

# Major Project Documentation

## 0.2

Generated by Doxygen 1.8.6

Mon Mar 31 2014 21:39:11



# Contents

<b>1</b>	<b>README</b>	<b>1</b>
<b>2</b>	<b>Data Structure Index</b>	<b>3</b>
2.1	Data Structures . . . . .	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List . . . . .	5
<b>4</b>	<b>Data Structure Documentation</b>	<b>7</b>
4.1	odometryTrackStruct Struct Reference . . . . .	7
4.1.1	Field Documentation . . . . .	7
4.1.1.1	configuration . . . . .	7
4.1.1.2	pos_left_prev . . . . .	7
4.1.1.3	pos_right_prev . . . . .	7
4.1.1.4	result . . . . .	7
4.1.1.5	state . . . . .	7
4.1.1.6	theta . . . . .	7
4.1.1.7	wheel_conversion . . . . .	7
4.1.1.8	wheel_distance . . . . .	8
4.1.1.9	x . . . . .	8
4.1.1.10	y . . . . .	8
<b>5</b>	<b>File Documentation</b>	<b>9</b>
5.1	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/e_puck_movement/build/release/e_puck_movement.d File Reference . . . . .	9
5.2	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/build/release/e_puck_movement.d File Reference . . . . .	9
5.3	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/mid_term_demo/build/release/e_puck_movement.d File Reference . . . . .	9
5.4	C:/Users/Stefan/Documents/GitHub/Final_Project/lib/build/release/e_puck_movement.d File Reference . . . . .	9
5.5	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/epuck_collision_avoidance/build/release/epuck_collision_avoidance.d File Reference . . . . .	9
5.6	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/epuck_collision_avoidance/epuck_collision_avoidance.c File Reference . . . . .	9

5.6.1	Macro Definition Documentation . . . . .	10
5.6.1.1	TIME_STEP . . . . .	10
5.6.2	Function Documentation . . . . .	10
5.6.2.1	main . . . . .	10
5.7	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/epuck_scanline/build/release/epuck_scanline.d File Reference . . . . .	10
5.8	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/epuck_scanline/epuck_scanline.c File Reference . . . . .	10
5.8.1	Macro Definition Documentation . . . . .	11
5.8.1.1	AXLE_LENGTH . . . . .	11
5.8.1.2	ENCODER_RESOLUTION . . . . .	11
5.8.1.3	FORWARD . . . . .	11
5.8.1.4	LEFT . . . . .	11
5.8.1.5	NB_DIST_SENS . . . . .	11
5.8.1.6	PS_LEFT_10 . . . . .	11
5.8.1.7	PS_LEFT_45 . . . . .	11
5.8.1.8	PS_LEFT_90 . . . . .	11
5.8.1.9	PS_LEFT_REAR . . . . .	11
5.8.1.10	PS_RIGHT_10 . . . . .	11
5.8.1.11	PS_RIGHT_45 . . . . .	11
5.8.1.12	PS_RIGHT_90 . . . . .	11
5.8.1.13	PS_RIGHT_REAR . . . . .	11
5.8.1.14	REALITY . . . . .	11
5.8.1.15	RIGHT . . . . .	11
5.8.1.16	SIMULATION . . . . .	11
5.8.1.17	STOP . . . . .	11
5.8.1.18	THRESHOLD_DIST . . . . .	11
5.8.1.19	TIME_STEP . . . . .	11
5.8.1.20	TURNLEFT . . . . .	11
5.8.1.21	TURNRIGHT . . . . .	11
5.8.1.22	UTURN . . . . .	12
5.8.1.23	WHEEL_RADIUS . . . . .	12
5.8.2	Function Documentation . . . . .	12
5.8.2.1	main . . . . .	12
5.8.3	Variable Documentation . . . . .	12
5.8.3.1	n . . . . .	12
5.8.3.2	obstacle . . . . .	12
5.8.3.3	old_encoder . . . . .	12
5.8.3.4	ps . . . . .	12
5.8.3.5	ps_offset_real . . . . .	12

5.8.3.6	ps_offset_sim . . . . .	12
5.8.3.7	ps_value . . . . .	12
5.8.3.8	speed . . . . .	12
5.8.3.9	state . . . . .	12
5.9	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/build/release/Main.d File Reference . . . . .	12
5.10	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/build/release/map_building.d File Reference . . . . .	12
5.11	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/mid_term_demo/build/release/map_building.d File Reference . . . . .	12
5.12	C:/Users/Stefan/Documents/GitHub/Final_Project/lib/build/release/map_building.d File Reference . . . . .	12
5.13	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/build/release/odometry.d File Reference . . . . .	12
5.14	C:/Users/Stefan/Documents/GitHub/Final_Project/lib/build/release/odometry.d File Reference . . . . .	12
5.15	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/Main.c File Reference . . . . .	13
5.15.1	Macro Definition Documentation . . . . .	13
5.15.1.1	LEFT . . . . .	13
5.15.1.2	MAP_SIZE . . . . .	13
5.15.1.3	NUM_DIST_SENS . . . . .	13
5.15.1.4	OCCUPANCE_DIST . . . . .	14
5.15.1.5	PS_LEFT_10 . . . . .	14
5.15.1.6	PS_LEFT_45 . . . . .	14
5.15.1.7	PS_LEFT_90 . . . . .	14
5.15.1.8	PS_LEFT_REAR . . . . .	14
5.15.1.9	PS_RIGHT_10 . . . . .	14
5.15.1.10	PS_RIGHT_45 . . . . .	14
5.15.1.11	PS_RIGHT_90 . . . . .	14
5.15.1.12	PS_RIGHT_REAR . . . . .	14
5.15.1.13	RIGHT . . . . .	14
5.15.1.14	THRESHOLD_DIST . . . . .	14
5.15.1.15	TIME_STEP . . . . .	14
5.15.2	Function Documentation . . . . .	14
5.15.2.1	main . . . . .	14
5.15.3	Variable Documentation . . . . .	14
5.15.3.1	led . . . . .	14
5.15.3.2	ot . . . . .	14
5.16	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/mid_term_demo/build/release/mid_term_demo.d File Reference . . . . .	14
5.17	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/mid_term_demo/mid_term_demo.c File Reference . . . . .	14
5.17.1	Macro Definition Documentation . . . . .	15
5.17.1.1	TIME_STEP . . . . .	15

5.17.2	Function Documentation . . . . .	15
5.17.2.1	main . . . . .	15
5.18	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/my_controller_epuk_go_forward/build/release/my_controller_epuk_go_forward.d File Reference . . . . .	15
5.19	C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/my_controller_epuk_go_forward/my_controller_epuk_go_forward.c File Reference . . . . .	15
5.19.1	Macro Definition Documentation . . . . .	15
5.19.1.1	TIME_STEP . . . . .	15
5.19.2	Function Documentation . . . . .	15
5.19.2.1	main . . . . .	15
5.20	C:/Users/Stefan/Documents/GitHub/Final_Project/epuck'_movement_download.c File Reference . . . . .	15
5.20.1	Macro Definition Documentation . . . . .	16
5.20.1.1	ENCODER_RESOLUTION . . . . .	16
5.20.1.2	M_PI . . . . .	16
5.20.1.3	MIN_DIST . . . . .	16
5.20.1.4	MIN_SPEED . . . . .	16
5.20.1.5	NTOURNMENTS . . . . .	16
5.20.1.6	RANGE . . . . .	16
5.20.1.7	REV_STEP . . . . .	16
5.20.1.8	STEP_TOLERANCE . . . . .	16
5.20.1.9	TIME_STEP . . . . .	16
5.20.1.10	WHEEL_RADIUS . . . . .	16
5.20.1.11	WHEELBASE . . . . .	16
5.20.2	Function Documentation . . . . .	16
5.20.2.1	main . . . . .	16
5.20.3	Variable Documentation . . . . .	16
5.20.3.1	dMSpeed . . . . .	16
5.20.3.2	dPrevEncPos . . . . .	17
5.20.3.3	led . . . . .	17
5.20.3.4	p_dOdometryData . . . . .	17
5.21	C:/Users/Stefan/Documents/GitHub/Final_Project/lib/e_puck_movement.c File Reference . . . . .	17
5.21.1	Macro Definition Documentation . . . . .	18
5.21.1.1	ANGLE_TOLERANCE . . . . .	18
5.21.1.2	EAST . . . . .	18
5.21.1.3	ENCODER_RESOLUTION . . . . .	18
5.21.1.4	INCREMENT_STEP . . . . .	18
5.21.1.5	M_PI . . . . .	18
5.21.1.6	MIN_DIST . . . . .	18
5.21.1.7	MIN_SPEED . . . . .	18
5.21.1.8	NORTH . . . . .	18

5.21.1.9	NUMTOURNAMENTS . . . . .	18
5.21.1.10	RANGE . . . . .	18
5.21.1.11	RTOD . . . . .	18
5.21.1.12	SOUTH . . . . .	18
5.21.1.13	STEP_TOLERANCE . . . . .	18
5.21.1.14	TIME_STEP . . . . .	18
5.21.1.15	WEST . . . . .	18
5.21.1.16	WHEEL_RADIUS . . . . .	18
5.21.1.17	WHEELBASE . . . . .	18
5.21.2	Function Documentation . . . . .	18
5.21.2.1	check_rotation . . . . .	18
5.21.2.2	compute_odometry_data . . . . .	19
5.21.2.3	get_encoder_positions . . . . .	19
5.21.2.4	measure_clockWise . . . . .	19
5.21.2.5	measure_CounterClockWise . . . . .	19
5.21.2.6	move_forward . . . . .	19
5.21.2.7	set_leds . . . . .	19
5.21.2.8	set_motor_speed . . . . .	19
5.21.2.9	stop_robot . . . . .	19
5.21.2.10	turn_angle . . . . .	19
5.21.2.11	turn_left . . . . .	19
5.21.2.12	turn_right . . . . .	19
5.21.2.13	UMBmark . . . . .	19
5.21.3	Variable Documentation . . . . .	20
5.21.3.1	dCurSpeed . . . . .	20
5.21.3.2	dPrevEncPos . . . . .	20
5.21.3.3	led . . . . .	20
5.21.3.4	point_dOdometryData . . . . .	20
5.21.3.5	weights_left . . . . .	20
5.21.3.6	weights_right . . . . .	20
5.22	C:/Users/Stefan/Documents/GitHub/Final_Project/lib/e_puck_movement.h File Reference . . . . .	20
5.22.1	Function Documentation . . . . .	20
5.22.1.1	check_rotation . . . . .	20
5.22.1.2	compute_odometry_data . . . . .	20
5.22.1.3	controll_angle . . . . .	20
5.22.1.4	get_encoder_positions . . . . .	20
5.22.1.5	measure_clockWise . . . . .	21
5.22.1.6	measure_CounterClockWise . . . . .	21
5.22.1.7	move_forward . . . . .	21
5.22.1.8	set_leds . . . . .	21

5.22.1.9	set_motor_speed . . . . .	21
5.22.1.10	stop_robot . . . . .	21
5.22.1.11	turn_angle . . . . .	21
5.22.1.12	turn_left . . . . .	21
5.22.1.13	turn_right . . . . .	21
5.22.1.14	UMBmark . . . . .	21
5.23	C:/Users/Stefan/Documents/GitHub/Final_Project/lib/map_building.c File Reference . . . . .	22
5.23.1	Macro Definition Documentation . . . . .	23
5.23.1.1	ANGLE_TOLERANCE . . . . .	23
5.23.1.2	CELL_SIZE . . . . .	23
5.23.1.3	EAST . . . . .	23
5.23.1.4	FORWARD . . . . .	23
5.23.1.5	LEFT . . . . .	23
5.23.1.6	MAP_SIZE . . . . .	23
5.23.1.7	NORTH . . . . .	23
5.23.1.8	NUM_DIST_SENS . . . . .	23
5.23.1.9	OCCUPANCE_DIST . . . . .	23
5.23.1.10	PS_LEFT_10 . . . . .	23
5.23.1.11	PS_LEFT_45 . . . . .	23
5.23.1.12	PS_LEFT_90 . . . . .	23
5.23.1.13	PS_LEFT_REAR . . . . .	23
5.23.1.14	PS_RIGHT_10 . . . . .	23
5.23.1.15	PS_RIGHT_45 . . . . .	23
5.23.1.16	PS_RIGHT_90 . . . . .	24
5.23.1.17	PS_RIGHT_REAR . . . . .	24
5.23.1.18	RIGHT . . . . .	24
5.23.1.19	RTOD . . . . .	24
5.23.1.20	SOUTH . . . . .	24
5.23.1.21	STOP . . . . .	24
5.23.1.22	THRESHOLD_DIST . . . . .	24
5.23.1.23	TIME_STEP . . . . .	24
5.23.1.24	TURNLEFT . . . . .	24
5.23.1.25	TURNRIGHT . . . . .	24
5.23.1.26	UTURN . . . . .	24
5.23.1.27	WEST . . . . .	24
5.23.2	Function Documentation . . . . .	24
5.23.2.1	check_direction . . . . .	24
5.23.2.2	init_display . . . . .	24
5.23.2.3	occupied_cell . . . . .	24
5.23.2.4	reset . . . . .	24



5.23.2.5	return_angle . . . . .	24
5.23.2.6	return_sensor_values . . . . .	24
5.23.2.7	run . . . . .	24
5.23.3	Variable Documentation . . . . .	25
5.23.3.1	angle_offset . . . . .	25
5.23.3.2	background . . . . .	25
5.23.3.3	display . . . . .	25
5.23.3.4	display_height . . . . .	25
5.23.3.5	display_width . . . . .	25
5.23.3.6	east . . . . .	25
5.23.3.7	led . . . . .	25
5.23.3.8	map . . . . .	25
5.23.3.9	new_encoder . . . . .	25
5.23.3.10	north . . . . .	25
5.23.3.11	ob_front . . . . .	25
5.23.3.12	ob_left . . . . .	25
5.23.3.13	ob_right . . . . .	25
5.23.3.14	obstacle . . . . .	25
5.23.3.15	ps . . . . .	25
5.23.3.16	ps_value . . . . .	25
5.23.3.17	robot_x . . . . .	25
5.23.3.18	robot_y . . . . .	25
5.23.3.19	south . . . . .	25
5.23.3.20	state . . . . .	25
5.23.3.21	west . . . . .	25
5.24	C:/Users/Stefan/Documents/GitHub/Final_Project/map_building.c File Reference . . . . .	25
5.24.1	Macro Definition Documentation . . . . .	27
5.24.1.1	CELL_SIZE . . . . .	27
5.24.1.2	FORWARD . . . . .	27
5.24.1.3	LEFT . . . . .	27
5.24.1.4	MAP_SIZE . . . . .	27
5.24.1.5	NUM_DIST_SENS . . . . .	27
5.24.1.6	OCCUPANCE_DIST . . . . .	27
5.24.1.7	PS_LEFT_10 . . . . .	27
5.24.1.8	PS_LEFT_45 . . . . .	27
5.24.1.9	PS_LEFT_90 . . . . .	27
5.24.1.10	PS_LEFT_REAR . . . . .	27
5.24.1.11	PS_RIGHT_10 . . . . .	27
5.24.1.12	PS_RIGHT_45 . . . . .	27
5.24.1.13	PS_RIGHT_90 . . . . .	27

5.24.1.14 PS_RIGHT_REAR . . . . .	27
5.24.1.15 RIGHT . . . . .	27
5.24.1.16 THRESHOLD_DIST . . . . .	27
5.24.1.17 TIME_STEP . . . . .	27
5.24.1.18 TURNLEFT . . . . .	27
5.24.1.19 TURNRIGHT . . . . .	27
5.24.1.20 UTURN . . . . .	27
5.24.2 Function Documentation . . . . .	27
5.24.2.1 init_display . . . . .	27
5.24.2.2 occupied_cell . . . . .	27
5.24.2.3 reset . . . . .	27
5.24.2.4 return_sensor_values . . . . .	28
5.24.2.5 run . . . . .	28
5.24.3 Variable Documentation . . . . .	28
5.24.3.1 angle_offset . . . . .	28
5.24.3.2 background . . . . .	28
5.24.3.3 display . . . . .	28
5.24.3.4 display_height . . . . .	28
5.24.3.5 display_width . . . . .	28
5.24.3.6 led . . . . .	28
5.24.3.7 map . . . . .	28
5.24.3.8 new_encoder . . . . .	28
5.24.3.9 ob_front . . . . .	28
5.24.3.10 ob_left . . . . .	28
5.24.3.11 ob_right . . . . .	28
5.24.3.12 obstacle . . . . .	28
5.24.3.13 ps . . . . .	28
5.24.3.14 ps_value . . . . .	28
5.24.3.15 robot_x . . . . .	28
5.24.3.16 robot_y . . . . .	28
5.24.3.17 state . . . . .	28
5.25 C:/Users/Stefan/Documents/GitHub/Final_Project/lib/map_building.h File Reference . . . . .	28
5.25.1 Function Documentation . . . . .	29
5.25.1.1 init_display . . . . .	29
5.25.1.2 occupied_cell . . . . .	29
5.25.1.3 reset . . . . .	29
5.25.1.4 return_angle . . . . .	29
5.25.1.5 return_sensor_values . . . . .	29
5.25.1.6 run . . . . .	29
5.26 C:/Users/Stefan/Documents/GitHub/Final_Project/lib/odometry.c File Reference . . . . .	29

5.26.1	Macro Definition Documentation . . . . .	30
5.26.1.1	M_PI . . . . .	30
5.26.2	Function Documentation . . . . .	30
5.26.2.1	odometry_track_start . . . . .	30
5.26.2.2	odometry_track_start_pos . . . . .	30
5.26.2.3	odometry_track_step . . . . .	30
5.26.2.4	odometry_track_step_pos . . . . .	30
5.26.3	Variable Documentation . . . . .	30
5.26.3.1	axis_wheel_ratio . . . . .	30
5.26.3.2	increments_per_tour . . . . .	30
5.26.3.3	scaling_factor . . . . .	30
5.26.3.4	wheel_radius . . . . .	30
5.27	C:/Users/Stefan/Documents/GitHub/Final_Project/odometry.c File Reference . . . . .	30
5.27.1	Macro Definition Documentation . . . . .	31
5.27.1.1	M_PI . . . . .	31
5.27.2	Function Documentation . . . . .	31
5.27.2.1	odometry_track_start . . . . .	31
5.27.2.2	odometry_track_start_pos . . . . .	31
5.27.2.3	odometry_track_step . . . . .	31
5.27.2.4	odometry_track_step_pos . . . . .	31
5.27.3	Variable Documentation . . . . .	31
5.27.3.1	axis_wheel_ratio . . . . .	31
5.27.3.2	increments_per_tour . . . . .	31
5.27.3.3	scaling_factor . . . . .	31
5.27.3.4	wheel_radius . . . . .	31
5.28	C:/Users/Stefan/Documents/GitHub/Final_Project/lib/odometry.h File Reference . . . . .	31
5.28.1	Function Documentation . . . . .	31
5.28.1.1	odometry_track_start . . . . .	31
5.28.1.2	odometry_track_start_pos . . . . .	32
5.28.1.3	odometry_track_step . . . . .	32
5.28.1.4	odometry_track_step_pos . . . . .	32
5.29	C:/Users/Stefan/Documents/GitHub/Final_Project/README.md File Reference . . . . .	32
5.30	C:/Users/Stefan/Documents/GitHub/Final_Project/scanline_backup.c File Reference . . . . .	32
5.30.1	Macro Definition Documentation . . . . .	33
5.30.1.1	AXLE_LENGTH . . . . .	33
5.30.1.2	ENCODER_RESOLUTION . . . . .	33
5.30.1.3	FORWARD . . . . .	33
5.30.1.4	LEFT . . . . .	33
5.30.1.5	NUM_DIST_SENS . . . . .	33
5.30.1.6	PS_LEFT_10 . . . . .	33

5.30.1.7	PS_LEFT_45 . . . . .	33
5.30.1.8	PS_LEFT_90 . . . . .	33
5.30.1.9	PS_LEFT_REAR . . . . .	33
5.30.1.10	PS_RIGHT_10 . . . . .	33
5.30.1.11	PS_RIGHT_45 . . . . .	33
5.30.1.12	PS_RIGHT_90 . . . . .	33
5.30.1.13	PS_RIGHT_REAR . . . . .	33
5.30.1.14	RIGHT . . . . .	33
5.30.1.15	STOP . . . . .	33
5.30.1.16	THRESHOLD_DIST . . . . .	33
5.30.1.17	TIME_STEP . . . . .	33
5.30.1.18	TURNLEFT . . . . .	33
5.30.1.19	TURNRIGHT . . . . .	33
5.30.1.20	UTURN . . . . .	33
5.30.1.21	WHEEL_RADIUS . . . . .	33
5.30.2	Function Documentation . . . . .	33
5.30.2.1	fsm . . . . .	34
5.30.2.2	main . . . . .	34
5.30.3	Variable Documentation . . . . .	34
5.30.3.1	fsm_speed . . . . .	34
5.30.3.2	obstacle . . . . .	34
5.30.3.3	ps . . . . .	34
5.30.3.4	ps_offset . . . . .	34
5.30.3.5	ps_value . . . . .	34
5.30.3.6	speed . . . . .	34
5.30.3.7	state . . . . .	34

## **Chapter 1**

# **README**

Mayor Project Documentation



## Chapter 2

# Data Structure Index

### 2.1 Data Structures

Here are the data structures with brief descriptions:

<b>odometryTrackStruct</b> . . . . .	7
--------------------------------------	---





## Chapter 3

# File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

C:/Users/Stefan/Documents/GitHub/Final_Project/epuck <sup>4</sup> _movement_download.c . . . . .	15
C:/Users/Stefan/Documents/GitHub/Final_Project/map_building.c . . . . .	25
C:/Users/Stefan/Documents/GitHub/Final_Project/odometry.c . . . . .	30
C:/Users/Stefan/Documents/GitHub/Final_Project/scanline_backup.c . . . . .	32
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/e_puck_movement/build/release/e_puck_ movement.d . . . . .	9
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/epuck_collision_avoidance/epuck_collision_ avoidance.c . . . . .	9
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/epuck_collision_avoidance/build/release/epuck_ collision_avoidance.d . . . . .	9
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/epuck_scanline/epuck_scanline.c . . . .	10
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/epuck_scanline/build/release/epuck_ scanline.d . . . . .	10
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/Main.c . . . . .	13
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/build/release/e_puck_movement.d .	9
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/build/release/Main.d . . . . .	12
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/build/release/map_building.d . . . .	12
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/Main/build/release/odometry.d . . . . .	12
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/mid_term_demo/mid_term_demo.c . . . .	14
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/mid_term_demo/build/release/e_puck_ movement.d . . . . .	9
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/mid_term_demo/build/release/map_ building.d . . . . .	12
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/mid_term_demo/build/release/mid_term_ demo.d . . . . .	14
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/my_controller_epuk_go_forward/my_ controller_epuk_go_forward.c . . . . .	15
C:/Users/Stefan/Documents/GitHub/Final_Project/controllers/my_controller_epuk_go_forward/build/release/my_ controller_epuk_go_forward.d . . . . .	15
C:/Users/Stefan/Documents/GitHub/Final_Project/lib/e_puck_movement.c . . . . .	17
C:/Users/Stefan/Documents/GitHub/Final_Project/lib/e_puck_movement.h . . . . .	20
C:/Users/Stefan/Documents/GitHub/Final_Project/lib/map_building.c . . . . .	22
C:/Users/Stefan/Documents/GitHub/Final_Project/lib/map_building.h . . . . .	28
C:/Users/Stefan/Documents/GitHub/Final_Project/lib/odometry.c . . . . .	29
C:/Users/Stefan/Documents/GitHub/Final_Project/lib/odometry.h . . . . .	31
C:/Users/Stefan/Documents/GitHub/Final_Project/lib/build/release/e_puck_movement.d . . . . .	9
C:/Users/Stefan/Documents/GitHub/Final_Project/lib/build/release/map_building.d . . . . .	12
C:/Users/Stefan/Documents/GitHub/Final_Project/lib/build/release/odometry.d . . . . .	12



## Chapter 4

# Data Structure Documentation

### 4.1 odometryTrackStruct Struct Reference

```
#include <odometry.h>
```

#### Data Fields

- struct {  
    float **wheel\_distance**  
    float **wheel\_conversion**  
} **configuration**
- struct {  
    int **pos\_left\_prev**  
    int **pos\_right\_prev**  
} **state**
- struct {  
    float **x**  
    float **y**  
    float **theta**  
} **result**

#### 4.1.1 Field Documentation

4.1.1.1 struct { ... } configuration

4.1.1.2 int pos\_left\_prev

4.1.1.3 int pos\_right\_prev

4.1.1.4 struct { ... } result

4.1.1.5 struct { ... } state

4.1.1.6 float theta

4.1.1.7 float wheel\_conversion

4.1.1.8 float wheel\_distance

4.1.1.9 float x

4.1.1.10 float y

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

- C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/**odometry.h**

## Chapter 5

# File Documentation

5.1 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/e\_puck\_movement/build/release/e\_puck\_movement.d File Reference

5.2 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/Main/build/release/e\_puck\_movement.d File Reference

5.3 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/mid\_term\_demo/build/release/e\_puck\_movement.d File Reference

5.4 C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/build/release/e\_puck\_movement.d File Reference

5.5 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/epuck\_collision\_avoidance/build/release/epuck\_collision\_avoidance.d File Reference

5.6 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/epuck\_collision\_avoidance/epuck\_collision\_avoidance.c File Reference

```
#include <webots/robot.h>
#include <webots/differential_wheels.h>
#include <webots/distance_sensor.h>
```

### Macros

- `#define TIME_STEP 64`

### Functions

- `int main (int argc, char **argv)`

## 5.6.1 Macro Definition Documentation

### 5.6.1.1 #define TIME\_STEP 64

## 5.6.2 Function Documentation

### 5.6.2.1 int main ( int argc, char \*\* argv )

## 5.7 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/epuck\_scanline/build/release/epuck-\_scanline.d File Reference

## 5.8 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/epuck\_scanline/epuck-\_scanline.c File Reference

```
#include <webots/robot.h>
#include <webots/distance_sensor.h>
#include <webots/differential_wheels.h>
#include <stdlib.h>
#include <time.h>
#include <stdio.h>
```

## Macros

- #define **THRESHOLD\_DIST** 300
- #define **TIME\_STEP** 32
- #define **SIMULATION** 0
- #define **REALITY** 2
- #define **LEFT** 0
- #define **RIGHT** 1
- #define **NB\_DIST\_SENS** 8
- #define **PS\_RIGHT\_10** 0
- #define **PS\_RIGHT\_45** 1
- #define **PS\_RIGHT\_90** 2
- #define **PS\_RIGHT\_REAR** 3
- #define **PS\_LEFT\_REAR** 4
- #define **PS\_LEFT\_90** 5
- #define **PS\_LEFT\_45** 6
- #define **PS\_LEFT\_10** 7
- #define **WHEEL\_RADIUS** 0.02
- #define **AXLE\_LENGTH** 0.026
- #define **ENCODER\_RESOLUTION** 159.23
- #define **FORWARD** 0
- #define **STOP** 1
- #define **UTURN** 2
- #define **TURNRIGHT** 3
- #define **TURNLEFT** 4

## Functions

- int **main** ()

## Variables

- WbDeviceTag **ps** [NB\_DIST\_SENS]
- int **ps\_value** [NB\_DIST\_SENS] = {0,0,0,0,0,0,0,0}
- int **ps\_offset\_sim** [NB\_DIST\_SENS] = {35,35,35,35,35,35,35,35}
- int **ps\_offset\_real** [NB\_DIST\_SENS] = {375,158,423,682,447,594,142,360}
- int **obstacle** [NB\_DIST\_SENS]
- int **speed** [2] = {0,0}
- int **state** = FORWARD
- int **n** =0
- int **old\_encoder** =0

## 5.8.1 Macro Definition Documentation

5.8.1.1 #define AXLE\_LENGTH 0.026

5.8.1.2 #define ENCODER\_RESOLUTION 159.23

5.8.1.3 #define FORWARD 0

5.8.1.4 #define LEFT 0

5.8.1.5 #define NB\_DIST\_SENS 8

5.8.1.6 #define PS\_LEFT\_10 7

5.8.1.7 #define PS\_LEFT\_45 6

5.8.1.8 #define PS\_LEFT\_90 5

5.8.1.9 #define PS\_LEFT\_REAR 4

5.8.1.10 #define PS\_RIGHT\_10 0

5.8.1.11 #define PS\_RIGHT\_45 1

5.8.1.12 #define PS\_RIGHT\_90 2

5.8.1.13 #define PS\_RIGHT\_REAR 3

5.8.1.14 #define REALITY 2

5.8.1.15 #define RIGHT 1

5.8.1.16 #define SIMULATION 0

5.8.1.17 #define STOP 1

5.8.1.18 #define THRESHOLD\_DIST 300

5.8.1.19 #define TIME\_STEP 32

5.8.1.20 #define TURNLEFT 4

5.8.1.21 #define TURNRIGHT 3

5.8.1.22 `#define UTURN 2`

5.8.1.23 `#define WHEEL_RADIUS 0.02`

## 5.8.2 Function Documentation

5.8.2.1 `int main ( )`

## 5.8.3 Variable Documentation

5.8.3.1 `int n =0`

5.8.3.2 `int obstacle[NB_DIST_SENS]`

5.8.3.3 `int old_encoder =0`

5.8.3.4 `WbDeviceTag ps`

5.8.3.5 `int ps_offset_real[NB_DIST_SENS] = {375,158,423,682,447,594,142,360}`

5.8.3.6 `int ps_offset_sim[NB_DIST_SENS] = {35,35,35,35,35,35,35,35}`

5.8.3.7 `int ps_value[NB_DIST_SENS] ={0,0,0,0,0,0,0,0}`

5.8.3.8 `int speed[2] = {0,0}`

5.8.3.9 `int state = FORWARD`

## 5.9 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/Main/build/release/-Main.d File Reference

### 5.10 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/Main/build/release/map\_building.d File Reference

### 5.11 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/mid\_term\_demo/build/release/map\_building.d File Reference

### 5.12 C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/build/release/map\_building.d File Reference

### 5.13 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/Main/build/release/odometry.d File Reference

### 5.14 C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/build/release/odometry.d File Reference



## 5.15 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/Main/Main.c File Reference

```
#include <webots/robot.h>
#include <webots/differential_wheels.h>
#include <webots/distance_sensor.h>
#include <webots/light_sensor.h>
#include <webots/camera.h>
#include <webots/display.h>
#include <webots/accelerometer.h>
#include <webots/led.h>
#include <math.h>
#include "../lib/odometry.h"
#include "../lib/e_puck_movement.h"
#include "../lib/map_building.h"
```

### Macros

- `#define TIME_STEP 8`
- `#define MAP_SIZE 70`
- `#define THRESHOLD_DIST 100`
- `#define OCCUPANCE_DIST 150`
- `#define LEFT 0`
- `#define RIGHT 1`
- `#define NUM_DIST_SENS 8`
- `#define PS_RIGHT_10 0`
- `#define PS_RIGHT_45 1`
- `#define PS_RIGHT_90 2`
- `#define PS_RIGHT_REAR 3`
- `#define PS_LEFT_REAR 4`
- `#define PS_LEFT_90 5`
- `#define PS_LEFT_45 6`
- `#define PS_LEFT_10 7`

### Functions

- `int main (int argc, char **argv)`

### Variables

- WbDeviceTag `led` [3]
- struct `odometryTrackStruct` `ot`

#### 5.15.1 Macro Definition Documentation

##### 5.15.1.1 `#define LEFT 0`

##### 5.15.1.2 `#define MAP_SIZE 70`

##### 5.15.1.3 `#define NUM_DIST_SENS 8`

5.15.1.4 `#define OCCUPANCE_DIST 150`

5.15.1.5 `#define PS_LEFT_10 7`

5.15.1.6 `#define PS_LEFT_45 6`

5.15.1.7 `#define PS_LEFT_90 5`

5.15.1.8 `#define PS_LEFT_REAR 4`

5.15.1.9 `#define PS_RIGHT_10 0`

5.15.1.10 `#define PS_RIGHT_45 1`

5.15.1.11 `#define PS_RIGHT_90 2`

5.15.1.12 `#define PS_RIGHT_REAR 3`

5.15.1.13 `#define RIGHT 1`

5.15.1.14 `#define THRESHOLD_DIST 100`

5.15.1.15 `#define TIME_STEP 8`

## 5.15.2 Function Documentation

5.15.2.1 `int main ( int argc, char ** argv )`

## 5.15.3 Variable Documentation

5.15.3.1 `WbDeviceTag led[3]`

5.15.3.2 `struct odometryTrackStruct ot`

## 5.16 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/mid\_term\_demo/build/release/mid-term\_demo.d File Reference

## 5.17 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/mid\_term\_demo/mid-term\_demo.c File Reference

```
#include <webots/robot.h>
#include "../lib/map_building.h"
#include "../lib/e_puck_movement.h"
```

## Macros

- `#define TIME_STEP 8`

## Functions

- `int main (int argc, char **argv)`

#### 5.17.1 Macro Definition Documentation

5.17.1.1 `#define TIME_STEP 8`

#### 5.17.2 Function Documentation

5.17.2.1 `int main ( int argc, char ** argv )`

### 5.18 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/my\_controller\_epuk\_go\_forward/build/release/my\_controller\_epuk\_go\_forward.d File Reference

### 5.19 C:/Users/Stefan/Documents/GitHub/Final\_Project/controllers/my\_controller\_epuk\_go\_forward/my\_controller\_epuk\_go\_forward.c File Reference

```
#include <webots/robot.h>
#include <webots/differential_wheels.h>
```

#### Macros

- `#define TIME_STEP 64`

#### Functions

- `int main (int argc, char **argv)`

#### 5.19.1 Macro Definition Documentation

5.19.1.1 `#define TIME_STEP 64`

#### 5.19.2 Function Documentation

5.19.2.1 `int main ( int argc, char ** argv )`

### 5.20 C:/Users/Stefan/Documents/GitHub/Final\_Project/epuck\_movement\_download.c File Reference

```
#include <webots/robot.h>
#include <webots/differential_wheels.h>
#include <webots/distance_sensor.h>
#include <webots/light_sensor.h>
#include <webots/camera.h>
#include <webots/accelerometer.h>
#include <webots/led.h>
#include <math.h>
```

#### Macros

- `#define M_PI 3.1415926535897932384626433832795L`

- `#define TIME_STEP 8`
- `#define WHEEL_RADIUS 0.0206625`
- `#define WHEELBASE 0.052`
- `#define ENCODER_RESOLUTION 159.23`
- `#define REV_STEP 1000`
- `#define STEP_TOLERANCE 6.0`
- `#define RANGE (1024 / 2)`
- `#define MIN_DIST 20.0f`
- `#define MIN_SPEED 10.0f`
- `#define NTOURNAMENTS 1`

## Functions

- `int main (int argc, char *argv[])`

## Variables

- `double dMSpeed [2] = {0.0f, 0.0f}`
- `double dPrevEncPos [2] = {0.0f, 0.0f}`
- `double * p_dOdometryData`
- `WbDeviceTag led [3]`

### 5.20.1 Macro Definition Documentation

5.20.1.1 `#define ENCODER_RESOLUTION 159.23`

5.20.1.2 `#define M_PI 3.1415926535897932384626433832795L`

5.20.1.3 `#define MIN_DIST 20.0f`

5.20.1.4 `#define MIN_SPEED 10.0f`

5.20.1.5 `#define NTOURNAMENTS 1`

5.20.1.6 `#define RANGE (1024 / 2)`

5.20.1.7 `#define REV_STEP 1000`

5.20.1.8 `#define STEP_TOLERANCE 6.0`

5.20.1.9 `#define TIME_STEP 8`

5.20.1.10 `#define WHEEL_RADIUS 0.0206625`

5.20.1.11 `#define WHEELBASE 0.052`

### 5.20.2 Function Documentation

5.20.2.1 `int main ( int argc, char * argv[] )`

### 5.20.3 Variable Documentation

5.20.3.1 `double dMSpeed[2] = {0.0f, 0.0f}`

5.20.3.2 `double dPrevEncPos[2] = {0.0f, 0.0f}`

5.20.3.3 `WbDeviceTag led[3]`

5.20.3.4 `double* p_dOdometryData`

## 5.21 C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/e\_puck\_movement.c File Reference

```
#include <webots/robot.h>
#include <webots/differential_wheels.h>
#include <webots/distance_sensor.h>
#include <webots/light_sensor.h>
#include <webots/camera.h>
#include <webots/accelerometer.h>
#include <webots/led.h>
#include <math.h>
#include "e_puck_movement.h"
#include "map_building.h"
#include "odometry.h"
```

### Macros

- `#define M_PI 3.1415926535897932384626433832795L`
- `#define TIME_STEP 8`
- `#define WHEEL_RADIUS 0.0206625`
- `#define WHEELBASE 0.052`
- `#define ENCODER_RESOLUTION 159.23`
- `#define INCREMENT_STEP 1000`
- `#define STEP_TOLERANCE 6.0`
- `#define RANGE (1024 / 2)`
- `#define MIN_DIST 20.0f`
- `#define MIN_SPEED 10.0f`
- `#define NUMTOURNAMENTS 1`
- `#define RTOD(r) ((r) * 180 / M_PI)`
- `#define ANGLE_TOLERANCE 5`
- `#define EAST 0`
- `#define NORTH 90`
- `#define WEST 180`
- `#define SOUTH 270`

### Functions

- `void stop_robot ()`
- `void move_forward (double dSpeed, double dDist)`
- `void turn_left (double dSpeed)`
- `void turn_right (double dSpeed)`
- `void turn_angle (double dAngle, double dSpeed)`
- `void set_motor_speed (double dSpeedL, double dSpeedR)`
- `double * get_encoder_positions ()`
- `void check_rotation (double cur_rot, double want_rot, double dSpeed)`
- `double * compute_odometry_data ()`

- void **UMBmark** (double dSpeed, double dDistance)
- void **measure\_clockWise** (double dSpeed, double dDistance)
- void **measure\_CounterClockWise** (double dSpeed, double dDistance)
- void **set\_leds** (int iActive)

## Variables

- double **dCurSpeed** [2] = {0.0f, 0.0f}
- double **dPrevEncPos** [2] = {0.0f, 0.0f}
- double \* **point\_dOdometryData**
- WbDeviceTag **led** [3]
- float **weights\_left** [8] = {-1,-1,-1,0.5,-0.5,0.5,1,2}
- float **weights\_right** [8] = {1,0.8,1,-0.5,0.5,-1,-1.6,-2}

## 5.21.1 Macro Definition Documentation

5.21.1.1 **#define** ANGLE\_TOLERANCE 5

5.21.1.2 **#define** EAST 0

5.21.1.3 **#define** ENCODER\_RESOLUTION 159.23

5.21.1.4 **#define** INCREMENT\_STEP 1000

5.21.1.5 **#define** M\_PI 3.1415926535897932384626433832795L

5.21.1.6 **#define** MIN\_DIST 20.0f

5.21.1.7 **#define** MIN\_SPEED 10.0f

5.21.1.8 **#define** NORTH 90

5.21.1.9 **#define** NUMTOURNAMENTS 1

5.21.1.10 **#define** RANGE (1024 / 2)

5.21.1.11 **#define** RTOD( *r* ) ((*r*) \* 180 / M\_PI)

5.21.1.12 **#define** SOUTH 270

5.21.1.13 **#define** STEP\_TOLERANCE 6.0

5.21.1.14 **#define** TIME\_STEP 8

5.21.1.15 **#define** WEST 180

5.21.1.16 **#define** WHEEL\_RADIUS 0.0206625

5.21.1.17 **#define** WHEELBASE 0.052

## 5.21.2 Function Documentation

5.21.2.1 void **check\_rotation** ( double *cur\_rot*, double *want\_rot*, double *dSpeed* )

FUnction to compare the current heading to the wanted heading and fix the heading should it surpass a threshold

**5.21.2.2 double \* compute\_odometry\_data ( )**

Function to compute the robots current odometry data this includes the x and y placement as well as the rotation theta

**5.21.2.3 double \* get\_encoder\_positions ( )**

Function to get and return the current encoder position of the wheels

**5.21.2.4 void measure\_clockWise ( double *dSpeed*, double *dDistance* )**

Function to measure the movement accuracy by driving a clockwise square. This is part of the UMBmark algorithm

**5.21.2.5 void measure\_CounterClockWise ( double *dSpeed*, double *dDistance* )**

Function to measure the movement accuracy by driving a counter-clockwise square. This is part of the UMBmark algorithm

**5.21.2.6 void move\_forward ( double *dSped*, double *dDis* )**

Function to move the robot forward a given distance at a given speed

**5.21.2.7 void set\_leds ( int *iActive* )**

set the status of the LEDs

**5.21.2.8 void set\_motor\_speed ( double *dLeftSpeed*, double *dRightSpeed* )**

Function to set the motor speed of the robot

**5.21.2.9 void stop\_robot ( )**

Function to stop the robot

**5.21.2.10 void turn\_angle ( double *dAngle*, double *dSpeed* )**

Function to turn the robot a given angle with a given speed

**5.21.2.11 void turn\_left ( double *dSpeed* )**

Function to turn left

**5.21.2.12 void turn\_right ( double *dSpeed* )**

Function to turn right

**5.21.2.13 void UMBmark ( double *dSpeed*, double *dDistance* )**

University of Michigan Benchmark

### 5.21.3 Variable Documentation

5.21.3.1 `double dCurSpeed[2] = {0.0f, 0.0f}`

5.21.3.2 `double dPrevEncPos[2] = {0.0f, 0.0f}`

5.21.3.3 `WbDeviceTag led[3]`

5.21.3.4 `double* point_dOdometryData`

5.21.3.5 `float weights_left[8] = {-1,-1,-1,0.5,-0.5,0.5,1,2}`

5.21.3.6 `float weights_right[8] = {1,0.8,1,-0.5,0.5,-1,-1.6,-2}`

## 5.22 C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/e\_puck\_movement.h File Reference

### Functions

- void **stop\_robot** ()
- void **move\_forward** (double dSpeed, double dDist)
- void **turn\_left** (double dSpeed)
- void **turn\_right** (double dSpeed)
- void **turn\_angle** (double dAngle, double dSpeed)
- void **set\_motor\_speed** (double dSpeedL, double dSpeedR)
- double \* **get\_encoder\_positions** ()
- double \* **compute\_odometry\_data** ()
- void **UMBmark** (double dSpeed, double dDistance)
- void **measure\_clockWise** (double dSpeed, double dDistance)
- void **measure\_CounterClockWise** (double dSpeed, double dDistance)
- void **set\_leds** (int iActive)
- void **controll\_angle** ()
- void **check\_rotation** (double cur\_rot, double want\_rot, double dSpeed)

### 5.22.1 Function Documentation

5.22.1.1 `void check_rotation ( double cur_rot, double want_rot, double dSpeed )`

Function to compare the current heading to the wanted heading and fix the heading should it surpass a threshold

5.22.1.2 `double* compute_odometry_data ( )`

Function to compute the robots current odometry data this includes the x and y placement as well as the rotation theta

5.22.1.3 `void controll_angle ( )`

5.22.1.4 `double* get_encoder_positions ( )`

Function to get and return the current encoder position of the wheels



#### 5.22.1.5 void `measure_clockWise` ( double *dSpeed*, double *dDistance* )

Function to measure the movement accuracy by driving a clockwise square. This is part of the UMBmark algorithm

#### 5.22.1.6 void `measure_CounterClockWise` ( double *dSpeed*, double *dDistance* )

Function to measure the movement accuracy by driving a counter-clockwise square. This is part of the UMBmark algorithm

#### 5.22.1.7 void `move_forward` ( double *dSpeed*, double *dDist* )

Function to move the robot forward a given distance at a given speed

#### 5.22.1.8 void `set_leds` ( int *iActive* )

set the status of the LEDs

#### 5.22.1.9 void `set_motor_speed` ( double *dSpeedL*, double *dSpeedR* )

Function to set the motor speed of the robot

#### 5.22.1.10 void `stop_robot` ( )

Function to stop the robot

#### 5.22.1.11 void `turn_angle` ( double *dAngle*, double *dSpeed* )

Function to turn the robot a given angle with a given speed

#### 5.22.1.12 void `turn_left` ( double *dSpeed* )

Function to turn left

#### 5.22.1.13 void `turn_right` ( double *dSpeed* )

Function to turn right

#### 5.22.1.14 void `UMBmark` ( double *dSpeed*, double *dDistance* )

University of Michigan Benchmark

## 5.23 C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/map\_building.c File Reference

```
#include <webots/robot.h>
#include <webots/differential_wheels.h>
#include <webots/distance_sensor.h>
#include <webots/light_sensor.h>
#include <math.h>
#include <webots/display.h>
#include <stdio.h>
#include "map_building.h"
#include "e_puck_movement.h"
#include "odometry.h"
```

### Macros

- **#define TIME\_STEP** 8
- **#define MAP\_SIZE** 70
- **#define CELL\_SIZE** 0.015
- **#define THRESHOLD\_DIST** 100
- **#define OCCUPANCE\_DIST** 150
- **#define LEFT** 0
- **#define RIGHT** 1
- **#define NUM\_DIST\_SENS** 8
- **#define PS\_RIGHT\_10** 0
- **#define PS\_RIGHT\_45** 1
- **#define PS\_RIGHT\_90** 2
- **#define PS\_RIGHT\_REAR** 3
- **#define PS\_LEFT\_REAR** 4
- **#define PS\_LEFT\_90** 5
- **#define PS\_LEFT\_45** 6
- **#define PS\_LEFT\_10** 7
- **#define FORWARD** 0
- **#define STOP** 1
- **#define TURNRIGHT** 2
- **#define TURNLEFT** 3
- **#define UTURN** 4
- **#define RTOD(r)** ((r) \* 180 / M\_PI)
- **#define ANGLE\_TOLERANCE** 20
- **#define EAST** 0
- **#define NORTH** 90
- **#define WEST** 180
- **#define SOUTH** 270

### Functions

- void **check\_direction** (double d)
- void **init\_display** ()
- void **occupied\_cell** (int x, int y, float theta)
- void **reset** ()
- void **run** (struct **odometryTrackStruct** \*ot)
- int \* **return\_sensor\_values** ()
- int **return\_angle** (double rad)

## Variables

- WbDeviceTag **led** [3]
- WbDeviceTag **ps** [NUM\_DIST\_SENS]
- WbDeviceTag **display**
- WbImageRef **background**
- int **display\_width**
- int **display\_height**
- int **map** [MAP\_SIZE][MAP\_SIZE]
- int **robot\_x** = MAP\_SIZE / 2
- int **robot\_y** = MAP\_SIZE / 2
- int **ps\_value** [NUM\_DIST\_SENS] = {0,0,0,0,0,0,0,0}
- int **obstacle** [NUM\_DIST\_SENS] = {0,0,0,0,0,0,0,0}
- bool **ob\_front**
- bool **ob\_right**
- bool **ob\_left**
- float **angle\_offset** [NUM\_DIST\_SENS] = {0.2793, 0.7854, 1.5708, 2.618, -2.618, -1.5708, -0.7854, -0.2793}
- int **new\_encoder**
- bool **north**
- bool **west**
- bool **south**
- bool **east**
- int **state** = FORWARD

### 5.23.1 Macro Definition Documentation

5.23.1.1 **#define** ANGLE\_TOLERANCE 20

5.23.1.2 **#define** CELL\_SIZE 0.015

5.23.1.3 **#define** EAST 0

5.23.1.4 **#define** FORWARD 0

5.23.1.5 **#define** LEFT 0

5.23.1.6 **#define** MAP\_SIZE 70

5.23.1.7 **#define** NORTH 90

5.23.1.8 **#define** NUM\_DIST\_SENS 8

5.23.1.9 **#define** OCCUPANCE\_DIST 150

5.23.1.10 **#define** PS\_LEFT\_10 7

5.23.1.11 **#define** PS\_LEFT\_45 6

5.23.1.12 **#define** PS\_LEFT\_90 5

5.23.1.13 **#define** PS\_LEFT\_REAR 4

5.23.1.14 **#define** PS\_RIGHT\_10 0

5.23.1.15 **#define** PS\_RIGHT\_45 1

5.23.1.16 `#define PS_RIGHT_90 2`

5.23.1.17 `#define PS_RIGHT_REAR 3`

5.23.1.18 `#define RIGHT 1`

5.23.1.19 `#define RTOD( r ) ((r) * 180 / M_PI)`

5.23.1.20 `#define SOUTH 270`

5.23.1.21 `#define STOP 1`

5.23.1.22 `#define THRESHOLD_DIST 100`

5.23.1.23 `#define TIME_STEP 8`

5.23.1.24 `#define TURNLEFT 3`

5.23.1.25 `#define TURNRIGHT 2`

5.23.1.26 `#define UTURN 4`

5.23.1.27 `#define WEST 180`

## 5.23.2 Function Documentation

5.23.2.1 `void check_direction ( double d )`

set booleans for the direction the robot is moving in

5.23.2.2 `void init_display ( )`

Initiate the display with a white color

5.23.2.3 `void occupied_cell ( int x, int y, float theta )`

Set the corresponding cell to 1 (occuM\_Pled) and display it

5.23.2.4 `void reset ( void )`

enables the needed sensor devices

5.23.2.5 `int return_angle ( double rad )`

returns the angle in which the robot is moving

5.23.2.6 `int* return_sensor_values ( )`

returns a pointer to an array of the current sensor values

5.23.2.7 `void run ( struct odometryTrackStruct * ot )`

run function

### 5.23.3 Variable Documentation

5.23.3.1 float angle\_offset[NUM\_DIST\_SENS] = {0.2793, 0.7854, 1.5708, 2.618, -2.618, -1.5708, -0.7854, -0.2793}

5.23.3.2 WbImageRef background

5.23.3.3 WbDeviceTag display

5.23.3.4 int display\_height

5.23.3.5 int display\_width

5.23.3.6 bool east

5.23.3.7 WbDeviceTag led[3]

5.23.3.8 int map[MAP\_SIZE][MAP\_SIZE]

5.23.3.9 int new\_encoder

5.23.3.10 bool north

5.23.3.11 bool ob\_front

5.23.3.12 bool ob\_left

5.23.3.13 bool ob\_right

5.23.3.14 int obstacle[NUM\_DIST\_SENS] = {0,0,0,0,0,0,0,0}

5.23.3.15 WbDeviceTag ps[NUM\_DIST\_SENS]

5.23.3.16 int ps\_value[NUM\_DIST\_SENS] = {0,0,0,0,0,0,0,0}

5.23.3.17 int robot\_x = MAP\_SIZE / 2

5.23.3.18 int robot\_y = MAP\_SIZE / 2

5.23.3.19 bool south

5.23.3.20 int state = FORWARD

5.23.3.21 bool west

## 5.24 C:/Users/Stefan/Documents/GitHub/Final\_Project/map\_building.c File Reference

```
#include <webots/robot.h>
#include <webots/differential_wheels.h>
#include <webots/distance_sensor.h>
#include <webots/light_sensor.h>
#include <math.h>
#include <webots/display.h>
#include "map_building.h"
#include "e_puck_movement.h"
#include "odometry.h"
```

## Macros

- `#define TIME_STEP 8`
- `#define MAP_SIZE 70`
- `#define CELL_SIZE 0.015`
- `#define THRESHOLD_DIST 100`
- `#define OCCUPANCE_DIST 150`
- `#define LEFT 0`
- `#define RIGHT 1`
- `#define NUM_DIST_SENS 8`
- `#define PS_RIGHT_10 0`
- `#define PS_RIGHT_45 1`
- `#define PS_RIGHT_90 2`
- `#define PS_RIGHT_REAR 3`
- `#define PS_LEFT_REAR 4`
- `#define PS_LEFT_90 5`
- `#define PS_LEFT_45 6`
- `#define PS_LEFT_10 7`
- `#define FORWARD 0`
- `#define TURNRIGHT 1`
- `#define TURNLEFT 2`
- `#define UTURN 3`

## Functions

- `void init_display ()`
- `void occupied_cell (int x, int y, float theta)`
- `void reset ()`
- `void run (struct odometryTrackStruct *ot)`
- `int * return_sensor_values ()`

## Variables

- `WbDeviceTag led [3]`
- `WbDeviceTag ps [NUM_DIST_SENS]`
- `WbDeviceTag display`
- `WbImageRef background`
- `int display_width`
- `int display_height`
- `int map [MAP_SIZE][MAP_SIZE]`
- `int robot_x = MAP_SIZE / 2`
- `int robot_y = MAP_SIZE / 2`
- `int ps_value [NUM_DIST_SENS] = {0,0,0,0,0,0,0,0}`
- `int obstacle [NUM_DIST_SENS] = {0,0,0,0,0,0,0,0}`
- `bool ob_front`
- `bool ob_right`
- `bool ob_left`
- `float angle_offset [NUM_DIST_SENS] = {0.2793, 0.7854, 1.5708, 2.618, -2.618, -1.5708, -0.7854, -0.2793}`
- `int new_encoder`
- `int state = FORWARD`

### 5.24.1 Macro Definition Documentation

5.24.1.1 `#define CELL_SIZE 0.015`

5.24.1.2 `#define FORWARD 0`

5.24.1.3 `#define LEFT 0`

5.24.1.4 `#define MAP_SIZE 70`

5.24.1.5 `#define NUM_DIST_SENS 8`

5.24.1.6 `#define OCCUPANCE_DIST 150`

5.24.1.7 `#define PS_LEFT_10 7`

5.24.1.8 `#define PS_LEFT_45 6`

5.24.1.9 `#define PS_LEFT_90 5`

5.24.1.10 `#define PS_LEFT_REAR 4`

5.24.1.11 `#define PS_RIGHT_10 0`

5.24.1.12 `#define PS_RIGHT_45 1`

5.24.1.13 `#define PS_RIGHT_90 2`

5.24.1.14 `#define PS_RIGHT_REAR 3`

5.24.1.15 `#define RIGHT 1`

5.24.1.16 `#define THRESHOLD_DIST 100`

5.24.1.17 `#define TIME_STEP 8`

5.24.1.18 `#define TURNLEFT 2`

5.24.1.19 `#define TURNRIGHT 1`

5.24.1.20 `#define UTURN 3`

### 5.24.2 Function Documentation

5.24.2.1 `void init_display ( )`

Initiate the display with a white color

5.24.2.2 `void occupied_cell ( int x, int y, float theta )`

Set the corresponding cell to 1 (occuM\_Pled) and display it

5.24.2.3 `void reset ( void )`

enables the needed sensor devices

5.24.2.4 `int* return_sensor_values ( )`

returns a pointer to an array of the current sensor values

5.24.2.5 `void run ( struct odometryTrackStruct * ot )`

### 5.24.3 Variable Documentation

5.24.3.1 `float angle_offset[NUM_DIST_SENS] = {0.2793, 0.7854, 1.5708, 2.618, -2.618, -1.5708, -0.7854, -0.2793}`

5.24.3.2 `WbImageRef background`

5.24.3.3 `WbDeviceTag display`

5.24.3.4 `int display_height`

5.24.3.5 `int display_width`

5.24.3.6 `WbDeviceTag led[3]`

5.24.3.7 `int map[MAP_SIZE][MAP_SIZE]`

5.24.3.8 `int new_encoder`

5.24.3.9 `bool ob_front`

5.24.3.10 `bool ob_left`

5.24.3.11 `bool ob_right`

5.24.3.12 `int obstacle[NUM_DIST_SENS] = {0,0,0,0,0,0,0,0}`

5.24.3.13 `WbDeviceTag ps[NUM_DIST_SENS]`

5.24.3.14 `int ps_value[NUM_DIST_SENS] = {0,0,0,0,0,0,0,0}`

5.24.3.15 `int robot_x = MAP_SIZE / 2`

5.24.3.16 `int robot_y = MAP_SIZE / 2`

5.24.3.17 `int state = FORWARD`

## 5.25 C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/map\_building.h File Reference

### Functions

- `void init_display ()`
- `void occupied_cell (int x, int y, float theta)`
- `void reset ()`
- `void run ()`
- `int * return_sensor_values ()`
- `int return_angle ()`



### 5.25.1 Function Documentation

#### 5.25.1.1 void init\_display ( )

Initiate the display with a white color

#### 5.25.1.2 void occupied\_cell ( int x, int y, float *theta* )

Set the corresponding cell to 1 (occuM\_Pled) and display it

#### 5.25.1.3 void reset ( void )

enables the needed sensor devices

#### 5.25.1.4 int return\_angle ( )

#### 5.25.1.5 int\* return\_sensor\_values ( )

returns a pointer to an array of the current sensor values

#### 5.25.1.6 void run ( )

## 5.26 C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/odometry.c File Reference

```
#include <webots/differential_wheels.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "odometry.h"
#include "e_puck_movement.h"
#include "errno.h"
```

### Macros

- `#define M_PI 3.1415926535897932384626433832795L`

### Functions

- int **odometry\_track\_start** (struct **odometryTrackStruct** \*ot)
- int **odometry\_track\_start\_pos** (struct **odometryTrackStruct** \*ot, double \*dEncPos)
- void **odometry\_track\_step** (struct **odometryTrackStruct** \*ot)
- void **odometry\_track\_step\_pos** (struct **odometryTrackStruct** \*ot, double \*dEncPos)

### Variables

- float **increments\_per\_tour** = 1000.0
- float **wheel\_radius** = 0.0207
- float **scaling\_factor** = 1
- float **axis\_wheel\_ratio** = 1.4134

## 5.26.1 Macro Definition Documentation

5.26.1.1 `#define M_PI 3.1415926535897932384626433832795L`

## 5.26.2 Function Documentation

5.26.2.1 `int odometry_track_start ( struct odometryTrackStruct * ot )`

Initializes the odometry algorithms

5.26.2.2 `int odometry_track_start_pos ( struct odometryTrackStruct * ot, double * dEncPos )`

Start the odometry tracking Updates the info in the **odometryTrackStruct** (p. 7) for the first time

5.26.2.3 `void odometry_track_step ( struct odometryTrackStruct * ot )`

5.26.2.4 `void odometry_track_step_pos ( struct odometryTrackStruct * ot, double * dEncPos )`

## 5.26.3 Variable Documentation

5.26.3.1 `float axis_wheel_ratio = 1.4134`

5.26.3.2 `float increments_per_tour = 1000.0`

5.26.3.3 `float scaling_factor = 1`

5.26.3.4 `float wheel_radius = 0.0207`

## 5.27 C:/Users/Stefan/Documents/GitHub/Final\_Project/odometry.c File Reference

```
#include <webots/differential_wheels.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <math.h>
#include "odometry.h"
#include "e_puck_movement.h"
#include "errno.h"
```

### Macros

- `#define M_PI 3.1415926535897932384626433832795L`

### Functions

- `int odometry_track_start (struct odometryTrackStruct *ot)`
- `int odometry_track_start_pos (struct odometryTrackStruct *ot, double *dEncPos)`
- `void odometry_track_step (struct odometryTrackStruct *ot)`
- `void odometry_track_step_pos (struct odometryTrackStruct *ot, double *dEncPos)`

## Variables

- float **increments\_per\_tour** = 1000.0
- float **wheel\_radius** = 0.0207
- float **scaling\_factor** = 1
- float **axis\_wheel\_ratio** = 1.4134

### 5.27.1 Macro Definition Documentation

5.27.1.1 `#define M_PI 3.1415926535897932384626433832795L`

### 5.27.2 Function Documentation

5.27.2.1 `int odometry_track_start ( struct odometryTrackStruct * ot )`

Initializes the odometry algorithms

5.27.2.2 `int odometry_track_start_pos ( struct odometryTrackStruct * ot, double * dEncPos )`

Start the odometry tracking Updates the info in the **odometryTrackStruct** (p. 7) for the first time

5.27.2.3 `void odometry_track_step ( struct odometryTrackStruct * ot )`

5.27.2.4 `void odometry_track_step_pos ( struct odometryTrackStruct * ot, double * dEncPos )`

### 5.27.3 Variable Documentation

5.27.3.1 `float axis_wheel_ratio = 1.4134`

5.27.3.2 `float increments_per_tour = 1000.0`

5.27.3.3 `float scaling_factor = 1`

5.27.3.4 `float wheel_radius = 0.0207`

## 5.28 C:/Users/Stefan/Documents/GitHub/Final\_Project/lib/odometry.h File Reference

### Data Structures

- struct **odometryTrackStruct**

### Functions

- int **odometry\_track\_start** (struct **odometryTrackStruct** \*ot)
- int **odometry\_track\_start\_pos** (struct **odometryTrackStruct** \*ot, double \*dEncPos)
- void **odometry\_track\_step** (struct **odometryTrackStruct** \*ot)
- void **odometry\_track\_step\_pos** (struct **odometryTrackStruct** \*ot, double \*dEncPos)

### 5.28.1 Function Documentation

5.28.1.1 `int odometry_track_start ( struct odometryTrackStruct * ot )`

Initializes the odometry algorithms

5.28.1.2 `int odometry_track_start_pos ( struct odometryTrackStruct * ot, double * dEncPos )`

Start the odometry tracking Updates the info in the **odometryTrackStruct** (p. 7) for the first time

5.28.1.3 `void odometry_track_step ( struct odometryTrackStruct * ot )`

5.28.1.4 `void odometry_track_step_pos ( struct odometryTrackStruct * ot, double * dEncPos )`

## 5.29 C:/Users/Stefan/Documents/GitHub/Final\_Project/README.md File Reference

## 5.30 C:/Users/Stefan/Documents/GitHub/Final\_Project/scanline\_backup.c File Reference

```
#include <webots/robot.h>
#include <webots/differential_wheels.h>
#include <webots/distance_sensor.h>
#include <stdlib.h>
#include <stdio.h>
```

### Macros

- `#define TIME_STEP 64`
- `#define LEFT 0`
- `#define RIGHT 1`
- `#define THRESHOLD_DIST 100`
- `#define NUM_DIST_SENS 8`
- `#define PS_RIGHT_10 0`
- `#define PS_RIGHT_45 1`
- `#define PS_RIGHT_90 2`
- `#define PS_RIGHT_REAR 3`
- `#define PS_LEFT_REAR 4`
- `#define PS_LEFT_90 5`
- `#define PS_LEFT_45 6`
- `#define PS_LEFT_10 7`
- `#define FORWARD 0`
- `#define STOP 1`
- `#define UTURN 2`
- `#define TURNRIGHT 3`
- `#define TURNLEFT 4`
- `#define WHEEL_RADIUS 0.02`
- `#define AXLE_LENGTH 0.026`
- `#define ENCODER_RESOLUTION 159.23`

### Functions

- `void fsm ()`
- `int main ()`

## Variables

- WbDeviceTag **ps** [NUM\_DIST\_SENS]
- int **ps\_value** [NUM\_DIST\_SENS] = {0,0,0,0,0,0,0,0}
- int **ps\_offset** [NUM\_DIST\_SENS] = {35,35,35,35,35,35,35,35}
- int **obstacle** [NUM\_DIST\_SENS]
- int **state** = FORWARD
- int **speed** [2] = {0,0}
- int **fsm\_speed** [2] = {0,0}

## 5.30.1 Macro Definition Documentation

5.30.1.1 `#define AXLE_LENGTH 0.026`

5.30.1.2 `#define ENCODER_RESOLUTION 159.23`

5.30.1.3 `#define FORWARD 0`

5.30.1.4 `#define LEFT 0`

5.30.1.5 `#define NUM_DIST_SENS 8`

5.30.1.6 `#define PS_LEFT_10 7`

5.30.1.7 `#define PS_LEFT_45 6`

5.30.1.8 `#define PS_LEFT_90 5`

5.30.1.9 `#define PS_LEFT_REAR 4`

5.30.1.10 `#define PS_RIGHT_10 0`

5.30.1.11 `#define PS_RIGHT_45 1`

5.30.1.12 `#define PS_RIGHT_90 2`

5.30.1.13 `#define PS_RIGHT_REAR 3`

5.30.1.14 `#define RIGHT 1`

5.30.1.15 `#define STOP 1`

5.30.1.16 `#define THRESHOLD_DIST 100`

5.30.1.17 `#define TIME_STEP 64`

5.30.1.18 `#define TURNLEFT 4`

5.30.1.19 `#define TURNRIGHT 3`

5.30.1.20 `#define UTURN 2`

5.30.1.21 `#define WHEEL_RADIUS 0.02`

## 5.30.2 Function Documentation

#### 5.30.2.1 void fsm ( )

Function which holds the FSM

#### 5.30.2.2 int main ( )

Main method

### 5.30.3 Variable Documentation

#### 5.30.3.1 int fsm\_speed[2] = {0,0}

#### 5.30.3.2 int obstacle[NUM\_DIST\_SENS]

#### 5.30.3.3 WbDeviceTag ps[NUM\_DIST\_SENS]

Global Values

#### 5.30.3.4 int ps\_offset[NUM\_DIST\_SENS] = {35,35,35,35,35,35,35,35}

#### 5.30.3.5 int ps\_value[NUM\_DIST\_SENS] = {0,0,0,0,0,0,0,0}

#### 5.30.3.6 int speed[2] = {0,0}

#### 5.30.3.7 int state = FORWARD