

Semántica del Diseño  
Un puente entre los requerimientos y el diseño  
Versión 0.8

Laura Pomponio

27 de septiembre de 2022

CONICET



---

C I F A S I S

# Índice general

<b>1. Introducción</b>	<b>3</b>
1.1. Acerca de este documento	3
1.2. Lectura de este documento	3
1.2.1. Árboles de invocación o utilización	3
<b>2. Semántica del Diseño</b>	<b>5</b>
2.1. Requerimiento RF-1.	5
2.2. Requerimiento RF-1.1.	5
2.3. Requerimiento RF-1.2.	5
2.4. Requerimiento RF-1.3.	5
2.5. Requerimiento RF-2.	5
2.6. Requerimientos RF-2.1. – RF-2.5.	5
2.7. Requerimiento RF-3.	6
2.8. Requerimiento RF-4.	6
2.9. Requerimiento RF-5.	6
2.10. Requerimiento RF-6.	6
2.11. Requerimiento RF-7.	6
2.12. Requerimiento RF-8.	6
2.13. Requerimiento RF-9.	7
2.14. Requerimiento RF-10.	7
2.15. Requerimiento RF-11.	7
2.16. Requerimiento RF-12.	7
2.17. Requerimiento RF-13.	7
2.18. Requerimiento RF-14.	7
2.19. Requerimiento RF-15.	7
2.20. Requerimiento RF-16.	8
2.21. Requerimiento RF-17.	8
2.22. Requerimiento RF-18.	8
2.23. Requerimientos RF-19. – RF-20.	8
2.24. Requerimientos RF-21. – RF-21.1.	8
2.25. Requerimientos RF-22., RF-22.1. – RF-22.4.	8
2.26. Requerimiento RF-23.	9
2.27. Requerimiento RF-24.	10
2.28. Requerimientos RF-25. – RF-26.	10
2.29. Requerimientos RF-27., RF-27.1. – RF-27.2.	11
2.30. Requerimiento RF-28.	11
2.31. Requerimiento RF-29.	11
2.32. Requerimiento RF-30.	11
2.33. Requerimiento RF-31.	12
2.34. Requerimiento RF-32.	12
2.35. Requerimiento RF-33.	13
2.36. Requerimiento RF-34.	13
2.37. Requerimiento RF-35.	13
2.38. Requerimientos RF-36. – RF-37.	14
2.39. Requerimiento RF-38.	14
2.40. Requerimiento RF-39.	15
2.41. Requerimiento RF-40.	15
2.42. Requerimiento RF-41.	16
2.43. Requerimientos RF-42. – RF-43.	16
2.44. Requerimiento RF-44.	16
2.45. Requerimientos RF-45. – RF-46.	16

2.46. Requerimientos RF-47. – RF-48.	17
2.47. Requerimiento RF-49.	17
2.48. Requerimiento RF-50.	17
2.49. Requerimiento RF-51.	18
2.50. Requerimiento RF-52.	18
2.51. Requerimientos RF-53. RF-54.	18
2.52. Requerimiento RF-55.	19
2.53. Requerimiento RF-56.	19
2.54. Requerimientos RF-57. – RF-58.	19
2.55. Requerimiento RF-59.	20
2.56. Requerimientos RF-60. RF-61.	20
2.57. Requerimiento RF-62.	21
2.58. Requerimiento RF-63.	21
2.59. Requerimiento RF-64.	21
2.60. Requerimiento RF-65.	22
2.61. Requerimiento RF-66.	22
2.62. Requerimiento RF-67.	23
2.63. Requerimiento RF-68.	25
2.64. Requerimiento RF-69.	26
2.65. Requerimiento RF-70.	26
2.66. Requerimiento RF-71.	27
2.67. Requerimiento RF-72.	28
2.68. Requerimiento RF-73.	29
2.69. Requerimiento RF-73.1.	29
2.70. Requerimiento RF-73.2.	29
2.71. Requerimiento RF-73.3.	29
2.72. Requerimientos RF-73.4. – RF-73.5.	30
2.73. Requerimientos RF-74. – RF-75.	30
2.74. Requerimiento RF-75.1.	31
2.75. Requerimiento RF-75.2.	31
2.76. Requerimiento RF-76.	31
2.77. Requerimiento RF-77.	31

# Capítulo 1

## Introducción

### 1.1. Acerca de este documento

El presente documento pretende establecer un puente entre los requerimientos funcionales y cómo estos serán implementados a través del diseño.

El diseño basado en ocultamiento de la información y modularización, si bien es muy flexible para adaptarse a cambios en los requerimientos, también es en ocasiones difícil de leer y comprender. Esto es debido a que el software estará constituido por un número importante de pequeñas porciones de código que se relacionarán de diversas formas entre sí. De este modo, ante un cambio en los requerimientos, resulta difícil establecer rápidamente dónde debe ser modificado el diseño y, en consecuencia, el código.

El objetivo de este documento es que fácilmente, el diseñador e incluso el programador, puedan detectar las porciones del sistema que implementarán cierto requerimiento. Tener una clara comprensión sobre dónde y cómo se implementan los requerimientos, permitirá detectar fácilmente cuáles son las modificaciones que debería sufrir el diseño ante algún cambio requerido. Estos cambios podrían ser importantes, involucrando varios módulos, o bien menores como el valor de alguna constante. Por ej, en este último caso, un valor constante puede ser utilizado sólo por un módulo y por tanto ser un valor oculto por este, o bien puede ser utilizado por varios módulos y entonces estar dentro de un módulo que defina valores constantes. Determinar de un modo simple dónde la constante es definida, agilizará el proceso de implementar el cambio.

Dados los requerimientos del microcontrolador del robot desmalezador definidos en [Pom22b] y el diseño del sistema descrito en [Pom22a], este documento describe la relación entre estos, a través de enlaces hacia ambos documentos.

### 1.2. Lectura de este documento

Este documento debe ir acompañado de los documentos de requerimientos y diseño que deberán estar disponibles en el mismo directorio. Esto se debe a que en el presente documento, cada requerimiento mencionado será un enlace al documento de requerimientos donde el mismo está descrito; y cada módulo o método introducido será un enlace al documento de diseño, en particular, a la figura que ilustra el módulo con sus métodos.

De este modo cada requerimiento estará asociado a una lista de módulos y/o métodos o bien a un árbol en el que aparezcan estos. En el primer caso, la lista indicará que los métodos o módulos descritos, están involucrados en la funcionalidad que cumple con el requerimiento. En el segundo caso, dicha funcionalidad se describe a través de árboles de invocación o utilización, los cuales son presentados más abajo.

A continuación introducimos el significado de ciertas notaciones, y en la siguiente subsección, describimos el significado de los árboles de invocación o utilización.

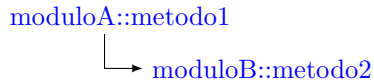
**RF-<número>** : enlace a la descripción del requerimiento en el documento de requerimientos.

**modulo | modulo::metodo** : enlace a la figura donde está descrito el mencionado módulo o método, dentro del documento de diseño.

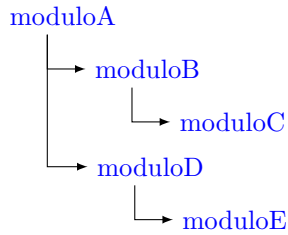
#### 1.2.1. Árboles de invocación o utilización

Estos árboles describen cierta secuencialidad o paralelismo entre la invocación o utilización de métodos o módulos. **El sentido de secuencialidad, refiere a que una cosa ocurre después de otra, pero pudiendo suceder entre ambas, otras cosas más.** Esto es porque estos árboles describirán cómo se irán ejecutando las distintas porciones del software que implementarán un requerimiento determinado. De este modo, en los árboles aparecerán los módulos o métodos involucrados en dicha ejecución que competen exclusivamente al requerimiento en cuestión, omitiendo los otros elementos que también son ejecutados pero que no son de interés al momento de comprender,

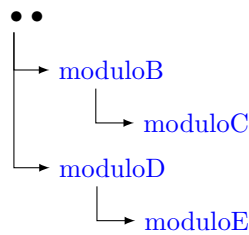
cómo el requerimiento se implementa en el diseño.



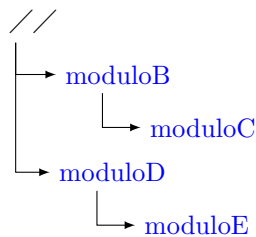
Un método (también podría ser un módulo), **invoca** o **utiliza** otro método (también podría ser un módulo).



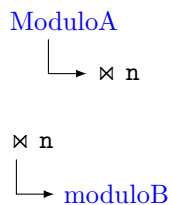
Invocación o uso secuencial originado en un módulo (o método). Este árbol describe cómo los módulos o métodos se invocan o utilizan en forma secuencial. El recorrido del árbol es de arriba hacia abajo, primero en profundidad. En el ejemplo, esto es: primero **moduloA** utiliza **moduloB**, el cual a su vez utiliza el **moduloC**. Luego, el **moduloA** utiliza **moduloD**, el cual utiliza el **moduloE**. Esto describe entonces un sólo hilo de ejecución, originado en el **moduloA**, donde una cosa ocurre después de otra.



Invocación o uso secuencial. El recorrido del árbol es de arriba hacia abajo, primero en profundidad. El mencionado árbol describe cómo las distintas ramas que se desprenden de los dos puntos, serán ejecutadas una después de otra. El origen de las ramas no es un módulo o método que invoca secuencialmente otros módulos o métodos; por el contrario, puede ser originado por distintos módulos o bien por el mismo módulo en dos ocasiones diferentes. Por ejemplo, podría ser la ocurrencia de un evento externo que ocurre dos veces. En el ejemplo, la secuencia es la siguiente: primero **moduloB** usa o invoca al **moduloC**; y luego, **moduloD** usa o invoca al **moduloE**.



Invocación o uso secuencial y ejecución asíncrona. Este árbol describe cómo las distintas ramas que se desprenden de las dos barras, serán ejecutadas en forma asíncrona. Esto es, serán hilos de ejecución independientes entre sí. Sin embargo, cada rama describirá una intervención secuencial de sus actores. El recorrido de cada una de estas ramas es de arriba hacia abajo, primero en profundidad. En el ejemplo, hay dos hilos de ejecución independientes. **moduloB** usa o invoca al **moduloC**; y a la vez, **moduloD** usa o invoca al **moduloE**.



El símbolo  $\bowtie$  seguido de un número  $n$  indica que el árbol continúa, en otro árbol descrito más abajo cuya raíz es  $\bowtie n$ . Si como continuación se presenta más de árbol con el mismo  $\bowtie n$  como raíz, se está indicando que hay más de un camino posible como continuación.

## Capítulo 2

# Semántica del Diseño

### 2.1. Requerimiento **RF-1.**

Wheel, WCtrlCommand

---

### 2.2. Requerimiento **RF-1.1.**

Wheel

---

### 2.3. Requerimiento **RF-1.2.**

Wheel, MapData, MapFunction

---

### 2.4. Requerimiento **RF-1.3.**

Wheel

---

### 2.5. Requerimiento **RF-2.**

SteeringDevice

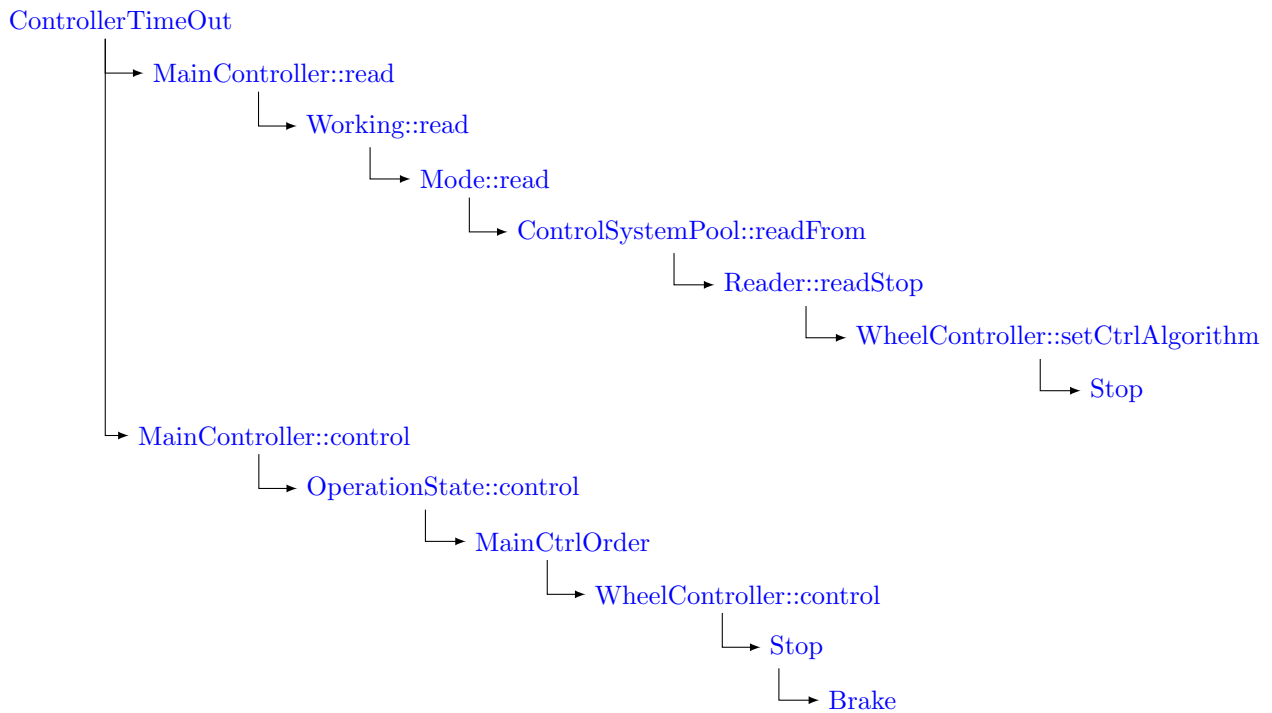
---

### 2.6. Requerimientos **RF-2.1. – RF-2.5.**

SteeringDevice

---

## 2.7. Requerimiento RF-3.



## 2.8. Requerimiento RF-4.

Mode, MainController

---

## 2.9. Requerimiento RF-5.

MCBuilder::getMC

---

## 2.10. Requerimiento RF-6.

RPM

---

## 2.11. Requerimiento RF-7.

RPM::setValue

---

## 2.12. Requerimiento RF-8.

RPM

---

2.13.   Requerimiento **RF-9.**

SignedPerc

---

2.14.   Requerimiento **RF-10.**

SignedPerc, Reader::readSetpointDir

---

2.15.   Requerimiento **RF-11.**

Percentage

---

2.16.   Requerimiento **RF-12.**

SignedPerc

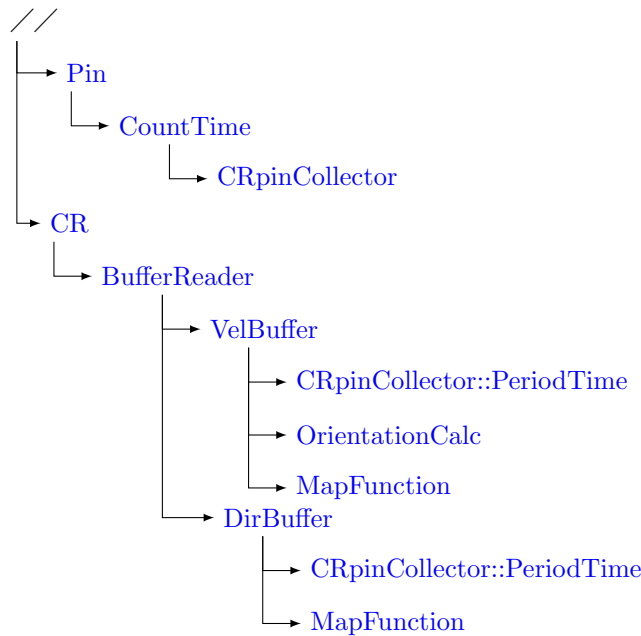
---

2.17.   Requerimiento **RF-13.**

MapFunction

---

2.18.   Requerimiento **RF-14.**



2.19.   Requerimiento **RF-15.**

Constants, VelBuffer::VelBuffer, DirBuffer::DirBuffer

---



2.20. Requerimiento **RF-16.**

Constants::Constants

---

2.21. Requerimiento **RF-17.**

Constants::Constants, VelBuffer::VelBuffer, BufferReader::BufferReader

---

2.22. Requerimiento **RF-18.**

Constants::Constants

---

2.23. Requerimientos **RF-19. – RF-20.**

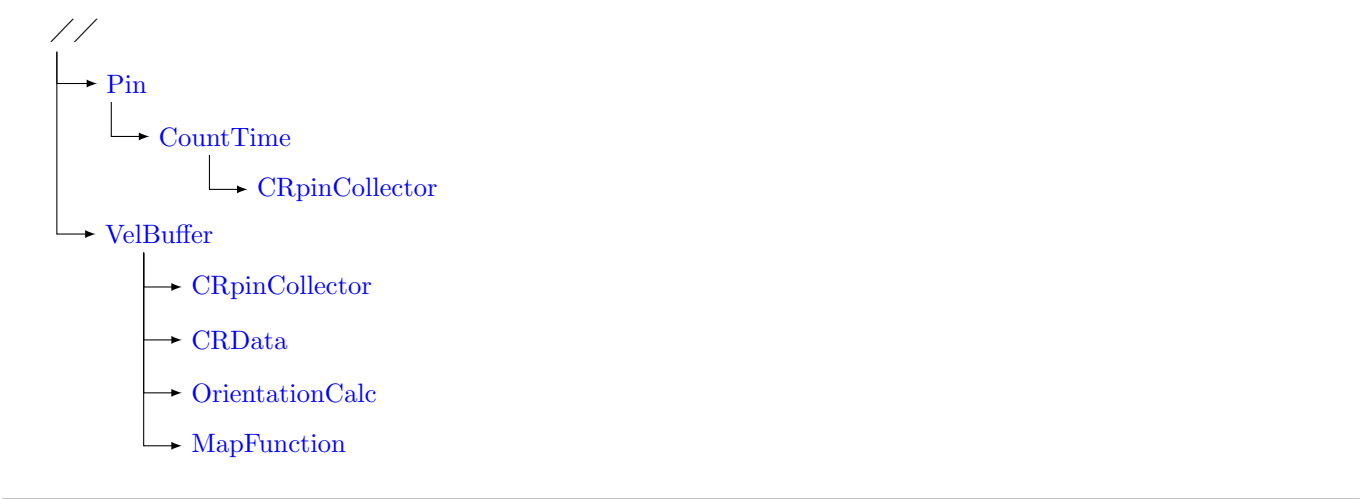
VelBuffer::VelBuffer

---

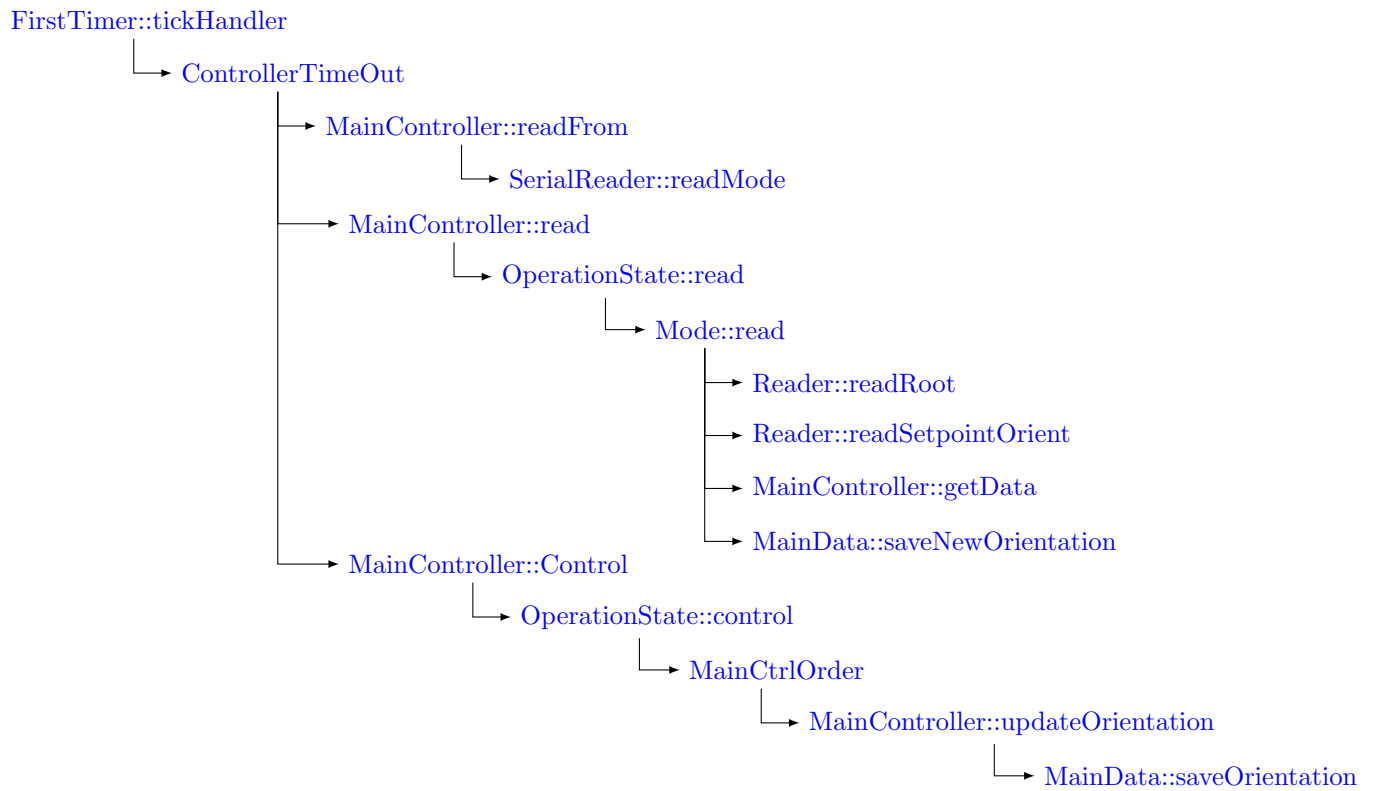
2.24. Requerimientos **RF-21. – RF-21.1.**



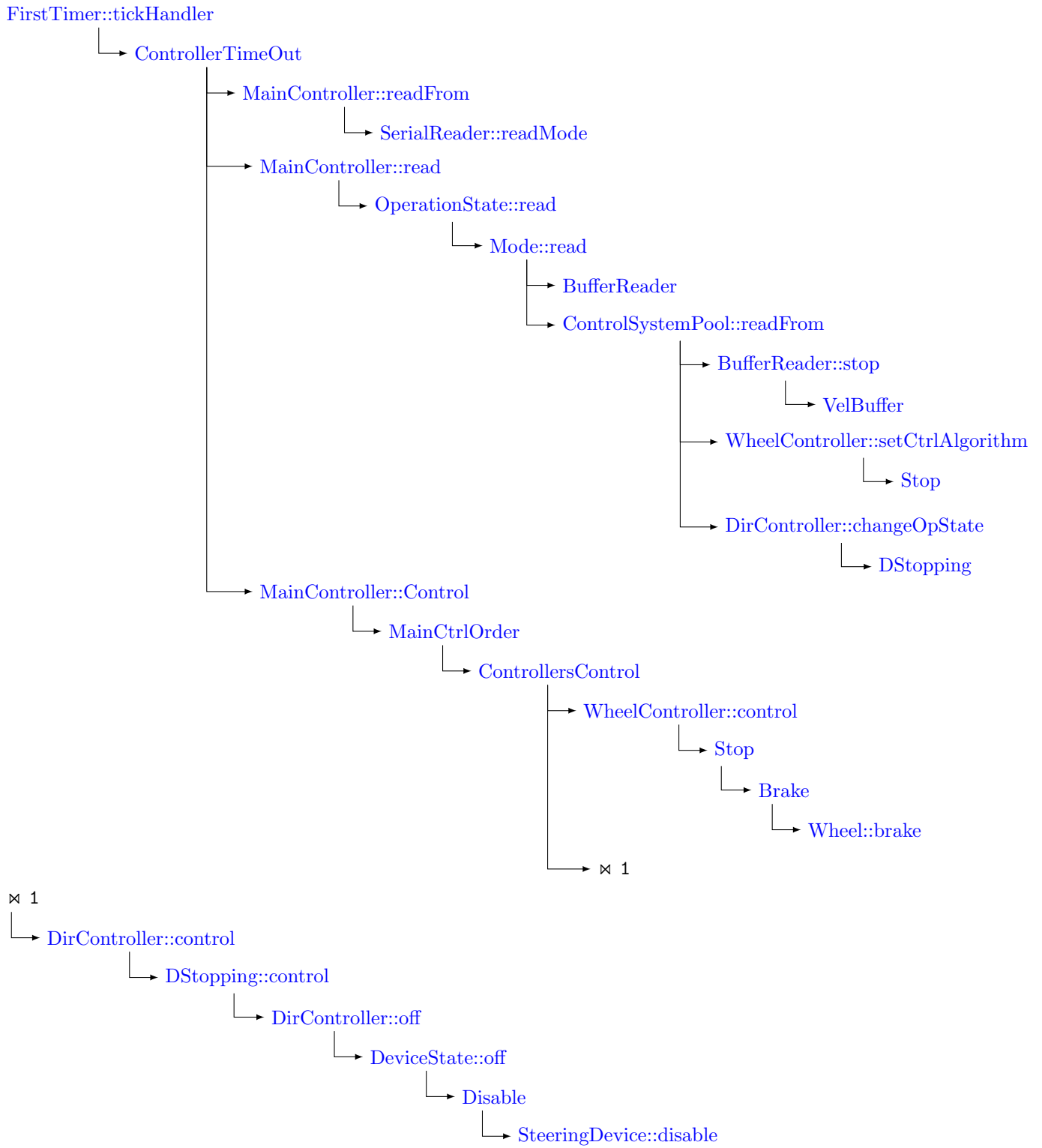
2.25. Requerimientos **RF-22., RF-22.1. – RF-22.4.**



## 2.26. Requerimiento RF-23.



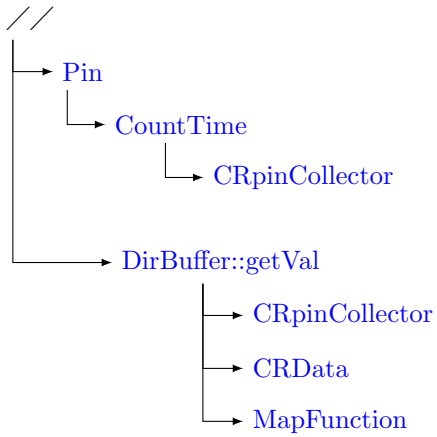
## 2.27. Requerimiento RF-24.



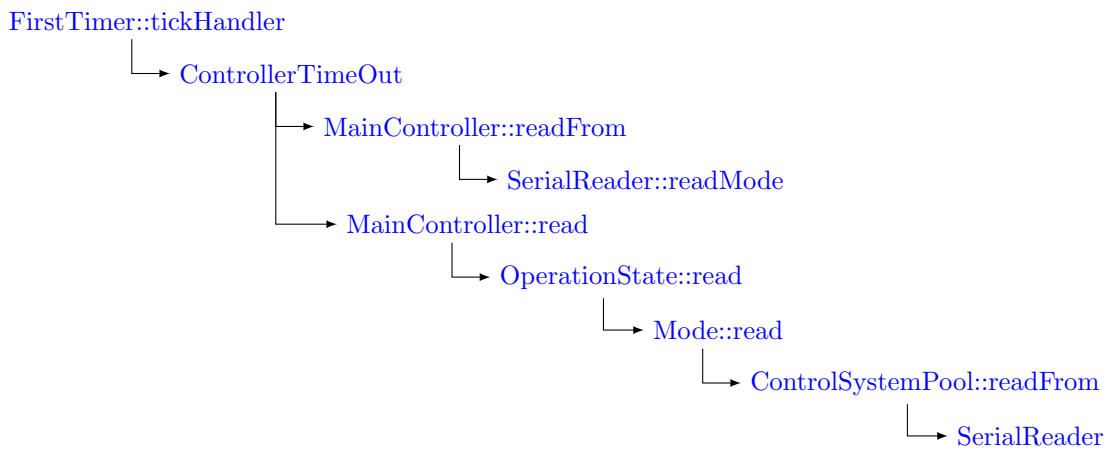
## 2.28. Requerimientos RF-25. – RF-26.

DirBuffer::DirBuffer

## 2.29. Requerimientos **RF-27.**, **RF-27.1.** – **RF-27.2.**



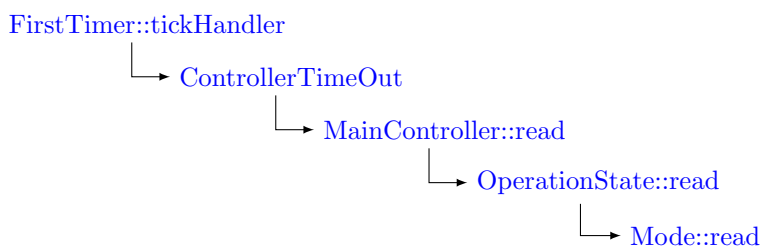
## 2.30. Requerimiento **RF-28.**



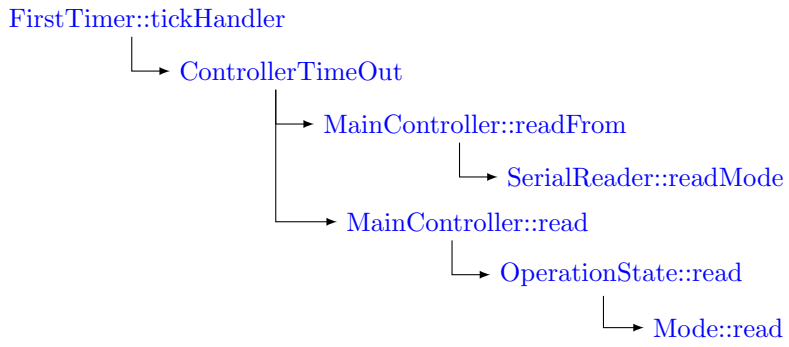
## 2.31. Requerimiento **RF-29.**

**SerialReader::newMessage**

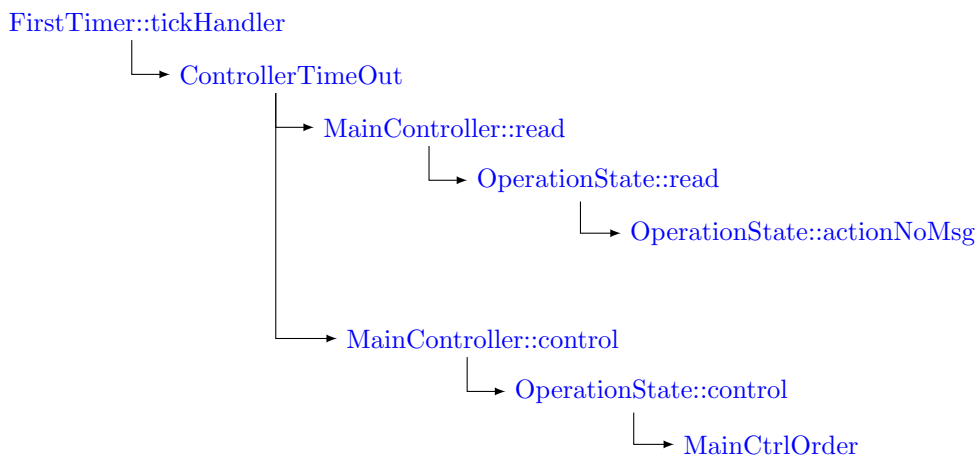
## 2.32. Requerimiento **RF-30.**



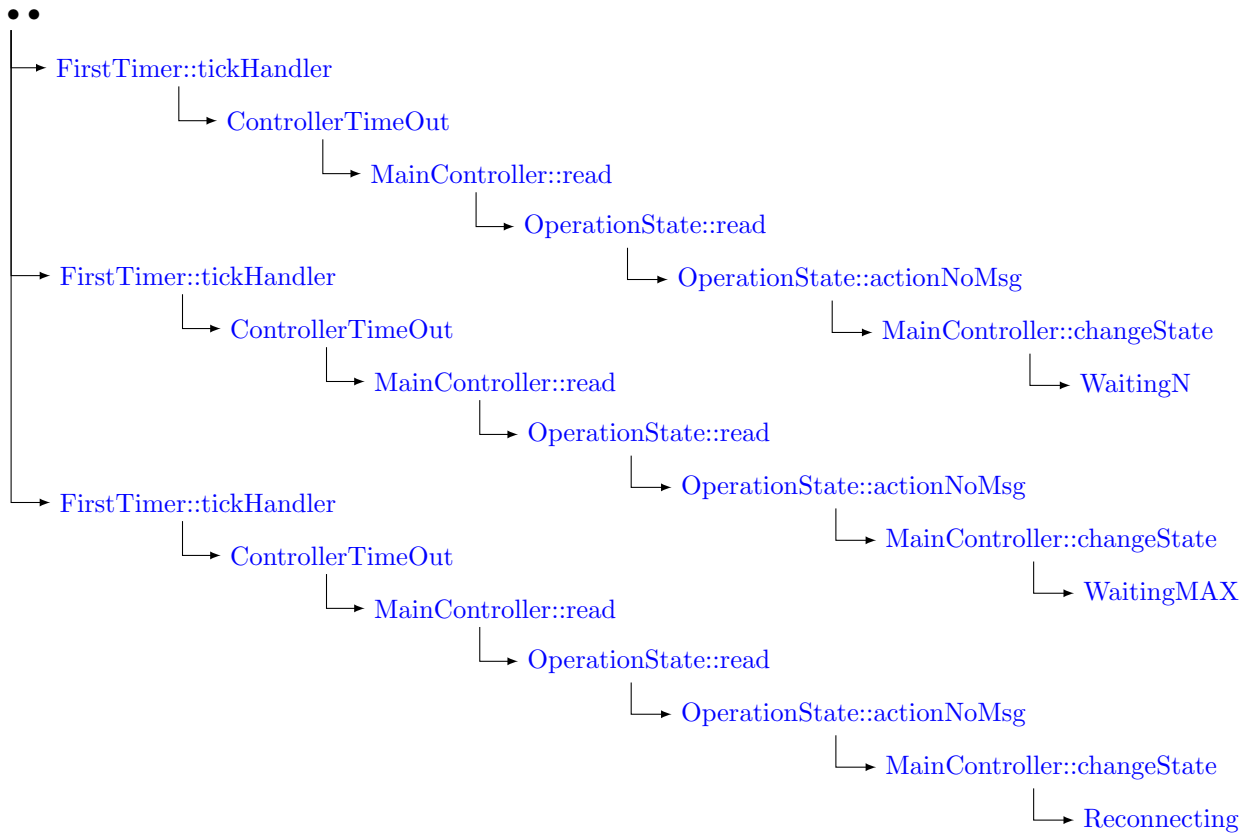
### 2.33. Requerimiento RF-31.



### 2.34. Requerimiento RF-32.



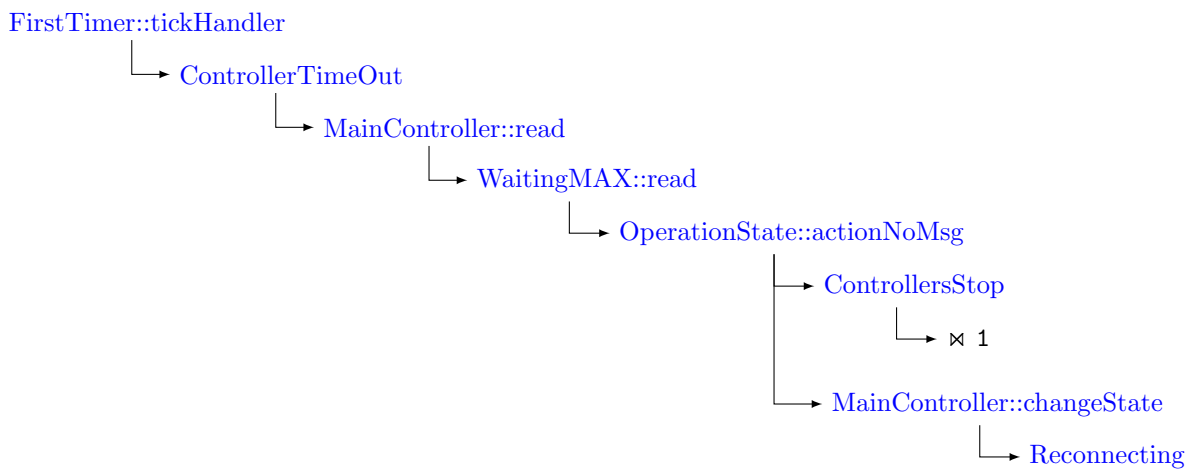
## 2.35. Requerimiento RF-33.

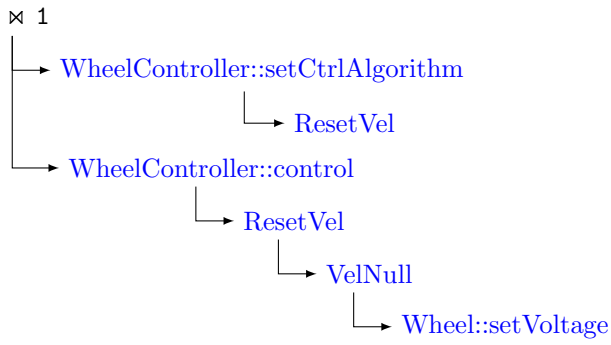


## 2.36. Requerimiento RF-34.

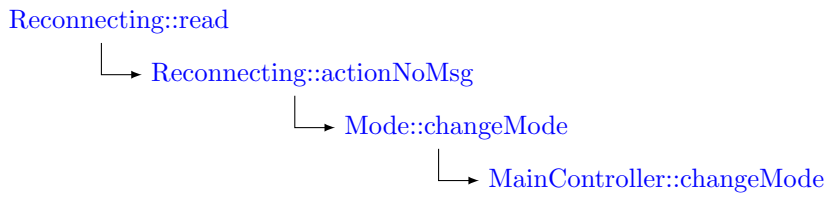
MCBuilder::buildOpStates

## 2.37. Requerimiento RF-35.

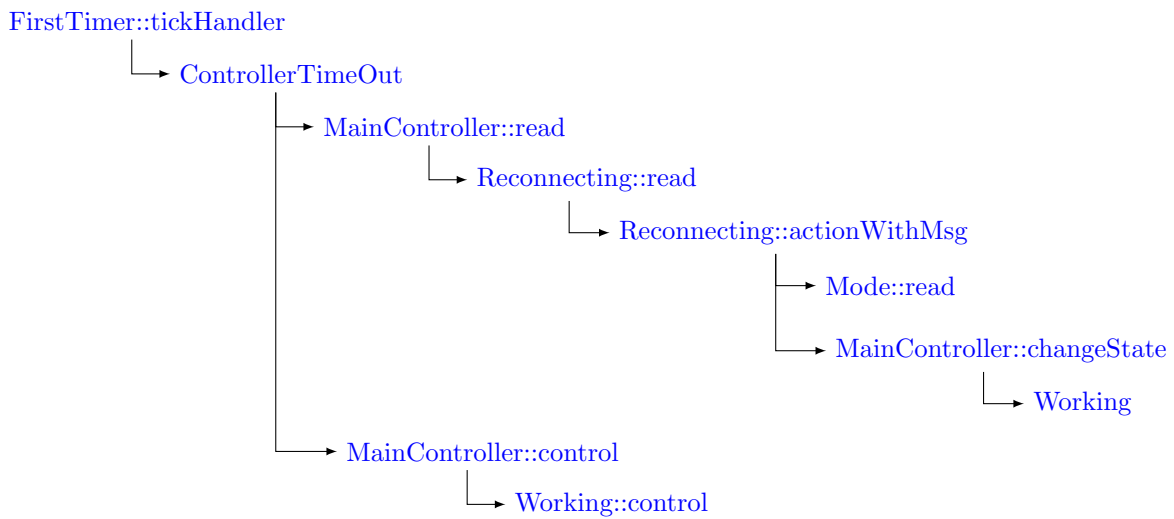




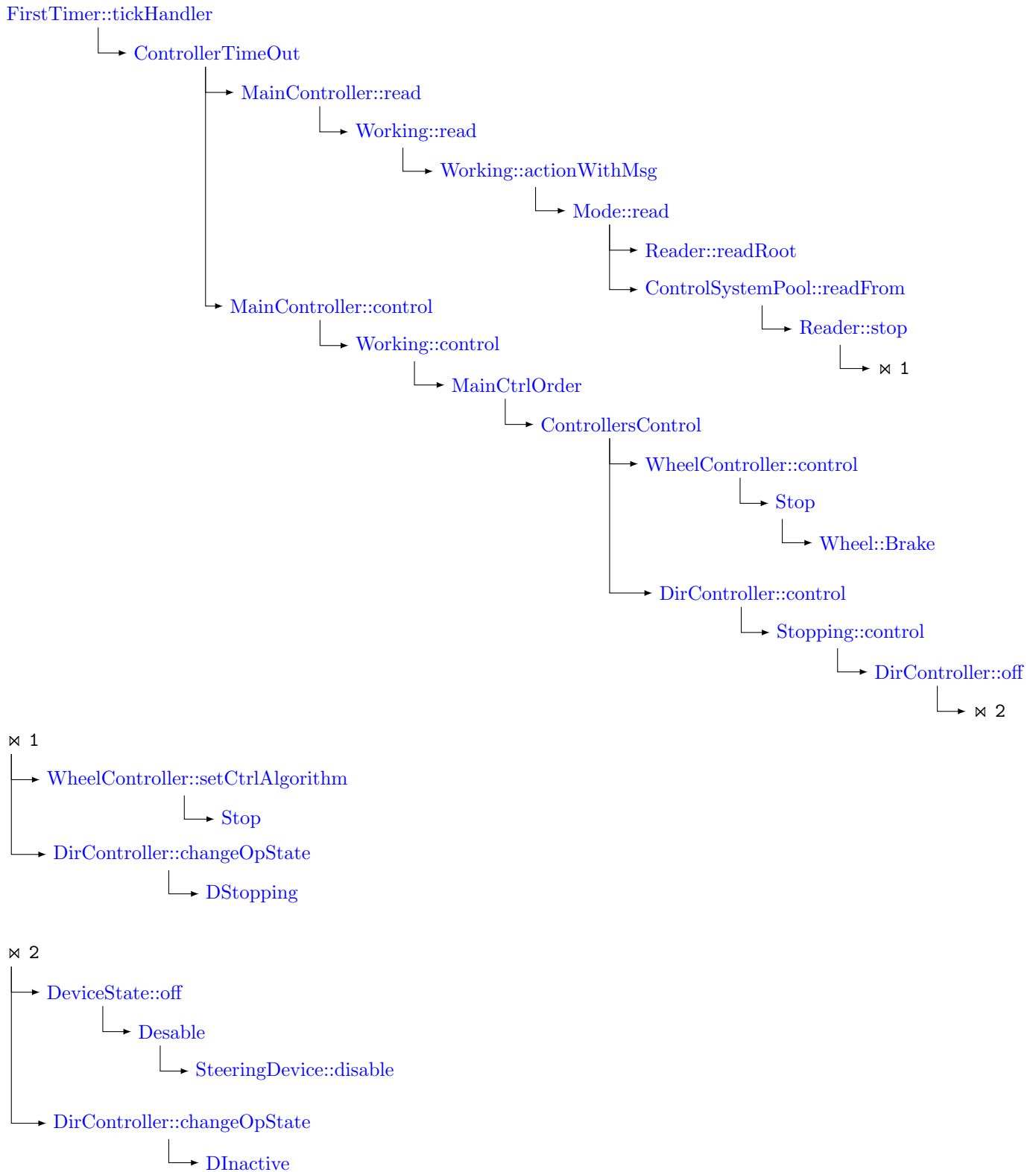
## 2.38. Requerimientos **RF-36.** – **RF-37.**



## 2.39. Requerimiento **RF-38.**



## 2.40. Requerimiento RF-39.



## 2.41. Requerimiento RF-40.

SignedPerc, SerialReader::readSetpointDir



## 2.42. Requerimiento RF-41.

RPM,  
SerialReader::readSetpointDD, SerialReader::readSetpointDI,  
SerialReader::readSetpointTD, SerialReader::readSetpointTI

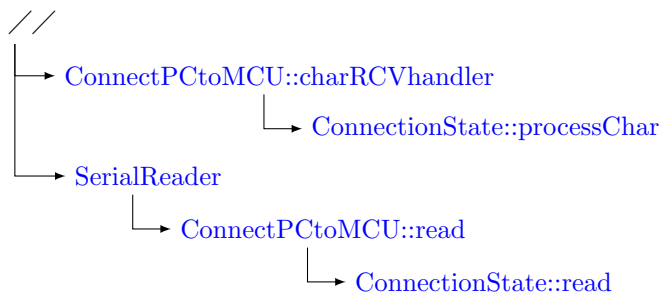
---

## 2.43. Requerimientos RF-42. – RF-43.

Percentage,  
SerialReader::readSetpointDD, SerialReader::readSetpointDI,  
SerialReader::readSetpointTD, SerialReader::readSetpointTI

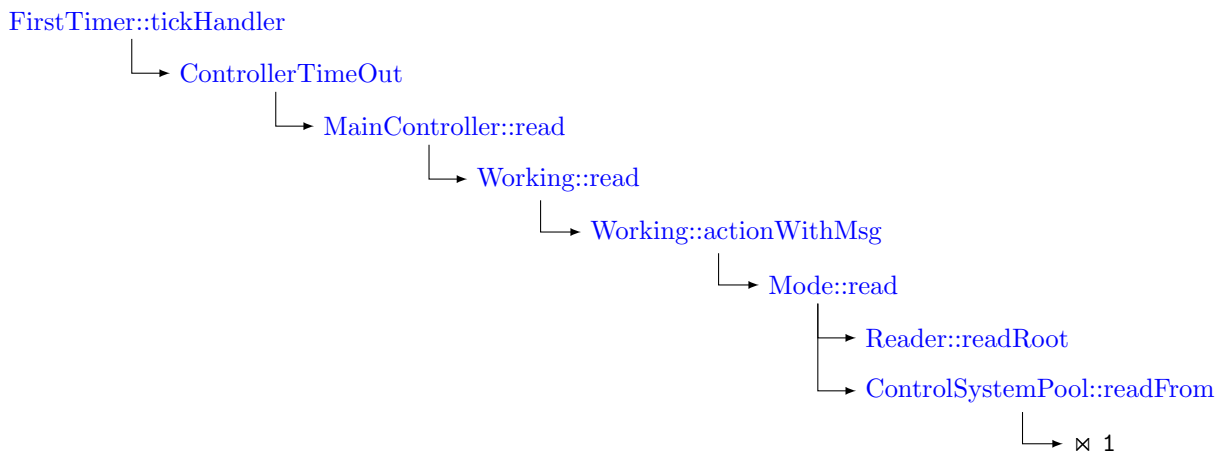
---

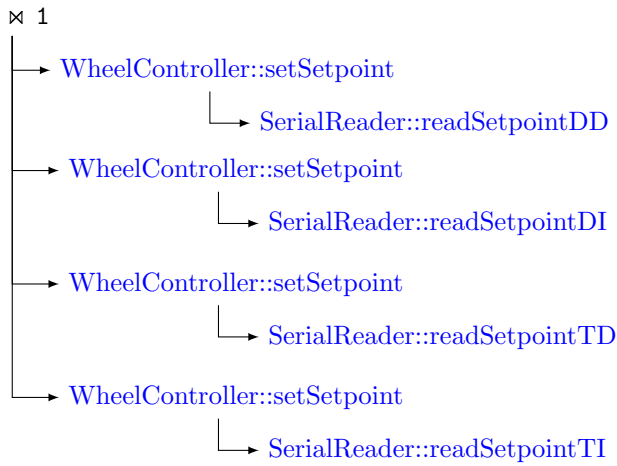
## 2.44. Requerimiento RF-44.



## 2.45. Requerimientos RF-45. – RF-46.

RPM, Percentage,  
SerialReader::readSetpointDD, SerialReader::readSetpointDI,  
SerialReader::readSetpointTD, SerialReader::readSetpointTI

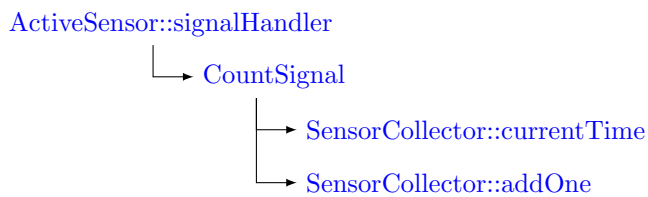




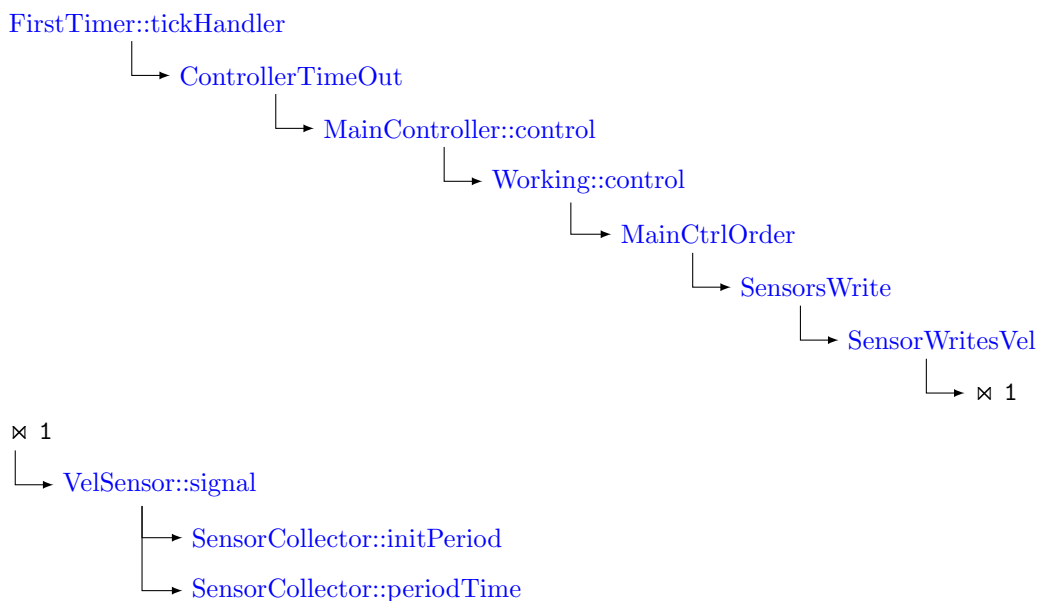
## 2.46. Requerimientos **RF-47.** – **RF-48.**



## 2.47. Requerimiento **RF-49.**



## 2.48. Requerimiento **RF-50.**

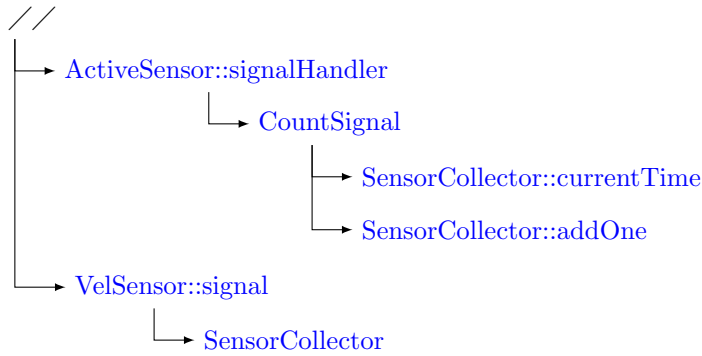


## 2.49. Requerimiento RF-51.

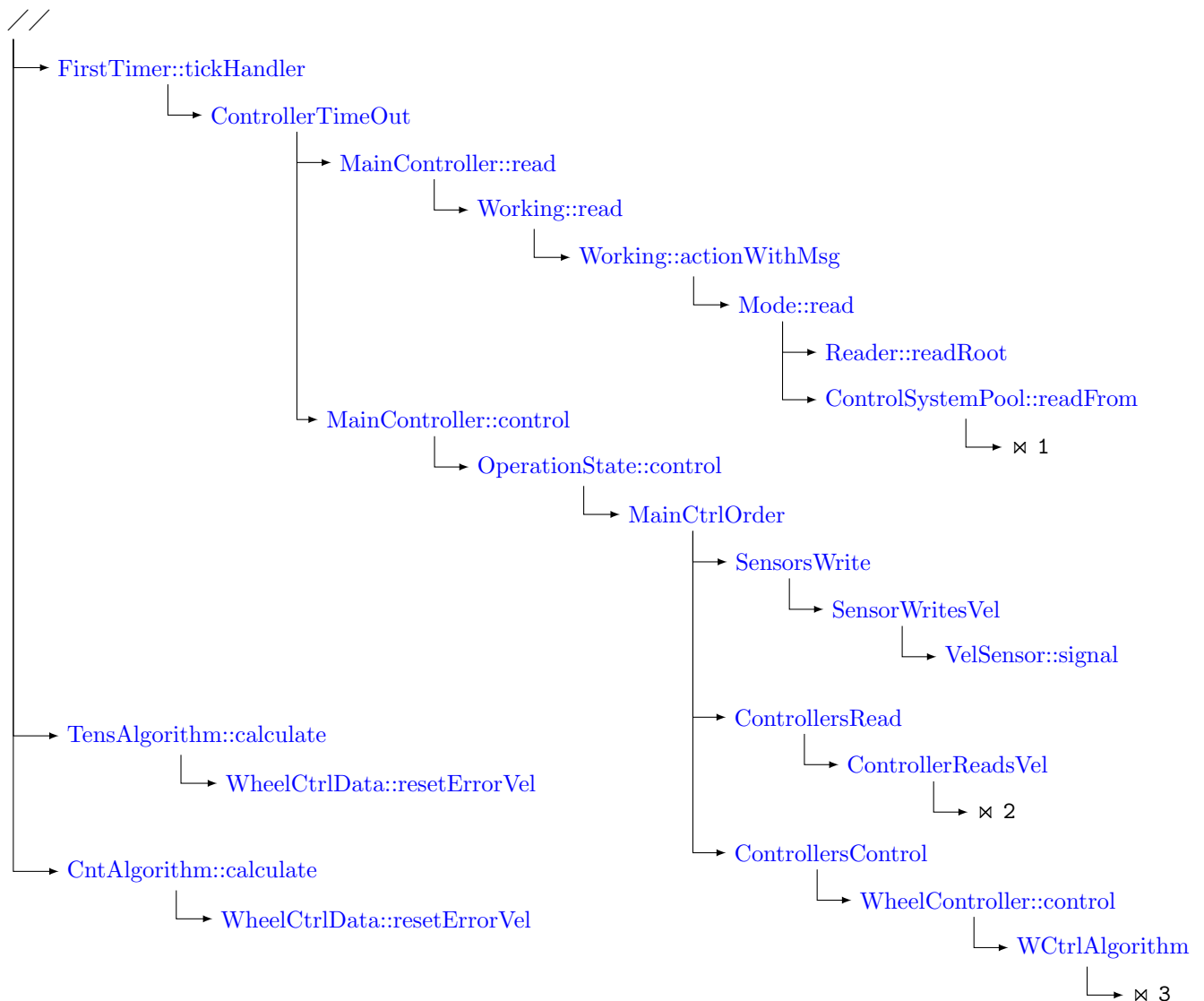
SensorCollector::initPeriod, SensorCollector::total

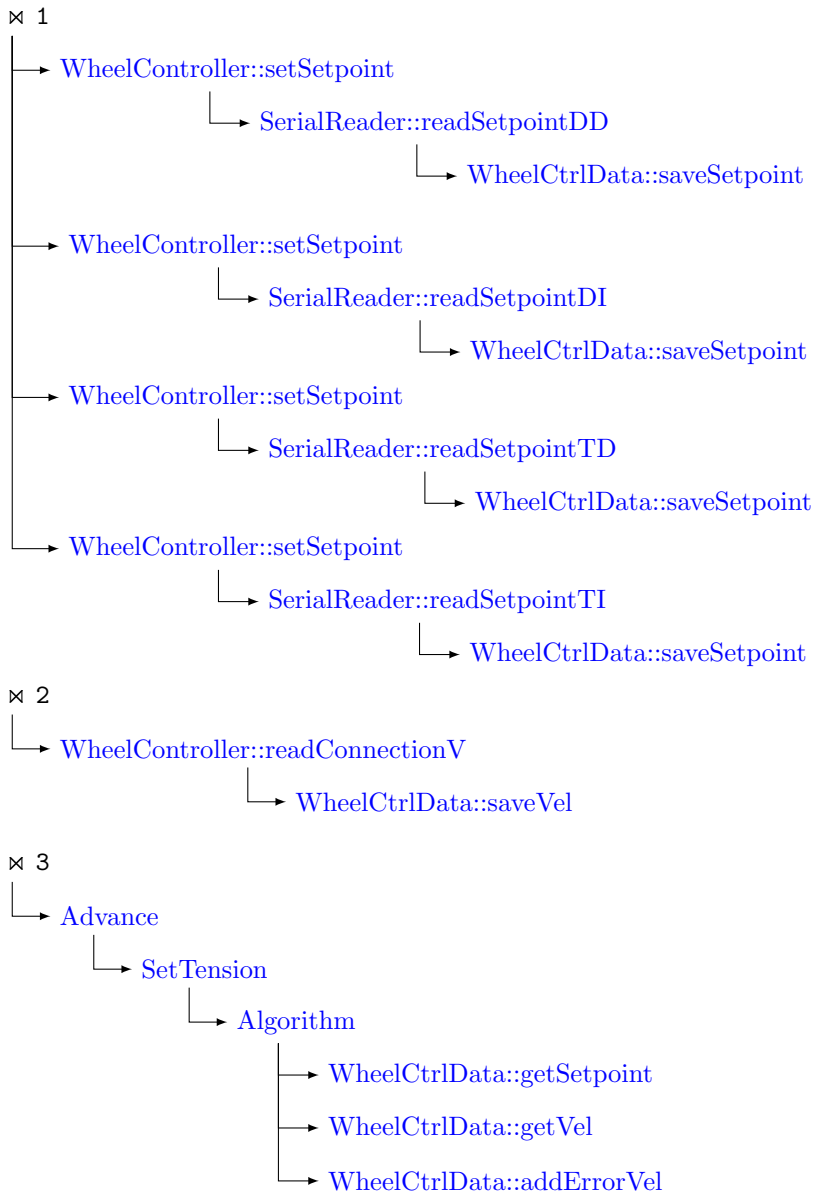
## 2.50. Requerimiento RF-52.

Constants::getDELTAT, VelSensor::VelSensor



## 2.51. Requerimientos RF-53. RF-54.





## 2.52. Requerimiento RF-55.

VelAlgorithm::VelAlgorithm

---

## 2.53. Requerimiento RF-56.

VelAlgorithm::calculate

---

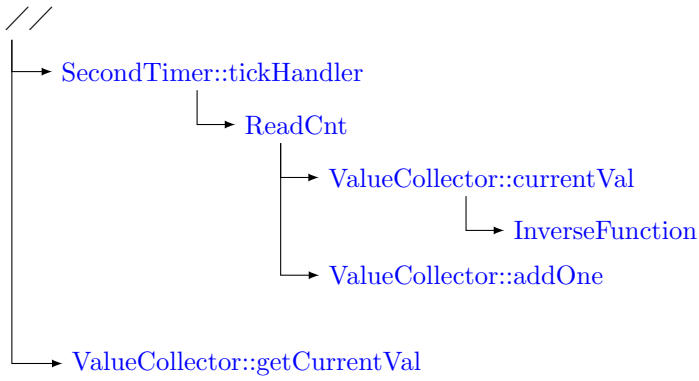
## 2.54. Requerimientos RF-57. – RF-58.

Main

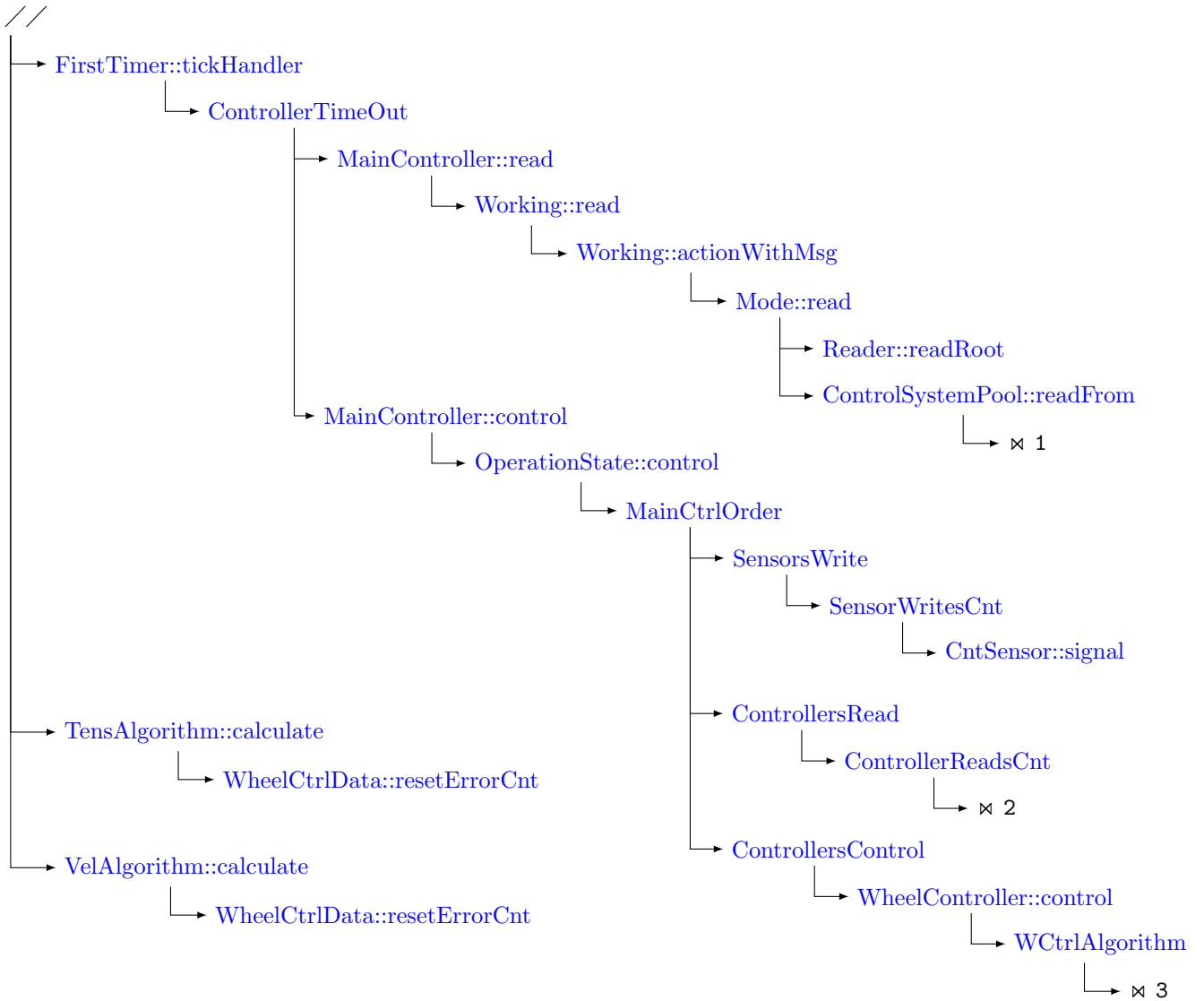
SecondTimer::setPeriod

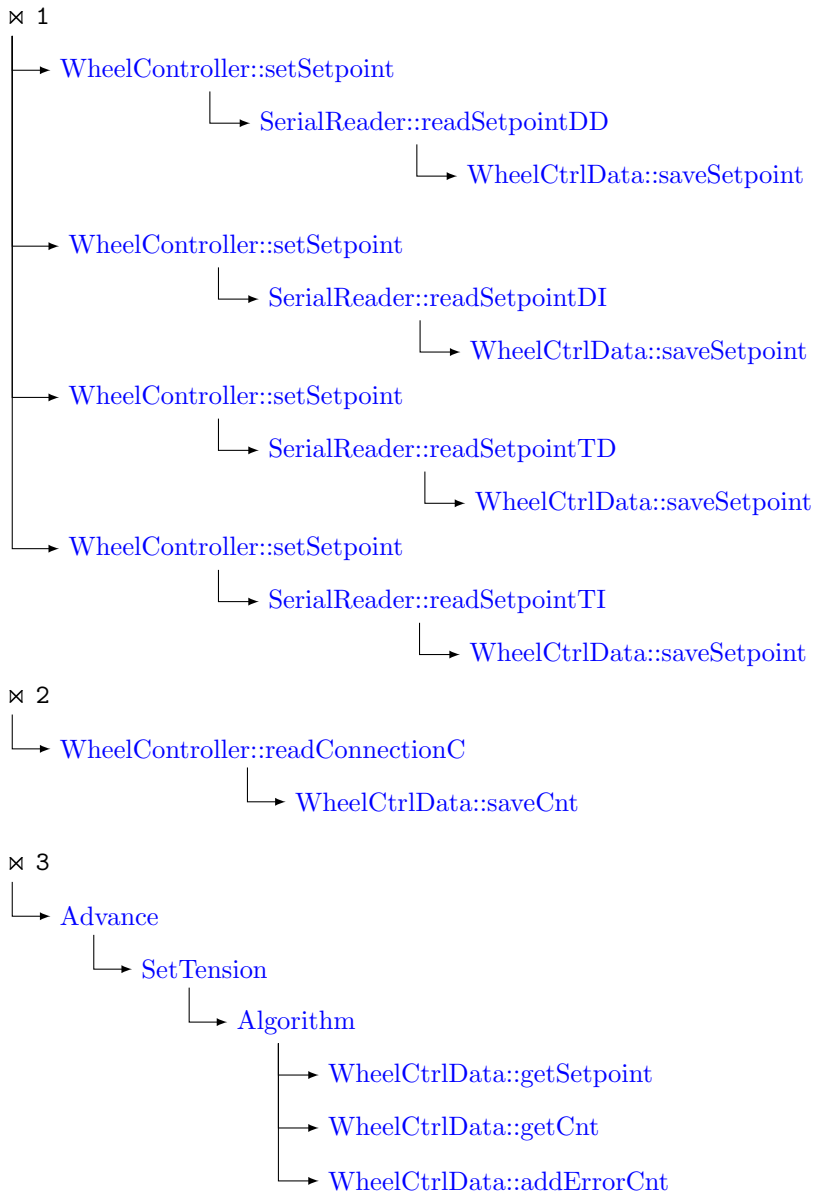
---

## 2.55. Requerimiento RF-59.



## 2.56. Requerimientos RF-60. RF-61.





## 2.57. Requerimiento **RF-62.**

CntAlgorithm::CntAlgorithm

---

## 2.58. Requerimiento **RF-63.**

CntAlgorithm::calculate

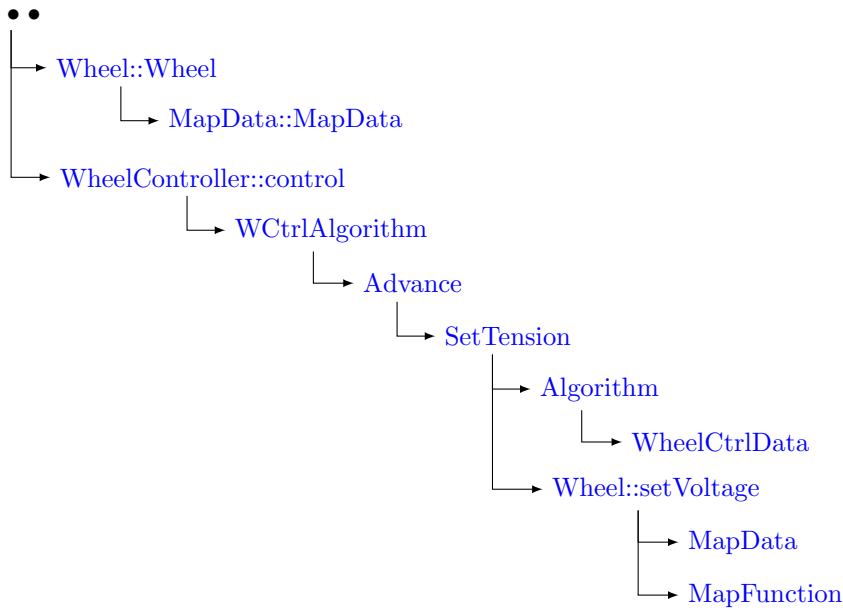
---

## 2.59. Requerimiento **RF-64.**

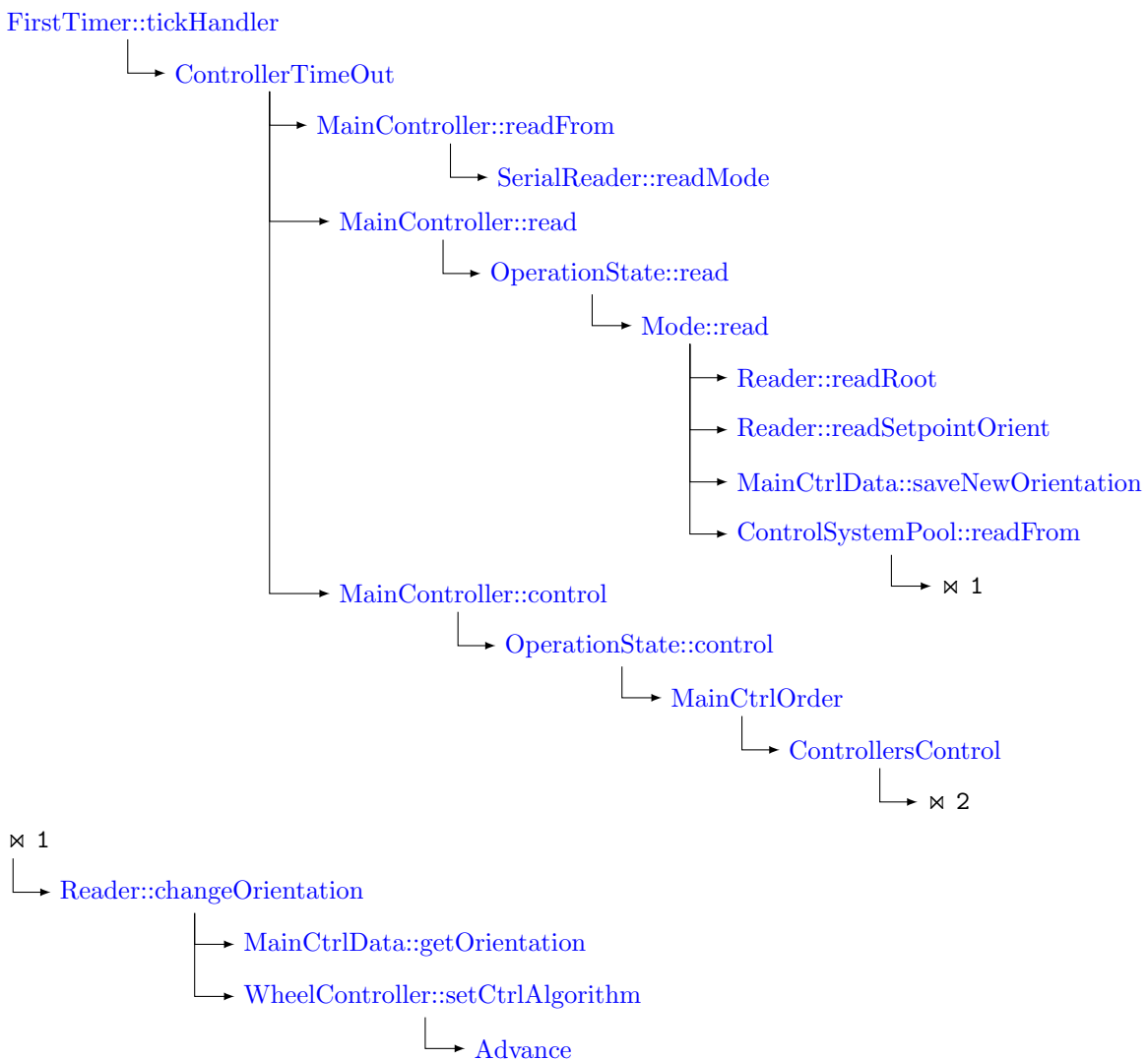
TensAlgorithm::calculate

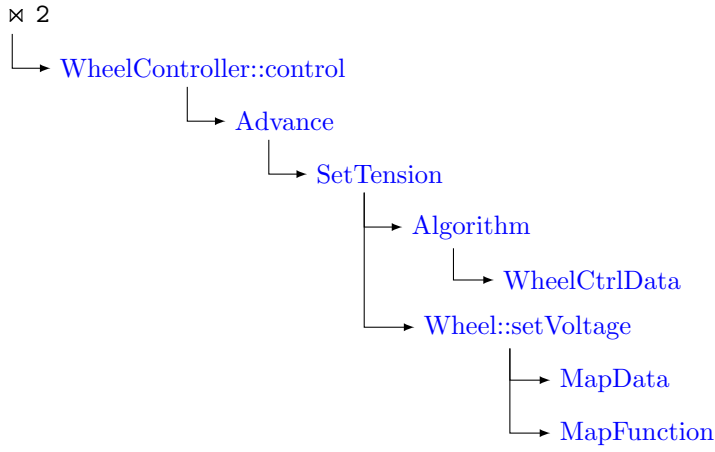
---

## 2.60. Requerimiento RF-65.

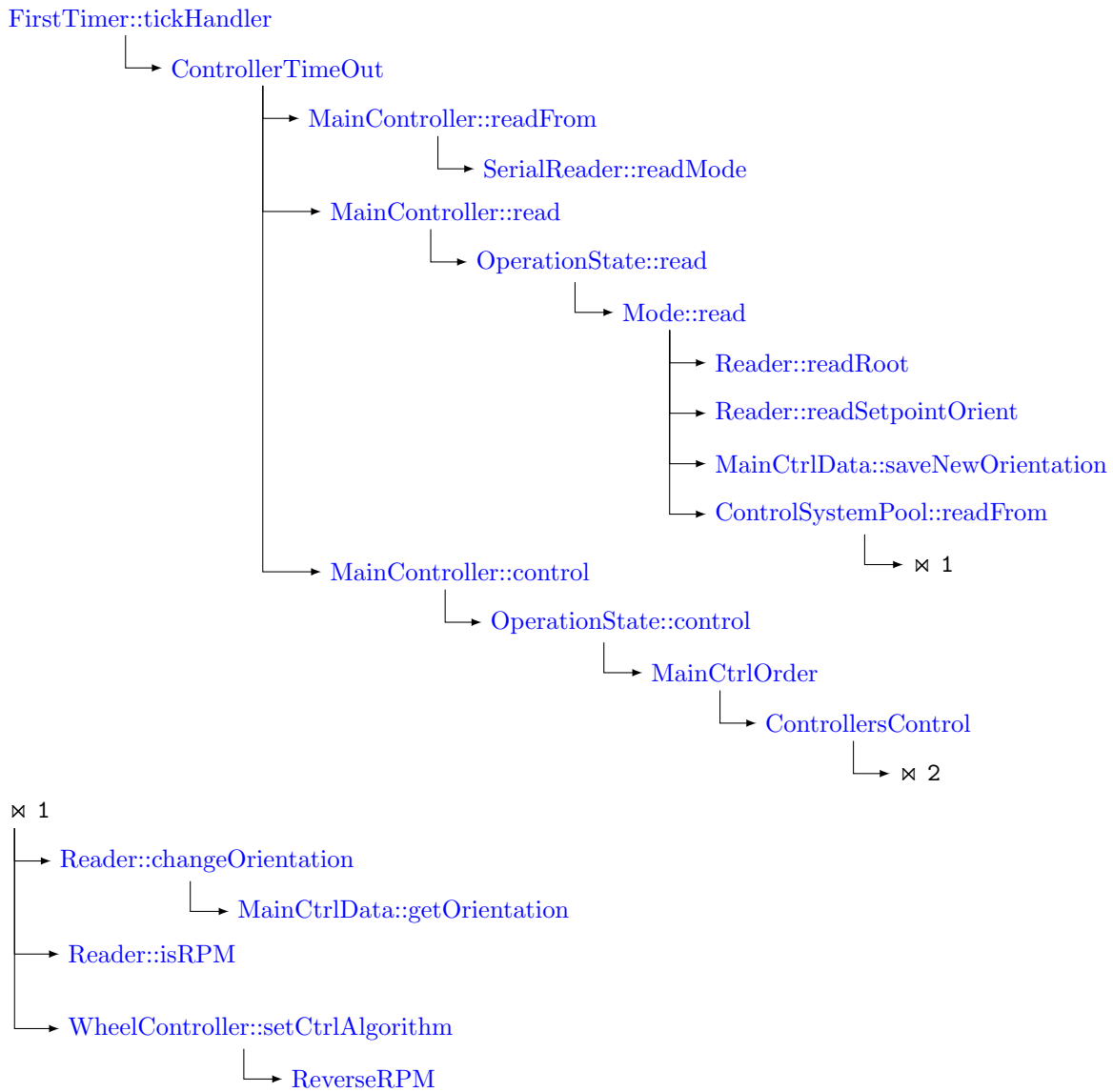


## 2.61. Requerimiento RF-66.

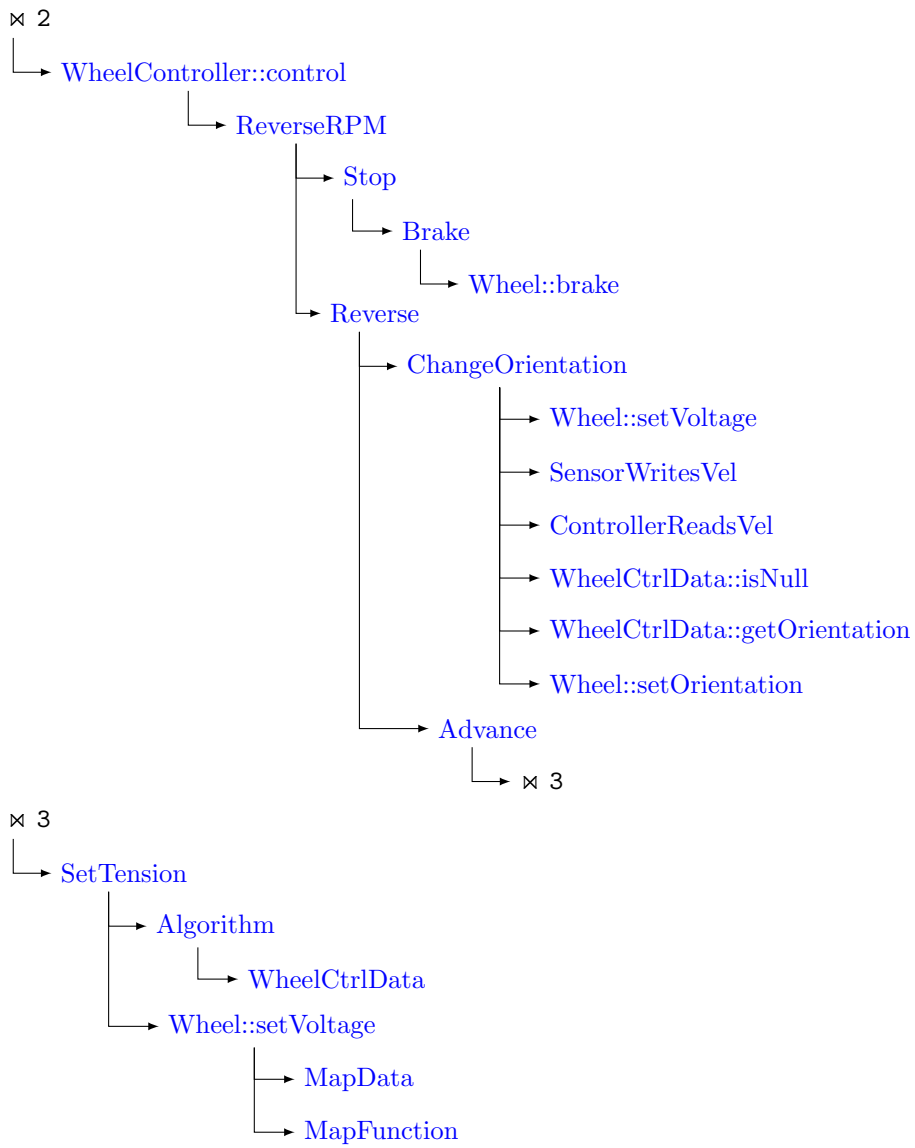




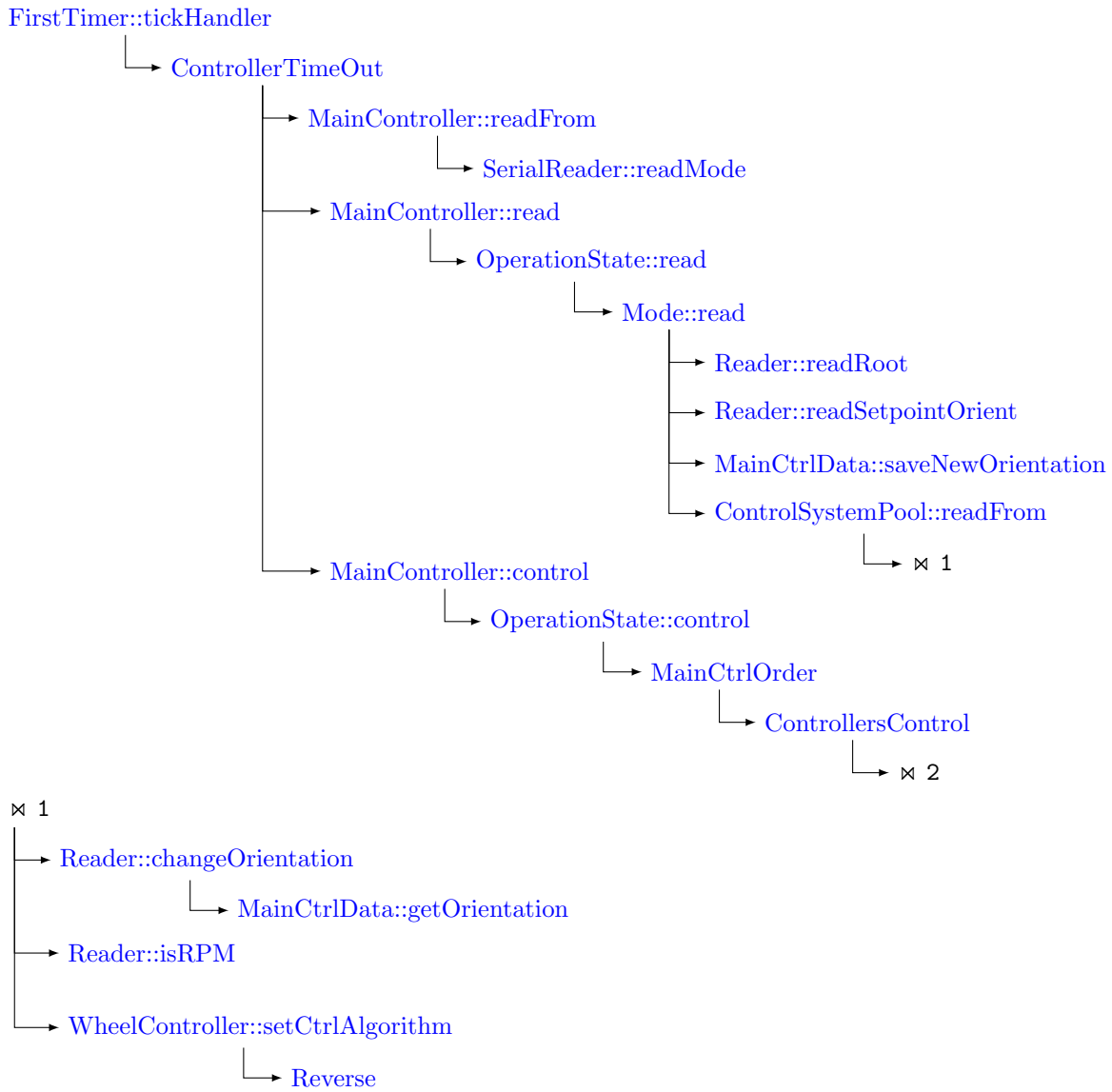
## 2.62. Requerimiento **RF-67**.

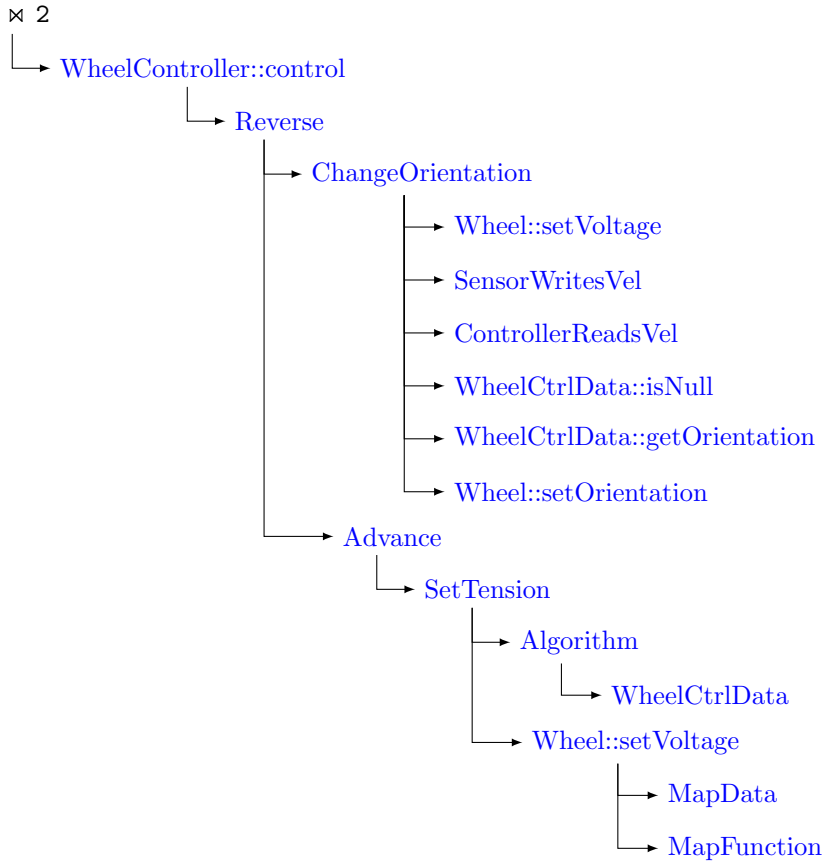






## 2.63. Requerimiento RF-68.

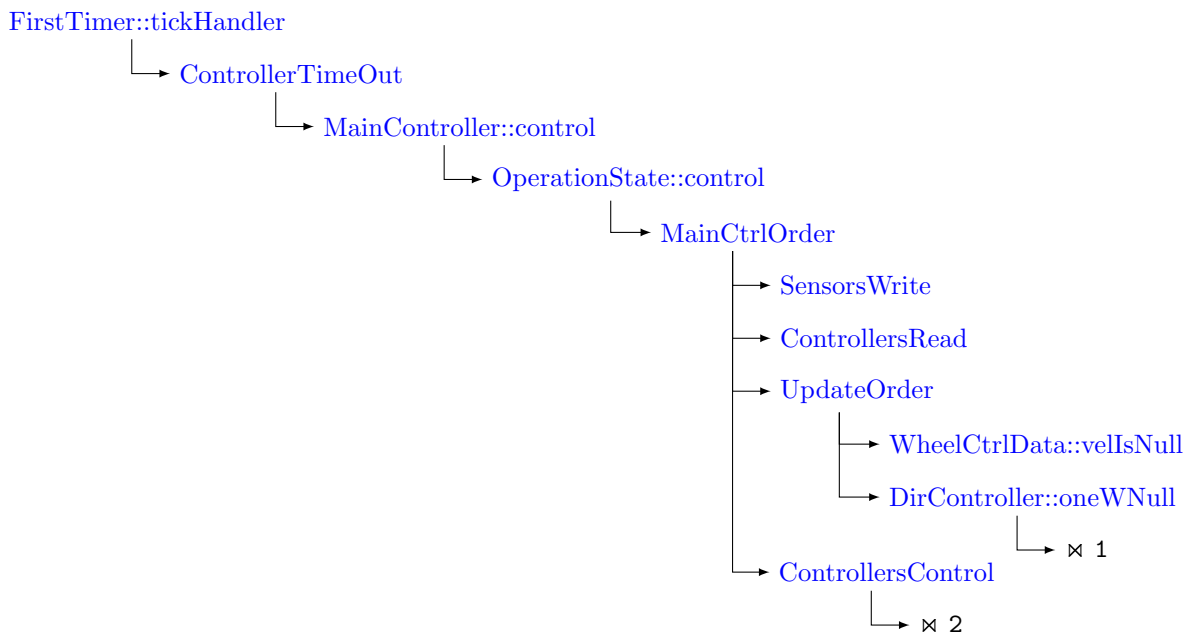


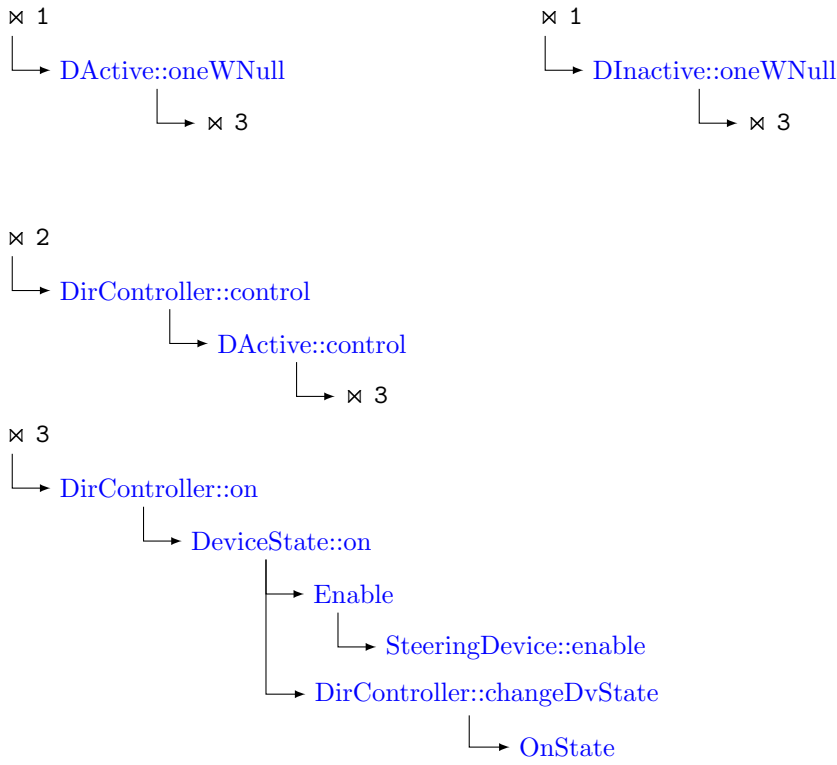


## 2.64. Requerimiento RF-69.

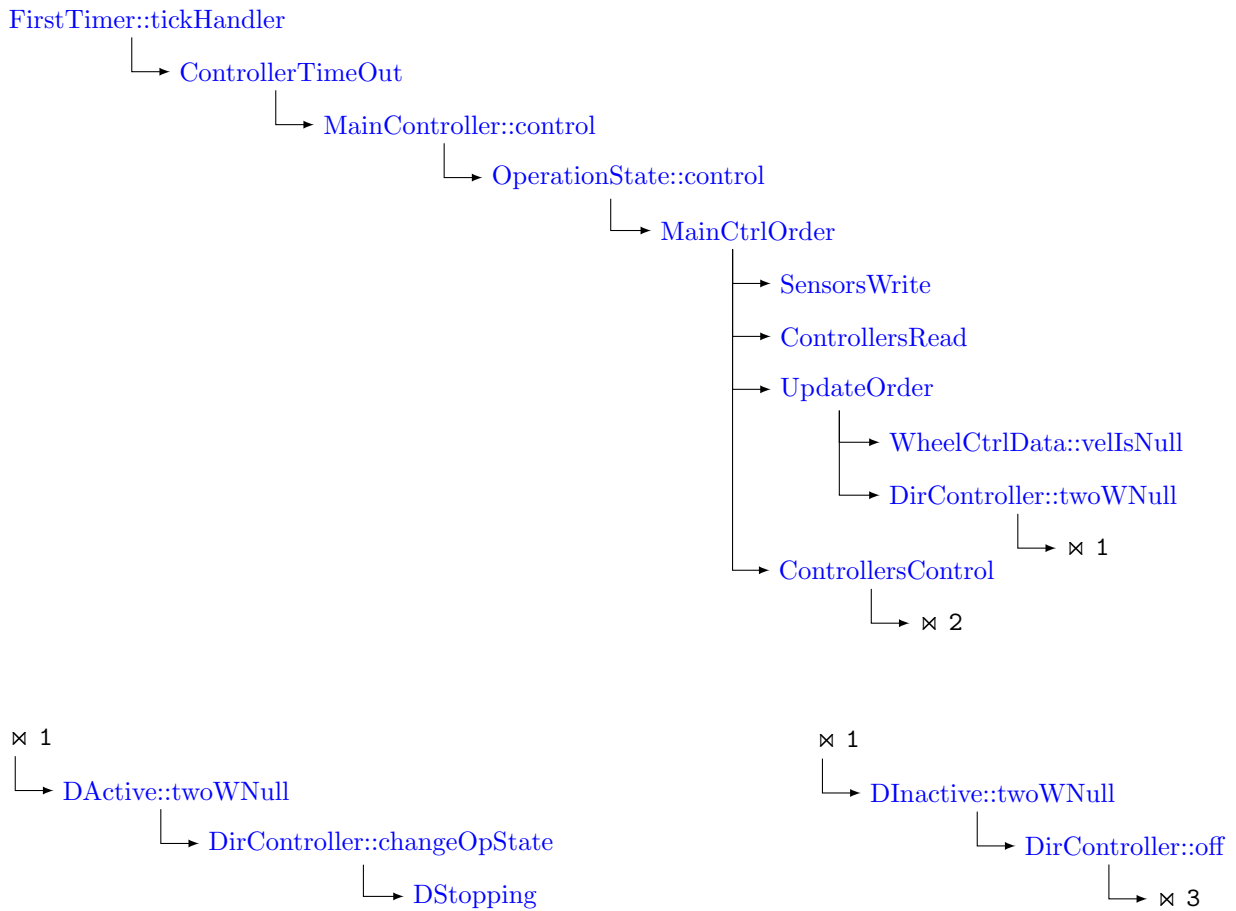
`DirSensor::DirSensor`, `DirSensor::signal`

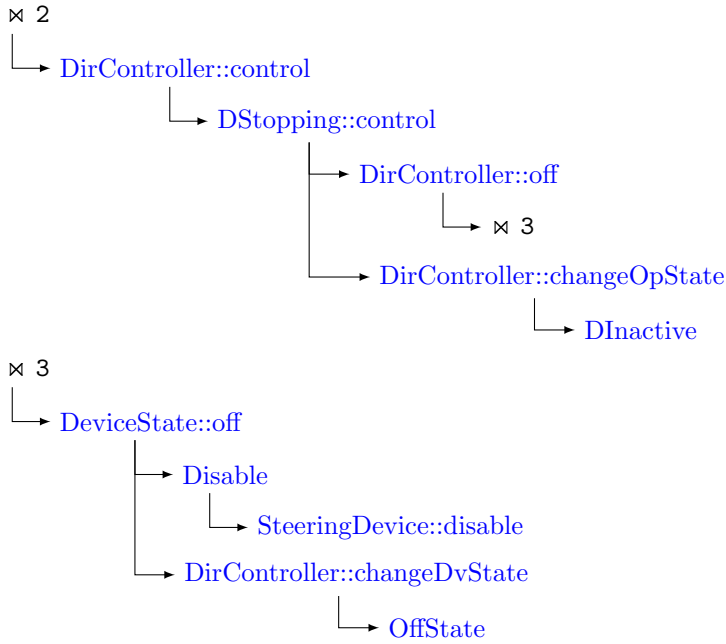
## 2.65. Requerimiento RF-70.



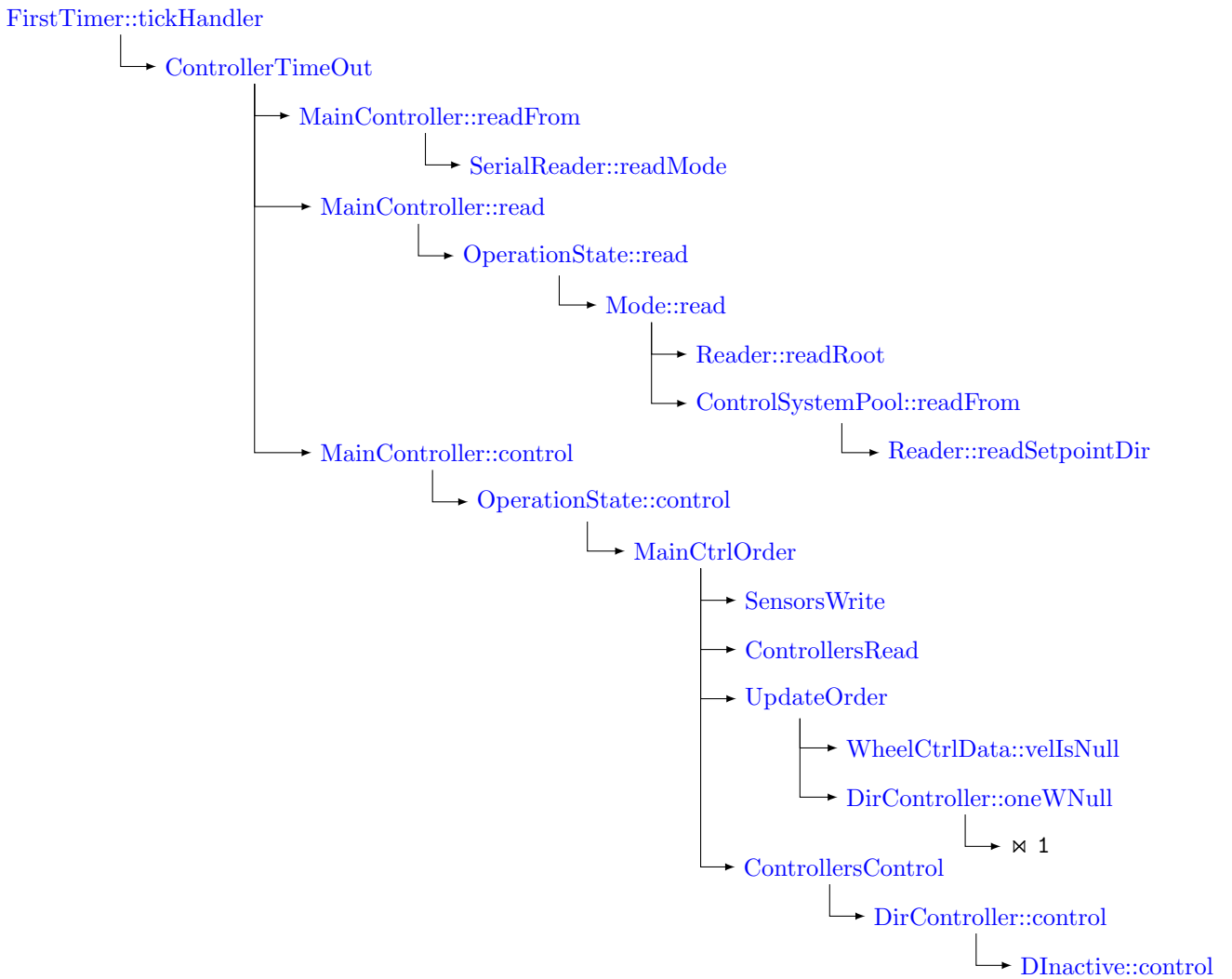


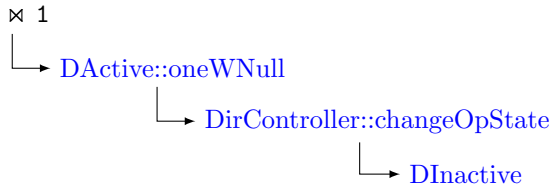
## 2.66. Requerimiento RF-71.



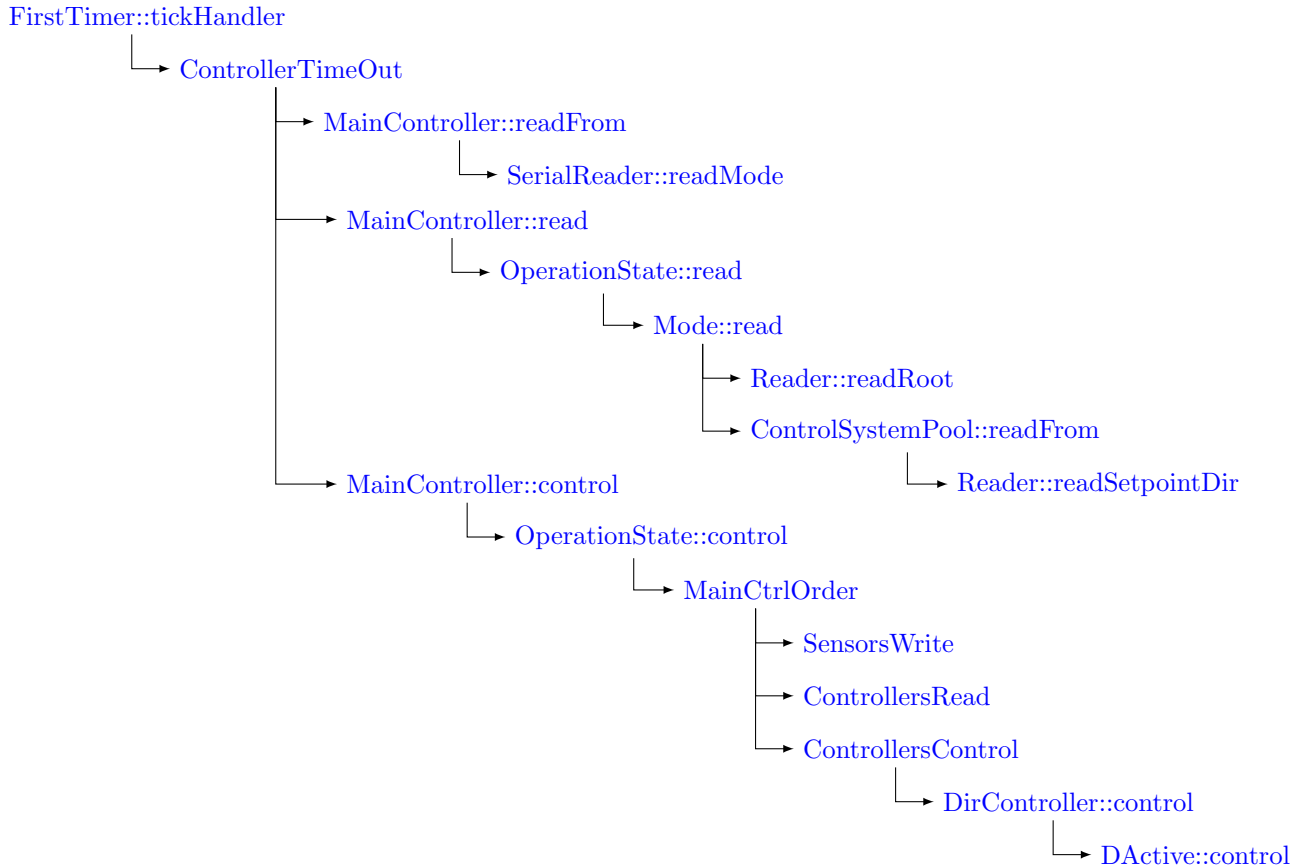


## 2.67. Requerimiento RF-72.





## 2.68. Requerimiento **RF-73**.

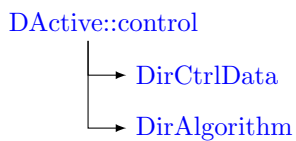


## 2.69. Requerimiento **RF-73.1**.

DActive::DActive, Constants::getM

---

## 2.70. Requerimiento **RF-73.2**.



## 2.71. Requerimiento **RF-73.3**.

DActive::control

---

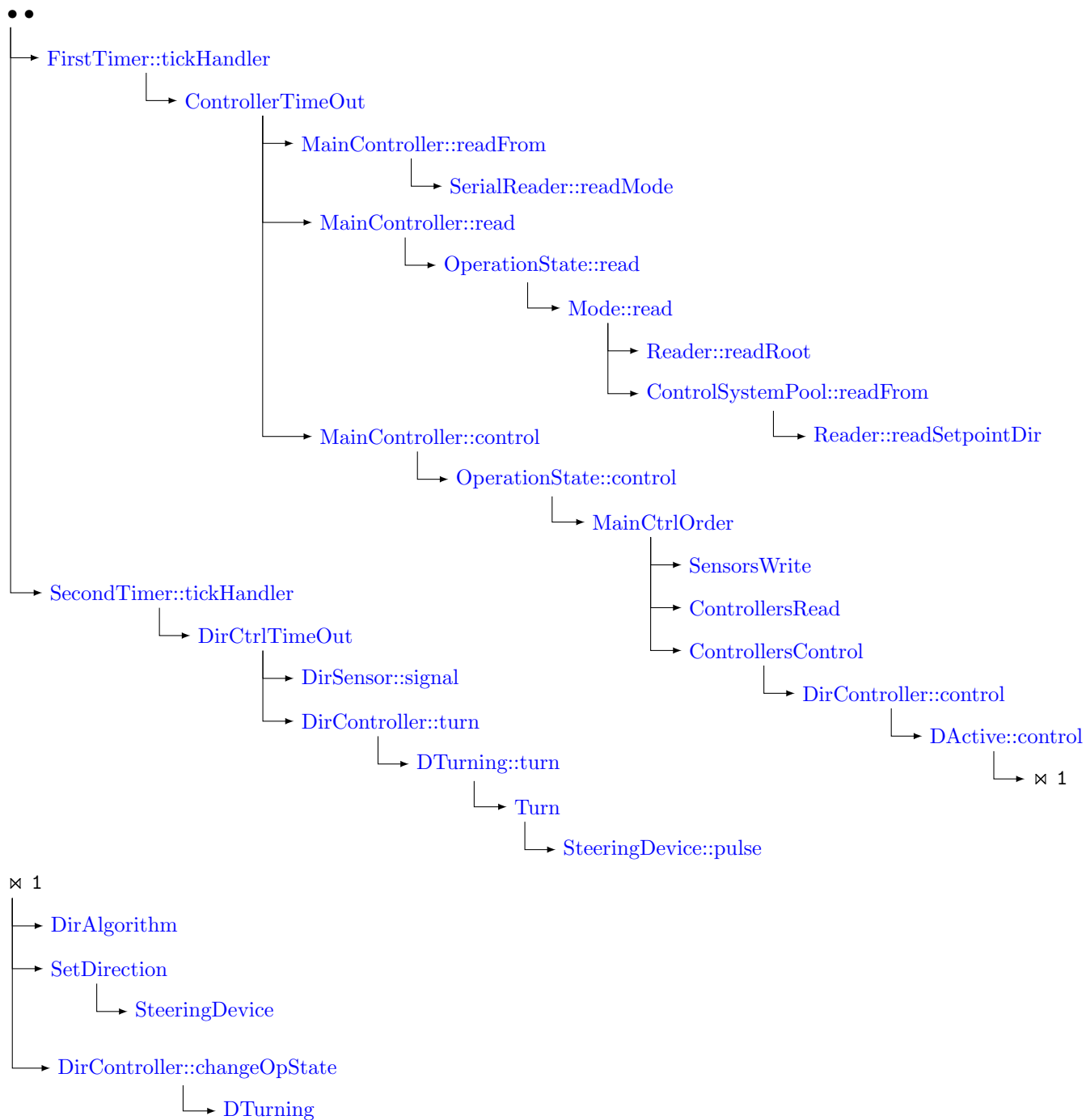
## 2.72. Requerimientos RF-73.4. – RF-73.5.

RF-73.4. – RF-73.5.

DActive::control

└─ DirAlgorithm

### 2.73. Requerimientos RF-74. – RF-75.



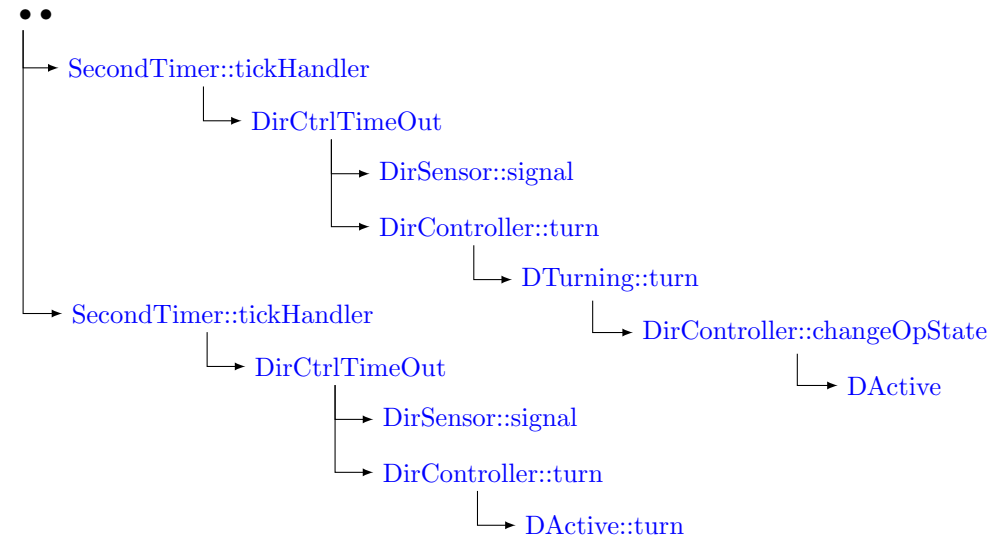
## 2.74. Requerimiento RF-75.1.

SteeringDevice::pulse

---

## 2.75. Requerimiento RF-75.2.

DTurning::DTurning



## 2.76. Requerimiento RF-76.

FirstTimer::tickHandler



## 2.77. Requerimiento RF-77.

SerialWriter

---



# Bibliografía

- [Pom22a] Laura Pomponio. Diseño del Sistema de Software del Microcontrlador del Robot Desmalezador. Versión 1.4.1. 2022.
- [Pom22b] Laura Pomponio. Requerimientos Funcionales de Software del Microcontrolador del Robot Desmalezador. Versión 1.2.1. 2022.