Ev3Dev

0.1.1

Generated by Doxygen 1.8.11

Contents

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 4

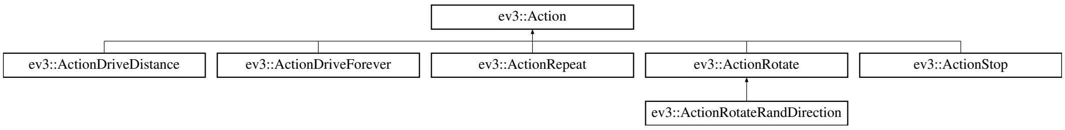
Class Documentation

4.1 ev3::Action Class Reference

Base class for all Action controlling classes.

`#include <Action.h>`

Inheritance diagram for ev3::Action:



Public Types

- enum ActionType {
  NOP, REPEAT, DRIVE_DISTANCE, ROTATE,
  ROTATE_RANDOM_DIR, STOP, DRIVE_FOREVER }

  Type of Action.
- typedef std::function< bool(void) > EndCondition

  Type for lambda functions to store end of Action condition.

Public Member Functions

- Action (CommandsVector commands, ActionType type)

  Constructor with CommandsVector and ActionType parameters.
- Action (CommandsVector commands)

  Constructor with CommandsVector parameter.
- Action (ActionType type)

  Constructor with ActionType parameter.
- virtual ∼Action ()

  Default destructor.
- virtual void execute ()

  Executes stored Commands in a sequence.

- virtual bool isFinished ()

  Check if Action condition is fullfilled.
- virtual bool isExecuted ()

  Check if action was executed.
- virtual std::string getActionPrototype ()

  Generate std::string prototype for Action.
- virtual std::string getString ()

  Get human-readable Action name.
- void setCommands (CommandsVector commands)

  Set Commands to be executed.
- void setEndCondition (EndCondition condition)

  Set end condition for Action.
- ActionType getType ()

  Get current Action type.

Static Public Attributes

- static const std::string EMPTY_PROTO

  String for empty Action prototype.

Protected Attributes

- ActionType _type

  Action type.
- CommandsVector _commands

  Vector of Commands.
- EndCondition _endCondition

  Lambda function defining Action end condition.
- bool **_isExecuted** = false

4.1.1 Detailed Description

Base class for all Action controlling classes.

Each Action contains of a sequence of many Commands and all of them are executed immediately, one after another. Action is valid, until specific Event occurs or its endCondition function returns true.

Action objects are instantiated accordingly to Robot model that uses them. Actions are predefined and cannot be dynamically created.

4.1.2 Member Enumeration Documentation

4.1.2.1 enum ev3::Action::ActionType

Type of Action.

It directly points to derived class being used.

See also

Robot::AvailableActions

Enumerator

NOP No operation.

REPEAT Repeats execution of other Actions.

DRIVE_DISTANCE Power Motor to reach certain distance.

ROTATE Rotate Robot for given angle.

ROTATE_RANDOM_DIR Rotate for given angle, clockwise or counterclockwise at random.

STOP Stop all active motors.

DRIVE_FOREVER Drive forward or backward infinetely.

4.1.3 Constructor & Destructor Documentation

4.1.3.1 Action::Action (CommandsVector *commands*, ActionType *type*)

Constructor with CommandsVector and ActionType parameters.

Parameters

commands	Commands stored within this Action.
type	Type of Action used.

4.1.3.2 Action::Action (CommandsVector *commands*)

Constructor with CommandsVector parameter.

Action type is set to Action::NOP .

Parameters

commands	Commands stored within this Action.

4.1.3.3 Action::Action (ActionType *type*)

Constructor with ActionType parameter.

Parameters

type	Type of Action used.

4.1.4 Member Function Documentation

4.1.4.1 std::string Action::getActionPrototype ( ) [virtual]

Generate std::string prototype for Action.

Returns

Encoded Action data into std::string.

Reimplemented in ev3::ActionDriveForever, ev3::ActionStop, ev3::ActionRotateRandDirection, ev3::ActionRotate, and ev3::ActionDriveDistance.

4.1.4.2 std::string Action::getString ( ) [virtual]

Get human-readable Action name.

Returns

String containing Action name.

Reimplemented in ev3::ActionDriveForever, ev3::ActionStop, ev3::ActionRotateRandDirection, ev3::ActionRotate, ev3::ActionDriveDistance, and ev3::ActionRepeat.

4.1.4.3 Action::ActionType Action::getType ( )

Get current Action type.

Returns

ActionType value.

4.1.4.4 bool Action::isExecuted ( ) [virtual]

Check if action was executed.

Returns

True if actcion was already executed, false otherwise.

4.1.4.5 bool Action::isFinished ( ) [virtual]

Check if Action condition is fullfilled.

Returns

Value returned from Action::_endCondition.

4.1.4.6 void Action::setCommands (CommandsVector *commands*)

Set Commands to be executed.

Parameters

commands	CommandsVector with pointers to commands.

4.1.4.7 void Action::setEndCondition (EndCondition *condition*)

Set end condition for Action.

Parameters

condition	Lambda function returning bool value.

4.1.5 Member Data Documentation

4.1.5.1 EndCondition ev3::Action::_endCondition [protected]

Initial value:

```
= []()
        {
            return true;
        }
```

Lambda function defining Action end condition.

The documentation for this class was generated from the following files:

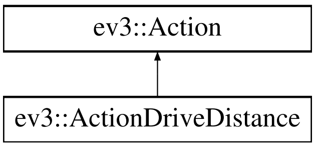
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Action.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Action.cpp

4.2 ev3::ActionDriveDistance Class Reference

Implements Robot simple task to drive straight for a given distance.

```
#include <Action.h>
```

Inheritance diagram for ev3::ActionDriveDistance:



Public Member Functions

- ActionDriveDistance (int distance)

  Constructor with distance parameter.
- ActionDriveDistance (CommandsVector commands, int distance)

  Constructor with CommandsVector and distance parameters.
- int getDistance ()

  Get distance the Robot has to drive.
- virtual std::string getActionPrototype ()

  Get ActionDriveDistance encoded name and its parameters.
- virtual std::string getString () override

  Get ActionDriveDistance human-readable name.

Additional Inherited Members

4.2.1 Detailed Description

Implements Robot simple task to drive straight for a given distance.

4.2.2 Constructor & Destructor Documentation

4.2.2.1 ActionDriveDistance::ActionDriveDistance (int *distance*)

Constructor with distance parameter.

Parameters

distance	Integer value in Robot units to be driven.

4.2.2.2 ActionDriveDistance::ActionDriveDistance (CommandsVector *commands,* int *distance*)

Constructor with CommandsVector and distance parameters.

Parameters

commands	Sequence of commands to be executed.
distance	Integer value in Robot units to be driven.

4.2.3 Member Function Documentation

4.2.3.1 std::string ActionDriveDistance::getActionPrototype ( ) [virtual]

Get ActionDriveDistance encoded name and its parameters.

Returns

String with encoded name and parameters.

Reimplemented from ev3::Action.

4.2.3.2 int ActionDriveDistance::getDistance ( )

Get distance the Robot has to drive.

Returns

Integer value in Robot units.

4.2.3.3 std::string ActionDriveDistance::getString ( ) [override], [virtual]

Get ActionDriveDistance human-readable name.

Returns

String with name and parameters

Reimplemented from ev3::Action.

The documentation for this class was generated from the following files:

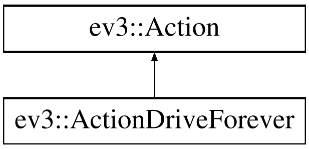
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Action.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Action.cpp

4.3 ev3::ActionDriveForever Class Reference

Implements Robot simple task to drive straight forever.

```
#include <Action.h>
```

Inheritance diagram for ev3::ActionDriveForever:



Public Member Functions

- ActionDriveForever (bool forward=true)

  Constructor with direction parameter.
- ActionDriveForever (CommandsVector commands, bool forward=true)

  Constructor with CommandsVector and direction parameter.
- virtual std::string getActionPrototype ()

  Get ActionDriveForever encoded name and its parameters.
- virtual std::string getString () override

  Get ActionDriveForever human-readable name.
- bool isForward ()

  Return specified direction.

Additional Inherited Members

4.3.1 Detailed Description

Implements Robot simple task to drive straight forever.

4.3.2 Constructor & Destructor Documentation

4.3.2.1 ActionDriveForever::ActionDriveForever (bool *forward* = `true`)

Constructor with direction parameter.

Parameters

forward	True to drive forward, false otherwise.

4.3.2.2 ActionDriveForever::ActionDriveForever (**CommandsVector** *commands*, bool *forward* = `true`)

Constructor with CommandsVector and direction parameter.

Parameters

forward	True to drive forward, false otherwise.

4.3.3 Member Function Documentation

4.3.3.1 std::string ActionDriveForever::getActionPrototype ( ) `[virtual]`

Get ActionDriveForever encoded name and its parameters.

Returns

String with encoded name and parameters.

Reimplemented from ev3::Action.

4.3.3.2 std::string ActionDriveForever::getString ( ) [override], [virtual]

Get ActionDriveForever human-readable name.

Returns

String with name and parameters

Reimplemented from ev3::Action.

4.3.3.3 bool ActionDriveForever::isForward ( )

Return specified direction.

Returns

True for forward, false for backward.

The documentation for this class was generated from the following files:

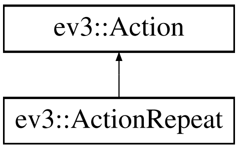
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Action.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Action.cpp

4.4 ev3::ActionRepeat Class Reference

Stores many Actions in a vector and executes them in loop.

```
#include <Action.h>
```

Inheritance diagram for ev3::ActionRepeat:



Public Member Functions

- ActionRepeat (StoredActions actions, unsigned int n)
  Constructor with StoredActions and iterations parameters.
- virtual void execute ()
  Continue with executing stored Actions.
- virtual std::string getString ()
  Return human-readable ActionRepeat name.

Additional Inherited Members

4.4.1 Detailed Description

Stores many Actions in a vector and executes them in loop.

Number of iterations is given and may be infinite.

4.4.2 Constructor & Destructor Documentation

4.4.2.1 ActionRepeat::ActionRepeat (StoredActions *actions*, unsigned int *n*)

Constructor with StoredActions and iterations parameters.

Parameters

actions	Vector of Actions to be executed in a loop.
n	Number of iterations. If 0 is given, loop will be infinite.

4.4.3 Member Function Documentation

4.4.3.1 std::string ActionRepeat::getString ( ) `[virtual]`

Return human-readable ActionRepeat name.

Returns

String containing Action name.

Reimplemented from ev3::Action.

The documentation for this class was generated from the following files:

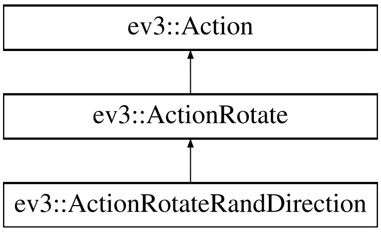
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Action.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Action.cpp

4.5 ev3::ActionRotate Class Reference

Implements Robot simple task to rotate a given angle, while not driving.

`#include <Action.h>`

Inheritance diagram for ev3::ActionRotate:



Public Member Functions

- ActionRotate (int rotation)

  Constructor with rotation parameter in degrees.
- ActionRotate (CommandsVector commands, int rotation)

  Constructor with CommandsVector and rotation parameters.
- int getRotation ()

  Get Robot rotation.
- virtual std::string getActionPrototype ()

  Get ActionRotate encoded name and its parameters.
- virtual std::string getString () override

  Get ActionRotate human-readable name.

Protected Attributes

- int _rotation

  Angle of rotation in degrees for the Robot.

Additional Inherited Members

4.5.1 Detailed Description

Implements Robot simple task to rotate a given angle, while not driving.

Rotation is made in place.

4.5.2 Constructor & Destructor Documentation

4.5.2.1 ActionRotate::ActionRotate (int *rotation*)

Constructor with rotation parameter in degrees.

Parameters

rotation	Number of degrees to rotate. Positive value rotates clockwise.

4.5.2.2 ActionRotate::ActionRotate (CommandsVector *commands,* int *rotation*)

Constructor with CommandsVector and rotation parameters.

Parameters

commands	Sequence of commands to be executed.
rotation	Integer value of Robot rotation in degrees.

4.5.3 Member Function Documentation

4.5.3.1 std::string ActionRotate::getActionPrototype ( ) [virtual]

Get ActionRotate encoded name and its parameters.

Returns

String with encoded name and parameters.

Reimplemented from ev3::Action.

Reimplemented in ev3::ActionRotateRandDirection.

4.5.3.2 int ActionRotate::getRotation ( )

Get Robot rotation.

Returns

Integer value of rotation in degrees.

4.5.3.3 std::string ActionRotate::getString ( ) [override], [virtual]

Get ActionRotate human-readable name.

Returns

String with name and parameters

Reimplemented from ev3::Action.

Reimplemented in ev3::ActionRotateRandDirection.

The documentation for this class was generated from the following files:

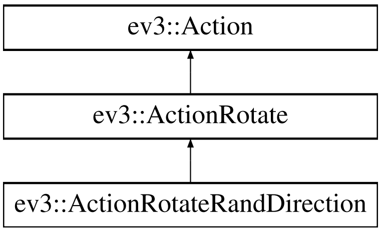
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Action.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Action.cpp

4.6 ev3::ActionRotateRandDirection Class Reference

Implements Robot simple task to rotate a random angle.

```
#include <Action.h>
```

Inheritance diagram for ev3::ActionRotateRandDirection:



Public Member Functions

- ActionRotateRandDirection (int rotation)

    Constructor with rotation parameter in degrees.
- ActionRotateRandDirection (CommandsVector commands, int rotation)

    Constructor with CommandsVector and rotation parameters.
- virtual std::string getActionPrototype ()

    Get ActionRotateRandDirection encoded name and its parameters.
- virtual std::string getString () override

    Get ActionRotateRandDirection human-readable name.
- virtual void execute () override

Additional Inherited Members

4.6.1 Detailed Description

Implements Robot simple task to rotate a random angle.

Rotation is performed in place. Maximum angle in degrees is passed via constructor argument.

4.6.2 Constructor & Destructor Documentation

4.6.2.1 ActionRotateRandDirection::ActionRotateRandDirection (int *rotation*)

Constructor with rotation parameter in degrees.

Parameters

rotation	Upper limit of degrees to rotate randomly. Positive value rotates clockwise.

4.6.2.2 ActionRotateRandDirection::ActionRotateRandDirection (CommandsVector *commands*, int *rotation*)

Constructor with CommandsVector and rotation parameters.

Parameters

commands	Sequence of commands to be executed.
rotation	Upper limit of degrees to rotate randomly. Positive value rotates clockwise.

4.6.3 Member Function Documentation

4.6.3.1 void ActionRotateRandDirection::execute ( ) [override], [virtual]

See also

Action::execute

Reimplemented from ev3::Action.

4.6.3.2 std::string ActionRotateRandDirection::getActionPrototype ( ) [virtual]

Get ActionRotateRandDirection encoded name and its parameters.

Returns

String with encoded name and parameters.

Reimplemented from ev3::ActionRotate.

4.6.3.3 std::string ActionRotateRandDirection::getString ( ) [override], [virtual]

Get ActionRotateRandDirection human-readable name.

Returns

String with name and parameters

Reimplemented from ev3::ActionRotate.

The documentation for this class was generated from the following files:

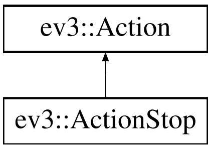
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Action.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Action.cpp

4.7 ev3::ActionStop Class Reference

Implements Robot simple task to stop all active motors.

```
#include <Action.h>
```

Inheritance diagram for ev3::ActionStop:



Public Member Functions

- ActionStop ()

  Default constructor.
- ActionStop (CommandsVector commands)

  Constructor with CommandsVector parameter.
- virtual std::string getActionPrototype ()

  Get ActionStop encoded name and its parameters.
- virtual std::string getString () override

  Get ActionStop human-readable name.

Additional Inherited Members

4.7.1 Detailed Description

Implements Robot simple task to stop all active motors.

4.7.2 Constructor & Destructor Documentation

4.7.2.1 ActionStop::ActionStop (CommandsVector *commands*)

Constructor with CommandsVector parameter.

Parameters

commands	Sequence of commands to be executed.

4.7.3 Member Function Documentation

4.7.3.1 std::string ActionStop::getActionPrototype ( ) `[virtual]`

Get ActionStop encoded name and its parameters.

Returns

String with encoded name and parameters.

Reimplemented from ev3::Action.

4.7.3.2 std::string ActionStop::getString ( ) `[override]`, `[virtual]`

Get ActionStop human-readable name.

Returns

String with name and parameters

Reimplemented from ev3::Action.

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Action.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Action.cpp

4.8 ev3::Agent Class Reference

Public Member Functions

- unsigned int **getId** ()
- void **setId** (const unsigned int id)
- unsigned int **getCommId** ()
- void **setCommId** (const unsigned int commId)
- void **processMessage** (Message ∗message, Message ∗retMessage)
- void **updateLastMessage** (Message ∗message)
- void **setBehaviour** (SharedPtrBehaviour behaviour)
- void **setMeasurement** (Measurements measurements)

The documentation for this class was generated from the following files:

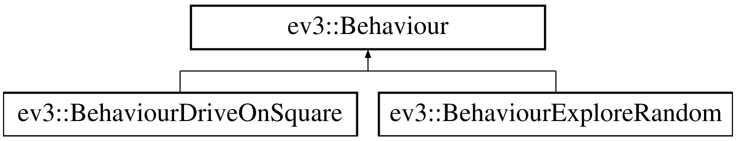
- /home/panda/Dokumenty/Repos/Ev3Dev/include/master/Agent.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/master/Agent.cpp

4.9 ev3::Behaviour Class Reference

Base class for all defined behaviours.

```
#include <Behaviour.h>
```

Inheritance diagram for ev3::Behaviour:



Public Types

- enum BehaviourType { CUSTOM, DRIVE_ON_SQUARE, EXPLORE_RANDOM }

  Type of Behaviour.

Public Member Functions

- Behaviour ()=default

  Default constructor.
- Behaviour (BehaviourType type, BehaviourStates states)

  Constructor with type and states vector parameters.
- Behaviour (BehaviourType type)

  Constructor with behaviour type.
- void setStates (BehaviourStates states)

  Available states setter.
- void setReactionStates (BehaviourStates reactionStates)

  Special reaction states which occur when event is fired.
- void setStopState (BehaviourState state)

  Special stop state, used mainly to get precise sensor measurements.
- void setMeasurements (Measurements measurements)

  Sensor which measurements will be required.
- virtual StringVector getPrototype ()

  Get Behaviour encoded name and its parameters.
- virtual std::string getString ()

  Get Behaviour human-readable name.
- virtual void process ()

  Updates behaviour in every iteration.
- void stop ()

  Stops Behaviour execution definetely.
- void pause ()

  Pauses Behaviour execution until it's resumed.
- void resume ()

  Resumes paused Behaviour.
- void start ()

  Starts Behaviour execution.
- void react (Event::EventType type)

  Performs special actions based on Event passed.

Static Public Member Functions

- static StringVector **getParameters** (StringVector proto)

Protected Attributes

- BehaviourType _type

  Type of Behaviour.
- BehaviourState _currentState

  Currently processed Behaviour.
- BehaviourState _stopState

  Special stop state for measurements and accurate action execution.
- BehaviourStates _states

  Vector with all Behaviour available states.
- BehaviourStates _reactionStates

  Vector with all reaction states, occuring after specific events.
- Measurements _measurements

  Vector of all Sensor ids that will be measured.
- bool _active = false

  Specified whether Behaviour is currently active or not.

4.9.1 Detailed Description

Base class for all defined behaviours.

It's responsible for maintaining active actions in a form of a state machine as well as keep track of sensors' measurements.

4.9.2 Member Enumeration Documentation

4.9.2.1 enum **ev3::Behaviour::BehaviourType**

Type of Behaviour.

Enumerator

CUSTOM Custom, user-defined behaviour.

DRIVE_ON_SQUARE Follow square-shaped route.

EXPLORE_RANDOM Drive in a direction and rotate randomly.

4.9.3 Constructor & Destructor Documentation

4.9.3.1 Behaviour::Behaviour (**BehaviourType** *type*, **BehaviourStates** *states*)

Constructor with type and states vector parameters.

Parameters

type	Behaviour type.
states	Vector of available Behaviour states.

4.9.3.2 Behaviour::Behaviour (**BehaviourType** *type*)

Constructor with behaviour type.

Parameters

type	Behaviour type.

4.9.4 Member Function Documentation

4.9.4.1 StringVector Behaviour::getPrototype ( ) `[virtual]`

Get Behaviour encoded name and its parameters.

Returns

StringVector with encoded name and parameters as its members.

Reimplemented in ev3::BehaviourExploreRandom, and ev3::BehaviourDriveOnSquare.

4.9.4.2 std::string Behaviour::getString ( ) [virtual]

Get Behaviour human-readable name.

Returns

String with name and parameters

Reimplemented in ev3::BehaviourExploreRandom, and ev3::BehaviourDriveOnSquare.

4.9.4.3 void Behaviour::react (Event::EventType *type*)

Performs special actions based on Event passed.

Parameters

type	Event type that will be processed.

4.9.4.4 void Behaviour::setMeasurements (Measurements *measurements*)

Sensor which measurements will be required.

Parameters

measurements	Vector of sensor types.

4.9.4.5 void Behaviour::setReactionStates (BehaviourStates *reactionStates*)

Special reaction states which occur when event is fired.

Parameters

reactionStates	Vector of reaction states for this Behaviour.

4.9.4.6 void Behaviour::setStates (BehaviourStates *states*)

Available states setter.

Parameters

states	Vector of states for this Behaviour.

4.9.4.7 void Behaviour::setStopState (BehaviourState *state*)

Special stop state, used mainly to get precise sensor measurements.

Parameters

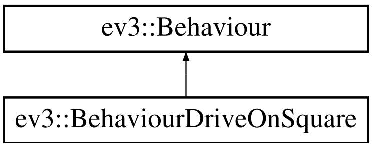
state	BehaviourState object for stop state.

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Behaviour.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Behaviour.cpp

4.10 ev3::BehaviourDriveOnSquare Class Reference

Inheritance diagram for ev3::BehaviourDriveOnSquare:



Public Member Functions

- **BehaviourDriveOnSquare** (unsigned int side, bool turningRight)
- **BehaviourDriveOnSquare** (BehaviourStates states, unsigned int side, bool turningRight)
- StringVector getPrototype () override

  Get Behaviour encoded name and its parameters.
- virtual std::string getString () override

  Get Behaviour human-readable name.

Additional Inherited Members

4.10.1 Member Function Documentation

4.10.1.1 StringVector BehaviourDriveOnSquare::getPrototype ( ) [override], [virtual]

Get Behaviour encoded name and its parameters.

Returns

StringVector with encoded name and parameters as its members.

Reimplemented from ev3::Behaviour.

4.10.1.2 std::string BehaviourDriveOnSquare::getString ( ) [override], [virtual]

Get Behaviour human-readable name.

Returns

String with name and parameters

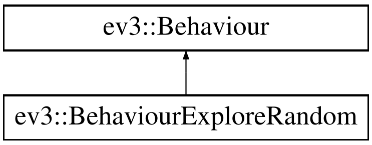
Reimplemented from ev3::Behaviour.

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Behaviour.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Behaviour.cpp

4.11 ev3::BehaviourExploreRandom Class Reference

Inheritance diagram for ev3::BehaviourExploreRandom:



Public Member Functions

- **BehaviourExploreRandom** (BehaviourStates states)
- StringVector getPrototype () override

  Get Behaviour encoded name and its parameters.
- virtual std::string getString () override

  Get Behaviour human-readable name.

Additional Inherited Members

4.11.1 Member Function Documentation

4.11.1.1 StringVector BehaviourExploreRandom::getPrototype ( ) [override], [virtual]

Get Behaviour encoded name and its parameters.

Returns

StringVector with encoded name and parameters as its members.

Reimplemented from ev3::Behaviour.

4.11.1.2 std::string BehaviourExploreRandom::getString ( ) [override], [virtual]

Get Behaviour human-readable name.

Returns

String with name and parameters

Reimplemented from ev3::Behaviour.

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Behaviour.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Behaviour.cpp

4.12 ev3::BehaviourState Class Reference

Public Member Functions

- **BehaviourState** (const BehaviourState &)=default
- **BehaviourState** (SharedPtrAction action, unsigned int nextState, bool isStopState=false)
- **BehaviourState** (SharedPtrAction action, unsigned int nextState, ReactionsTransitions reactions)
- unsigned int **process** ()
- SharedPtrAction **getAction** ()
- void **setNextState** (const unsigned int next)
- bool **isStopState** ()
- void **setReactions** (ReactionsTransitions reactions)
- int **getReaction** (Event::EventType type)

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/BehaviourState.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/BehaviourState.cpp

4.13 ev3dev::button Class Reference

Public Member Functions

- **button** (int bit)
- bool **pressed** () const
- bool **process** ()

Static Public Member Functions

- static bool **process_all** ()

Public Attributes

- std::function< void(bool)> **onclick**

Static Public Attributes

- static button **back**
- static button **left**
- static button **right**
- static button **up**
- static button **down**
- static button **enter**

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.14 ev3::CircularBuffer< T > Class Template Reference

Public Member Functions

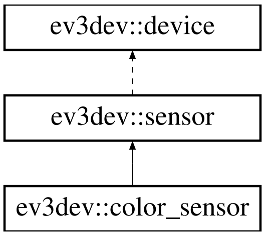
- **CircularBuffer** (unsigned int limit)
- void **push** (T object)
- bool **contain** (T object)

The documentation for this class was generated from the following file:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/utils/CircularBuffer.h

4.15 ev3dev::color_sensor Class Reference

Inheritance diagram for ev3dev::color_sensor:



Public Member Functions

- **color_sensor** (address_type address=INPUT_AUTO)
- int **reflected_light_intensity** ()
- int **ambient_light_intensity** ()
- int **color** ()
- int **red** ()
- int **green** ()
- int **blue** ()

Static Public Attributes

- static const std::string **mode_col_reflect** { "COL-REFLECT" }
- static const std::string **mode_col_ambient** { "COL-AMBIENT" }
- static const std::string **mode_col_color** { "COL-COLOR" }
- static const std::string **mode_ref_raw** { "REF-RAW" }
- static const std::string **mode_rgb_raw** { "RGB-RAW" }

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.16 ev3::ColorUtils Class Reference

Public Types

- typedef std::string **colorCode**

Static Public Member Functions

- static void **printColorTest** ()

Static Public Attributes

- static const colorCode **BLACK** {"\033[30m"}
- static const colorCode **RED** {"\033[31m"}
- static const colorCode **GREEN** {"\033[32m"}
- static const colorCode **YELLOW** {"\033[33m"}
- static const colorCode **BLUE** {"\033[34m"}
- static const colorCode **MAGENTA** {"\033[35m"}
- static const colorCode **CYAN** {"\033[36m"}
- static const colorCode **WHITE** {"\033[37m"}
- static const colorCode **BLACK_BOLD** {"\033[30;1m"}
- static const colorCode **RED_BOLD** {"\033[31;1m"}
- static const colorCode **GREEN_BOLD** {"\033[32;1m"}
- static const colorCode **YELLOW_BOLD** {"\033[33;1m"}
- static const colorCode **BLUE_BOLD** {"\033[34;1m"}
- static const colorCode **MAGENTA_BOLD** {"\033[35;1m"}
- static const colorCode **CYAN_BOLD** {"\033[36;1m"}
- static const colorCode **WHITE_BOLD** {"\033[37;1m"}
- static const colorCode **BLACK_FAINT** {"\033[30;2m"}
- static const colorCode **RED_FAINT** {"\033[31;2m"}
- static const colorCode **GREEN_FAINT** {"\033[32;2m"}
- static const colorCode **YELLOW_FAINT** {"\033[33;2m"}
- static const colorCode **BLUE_FAINT** {"\033[34;2m"}
- static const colorCode **MAGENTA_FAINT** {"\033[35;2m"}
- static const colorCode **CYAN_FAINT** {"\033[36;2m"}
- static const colorCode **WHITE_FAINT** {"\033[37;2m"}
- static const colorCode **RESET** {"\033[39;0m"}

The documentation for this class was generated from the following files:

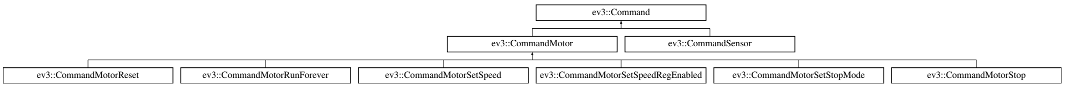
- /home/panda/Dokumenty/Repos/Ev3Dev/include/utils/ColorUtils.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/utils/ColorUtils.cpp

4.17 ev3::Command Class Reference

Base class for all command controlling classes.

`#include <Command.h>`

Inheritance diagram for ev3::Command:


ev3::Command
ev3::CommandMotor
ev3::CommandSensor
ev3::CommandMotorReset
ev3::CommandMotorRunForever
ev3::CommandMotorSetSpeed
ev3::CommandMotorSetSpeedRegEnabled
ev3::CommandMotorSetStopMode
ev3::CommandMotorStop

Public Member Functions

- Command ()

  Default constructor.
- virtual void execute ()

  Execute device specific command.
- virtual std::string getString ()

  Return Command's name.

Protected Attributes

- std::string _debugInfo = ""

  String containing Command's name.

4.17.1 Detailed Description

Base class for all command controlling classes.

Each Command class encapsulates basic motor or sensor operation.

4.17.2 Member Function Documentation

4.17.2.1 std::string Command::getString ( ) [virtual]

Return Command's name.

Returns

The documentation for this class was generated from the following files:

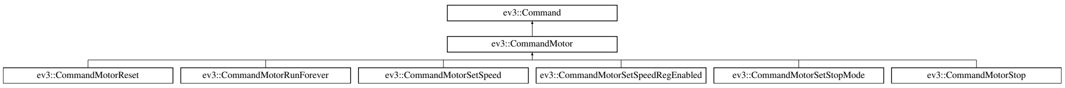
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Command.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/Command.cpp

4.18 ev3::CommandMotor Class Reference

Base class for all motor controlling commands.

`#include <CommandMotor.h>`

Inheritance diagram for ev3::CommandMotor:


ev3::Command
ev3::CommandMotor
ev3::CommandMotorReset
ev3::CommandMotorRunForever
ev3::CommandMotorSetSpeed
ev3::CommandMotorSetSpeedRegEnabled
ev3::CommandMotorSetStopMode
ev3::CommandMotorStop

Public Member Functions

- CommandMotor (Motor &motor)

  Constructor with ev3dev::motor parameter.
- Motor getMotor ()

  Get motor associated with Command.

Protected Attributes

- const std::string SPEED_REGULATION_ON = "on"

  Command parameter to turn speed regulation on a Motor on.
- const std::string SPEED_REGULATION_OFF = "off"

  Command parameter to turn speed regulation on a Motor off.
- Motor _motor

  Motor on which this CommandMotor will be executed.

4.18.1 Detailed Description

Base class for all motor controlling commands.

See also

ev3dev::motor

4.18.2 Constructor & Destructor Documentation

4.18.2.1 CommandMotor::CommandMotor (Motor & *motor*)

Constructor with ev3dev::motor parameter.

Parameters

motor	Motor to execute CommandMotor on.

4.18.3 Member Function Documentation

4.18.3.1 Motor CommandMotor::getMotor ( )

Get motor associated with Command.

Returns

Motor class object.

The documentation for this class was generated from the following files:

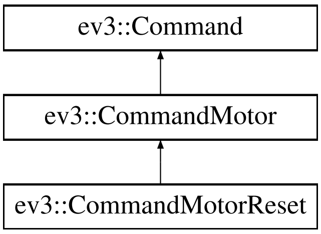
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/CommandMotor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/CommandMotor.cpp

4.19 ev3::CommandMotorReset Class Reference

Calls `reset()` method of containing Motor.

`#include <CommandMotor.h>`

Inheritance diagram for ev3::CommandMotorReset:



Public Member Functions

- CommandMotorReset (Motor &motor)

  Constructor with ev3dev::motor parameter.
- void execute () override

  Perform `reset()` method on Motor.

Additional Inherited Members

4.19.1 Detailed Description

Calls `reset()` method of containing Motor.

4.19.2 Constructor & Destructor Documentation

4.19.2.1 CommandMotorReset::CommandMotorReset (Motor & *motor*)

Constructor with ev3dev::motor parameter.

Parameters

motor	Motor to execute CommandMotor on.

The documentation for this class was generated from the following files:

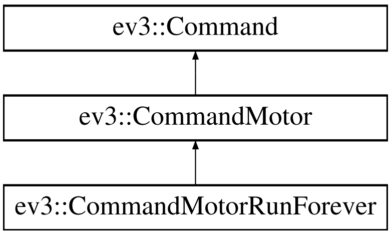
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/CommandMotor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/CommandMotor.cpp

4.20 ev3::CommandMotorRunForever Class Reference

Calls `run_forever()` method of containing Motor.

`#include <CommandMotor.h>`

Inheritance diagram for ev3::CommandMotorRunForever:



Public Member Functions

- CommandMotorRunForever (Motor &motor)

  Constructor with ev3dev::motor parameter.
- void execute () override

  Perform `run_forever()` method on Motor.

Additional Inherited Members

4.20.1 Detailed Description

Calls `run_forever()` method of containing Motor.

4.20.2 Constructor & Destructor Documentation

4.20.2.1 CommandMotorRunForever::CommandMotorRunForever (**Motor** & *motor*)

Constructor with ev3dev::motor parameter.

Parameters

motor	Motor to execute CommandMotor on.

The documentation for this class was generated from the following files:

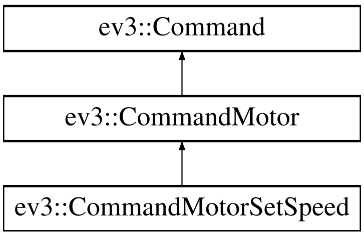
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/CommandMotor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/CommandMotor.cpp

4.21 ev3::CommandMotorSetSpeed Class Reference

Call `set_speed_sp()` method of containing Motor.

```
#include <CommandMotor.h>
```

Inheritance diagram for ev3::CommandMotorSetSpeed:



Public Member Functions

- CommandMotorSetSpeed (Motor &motor, int value)

  Constructor with ev3dev::motor parameter.
- void execute () override

  Perform `set_speed_sp()` method on Motor.

Additional Inherited Members

4.21.1 Detailed Description

Call `set_speed_sp()` method of containing Motor.

4.21.2 Constructor & Destructor Documentation

4.21.2.1 CommandMotorSetSpeed::CommandMotorSetSpeed (**Motor** & *motor,* int *value*)

Constructor with ev3dev::motor parameter.

Parameters

motor	Motor to execute CommandMotor on.
value	Speed value in tacho pulses per second.

Warning

Speed regulation must be turned on for this to take effect.

See also

CommandMotorSetSpeedRegEnabled

The documentation for this class was generated from the following files:

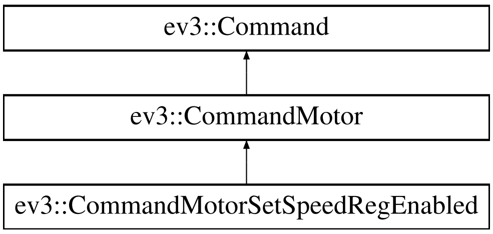
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/CommandMotor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/CommandMotor.cpp

4.22 ev3::CommandMotorSetSpeedRegEnabled Class Reference

Calls `set_speed_regulation_enabled()` method of containing Motor.

`#include <CommandMotor.h>`

Inheritance diagram for ev3::CommandMotorSetSpeedRegEnabled:



Public Member Functions

- CommandMotorSetSpeedRegEnabled (Motor &motor, bool value)

  Constructor with ev3dev::motor parameter.
- void execute () override

  Perform `set_speed_regulation_enabled()` on Motor.

Additional Inherited Members

4.22.1 Detailed Description

Calls `set_speed_regulation_enabled()` method of containing Motor.

4.22.2 Constructor & Destructor Documentation

4.22.2.1 CommandMotorSetSpeedRegEnabled::CommandMotorSetSpeedRegEnabled (Motor & *motor,* bool *value*)

Constructor with ev3dev::motor parameter.

Parameters

motor	Motor to execute CommandMotor on.
value	If true, turn speed regulation on, false to turn it off.

The documentation for this class was generated from the following files:

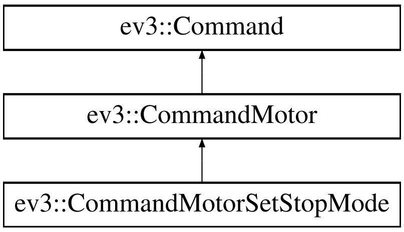
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/CommandMotor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/CommandMotor.cpp

4.23 ev3::CommandMotorSetStopMode Class Reference

Calls `set_stop_command()` method of containing Motor.

```
#include <CommandMotor.h>
```

Inheritance diagram for ev3::CommandMotorSetStopMode:



Public Types

- enum StopMode { COAST, BRAKE, HOLD }

  Stop modes for motors.

Public Member Functions

- CommandMotorSetStopMode (Motor &motor, StopMode mode)

  Constructor with ev3dev::motor parameter.
- void execute () override

  Perform `set_stop_command()` method on Motor.

Additional Inherited Members

4.23.1 Detailed Description

Calls `set_stop_command()` method of containing Motor.

4.23.2 Member Enumeration Documentation

4.23.2.1 enum ev3::CommandMotorSetStopMode::StopMode

Stop modes for motors.

Enumerator

COAST No voltage. Motor slowly stops.

BRAKE Passive braking. Motor stops faster.

HOLD Active braking. Hardly prevent motor from any movement.

4.23.3 Constructor & Destructor Documentation

4.23.3.1 CommandMotorSetStopMode::CommandMotorSetStopMode (**Motor** & *motor,* **StopMode** *mode*)

Constructor with ev3dev::motor parameter.

Parameters

motor	Motor to execute CommandMotor on.
mode	Stop mode chosen from StopMode.

The documentation for this class was generated from the following files:

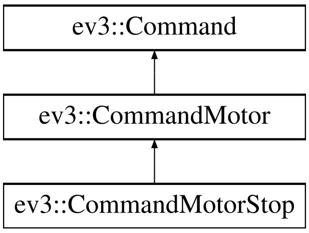
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/CommandMotor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/CommandMotor.cpp

4.24 ev3::CommandMotorStop Class Reference

Calls `stop()` method of containing Motor.

```
#include <CommandMotor.h>
```

Inheritance diagram for ev3::CommandMotorStop:



Public Member Functions

- CommandMotorStop (Motor &motor)

    Constructor with ev3dev::motor parameter.
- void execute () override

    Perform `stop()` method on Motor.

Additional Inherited Members

4.24.1 Detailed Description

Calls `stop()` method of containing Motor.

4.24.2 Constructor & Destructor Documentation

4.24.2.1 CommandMotorStop::CommandMotorStop (**Motor &** ***motor***)

Constructor with ev3dev::motor parameter.

Parameters

motor	Motor to execute CommandMotor on.

The documentation for this class was generated from the following files:

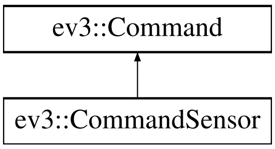
- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/CommandMotor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/CommandMotor.cpp

4.25 ev3::CommandSensor Class Reference

Base class for all sensor controlling commands.

`#include <CommandSensor.h>`

Inheritance diagram for ev3::CommandSensor:



Public Member Functions

- CommandSensor (Sensor &sensor)

    Constructor with ev3dev::sensor parameter.
- Sensor getSensor ()

    Get sensor associated with Command.

Protected Attributes

- Sensor _sensor

    Sensor on which this CommandSensor will be executed.

4.25.1 Detailed Description

Base class for all sensor controlling commands.

See also

ev3dev::sensor

4.25.2 Constructor & Destructor Documentation

4.25.2.1 CommandSensor::CommandSensor (Sensor & *sensor*)

Constructor with ev3dev::sensor parameter.

Parameters

sensor	Sensor to execute CommandSensor on.

4.25.3 Member Function Documentation

4.25.3.1 Sensor CommandSensor::getSensor ( )

Get sensor associated with Command.

Returns

Sensor class object.

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/action/CommandSensor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/action/CommandSensor.cpp

4.26 ev3::Communication Class Reference

Public Member Functions

- std::thread **createThread** (Queue< Message > ∗sendQueue, Queue< Message > ∗receiveQueue, bool isMaster=false)
- void **run** (Queue< Message > ∗sendQueue, Queue< Message > ∗receiveQueue, bool isMaster=false)

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/communication/Communication.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/communication/Communication.cpp

4.27 ev3::CommUtils Class Reference

Classes

- struct NetworkNode

Public Member Functions

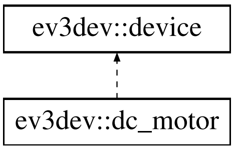
- int **preparePassiveSocket** (unsigned int portNumber)
- int **sendMessage** (unsigned int socket, unsigned int port, Message &message, std::string &proto, bool isMaster, unsigned int repeat=SENT_MESSAGE_COPIES)
- int **receiveMessage** (unsigned int socket, Message &msg, NetworkNode &sender)
- int **receiveMessageDelay** (unsigned int socket, Message &msg, NetworkNode &sender, unsigned int msDelay=DEFAULT_RECEIVE_DELAY)

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/communication/CommUtils.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/communication/CommUtils.cpp

4.28 ev3dev::dc_motor Class Reference

Inheritance diagram for ev3dev::dc_motor:



Public Member Functions

- **dc_motor** (address_type address=OUTPUT_AUTO)
- auto **set_command** (std::string v) -> decltype(∗this)
- mode_set **commands** () const
- std::string **driver_name** () const
- int **duty_cycle** () const
- int **duty_cycle_sp** () const
- auto **set_duty_cycle_sp** (int v) -> decltype(∗this)
- std::string **polarity** () const
- auto **set_polarity** (std::string v) -> decltype(∗this)
- std::string **address** () const
- int **ramp_down_sp** () const
- auto **set_ramp_down_sp** (int v) -> decltype(∗this)
- int **ramp_up_sp** () const
- auto **set_ramp_up_sp** (int v) -> decltype(∗this)
- mode_set **state** () const
- auto **set_stop_command** (std::string v) -> decltype(∗this)
- mode_set **stop_commands** () const
- int **time_sp** () const
- auto **set_time_sp** (int v) -> decltype(∗this)
- void **run_forever** ()
- void **run_timed** ()
- void **run_direct** ()
- void **stop** ()

Static Public Attributes

- static const std::string **command_run_forever** { "run-forever" }
- static const std::string **command_run_timed** { "run-timed" }
- static const std::string **command_run_direct** { "run-direct" }
- static const std::string **command_stop** { "stop" }
- static const std::string **polarity_normal** { "normal" }
- static const std::string **polarity_inversed** { "inversed" }
- static const std::string **stop_command_coast** { "coast" }
- static const std::string **stop_command_brake** { "brake" }

Protected Attributes

- std::string **_port_name**

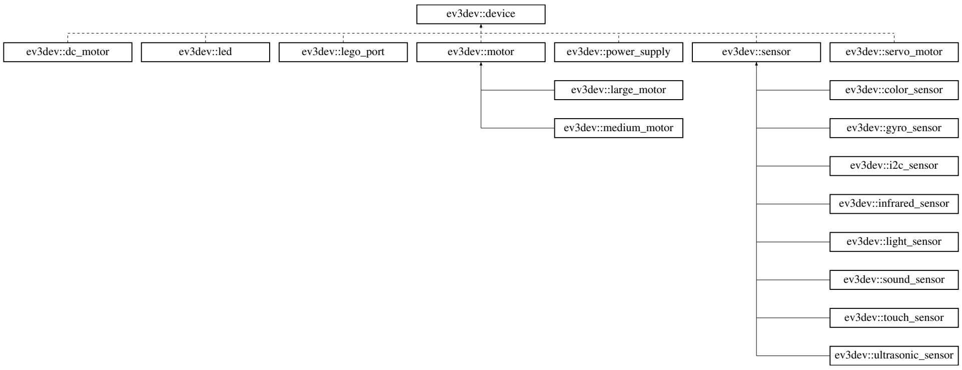
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.29 ev3dev::device Class Reference

Inheritance diagram for ev3dev::device:



Public Member Functions

- bool **connect** (const std::string &dir, const std::string &pattern, const std::map< std::string, std::set< std↩ ::string >> &match) noexcept
- bool **connected** () const
- int **device_index** () const
- int **get_attr_int** (const std::string &name) const
- void **set_attr_int** (const std::string &name, int value)
- std::string **get_attr_string** (const std::string &name) const
- void **set_attr_string** (const std::string &name, const std::string &value)
- std::string **get_attr_line** (const std::string &name) const
- mode_set **get_attr_set** (const std::string &name, std::string ∗pCur=nullptr) const
- std::string **get_attr_from_set** (const std::string &name) const

Protected Attributes

- std::string **_path**
- int **_device_index** = -1

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.30 ev3::Devices Class Reference

Public Types

- typedef std::map< ev3dev::port_type, Motor > **MotorsVector**
- typedef std::map< ev3dev::port_type, Sensor > **SensorsVector**
- typedef std::vector< std::pair< ev3dev::port_type, ev3dev::device_type > > **RequiredDevices**
- typedef std::map< ev3dev::port_type, SensorValue > **SensorStatus**

Public Member Functions

- bool **checkDevices** (RequiredDevices &devices)
- void **update** ()
- void **addListener** (Sensor::SensorType type)
- void **removeListener** (Sensor::SensorType type)
- Motor **getMotor** (ev3dev::port_type port)
- Sensor **getSensor** (ev3dev::port_type port)
- void **setSafetyTouchSensor** (ev3dev::port_type port)
- void **setProximitySensor** (ev3dev::port_type port)
- void **stopAllDevices** ()

Static Public Member Functions

- static Devices ∗ **getInstance** ()
- static void **destroy** ()

Static Public Attributes

- static const ev3dev::port_type **PORT_ANY** {"any"}

Protected Member Functions

- **Devices** (const Devices &)
- Devices & **operator=** (const Devices &)

Protected Attributes

- std::map< Sensor::SensorType, bool > **_listeners**
- std::map< ev3dev::port_type, int > **_safetyTouchSensors**
- std::map< ev3dev::port_type, int > **_proximitySensors**
- MotorsVector **_motors**
- SensorsVector **_sensors**
- SensorStatus **_status**

Static Protected Attributes

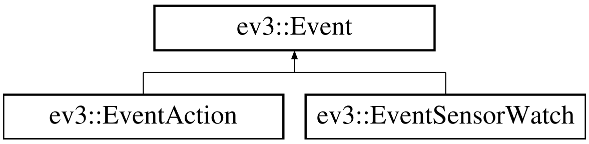
- static Devices ∗ **_instance** = nullptr

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/Devices.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/Devices.cpp

4.31 ev3::Event Class Reference

Inheritance diagram for ev3::Event:



Public Types

- enum **EventType** {
EMPTY, **BEHAVIOUR_START**, **BEHAVIOUR_STOP**, **SENSOR_WATCH**,
OBSTACLE_DETECTED, **PROXIMITY_ALERT**, **ACTION_FINISHED**, **ACTION_INTERR** }

Public Member Functions

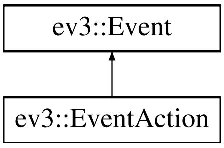
- **Event** (EventType type)
- EventType **getType** ()
- std::string **getStringType** ()

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/communication/Event.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/communication/Event.cpp

4.32 ev3::EventAction Class Reference

Inheritance diagram for ev3::EventAction:



Public Member Functions

- **EventAction** (EventType eventType, Action::ActionType actionType)
- Action::ActionType **getActionType** ()

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/communication/Event.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/communication/Event.cpp

4.33 ev3::EventQueue Class Reference

Public Member Functions

- void **push** (SharedPtrEvent message)
- SharedPtrEvent **pop** ()
- bool **empty** ()
- unsigned int **size** ()

Static Public Member Functions

- static EventQueue ∗ **getInstance** ()
- static void **destroy** ()

Protected Member Functions

- **EventQueue** (const EventQueue &)
- EventQueue & **operator=** (const EventQueue &)

Protected Attributes

- std::queue< SharedPtrEvent > **_queue**
- std::mutex **_mutex**

Static Protected Attributes

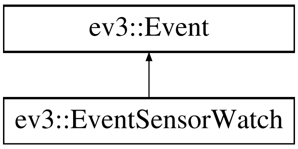
- static EventQueue ∗ **_instance** = nullptr

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/utils/EventQueue.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/utils/EventQueue.cpp

4.34 ev3::EventSensorWatch Class Reference

Inheritance diagram for ev3::EventSensorWatch:



Public Member Functions

- **EventSensorWatch** (Sensor::SensorType type, SensorValue value)
- SensorValue **getValue** ()
- Sensor::SensorType **getType** ()

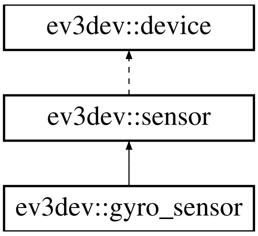
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/communication/Event.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/communication/Event.cpp

4.35 ev3dev::gyro_sensor Class Reference

Inheritance diagram for ev3dev::gyro_sensor:



Public Member Functions

- **gyro_sensor** (address_type address=INPUT_AUTO)
- int **angle** ()
- int **rate** ()

Static Public Attributes

- static const std::string **mode_gyro_ang** { "GYRO-ANG" }
- static const std::string **mode_gyro_rate** { "GYRO-RATE" }
- static const std::string **mode_gyro_fas** { "GYRO-FAS" }
- static const std::string **mode_gyro_g_a** { "GYRO-G&A" }
- static const std::string **mode_gyro_cal** { "GYRO-CAL" }

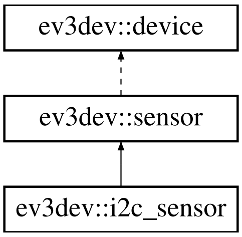
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.36 ev3dev::i2c_sensor Class Reference

Inheritance diagram for ev3dev::i2c_sensor:



Public Member Functions

- **i2c_sensor** (address_type address=INPUT_AUTO)
- std::string **fw_version** () const
- int **poll_ms** () const
- auto **set_poll_ms** (int v) -> decltype(∗this)

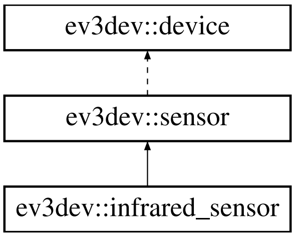
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.37 ev3dev::infrared_sensor Class Reference

Inheritance diagram for ev3dev::infrared_sensor:



Public Member Functions

- **infrared_sensor** (address_type address=INPUT_AUTO)
- int **proximity** ()

Static Public Attributes

- static const std::string **mode_ir_prox** { "IR-PROX" }
- static const std::string **mode_ir_seek** { "IR-SEEK" }
- static const std::string **mode_ir_remote** { "IR-REMOTE" }
- static const std::string **mode_ir_rem_a** { "IR-REM-A" }
- static const std::string **mode_ir_cal** { "IR-CAL" }

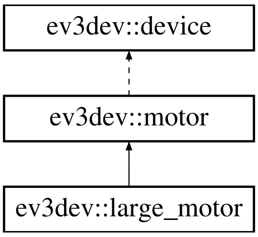
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.38 ev3dev::large_motor Class Reference

Inheritance diagram for ev3dev::large_motor:



Public Member Functions

- **large_motor** (address_type address=OUTPUT_AUTO)

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.39 ev3dev::lcd Class Reference

Public Member Functions

- bool **available** () const
- uint32_t **resolution_x** () const
- uint32_t **resolution_y** () const
- uint32_t **bits_per_pixel** () const
- uint32_t **frame_buffer_size** () const
- uint32_t **line_length** () const
- unsigned char $*$ **frame_buffer** ()
- void **fill** (unsigned char pixel)

Protected Member Functions

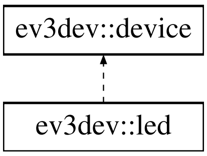
- void **init** ()
- void **deinit** ()

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.40 ev3dev::led Class Reference

Inheritance diagram for ev3dev::led:



Public Member Functions

- **led** (std::string name)
- int **max_brightness** () const
- int **brightness** () const
- auto **set_brightness** (int v) -> decltype(∗this)
- mode_set **triggers** () const
- std::string **trigger** () const
- auto **set_trigger** (std::string v) -> decltype(∗this)
- int **delay_on** () const
- auto **set_delay_on** (int v) -> decltype(∗this)
- int **delay_off** () const
- auto **set_delay_off** (int v) -> decltype(∗this)
- float **brightness_pct** () const
- auto **set_brightness_pct** (float v) -> decltype(∗this)
- void **on** ()
- void **off** ()
- void **flash** (unsigned on_ms, unsigned off_ms)

Static Public Member Functions

- static void **set_color** (const std::vector< led ∗ > &group, const std::vector< float > &color)
- static void **all_off** ()

Static Public Attributes

- static led **red_left** {"ev3:left:red:ev3dev"}
- static led **red_right** {"ev3:right:red:ev3dev"}
- static led **green_left** {"ev3:left:green:ev3dev"}
- static led **green_right** {"ev3:right:green:ev3dev"}
- static std::vector< led ∗ > **left** { &led::red_left, &led::green_left }
- static std::vector< led ∗ > **right** { &led::red_right, &led::green_right }
- static std::vector< float > **red** { static_cast<float>(1), static_cast<float>(0) }
- static std::vector< float > **green** { static_cast<float>(0), static_cast<float>(1) }
- static std::vector< float > **amber** { static_cast<float>(1), static_cast<float>(1) }
- static std::vector< float > **orange** { static_cast<float>(1), static_cast<float>(0.5) }
- static std::vector< float > **yellow** { static_cast<float>(0.5), static_cast<float>(1) }

Protected Attributes

- int **_max_brightness** = 0

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.41 ev3::LedControl Class Reference

Public Types

- enum **LedType** {
  RED_L = 1, **RED_R** = 1 << 1, **GREEN_L** = 1 << 2, **GREEN_R** = 1 << 3,
  RED_ALL = RED_L | RED_R, **GREEN_ALL** = GREEN_L | GREEN_R, **ALL** = RED_ALL | GREEN_ALL }
- enum **LedColors** { **RED**, **AMBER**, **YELLOW**, **GREEN** }

Public Member Functions

- void **on** (unsigned int leds=LedType::ALL, unsigned int brightness=MAX_BRIGHTNESS)
- void **onExclusive** (unsigned int leds=LedType::ALL, unsigned int brightness=MAX_BRIGHTNESS)
- void **off** (unsigned int leds=LedType::ALL)
- void **setColor** (LedColors color)
- void **reset** ()
- void **flash** (unsigned int leds, unsigned int msInterval, unsigned int repeat=1, unsigned int brightnessRed=MAX_BRIGHTNESS, unsigned int brightnessGreen=MAX_BRIGHTNESS)
- void **flashColor** (LedColors color, unsigned int msInterval, unsigned int repeat=1)
- void **endFlashing** ()

Static Public Attributes

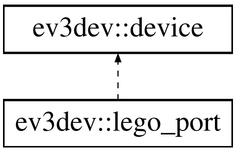
- static const unsigned int **MAX_BRIGHTNESS** = 255

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/control/LedControl.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/control/LedControl.cpp

4.42 ev3dev::lego_port Class Reference

Inheritance diagram for ev3dev::lego_port:



Public Member Functions

- **lego_port** (address_type)
- std::string **driver_name** () const
- mode_set **modes** () const
- std::string **mode** () const
- auto **set_mode** (std::string v) -> decltype(*this)
- std::string **address** () const
- auto **set_set_device** (std::string v) -> decltype(*this)
- std::string **status** () const

Protected Member Functions

- bool **connect** (const std::map< std::string, std::set< std::string >> &) noexcept

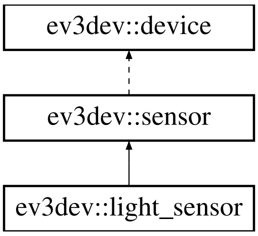
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.43 ev3dev::light_sensor Class Reference

Inheritance diagram for ev3dev::light_sensor:



Public Member Functions

- **light_sensor** (address_type address=INPUT_AUTO)
- float **reflected_light_intensity** ()
- float **ambient_light_intensity** ()

Static Public Attributes

- static const std::string **mode_reflect** { "REFLECT" }
- static const std::string **mode_ambient** { "AMBIENT" }

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.44 ev3::Logger Class Reference

Public Types

- enum **LogLevel** {
  DEBUG = 1, **VERBOSE** = 1 << 1, **INFO** = 1 << 2, **WARNING** = 1 << 3,
  ERROR = 1 << 4 }
- enum **LogOutput** { **STD_OUT** = 1, **STD_ERR** = 1 << 1, **FILE** = 1 << 2 }

Public Member Functions

- void **log** (std::string message, LogLevel level, LogOutput output=STD_OUT)
- void **setLogLevel** (LogLevel level)
- void **setLogLevel** (std::string level)
- void **setLogOutput** (LogOutput output)

Static Public Member Functions

- static Logger ∗ **getInstance** ()
- static void **destroy** ()

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/utils/Logger.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/utils/Logger.cpp

4.45 ev3::Master Class Reference

Public Types

- typedef std::map< unsigned int, Agent > **AgentMap**

Public Member Functions

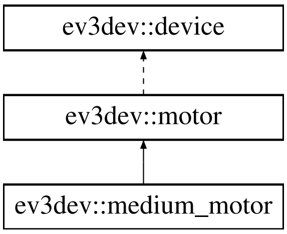
- std::thread **createThread** (Queue< Message > ∗sendQueue, Queue< Message > ∗receiveQueue)
- void **run** (Queue< Message > ∗sendQueue, Queue< Message > ∗receiveQueue)
- void **send** (Message message, bool recordMessage=true)
- void **stop** ()

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/master/Master.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/master/Master.cpp

4.46 ev3dev::medium_motor Class Reference

Inheritance diagram for ev3dev::medium_motor:



Public Member Functions

- **medium_motor** (address_type address=OUTPUT_AUTO)

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.47 ev3::Message Class Reference

Public Types

- enum **MessageType** {
  EMPTY, **ACK**, **NOT**, **AGENT**,
  MASTER, **MASTER_OVER**, **PING**, **PONG**,
  AGENT_OVER, **ABORT**, **BEHAVIOUR**, **START**,
  RESUME, **PAUSE**, **ACTION_OK**, **ACTION_INTERR**,
  SENSOR_VALUE, **MEASURE** }

Public Member Functions

- **Message** (unsigned int senderId, unsigned int receiverId, unsigned int messageId, MessageType type, StringVector parameters={})
- unsigned int **getSenderId** ()
- unsigned int **getReceiverId** ()
- unsigned int **getMessageId** ()
- MessageType **getType** ()
- StringVector **getParameters** ()
- void **setSenderId** (unsigned int id)
- void **setReceiverId** (unsigned int id)
- void **setMessageId** (unsigned int id)
- void **setType** (MessageType type)
- void **setParameters** (StringVector parameters)
- bool **empty** ()
- std::string **getString** ()
- void **reset** ()

Static Public Member Functions

- static std::string **encodeMessage** (Message &message)
- static Message **decodeMessage** (const std::string msg)

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/communication/Message.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/communication/Message.cpp

4.48 ev3::Motor Class Reference

Public Member Functions

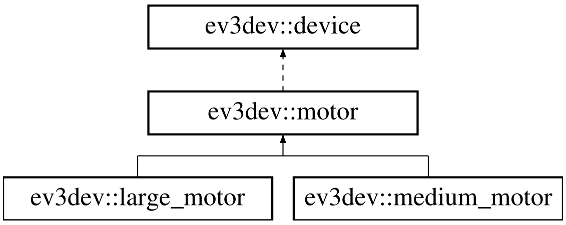
- **Motor** (ev3dev::motor motor)
- ev3dev::motor **getMotor** ()

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/Motor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/Motor.cpp

4.49 ev3dev::motor Class Reference

Inheritance diagram for ev3dev::motor:



Public Types

- typedef device_type **motor_type**

Public Member Functions

- **motor** (address_type)
- **motor** (address_type, const motor_type &)
- auto **set_command** (std::string v) -> decltype(∗this)
- mode_set **commands** () const
- int **count_per_rot** () const
- std::string **driver_name** () const
- int **duty_cycle** () const
- int **duty_cycle_sp** () const
- auto **set_duty_cycle_sp** (int v) -> decltype(∗this)
- std::string **encoder_polarity** () const
- auto **set_encoder_polarity** (std::string v) -> decltype(∗this)
- std::string **polarity** () const
- auto **set_polarity** (std::string v) -> decltype(∗this)
- std::string **address** () const
- int **position** () const
- auto **set_position** (int v) -> decltype(∗this)
- int **position_p** () const
- auto **set_position_p** (int v) -> decltype(∗this)
- int **position_i** () const
- auto **set_position_i** (int v) -> decltype(∗this)
- int **position_d** () const
- auto **set_position_d** (int v) -> decltype(∗this)
- int **position_sp** () const
- auto **set_position_sp** (int v) -> decltype(∗this)
- int **speed** () const
- int **speed_sp** () const
- auto **set_speed_sp** (int v) -> decltype(∗this)
- int **ramp_up_sp** () const
- auto **set_ramp_up_sp** (int v) -> decltype(∗this)
- int **ramp_down_sp** () const
- auto **set_ramp_down_sp** (int v) -> decltype(∗this)
- std::string **speed_regulation_enabled** () const
- auto **set_speed_regulation_enabled** (std::string v) -> decltype(∗this)
- int **speed_regulation_p** () const
- auto **set_speed_regulation_p** (int v) -> decltype(∗this)
- int **speed_regulation_i** () const
- auto **set_speed_regulation_i** (int v) -> decltype(∗this)
- int **speed_regulation_d** () const
- auto **set_speed_regulation_d** (int v) -> decltype(∗this)
- mode_set **state** () const
- std::string **stop_command** () const
- auto **set_stop_command** (std::string v) -> decltype(∗this)
- mode_set **stop_commands** () const
- int **time_sp** () const
- auto **set_time_sp** (int v) -> decltype(∗this)
- void **run_forever** ()
- void **run_to_abs_pos** ()
- void **run_to_rel_pos** ()
- void **run_timed** ()
- void **run_direct** ()
- void **stop** ()
- void **reset** ()
- motor_type **type_name** ()

Static Public Attributes

- static const motor_type **motor_large** { "lego-ev3-l-motor" }
- static const motor_type **motor_medium** { "lego-ev3-m-motor" }
- static const std::string **command_run_forever** { "run-forever" }
- static const std::string **command_run_to_abs_pos** { "run-to-abs-pos" }
- static const std::string **command_run_to_rel_pos** { "run-to-rel-pos" }
- static const std::string **command_run_timed** { "run-timed" }
- static const std::string **command_run_direct** { "run-direct" }
- static const std::string **command_stop** { "stop" }
- static const std::string **command_reset** { "reset" }
- static const std::string **encoder_polarity_normal** { "normal" }
- static const std::string **encoder_polarity_inversed** { "inversed" }
- static const std::string **polarity_normal** { "normal" }
- static const std::string **polarity_inversed** { "inversed" }
- static const std::string **speed_regulation_on** { "on" }
- static const std::string **speed_regulation_off** { "off" }
- static const std::string **stop_command_coast** { "coast" }
- static const std::string **stop_command_brake** { "brake" }
- static const std::string **stop_command_hold** { "hold" }

Protected Member Functions

- bool **connect** (const std::map< std::string, std::set< std::string >> &) noexcept

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.50 ev3::CommUtils::NetworkNode Struct Reference

Public Attributes

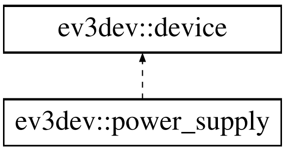
- unsigned int **port**
- std::string **ipAddress**

The documentation for this struct was generated from the following file:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/communication/CommUtils.h

4.51 ev3dev::power_supply Class Reference

Inheritance diagram for ev3dev::power_supply:



Public Member Functions

- **power_supply** (std::string name)
- int **measured_current** () const
- int **measured_voltage** () const
- int **max_voltage** () const
- int **min_voltage** () const
- std::string **technology** () const
- std::string **type** () const
- float **measured_amps** () const
- float **measured_volts** () const

Static Public Attributes

- static power_supply **battery** { "" }

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.52 ev3::Queue< T > Class Template Reference

Public Member Functions

- void **push** (T message)
- T **pop** ()
- bool **empty** ()

The documentation for this class was generated from the following file:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/utils/Queue.h

4.53 ev3dev::remote_control Class Reference

Public Types

- enum **buttons** {
  red_up = (1 << 0), **red_down** = (1 << 1), **blue_up** = (1 << 2), **blue_down** = (1 << 3),
  beacon = (1 << 4) }

Public Member Functions

- **remote_control** (unsigned channel=1)
- **remote_control** (infrared_sensor &, unsigned channel=1)
- bool **connected** () const
- unsigned **channel** () const
- bool **process** ()

Public Attributes

- std::function< void(bool)> **on_red_up**
- std::function< void(bool)> **on_red_down**
- std::function< void(bool)> **on_blue_up**
- std::function< void(bool)> **on_blue_down**
- std::function< void(bool)> **on_beacon**
- std::function< void(int)> **on_state_change**

Protected Member Functions

- virtual void **on_value_changed** (int value)

Protected Attributes

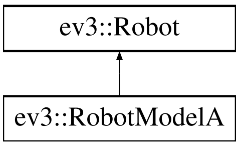
- infrared_sensor * **_sensor** = nullptr
- bool **_owns_sensor** = false
- unsigned **_channel** = 0
- int **_value** = 0
- int **_state** = 0

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.54 ev3::Robot Class Reference

Inheritance diagram for ev3::Robot:



Public Types

- typedef std::vector< Action::ActionType > AvailableActions

  Type for specifying all available actions for given Robot model.

Public Member Functions

- **Robot** (Devices::RequiredDevices devices, AvailableActions actions)
- std::thread **createThread** (Queue< Message > ∗sendQueue, Queue< Message > ∗receiveQueue)
- virtual void **run** (Queue< Message > ∗sendQueue, Queue< Message > ∗receiveQueue)
- void **stop** ()
- void **send** (Message message)
- virtual std::string **getString** ()

Protected Member Functions

- virtual SharedPtrBehaviour **generateBehaviour** (Behaviour::BehaviourType type, StringVector parameters)

Protected Attributes

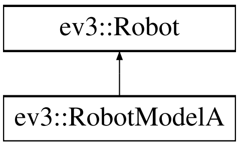
- unsigned int **_id** = 0
- unsigned int **_commId** = 0
- float **_pulsePerUnitRatio** = 1.f
- Devices::RequiredDevices **_requiredDevices**
- AvailableActions **_availableActions**
- Queue< Message > ∗ **_sendQueue**
- Queue< Message > ∗ **_receiveQueue**
- LedControl **_ledControl**
- RobotState ∗ **_state** = new RobotStateIdle(&_ledControl)

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/Robot.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/Robot.cpp

4.55 ev3::RobotModelA Class Reference

Inheritance diagram for ev3::RobotModelA:



Public Member Functions

- virtual std::string **getString** () override

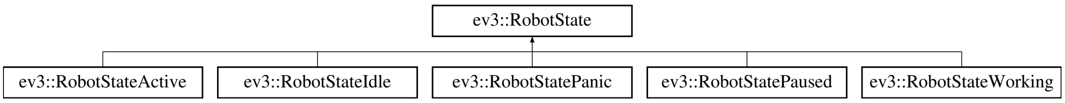
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/RobotModelA.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/RobotModelA.cpp

4.56 ev3::RobotState Class Reference

Inheritance diagram for ev3::RobotState:



Public Types

- enum **States** {
  IDLE, **ACTIVE**, **WORKING**, **PAUSED**,
  PANIC }
- typedef std::map< Message::MessageType, States > **ChangeMap**

Public Member Functions

- **RobotState** (ChangeMap changes, LedControl ∗led)
- virtual RobotState ∗ **process** (Message msg)
- Message::MessageType **getPendingMessage** ()
- void **updateTimer** ()
- bool **isPendingEnabled** ()
- void **setBehaviour** (SharedPtrBehaviour behaviour)
- SharedPtrBehaviour **getBehaviour** ()

Static Public Attributes

- static const float **MASTER_TIMEOUT** = 10.f ∗ 1000
- static const float **MASTER_PING_TIME** = 3.f ∗ 1000

Protected Member Functions

- RobotState ∗ **switchState** (Message::MessageType type)
- RobotState ∗ **changeState** (States state)

Protected Attributes

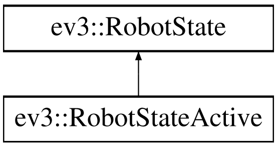
- SharedPtrBehaviour **_currentBehaviour**
- States **_state**
- ChangeMap **_changes**
- LedControl ∗ **_led**
- Message::MessageType **_pendingMessage** = Message::EMPTY
- float **_pendingTimeout** = 0.f
- HighResClock::time_point **_masterTimeout** = HighResClock::now()
- HighResClock::time_point **_messageDelay** = HighResClock::now()

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/RobotState.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/RobotState.cpp

4.57 ev3::RobotStateActive Class Reference

Inheritance diagram for ev3::RobotStateActive:



Public Member Functions

- **RobotStateActive** (LedControl ∗led)
- RobotState ∗ **process** (Message msg)

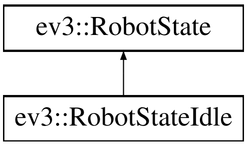
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/RobotState.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/RobotState.cpp

4.58 ev3::RobotStateIdle Class Reference

Inheritance diagram for ev3::RobotStateIdle:



Public Member Functions

- **RobotStateIdle** (LedControl ∗led)
- RobotState ∗ **process** (Message msg)

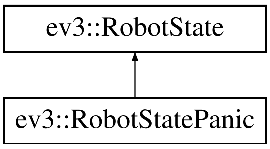
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/RobotState.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/RobotState.cpp

4.59 ev3::RobotStatePanic Class Reference

Inheritance diagram for ev3::RobotStatePanic:



Public Member Functions

- **RobotStatePanic** (LedControl ∗led)
- RobotState ∗ **process** (Message msg)

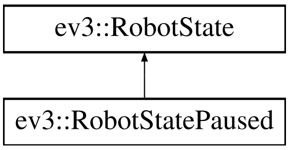
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/RobotState.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/RobotState.cpp

4.60 ev3::RobotStatePaused Class Reference

Inheritance diagram for ev3::RobotStatePaused:



Public Member Functions

- **RobotStatePaused** (LedControl ∗led)
- RobotState ∗ **process** (Message msg)

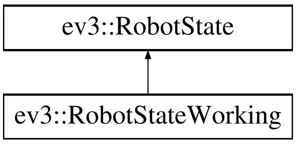
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/RobotState.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/RobotState.cpp

4.61 ev3::RobotStateWorking Class Reference

Inheritance diagram for ev3::RobotStateWorking:



Public Member Functions

- **RobotStateWorking** (LedControl ∗led)
- RobotState ∗ **process** (Message msg)

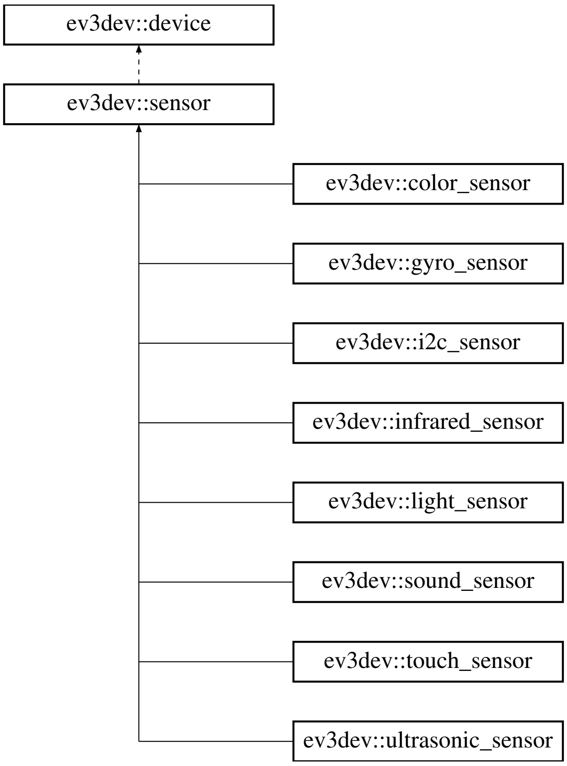
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/RobotState.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/RobotState.cpp

4.62 ev3dev::sensor Class Reference

Inheritance diagram for ev3dev::sensor:



Public Types

- typedef device_type **sensor_type**

Public Member Functions

- **sensor** (address_type)
- **sensor** (address_type, const std::set< sensor_type > &)
- int **value** (unsigned index=0) const
- float **float_value** (unsigned index=0) const
- std::string **type_name** () const
- std::string **bin_data_format** () const
- const std::vector< char > & **bin_data** () const
- template<class T >
  void **bin_data** (T ∗buf) const
- auto **set_command** (std::string v) -> decltype(∗this)
- mode_set **commands** () const
- int **decimals** () const
- std::string **driver_name** () const
- std::string **mode** () const
- auto **set_mode** (std::string v) -> decltype(∗this)
- mode_set **modes** () const
- int **num_values** () const
- std::string **address** () const
- std::string **units** () const

Static Public Attributes

- static const sensor_type **ev3_touch** { "lego-ev3-touch" }
- static const sensor_type **ev3_color** { "lego-ev3-color" }
- static const sensor_type **ev3_ultrasonic** { "lego-ev3-us" }
- static const sensor_type **ev3_gyro** { "lego-ev3-gyro" }
- static const sensor_type **ev3_infrared** { "lego-ev3-ir" }
- static const sensor_type **nxt_touch** { "lego-nxt-touch" }
- static const sensor_type **nxt_light** { "lego-nxt-light" }
- static const sensor_type **nxt_sound** { "lego-nxt-sound" }
- static const sensor_type **nxt_ultrasonic** { "lego-nxt-us" }
- static const sensor_type **nxt_i2c_sensor** { "nxt-i2c-sensor" }
- static const sensor_type **nxt_analog** { "nxt-analog" }

Protected Member Functions

- bool **connect** (const std::map< std::string, std::set< std::string >> &) noexcept

Protected Attributes

- std::vector< char > **_bin_data**

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.63 ev3::Sensor Class Reference

Public Types

- enum **SensorType** {
  TOUCH, **COLOR**, **ULTRASONIC**, **GYRO**,
  INFRARED, **SOUND**, **LIGHT** }

Public Member Functions

- **Sensor** (ev3dev::sensor sensor, SensorType type)
- ev3dev::sensor **getSensor** ()
- int **getValue** (unsigned int n)
- float **getValueF** (unsigned int n)
- int **getDecimals** ()
- unsigned int **getNumValues** ()
- SensorType **getType** ()

Static Public Member Functions

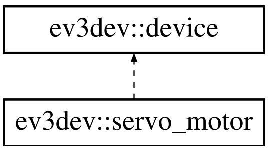
- static StringVector **prepareMessage** (SensorValue value, SensorType type)

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/robot/Sensor.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/robot/Sensor.cpp

4.64 ev3dev::servo_motor Class Reference

Inheritance diagram for ev3dev::servo_motor:



Public Member Functions

- **servo_motor** (address_type address=OUTPUT_AUTO)
- auto **set_command** (std::string v) -> decltype(∗this)
- std::string **driver_name** () const
- int **max_pulse_sp** () const
- auto **set_max_pulse_sp** (int v) -> decltype(∗this)
- int **mid_pulse_sp** () const
- auto **set_mid_pulse_sp** (int v) -> decltype(∗this)
- int **min_pulse_sp** () const
- auto **set_min_pulse_sp** (int v) -> decltype(∗this)
- std::string **polarity** () const
- auto **set_polarity** (std::string v) -> decltype(∗this)
- std::string **address** () const
- int **position_sp** () const
- auto **set_position_sp** (int v) -> decltype(∗this)
- int **rate_sp** () const
- auto **set_rate_sp** (int v) -> decltype(∗this)
- mode_set **state** () const
- void **run** ()
- void **float_** ()

Static Public Attributes

- static const std::string **command_run** { "run" }
- static const std::string **command_float** { "float" }
- static const std::string **polarity_normal** { "normal" }
- static const std::string **polarity_inversed** { "inversed" }

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.65 ev3::SignalHandler Class Reference

Static Public Member Functions

- static void **HandleSignal** (int signum)

Static Public Attributes

- static Robot $*$ **robot** = nullptr
- static Master $*$ **master** = nullptr

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/utils/SignalHandler.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/utils/SignalHandler.cpp

4.66 ev3dev::sound Class Reference

Static Public Member Functions

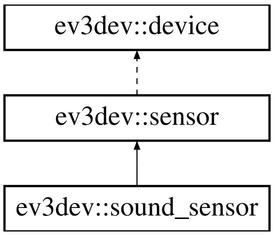
- static void **beep** (const std::string &args="", bool bSynchronous=false)
- static void **tone** (float frequency, float ms, bool bSynchronous=false)
- static void **tone** (const std::vector< std::vector< float > > &sequence, bool bSynchronous=false)
- static void **play** (const std::string &soundfile, bool bSynchronous=false)
- static void **speak** (const std::string &text, bool bSynchronous=false)

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.67 ev3dev::sound_sensor Class Reference

Inheritance diagram for ev3dev::sound_sensor:



Public Member Functions

- **sound_sensor** (address_type address=INPUT_AUTO)
- float **sound_pressure** ()
- float **sound_pressure_low** ()

Static Public Attributes

- static const std::string **mode_db** { "DB" }
- static const std::string **mode_dba** { "DBA" }

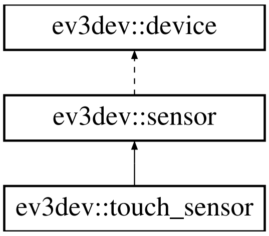
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.68 ev3dev::touch_sensor Class Reference

Inheritance diagram for ev3dev::touch_sensor:



Public Member Functions

- **touch_sensor** (address_type address=INPUT_AUTO)
- bool **is_pressed** ()

Static Public Attributes

- static const std::string **mode_touch** { "TOUCH" }

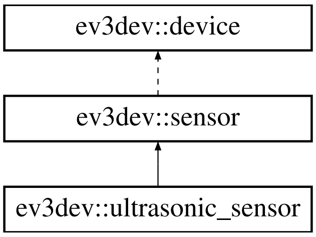
Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

4.69 ev3dev::ultrasonic_sensor Class Reference

Inheritance diagram for ev3dev::ultrasonic_sensor:



Public Member Functions

- **ultrasonic_sensor** (address_type address=INPUT_AUTO)
- float **distance_centimeters** ()
- float **distance_inches** ()
- bool **other_sensor_present** ()

Static Public Attributes

- static const std::string **mode_us_dist_cm** { "US-DIST-CM" }
- static const std::string **mode_us_dist_in** { "US-DIST-IN" }
- static const std::string **mode_us_listen** { "US-LISTEN" }
- static const std::string **mode_us_si_cm** { "US-SI-CM" }
- static const std::string **mode_us_si_in** { "US-SI-IN" }

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/panda/Dokumenty/Repos/Ev3Dev/include/ev3dev/ev3dev.h
- /home/panda/Dokumenty/Repos/Ev3Dev/src/ev3dev/ev3dev.cpp

Chapter 5

File Documentation

5.1 /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Action.h File Reference

Contains all Action classes.

```
#include "CommandMotor.h"
#include <memory>
```

Classes

- class ev3::Action

  Base class for all Action controlling classes.
- class ev3::ActionRepeat

  Stores many Actions in a vector and executes them in loop.
- class ev3::ActionDriveDistance

  Implements Robot simple task to drive straight for a given distance.
- class ev3::ActionRotate

  Implements Robot simple task to rotate a given angle, while not driving.
- class ev3::ActionRotateRandDirection

  Implements Robot simple task to rotate a random angle.
- class ev3::ActionStop

  Implements Robot simple task to stop all active motors.
- class ev3::ActionDriveForever

  Implements Robot simple task to drive straight forever.

Typedefs

- typedef std::shared_ptr< Action > ev3::SharedPtrAction

  Type for Action shared_ptr.
- typedef std::vector< SharedPtrAction > ev3::StoredActions

  Type for storing many Actions in one container.
- typedef std::shared_ptr< Command > ev3::SharedPtrCommand

  Type for Command shared_ptr.
- typedef std::vector< SharedPtrCommand > ev3::CommandsVector

  Type for containing associated Command pointers.

5.1.1 Detailed Description

Contains all Action classes.

5.1.2 Typedef Documentation

5.1.2.1 typedef std::vector<SharedPtrAction> ev3::StoredActions

Type for storing many Actions in one container.

See also

ActionRepeat

5.2 /home/panda/Dokumenty/Repos/Ev3Dev/include/action/Behaviour.h File Reference

Contains all Behaviour classes.

```
#include "Action.h"
#include "Utils.h"
#include "Sensor.h"
#include "Event.h"
#include "BehaviourState.h"
#include <unistd.h>
#include <string>
```

Classes

- class ev3::Behaviour

  Base class for all defined behaviours.
- class ev3::BehaviourDriveOnSquare
- class ev3::BehaviourExploreRandom

Typedefs

- typedef std::shared_ptr< Behaviour > ev3::SharedPtrBehaviour

  Type for Behaviour shared_ptr.
- typedef std::vector< BehaviourState > ev3::BehaviourStates

  Type for storing Behaviour states in one container.
- typedef std::vector< Sensor::SensorType > ev3::Measurements

  Type for storing sensors' desired measurements in one container.

5.2.1 Detailed Description

Contains all Behaviour classes.

5.3 /home/panda/Dokumenty/Repos/Ev3Dev/include/action/BehaviourState.h File Reference

Contains BehaviourState class.

```
#include "Action.h"
#include "Event.h"
```

Classes

- class ev3::BehaviourState

Typedefs

- typedef std::map< Event::EventType, unsigned int > **ev3::ReactionsTransitions**

5.3.1 Detailed Description

Contains BehaviourState class.

Index