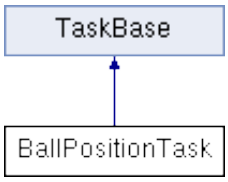
BallPositionTask Class Reference

Implements a task to determine the ball position. More...

`#include <BallPositionTask.h>`

Inheritance diagram for BallPositionTask:



Public Member Functions

	BallPositionTask (const char *a_name, unsigned portBASE_TYPE a_priority, size_t a_stack_size, **emstream** *p_ser_dev) Construct a BallPosition task. More...
void	**run** (void) The run method of the BallPosition task that is repeatedly called by the RTOS scheduler. More...

- **Public Member Functions inherited from TaskBase**

Additional Inherited Members

- **Static Public Member Functions inherited from TaskBase**
- **Protected Member Functions inherited from TaskBase**
- **Protected Attributes inherited from TaskBase**

Detailed Description

Implements a task to determine the ball position.

This class is an extension of `TaskBase`. The purpose of the class is to determine the physical states of the ball.

Constructor & Destructor Documentation

◆ BallPositionTask()

```
BallPositionTask::BallPositionTask ( const char *               a_name,
                                     unsigned portBASE_TYPE     a_priority,
                                     size_t                     a_stack_size,
                                     emstream *                 p_ser_dev
                                   )
```

Construct a BallPosition task.

Constructor which creates and initializes a BallPosition task object.

This constructor sets up the task name, priority, stack size, and serial stream.

Parameters

a_name	A character string which will be the name of this task
a_priority	The priority at which this task will initially run (default: 0)
a_stack_size	The size of this task's stack in bytes (default: `configMINIMAL_STACK_SIZE`)
p_ser_dev	Pointer to a serial device (port, radio, SD card, etc.) which can be used by this task to communicate (default: NULL)

This constructor creates a FreeRTOS task with the given task run function, name, priority, and stack size. Its purpose is to determine the ball position and velocity from the sensor measurements and update the appropriate shared variables.

Parameters

a_name	A character string which will be the name of this task
a_priority	The priority at which this task will initially run (default: 0)
a_stack_size	The size of this task's stack in bytes (default: `configMINIMAL_STACK_SIZE`)
p_ser_dev	Pointer to a serial device (port, radio, SD card, etc.) which can be used by this task to communicate (default: NULL)

Member Function Documentation

◆ run()

void BallPositionTask::run (void) virtual

The run method of the BallPosition task that is repeatedly called by the RTOS scheduler.

The `run()` function for the BallPosition task.

This method is called by the RTOS scheduler. The function converts the linear potentiometer measurements to ball position in m and ball velocity in m/s. Shared variables are updated after the calculations are performed.

Implements **TaskBase**.

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

- DoxygenFiles/**BallPositionTask.h**
- DoxygenFiles/BallPositionTask.cpp
