0106

actual scene --> Environment Model Composer R0155

--> Real Time Video Processing Chip R0201

Environment Model Composer C0120

--> Tactics Calculator R0156

--> General Purpose Partition 1 R0203

Tactics Calculator C0108

Calculate tactics based on given strategy and current situation model

--> General Purpose Partition 2 R0202

Motor Controller C0109

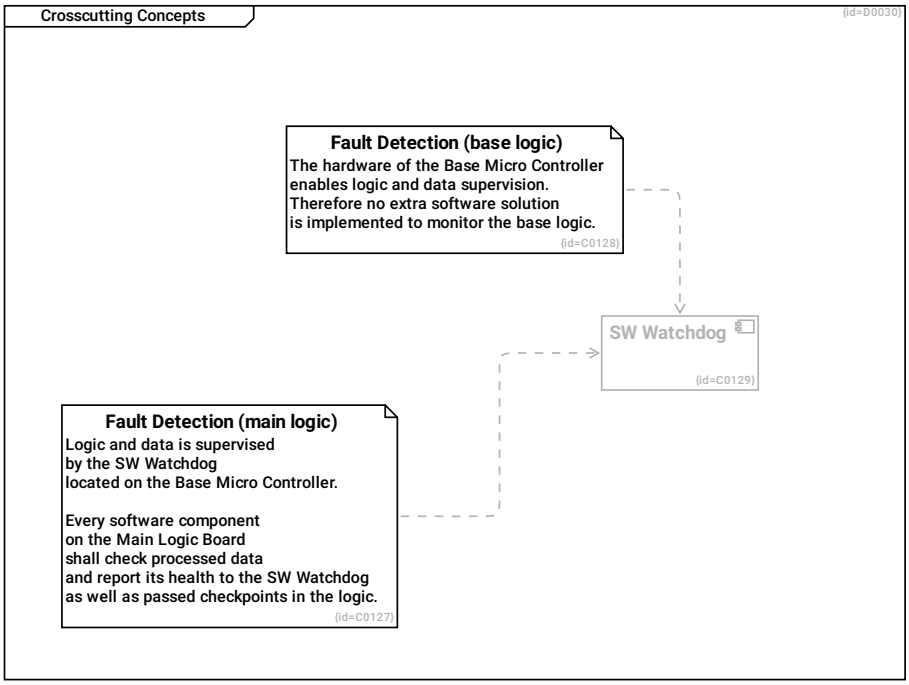
Move motors according to calculated tactics

movement info --> Environment Model Composer R0153

--> Base Logic SW Partition R0200

1.5.3.8 Crosscutting Concepts

This section shows the recurring concepts within the the designed solution. (Solution Space, Software Level L3)



Fault Detection (base logic) C0128

The hardware of the Base Micro Controller enables logic and data supervision. Therefore no extra software solution is implemented to monitor the base logic.

--> SW Watchdog R0198

Fault Detection (main logic) C0127

Logic and data is supervised by the SW Watchdog located on the Base Micro Controller.

Every software component on the Main Logic Board shall check processed data and report its health to the SW Watchdog as well as passed checkpoints in the logic.

--> SW Watchdog R0175

SW Watchdog C0129

The SW Watchdog shall check

- validity of data as well as

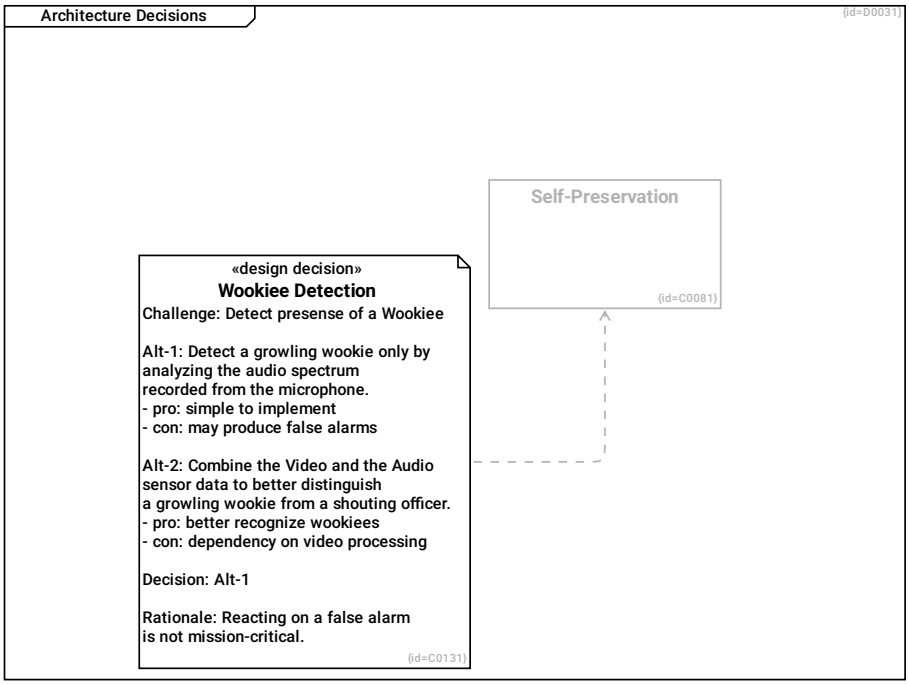
- validity of sequence of checkpoints

received from software components on the Main Logic Board.

See also Section 1.5.3.8: Crosscutting Concepts.

1.5.3.9 Architecture Decisions

This section explains the major and non-obvious design decisions. (Solution Space, Software Level L3)



Self-Preservation C0081

In case a wookiee growls at the MoD5G, it shall flee for self-preservation

Wookiee Detection C0131

Challenge: Detect presense of a Wookiee

Alt-1: Detect a growling wookie only by analyzing the audio spectrum recorded from the microphone.

- pro: simple to implement

- con: may produce false alarms

Alt-2: Combine the Video and the Audio sensor data to better distinguish a growling wookie from a shouting officer.

- pro: better recognize wookiees

- con: dependency on video processing

Decision: Alt-1

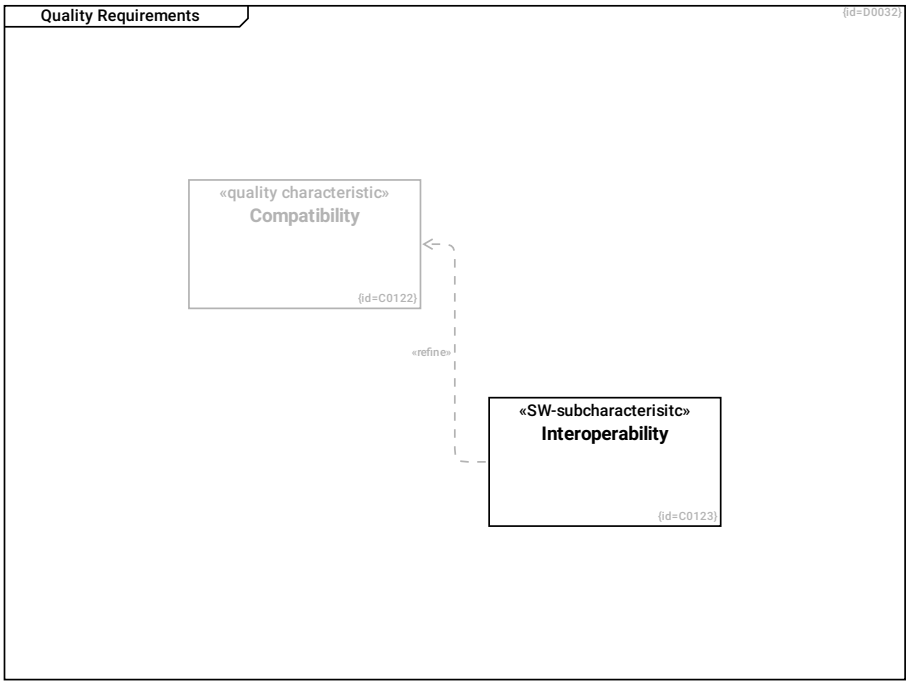
Rationale: Reacting on a false alarm is not mission-critical.

--> Self-Preservation R0177

1.5.3.10 Quality Requirements

This section shows the major quality scenarios. (Problem Space, Software Level L3)

Similar to Section 1.10: Quality Requirements for system level L1, this section shows quality expectations: The WHAT shall be implemented, not the HOW.



Compatibility C0122

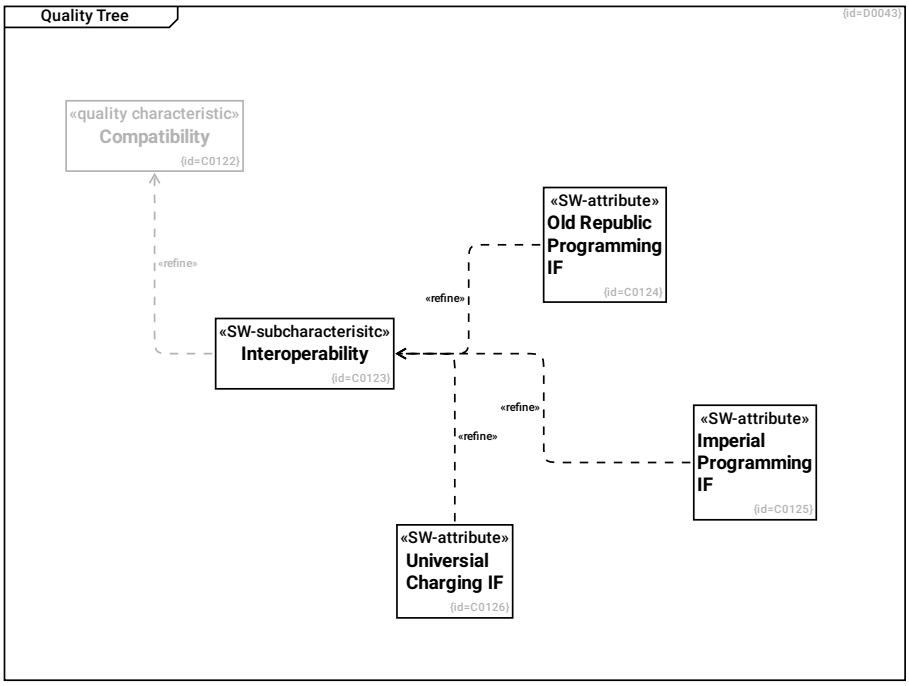
Interoperability C0123

The programming and charging interfaces of the MoD5G shall be compatible to

- old republic terminals
- imperial terminals

--> Compatibility R0158

1.5.3.10.1 Quality Tree



Compatibility C0122

Old Republic Programming IF C0124

The old republic protocol for programming a droid shall be supported.

--> Interoperability R0159

Interoperability C0123

The programming and charging interfaces of the MoD5G shall be compatible to

- old republic terminals
- imperial terminals

--> Compatibility R0158

Imperial Programming IF C0125

The imperial protocol for programming a droid shall be supported.

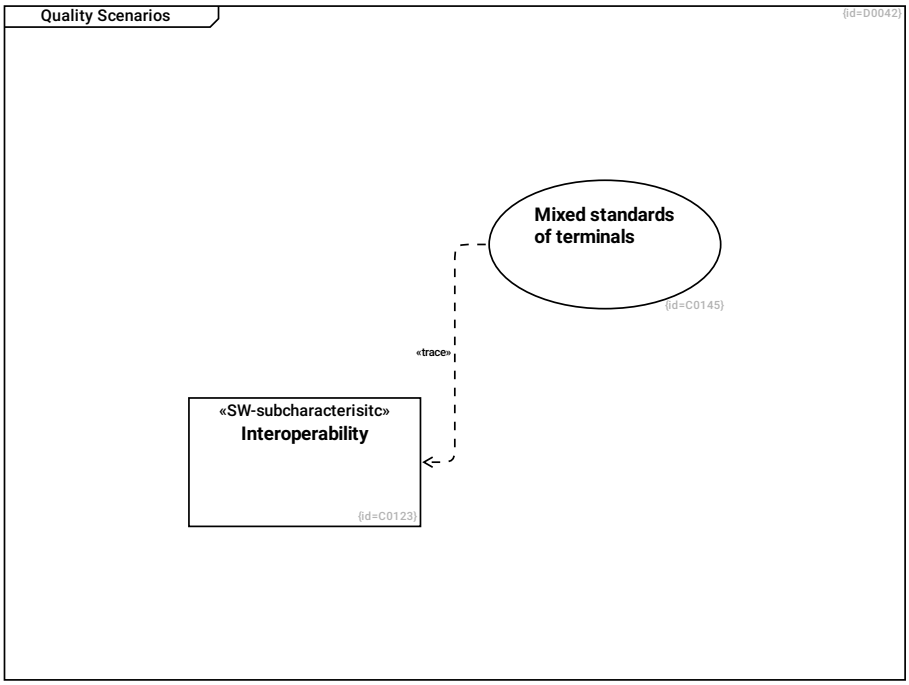
--> Interoperability R0160

Universial Charging IF C0126

The intergalactic standard protocol for power charging shall be supported.

--> Interoperability R0161

1.5.3.10.2 Quality Scenarios



Mixed standards of terminals C0145

precondition:

- The MoD5G operates in an environment providing mixed terminal standards

trigger:

- The MoD5G drives to a charging or programming terminal which complied to either old republic or imperial standard.

scenario:

- The MoD5G determines the applicable standard
- The MoD5G uses the terminal for programming or charging

--> Interoperability R0227

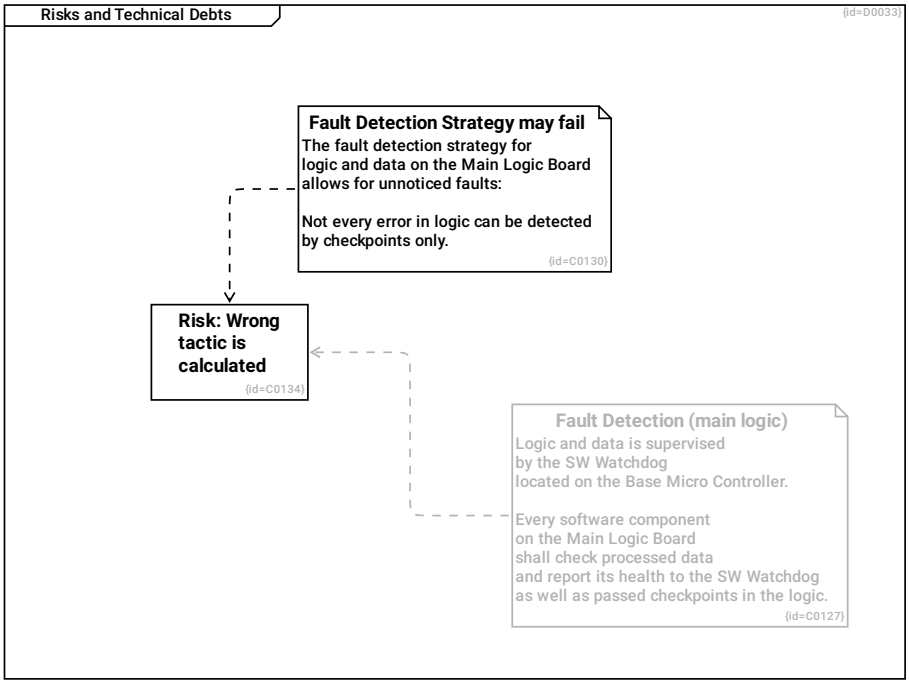
Interoperability C0123

The programming and charging interfaces of the MoD5G shall be compatible to

- old republic terminals
- imperial terminals

1.5.3.11 Risks and Technical Debts

This section lists the risks and not-yet-addressed requirements. (Solution Space, Software Level L3)



Risk: Wrong tactic is calculated C0134

- cause/fault: Due to cosmic rays, the main logic board performs a miscalculation that goes unnoticed by control flow supervision
- risk/failure: the MoD5G calculates a tactic that results in falling off a cliff

Fault Detection (main logic) C0127

Logic and data is supervised by the SW Watchdog located on the Base Micro Controller.

Every software component on the Main Logic Board shall check processed data and report its health to the SW Watchdog as well as passed checkpoints in the logic.

--> Risk: Wrong tactic is calculated R0199

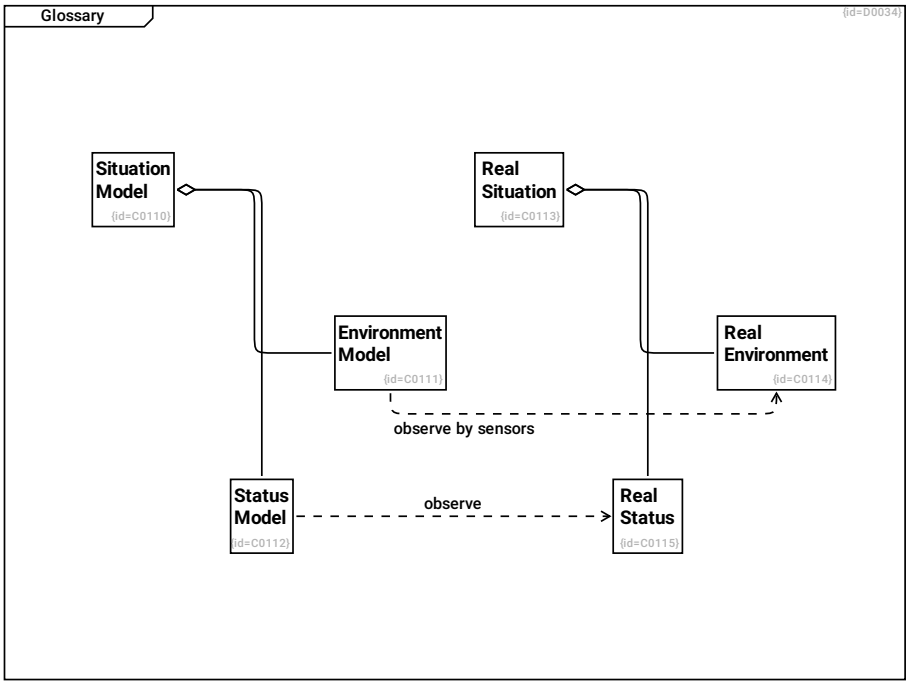
Fault Detection Strategy may fail C0130

The fault detection strategy for logic and data on the Main Logic Board allows for unnoticed faults: Not every error in logic can be detected by checkpoints only.

--> Risk: Wrong tactic is calculated R0181

1.5.3.12 Glossary

This section explains the used terms. (Domain and Solution Space, Software Level L3)



Real Situation C0113

The real situation refers to the reality of system status and environment.

--> Real Status R0144

--> Real Environment R0145

Situation Model C0110

The situation model refers to the (limited) knowledge of the software on environment and status.

--> Status Model R0142

--> Environment Model R0143

Environment Model C0111

The enironment model refers to the (limited) knowledge of the software on the real environment.

observe by sensors --> Real Environment R0146

Real Environment C0114

The real environment refers to the physical environment of the system.

Status Model C0112

The status model refers to the (limited) knowledge of the software on the real status.

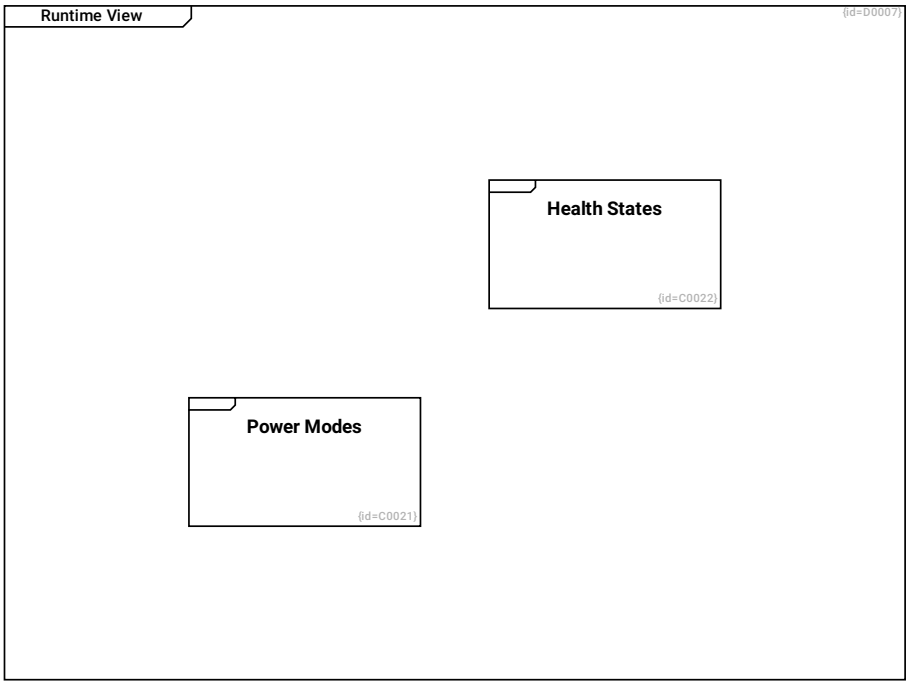
observe --> Real Status R0147

Real Status C0115

The real status refers to the real system status. This may differ from what the sensors report.

1.6 Runtime View

This section shows the dynamic behavior of the system (Solution Space, System Level L1)



Health States C0022

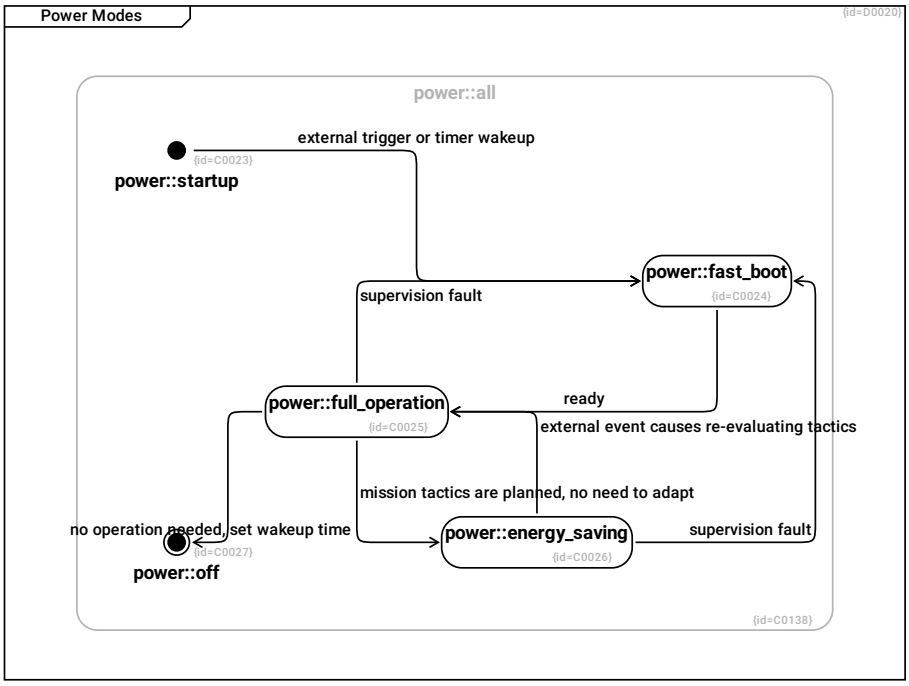
refers to Section 1.6.2: Health States

Power Modes C0021

refers to Section 1.6.1: Power Modes

1.6.1 Power Modes

This diagram shows the power states that are globally valid to all parts of the system.



power::all C0138

--> power::fast_boot R0192

--> power::energy_saving R0193

--> power::full_operation R0194

--> power::startup R0195

--> power::off R0196

power::full_operation C0025

mission tactics are planned, no need to adapt --> power::energy_saving R0017

no operation needed, set wakeup time --> power::off R0019

supervision fault --> power::fast_boot R0033

power::fast_boot C0024

ready --> power::full_operation R0016

power::energy_saving C0026

external event causes re-evaluating tactics --> power::full_operation R0018

supervision fault --> power::fast_boot R0032

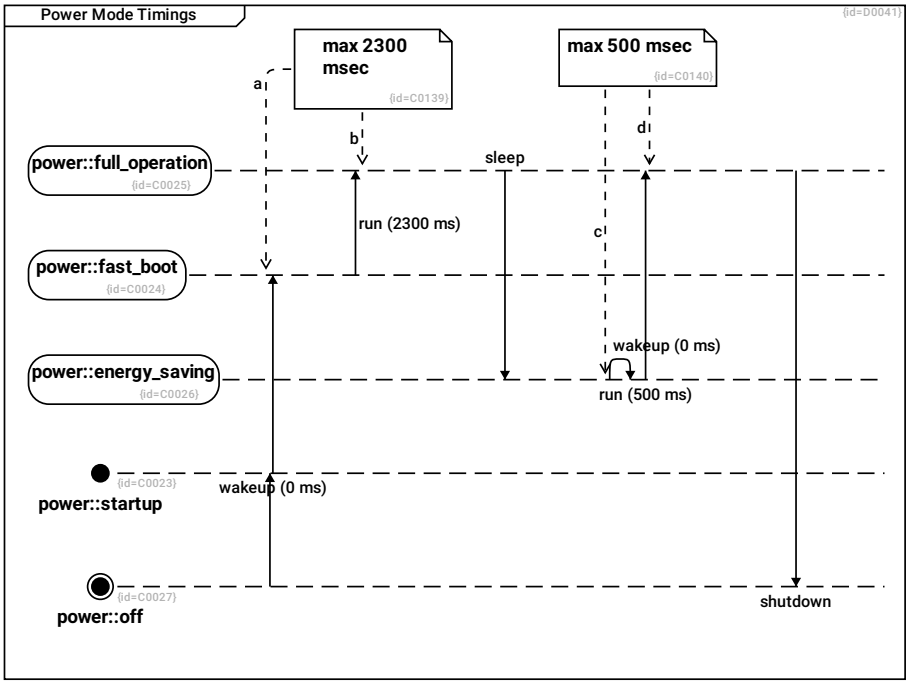
power::startup C0023

external trigger or timer wakeup --> power::fast_boot R0015

power::off C0027

1.6.1.1 Power Mode Timings

This diagram shows the expected startup and shutdown timings.



power::full_operation C0025

sleep --> power::energy_saving R0212

shutdown --> power::off R0215

power::fast_boot C0024

run (2300 ms) --> power::full_operation R0211

power::energy_saving C0026

wakeup (0 ms) --> power::energy_saving R0213

run (500 ms) --> power::full_operation R0214

max 2300 msec C0139

a --> power::fast_boot R0218

b --> power::full_operation R0216

power::startup C0023

wakeup (0 ms) --> power::fast_boot R0210

power::off C0027

--> power::startup R0209

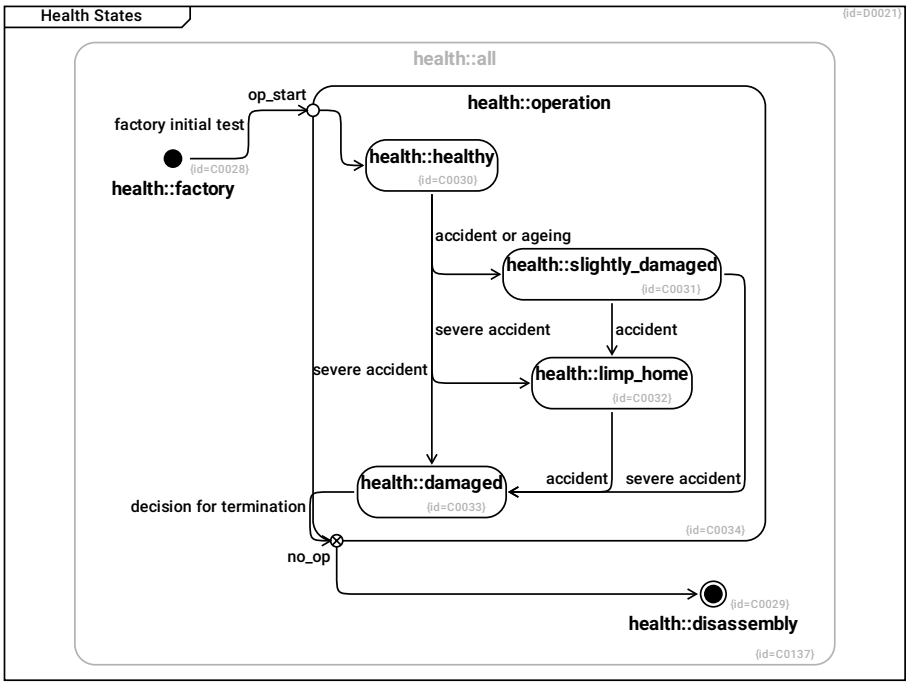
max 500 msec C0140

c --> power::energy_saving R0219

d --> power::full_operation R0217

1.6.2 Health States

This diagram shows the health states of the MoD5G system.



health::healthy C0030

accident or ageing --> health::slightly_damaged R0021

severe accident --> health::damaged R0029

severe accident --> health::limp_home R0030

health::slightly_damaged C0031

accident --> health::limp_home R0022

severe accident --> health::damaged R0031

health::limp_home C0032

accident --> health::damaged R0023

health::factory C0028

factory initial test --> health::operation R0207

health::operation C0034

op_start F0019

no_op F0018

--> health::limp_home R0024

--> health::slightly_damaged R0025

--> health::healthy R0026

--> health::damaged R0027

--> health::disassembly R0205

--> health::healthy R0208

health::all C0137

--> health::factory R0189

--> health::operation R0190

--> health::disassembly R0191

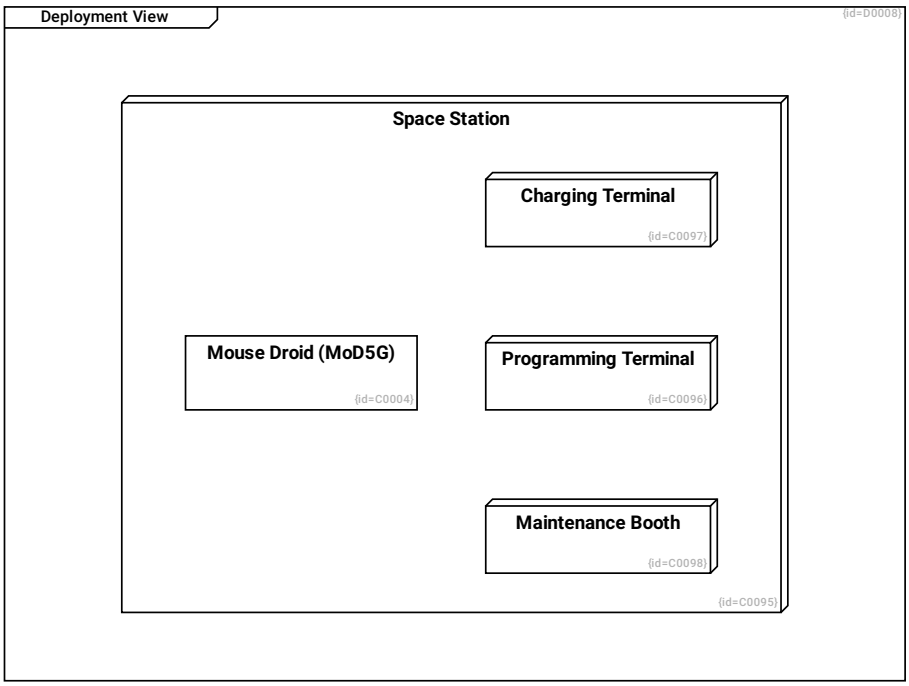
health::damaged C0033

decision for termination --> health::operation R0206

health::disassembly C0029

1.7 Deployment View

This section shows the deployment of the solution into the environment. (Solution Space, System Level L1)



Space Station C0095

--> Programming Terminal R0129

--> Charging Terminal R0130

--> Maintenance Booth R0131

--> Mouse Droid (MoD5G) R0128

Programming Terminal C0096

Charging Terminal C0097

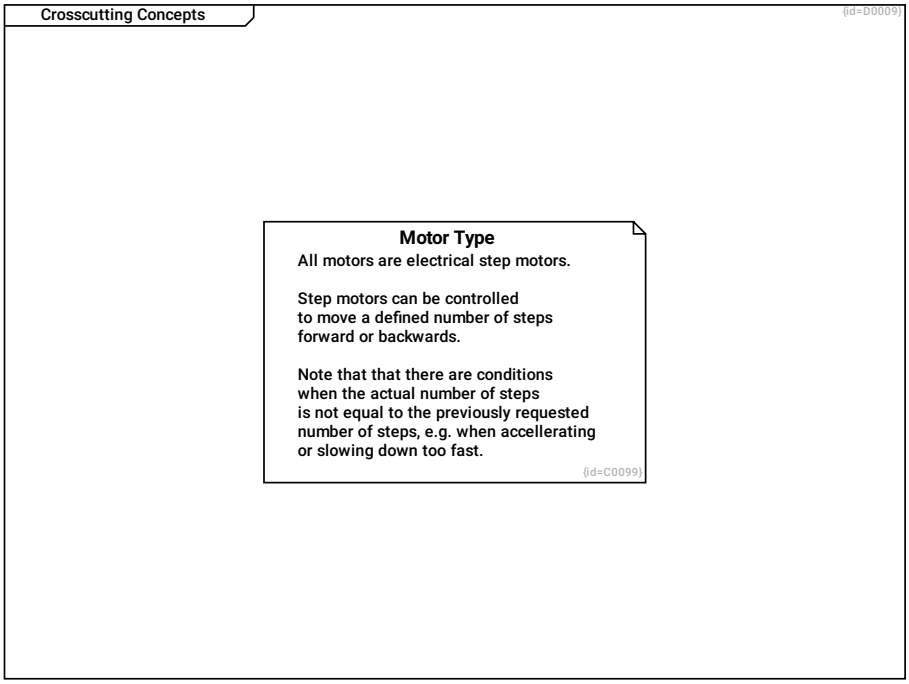
Maintenance Booth C0098

Mouse Droid (MoD5G) C0004

The Mouse Droid (MoD5G) is a repair droid that can be instructed to perform a mission and which autonomously selects tactics to achieve the mission goals

1.8 Crosscutting Concepts

This section shows the recurring concepts within the the designed solution. (Solution Space, System Level L1)



Motor Type C0099

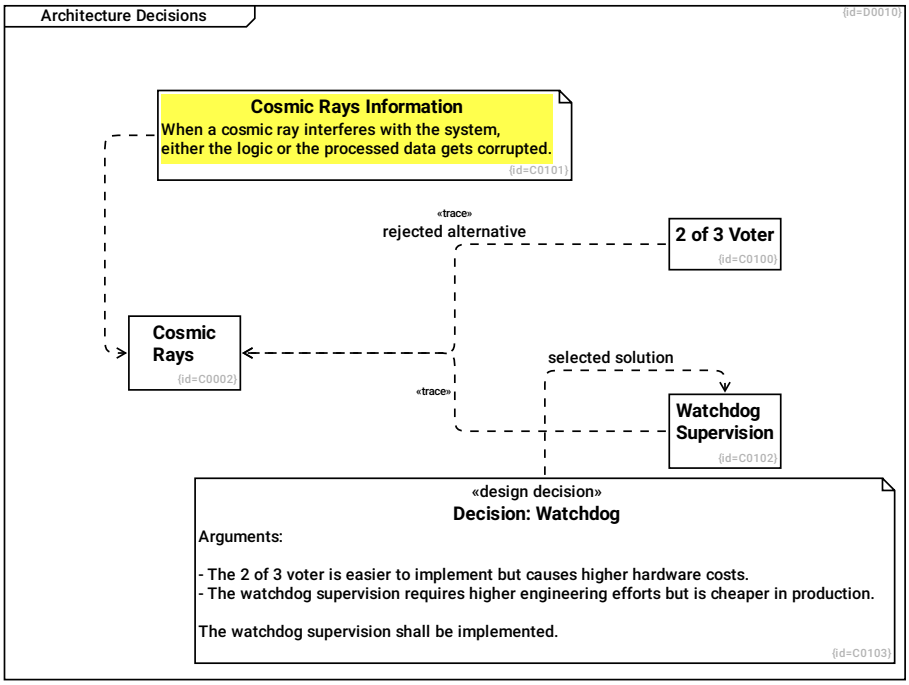
All motors are electrical step motors.

Step motors can be controlled to move a defined number of steps forward or backwards.

Note that that there are conditions when the actual number of steps is not equal to the previously requested number of steps, e.g. when accellerating or slowing down too fast.

1.9 Architecture Decisions

This section explains the major and non-obvious design decisions. (Solution Space, System Level L1)



2 of 3 Voter C0100

In order to support integrity of the system, the logic boards and the data storages are deployed three times as three identical parts.

All three parts shall produce the same outcomes given the same input.

If one deviates, it's result is ignored and the part is rebooted.

rejected alternative --> Cosmic Rays R0135

Watchdog Supervision C0102

In order to support integrity of the logic and data, a multi-stage hierarchy supervision shall be implemented.

Software watchdogs shall supervise the running software parts in a way that logic errors and corrupted data can be detected.

A hardware watchdog shall supervise the software watchdogs.

In case of a failure in the supervised logic/data, the system shall reboot. In case of a failure in the monitors, the system may reboot or it shall fall back to a valid supervision mode.

--> Cosmic Rays R0134

Cosmic Rays C0002

The droid shall ensure data and program integrity and continue operation after cosmic rays may have interfered with normal operation.

Corrupted data must not be stored permanently.

Cosmic Rays Information C0101

When a cosmic ray interferes with the system, either the logic or the processed data gets corrupted.

--> Cosmic Rays R0132

Decision: Watchdog C0103

Arguments:

- The 2 of 3 voter is easier to implement but causes higher hardware costs.

- The watchdog supervision requires higher engineering efforts but is cheaper in production.

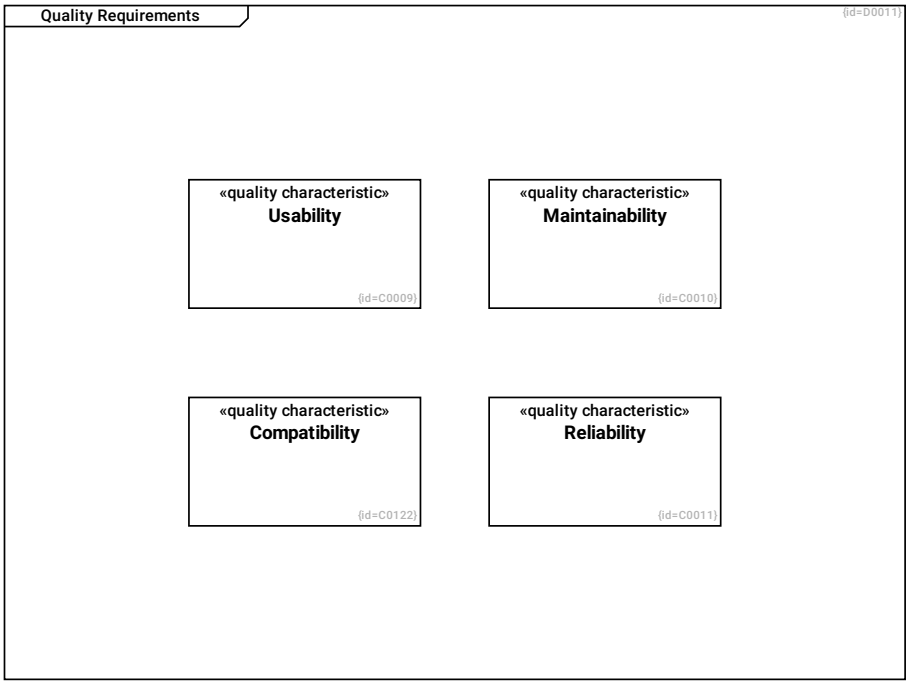
The watchdog supervision shall be implemented.

selected solution --> Watchdog Supervision R0133

1.10 Quality Requirements

This section shows the major quality requrements and scenarios. (Problem Space, System Level L1)

In the following, requirements and scenarios are selected that show the quality expectations: The WHAT shall be implemented, not the HOW.



Compatibility C0122

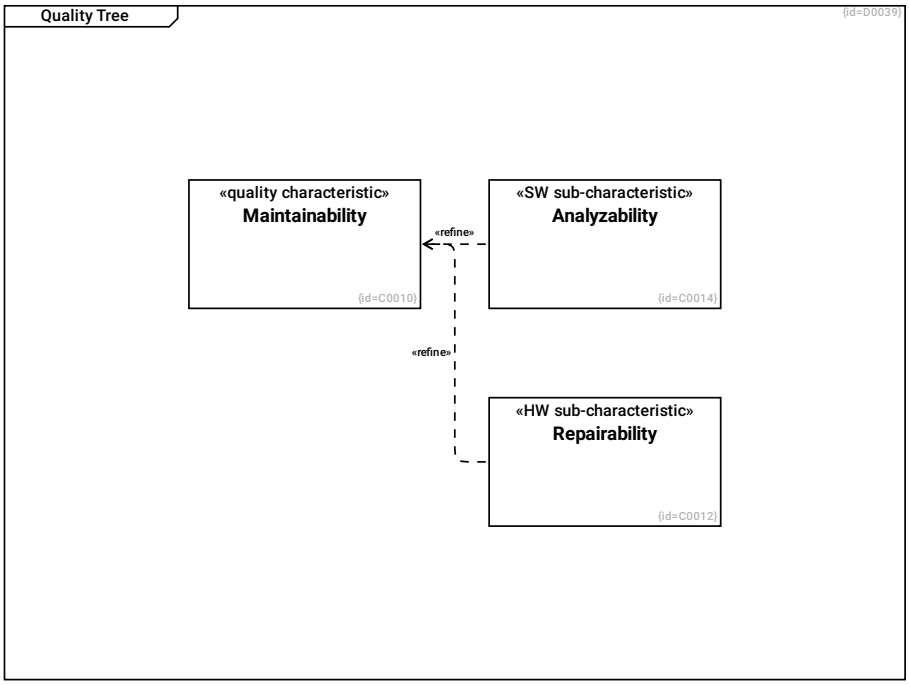
Maintainability C0010

Reliability C0011

Usability C0009

1.10.1 Quality Tree

This section shows the quality requirements ordered by quality characteristics.



Analyzability C0014

The MoD5G shall allow to analyze faults that occurred during operation.

--> Maintainability R0009

Maintainability C0010

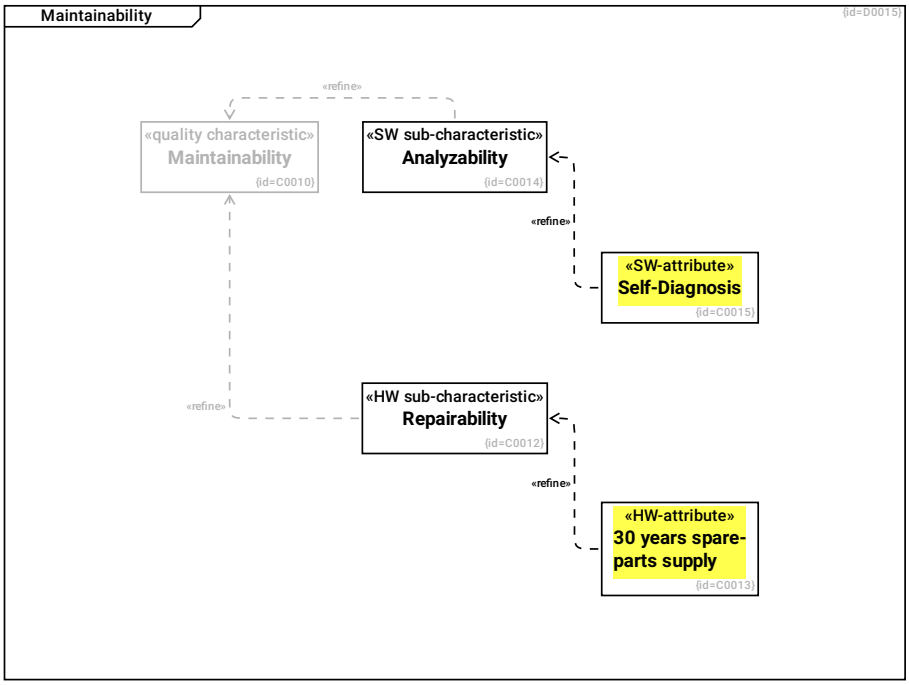
Repairability C0012

The MoD5 hardware parts shall be exchangeable in case they are damaged.

--> Maintainability R0007

1.10.1.1 Maintainability

This diagram shows the quality requirements related to the characteristic "Maintainability".



Analyzability C0014

The MoD5G shall allow to analyze faults that occurred during operation.

--> Maintainability R0009

Self-Diagnosis C0015

At the maintenance booth, the MoD5G shall provide an error log. This error log contains detected errors from operation and related environment conditions. It also lists possible causes(faults).

--> Analyzability R0010

Maintainability C0010

Repairability C0012

The MoD5 hardware parts shall be exchangeable in case they are damaged.

--> Maintainability R0007

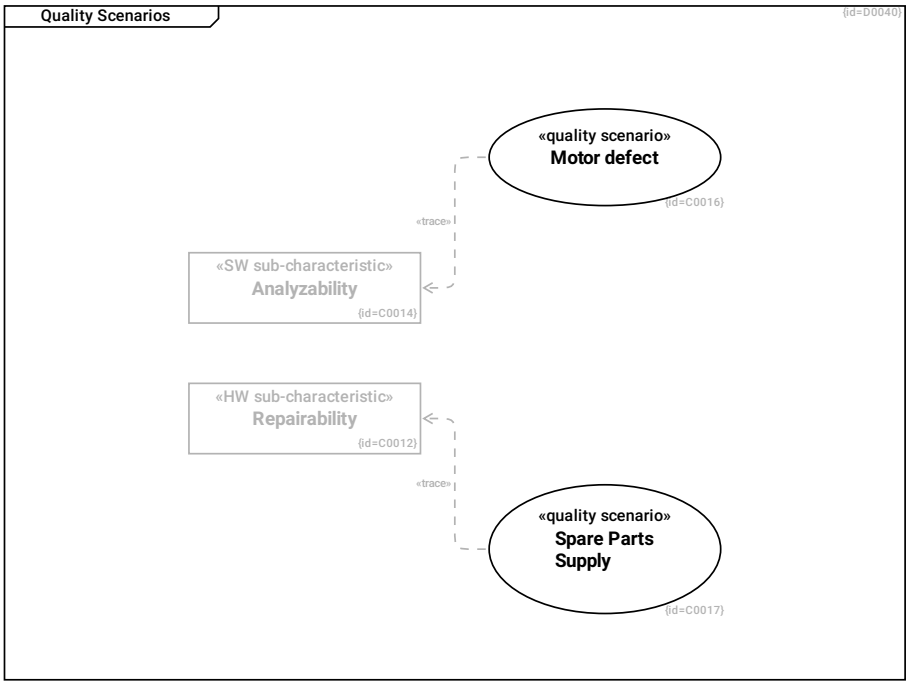
30 years spare-parts supply C0013

The mechanical and electrical/electronics parts of the MoD5 shall be produceable in identical or similar form and quality for 30 years after production of the unit.

--> Repairability R0008

1.10.2 Quality Scenarios

This section shows the Quality Scenarios in which the Quality Requirements shown in Section 1.10.1: Quality Tree are of special importance.



Motor defect C0016

pre-condition:

- the MoD5G is performing a 1-day mission autonomously

trigger:

- a motor fails to operate
- the goals of the 1-day mission cannot be accomplished anymore

scenario:

- the MoD5G cancels the mission and returns to the service point
- a service mechanic reads out the error log
- the MoD5G proposes to replace the suspicious motor
- the service mechanic replaces the motor

--> Analyzability R0197

Analyzability C0014

The MoD5G shall allow to analyze faults that occurred during operation.

Spare Parts Supply C0017

pre-condition:

- the stock of MoD5G spare parts is empty

trigger:

- 20 years after production, a MoD5G needs a spare part that is not available anymore

scenario:

- a service mechanic orders a batch of parts
- a factory creates the parts that fit in form and quality to the MoD5G
- spare parts are delivered

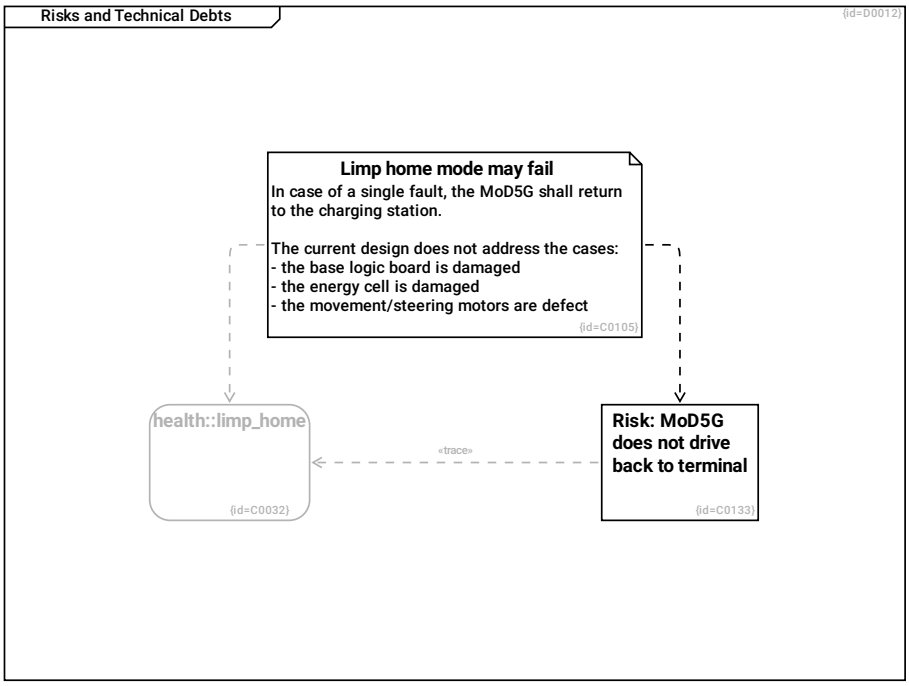
--> Repairability R0012

Repairability C0012

The MoD5 hardware parts shall be exchangeable in case they are damaged.

1.11 Risks and Technical Debts

This section lists the risks and not-yet-addressed requirements. (Solution Space, System Level L1)



health::limp_home C0032

Risk: MoD5G does not drive back to terminal C0133

- cause/fault: the base logic board is damaged
- risk/failure: the MoD5G cannot drive anymore

--> health::limp_home R0180

Limp home mode may fail C0105

In case of a single fault, the MoD5G shall return to the charging station.

The current design does not address the cases:

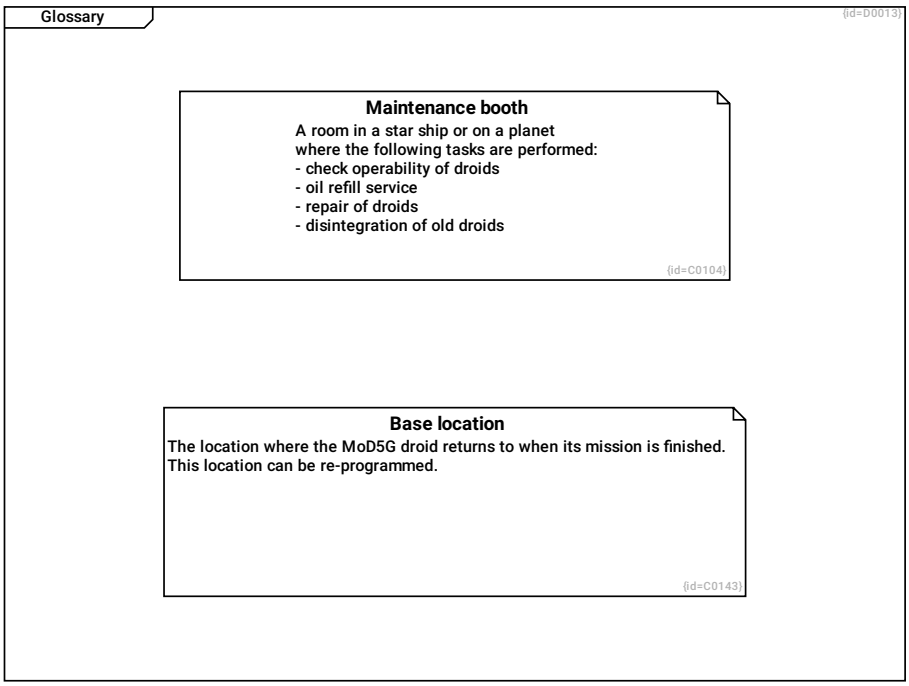
- the base logic board is damaged
- the energy cell is damaged
- the movement/steering motors are defect

--> health::limp_home R0136

--> Risk: MoD5G does not drive back to terminal R0179

1.12 Glossary

This section explains the used terms. (Domain and Solution Space, System Level L1)



Maintenance booth C0104

A room in a star ship or on a planet where the following tasks are performed:

- check operability of droids
- oil refill service
- repair of droids
- disintegration of old droids

Base location C0143

The location where the MoD5G droid returns to when its mission is finished. This location can be re-programmed.