



# Savari V2X Library API (v5.9.0)

CONFIDENTIAL

October 2017



# Contents

<b>1</b>	<b>Savari V2X Libraries</b>	<b>1</b>
1.1	Introduction	1
1.2	V2X Common Library	2
1.3	V2X Message Library	2
1.4	V2X Facility Library	2
1.4.1	V2X Basic Safety Service Library	3
1.4.2	V2X Cooperative Service Library	3
1.5	V2X Security Library	3
1.6	V2X Transport and Network Services Library	3
1.6.1	Dynamic and static radio selection	3
1.7	V2X Sensor Library	3
<b>2</b>	<b>Module Index</b>	<b>5</b>
2.1	Modules	5
<b>3</b>	<b>Data Structure Index</b>	<b>7</b>
3.1	Data Structures	7
<b>4</b>	<b>File Index</b>	<b>11</b>
4.1	File List	11
<b>5</b>	<b>Module Documentation</b>	<b>13</b>
5.1	The V2X Common API	13
5.1.1	Detailed Description	13
5.1.2	Enumeration Type Documentation	13
5.1.2.1	v2x_status_t	13
5.1.3	Function Documentation	14
5.1.3.1	v2x_debug_level_set	14
5.1.3.2	v2x_log	14
5.1.3.3	v2x_strerror	14
5.2	The V2X Message API (US)	16
5.2.1	Detailed Description	19
5.2.2	Typedef Documentation	20

5.2.2.1	J2735_2016_Connectsto_Lanes_t	20
5.2.2.2	J2735_2016_DSRC_MsgID_t	20
5.2.2.3	J2735_2016_HeadingSlice_t	20
5.2.2.4	J2735_2016_LaneAttribute_t	20
5.2.2.5	J2735_2016_LaneDir_t	20
5.2.2.6	J2735_2016_LaneList_t	20
5.2.2.7	J2735_2016_LaneTypeAttrCrossWk_t	20
5.2.2.8	J2735_2016_LaneTypeAttrVeh_t	20
5.2.2.9	J2735_2016_NodeAttr_t	20
5.2.2.10	J2735_2016_NodeList_t	20
5.2.2.11	J2735_2016_NodePoint_t	20
5.2.2.12	J2735_2016_PathHistory_t	21
5.2.2.13	J2735_2016_PathPrediction_t	21
5.2.2.14	J2735_2016_WiperSet_t	21
5.2.2.15	J2735Extent_t	21
5.2.2.16	J2735Position3D_t	21
5.2.2.17	v2x_msg_bsm_t	21
5.2.2.18	v2x_msg_eva_t	21
5.2.2.19	v2x_msg_map_t	21
5.2.2.20	v2x_msg_rtcn_t	21
5.2.2.21	v2x_msg_spat_t	21
5.2.2.22	v2x_msg_srm_t	21
5.2.2.23	v2x_msg_ssm_t	21
5.2.2.24	v2x_msg_tim_t	22
5.2.3	Enumeration Type Documentation	22
5.2.3.1	J2735_2016_AllowedManeuvers_t	22
5.2.3.2	J2735_2016_DirofUse_t	23
5.2.3.3	J2735_2016_ExteriorLights_t	23
5.2.3.4	J2735_2016_HeadingSlice	23
5.2.3.5	J2735_2016_NodeAttrType_t	24
5.2.3.6	J2735_2016_NodeType_t	25
5.2.3.7	J2735_2016_VehicleEventFlags_t	25
5.2.3.8	J2735_2016_WiperStatus_t	25
5.2.3.9	j2735_dsrc_msgids	26
5.2.3.10	J2735Extent	26
5.2.3.11	lanedirection	26
5.2.3.12	lanetypecrosswalk	26
5.2.3.13	lanetypevehicle	27
5.2.4	Function Documentation	27
5.2.4.1	v2x_map_free	27

5.2.4.2	<a href="#">v2x_msg_bsm_decode</a>	27
5.2.4.3	<a href="#">v2x_msg_bsm_encode</a>	28
5.2.4.4	<a href="#">v2x_msg_bsm_init</a>	28
5.2.4.5	<a href="#">v2x_msg_eva_decode</a>	29
5.2.4.6	<a href="#">v2x_msg_eva_encode</a>	29
5.2.4.7	<a href="#">v2x_msg_eva_print</a>	29
5.2.4.8	<a href="#">v2x_msg_eva_reset</a>	30
5.2.4.9	<a href="#">v2x_msg_map_decode</a>	30
5.2.4.10	<a href="#">v2x_msg_map_encode</a>	30
5.2.4.11	<a href="#">v2x_msg_map_init</a>	31
5.2.4.12	<a href="#">v2x_msg_psm_decode</a>	31
5.2.4.13	<a href="#">v2x_msg_psm_encode</a>	31
5.2.4.14	<a href="#">v2x_msg_psm_reset</a>	32
5.2.4.15	<a href="#">v2x_msg_rtcn_decode</a>	32
5.2.4.16	<a href="#">v2x_msg_rtcn_encode</a>	33
5.2.4.17	<a href="#">v2x_msg_rtcn_print</a>	33
5.2.4.18	<a href="#">v2x_msg_spat_decode</a>	33
5.2.4.19	<a href="#">v2x_msg_spat_encode</a>	34
5.2.4.20	<a href="#">v2x_msg_tim_decode</a>	34
5.2.4.21	<a href="#">v2x_msg_tim_encode</a>	35
5.2.4.22	<a href="#">v2x_msg_tim_free</a>	36
5.2.4.23	<a href="#">v2x_pvd_decode</a>	36
5.2.4.24	<a href="#">v2x_pvd_encode</a>	36
5.2.4.25	<a href="#">v2x_pvd_reset</a>	37
5.2.4.26	<a href="#">v2x_pvd_snapshot_reset</a>	37
5.2.4.27	<a href="#">v2x_spat_free</a>	37
5.2.4.28	<a href="#">v2x_srm_decode</a>	37
5.2.4.29	<a href="#">v2x_srm_encode</a>	38
5.2.4.30	<a href="#">v2x_srm_reset</a>	38
5.2.4.31	<a href="#">v2x_ssm_decode</a>	38
5.2.4.32	<a href="#">v2x_ssm_encode</a>	39
5.2.4.33	<a href="#">v2x_ssm_reset</a>	39
5.3	<a href="#">The V2X Message API (EU)</a>	40
5.3.1	<a href="#">Detailed Description</a>	40
5.3.2	<a href="#">Function Documentation</a>	40
5.3.2.1	<a href="#">v2x_etsi_cam_decode</a>	40
5.3.2.2	<a href="#">v2x_etsi_cam_encode</a>	41
5.3.2.3	<a href="#">v2x_etsi_cam_print</a>	41
5.3.2.4	<a href="#">v2x_etsi_cam_reset</a>	41
5.3.2.5	<a href="#">v2x_etsi_denm_decode</a>	42

5.3.2.6	<a href="#">v2x_etsi_denm_dumphex</a>	42
5.3.2.7	<a href="#">v2x_etsi_denm_encode</a>	42
5.3.2.8	<a href="#">v2x_etsi_map_decode</a>	43
5.3.2.9	<a href="#">v2x_etsi_map_encode</a>	43
5.3.2.10	<a href="#">v2x_etsi_map_free</a>	44
5.3.2.11	<a href="#">v2x_etsi_map_print</a>	44
5.3.2.12	<a href="#">v2x_etsi_map_reset</a>	45
5.3.2.13	<a href="#">v2x_etsi_print_denm</a>	46
5.3.2.14	<a href="#">v2x_etsi_reset_denm</a>	46
5.3.2.15	<a href="#">v2x_etsi_reset_spat</a>	46
5.3.2.16	<a href="#">v2x_etsi_spat_decode</a>	47
5.3.2.17	<a href="#">v2x_etsi_spat_encode</a>	47
5.3.2.18	<a href="#">v2x_etsi_spat_print</a>	48
5.4	<a href="#">The V2X Network and Transport module API (EU)</a>	49
5.4.1	<a href="#">Detailed Description</a>	49
5.4.2	<a href="#">Function Documentation</a>	49
5.4.2.1	<a href="#">v2x_btp_deinit</a>	49
5.4.2.2	<a href="#">v2x_btp_get_sock_id</a>	49
5.4.2.3	<a href="#">v2x_btp_init</a>	50
5.4.2.4	<a href="#">v2x_btp_receive</a>	50
5.4.2.5	<a href="#">v2x_btp_transmit</a>	51
5.5	<a href="#">The V2X Network and Transport module API (US)</a>	52
5.5.1	<a href="#">Detailed Description</a>	53
5.5.2	<a href="#">Typedef Documentation</a>	53
5.5.2.1	<a href="#">v2x_wsmp_callbacks_t</a>	53
5.5.2.2	<a href="#">v2x_wsmp_handler_t</a>	53
5.5.2.3	<a href="#">v2x_wsmp_rcv_indication_t</a>	53
5.5.3	<a href="#">Enumeration Type Documentation</a>	53
5.5.3.1	<a href="#">anonymous enum</a>	53
5.5.3.2	<a href="#">anonymous enum</a>	54
5.5.3.3	<a href="#">anonymous enum</a>	54
5.5.3.4	<a href="#">anonymous enum</a>	54
5.5.3.5	<a href="#">LIBWME_RC_RESULT_CODE</a>	54
5.5.4	<a href="#">Function Documentation</a>	54
5.5.4.1	<a href="#">v2x_wme_open</a>	54
5.5.4.2	<a href="#">v2x_wme_register_user</a>	55
5.5.4.3	<a href="#">v2x_wme_unregister_user</a>	55
5.5.4.4	<a href="#">v2x_wme_user_service_confirm</a>	56
5.5.4.5	<a href="#">v2x_wsmp_convert_psid_be</a>	56
5.5.4.6	<a href="#">v2x_wsmp_rcv</a>	57

5.5.4.7	<a href="#">v2x_wsmp_send</a>	57
5.5.5	<a href="#">Variable Documentation</a>	58
5.5.5.1	<a href="#">LIBWME_USER_ACTION</a>	58
5.5.5.2	<a href="#">P1609_CHANNEL_ACCESS</a>	58
5.5.5.3	<a href="#">P1609_WSATYPE</a>	58
5.6	<a href="#">The V2X Security module API</a>	59
5.6.1	<a href="#">Detailed Description</a>	59
5.6.2	<a href="#">Typedef Documentation</a>	59
5.6.2.1	<a href="#">v2x_sec_rx_cblist_t</a>	59
5.6.2.2	<a href="#">v2x_security_profile_t</a>	59
5.6.2.3	<a href="#">v2x_sign_sec_profile_t</a>	59
5.6.2.4	<a href="#">v2x_verify_sec_profile_t</a>	60
5.6.3	<a href="#">Enumeration Type Documentation</a>	60
5.6.3.1	<a href="#">v2x_security_profile_type_t</a>	60
5.6.4	<a href="#">Function Documentation</a>	60
5.6.4.1	<a href="#">v2x_sec_cert_change_register</a>	60
5.6.4.2	<a href="#">v2x_sec_cert_change_unregister</a>	60
5.6.4.3	<a href="#">v2x_sec_deinit</a>	61
5.6.4.4	<a href="#">v2x_sec_idchange_lock</a>	61
5.6.4.5	<a href="#">v2x_sec_idchange_unlock</a>	61
5.6.4.6	<a href="#">v2x_sec_init</a>	62
5.6.4.7	<a href="#">v2x_sec_rx</a>	62
5.6.4.8	<a href="#">v2x_sec_sign_request</a>	62
5.6.4.9	<a href="#">v2x_sec_verify_request</a>	63
5.7	<a href="#">The V2X Sensors API</a>	64
5.7.1	<a href="#">Detailed Description</a>	64
5.7.2	<a href="#">Typedef Documentation</a>	64
5.7.2.1	<a href="#">pf_gnss_data_t</a>	64
5.7.2.2	<a href="#">v2x_gnss_data_t</a>	64
5.7.2.3	<a href="#">v2x_gnss_handle_t</a>	64
5.7.2.4	<a href="#">v2x_pf_handle</a>	65
5.7.3	<a href="#">Function Documentation</a>	65
5.7.3.1	<a href="#">v2x_can_close</a>	65
5.7.3.2	<a href="#">v2x_can_open</a>	65
5.7.3.3	<a href="#">v2x_can_recv</a>	65
5.7.3.4	<a href="#">v2x_gnss_close</a>	67
5.7.3.5	<a href="#">v2x_gnss_filters_deinit</a>	67
5.7.3.6	<a href="#">v2x_gnss_filters_init</a>	68
5.7.3.7	<a href="#">v2x_gnss_open</a>	68
5.7.3.8	<a href="#">v2x_gnss_read</a>	68

5.7.3.9	v2x_libpcan_net_deinit . . . . .	69
5.7.3.10	v2x_libpcan_net_recv . . . . .	69
<b>6</b>	<b>Data Structure Documentation</b>	<b>71</b>
6.1	_J2735_2016_EnableLaneList_t Struct Reference . . . . .	71
6.1.1	Detailed Description . . . . .	71
6.1.2	Field Documentation . . . . .	71
6.1.2.1	lane_id . . . . .	71
6.2	_J2735_2016_ManeuverAssistList_t Struct Reference . . . . .	71
6.2.1	Detailed Description . . . . .	71
6.2.2	Field Documentation . . . . .	72
6.2.2.1	connection_id . . . . .	72
6.3	_J2735_2016_MovementEventList_t Struct Reference . . . . .	72
6.3.1	Detailed Description . . . . .	72
6.3.2	Field Documentation . . . . .	72
6.3.2.1	phase_state . . . . .	72
6.3.2.2	time_change_info . . . . .	72
6.4	_J2735_2016_MovementState_t Struct Reference . . . . .	72
6.4.1	Detailed Description . . . . .	73
6.4.2	Field Documentation . . . . .	73
6.4.2.1	movement_event . . . . .	73
6.4.2.2	num_movement_event . . . . .	73
6.4.2.3	signal_group_id . . . . .	73
6.5	_J2735_2016_PathPrediction Struct Reference . . . . .	73
6.5.1	Detailed Description . . . . .	73
6.5.2	Field Documentation . . . . .	73
6.5.2.1	confidence . . . . .	73
6.5.2.2	radius_of_curvature . . . . .	73
6.6	_J2735_2016_PHObject Struct Reference . . . . .	74
6.6.1	Detailed Description . . . . .	74
6.6.2	Field Documentation . . . . .	74
6.6.2.1	fps . . . . .	74
6.6.2.2	gps_status . . . . .	74
6.6.2.3	ph_points . . . . .	74
6.6.2.4	ph_points_len . . . . .	74
6.6.2.5	valid_fps . . . . .	74
6.7	_J2735_2016_ProbeSnapshot Struct Reference . . . . .	74
6.7.1	Detailed Description . . . . .	75
6.7.2	Field Documentation . . . . .	75
6.7.2.1	fps . . . . .	75



6.7.2.2	valid_veh_safety_ext . . . . .	75
6.7.2.3	valid_veh_status . . . . .	75
6.7.2.4	veh_safety_ext . . . . .	75
6.7.2.5	vehicle_status . . . . .	75
6.8	_J2735_2016_PVD Struct Reference . . . . .	75
6.8.1	Detailed Description . . . . .	76
6.8.2	Field Documentation . . . . .	76
6.8.2.1	full_posvec . . . . .	76
6.8.2.2	min_of_year . . . . .	76
6.8.2.3	probe_seg_num . . . . .	76
6.8.2.4	snapshot . . . . .	76
6.8.2.5	snapshot_len . . . . .	76
6.8.2.6	valid_fullpos_vec . . . . .	76
6.8.2.7	veh_class . . . . .	76
6.8.2.8	vid_info . . . . .	76
6.8.2.9	vid_info_present . . . . .	76
6.9	_J2735_2016_TimeChange_t Struct Reference . . . . .	77
6.9.1	Detailed Description . . . . .	77
6.9.2	Field Documentation . . . . .	77
6.9.2.1	confidence . . . . .	77
6.9.2.2	likely_time . . . . .	77
6.9.2.3	max_end_time . . . . .	77
6.9.2.4	min_end_time . . . . .	77
6.9.2.5	next_time . . . . .	77
6.9.2.6	start_time . . . . .	77
6.10	_J2735_2016_VehicleMeasurements Struct Reference . . . . .	78
6.11	_J2735_2016_VehicleStatus Struct Reference . . . . .	78
6.11.1	Detailed Description . . . . .	78
6.11.2	Field Documentation . . . . .	78
6.11.2.1	exterior_lights . . . . .	78
6.11.2.2	lightbar_in_use . . . . .	78
6.11.2.3	valid_vehicle_data . . . . .	78
6.11.2.4	valid_wiper_set . . . . .	78
6.11.2.5	vehicle_data . . . . .	79
6.11.2.6	wiper_set . . . . .	79
6.12	_J2735_2016_VehSafetyExtensions Struct Reference . . . . .	79
6.12.1	Detailed Description . . . . .	79
6.12.2	Field Documentation . . . . .	79
6.12.2.1	event_flags . . . . .	79
6.12.2.2	exterior_lights . . . . .	79

6.12.2.3	ph	79
6.12.2.4	pp	79
6.12.2.5	valid_ph	79
6.12.2.6	valid_pp	80
6.13	_J2735_2016_Wiperset Struct Reference	80
6.13.1	Detailed Description	80
6.13.2	Field Documentation	80
6.13.2.1	wiper_front	80
6.13.2.2	wiper_front_rate	80
6.13.2.3	wiper_rear	80
6.13.2.4	wiper_rear_rate	80
6.14	ae_ui_rx_callbacks Struct Reference	80
6.14.1	Detailed Description	81
6.14.2	Field Documentation	81
6.14.2.1	ae_sign_callback	81
6.14.2.2	ae_verify_callback	81
6.15	aerolink_sign_secprofile Struct Reference	82
6.15.1	Field Documentation	82
6.15.1.1	cert_attach_rate	82
6.15.1.2	use_expiry_time	82
6.15.1.3	use_generation_loc	82
6.15.1.4	use_generation_time	82
6.16	aerolink_verify_secprofile Struct Reference	82
6.16.1	Field Documentation	82
6.16.1.1	check_expiry_time	82
6.16.1.2	check_generation_location	82
6.16.1.3	check_generation_time	83
6.16.1.4	check_replay	83
6.16.1.5	verification_interval	83
6.17	connects_to Struct Reference	83
6.17.1	Detailed Description	83
6.17.2	Field Documentation	83
6.17.2.1	id	83
6.17.2.2	signal_group_id	83
6.18	datetime Struct Reference	83
6.19	encbrakestatus Struct Reference	84
6.20	encpadbytes Struct Reference	84
6.20.1	Detailed Description	84
6.21	encpointsetstype04 Struct Reference	84
6.22	encprndlspeed Struct Reference	85

6.23	entimestamp Struct Reference	85
6.23.1	Detailed Description	85
6.24	encvehiclesize Struct Reference	85
6.25	etsi_cam_dgc Struct Reference	86
6.26	etsi_cam_ec Struct Reference	86
6.27	etsi_cam_ptc Struct Reference	86
6.28	etsi_cam_rc Struct Reference	86
6.29	etsi_cam_rwc_basic Struct Reference	87
6.30	etsi_cam_scc Struct Reference	87
6.31	etsi_cam_stc Struct Reference	87
6.32	etsi_connects Struct Reference	87
6.32.1	Field Documentation	87
6.32.1.1	lane_id	87
6.32.1.2	maneuvers	88
6.32.1.3	signal_group_id	88
6.33	etsi_node Struct Reference	88
6.33.1	Field Documentation	88
6.33.1.1	node_attr	88
6.33.1.2	node_point	88
6.34	etsi_node_attr Struct Reference	88
6.35	etsi_pos3d Struct Reference	88
6.36	ETSIItineraryPath_t Struct Reference	89
6.37	ETSIRefDENMs_t Struct Reference	89
6.38	ETSIRestrictedTypes_t Struct Reference	89
6.39	intersec_access_pt Struct Reference	89
6.40	intersection_data Struct Reference	90
6.40.1	Detailed Description	90
6.40.2	Field Documentation	90
6.40.2.1	enabled_lanes	90
6.40.2.2	intersection_id	90
6.40.2.3	intersection_status	90
6.40.2.4	maneuver_assist	90
6.40.2.5	min_of_the_year	90
6.40.2.6	movement_states	90
6.40.2.7	msgCount	91
6.40.2.8	n_lane_list	91
6.40.2.9	num_maneuver_assist	91
6.40.2.10	num_movement_states	91
6.40.2.11	timestamp_ms	91
6.41	intersection_geodescr Struct Reference	91

6.41.1 Detailed Description . . . . .	91
6.41.2 Field Documentation . . . . .	91
6.41.2.1 intersection_id . . . . .	91
6.41.2.2 lanelist . . . . .	92
6.41.2.3 lanewidth . . . . .	92
6.41.2.4 msg_revision . . . . .	92
6.41.2.5 n_lanelist . . . . .	92
6.41.2.6 ref_pt . . . . .	92
6.42 J2735_2016_Computed_Lanes Struct Reference . . . . .	92
6.42.1 Detailed Description . . . . .	92
6.43 j2735_2016_fullposvec Struct Reference . . . . .	92
6.44 J2735_2016_GeometricProjection Struct Reference . . . . .	93
6.44.1 Detailed Description . . . . .	93
6.44.2 Field Documentation . . . . .	93
6.44.2.1 circle . . . . .	93
6.44.2.2 extent . . . . .	93
6.44.2.3 heading_slice . . . . .	93
6.44.2.4 lane_width . . . . .	93
6.45 J2735_2016_LatLong Struct Reference . . . . .	93
6.45.1 Detailed Description . . . . .	93
6.45.2 Field Documentation . . . . .	94
6.45.2.1 latlong_offsets . . . . .	94
6.46 J2735_2016_NodeSet_XY Struct Reference . . . . .	94
6.46.1 Detailed Description . . . . .	94
6.46.2 Field Documentation . . . . .	94
6.46.2.1 n_node_offsets . . . . .	94
6.46.2.2 node_offsets . . . . .	94
6.47 J2735_2016_NodeXY Struct Reference . . . . .	94
6.47.1 Detailed Description . . . . .	94
6.47.2 Field Documentation . . . . .	95
6.47.2.1 computed_lanes . . . . .	95
6.47.2.2 node_xy_type . . . . .	95
6.47.2.3 nodeset . . . . .	95
6.48 J2735_2016_OffsetSystem Struct Reference . . . . .	95
6.48.1 Detailed Description . . . . .	95
6.48.2 Field Documentation . . . . .	95
6.48.2.1 node_set_ll . . . . .	95
6.48.2.2 node_set_xy . . . . .	95
6.48.2.3 off_sys_type . . . . .	95
6.48.2.4 zoom_level . . . . .	96

6.49 J2735_2016_RegionOffsets Struct Reference . . . . .	96
6.49.1 Detailed Description . . . . .	96
6.50 J2735_2016_RequestorType_t Struct Reference . . . . .	96
6.51 J2735_2016_SignalRequest Struct Reference . . . . .	96
6.51.1 Detailed Description . . . . .	97
6.51.2 Field Documentation . . . . .	97
6.51.2.1 basic_vehicle_role . . . . .	97
6.51.2.2 msgcount . . . . .	97
6.51.2.3 request_id . . . . .	97
6.51.2.4 requestor_type . . . . .	97
6.51.2.5 requestor_type_present . . . . .	97
6.51.2.6 vehicle_id . . . . .	97
6.52 J2735_2016_SignalRequest_t Struct Reference . . . . .	97
6.52.1 Detailed Description . . . . .	98
6.52.2 Field Documentation . . . . .	98
6.52.2.1 iid . . . . .	98
6.52.2.2 in_bound_access_point . . . . .	98
6.52.2.3 out_bound_access_point . . . . .	98
6.52.2.4 req_type . . . . .	98
6.52.2.5 request_id . . . . .	98
6.53 J2735_2016_SignalRequestorInfo_t Struct Reference . . . . .	98
6.54 J2735_2016_SignalStatusList Struct Reference . . . . .	98
6.54.1 Detailed Description . . . . .	99
6.54.2 Field Documentation . . . . .	99
6.54.2.1 iid . . . . .	99
6.54.2.2 msgcount . . . . .	99
6.54.2.3 ss_pkg_list . . . . .	99
6.54.2.4 ss_pkg_list_len . . . . .	99
6.55 J2735_2016_SignalStatusPackageList Struct Reference . . . . .	99
6.55.1 Detailed Description . . . . .	99
6.55.2 Field Documentation . . . . .	99
6.55.2.1 duration . . . . .	99
6.55.2.2 in_bound_access_point . . . . .	100
6.55.2.3 min_of_year . . . . .	100
6.55.2.4 out_bound_access_point . . . . .	100
6.55.2.5 prio_resp . . . . .	100
6.55.2.6 second . . . . .	100
6.55.2.7 signal_request . . . . .	100
6.55.2.8 valid_out_bound_access_point . . . . .	100
6.56 j2735_2016_srm Struct Reference . . . . .	100

6.56.1 Detailed Description . . . . .	100
6.56.2 Field Documentation . . . . .	101
6.56.2.1 dsecond . . . . .	101
6.56.2.2 msg_count . . . . .	101
6.56.2.3 requestor . . . . .	101
6.56.2.4 srm_list_count . . . . .	101
6.56.2.5 srmlist . . . . .	101
6.57 j2735_2016_ssm Struct Reference . . . . .	101
6.57.1 Detailed Description . . . . .	101
6.57.2 Field Documentation . . . . .	101
6.57.2.1 dsecond . . . . .	101
6.57.2.2 msgcount . . . . .	101
6.57.2.3 sslist . . . . .	102
6.57.2.4 sslist_len . . . . .	102
6.58 J2735_2016_VehicleClassification_t Struct Reference . . . . .	102
6.59 J2735_2016_VehicleIdent_t Struct Reference . . . . .	102
6.60 J2735CircularRegion Struct Reference . . . . .	102
6.60.1 Detailed Description . . . . .	103
6.60.2 Field Documentation . . . . .	103
6.60.2.1 position . . . . .	103
6.60.2.2 radius_type . . . . .	103
6.60.2.3 value . . . . .	103
6.61 J2735Position3D Struct Reference . . . . .	103
6.61.1 Detailed Description . . . . .	103
6.61.2 Field Documentation . . . . .	103
6.61.2.1 elevation . . . . .	103
6.61.2.2 latitude . . . . .	104
6.61.2.3 longitude . . . . .	104
6.62 J2735RegionPointSet Struct Reference . . . . .	104
6.62.1 Field Documentation . . . . .	104
6.62.1.1 num_wp . . . . .	104
6.62.1.2 position . . . . .	104
6.62.1.3 reg_offsets . . . . .	104
6.62.1.4 region_offset_len . . . . .	104
6.62.1.5 wplist . . . . .	104
6.62.1.6 zoom_level . . . . .	104
6.63 J2735RoadSignID Struct Reference . . . . .	105
6.63.1 Detailed Description . . . . .	105
6.63.2 Field Documentation . . . . .	105
6.63.2.1 heading_slice . . . . .	105

6.63.2.2	mutcd_code	105
6.63.2.3	position	105
6.64	J2735ShapePointSet Struct Reference	105
6.64.1	Detailed Description	105
6.64.2	Field Documentation	106
6.64.2.1	lanewidth	106
6.64.2.2	num_wp	106
6.64.2.3	position	106
6.64.2.4	wplist	106
6.65	J2735TIM Struct Reference	106
6.65.1	Detailed Description	107
6.65.2	Field Documentation	107
6.65.2.1	advisory_codes	107
6.65.2.2	content_type	107
6.65.2.3	duration	107
6.65.2.4	exitsservice_codes	107
6.65.2.5	further_info_id	107
6.65.2.6	genericsign_codes	107
6.65.2.7	geographical_path	107
6.65.2.8	n_geo_paths	107
6.65.2.9	num_advisories	107
6.65.2.10	num_exitsservices	107
6.65.2.11	num_genericsigns	107
6.65.2.12	num_speedlimits	108
6.65.2.13	num_workzones	108
6.65.2.14	priority	108
6.65.2.15	road_sign_id	108
6.65.2.16	speedlimit_codes	108
6.65.2.17	ssp_loc_rights	108
6.65.2.18	ssp_msg_rights1	108
6.65.2.19	ssp_msg_rights2	108
6.65.2.20	ssp_tim_rights	108
6.65.2.21	start_minute_of_year	108
6.65.2.22	start_year	108
6.65.2.23	type	109
6.65.2.24	workzone_codes	109
6.66	J2735TIMGeoPath Struct Reference	109
6.66.1	Detailed Description	109
6.66.2	Field Documentation	109
6.66.2.1	closedpath	109

6.66.2.2	directionality	109
6.66.2.3	geometric_projection	109
6.66.2.4	heading_slice	110
6.66.2.5	id	110
6.66.2.6	lane_width	110
6.66.2.7	offset_system	110
6.66.2.8	pos3d	110
6.66.2.9	projection_type	110
6.66.2.10	valid_pos3d	110
6.66.2.11	valid_region	110
6.67	J2735ValidRegion Struct Reference	110
6.67.1	Detailed Description	111
6.67.2	Field Documentation	111
6.67.2.1	circle	111
6.67.2.2	extent	111
6.67.2.3	heading_slice	111
6.67.2.4	region_flag	111
6.67.2.5	regionpoint	111
6.67.2.6	shape_points	111
6.68	J2735VehicleID_t Struct Reference	111
6.69	lane_validity_params Struct Reference	112
6.70	lane_validity_time Struct Reference	112
6.71	laneattr Struct Reference	112
6.71.1	Detailed Description	112
6.71.2	Field Documentation	112
6.71.2.1	crwattr	112
6.71.2.2	direction	112
6.71.2.3	sharedlane	113
6.71.2.4	vehattr	113
6.72	lanedataattr_list Struct Reference	113
6.73	laneset Struct Reference	113
6.73.1	Detailed Description	113
6.73.2	Field Documentation	114
6.73.2.1	connects_to	114
6.73.2.2	lane_attr	114
6.73.2.3	lane_id	114
6.73.2.4	man	114
6.73.2.5	n_connect	114
6.73.2.6	n_nodepoint	114
6.73.2.7	node_point	114



6.73.2.8	phase	114
6.73.2.9	signal_group_id	114
6.74	libcan_filters Struct Reference	114
6.75	libwme_gpsinfo Struct Reference	115
6.76	libwme_radio_settings::libwme_radio Struct Reference	115
6.77	libwme_radio_settings Struct Reference	115
6.78	nodeattrs Struct Reference	115
6.78.1	Detailed Description	116
6.78.2	Field Documentation	116
6.78.2.1	dElev	116
6.78.2.2	dWidth	116
6.78.2.3	lane_data_attr	116
6.78.2.4	n_lane_data_attr	116
6.78.2.5	type	116
6.79	odelist Struct Reference	116
6.79.1	Detailed Description	116
6.79.2	Field Documentation	117
6.79.2.1	lanewidth	117
6.79.2.2	node	117
6.80	nodepoint_list Struct Reference	117
6.80.1	Detailed Description	117
6.80.2	Field Documentation	117
6.80.2.1	node_id	117
6.80.2.2	nodeattr	117
6.81	path_history Struct Reference	117
6.81.1	Detailed Description	118
6.81.2	Field Documentation	118
6.81.2.1	elev_offset	118
6.81.2.2	heading	118
6.81.2.3	lat_offset	118
6.81.2.4	long_offset	118
6.81.2.5	positional_accuracy	118
6.81.2.6	time_offset	118
6.82	pathhistoryframe Struct Reference	118
6.82.1	Detailed Description	119
6.82.2	Field Documentation	119
6.82.2.1	elev_offset	119
6.82.2.2	heading	119
6.82.2.3	lat_offset	119
6.82.2.4	long_offset	119

6.82.2.5	positional_accuracy	119
6.82.2.6	time_offset	119
6.83	pf_gnss_data Struct Reference	119
6.83.1	Detailed Description	120
6.83.2	Field Documentation	120
6.83.2.1	altitude	120
6.83.2.2	dsecond	120
6.83.2.3	heading	120
6.83.2.4	latitude	121
6.83.2.5	longitude	121
6.83.2.6	mode	121
6.83.2.7	speed	121
6.83.2.8	time	121
6.83.2.9	utc_time	121
6.84	position_confidence Struct Reference	121
6.85	positional_accuracy_data Struct Reference	121
6.86	positional_accuracy_data_2016 Struct Reference	122
6.87	road_segment Struct Reference	122
6.88	RoadSegmentRefId Struct Reference	122
6.88.1	Detailed Description	122
6.88.2	Field Documentation	123
6.88.2.1	region_id	123
6.88.2.2	roadsegment_id	123
6.89	savari1609Wra Struct Reference	123
6.89.1	Field Documentation	123
6.89.1.1	default_gw	123
6.89.1.2	gw_macaddr	123
6.89.1.3	ipv6addr	123
6.89.1.4	lifetime	123
6.89.1.5	prefixlen	123
6.89.1.6	primarydns	123
6.89.1.7	secondarydns	124
6.90	savari1609WsasService Struct Reference	124
6.90.1	Detailed Description	124
6.91	savari_antennaoffset Struct Reference	124
6.92	savari_rtcn_payload Struct Reference	125
6.93	security_cmd_profile_msg Struct Reference	125
6.93.1	Field Documentation	125
6.93.1.1	prof_type	125
6.93.1.2	sign_profile	125

6.93.1.3	verify_profile	125
6.94	spedelimits Struct Reference	125
6.95	srm_list Struct Reference	126
6.95.1	Detailed Description	126
6.95.2	Field Documentation	126
6.95.2.1	duration	126
6.95.2.2	min_of_year	126
6.95.2.3	second	126
6.95.2.4	signal_request	126
6.96	srm_requestor Struct Reference	126
6.96.1	Detailed Description	127
6.96.2	Field Documentation	127
6.96.2.1	full_pos_valid	127
6.96.2.2	id	127
6.96.2.3	req_pos_vector	127
6.96.2.4	requestor_type	127
6.96.2.5	requestor_type_valid	127
6.97	testdataframe Struct Reference	127
6.97.1	Field Documentation	128
6.97.1.1	channelmode	128
6.97.1.2	datarate	128
6.97.1.3	formatversion	128
6.97.1.4	msgrate	128
6.97.1.5	msgseqnum	128
6.97.1.6	numobe	128
6.97.1.7	obeid	128
6.97.1.8	testcheck	128
6.97.1.9	testconfiguration	128
6.98	timestamp Struct Reference	128
6.99	utcTimeFrame Struct Reference	129
6.99.1	Detailed Description	129
6.99.2	Field Documentation	129
6.99.2.1	day	129
6.99.2.2	hour	129
6.99.2.3	minute	129
6.99.2.4	month	129
6.99.2.5	second	129
6.99.2.6	year	129
6.100v2x	canctl Struct Reference	130
6.100.1	Detailed Description	130

6.100.2 Field Documentation	130
6.100.2.1 can_fd	130
6.101v2x_emergency_details Struct Reference	130
6.101.1 Detailed Description	130
6.101.2 Field Documentation	131
6.101.2.1 lightbar_in_use	131
6.101.2.2 multi_vehicle_response	131
6.101.2.3 siren_in_use	131
6.101.2.4 ssp_rights	131
6.102v2x_etsi_cam Struct Reference	131
6.102.1 Detailed Description	131
6.102.2 Field Documentation	131
6.102.2.1 bc	131
6.102.2.2 generation_delta_time	132
6.102.2.3 hf	132
6.102.2.4 lf	132
6.102.2.5 pdu	132
6.102.2.6 type	132
6.103v2x_etsi_cam_bc Struct Reference	132
6.103.1 Field Documentation	132
6.103.1.1 station_position	132
6.103.1.2 station_type	132
6.104v2x_etsi_cam_hf Struct Reference	132
6.104.1 Field Documentation	133
6.104.1.1 rsu	133
6.104.1.2 type	133
6.104.1.3 vehicle	133
6.105v2x_etsi_cam_hf_rsu Struct Reference	133
6.105.1 Field Documentation	133
6.105.1.1 n_zones	133
6.105.1.2 zonelist	133
6.106v2x_etsi_cam_hf_vehicle Struct Reference	133
6.106.1 Field Documentation	134
6.106.1.1 accel_ctrl	134
6.106.1.2 curvature	134
6.106.1.3 curvature_calc	134
6.106.1.4 curvature_conf	134
6.106.1.5 drive_direction	134
6.106.1.6 heading	134
6.106.1.7 heading_acc	135

6.106.1.8 lane_position . . . . .	135
6.106.1.9 lateral_accel . . . . .	135
6.106.1.10 lateral_accel_conf . . . . .	135
6.106.1.11 longitude_accel . . . . .	135
6.106.1.12 longitude_accel_conf . . . . .	135
6.106.1.13 performance_class . . . . .	135
6.106.1.14 speed . . . . .	135
6.106.1.15 speed_conf . . . . .	135
6.106.1.16 steeringwhl_angle . . . . .	135
6.106.1.17 steeringwhl_angle_conf . . . . .	135
6.106.1.18 tollzone_id . . . . .	135
6.106.1.19 tollzone_lat . . . . .	136
6.106.1.20 tollzone_long . . . . .	136
6.106.1.21 vehicle_confidence . . . . .	136
6.106.1.22 vehicle_length . . . . .	136
6.106.1.23 vehicle_width . . . . .	136
6.106.1.24 vertical_accel . . . . .	136
6.106.1.25 vertical_accel_conf . . . . .	136
6.106.1.26 yawrate . . . . .	136
6.106.1.27 yawrate_conf . . . . .	136
6.107v2x_etsi_cam_If Struct Reference . . . . .	136
6.107.1 Field Documentation . . . . .	137
6.107.1.1 ext_lights . . . . .	137
6.107.1.2 ph . . . . .	137
6.107.1.3 role . . . . .	137
6.108v2x_etsi_denm Struct Reference . . . . .	137
6.108.1 Detailed Description . . . . .	137
6.108.2 Field Documentation . . . . .	137
6.108.2.1 ac . . . . .	137
6.108.2.2 lc . . . . .	137
6.108.2.3 mc . . . . .	137
6.108.2.4 pdu . . . . .	138
6.108.2.5 sc . . . . .	138
6.108.2.6 type . . . . .	138
6.109v2x_etsi_denm_ac Struct Reference . . . . .	138
6.109.1 Detailed Description . . . . .	138
6.109.2 Field Documentation . . . . .	138
6.109.2.1 bit_mask . . . . .	138
6.109.2.2 rwc . . . . .	138
6.110v2x_etsi_denm_lc Struct Reference . . . . .	138

6.110.1 Detailed Description . . . . .	139
6.110.2 Field Documentation . . . . .	139
6.110.2.1 event_position_heading . . . . .	139
6.110.2.2 event_position_heading_conf . . . . .	139
6.110.2.3 event_speed . . . . .	139
6.110.2.4 event_speed_conf . . . . .	139
6.110.2.5 event_traces . . . . .	139
6.110.2.6 road_type . . . . .	139
6.111v2x_etsi_denm_mc Struct Reference . . . . .	139
6.111.1 Detailed Description . . . . .	140
6.111.2 Field Documentation . . . . .	140
6.111.2.1 action_id . . . . .	140
6.111.2.2 detection_time . . . . .	140
6.111.2.3 event_position . . . . .	140
6.111.2.4 reference_time . . . . .	140
6.111.2.5 relevance_distance . . . . .	140
6.111.2.6 relevance_traf_dir . . . . .	140
6.111.2.7 station_type . . . . .	140
6.111.2.8 termination . . . . .	140
6.111.2.9 transmit_interval . . . . .	141
6.111.2.10 validity_duration . . . . .	141
6.112v2x_etsi_denm_rwc Struct Reference . . . . .	141
6.112.1 Detailed Description . . . . .	141
6.112.2 Field Documentation . . . . .	141
6.112.2.1 bit_mask . . . . .	141
6.112.2.2 closed_lanes . . . . .	141
6.112.2.3 incident_indication . . . . .	141
6.112.2.4 itinerary_path . . . . .	141
6.112.2.5 lb_siren . . . . .	142
6.112.2.6 ref_denms . . . . .	142
6.112.2.7 restricted_types . . . . .	142
6.112.2.8 speed_limit . . . . .	142
6.112.2.9 starting_point_splimit . . . . .	142
6.112.2.10 traffic_rule . . . . .	142
6.113v2x_etsi_denm_sc Struct Reference . . . . .	142
6.113.1 Detailed Description . . . . .	142
6.113.2 Field Documentation . . . . .	142
6.113.2.1 event_history_list . . . . .	142
6.113.2.2 event_type . . . . .	143
6.113.2.3 information_quality . . . . .	143

6.113.2.4 linked_cause . . . . .	143
6.114v2x_etsi_lane Struct Reference . . . . .	143
6.114.1 Field Documentation . . . . .	143
6.114.1.1 approach_type . . . . .	143
6.114.1.2 connects . . . . .	143
6.114.1.3 connects_len . . . . .	143
6.114.1.4 egress_approach . . . . .	143
6.114.1.5 ingress_approach . . . . .	144
6.114.1.6 lane_id . . . . .	144
6.114.1.7 laneattr . . . . .	144
6.114.1.8 maneuvers . . . . .	144
6.114.1.9 nodes . . . . .	144
6.114.1.10nodes_len . . . . .	144
6.114.1.11phase_no . . . . .	144
6.114.1.12signal_group_id . . . . .	144
6.115v2x_etsi_lane_attr Struct Reference . . . . .	144
6.116v2x_etsi_map Struct Reference . . . . .	145
6.116.1 Detailed Description . . . . .	145
6.116.2 Field Documentation . . . . .	145
6.116.2.1 map_msg . . . . .	145
6.116.2.2 pdu . . . . .	145
6.117v2x_etsi_map_isec Struct Reference . . . . .	145
6.117.1 Field Documentation . . . . .	145
6.117.1.1 iid . . . . .	145
6.117.1.2 laneset . . . . .	145
6.117.1.3 laneset_len . . . . .	146
6.117.1.4 lanewidth . . . . .	146
6.117.1.5 msg_count . . . . .	146
6.117.1.6 ref_pt . . . . .	146
6.117.1.7 region . . . . .	146
6.118v2x_etsi_map_msg Struct Reference . . . . .	146
6.118.1 Field Documentation . . . . .	146
6.118.1.1 ilist . . . . .	146
6.118.1.2 ilist_len . . . . .	146
6.118.1.3 msg_count . . . . .	146
6.118.1.4 msg_subid . . . . .	147
6.119v2x_etsi_protected_zone Struct Reference . . . . .	147
6.120v2x_etsi_spat Struct Reference . . . . .	147
6.120.1 Detailed Description . . . . .	147
6.120.2 Field Documentation . . . . .	147

6.120.2.1 pdu . . . . .	147
6.120.2.2 spat_msg . . . . .	147
6.121v2x_etsi_spat_isec Struct Reference . . . . .	148
6.121.1 Field Documentation . . . . .	148
6.121.1.1 dsecond . . . . .	148
6.121.1.2 enabled_lanes . . . . .	148
6.121.1.3 enabled_lanes_len . . . . .	148
6.121.1.4 isec_id . . . . .	148
6.121.1.5 isec_status . . . . .	148
6.121.1.6 min_of_yr . . . . .	148
6.121.1.7 mmlist . . . . .	148
6.121.1.8 mmlist_len . . . . .	148
6.121.1.9 msg_count . . . . .	149
6.122v2x_etsi_spat_mm_evt Struct Reference . . . . .	149
6.122.1 Field Documentation . . . . .	149
6.122.1.1 confidence . . . . .	149
6.122.1.2 likely_time . . . . .	149
6.122.1.3 max_end_time . . . . .	149
6.122.1.4 min_end_time . . . . .	149
6.122.1.5 next_time . . . . .	149
6.122.1.6 phase_state . . . . .	149
6.122.1.7 start_time . . . . .	149
6.123v2x_etsi_spat_mm_state Struct Reference . . . . .	150
6.123.1 Field Documentation . . . . .	150
6.123.1.1 mm_evt . . . . .	150
6.123.1.2 mm_evt_len . . . . .	150
6.123.1.3 signal_group_id . . . . .	150
6.124v2x_etsi_spat_msg Struct Reference . . . . .	150
6.124.1 Field Documentation . . . . .	150
6.124.1.1 isec_list . . . . .	150
6.124.1.2 isec_list_len . . . . .	150
6.124.1.3 spat_msg_subid . . . . .	151
6.125v2x_msg_bsm Struct Reference . . . . .	151
6.125.1 Detailed Description . . . . .	152
6.125.2 Field Documentation . . . . .	152
6.125.2.1 abs . . . . .	152
6.125.2.2 angle . . . . .	152
6.125.2.3 autolightcontrol . . . . .	153
6.125.2.4 auxbrakes . . . . .	153
6.125.2.5 brakeboost . . . . .	153



6.125.2.6 bumperheight_front . . . . .	153
6.125.2.7 bumperheight_rear . . . . .	153
6.125.2.8 confidence . . . . .	153
6.125.2.9 count . . . . .	153
6.125.2.10dtimerunlights . . . . .	153
6.125.2.11elev_confidence . . . . .	153
6.125.2.12elevation . . . . .	153
6.125.2.13emergency_type_event . . . . .	153
6.125.2.14event_absactivate . . . . .	153
6.125.2.15event_airbag . . . . .	154
6.125.2.16event_descr . . . . .	154
6.125.2.17event_descr_len . . . . .	154
6.125.2.18event_hazardbraking . . . . .	154
6.125.2.19event_hazardlights . . . . .	154
6.125.2.20event_stabilityctrlactivated . . . . .	154
6.125.2.21event_tractionctrlloss . . . . .	154
6.125.2.22foglights . . . . .	154
6.125.2.23gpsstatus . . . . .	154
6.125.2.24hazardlights . . . . .	154
6.125.2.25head_confidence . . . . .	154
6.125.2.26heading . . . . .	154
6.125.2.27highbeam . . . . .	155
6.125.2.28ataccel . . . . .	155
6.125.2.29atitude . . . . .	155
6.125.2.30leftturnsignal . . . . .	155
6.125.2.31length . . . . .	155
6.125.2.32lightbar_in_use . . . . .	155
6.125.2.33lightbarinuse . . . . .	155
6.125.2.34ongaccel . . . . .	155
6.125.2.35ongitude . . . . .	155
6.125.2.36owbeam . . . . .	155
6.125.2.37msgcount . . . . .	155
6.125.2.38msgid . . . . .	155
6.125.2.39parkinglights . . . . .	156
6.125.2.40phf . . . . .	156
6.125.2.41pos_confidence . . . . .	156
6.125.2.42positionalaccuracy . . . . .	156
6.125.2.43radius . . . . .	156
6.125.2.44rightturnsignal . . . . .	156
6.125.2.45secmark . . . . .	156

6.125.2.46speed	156
6.125.2.47speed_confidence	156
6.125.2.48stabilitycontrol	156
6.125.2.49df	156
6.125.2.50temp_id	156
6.125.2.51throtl_confidence	157
6.125.2.52throttle_pos	157
6.125.2.53time_confidence	157
6.125.2.54traction	157
6.125.2.55transmissionstate	157
6.125.2.56time	157
6.125.2.57vehicleheight	157
6.125.2.58vehiclemass	157
6.125.2.59vehicletype	157
6.125.2.60vertaccel	157
6.125.2.61wheelbrake	157
6.125.2.62wheelbrakeavailable	157
6.125.2.63width	158
6.125.2.64wipers_rtfmt	158
6.125.2.65wipers_rtrear	158
6.125.2.66wipers_swfmt	158
6.125.2.67wipers_swrear	158
6.125.2.68yawrate	158
6.126v2x_msg_eva Struct Reference	158
6.126.1 Detailed Description	158
6.126.2 Field Documentation	158
6.126.2.1 emergency_details	158
6.126.2.2 emergency_details_avail	159
6.126.2.3 resp_type	159
6.126.2.4 rsa_msg	159
6.126.2.5 tempid	159
6.127v2x_msg_map Struct Reference	159
6.127.1 Detailed Description	159
6.127.2 Field Documentation	159
6.127.2.1 isec_geo_descr	159
6.127.2.2 msg_revision	159
6.127.2.3 n_geo_descr	159
6.128v2x_msg_psm Struct Reference	160
6.128.1 Detailed Description	160
6.128.2 Field Documentation	160

6.128.2.1 confidence	160
6.128.2.2 cross_request	160
6.128.2.3 device_usage_state	160
6.128.2.4 elevation	160
6.128.2.5 heading	160
6.128.2.6 latitude	161
6.128.2.7 longitude	161
6.128.2.8 msgcount	161
6.128.2.9 personal_device_use_type	161
6.128.2.10 pos_accuracy	161
6.128.2.11 radius	161
6.128.2.12 secmark	161
6.128.2.13 speed	161
6.128.2.14 temp_id	161
6.129v2x_msg_rtcn Struct Reference	161
6.129.1 Detailed Description	162
6.130v2x_msg_spat Struct Reference	162
6.130.1 Detailed Description	162
6.130.2 Field Documentation	162
6.130.2.1 Intersection_list	162
6.130.2.2 minute_of_the_year	162
6.130.2.3 n_intersections	162
6.131v2x_msg_tim Struct Reference	162
6.131.1 Detailed Description	163
6.131.2 Field Documentation	164
6.131.2.1 dataframes	164
6.131.2.2 hopinfo	164
6.131.2.3 message_id	164
6.131.2.4 msg_crc	164
6.131.2.5 num_dataframes	164
6.131.2.6 unique_id	164
6.132v2x_wme_reg_req Struct Reference	164
6.132.1 Detailed Description	165
6.132.2 Field Documentation	165
6.132.2.1 channel	165
6.132.2.2 channel_access	165
6.132.2.3 destmacaddr	165
6.132.2.4 extended_access	165
6.132.2.5 immediate_access	165
6.132.2.6 local_service_index	165

6.132.2.7 priority . . . . .	165
6.132.2.8 psc . . . . .	165
6.132.2.9 psc_length . . . . .	166
6.132.2.10psid . . . . .	166
6.132.2.11repeatrate . . . . .	166
6.132.2.12request_type . . . . .	166
6.132.2.13secondradio . . . . .	166
6.132.2.14service_ipv6addr . . . . .	166
6.132.2.15service_port . . . . .	166
6.132.2.16srcmacaddr . . . . .	166
6.132.2.17wsatype . . . . .	166
6.133v2x_wsmp_callbacks Struct Reference . . . . .	167
6.133.1 Detailed Description . . . . .	167
6.133.2 Field Documentation . . . . .	167
6.133.2.1 wme_cch_confirm . . . . .	167
6.133.2.2 wme_connected . . . . .	168
6.133.2.3 wme_disconnected . . . . .	168
6.133.2.4 wme_get_rse_info . . . . .	168
6.133.2.5 wme_provider_confirm . . . . .	168
6.133.2.6 wme_user_confirm . . . . .	169
6.133.2.7 wme_wsm_indication . . . . .	169
6.134v2x_wsmp_rcv_indication Struct Reference . . . . .	169
6.134.1 Detailed Description . . . . .	170
6.134.2 Field Documentation . . . . .	170
6.134.2.1 channel . . . . .	170
6.134.2.2 datarate . . . . .	170
6.134.2.3 num_rx . . . . .	170
6.134.2.4 plcp_length . . . . .	170
6.134.2.5 priority . . . . .	170
6.134.2.6 psid . . . . .	170
6.134.2.7 rssi . . . . .	170
6.134.2.8 rx_buf . . . . .	170
6.134.2.9 rx_buf_length . . . . .	170
6.134.2.10rx_mac . . . . .	170
6.134.2.11rx_supp . . . . .	170
6.134.2.12stamp . . . . .	171
6.134.2.13xpower . . . . .	171
6.134.2.14version . . . . .	171
6.135v2x_wsmp_send_req Struct Reference . . . . .	171
6.135.1 Detailed Description . . . . .	171

6.135.2 Field Documentation	171
6.135.2.1 channel	171
6.135.2.2 datarate	171
6.135.2.3 element_id	172
6.135.2.4 expiry_time	172
6.135.2.5 header_ext	172
6.135.2.6 mac	172
6.135.2.7 priority	172
6.135.2.8 psid	172
6.135.2.9 safetysupp	172
6.135.2.10srcmac	172
6.135.2.11supp_enable	172
6.135.2.12x_length	172
6.135.2.13xpower	172
6.136 vendor_frame Struct Reference	173
6.136.1 Detailed Description	173
<b>7 File Documentation</b>	<b>175</b>
7.1 v2x_cancel.h File Reference	175
7.1.1 Detailed Description	175
7.2 v2x_debug.h File Reference	176
7.2.1 Detailed Description	176
7.3 v2x_error.h File Reference	176
7.3.1 Detailed Description	177
7.4 v2x_msg_bsm.h File Reference	177
7.4.1 Detailed Description	179
7.4.2 Macro Definition Documentation	179
7.4.2.1 ASN1_BSM_MSG_ID	179
7.4.3 Enumeration Type Documentation	179
7.4.3.1 BSM_TYPE	179
7.4.3.2 BSMAbsActive_t	179
7.4.3.3 BSMBrakeAppliedStatus_t	180
7.4.3.4 BSMStabilityControlStatus_t	180
7.4.3.5 BSMTractionControl_t	180
7.4.3.6 BSMWiperStatus_t	180
7.5 v2x_msg_cam.h File Reference	180
7.5.1 Detailed Description	182
7.5.2 Macro Definition Documentation	182
7.5.2.1 V2X_ETSI_BC_PRESENT	182
7.5.2.2 V2X_ETSI_SET_BC	182

7.6	<a href="#">v2x_msg_common.h File Reference</a>	183
7.6.1	<a href="#">Detailed Description</a>	197
7.6.2	<a href="#">Enumeration Type Documentation</a>	197
7.6.2.1	<a href="#">BSMPartII_ID_Types_t</a>	197
7.6.2.2	<a href="#">laneshare</a>	198
7.7	<a href="#">v2x_msg_denm.h File Reference</a>	198
7.7.1	<a href="#">Detailed Description</a>	200
7.7.2	<a href="#">Macro Definition Documentation</a>	200
7.7.2.1	<a href="#">V2X_ETSI_DENM_HAS_MC</a>	200
7.7.2.2	<a href="#">V2X_ETSI_IS_DENM_HAS_MC</a>	200
7.7.2.3	<a href="#">V2X_ETSI_RESET_DENM_SC</a>	200
7.8	<a href="#">v2x_msg_etsi_map.h File Reference</a>	200
7.8.1	<a href="#">Detailed Description</a>	203
7.9	<a href="#">v2x_msg_etsi_spat.h File Reference</a>	203
7.9.1	<a href="#">Detailed Description</a>	205
7.10	<a href="#">v2x_msg_eva.h File Reference</a>	206
7.10.1	<a href="#">Detailed Description</a>	206
7.10.2	<a href="#">Typedef Documentation</a>	206
7.10.2.1	<a href="#">V2XEvaEmergencyDetails_t</a>	206
7.10.3	<a href="#">Enumeration Type Documentation</a>	207
7.10.3.1	<a href="#">v2x_eva_response_type_t</a>	207
7.11	<a href="#">v2x_msg_frame.h File Reference</a>	207
7.11.1	<a href="#">Detailed Description</a>	207
7.12	<a href="#">v2x_msg_map.h File Reference</a>	207
7.12.1	<a href="#">Detailed Description</a>	208
7.12.2	<a href="#">Typedef Documentation</a>	208
7.12.2.1	<a href="#">J2735_2016_lsecGeoDescr_t</a>	208
7.13	<a href="#">v2x_msg_psm.h File Reference</a>	209
7.13.1	<a href="#">Detailed Description</a>	209
7.13.2	<a href="#">Typedef Documentation</a>	210
7.13.2.1	<a href="#">personal_device_usage_state_t</a>	210
7.13.2.2	<a href="#">personal_device_user_type_t</a>	210
7.13.2.3	<a href="#">v2x_msg_psm_t</a>	210
7.13.3	<a href="#">Enumeration Type Documentation</a>	210
7.13.3.1	<a href="#">personal_device_usage_state</a>	210
7.13.3.2	<a href="#">personal_device_user_type</a>	210
7.14	<a href="#">v2x_msg_pvd.h File Reference</a>	210
7.14.1	<a href="#">Detailed Description</a>	212
7.14.2	<a href="#">Typedef Documentation</a>	212
7.14.2.1	<a href="#">J2735_2016_PHObject_t</a>	212

7.14.2.2	J2735_2016_ProbeSnapshot_t	212
7.14.2.3	J2735_2016_VehicleStatus_t	212
7.14.2.4	J2735_2016_VehSafetyExtensions_t	212
7.14.2.5	v2x_msg_pvd_t	212
7.14.3	Enumeration Type Documentation	212
7.14.3.1	J2735_2016_GPSStatus_t	212
7.15	v2x_msg_rtcn.h File Reference	213
7.15.1	Detailed Description	214
7.16	v2x_msg_spat.h File Reference	214
7.16.1	Detailed Description	216
7.16.2	Macro Definition Documentation	216
7.16.2.1	J2735_PED_DETECT_TRUE	216
7.16.2.2	SPAT_MAX_MMT_STATES	216
7.16.3	Typedef Documentation	216
7.16.3.1	J2735_2016_EnableLaneList_t	216
7.16.3.2	J2735_2016_Intersection_t	216
7.16.3.3	J2735_2016_IntersectionStatus_t	217
7.16.3.4	J2735_2016_ManeuverAssistList_t	217
7.16.3.5	J2735_2016_MovementEventList_t	217
7.16.3.6	J2735_2016_MovementState_t	217
7.16.3.7	J2735_2016_TimeChange_t	217
7.16.4	Enumeration Type Documentation	217
7.16.4.1	_J2735_2016_MovementPhaseState_t	217
7.16.4.2	intersection_status	217
7.16.5	Function Documentation	218
7.16.5.1	__attribute__	218
7.16.6	Variable Documentation	218
7.16.6.1	J2735_2016_PHASE_STATE_CAUTION_CONFLICTING_TRAFFIC	218
7.16.6.2	J2735_2016_PHASE_STATE_DARK	218
7.16.6.3	J2735_2016_PHASE_STATE_PERMISSIVE_CLEARANCE	218
7.16.6.4	J2735_2016_PHASE_STATE_PERMISSIVE_MOVEMENT_ALLOWED	218
7.16.6.5	J2735_2016_PHASE_STATE_PRE_MOVEMENT	218
7.16.6.6	J2735_2016_PHASE_STATE_PROTECTED_CLEARANCE	219
7.16.6.7	J2735_2016_PHASE_STATE_PROTECTED_MOVEMENT_ALLOWED	219
7.16.6.8	J2735_2016_PHASE_STATE_STOP_AND_REMAIN	219
7.16.6.9	J2735_2016_PHASE_STATE_STOP_THEN_PROCEED	219
7.16.6.10	J2735_2016_PHASE_STATE_UNKNOWN	219
7.17	v2x_msg_srm.h File Reference	219
7.17.1	Detailed Description	220
7.17.2	Typedef Documentation	220

7.17.2.1	J2735SRMList_t	220
7.17.2.2	J2735SRMRequestor_t	221
7.17.3	Enumeration Type Documentation	221
7.17.3.1	J2735_2016_PriorityReqType_t	221
7.18	v2x_msg_ssm.h File Reference	221
7.18.1	Detailed Description	222
7.18.2	Typedef Documentation	222
7.18.2.1	J2735_2016_SignalRequesterInfo_t	222
7.18.2.2	J2735_2016_SignalStatusList_t	222
7.18.2.3	J2735_2016_SignalStatusPackageList_t	222
7.19	v2x_msg_tim.h File Reference	222
7.19.1	Detailed Description	229
7.19.2	Macro Definition Documentation	229
7.19.2.1	j2735_free_tim_contents_only	229
7.19.2.2	j2735_free_tim_struct	229
7.19.3	Typedef Documentation	229
7.19.3.1	J2735_2016_Computed_Lanes_t	229
7.19.3.2	J2735_2016_GeometricProjection_t	229
7.19.3.3	J2735_2016_LatLong_t	229
7.19.3.4	J2735_2016_NodeSet_XY_t	229
7.19.3.5	J2735_2016_NodeXY_t	229
7.19.3.6	J2735_2016_OffsetSystem_t	229
7.19.3.7	J2735CircularRegion_t	230
7.19.3.8	J2735ContentType_t	230
7.19.3.9	J2735DATAFRAME_t	230
7.19.3.10	J2735DirectionOfUse_t	230
7.19.3.11	J2735MUTCDCode_t	230
7.19.3.12	J2735RadiusType_t	230
7.19.3.13	J2735RegionsList_t	230
7.19.3.14	J2735RoadSignID_t	230
7.19.3.15	J2735ShapePoints_t	230
7.19.3.16	J2735TIMGeoPath_t	230
7.19.3.17	J2735TravelerInfoType_t	230
7.19.3.18	J2735ValidRegion_t	231
7.19.3.19	RoadSegmentRefId_t	231
7.19.4	Enumeration Type Documentation	231
7.19.4.1	J2735_2016_NodeXY_type_t	231
7.19.4.2	J2735_2016_OffsetSystemType_t	231
7.19.4.3	J2735_2016_ProjectionType_t	231
7.19.4.4	J2735ContentType	231



7.19.4.5	J2735DirectionOfUse	232
7.19.4.6	J2735HeadingSlice	232
7.19.4.7	J2735MUTCDCode	232
7.19.4.8	J2735RadiusType	232
7.19.4.9	J2735Regions	233
7.19.4.10	J2735TravelerInfoType	233
7.19.5	Function Documentation	233
7.19.5.1	__attribute__	233
7.19.5.2	free_tim	233
7.19.6	Variable Documentation	233
7.19.6.1	J2735_HS_ALL_HEADINGS	233
7.19.6.2	J2735_HS_FROM000_0TO022_5DEGREES	233
7.19.6.3	J2735_HS_FROM022_5TO045_0DEGREES	233
7.19.6.4	J2735_HS_FROM045_0TO067_5DEGREES	234
7.19.6.5	J2735_HS_FROM067_5TO090_0DEGREES	234
7.19.6.6	J2735_HS_FROM090_0TO112_5DEGREES	234
7.19.6.7	J2735_HS_FROM112_5TO135_0DEGREES	234
7.19.6.8	J2735_HS_FROM135_0TO157_5DEGREES	234
7.19.6.9	J2735_HS_FROM157_5TO180_0DEGREES	234
7.19.6.10	J2735_HS_FROM180_0TO202_5DEGREES	234
7.19.6.11	J2735_HS_FROM202_5TO225_0DEGREES	234
7.19.6.12	J2735_HS_FROM225_0TO247_5DEGREES	234
7.19.6.13	J2735_HS_FROM247_5TO270_0DEGREES	234
7.19.6.14	J2735_HS_FROM270_0TO292_5DEGREES	234
7.19.6.15	J2735_HS_FROM292_5TO315_0DEGREES	234
7.19.6.16	J2735_HS_FROM315_0TO337_5DEGREES	235
7.19.6.17	J2735_HS_FROM337_5TO360_0DEGREES	235
7.19.6.18	J2735_HS_NO_HEADING	235
7.20	v2x_net_btp.h File Reference	235
7.20.1	Detailed Description	235
7.20.2	Macro Definition Documentation	235
7.20.2.1	v2x_btp_get_data	235
7.20.2.2	v2x_btp_get_txreq	236
7.20.2.3	v2x_btp_packet_len	236
7.20.2.4	v2x_btp_reserve	236
7.20.2.5	v2x_btp_reserve_data	236
7.21	v2x_net_wme.h File Reference	236
7.21.1	Detailed Description	240
7.21.2	Macro Definition Documentation	240
7.21.2.1	ADVERTISER_LEN	240

7.21.2.2	FAIL	240
7.21.2.3	LIBWME_CHAN_MODE_ALT	240
7.21.2.4	LIBWME_CHAN_MODE_CONT	240
7.21.2.5	LIBWME_PSC_STRING_LENGTH	240
7.21.2.6	P16093_RADIO_INTF1	240
7.21.2.7	P16093_RADIO_INTF2	240
7.21.2.8	P16093_RADIO_INTFX	241
7.21.2.9	P16093_SRV_IP	241
7.21.2.10	P1609_ADVID_LEN_MAX	241
7.21.2.11	PSID_LEN	241
7.21.2.12	savariwme_cbs	241
7.21.2.13	savariwme_reg_req	241
7.21.2.14	savariwme_rx_indication	241
7.21.2.15	savariwme_tx_req	241
7.21.2.16	SSI_LEN	241
7.21.2.17	SUCCESS	241
7.21.3	Function Documentation	241
7.21.3.1	v2x_wsmp_close	241
7.21.3.2	wme_getpsidlen	242
7.21.3.3	wme_process_wme_cmd	242
7.21.3.4	wme_provider_service_confirm	242
7.21.3.5	wme_register_cch_request	242
7.21.3.6	wme_register_provider	243
7.21.3.7	wme_send_wmectl	243
7.21.3.8	wme_set_advertiserid	243
7.21.3.9	wme_set_gpslocation	243
7.21.3.10	wme_set_inactivity_timer	243
7.21.3.11	wme_set_wsaintvl	244
7.21.3.12	wme_set_wsatxpower	244
7.21.3.13	wme_unregister_cch_request	244
7.21.3.14	wme_unregister_provider	244
7.21.4	Variable Documentation	245
7.21.4.1	channel	245
7.21.4.2	channel_access	245
7.21.4.3	datarate	245
7.21.4.4	default_gw	245
7.21.4.5	destmacaddr	245
7.21.4.6	element_id	245
7.21.4.7	expiry_time	245
7.21.4.8	extended_access	245

7.21.4.9 gw_macaddr . . . . .	245
7.21.4.10 header_ext . . . . .	246
7.21.4.11 immediate_access . . . . .	246
7.21.4.12 ipv6addr . . . . .	246
7.21.4.13 lifetime . . . . .	246
7.21.4.14 local_service_index . . . . .	246
7.21.4.15 mac . . . . .	246
7.21.4.16 num_rx . . . . .	246
7.21.4.17 plcp_length . . . . .	246
7.21.4.18 prefixlen . . . . .	246
7.21.4.19 primarydns . . . . .	246
7.21.4.20 priority . . . . .	246
7.21.4.21 psc . . . . .	247
7.21.4.22 psc_length . . . . .	247
7.21.4.23 psid . . . . .	247
7.21.4.24 repeatrate . . . . .	247
7.21.4.25 request_type . . . . .	247
7.21.4.26 rssi . . . . .	247
7.21.4.27 rx_buf . . . . .	247
7.21.4.28 rx_buf_length . . . . .	247
7.21.4.29 rx_mac . . . . .	247
7.21.4.30 rx_supp . . . . .	248
7.21.4.31 safetysupp . . . . .	248
7.21.4.32 secondarydns . . . . .	248
7.21.4.33 secondradio . . . . .	248
7.21.4.34 service_ipv6addr . . . . .	248
7.21.4.35 service_port . . . . .	248
7.21.4.36 srcmac . . . . .	248
7.21.4.37 srcmacaddr . . . . .	248
7.21.4.38 supp_enable . . . . .	248
7.21.4.39 tstamp . . . . .	248
7.21.4.40 tx_length . . . . .	248
7.21.4.41 txpower . . . . .	249
7.21.4.42 version . . . . .	249
7.21.4.43 wsatype . . . . .	249
7.22 v2x_security.h File Reference . . . . .	249
7.22.1 Detailed Description . . . . .	249
7.23 v2x_security_base.h File Reference . . . . .	249
7.23.1 Detailed Description . . . . .	251
7.23.2 Enumeration Type Documentation . . . . .	251

7.23.2.1	v2x_sec_signer_id_t . . . . .	251
7.23.3	Variable Documentation . . . . .	251
7.23.3.1	cert_attach_rate . . . . .	251
7.23.3.2	check_expiry_time . . . . .	251
7.23.3.3	check_generation_location . . . . .	251
7.23.3.4	check_generation_time . . . . .	251
7.23.3.5	check_replay . . . . .	251
7.23.3.6	prof_type . . . . .	251
7.23.3.7	sign_profile . . . . .	251
7.23.3.8	use_expiry_time . . . . .	251
7.23.3.9	use_generation_loc . . . . .	252
7.23.3.10	use_generation_time . . . . .	252
7.23.3.11	verification_interval . . . . .	252
7.23.3.12	verify_profile . . . . .	252
7.24	v2x_sensor_can.h File Reference . . . . .	252
7.24.1	Detailed Description . . . . .	252
7.25	v2x_sensor_gnss.h File Reference . . . . .	252
7.25.1	Detailed Description . . . . .	253
7.25.2	Macro Definition Documentation . . . . .	253
7.25.2.1	FAIL . . . . .	253
7.25.2.2	SUCCESS . . . . .	254
7.25.3	Function Documentation . . . . .	254
7.25.3.1	pf_gnss_get_nmea . . . . .	254
7.25.3.2	pf_gnss_get_pfname . . . . .	254
<b>Index</b>		<b>255</b>

# Chapter 1

## Savari V2X Libraries

### 1.1 Introduction

Savari V2X Libraries consists of set libraries that helps in development of V2V and V2I applications including safety, mobility and other V2X applications.

Following figure depicts the Savari V2X library modules and its relationship with application and other library modules.

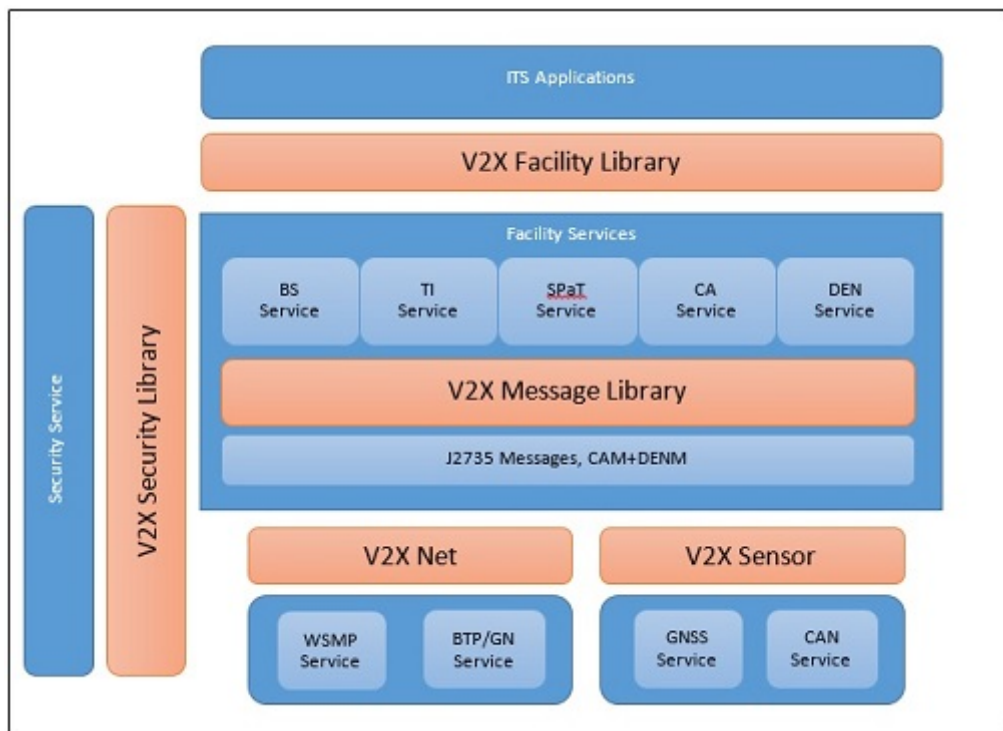


Figure 1.1: Savari V2X Libraries

The Savari V2X libraries are broadly classified as follows,

- V2X Common Library
- V2X Message Library
- V2X Facility Services Library

- V2X Security Service Library
- V2X Transport and Network Service Library
- V2X Sensor Service Library

## 1.2 V2X Common Library

The V2X common library provides set of API functions and definitions that are common to other V2X libraries and application. The V2X common library and its header files define common V2X API function return status values, error codes, definitions, enumerations, debugging API functions and related API functions.

The V2X common library API functions and data structures are defined in following files,

- [v2x\\_error.h](#)
- [v2x\\_debug.h](#)

Applications include above mentioned header files and link with libv2x\_common.so shared library.

## 1.3 V2X Message Library

The V2X message library provides functions to generate SAE J2735 encoded messages for transmission and decode received SAE J2735 messages. The APIs are synchronous in nature and runs in the same context of caller of the API function.

The message library provides APIs to encode and decode set of J2735 messages. The encode API function takes message structure as input and returns an encoded message buffer. The decode API function takes encoded buffer as input and outputs message structure filled with decoded values.

When calling message encode API, the input values are real world values. Internally the library converts it to fit in the message field as per J2735. For example, input degree value is converted to units of 1/10th of degree, speed value is converted to 0.1 meters per second etc. Similarly the message decode API will return real world value.

The V2X message library function prototypes, data structures, enumeration and defines are specified following header files,

- [v2x\\_msg\\_<msgtype>.h](#)

Where *msgtype* corresponds to the type of message for which the header file is defined (For example: [v2x\\_msg\\_bsm.h](#), [v2x\\_msg\\_spat.h](#), [v2x\\_msg\\_map.h](#), [v2x\\_msg\\_cam.h](#) etc.)

Applications include above mentioned header files and link with libv2x\_message.so shared library.

## 1.4 V2X Facility Library

The V2X Facility library is set of libraries that provides API interface to facility layer services. The facility layer services consist of following services,

- Basic Safety service
- Traveler information service
- SPaT and MAP service
- Cooperative Awareness service
- Decentralized Environmental Notification Service

### 1.4.1 V2X Basic Safety Service Library

The Basic Safety Service is a service provided by the Savari MobiWAVE system. It provides periodic generation of host vehicle Basic Safety Messages (BSM) as well of receiving the BSMs from remote vehicles. It also provides optional extended services like target classification and filtering of remote vehicle BSMs.

### 1.4.2 V2X Cooperative Service Library

The Cooperative Service is a service provided by the Savari MobiWAVE system. It provides periodic generation of host vehicle Cooperative Awareness Messages (CAM) as well of receiving the CAMs from remote vehicles. It also provides optional extended services like target classification and filtering of remote vehicle CAMs.

## 1.5 V2X Security Library

The V2X security library provides functions to securely sign the messages for transmission and verify received messages. The APIs are asynchronous in nature. The security library uses socket based IPC mechanism to communicate with Security process that implements signing and verification of messages.

The security signing API function takes message buffer as input. Once the security signing is complete, the signed message buffer is returned asynchronously to the caller by security process.

The security verification API function takes received message buffer as input. Once the security verification is complete, the verification status is returned asynchronously to the caller by security process.

## 1.6 V2X Transport and Network Services Library

The V2X transport and network services library provides API interface to 1609.3 WAVE networking services. It provides functions to send and receive WAVE short messages (WSM) and WAVE Service Advertisements (WSA). It includes API interface for provider service request, user service request, WSM service request.

### 1.6.1 Dynamic and static radio selection

This section is applicable only to ASD's 1609.3 protocol. It is not applicable to RSU.

Savari ASD's 1609.3 protocol implements one radio as a dynamic radio and another as a static radio. By default the second radio is treated as a static radio and first radio is treated as a dynamic radio. The applications that want to transmit and receive (using the V2V) on a fixed radio can register on the static radio. This is enough for the two way communication between the nodes.

The dynamic radio is used to listen for the WSA over 178. This allows the 1609.3 protocol to provide the information of the RSUs to the ASDs via the available services. The application then chooses the service of interest and provides the chosen service back to the 1609.3 service to provide the channel access. The application then receives the packets from the RSU that it is interested in.

The SDK allows this radio selection via one configuration variable called "secondradio". The secondradio when set to 1, makes the application run on the static radio. The secondradio when set to 0, makes the application run on the dynamic radio to match with the services available from the received WSA.

The secondradio is part of `v2x_wme_reg_req_t` data structure.

## 1.7 V2X Sensor Library

The V2X sensor library provides functions to get sensor information from sensor related services like GNSS and CAN services.

The V2X sensor library function prototypes, data structures, enumeration and defines are specified in following header files.

- [v2x\\_sensor\\_gnss.h](#)
- [v2x\\_sensor\\_can.h](#)



## Chapter 2

# Module Index

### 2.1 Modules

Here is a list of all modules:

The V2X Common API . . . . .	13
The V2X Message API (US) . . . . .	16
The V2X Message API (EU) . . . . .	40
The V2X Network and Transport module API (EU) . . . . .	49
The V2X Network and Transport module API (US) . . . . .	52
The V2X Security module API . . . . .	59
The V2X Sensors API . . . . .	64



## Chapter 3

# Data Structure Index

### 3.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">_J2735_2016_EnableLaneList_t</a>	71
<a href="#">_J2735_2016_ManeuverAssistList_t</a>	71
<a href="#">_J2735_2016_MovementEventList_t</a>	72
<a href="#">_J2735_2016_MovementState_t</a>	72
<a href="#">_J2735_2016_PathPrediction</a>	73
<a href="#">_J2735_2016_PHObject</a>	74
<a href="#">_J2735_2016_ProbeSnapshot</a>	74
<a href="#">_J2735_2016_PVD</a>	75
<a href="#">_J2735_2016_TimeChange_t</a>	77
<a href="#">_J2735_2016_VehicleMeasurements</a>	78
<a href="#">_J2735_2016_VehicleStatus</a>	78
<a href="#">_J2735_2016_VehSafetyExtensions</a>	79
<a href="#">_J2735_2016_Wiperset</a>	80
<a href="#">ae_ui_rx_callbacks</a>	80
<a href="#">aerolink_sign_secprofile</a>	82
<a href="#">aerolink_verify_secprofile</a>	82
<a href="#">connects_to</a>	83
<a href="#">datetime</a>	83
<a href="#">encbrakestatus</a>	84
<a href="#">encpadbytes</a>	84
<a href="#">encpointsetstype04</a>	84
<a href="#">encprndlspeed</a>	85
<a href="#">enctimestamp</a>	85
<a href="#">encvehiclesize</a>	85
<a href="#">etsi_cam_dgc</a>	86
<a href="#">etsi_cam_ec</a>	86
<a href="#">etsi_cam_ptc</a>	86
<a href="#">etsi_cam_rc</a>	86
<a href="#">etsi_cam_rwc_basic</a>	87
<a href="#">etsi_cam_scc</a>	87
<a href="#">etsi_cam_stc</a>	87
<a href="#">etsi_connects</a>	87
<a href="#">etsi_node</a>	88
<a href="#">etsi_node_attr</a>	88
<a href="#">etsi_pos3d</a>	88
<a href="#">ETSIItineraryPath_t</a>	89
<a href="#">ETSIRefDENMs_t</a>	89
<a href="#">ETSIRestrictedTypes_t</a>	89

intersec_access_pt	89
intersection_data	90
intersection_geodescr	91
J2735_2016_Computed_Lanes	92
j2735_2016_fullposvec	92
J2735_2016_GeometricProjection	93
J2735_2016_LatLong	93
J2735_2016_NodeSet_XY	94
J2735_2016_NodeXY	94
J2735_2016_OffsetSystem	95
J2735_2016_RegionOffsets	96
J2735_2016_RequestorType_t	96
J2735_2016_SignalRequest	96
J2735_2016_SignalRequest_t	97
J2735_2016_SignalRequestorInfo_t	98
J2735_2016_SignalStatusList	98
J2735_2016_SignalStatusPackageList	99
j2735_2016_srm	100
j2735_2016_ssm	101
J2735_2016_VehicleClassification_t	102
J2735_2016_VehicleIdent_t	102
J2735CircularRegion	102
J2735Position3D	103
J2735RegionPointSet	104
J2735RoadSignID	105
J2735ShapePointSet	105
J2735TIM	106
J2735TIMGeoPath	109
J2735ValidRegion	110
J2735VehicleID_t	111
lane_validity_params	112
lane_validity_time	112
laneattr	112
lanedataattr_list	113
laneset	113
libcan_filters	114
libwme_gpsinfo	115
libwme_radio_settings::libwme_radio	115
libwme_radio_settings	115
nodeattrs	115
odelist	116
nodepoint_list	117
path_history	117
pathhistoryframe	118
pf_gnss_data	119
position_confidence	121
positional_accuracy_data	121
positional_accuracy_data_2016	122
road_segment	122
RoadSegmentRefId	122
savari1609Wra	123
savari1609WsaService	124
savari_antennaoffset	124
savari_rtc_payload	125
security_cmd_profile_msg	125
spedelimits	125
srm_list	126
srm_requestor	126

testdataframe	127
timestamp	128
utcTimeFrame	129
v2x_cancel	130
v2x_emergency_details	130
v2x_etsi_cam	131
v2x_etsi_cam_bc	132
v2x_etsi_cam_hf	132
v2x_etsi_cam_hf_rsu	133
v2x_etsi_cam_hf_vehicle	133
v2x_etsi_cam_lf	136
v2x_etsi_denm	137
v2x_etsi_denm_ac	138
v2x_etsi_denm_lc	138
v2x_etsi_denm_mc	139
v2x_etsi_denm_rwc	141
v2x_etsi_denm_sc	142
v2x_etsi_lane	143
v2x_etsi_lane_attr	144
v2x_etsi_map	145
v2x_etsi_map_isec	145
v2x_etsi_map_msg	146
v2x_etsi_protected_zone	147
v2x_etsi_spat	147
v2x_etsi_spat_isec	148
v2x_etsi_spat_mm_evt	149
v2x_etsi_spat_mm_state	150
v2x_etsi_spat_msg	150
v2x_msg_bsm	151
v2x_msg_eva	158
v2x_msg_map	159
v2x_msg_psm	160
v2x_msg_rtc	161
v2x_msg_spat	162
v2x_msg_tim	162
v2x_wme_reg_req	164
v2x_wsmp_callbacks	167
v2x_wsmp_rcv_indication	169
v2x_wsmp_send_req	171
vendor_frame	173



## Chapter 4

# File Index

### 4.1 File List

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

<a href="#">v2x_canctl.h</a>	175
<a href="#">v2x_debug.h</a>	
V2X debug levels and related functions	176
<a href="#">v2x_error.h</a>	
V2X Status values, error codes and related functions	176
<a href="#">v2x_msg_bsm.h</a>	
BSM message API definitions and functions	177
<a href="#">v2x_msg_cam.h</a>	
CAM data APIs and data structures	180
<a href="#">v2x_msg_common.h</a>	183
<a href="#">v2x_msg_denm.h</a>	
DEN Message APIs and data structures	198
<a href="#">v2x_msg_etsi_map.h</a>	
MAP message APIs and data structures	200
<a href="#">v2x_msg_etsi_spat.h</a>	
SPAT Message APIs and data structures	203
<a href="#">v2x_msg_eva.h</a>	
EVA message API definitions and functions	206
<a href="#">v2x_msg_frame.h</a>	207
<a href="#">v2x_msg_map.h</a>	
MAP message API definitions and functions	207
<a href="#">v2x_msg_psm.h</a>	
PSM message API definitions and functions	209
<a href="#">v2x_msg_pvd.h</a>	
PVD message API definitions and functions	210
<a href="#">v2x_msg_rtcn.h</a>	
RTCM message API definitions and functions	213
<a href="#">v2x_msg_spat.h</a>	
SPAT message API definitions and functions	214
<a href="#">v2x_msg_srm.h</a>	
SRM message API definitions and functions	219
<a href="#">v2x_msg_ssm.h</a>	
SSM message API definitions and functions	221
<a href="#">v2x_msg_tim.h</a>	
TIM message API definitions and functions	222
<a href="#">v2x_net_btp.h</a>	
BTP API definitions and functions	235

<a href="#">v2x_net_wme.h</a>	
WME API definitions and functions . . . . .	236
<b>v2x_pf.h</b> . . . . .	??
<a href="#">v2x_security.h</a>	
1609.2 API definitions and functions . . . . .	249
<a href="#">v2x_security_base.h</a>	
1609.2 API definitions and functions . . . . .	249
<a href="#">v2x_sensor_can.h</a>	
CAN sensor API definitions and function . . . . .	252
<a href="#">v2x_sensor_gnss.h</a>	
Navigation sensor API definitions and functions . . . . .	252



## Chapter 5

# Module Documentation

### 5.1 The V2X Common API

#### Enumerations

- enum `v2x_status_t` {  
    `V2X_SUCCESS` = 0, `V2X_EINVAL` = -1,  
    `V2X_ENOMEM` = -2, `V2X_EFAULT` = -3,  
    `V2X_EENCODE` = -4, `V2X_EDECODE` = -5,  
    `V2X_EACCES` = -6, `V2X_EIO` = -7,  
    `V2X_PEER_CLOSE` = -8, `V2X_ECONN_REFUSED` = -9,  
    `V2X_ELENGTH` = -10, `V2X_ECHANNEL` = -11 }

#### Functions

- `v2x_status_t v2x_debug_level_set` (int dbg\_level)
- void `v2x_log` (int log\_level, const char \*fmt,...)
- `v2x_status_t v2x_strerror` (`v2x_status_t` errnum, char \*buf, size\_t buflen)

#### 5.1.1 Detailed Description

This module defines all the API functions of V2X common library module

The V2X common module defines data structures and API functions that are common to rest of V2X library modules. This include status values, error definitions and related functions.

#### 5.1.2 Enumeration Type Documentation

##### 5.1.2.1 enum `v2x_status_t`

Defines status value and error codes.

#### Enumerator

- `V2X_SUCCESS`** operation successful
- `V2X_EINVAL`** Invalid value
- `V2X_ENOMEM`** Out of memory
- `V2X_EFAULT`** Bad pointer
- `V2X_EENCODE`** Encode error

**V2X\_EDECODE** Decode error

**V2X\_EACCES** Error in accessing a device

**V2X\_EIO** Device read/write error

**V2X\_PEER\_CLOSE** Connection to peer socket closed

**V2X\_ECONN\_REFUSED** Connection refused .. server unavailable

**V2X\_ELENGTH** invalid length of the input data

**V2X\_ECHANNEL** invalid value of channel

### 5.1.3 Function Documentation

#### 5.1.3.1 v2x\_status\_t v2x\_debug\_level\_set ( int dbg\_level )

Set debug level

Parameters

in	<i>dbg_level</i>	Debug level to set
----	------------------	--------------------

Returns

Returns V2X\_SUCCESS on success and error code on failure.

#### 5.1.3.2 void v2x\_log ( int log\_level, const char \* fmt, ... )

Savari v2x print API

Parameters

in	<i>log_level</i>	The log level
in	<i>fmt</i>	Printf style format specifier

Returns

None.

#### 5.1.3.3 v2x\_status\_t v2x\_strerror ( v2x\_status\_t errnum, char \* buf, size\_t buflen )

This function provides error string that describes the error code passed in errnum

Parameters

in	<i>errnum</i>	Error code for which error string is needed
out	<i>buf</i>	Buffer to fill with error string
in	<i>buflen</i>	Length of the buffer

Description

This functions returns error string corresponding to the sepcified error code.

Returns

V2X\_SUCCESS on success and error code on failure.

## Return values

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.

## 5.2 The V2X Message API (US)

### Files

- file [v2x\\_msg\\_common.h](#)
- file [v2x\\_msg\\_frame.h](#)

### Data Structures

- struct [v2x\\_msg\\_bsm](#)
- struct [J2735Position3D](#)
- struct [nodeattrs](#)
- struct [odelist](#)
- struct [nodepoint\\_list](#)
- struct [\\_J2735\\_2016\\_PathPrediction](#)
- struct [path\\_history](#)
- struct [\\_J2735\\_2016\\_Wiperset](#)
- struct [laneattr](#)
- struct [connects\\_to](#)
- struct [laneset](#)
- struct [v2x\\_msg\\_eva](#)
- struct [v2x\\_msg\\_map](#)
- struct [v2x\\_msg\\_rtc](#)
- struct [v2x\\_msg\\_spat](#)
- struct [j2735\\_2016\\_srm](#)
- struct [j2735\\_2016\\_ssm](#)
- struct [v2x\\_msg\\_tim](#)

### Typedefs

- typedef struct [v2x\\_msg\\_bsm](#) [v2x\\_msg\\_bsm\\_t](#)
- typedef enum [j2735\\_dsrm\\_msgids](#) [J2735\\_2016\\_DSRC\\_MsgID\\_t](#)
- typedef struct [J2735Position3D](#) [J2735Position3D\\_t](#)
- typedef struct [nodeattrs](#) [J2735\\_2016\\_NodeAttr\\_t](#)
- typedef struct [odelist](#) [J2735\\_2016\\_NodeList\\_t](#)
- typedef struct [nodepoint\\_list](#) [J2735\\_2016\\_NodePoint\\_t](#)
- typedef enum [lanedirection](#) [J2735\\_2016\\_LaneDir\\_t](#)
- *MAP & LANE specific objects / definitions.*
- typedef struct [\\_J2735\\_2016\\_PathPrediction](#) [J2735\\_2016\\_PathPrediction\\_t](#)
- typedef struct [path\\_history](#) [J2735\\_2016\\_PathHistory\\_t](#)
- typedef struct [\\_J2735\\_2016\\_Wiperset](#) [J2735\\_2016\\_WiperSet\\_t](#)
- typedef enum [lanetypevehicle](#) [J2735\\_2016\\_LaneTypeAttrVeh\\_t](#)
- typedef enum [lanetypecrosswalk](#) [J2735\\_2016\\_LaneTypeAttrCrossWk\\_t](#)
- typedef struct [laneattr](#) [J2735\\_2016\\_LaneAttribute\\_t](#)
- typedef struct [connects\\_to](#) [J2735\\_2016\\_Connectsto\\_Lanes\\_t](#)
- typedef struct [laneset](#) [J2735\\_2016\\_LaneList\\_t](#)
- typedef enum [J2735\\_2016\\_HeadingSlice](#) [J2735\\_2016\\_HeadingSlice\\_t](#)
- typedef enum [J2735Extent](#) [J2735Extent\\_t](#)
- typedef struct [v2x\\_msg\\_eva](#) [v2x\\_msg\\_eva\\_t](#)
- typedef struct [v2x\\_msg\\_map](#) [v2x\\_msg\\_map\\_t](#)
- typedef struct [v2x\\_msg\\_rtc](#) [v2x\\_msg\\_rtc\\_t](#)

- typedef struct `v2x_msg_spat` `v2x_msg_spat_t`
- typedef struct `j2735_2016_srm` `v2x_msg_srm_t`
- typedef struct `j2735_2016_ssm` `v2x_msg_ssm_t`
- typedef struct `v2x_msg_tim` `v2x_msg_tim_t`

## Enumerations

- enum `j2735_dssrc_msgids` {  
`J2735_2016_BSM_MSG_D` = 0x02, `J2735_2016_TIM_MSG_D` = 0x10,  
`J2735_2016_RTCM_CORRECTIONS_MSG_D` = 0x0C, `J2735_2016_UPERFRAME_D` = 0x11,  
`J2735_2016_MAP_MSG_P` = 0x12, `J2735_2016_SPaT_MSG_P` = 0x13,  
`J2735_2016_TIM_MSG_P` = 0x1F, `J2735_2016_BSM_MSG_U` = 0x14,  
`J2735_2016_CSR_MSG_U` = 0x15, `J2735_2016_EVA_MSG_U` = 0x16,  
`J2735_2016_IC_MSG_U` = 0x17, `J2735_2016_NMEA_CORRECTION_MSG_U` = 0x18,  
`J2735_2016_PDM_MSG_U` = 0x19, `J2735_2016_PVD_MSG_U` = 0x1A,  
`J2735_2016_RSA_MSG_U` = 0x1B, `J2735_2016_RTCM_CORRECTIONS_MSG_U` = 0x1C,  
`J2735_2016_SRM_MSG_U` = 0x1D, `J2735_2016_SSM_MSG_U` = 0x1E,  
`J2735_2016_TIM_MSG_U` = 0x1F, `J2735_2016_PSM_MSG_U` = 0x20 }
  - enum `J2735_2016_NodeType_t` {  
`J2735_2016_NODE_TYPE_NOT_PRESENT` = 0, `J2735_2016_NODE_TYPE_NODE_OBJS`,  
`J2735_2016_NODE_TYPE_COMP` }
  - enum `J2735_2016_NodeAttrType_t` {  
`J2735_2016_NODEATTR_NODEATTR` = 0x01, `J2735_2016_NODEATTR_DISABLED_SEGMENTS` =  
0x02,  
`J2735_2016_NODEATTR_ENABLED_SEGMENTS` = 0x04, `J2735_2016_NODEATTR_LANEDATA` = 0x08,  
`J2735_2016_NODEATTR_UNAVAIL` = 0x00 }
  - enum `lanedirection` {  
`J2735_2016_LANE_DIRECTION_UNAVAIL` = 0x00, `J2735_2016_LANE_DIRECTION_INGRESS` = 0x01,  
`J2735_2016_LANE_DIRECTION_EGRESS` = 0x02 }
- MAP & LANE specific objects / definitions.*
- enum `J2735_2016_DirofUse_t` {  
`J2735_2016_DIROFUSE_FORWARD` = 0, `J2735_2016_DIROFUSE_REVERSE`,  
`J2735_2016_DIROFUSE_BOTH` }
  - enum `J2735_2016_VehicleEventFlags_t` {  
`J2735_2016_EVT_HAZARD_LIGHTS` = 0x0001, `J2735_2016_EVT_STOPLINE_VIOLATION` = 0x0002,  
`J2735_2016_EVT_ABS_ACTIVATED` = 0x0004, `J2735_2016_EVT_TRACTION_CONTROL` = 0x0008,  
`J2735_2016_EVT_STABILITY_CONTROL` = 0x0010, `J2735_2016_EVT_HAZARDOUS_MATERIAL` =  
0x0020,  
`J2735_2016_EVT_RESERVED1` = 0x0040, `J2735_2016_EVT_HARDBRAKING` = 0x0080,  
`J2735_2016_EVT_LIGHTS_CHANGED` = 0x0100, `J2735_2016_EVT_WIPERS_CHANGED` = 0x0200,  
`J2735_2016_EVT_FLATTIRE` = 0x0400, `J2735_2016_EVT_DISABLED_VEHICLE` = 0x0800,  
`J2735_2016_EVT_AIRBAG_DEPLOYMENT` = 0x1000 }
  - enum `J2735_2016_ExteriorLights_t` {  
`J2735_2016_EXT_LOWBEAM_ON` = 0x0001, `J2735_2016_EXT_HIGHBEAM_ON` = 0x0002,  
`J2735_2016_EXT_LEFTTURN_ON` = 0x0004, `J2735_2016_EXT_RIGHTTURN_ON` = 0x0008,  
`J2735_2016_EXT_HAZARDSIGNAL_ON` = 0x0010, `J2735_2016_EXT_AUTOMATIC_LIGHT_CONTROL_`  
`ON` = 0x0020,  
`J2735_2016_EXT_DAYTIMERUNLIGHTS_ON` = 0x0040, `J2735_2016_EXT_FOGLIGHTS_ON` = 0x0080,  
`J2735_2016_EXT_PARKINGLIGHTS_ON` = 0x0100 }
  - enum `J2735_2016_WiperStatus_t` {  
`J2735_2016_WIPER_STATUS_UNAVAIL` = 0, `J2735_2016_WIPER_STATUS_OFF`,  
`J2735_2016_WIPER_INTERMITTENT`, `J2735_2016_WIPER_LOW`,  
`J2735_2016_WIPER_HIGH`, `J2735_2016_WIPER_WASHER_IN_USE`,  
`J2735_2016_WIPER_AUTOMATIC_PRESENT` }

- enum `lanetypevehicle` {  
`J2735_2016_VEH_IS_VEHICLE_REVOCABLE_LANE` = 0x01, `J2735_2016_VEH_IS_VEHICLE_FLYOVER_LANE` = 0x02,  
`J2735_2016_VEH_HOV_USE_ONLY` = 0x04, `J2735_2016_VEH_RESTRICTED_TO_BUS_USE` = 0x08,  
`J2735_2016_VEH_RESTRICTED_TO_TAXI_USE` = 0x10, `J2735_2016_VEH_RESTRICTED_FROM_PUBLIC_USE` = 0x20,  
`J2735_2016_VEH_HAS_IR_BEACON_COVERAGE` = 0x40, `J2735_2016_VEH_PERMISSION_ON_REQUEST` = 0x80 }
- enum `lanetypecrosswalk` {  
`J2735_2016_CRW_CRW_REVOCABLE_LANE` = 0x0001, `J2735_2016_CRW_BICYCLE_USE_ONLY` = 0x0002,  
`J2735_2016_CRW_IS_FLYOVER_CRW_LANE` = 0x0004, `J2735_2016_CRW_FIXED_CYCLE_TIME` = 0x0008,  
`J2735_2016_CRW_BIDIRECTIONAL_CYCLE_TIME` = 0x0010, `J2735_2016_CRW_HAS_PUSH_TO_WALK_BUTTON` = 0x0020,  
`J2735_2016_CRW_AUDIO_SUPPORT` = 0x0040, `J2735_2016_CRW_RFSIGNAL_REQ_PRESENT` = 0x0080,  
`J2735_2016_CRW_UNSIGNALIZED_SEGMENTS_PRESENT` = 0x0100 }
- enum `J2735_2016_AllowedManeuvers_t` {  
`J2735_2016_MAN_STRAIGHT_ALLOWED` = 0x0001, `J2735_2016_MAN_LEFT_ALLOWED` = 0x0002,  
`J2735_2016_MAN_RIGHT_ALLOWED` = 0x0004, `J2735_2016_MAN_UTURN_ALLOWED` = 0x0008,  
`J2735_2016_MAN_LEFT_TURN_ON_RED_ALLOWED` = 0x0010, `J2735_2016_MAN_RIGHT_TURN_ON_RED_ALLOWED` = 0x0020,  
`J2735_2016_MAN_LANE_CHANGE_ALLOWED` = 0x0040, `J2735_2016_MAN_NO_STOPPING_ALLOWED` = 0x0080,  
`J2735_2016_MAN_YIELD_ALLWAYS_REQUIRED` = 0x0100, `J2735_2016_MAN_GOWITH_HALT` = 0x0200,  
`J2735_2016_MAN_CAUTION` = 0x0400, `J2735_2016_MAN_RESERVED1` = 0x0800 }
- enum `J2735_2016_HeadingSlice` {  
`J2735_2016_HS_FROM00_0_TO_22_5_DEGREES` = 0x0001, `J2735_2016_HS_FROM22_5_TO_45_0_DEGREES` = 0x0002,  
`J2735_2016_HS_FROM45_0_TO_67_5_DEGREES` = 0x0004, `J2735_2016_HS_FROM67_5_TO_90_0_DEGREES` = 0x0008,  
`J2735_2016_HS_FROM90_0_TO_112_5_DEGREES` = 0x0010, `J2735_2016_HS_FROM112_5_TO_135_0_DEGREES` = 0x0020,  
`J2735_2016_HS_FROM135_0_TO_157_5_DEGREES` = 0x0040, `J2735_2016_HS_FROM157_5_TO_180_0_DEGREES` = 0x0080,  
`J2735_2016_HS_FROM180_0_TO_202_5_DEGREES` = 0x0100, `J2735_2016_HS_FROM202_5_TO_225_0_DEGREES` = 0x0200,  
`J2735_2016_HS_FROM225_0_TO_247_5_DEGREES` = 0x0400, `J2735_2016_HS_FROM247_5_TO_270_0_DEGREES` = 0x0800,  
`J2735_2016_HS_FROM270_0_TO_292_5_DEGREES` = 0x1000, `J2735_2016_HS_FROM292_5_TO_315_0_DEGREES` = 0x2000,  
`J2735_2016_HS_FROM315_0_TO_337_5_DEGREES` = 0x4000, `J2735_2016_HS_FROM337_5_TO_360_0_DEGREES` = 0x8000 }
- enum `J2735Extent` {  
`J2735_EXTENT_USEINSTANTLYONLY` = 0, `J2735_EXTENT_USEFOR3METERS` = 1,  
`J2735_EXTENT_USEFOR10METERS` = 2, `J2735_EXTENT_USEFOR50METERS` = 3,  
`J2735_EXTENT_USEFOR100METERS` = 4, `J2735_EXTENT_USEFOR500METERS` = 5,  
`J2735_EXTENT_USEFOR1000METERS` = 6, `J2735_EXTENT_USEFOR5000METERS` = 7,  
`J2735_EXTENT_USEFOR10000METERS` = 8, `J2735_EXTENT_USEFOR50000METERS` = 9,  
`J2735_EXTENT_USEFOR100000METERS` = 10, `J2735_EXTENT_USEFOR500000METERS` = 11,  
`J2735_EXTENT_USEFOR1000000METERS` = 12, `J2735_EXTENT_USEFOR5000000METERS` = 13,  
`J2735_EXTENT_USEFOR10000000METERS` = 14, `J2735_EXTENT_FOREVER` = 15 }

## Functions

- void `v2x_msg_bsm_init` (`v2x_msg_bsm_t` \*bsm)

- `v2x_status_t v2x_msg_bsm_encode (v2x_msg_bsm_t *bsm, BSM_TYPE type, unsigned char *msgbuf, int msgbuf_size, int *enclen)`
- `v2x_status_t v2x_msg_bsm_decode (v2x_msg_bsm_t *bsm, unsigned char *msgbuf, int msgbuf_len, BSM_TYPE *type, uint32_t *oob, int debug)`
- `v2x_status_t v2x_msg_eva_decode (v2x_msg_eva_t *eva, uint8_t *encbytes, int enclen)`
- `v2x_status_t v2x_msg_eva_encode (v2x_msg_eva_t *eva, uint8_t *encbytes, int encbytes_size, int *enc_len)`
- `void v2x_msg_eva_reset (v2x_msg_eva_t *eva)`
- `void v2x_msg_eva_print (FILE *fp, v2x_msg_eva_t *eva)`
- `void v2x_msg_map_init (v2x_msg_map_t *map_msg)`
- `v2x_status_t v2x_msg_map_encode (v2x_msg_map_t *map, uint8_t *msgbuf, int msgbuf_size, int *enclen)`
- `v2x_status_t v2x_msg_map_decode (v2x_msg_map_t *map, uint8_t *buf, int bufsize, int debug)`
- `void v2x_map_free (v2x_msg_map_t *map)`
- `v2x_status_t v2x_msg_psm_encode (v2x_msg_psm_t *psm, unsigned char *encoded_buffer, int bufsize, int *encoded_len)`
- `v2x_status_t v2x_msg_psm_decode (v2x_msg_psm_t *psm, uint8_t *enc_buf, int buflen, uint32_t *oob, int debug)`
- `void v2x_msg_psm_reset (v2x_msg_psm_t *psm)`
- `v2x_status_t v2x_pvd_encode (v2x_msg_pvd_t *pvd, uint8_t *encbuf, int size, int *encbuf_len)`
- `v2x_status_t v2x_pvd_decode (v2x_msg_pvd_t *pvd, int type, uint8_t *encbuf, int buflen, uint32_t *oob, int *dec_len)`
- `void v2x_pvd_reset (v2x_msg_pvd_t *pvd)`
- `void v2x_pvd_snapshot_reset (J2735_2016_ProbeSnapshot_t *snapshot)`
- `v2x_status_t v2x_msg_rtcml_encode (v2x_msg_rtcml_t *rtcml, unsigned char *msgbuf, int msgbuf_size, int *enclen)`
- `v2x_status_t v2x_msg_rtcml_decode (v2x_msg_rtcml_t *rtcml, uint8_t *msgbuf, int msgbuf_len, int *declen)`
- `void v2x_msg_rtcml_print (FILE *fp, v2x_msg_rtcml_t *rtcml)`
- `v2x_status_t v2x_msg_spat_encode (v2x_msg_spat_t *spatmsg, uint8_t *buf, int len, int *enc_len)`
- `int v2x_msg_spat_decode (v2x_msg_spat_t *spatmsg, uint8_t *buf, int len, int *dec_len, int debug)`
- `void v2x_spat_free (v2x_msg_spat_t *spat)`
- `v2x_status_t v2x_srm_encode (v2x_msg_srm_t *srm, uint8_t *encbuf, int *encbuf_len, int size)`
- `v2x_status_t v2x_srm_decode (v2x_msg_srm_t *srm, uint8_t *encbuf, int *encbuf_len, int size)`
- `void v2x_srm_reset (v2x_msg_srm_t *srm)`
- `v2x_status_t v2x_ssm_encode (v2x_msg_ssm_t *ssm, uint8_t *encbuf, int *encbuf_len, int size)`
- `v2x_status_t v2x_ssm_decode (v2x_msg_ssm_t *ssm, int type, uint8_t *encbuf, int buflen, uint32_t *oob, int *dec_len)`
- `void v2x_ssm_reset (v2x_msg_ssm_t *ssm)`
- `v2x_status_t v2x_msg_tim_encode (v2x_msg_tim_t *tim, int type, uint8_t *msgbuf, int msgbuf_size, int *enclen)`
- `v2x_status_t v2x_msg_tim_decode (v2x_msg_tim_t *tim, int type, uint8_t *msgbuf, int msgbuf_len, uint32_t *oob, int *declen, int debug)`
- `void v2x_msg_tim_free (v2x_msg_tim_t *tim)`

### 5.2.1 Detailed Description

The V2X message module defines data structures and API functions for applications to work with J2735 & ETSI defined messages. It provides functions for encoding and decoding various J2735 and ETSI defined V2X messages including BSM, SPaT, MAP, TIM, CAM & DENM etc.

Description:

This header contains the the necessary API and data structures to encode / decode PSM message. link with -lj2735-2016, -lsae\_2016 -losstd and -lm

The V2X message module defines the data structures and API functions for applications to work with J2735 & ETSI defined messages. It provides functions for encoding and decoding various J2735 and ETSI defined V2X messages including BSM, SPaT, MAP, TIM, CAM & DENM etc.

## 5.2.2 Typedef Documentation

### 5.2.2.1 typedef struct connects\_to J2735\_2016\_Connectsto\_Lanes\_t

connects to list describing the lane connection with lane id and the signal group ID

### 5.2.2.2 typedef enum j2735\_dsrmsgids J2735\_2016\_DSRC\_MsgID\_t

List of DSRC message IDS in the J2735

### 5.2.2.3 typedef enum J2735\_2016\_HeadingSlice J2735\_2016\_HeadingSlice\_t

Heading slice describing 16 slices. Please refer to SAEJ2735 for further information

### 5.2.2.4 typedef struct laneattr J2735\_2016\_LaneAttribute\_t

Lane attributes describing the lane such as the direction, shared lanes, vehicle or cross walk attributes

### 5.2.2.5 typedef enum lanedirection J2735\_2016\_LaneDir\_t

MAP & LANE specific objects / definitions.

lane direction

### 5.2.2.6 typedef struct laneset J2735\_2016\_LaneList\_t

lane describing the information pertaining to the lane such as its phase, allowed maneuvers, signal group ID, lane attributes and a set of nodes

### 5.2.2.7 typedef enum lanetypecrosswalk J2735\_2016\_LaneTypeAttrCrossWk\_t

Cross walk attributes

### 5.2.2.8 typedef enum lanetypevehicle J2735\_2016\_LaneTypeAttrVeh\_t

vehicle attributes

### 5.2.2.9 typedef struct nodeattrs J2735\_2016\_NodeAttr\_t

nodeattributes of a particular node point

### 5.2.2.10 typedef struct nodelist J2735\_2016\_NodeList\_t

list of nodes

### 5.2.2.11 typedef struct nodepoint\_list J2735\_2016\_NodePoint\_t

•

node point describing a node and its attributes



**5.2.2.12 typedef struct path\_history J2735\_2016\_PathHistory\_t**

path history information represented in the positional offsets

**5.2.2.13 typedef struct \_J2735\_2016\_PathPrediction J2735\_2016\_PathPrediction\_t**

path prediction information

**5.2.2.14 typedef struct \_J2735\_2016\_Wiperset J2735\_2016\_WiperSet\_t**

wiper set - describing the front, rear wipers and the rates

**5.2.2.15 typedef enum J2735Extent J2735Extent\_t**

Extent object. Please refer to DE\_Extent of SAEj2735 for further details.

**5.2.2.16 typedef struct J2735Position3D J2735Position3D\_t**

J2735Position3D\_t - position 3d object

**5.2.2.17 typedef struct v2x\_msg\_bsm v2x\_msg\_bsm\_t**

Basic Safety Message structure is used for communicating BSM data with an application program. For more information about the BSM message elements, see SAE J2735 document. The BSM part2 starts from the events. The events are dictated based on vehicle CAN messages.

**5.2.2.18 typedef struct v2x\_msg\_eva v2x\_msg\_eva\_t**

EVA message

**5.2.2.19 typedef struct v2x\_msg\_map v2x\_msg\_map\_t**

MAP message structure

**5.2.2.20 typedef struct v2x\_msg\_rtcn v2x\_msg\_rtcn\_t**

RTCM corrections structure

**5.2.2.21 typedef struct v2x\_msg\_spat v2x\_msg\_spat\_t**

SPAT message structure

**5.2.2.22 typedef struct j2735\_2016\_srm v2x\_msg\_srm\_t**

SRM message

**5.2.2.23 typedef struct j2735\_2016\_ssm v2x\_msg\_ssm\_t**

SSM message

### 5.2.2.24 typedef struct v2x\_msg\_tim v2x\_msg\_tim\_t

The TIM Message structure

The format of a tim message looks like this

```
+-----+
| TIM_MSG_ID | TIM_UNIQUE_ID | TIM_DATAFRAME_1 | TIM_DATAFRAME_2 | - - - | TIM_CRC |
+-----+
```

Each dataframe have its header, region and contents like the below

```
+-----+
| TIM_DF_HEAD | TIM_DF_REGIONS | TIM_DF_CONTENTS |
+-----+
```

Each TIM\_DF\_HEAD consists of one of two types, further info id or roadsign id. FURTHER\_INFO\_ID is two byte number or ROADSIGN\_ID is a dataframe as shown below

```
+-----+
| LAT | LONG | ELEV | HEADING_SLICE | MUTCDCODE |
+-----+
```

Each TIM\_DF\_REGION consists of a anchor point and heading slice and a extent field, and lastly a region field.

```
+-----+
| LAT | LONG | ELEV | HEADING_SLICE | EXTENT | REGION |
+-----+
```

Each REGION again will be one of three circle, polygon, and shapepoints. as of now this library supports only shapepoints we specify only shapepoints now.

Each shapepoint consists of a set of offsets with heading slice and a anchor point. The offsets added to the anchor point to get a shapepoint region

```
+-----+
| ANCHOR | HEADING_SLICE | LANEWIDTH | XOFF_1|YOFF_1|ZOFF_1| X_OFF2|Y_OFF2|Z_OFF2| --- | XOFF_N|YOFF_N|ZOFF_N|
+-----+
```

Each and every content filed (For Ex. advisory\_codes, workzone\_codes, genericsign\_code and speedlimit\_codes) should follow a correct assignment of values.

```
Example:
for (i = 0; i < num_advisories; i++) {
    advisory_codes[i] = *user_configured_adv_code;
    user_configured_adv_code++;
}
```

where the user\_configured\_adv\_code is a uint32\_t pointer.

## 5.2.3 Enumeration Type Documentation

### 5.2.3.1 enum J2735\_2016\_AllowedManeuvers\_t

maneuver list describing the single movement in an intersection to cross an intersection

Enumerator

**J2735\_2016\_MAN\_STRAIGHT\_ALLOWED** J2735\_2016\_MAN\_STRAIGHT\_ALLOWED - straight maneuver allowed

**J2735\_2016\_MAN\_LEFT\_ALLOWED** J2735\_2016\_MAN\_LEFT\_ALLOWED - left maneuver allowed

**J2735\_2016\_MAN\_RIGHT\_ALLOWED** J2735\_2016\_MAN\_RIGHT\_ALLOWED - right maneuver allowed

**J2735\_2016\_MAN\_UTURN\_ALLOWED** J2735\_2016\_MAN\_UTURN\_ALLOWED - uturn allowed

**J2735\_2016\_MAN\_LEFT\_TURN\_ON\_RED\_ALLOWED** J2735\_2016\_MAN\_LEFT\_TURN\_ON\_RED\_ALLOWED - left turn on red allowed

**J2735\_2016\_MAN\_RIGHT\_TURN\_ON\_RED\_ALLOWED** J2735\_2016\_MAN\_RIGHT\_TURN\_ON\_RED\_ALLOWED - right turn on red allowed

**J2735\_2016\_MAN\_LANE\_CHANGE\_ALLOWED** J2735\_2016\_MAN\_LANE\_CHANGE\_ALLOWED - lane change allowed

**J2735\_2016\_MAN\_NO\_STOPPING\_ALLOWED** J2735\_2016\_MAN\_NO\_STOPPING\_ALLOWED - no stopping allowed

**J2735\_2016\_MAN\_YIELD\_ALLWAYS\_REQUIRED** J2735\_2016\_MAN\_YIELD\_ALLWAYS\_REQUIRED - all ways yield

**J2735\_2016\_MAN\_GOWITH\_HALT** J2735\_2016\_MAN\_GOWITH\_HALT - go with halt

**J2735\_2016\_MAN\_CAUTION** J2735\_2016\_MAN\_CAUTION - caution

#### 5.2.3.2 enum J2735\_2016\_DirofUse\_t

direction of use

Enumerator

**J2735\_2016\_DIROFUSE\_FORWARD** J2735\_2016\_DIROFUSE\_FORWARD - forward

**J2735\_2016\_DIROFUSE\_REVERSE** J2735\_2016\_DIROFUSE\_REVERSE - reverse

**J2735\_2016\_DIROFUSE\_BOTH** J2735\_2016\_DIROFUSE\_BOTH - both

#### 5.2.3.3 enum J2735\_2016\_ExteriorLights\_t

exterior lights information from the vehicle CAN

Enumerator

**J2735\_2016\_EXT\_LOWBEAM\_ON** J2735\_2016\_EXT\_LOWBEAM\_ON - low beam on

**J2735\_2016\_EXT\_HIGHBEAM\_ON** J2735\_2016\_EXT\_HIGHBEAM\_ON - high beam on

**J2735\_2016\_EXT\_LEFTTURN\_ON** J2735\_2016\_EXT\_LEFTTURN\_ON - leftturn on

**J2735\_2016\_EXT\_RIGHTTURN\_ON** J2735\_2016\_EXT\_RIGHTTURN\_ON - rightturn on

**J2735\_2016\_EXT\_HAZARDSIGNAL\_ON** J2735\_2016\_EXT\_HAZARDSIGNAL\_ON - hazard signal on

**J2735\_2016\_EXT\_AUTOMATIC\_LIGHT\_CONTROL\_ON** J2735\_2016\_EXT\_AUTOMATIC\_LIGHT\_CONTROL\_ON - automatic light control on

**J2735\_2016\_EXT\_DAYTIMERUNLIGHTS\_ON** J2735\_2016\_EXT\_DAYTIMERUNLIGHTS\_ON - daytime run lights on

**J2735\_2016\_EXT\_FOGLIGHTS\_ON** J2735\_2016\_EXT\_FOGLIGHTS\_ON - fog lights on

**J2735\_2016\_EXT\_PARKINGLIGHTS\_ON** J2735\_2016\_EXT\_PARKINGLIGHTS\_ON - parking lights on

#### 5.2.3.4 enum J2735\_2016\_HeadingSlice

Heading slice describing 16 slices. Please refer to SAEJ2735 for further information

Enumerator

**J2735\_2016\_HS\_FROM00\_0\_TO\_22\_5\_DEGREES** J2735\_2016\_HS\_FROM00\_0\_TO\_22\_5\_DEGREES - 0 to 22.5 degrees

**J2735\_2016\_HS\_FROM22\_5\_TO\_45\_0\_DEGREES** J2735\_2016\_HS\_FROM22\_5\_TO\_45\_0\_DEGREES - 22.5 to 45 degrees

**J2735\_2016\_HS\_FROM45\_0\_TO\_67\_5\_DEGREES** J2735\_2016\_HS\_FROM45\_0\_TO\_67\_5\_DEGREES - 45 to 67.5 degrees

**J2735\_2016\_HS\_FROM67\_5\_TO\_90\_0\_DEGREES** J2735\_2016\_HS\_FROM67\_5\_TO\_90\_0\_DEGREES - 67.5 to 90 degrees

**J2735\_2016\_HS\_FROM90\_0\_TO\_112\_5\_DEGREES** J2735\_2016\_HS\_FROM90\_0\_TO\_112\_5\_DEGREE-  
S - 90 to 112.5 degrees

**J2735\_2016\_HS\_FROM112\_5\_TO\_135\_0\_DEGREES** J2735\_2016\_HS\_FROM112\_5\_TO\_135\_0\_DEGR-  
EES - 112.5 to 135 degrees

**J2735\_2016\_HS\_FROM135\_0\_TO\_157\_5\_DEGREES** J2735\_2016\_HS\_FROM135\_0\_TO\_157\_5\_DEGR-  
EES - 135 to 157.5 degrees

**J2735\_2016\_HS\_FROM157\_5\_TO\_180\_0\_DEGREES** J2735\_2016\_HS\_FROM157\_5\_TO\_180\_0\_DEGR-  
EES - 157.5 to 180 degrees

**J2735\_2016\_HS\_FROM180\_0\_TO\_202\_5\_DEGREES** J2735\_2016\_HS\_FROM180\_0\_TO\_202\_5\_DEGR-  
EES - 180 to 202.5 degrees

**J2735\_2016\_HS\_FROM202\_5\_TO\_225\_0\_DEGREES** J2735\_2016\_HS\_FROM202\_5\_TO\_225\_0\_DEGR-  
EES - 202.5 to 225 degrees

**J2735\_2016\_HS\_FROM225\_0\_TO\_247\_5\_DEGREES** J2735\_2016\_HS\_FROM225\_0\_TO\_247\_5\_DEGR-  
EES - 225 to 247.5 degrees

**J2735\_2016\_HS\_FROM247\_5\_TO\_270\_0\_DEGREES** J2735\_2016\_HS\_FROM247\_5\_TO\_270\_0\_DEGR-  
EES - 247.5 to 270 degrees

**J2735\_2016\_HS\_FROM270\_0\_TO\_292\_5\_DEGREES** J2735\_2016\_HS\_FROM270\_0\_TO\_292\_5\_DEGR-  
EES - 270 to 292.5 degrees

**J2735\_2016\_HS\_FROM292\_5\_TO\_315\_0\_DEGREES** J2735\_2016\_HS\_FROM292\_5\_TO\_315\_0\_DEGR-  
EES - 292.5 to 315 degrees

**J2735\_2016\_HS\_FROM315\_0\_TO\_337\_5\_DEGREES** J2735\_2016\_HS\_FROM315\_0\_TO\_337\_5\_DEGR-  
EES - 315 to 337.5 degrees

**J2735\_2016\_HS\_FROM337\_5\_TO\_360\_0\_DEGREES** J2735\_2016\_HS\_FROM337\_5\_TO\_360\_0\_DEGR-  
EES - 337.5 to 360 degrees

#### 5.2.3.5 enum J2735\_2016\_NodeAttrType\_t

•

Node attribute type

#### Enumerator

**J2735\_2016\_NODEATTR\_NODEATTR** J2735\_2016\_NODEATTR\_NODEATTR - node attribute

**J2735\_2016\_NODEATTR\_DISABLED\_SEGMENTS** J2735\_2016\_NODEATTR\_DISABLED\_SEGMENTS - disabled segment

**J2735\_2016\_NODEATTR\_ENABLED\_SEGMENTS** J2735\_2016\_NODEATTR\_ENABLED\_SEGMENTS - enabled segment

**J2735\_2016\_NODEATTR\_LANEDATA** J2735\_2016\_NODEATTR\_LANEDATA - lanedata

**J2735\_2016\_NODEATTR\_UNAVAIL** J2735\_2016\_NODEATTR\_UNAVAIL - unavailable

## 5.2.3.6 enum J2735\_2016\_NodeType\_t

NodeTypes to describe the area of a lane or segment of a road

## Enumerator

**J2735\_2016\_NODE\_TYPE\_NOT\_PRESENT** not present  
**J2735\_2016\_NODE\_TYPE\_NODE\_OBJS** node objects meaning way point sets  
**J2735\_2016\_NODE\_TYPE\_COMP** computed lanes - unsupported

## 5.2.3.7 enum J2735\_2016\_VehicleEventFlags\_t

vehicle event flags representing the vehicle CAN events

## Enumerator

**J2735\_2016\_EVT\_HAZARD\_LIGHTS** J2735\_2016\_EVT\_HAZARD\_LIGHTS - hazard lights activated  
**J2735\_2016\_EVT\_STOPLINE\_VIOLATION** J2735\_2016\_EVT\_STOPLINE\_VIOLATION - stop line violation  
**J2735\_2016\_EVT\_ABS\_ACTIVATED** J2735\_2016\_EVT\_ABS\_ACTIVATED - ABS activated  
**J2735\_2016\_EVT\_TRACTION\_CONTROL** J2735\_2016\_EVT\_TRACTION\_CONTROL - traction control activated  
**J2735\_2016\_EVT\_STABILITY\_CONTROL** J2735\_2016\_EVT\_STABILITY\_CONTROL - stability control activated  
**J2735\_2016\_EVT\_HAZARDOUS\_MATERIAL** J2735\_2016\_EVT\_HAZARDOUS\_MATERIAL - hazardous material  
**J2735\_2016\_EVT\_RESERVED1** J2735\_2016\_EVT\_RESERVED1 - reserved  
**J2735\_2016\_EVT\_HARDBRAKING** J2735\_2016\_EVT\_HARDBRAKING - hard braking  
**J2735\_2016\_EVT\_LIGHTS\_CHANGED** J2735\_2016\_EVT\_LIGHTS\_CHANGED - lights changed  
**J2735\_2016\_EVT\_WIPERS\_CHANGED** J2735\_2016\_EVT\_WIPERS\_CHANGED - wipers changed  
**J2735\_2016\_EVT\_FLATTIRE** J2735\_2016\_EVT\_FLATTIRE - flat tire  
**J2735\_2016\_EVT\_DISABLED\_VEHICLE** J2735\_2016\_EVT\_DISABLED\_VEHICLE - disabled vehicle  
**J2735\_2016\_EVT\_AIRBAG\_DEPLOYMENT** J2735\_2016\_EVT\_AIRBAG\_DEPLOYMENT - airbag deployment

## 5.2.3.8 enum J2735\_2016\_WiperStatus\_t

Wiper status

## Enumerator

**J2735\_2016\_WIPER\_STATUS\_UNAVAIL** J2735\_2016\_WIPER\_STATUS\_UNAVAIL - unavailable  
**J2735\_2016\_WIPER\_STATUS\_OFF** J2735\_2016\_WIPER\_STATUS\_OFF - wipers off  
**J2735\_2016\_WIPER\_INTERMITTENT** J2735\_2016\_WIPER\_INTERMITTENT - wipers intermittent  
**J2735\_2016\_WIPER\_LOW** J2735\_2016\_WIPER\_LOW - wipers low  
**J2735\_2016\_WIPER\_HIGH** J2735\_2016\_WIPER\_HIGH - wipers high  
**J2735\_2016\_WIPER\_WASHER\_IN\_USE** J2735\_2016\_WIPER\_WASHER\_IN\_USE - wipers washer in use  
**J2735\_2016\_WIPER\_AUTOMATIC\_PRESENT** J2735\_2016\_WIPER\_AUTOMATIC\_PRESENT - wipers automatic

### 5.2.3.9 enum j2735\_dsrc\_msgids

List of DSRC message IDS in the J2735

#### Enumerator

**J2735\_2016\_BSM\_MSG\_D** J2735\_2016\_BSM\_MSG\_D - BSM MSG - DER / BER - unsupported  
**J2735\_2016\_TIM\_MSG\_D** J2735\_2016\_TIM\_MSG\_D - TIM MSG - DER / BER - unsupported  
**J2735\_2016\_RTCM\_CORRECTIONS\_MSG\_D** J2735\_2016\_RTCM\_CORRECTIONS\_MSG\_D - RTCM Corrections MSG - DER / BER - unsupported  
**J2735\_2016\_MAP\_MSG\_P** J2735\_2016\_MAP\_MSG\_P - MAP MSG - UPER - supported  
**J2735\_2016\_SPaT\_MSG\_P** J2735\_2016\_SPaT\_MSG\_P - SPAT MSG - UPER - supported  
**J2735\_2016\_TIM\_MSG\_P** J2735\_2016\_TIM\_MSG\_P - TIM MSG - UPER - supported  
**J2735\_2016\_BSM\_MSG\_U** J2735\_2016\_BSM\_MSG\_U - BSM MSG - UPER - supported  
**J2735\_2016\_RTCM\_CORRECTIONS\_MSG\_U** J2735\_2016\_RTCM\_CORRECTIONS\_MSG\_U - RTCM MSG - UPER - supported  
**J2735\_2016\_SRM\_MSG\_U** J2735\_2016\_SRM\_MSG\_U - SRM MSG - UPER - supported  
**J2735\_2016\_SSM\_MSG\_U** J2735\_2016\_SSM\_MSG\_U - SSM MSG - UPER - supported  
**J2735\_2016\_TIM\_MSG\_U** J2735\_2016\_TIM\_MSG\_U - TIM MSG - UPER - supported

### 5.2.3.10 enum J2735Extent

Extent object. Please refer to DE\_Extent of SAEj2735 for further details.

### 5.2.3.11 enum lanedirection

MAP & LANE specific objects / definitions.

lane direction

#### Enumerator

**J2735\_2016\_LANE\_DIRECTION\_UNAVAIL** J2735\_2016\_LANE\_DIRECTION\_UNAVAIL - unavailable  
**J2735\_2016\_LANE\_DIRECTION\_INGRESS** J2735\_2016\_LANE\_DIRECTION\_INGRESS - ingress lane  
**J2735\_2016\_LANE\_DIRECTION\_EGRESS** J2735\_2016\_LANE\_DIRECTION\_EGRESS - egress lane

### 5.2.3.12 enum lanetypecrosswalk

Cross walk attributes

#### Enumerator

**J2735\_2016\_CRW\_CRW\_REVOCABLE\_LANE** J2735\_2016\_CRW\_CRW\_REVOCABLE\_LANE - cross-walk revocable lane  
**J2735\_2016\_CRW\_BICYCLE\_USE\_ONLY** J2735\_2016\_CRW\_BICYCLE\_USE\_ONLY - bicycle use only  
**J2735\_2016\_CRW\_IS\_FLYOVER\_CRW\_LANE** J2735\_2016\_CRW\_IS\_FLYOVER\_CRW\_LANE - cross-walk flyover lane  
**J2735\_2016\_CRW\_FIXED\_CYCLE\_TIME** J2735\_2016\_CRW\_FIXED\_CYCLE\_TIME - fixed cycle time in use  
**J2735\_2016\_CRW\_BIDIRECTIONAL\_CYCLE\_TIME** J2735\_2016\_CRW\_BIDIRECTIONAL\_CYCLE\_TIME - bidirectional cycle time

**J2735\_2016\_CRW\_HAS\_PUSH\_TO\_WALK\_BUTTON** J2735\_2016\_CRW\_HAS\_PUSH\_TO\_WALK\_BUTTON - the crosswalk has a push to talk button

**J2735\_2016\_CRW\_AUDIO\_SUPPORT** J2735\_2016\_CRW\_AUDIO\_SUPPORT - the crosswalk has the audio support

**J2735\_2016\_CRW\_RFSIGNAL\_REQ\_PRESENT** J2735\_2016\_CRW\_RFSIGNAL\_REQ\_PRESENT - the cross walk has the RF signal request

**J2735\_2016\_CRW\_UNSIGNALIZED\_SEGMENTS\_PRESENT** J2735\_2016\_CRW\_UNSIGNALIZED\_SEGMENTS\_PRESENT - unsignallized segments are present

### 5.2.3.13 enum lanetypevehicle

vehicle attributes

Enumerator

**J2735\_2016\_VEH\_IS\_VEHICLE\_REVOCABLE\_LANE** vehicle revocable lanes

**J2735\_2016\_VEH\_IS\_VEHICLE\_FLYOVER\_LANE** vehicle flyover lanes

**J2735\_2016\_VEH\_HOV\_USE\_ONLY** vehicle hov useonly lanes

**J2735\_2016\_VEH\_RESTRICTED\_TO\_BUS\_USE** bus restricted lanes

**J2735\_2016\_VEH\_RESTRICTED\_TO\_TAXI\_USE** taxi restricted lanes

**J2735\_2016\_VEH\_RESTRICTED\_FROM\_PUBLIC\_USE** restricted from public use

## 5.2.4 Function Documentation

### 5.2.4.1 void v2x\_map\_free ( v2x\_msg\_map\_t \* map )

free MAP message

Parameters

in	mapmsg	MAP structure
----	--------	---------------

Description

This function frees the map data structure

Returns

none

### 5.2.4.2 v2x\_status\_t v2x\_msg\_bsm\_decode ( v2x\_msg\_bsm\_t \* bsm, unsigned char \* msgbuf, int msgbuf\_len, BSM\_TYPE \* type, uint32\_t \* oob, int debug )

Decode BSM message and fill the bsm structure.

Parameters

out	bsm	BSM structure, filled up by the lib with decoded BSM data
in	msgbuf	Buffer containing encoded message.

in	<i>msgbuf_len</i>	Length of the message buffer.
out	<i>type</i>	Filled with type of BSM data present.
in	<i>oob</i>	A valid value if there are any outofbounds value

### Description

This function decodes the BSM message and fills BSM structure with decoded data. The 'msgbuf' point to BSM message buffer and 'msgbuf\_len' specifies length of 'msgbuf' Type is set with types of BSM data present which is OR of all the elements present in the BSM. For ex. if BSM part1 is present then type = BSM\_PART1. if BSM part1 and ph, pp and vehiclestatus were present then type = PART1 | PH | PP | VEHSTATUS.

### Returns

V2X\_SUCCESS on success and error number on failure

### Return values

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_EDECODE</i>	Decoder failed

#### 5.2.4.3 v2x\_status\_t v2x\_msg\_bsm\_encode ( v2x\_msg\_bsm\_t \* bsm, BSM\_TYPE type, unsigned char \* msgbuf, int msgbuf\_size, int \* enclen )

Encode BSM message with data from BSM structure.

### Parameters

in	<i>bsm</i>	BSM structure with BSM data for encoding
in	<i>type</i>	BSM type fields that are available in BSM structure
out	<i>msgbuf</i>	Buffer into which the encoded message is copied.
in	<i>msgbuf_size</i>	Maximum size of the buffer.
out	<i>enclen</i>	Length of encoded message.

### Description

This function takes data from BSM structure and encodes it using DER encoder. The encoded data is available in 'msgbuf'. The 'enclen' is set to size of encoded message on success.

### Returns

V2X\_SUCCESS on success and error number on failure

### Return values

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_EENCODE</i>	Encoder failed

#### 5.2.4.4 void v2x\_msg\_bsm\_init ( v2x\_msg\_bsm\_t \* bsm )

Initialize BSM elements to 'unavailable' values



## Parameters

in	<i>bsm</i>	BSM structure
----	------------	---------------

## Description

This function sets all the elements of the BSM structure to unavailable values. The structure can then be used to fill with available values which then can be used for encoding. This is to make sure that 'only' the needed values can be encodable and sent over the air.

## Returns

None

5.2.4.5 `v2x_status_t v2x_msg_eva_decode ( v2x_msg_eva_t * eva, uint8_t * encbytes, int enclen )`

## Decode EVA

## Parameters

in	<i>eva</i>	EVA message
out	<i>encbytes</i>	encoded buffer
out	<i>enclen</i>	encoded buffer length

## Description

This message decodes the encoded EVA into the EVA structure

## Returns

V2X\_SUCCESS on success and V2X\_EINVAL on failure

5.2.4.6 `v2x_status_t v2x_msg_eva_encode ( v2x_msg_eva_t * eva, uint8_t * encbytes, int encbytes_size, int * enc_len )`

## Encode EVA

## Parameters

out	<i>eva</i>	EVA message
out	<i>encbytes</i>	encoded buffer
in	<i>encbytes_size</i>	encoded buffer size
	<i>[out]</i>	enc_len encoded buffer length

## Description

This message encodes the EVA structure into the EVA message. The length of the buffer is copied into the `enc_len`.

## Returns

V2X\_SUCCESS on success and V2X\_EINVAL on failure

5.2.4.7 `void v2x_msg_eva_print ( FILE * fp, v2x_msg_eva_t * eva )`

## Print EVA

**Parameters**

in	<i>fp</i>	file handle to print the EVA structure
in	<i>eva</i>	EVA message

**Description**

This message prints the EVA structure into the file pointed by *fp*. If *fp* is NULL, it will be stderr.

**5.2.4.8** void *v2x\_msg\_eva\_reset* ( *v2x\_msg\_eva\_t* \* *eva* )

**Reset EVA****Parameters**

in	<i>eva</i>	EVA message
----	------------	-------------

**Description**

This function resets the EVA structure

**5.2.4.9** *v2x\_status\_t* *v2x\_msg\_map\_decode* ( *v2x\_msg\_map\_t* \* *map*, *uint8\_t* \* *buf*, *int* *bufsize*, *int* *debug* )

Decode MAP message and fill MAP structure with decoded data.

**Parameters**

out	<i>map</i>	MAP structure to be filled with decoded data
in	<i>buf</i>	Buffer containing encoded data that needs to be decoded.
in	<i>bufsize</i>	Size of buffer

**Description**

This function decodes the MAP message and fills the structure *map* with the decoded data.

**Returns**

V2X\_SUCCESS on success or error number on failure.

**Return values**

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_EDECODE</i>	Decoder failed

**5.2.4.10** *v2x\_status\_t* *v2x\_msg\_map\_encode* ( *v2x\_msg\_map\_t* \* *map*, *uint8\_t* \* *msgbuf*, *int* *msgbuf\_size*, *int* \* *enclen* )

Encode MAP message with data from MAP structure.

**Parameters**

in	<i>map</i>	MAP data to be encoded
----	------------	------------------------

out	<i>msgbuf</i>	Buffer into which the encoded message is copied.
in	<i>msgbuf_size</i>	Size of the buffer.
out	<i>enclen</i>	Length of encoded message.

**Description**

This function takes data from TIM structure and encodes it using DER encoder. The encoded data is available in 'msgbuf'. The 'enclen' is set to size of encoded message on success.

**Returns**

V2X\_SUCCESS on success or error number on failure.

**Return values**

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_EENCODE</i>	Encoder failed

**5.2.4.11 void v2x\_msg\_map\_init ( v2x\_msg\_map\_t \* map\_msg )**

Initialize MAP msg structure with unavailable values

**Parameters**

in	<i>map_msg</i>	MAP structure to be initialized
----	----------------	---------------------------------

**Returns**

None

**5.2.4.12 v2x\_status\_t v2x\_msg\_psm\_decode ( v2x\_msg\_psm\_t \* psm, uint8\_t \* enc\_buf, int buflen, uint32\_t \* oob, int debug )**

Decode PSM message

**Parameters**

out	<i>psm</i>	PSM data structure
in	<i>enc_buf</i>	Encoded buffer
in	<i>buflen</i>	Encoded buffer length
out	<i>oob</i>	out of bounds variable
in	<i>debug</i>	debug variable

**Description**

This function decodes the PSM encoded buffer in the enc\_buf, into psm structure. The debug variable can be 1 or 0. If 1, it prints out the Raw decoded payload on the screen for debugging purposes

**Returns**

V2X\_SUCCESS on success and V2X\_EINVAL on failure

**5.2.4.13 v2x\_status\_t v2x\_msg\_psm\_encode ( v2x\_msg\_psm\_t \* psm, unsigned char \* encoded\_buffer, int bufsz, int \* encoded\_len )**

Encode PSM message

**Parameters**

in	<i>psm</i>	PSM data structure
out	<i>encoded_buffer</i>	Encoded buffer
in	<i>bufsize</i>	buffer size of the encoded_buffer
out	<i>encoded_len</i>	encoded length

**Description**

This function encodes the PSM message passed as psm structure, into encoded\_buffer. The output encoded length is stored in encoded\_len

**Returns**

V2X\_SUCCESS on success and V2X\_EINVAL on failure

**5.2.4.14 void v2x\_msg\_psm\_reset ( v2x\_msg\_psm\_t \* psm )**

Reset PSM message

**Parameters**

in	<i>psm</i>	PSM data structure
----	------------	--------------------

**Description**

This function resets the PSM data structure. Call this before calling v2x\_msg\_psm\_encode or v2x\_msg\_psm\_decode API

**5.2.4.15 v2x\_status\_t v2x\_msg\_rtcn\_decode ( v2x\_msg\_rtcn\_t \* rtcn, uint8\_t \* msgbuf, int msgbuf\_len, int \* declen )**

Decode RTCM message and fill RTCM structure with decoded data.

**Parameters**

out	<i>rtcn</i>	Structure filled with decoded RTCM data.
in	<i>msgbuf</i>	Message buffer containing encoded RTCM message.
in	<i>msgbuf_len</i>	Length of message buffer.
in	<i>declen</i>	Length of 'msgbuf' consumed by decoder.

**Description**

This function decodes the RTCM message and fills RTCM structure with decoded data. The 'msgbuf' point to RTCM message buffer and 'msgbuf\_len' specifies length of 'msgbuf'

**Returns**

V2X\_SUCCESS on success or error number on failure.

**Return values**

<i>V2X_SUCCESS</i>	API function was successful.
--------------------	------------------------------

<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_DECODE</i>	Decoder failed.

**5.2.4.16** `v2x_status_t v2x_msg_rtcn_encode ( v2x_msg_rtcn_t * rtcn, unsigned char * msgbuf, int msgbuf_size, int * enclen )`

Encode RTCM message with data from RTCM structure.

#### Parameters

in	<i>rtcn</i>	Structure containing RTCM corrections data.
out	<i>msgbuf</i>	Buffer into which the encoded message is copied.
in	<i>msgbuf_size</i>	Maximum size of the buffer.
out	<i>enclen</i>	Length of encoded message.

#### Description

This function takes data from RTCM structure and encodes it using DER encoder. The encoded data is available in 'msgbuf'. The 'enclen' is set to size of encoded message on success.

#### Returns

V2X\_SUCCESS on success or error number on failure.

#### Return values

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_EENCODE</i>	Encoder failed.

**5.2.4.17** `void v2x_msg_rtcn_print ( FILE * fp, v2x_msg_rtcn_t * rtcn )`

Print RTCM structure

#### Parameters

in	<i>fp</i>	file handle to print the RTCM structure
in	<i>rtcn</i>	rtcm message

#### Description

This API prints the RTCM structure to the file handle. If the file handle is NULL, it will be stderr.

**5.2.4.18** `int v2x_msg_spat_decode ( v2x_msg_spat_t * spatmsg, uint8_t * buf, int len, int * dec_len, int debug )`

Decode SPAT message and fill the input SPAT structure.

#### Parameters

out	<i>spatmsg</i>	Structure to be filled with decoded data
-----	----------------	--

in	<i>buf</i>	data to be decoded
in	<i>len</i>	size of buf
out	<i>dec_len</i>	to be filled with the size of decoded data

**Description** This function decodes the SPAT message and fills the

structure *spatmsg* with the decoded data. *dec\_len* is updated with the size of the decoded data

#### Returns

V2X\_SUCCESS on success or error number on failure.

#### Return values

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_EDECODE</i>	Decoder failed

#### 5.2.4.19 `v2x_status_t v2x_msg_spat_encode ( v2x_msg_spat_t * spatmsg, uint8_t * buf, int len, int * enc_len )`

Encode SPAT message according to input SPAT structure.

#### Parameters

in	<i>spatmsg</i>	SPAT data to be encoded
out	<i>buf</i>	Buffer to fill with encoded data
in	<i>len</i>	Size of buf
out	<i>enc_len</i>	To be filled with encoded data length

#### Description

This function encodes the SPAT message and fills the input buffer with the encoded data. The *enc\_len* is filled with the size of the encoded data length. encoder type is the type of encoder to be used.

#### Returns

V2X\_SUCCESS on success or error number on failure.

#### Return values

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_EENCODE</i>	Encoder failed

#### 5.2.4.20 `v2x_status_t v2x_msg_tim_decode ( v2x_msg_tim_t * tim, int type, uint8_t * msgbuf, int msgbuf_len, uint32_t * oob, int * declen, int debug )`

Decode TIM message and fill TIM structure with decoded data.

#### Parameters

out	<i>tim</i>	Structure filled with decoded TIM data.
in	<i>type</i>	Reserved for future use. Set to 0.
in	<i>msgbuf</i>	Message buffer containing encoded TIM message.
in	<i>msgbuf_len</i>	Length of message buffer.
in	<i>declen</i>	Length of 'msgbuf' consumed by decoder.
out	<i>oob</i>	Indicates if data crossed the bounds of min and max values of a parameter.

**Description**

This function decodes the TIM message and fills TIM structure with decoded data. The 'msgbuf' point to TIM message buffer and 'msgbuf\_len' specifies length of 'msgbuf'

**Note**

This function also verifies the CRC value of the TIM message.

**Returns**

V2X\_SUCCESS on success or error number on failure.

**Return values**

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_DECODE</i>	Decoder failed.

#### 5.2.4.21 `v2x_status_t v2x_msg_tim_encode ( v2x_msg_tim_t * tim, int type, uint8_t * msgbuf, int msgbuf_size, int * enclen )`

Encode TIM message with data from TIM structure.

**Parameters**

in	<i>tim</i>	Structure containing TIM data.
in	<i>type</i>	Reserved for future use. Set to 0.
out	<i>msgbuf</i>	Buffer into which the encoded message is copied.
in	<i>msgbuf_size</i>	Maximum size of the buffer.
out	<i>enclen</i>	Length of encoded message.

**Description**

This function takes data from TIM structure and encodes it using DER encoder. The encoded data is available in 'msgbuf'. The 'enclen' is set to size of encoded message on success.

**Note**

The msgbuf should reasonably be a long msgbuf\_size size. Preferred size is 2000 bytes. The application need not worry about the CRC value the TIM has. It is handled by the library and no user intervention is required

**Returns**

V2X\_SUCCESS on success or error number on failure.

## Return values

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EINVAL</i>	Invalid argument passed.
<i>V2X_EFAULT</i>	An invalid user space address was specified for an argument.
<i>V2X_EENCODE</i>	Encoder failed.

## 5.2.4.22 void v2x\_msg\_tim\_free ( v2x\_msg\_tim\_t \* tim )

## Free TIM message

## Parameters

in	<i>tim</i>	TIM structure from the application
----	------------	------------------------------------

## Description

This function frees up any pointers that are allocated in the TIM structure.

## Returns

V2X\_SUCCESS on success or error number on failure.

## Return values

<i>V2X_SUCCESS</i>	API function was successful.
<i>V2X_EFAULT</i>	An invalid userspace address was specified for an argument.

## 5.2.4.23 v2x\_status\_t v2x\_pvd\_decode ( v2x\_msg\_pvd\_t \* pvd, int type, uint8\_t \* encbuf, int buflen, uint32\_t \* oob, int \* dec\_len )

## Decode PVD message

## Parameters

out	<i>pvd</i>	PVD message
in	<i>type</i>	unused
in	<i>encbuf</i>	encoded buffer
in	<i>buflen</i>	encoded buffer length
out	<i>oob</i>	unused
out	<i>dec_len</i>	unused

## Description

This function decodes the PVD message into the data structure.

## Returns

V2X\_SUCCESS on success and V2X\_EINVAL on failure

## 5.2.4.24 v2x\_status\_t v2x\_pvd\_encode ( v2x\_msg\_pvd\_t \* pvd, uint8\_t \* encbuf, int size, int \* encbuf\_len )

## Encode PVD message



**Parameters**

in	<i>pvd</i>	PVD message
out	<i>encbuf</i>	encoded buffer
in	<i>size</i>	encoded buffer length size
out	<i>encbuf_len</i>	encoded buffer length

**Description**

This function encodes the PVD data structure into the message. The *encbuf\_len* contains the final encoded length

**Returns**

V2X\_SUCCESS on success and V2X\_EINVAL on failure

5.2.4.25 void v2x\_pvd\_reset ( v2x\_msg\_pvd\_t \* *pvd* )

Reset PVD structure

**Parameters**

in	<i>pvd</i>	PVD message
----	------------	-------------

**Description**

This function resets the PVD structure

5.2.4.26 void v2x\_pvd\_snapshot\_reset ( J2735\_2016\_ProbeSnapshot\_t \* *snapshot* )

Reset PVD snapshot

**Parameters**

in	<i>snapshot</i>	PVD snapshot
----	-----------------	--------------

**Description**

This function resets the PVD snapshot structure

5.2.4.27 void v2x\_spat\_free ( v2x\_msg\_spat\_t \* *spat* )

Free the spat message.

**Parameters**

in	<i>spat</i>	SPAT data structure
----	-------------	---------------------

**Description** This function frees up the allocate spat

structure.

**Returns**

none

5.2.4.28 v2x\_status\_t v2x\_srm\_decode ( v2x\_msg\_srm\_t \* *srm*, uint8\_t \* *encbuf*, int \* *encbuf\_len*, int *size* )

Decode SRM

**Parameters**

out	<i>srn</i>	SRM message
in	<i>encbuf</i>	Encoded buffer
in	<i>encbuf_len</i>	keep valid pointer, unused
in	<i>size</i>	length of the encoded buffer

**Description**

This API decodes the SRM message into the SRM structure.

**Returns**

V2X\_SUCCESS on success and V2X\_EINVAL on failure

**5.2.4.29** `v2x_status_t v2x_srm_encode ( v2x_msg_srm_t * srm, uint8_t * encbuf, int * encbuf_len, int size )`

**Encode SRM****Parameters**

in	<i>srn</i>	SRM message
out	<i>encbuf</i>	Encoded buffer
out	<i>encbuf_len</i>	Encoded buffer length
in	<i>size</i>	size of encoded buffer

**Description**

This API encodes the SRM structure into the SRM message. The *encbuf\_len* contains the final encoded SRM length

**Returns**

V2X\_SUCCESS on success and V2X\_EINVAL on failure

**5.2.4.30** `void v2x_srm_reset ( v2x_msg_srm_t * srm )`

**Reset SRM****Parameters**

in	<i>srn</i>	SRM message
----	------------	-------------

**Description**

This API resets the SRM message

**5.2.4.31** `v2x_status_t v2x_ssm_decode ( v2x_msg_ssm_t * ssm, int type, uint8_t * encbuf, int buflen, uint32_t * oob, int * dec_len )`

**Decode SSM**

**Parameters**

out	<i>ssm</i>	SSM message
in	<i>type</i>	unused
in	<i>encbuf</i>	encoded buffer
in	<i>buflen</i>	encoded buffer length
out	<i>oob</i>	unused
out	<i>dec_len</i>	unused

**Description**

This message decodes the SSM message into SSM structure.

**Returns**

V2X\_SUCCESS on success and V2X\_EINVAL on failure

5.2.4.32 `v2x_status_t v2x_ssm_encode ( v2x_msg_ssm_t * ssm, uint8_t * encbuf, int * encbuf_len, int size )`

**Encode SSM****Parameters**

in	<i>ssm</i>	SSM message
out	<i>encbuf</i>	encode buffer
out	<i>encbuf_len</i>	encode buffer length
in	<i>size</i>	encode buffer size

**Description**

This message encodes the SSM structure into SSM message. The encoded message length is kept into `encbuf_len`

**Returns**

V2X\_SUCCESS on success and V2X\_EINVAL on failure

5.2.4.33 `void v2x_ssm_reset ( v2x_msg_ssm_t * ssm )`

**Reset SSM****Parameters**

in	<i>ssm</i>	SSM message
----	------------	-------------

**Description**

This message resets the SSM

## 5.3 The V2X Message API (EU)

### Data Structures

- struct [v2x\\_etsi\\_cam](#)
- struct [v2x\\_etsi\\_denm](#)
- struct [v2x\\_etsi\\_map](#)
- struct [v2x\\_etsi\\_spat](#)

### Functions

- [v2x\\_status\\_t v2x\\_etsi\\_cam\\_encode](#) (struct [v2x\\_etsi\\_cam](#) \*cam, uint8\_t \*encbuf, int buf\_len, int \*encbuf\_len)
- [v2x\\_status\\_t v2x\\_etsi\\_cam\\_decode](#) (struct [v2x\\_etsi\\_cam](#) \*cam, uint8\_t \*encbuf, int encbuf\_len)
- void [v2x\\_etsi\\_cam\\_reset](#) (struct [v2x\\_etsi\\_cam](#) \*cam)
- void [v2x\\_etsi\\_cam\\_print](#) (FILE \*fp, struct [v2x\\_etsi\\_cam](#) \*cam)
- [v2x\\_status\\_t v2x\\_etsi\\_denm\\_encode](#) (struct [v2x\\_etsi\\_denm](#) \*denm, uint8\_t \*encbuf, int buf\_len, int \*encbuf\_len)
- [v2x\\_status\\_t v2x\\_etsi\\_denm\\_decode](#) (struct [v2x\\_etsi\\_denm](#) \*denm, uint8\_t \*encbuf, int encbuf\_len)
- void [v2x\\_etsi\\_reset\\_denm](#) (struct [v2x\\_etsi\\_denm](#) \*denm)
- void [v2x\\_etsi\\_print\\_denm](#) (FILE \*fp, struct [v2x\\_etsi\\_denm](#) \*denm)
- void [v2x\\_etsi\\_denm\\_dumphex](#) (uint8\_t \*encbuf, int encbuf\_len)
- [v2x\\_status\\_t v2x\\_etsi\\_map\\_encode](#) (struct [v2x\\_etsi\\_map](#) \*map, uint8\_t \*encbuf, int buf\_len, int \*encbuf\_len)
- [v2x\\_status\\_t v2x\\_etsi\\_map\\_decode](#) (struct [v2x\\_etsi\\_map](#) \*map, uint8\_t \*encbuf, int encbuf\_len)
- void [v2x\\_etsi\\_map\\_reset](#) (struct [v2x\\_etsi\\_map](#) \*map)
- void [v2x\\_etsi\\_map\\_free](#) (struct [v2x\\_etsi\\_map](#) \*map)
- void [v2x\\_etsi\\_map\\_print](#) (struct [v2x\\_etsi\\_map](#) \*map, FILE \*fp)
- [v2x\\_status\\_t v2x\\_etsi\\_spat\\_encode](#) (struct [etsi\\_spat](#) \*spat, uint8\_t \*encbuf, int buf\_len, int \*encbuf\_len)
- [v2x\\_status\\_t v2x\\_etsi\\_spat\\_decode](#) (struct [etsi\\_spat](#) \*spat, uint8\_t \*encbuf, int encbuf\_len)
- void [v2x\\_etsi\\_reset\\_spat](#) (struct [v2x\\_etsi\\_spat](#) \*spat)
- void [v2x\\_etsi\\_spat\\_print](#) (struct [v2x\\_etsi\\_spat](#) \*spat, FILE \*fp)

#### 5.3.1 Detailed Description

Author: Devendra (Dev) Naga

Written by Devendra (Dev) Naga 2014-2015

Savari inc Allrights reserved

#### 5.3.2 Function Documentation

##### 5.3.2.1 [v2x\\_status\\_t v2x\\_etsi\\_cam\\_decode](#) ( struct [v2x\\_etsi\\_cam](#) \* cam, uint8\_t \* encbuf, int encbuf\_len )

Decode CAM data and fill the CAM data structure

##### Parameters

out	<i>cam</i>	Structure to be filled with decoded CAM
in	<i>encbuf</i>	Buffer with encoded CAM
in	<i>encbuf_len</i>	Length of the encoded CAM

##### Description

This function decoded the CAM message and fills the CAM data structure with decoded data. The 'encbuf' contains the encoded CAM data and 'encbuf\_len' specifies the length of encoded CAM.

**Returns**

V2X\_SUCCESS on success and error number on failure

**Return values**

V2X_SUCCESS	API function is successful
V2X_EINVAL	Invalid argument passed
V2X_EDECODE	Decode failed

### 5.3.2.2 v2x\_status\_t v2x\_etsi\_cam\_encode ( struct v2x\_etsi\_cam \* cam, uint8\_t \* encbuf, int buf\_len, int \* encbuf\_len )

Encode CAM with data from CAM structure

**Parameters**

in	cam	Structure containing CAM data
out	encbuf	Buffer in which encoded CAM is filled
in	buf_len	Maximum size of the 'encbuf'
out	encbuf_len	Length of encoded CAM

**Description**

This function takes data from CAM data structure and encodes it using UPER encoder. The encoded data is then filled into the 'encbuf'. The 'encbuf\_len' is set the length of encoded CAM on upon success

**Returns**

V2X\_SUCCESS on success and error number on failure

**Return values**

V2X_SUCCESS	API function is successful
V2X_EINVAL	Invalid argument passed
V2X_ENOMEM	Failed to allocate memory
V2X_EENCODE	Encode failed

### 5.3.2.3 void v2x\_etsi\_cam\_print ( FILE \* fp, struct v2x\_etsi\_cam \* cam )

Print the CAM structure

**Parameters**

in	fp	File handle to print the structure to
in	cam	the CAM data structure

**Description**

This function prints the CAM data structure on to the fp. If fp is NULL, it prints to stderr

**Returns**

void

### 5.3.2.4 void v2x\_etsi\_cam\_reset ( struct v2x\_etsi\_cam \* cam )

Reset the CAM data structure

**Parameters**

in	cam	The CAM data structure
----	-----	------------------------

**Description**

This function resets the CAM data structure to default values to avoid garbage values

**Returns**

void

### 5.3.2.5 v2x\_status\_t v2x\_etsi\_denm\_decode ( struct v2x\_etsi\_denm \* denm, uint8\_t \* encbuf, int encbuf\_len )

Decode DEN message and fill DEN structure

**Parameters**

out	denm	Structure to be filled with decoded data
in	encbuf	Buffer containing encoded DEN message
in	encbuf_len	Length of the buffer 'encbuf'

**Description**

This function decodes the DEN message and fills the DEN structure with decoded data. The 'encbuf' contains the encoded DEN and 'encbuf\_len' is the length of the 'encbuf'

**Returns**

V2X\_SUCCESS on success and error number on failure.

**Return values**

V2X_SUCCESS	API function is successful
V2X_EINVAL	Invalid argument passed
V2X_EDECODE	Decode failed

### 5.3.2.6 void v2x\_etsi\_denm\_dumphex ( uint8\_t \* encbuf, int encbuf\_len )

Print hexdump of encoded DENM

**Parameters**

in	encbuf	Buffer with encoded DENM filled by 'v2x_etsi_denm_encode' API
in	encbuf_len	Length of the encode DENM

**Description**

This function prints hexdump of the encoded DENM stored in the buffer 'encbuf'. This buffer is filled by the function 'v2x\_etsi\_denm\_encode'

**Returns**

void

### 5.3.2.7 v2x\_status\_t v2x\_etsi\_denm\_encode ( struct v2x\_etsi\_denm \* denm, uint8\_t \* encbuf, int buf\_len, int \* encbuf\_len )

Encode DEN message with data from DEN structure

**Parameters**

in	<i>denm</i>	Structure containing DEN data
out	<i>encbuf</i>	Buffer in which encoded DEN message is filled
in	<i>buf_len</i>	Maximum size of the buffer 'encbuf'
out	<i>encbuf_len</i>	Length of the encoded message

**Description**

This function takes data from DEN structure and encodes it using UPER encoder. The encoded data is filled into the 'encbuf'. The 'encbuf\_len' is set to the length of encoded DEN message on success.

**Returns**

V2X\_SUCCESS on success or error number on failure.

**Return values**

<i>V2X_SUCCESS</i>	API function is successful
<i>V2X_EINVAL</i>	Invalid argument passed
<i>V2X_ENOMEM</i>	Failed to allocate memory
<i>V2X_EENCODE</i>	Encode failed

### 5.3.2.8 v2x\_status\_t v2x\_etsi\_map\_decode ( struct v2x\_etsi\_map \* map, uint8\_t \* encbuf, int encbuf\_len )

Decode MAP message and fill MAP data structure

**Parameters**

out	<i>map</i>	Structure to be filled with decoded data
in	<i>encbuf</i>	Buffer containing encoded MAP message
in	<i>encbuf_len</i>	Length of the buffer 'encbuf'

**Description**

This function decodes the MAP message and fills the MAP data structure with decoded data. The 'encbuf' contains the encoded MAP message and 'encbuf\_len' specifies the length of 'encbuf'

**Note**

The MAP data structure is dynamic and must be freed after decoding. Call 'v2x\_etsi\_map\_free' API to free it.

**Returns**

V2X\_SUCCESS on success and error number on failure

**Return values**

<i>V2X_SUCCESS</i>	API function is successful
<i>V2X_EINVAL</i>	Invalid argument is passed
<i>V2X_EDECODE</i>	Decode failed

### 5.3.2.9 v2x\_status\_t v2x\_etsi\_map\_encode ( struct v2x\_etsi\_map \* map, uint8\_t \* encbuf, int buf\_len, int \* encbuf\_len )

**Parameters**

in	<i>map</i>	Structure containing MAP data
out	<i>encbuf</i>	Buffer in which encoded MAP message if filled
in	<i>buf_len</i>	Maximum size of the buffer 'encbuf'
out	<i>encbuf_len</i>	Length of the encoded MAP message

**Description**

This function takes data from MAP data structure and encoded it using UPER encoder. The encoded data is then filled into the 'encbuf'. The 'encbuf\_len' is set to length of encoded MAP message on success

**Note**

The MAP data structure is dynamic and must be freed after encoding. Call 'v2x\_etsi\_map\_free' API to free it.

**Returns**

V2X\_SUCCESS on success and error number on failure

**Return values**

<i>V2X_SUCCESS</i>	API function is successful
<i>V2X_EINVAL</i>	Invalid argument is passed
<i>V2X_ENOMEM</i>	Failed to allocate memory
<i>V2X_EENCODE</i>	Encode failed

**5.3.2.10 void v2x\_etsi\_map\_free ( struct v2x\_etsi\_map \* map )**

Free the memory allocated to MAP data structure

**Parameters**

in	<i>map</i>	MAP data structure
----	------------	--------------------

**Description**

This function frees the memory allocated to the MAP data structure

**Returns**

void

**5.3.2.11 void v2x\_etsi\_map\_print ( struct v2x\_etsi\_map \* map, FILE \* fp )**

Print the contents of MAP data structures

**Parameters**

in	<i>fp</i>	the file handle to print the MAP structure
in	<i>map</i>	MAP data structure

**Description**

This function prints the contents of the MAP data structure. This can be used to cross check the data before encoding and after decoding

**Returns**

void



5.3.2.12 void v2x\_etsi\_map\_reset ( struct v2x\_etsi\_map \* *map* )

Reset the MAP data structure

**Parameters**

<i>in</i>	<i>map</i>	MAP data structure
-----------	------------	--------------------

**Description**

This function resets the MAP data structre to defaults to avoid garbage values

**Returns**

void

**5.3.2.13** void v2x\_etsi\_print\_denm ( FILE \* *fp*, struct v2x\_etsi\_denm \* *denm* )

Print the DEN data structure

**Parameters**

<i>in</i>	<i>fp</i>	The file handle to print the DENM
<i>in</i>	<i>denm</i>	The DEN data structure

**Description**

This function is used to print the DEN data structure. This can be used to check the values filled in the structure before/after encoding/decoding.

**Returns**

void

**5.3.2.14** void v2x\_etsi\_reset\_denm ( struct v2x\_etsi\_denm \* *denm* )

Reset the DEN Message Data Structure

**Parameters**

<i>in</i>	<i>denm</i>	The DEN data structure
-----------	-------------	------------------------

**Description**

This function is used to reset the DEN data structure to default values to avoid garbage values.

**Returns**

void

**5.3.2.15** void v2x\_etsi\_reset\_spat ( struct v2x\_etsi\_spat \* *spat* )

Reset SPAT data structure

**Parameters**

--

in	<i>spat</i>	SPAT data structure
----	-------------	---------------------

**Description**

This function resets the SPAT data structure to default values to avoid garbage values

**Returns**

void

### 5.3.2.16 `v2x_status_t v2x_etsi_spat_decode ( struct etsi_spat * spat, uint8_t * encbuf, int encbuf_len )`

Decode SPAT message and fill SPAT structure

**Parameters**

out	<i>spat</i>	Structure to be filled with decoded data
in	<i>encbuf</i>	Buffer containing encoded SPAT message
in	<i>encbuf_len</i>	Length of the buffer 'encbuf'

**Description**

This function decodes the SPAT message and fills the SPAT structure with decoded data. The 'encbuf' contains the encoded SPAT and 'encbuf\_len' is the length of the 'encbuf'

**Returns**

V2X\_SUCCESS on success and error number on failure

**Return values**

<i>V2X_SUCCESS</i>	API function is successful
<i>V2X_EINVAL</i>	Invalid argument is passed
<i>V2X_EDECODE</i>	Decode failed

### 5.3.2.17 `v2x_status_t v2x_etsi_spat_encode ( struct etsi_spat * spat, uint8_t * encbuf, int buf_len, int * encbuf_len )`

Encode SPAT message with data from SPAT data structure

**Parameters**

in	<i>spat</i>	Structure containing SPAT data
out	<i>encbuf</i>	Buffer in which encoded SPAT data is filled
in	<i>buf_len</i>	Maximum size of the buffer 'encbuf'
out	<i>encbuf_len</i>	Length of the encoded SPAT

**Description**

This function takes data from SPAT structure and encodes it using UPER encoder. The encoded data is filled into 'encbuf'. The 'encbuf\_len' is set to the length of encoded SPAT message on success

**Returns**

V2X\_SUCCESS on success and error number on failure

## Return values

<i>V2X_SUCCESS</i>	API function is successful
<i>V2X_EINVAL</i>	Invalid argument is passed
<i>V2X_ENOMEM</i>	Failed to allocate memory
<i>V2X_EENCODE</i>	Encode failed

5.3.2.18 void v2x\_etsi\_spat\_print ( struct v2x\_etsi\_spat \* *spat*, FILE \* *fp* )

Print SPAT data structure

## Parameters

in	<i>fp</i>	the file pointer to print the SPAT structure
in	<i>spat</i>	SPAT data structure

## Description

This function prints the data in the SPAT data structure. This can be used to print the structure before encoding/ after decoding

## Returns

void

## 5.4 The V2X Network and Transport module API (EU)

### Functions

- [v2x\\_status\\_t v2x\\_btp\\_init](#) (int port, void \*\*btp\_handle)
- [v2x\\_status\\_t v2x\\_btp\\_get\\_sock\\_id](#) (void \*btp\_handler, int \*sock)
- [v2x\\_status\\_t v2x\\_btp\\_transmit](#) (void \*btp\_handler, BTPReq\_t \*req)
- [v2x\\_status\\_t v2x\\_btp\\_receive](#) (void \*btp\_handler, void \*app\_ctx, void(\*recv\_callback)(BTPInd\_t \*ind, void \*app\_ctx))
- [v2x\\_status\\_t v2x\\_btp\\_deinit](#) (void \*btp\_handle)

### 5.4.1 Detailed Description

The V2X Network and Transport module defines the data structures and API's for

- WAVE WSMP & WME services (1609.3)
- ETSI GeoNetworking/BTP Service

### 5.4.2 Function Documentation

#### 5.4.2.1 [v2x\\_status\\_t v2x\\_btp\\_deinit](#) ( void \* *btp\_handle* )

Close the connection.

#### Description

This API is used to close the connection with GeoNetworking layer via BTP

#### Parameters

<i>in</i>	<i>btp_handler</i>	The handler returned by <a href="#">v2_btp_init</a> API
-----------	--------------------	---

#### Returns

- Returns V2X\_SUCCESS on success and error number on failure

#### Return values

<i>V2X_SUCCESS</i>	Deinit is successful
--------------------	----------------------

#### 5.4.2.2 [v2x\\_status\\_t v2x\\_btp\\_get\\_sock\\_id](#) ( void \* *btp\_handler*, int \* *sock* )

Get the socket descriptor to communicate with GeoNetworking

#### Description

This API returns a socket descriptor to communicate with GeoNetworking layer

#### Parameters

in	<i>btp_handler</i>	The handler returned by v2x_btp_init API
out	<i>sock</i>	Socket descriptor to communicate with GeoNetworking.

**Returns**

- Returns V2X\_SUCCESS on success and error number on failure

**Return values**

<i>V2X_SUCCESS</i>	API Operation is successful
<i>V2X_EINVAL</i>	Invalid argument passed
<i>V2X_EFAULT</i>	Invalid address(NULL pointer)

**5.4.2.3 v2x\_status\_t v2x\_btp\_init ( int port, void \*\* btp\_handle )**

Open a connection with BTP

**Description**

The v2x\_btp\_init initialises and connects to BTP layer and returns a btp\_handler. This handler is further used to transmit/receive messages from geonetworking layer.

**Parameters**

in	<i>port</i>	Port, a unique number in the range 0-65535 to identify a packet.
out	<i>btp_handler</i>	The handler that is filled by this API This handler contains the connection information to connect to GeoNetworking layer

**Returns**

- Returns V2X\_SUCCESS on success and error number on failure

**Return values**

<i>V2X_SUCCESS</i>	API operation was successful
<i>V2X_EIO</i>	Connection Failed
<i>V2X_ENOMEM</i>	Failed to allocate memory

**5.4.2.4 v2x\_status\_t v2x\_btp\_receive ( void \* btp\_handler, void \* app\_ctx, void (\*)(BTPInd\_t \*ind, void \*app\_ctx) rcv\_callback )**

Receive packets from GeoNetwork layer

**Description**

This API is used to receive packets from GeoNetworking layer via BTP

**Parameters**

in	<i>btp_handler</i>	The handler returned by v2x_btp_init API
in	<i>app_ctx</i>	User context to send it to rcv_callback

in	<i>recv_callback</i>	Call back function in which the packets are received
----	----------------------	--

**Returns**

- Returns V2X\_SUCCESS on success and error number on failure

**Return values**

<i>V2X_SUCCESS</i>	Reception of packet from GeoNetworking layer is successful
<i>V2X_EIO</i>	Error in reception

**5.4.2.5 v2x\_status\_t v2x\_btp\_transmit ( void \* btp\_handler, BTPReq\_t \* req )**

Transmit packets to GeoNetworking layer via BTP

**Description**

This API is used to transmit packets to GeoNetworking layer via BTP.

**Parameters**

in	<i>btp_handler</i>	The handler returned by v2x_btp_init API
in	<i>req</i>	Request of type BTPReq_t * where the packet attributes(BTP Type, Port etc) are set

**Returns**

- Returns V2X\_SUCCESS on success and error number on failure

**Return values**

<i>V2X_SUCCESS</i>	Transmission to GeoNetworking layer is successful
<i>V2X_EIO</i>	Error in Transmission or Packet size exceeded maximum

## 5.5 The V2X Network and Transport module API (US)

### Data Structures

- struct [v2x\\_wme\\_reg\\_req](#)
- struct [v2x\\_wsmp\\_rcv\\_indication](#)
- struct [savari1609Wra](#)
- struct [savari1609WsaService](#)
- struct [v2x\\_wsmp\\_send\\_req](#)
- struct [v2x\\_wsmp\\_callbacks](#)

### Typedefs

- typedef [savari\\_socket\\_desc\\_t](#) [v2x\\_wsmp\\_handler\\_t](#)
- typedef struct [v2x\\_wsmp\\_rcv\\_indication](#) [v2x\\_wsmp\\_rcv\\_indication\\_t](#)
- typedef struct [v2x\\_wsmp\\_callbacks](#) [v2x\\_wsmp\\_callbacks\\_t](#)

### Enumerations

- enum [LIBWME\\_RC\\_RESULT\\_CODE](#) {  
[LIBWME\\_RC\\_ACCEPTED](#), [LIBWME\\_RC\\_INVALID\\_PARAMETERS](#),  
[LIBWME\\_RC\\_UNSPECIFIED](#) }
- enum {  
[LIBWME\\_USER\\_AUTOACCESS\\_ONMATCH](#), [LIBWME\\_USER\\_AUTOACCESS\\_UNCOND](#),  
[LIBWME\\_USER\\_NOSCHACCESS](#) }  
*Requested channel access type.*
- enum {  
[LIBWME\\_ACTION\\_ADD](#), [LIBWME\\_ACTION\\_DELETE](#),  
[LIBWME\\_ACTION\\_CHANGE](#) }  
*enums for the registration confirmation*
- enum {  
[P1609\\_CHANNEL\\_ACCESS\\_CONTINUOUS](#) = 0, [P1609\\_CHANNEL\\_ACCESS\\_ALTERNATING](#),  
[P1609\\_CHANNEL\\_ACCESS\\_ONMATCH](#), [P1609\\_CHANNEL\\_ACCESS\\_NOSCHACCESS](#),  
[P1609\\_CHANNEL\\_ACCESS\\_UNKNOWN](#) = 0xFF }
- enum { [P1609\\_WSA\\_UNSECURED](#) = 0, [P1609\\_WSA\\_SECURED](#) = 1 }

### Functions

- struct [v2x\\_wme\\_reg\\_req](#) [\\_\\_attribute\\_\\_\(\(packed\)\)](#)
- struct [savari1609WsaService](#) [\\_\\_attribute\\_\\_\(\(packed\)\)](#)
- [v2x\\_status\\_t](#) [v2x\\_wme\\_open](#) (char \*wme\_ip, char \*iface, [v2x\\_wsmp\\_handler\\_t](#) \*wme\_handler)
- [v2x\\_status\\_t](#) [v2x\\_wme\\_register\\_user](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, [v2x\\_wme\\_reg\\_req\\_t](#) \*wme\_req)
- [v2x\\_status\\_t](#) [v2x\\_wme\\_unregister\\_user](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, [v2x\\_wme\\_reg\\_req\\_t](#) \*wme\_req)
- [v2x\\_status\\_t](#) [v2x\\_wme\\_user\\_service\\_confirm](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, int action, [v2x\\_wme\\_reg\\_req\\_t](#) \*wme\_req)
- [v2x\\_status\\_t](#) [v2x\\_wsmp\\_send](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, [v2x\\_wsmp\\_send\\_req\\_t](#) \*msg\_param, [uint8\\_t](#) \*msgbuf)
- [v2x\\_status\\_t](#) [v2x\\_wsmp\\_rcv](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, [v2x\\_wsmp\\_callbacks\\_t](#) \*wme\_cbs, void \*ctx)
- [v2x\\_status\\_t](#) [v2x\\_wsmp\\_convert\\_psid\\_be](#) ([uint32\\_t](#) psid, [uint32\\_t](#) \*psid\_be)



## Variables

- enum { ... } [LIBWME\\_USERREQUEST](#)  
*Requested channel access type.*
- enum { ... } [LIBWME\\_USER\\_ACTION](#)  
*enums for the registration confirmation*
- enum { ... } [P1609\\_CHANNEL\\_ACCESS](#)
- enum { ... } [P1609\\_WSATYPE](#)
- struct [vendor\\_frame](#) [\\_\\_attribute\\_\\_](#)

### 5.5.1 Detailed Description

The V2X Network and transport module defines the data structures and API's for

- WAVE WSA, WSMP & WME services (1609.3)
- ETSI GeoNetworking/BTP service.

### 5.5.2 Typedef Documentation

#### 5.5.2.1 typedef struct v2x\_wsmp\_callbacks v2x\_wsmp\_callbacks\_t

This structure contains a set of callbacks associated with the application about the indication of WSMPs/commands etc.

This is set to a list of function pointers, and they will be called from the library based on the communication protocol type between the caller and the 1609.3. The protocol sends a confirmation upon a call to `wme_register_user(provider)` request. the confirmation callback `wme_user(provider)_confirm` is called (if its a valid pointer) and the application can decide to transmit/receive WSMP.

The protocol sends a WSMP decoded packet and fills into the `savariwme_rx_indication` and callback `wme_wsm_indication` is called.

#### 5.5.2.2 typedef savari\_socket\_desc\_t v2x\_wsmp\_handler\_t

WSMP handler returned from the `v2x_wsmp_init`

#### 5.5.2.3 typedef struct v2x\_wsmp\_rcv\_indication v2x\_wsmp\_rcv\_indication\_t

This structure used to indicate received WSMP packets to an application.

### 5.5.3 Enumeration Type Documentation

#### 5.5.3.1 anonymous enum

Requested channel access type.

#### Enumerator

**LIBWME\_USER\_AUTOACCESS\_ONMATCH** Provide the service to the application when a matched service is advertised over WSA

**LIBWME\_USER\_AUTOACCESS\_UNCOND** Force 1609.3 protocol to switch between a given service channel and default control channel(178)

**LIBWME\_USER\_NOSCHACCESS** Wait for the WSA and match the given service irrespective of the service channel

### 5.5.3.2 anonymous enum

enums for the registration confirmation

These are used to whether join/delete/change a service

Enumerator

**LIBWME\_ACTION\_ADD** Add the current user/provider to the service table

**LIBWME\_ACTION\_DELETE** Remove the current user/provider from the service table

**LIBWME\_ACTION\_CHANGE** Change the current user/provider in the service table

### 5.5.3.3 anonymous enum

channel access types

Enumerator

**P1609\_CHANNEL\_ACCESS\_CONTINUOUS** Continuous channel access

**P1609\_CHANNEL\_ACCESS\_ALTERNATING** Alternating channel access

**P1609\_CHANNEL\_ACCESS\_ONMATCH** Onmatch channel access

**P1609\_CHANNEL\_ACCESS\_NOSCHACCESS** No SCH access

**P1609\_CHANNEL\_ACCESS\_UNKNOWN** Unavailable / not known

### 5.5.3.4 anonymous enum

WSA types

Enumerator

**P1609\_WSA\_UNSECURED** P1609\_WSA\_UNSECURED - unsecured WSA

**P1609\_WSA\_SECURED** P1609\_WSA\_SECURED - secured WSA

### 5.5.3.5 enum LIBWME\_RC\_RESULT\_CODE

Result codes when a confirmation comes from 1609.3, generally these codes will go into as arguments of wme\_provider\_confirm or wme\_user\_confirm callback functions

Enumerator

**LIBWME\_RC\_ACCEPTED** Registration successful

**LIBWME\_RC\_INVALID\_PARAMETERS** Invalid parameters in registration

**LIBWME\_RC\_UNSPECIFIED** Unknown/Unspecified error occurred

## 5.5.4 Function Documentation

### 5.5.4.1 v2x\_status\_t v2x\_wme\_open ( char \* wme\_ip, char \* iface, v2x\_wsmp\_handler\_t \* wme\_handler )

Open a connection with WME engine

Description

The wme\_init initialises and connects to the wme stack and returns a savari\_wme\_handler\_t handler. this handler is further used to transmit and receive messages to/from the daemon. the messages may consist of a set of confirmations, or WSM data.

**Parameters**

in	<i>wme_ip</i>	IP address of the WME engine to connect to, default is 127.0.0.1
in	<i>iface</i>	Interface to register for tx/rx of DSRC packets
out	<i>wme_handler</i>	handler of type <code>v2x_wsmp_handler_t</code> , filled by API

**Returns**

V2X\_SUCCESS on success or error number on failure

**Return values**

V2X_SUCCESS	API operation was successful
V2X_EACCES	error in creating the socket to WME engine or failure in other socket related operations
V2X_EINVAL	Invalid value (passed as argument or computed)
V2X_EFAULT	NULL pointer passed as argument

#### 5.5.4.2 `v2x_status_t v2x_wme_register_user ( v2x_wsmp_handler_t handler, v2x_wme_reg_req_t * wme_req )`

Register the application as user of WME service

**Description**

The `wme_register_user` registers the application in user mode to the stack for the purpose of sending /receiving WSMs or joining a service. the stack recognises the application using PSID's.

**Parameters**

in	<i>handler</i>	handler of type <code>v2x_wsmp_handler_t</code> , returned by <a href="#">v2x_wme_open()</a>
in	<i>wme_req</i>	The request structure, filled by the application

**Returns**

- returns V2X\_SUCCESS on success and error number on failure

**Return values**

V2X_SUCCESS	API operation was successful
V2X_EINVAL	Invalid argument passed to API
V2X_EIO	error in sending the request to WME engine

#### 5.5.4.3 `v2x_status_t v2x_wme_unregister_user ( v2x_wsmp_handler_t handler, v2x_wme_reg_req_t * wme_req )`

Unregister the user application from WME engine

**Description**

The `wme_unregister_user` unregisters the user application from the stack and stops receiving the WSMs or commands on behalf of the application.

**Parameters**

in	<i>handler</i>	handler of type <code>v2x_wsmp_handler_t</code> , filled by a call to <a href="#">v2x_wme_open()</a>
in	<i>wme_req</i>	The request structure

**Returns**

V2X\_SUCCESS on success or error number on failure

**Return values**

<i>V2X_SUCCESS</i>	API operation was successful
<i>V2X_EINVAL</i>	Invalid value passed as argument
<i>V2X_EFAULT</i>	Invalid address(NULL pointer) passed as argument

#### 5.5.4.4 `v2x_status_t v2x_wme_user_service_confirm ( v2x_wsmp_handler_t handler, int action, v2x_wme_reg_req_t * wme_req )`

confirm user application registration to 1609.3 stack

**Description**

This function confirms the registration of a user application with the WME engine

**Parameters**

in	<i>handler</i>	handler of type <code>v2x_wsmp_handler_t</code> . Filled by <a href="#">v2x_wme_open()</a>
in	<i>action</i>	Action to specify for the application. Valid values are SAVARI1609_ACTION_ADD, SAVARI1609_ACTION_DELETE, SAVARI1609_ACTION_CHANGE
in	<i>wme_req</i>	The request structure

**Returns**

V2X\_SUCCESS on success or error number on failure

**Return values**

<i>V2X_SUCCESS</i>	API operation was successful
<i>V2X_EINVAL</i>	Invalid argument passed
<i>V2X_EFAULT</i>	NULL pointer passed as argument

#### 5.5.4.5 `v2x_status_t v2x_wsmp_convert_psid_be ( uint32_t psid, uint32_t * psid_be )`

converts the psid to big-endian

**Description**

This function converts the PSID to big endian notation

**Parameters**

in	<i>psid</i>	PSID to convert to big-endian
out	<i>psid_be</i>	converted PSID

**Returns**

V2X\_SUCCESS on success and error number on failure

## Return values

<i>V2X_SUCCESS</i>	API operation was successful
<i>V2X_EFAULT</i>	NULL pointer passed as argument

**5.5.4.6** `v2x_status_t v2x_wsmp_rcv ( v2x_wsmp_handler_t handler, v2x_wsmp_callbacks_t * wme_cbs, void * ctx )`

Receive WSMP packet

## Description

This function should be called by the application whenever there is data available on handler filled by `v2x-_wme_open()`. Stack will then parse that data and calls applications callbacks registered using `v2x_wsmp_callbacks`. for a application provider confirmation calls `wme_provider_confirm` for a application user confirmation calls `wme_user_confirm` for a cch confirmation calls `wme_cch_confirm` for a WSM received. calls `wme_wsm_` indication for a wme cmd responses, calls `wme_cmd`

## Parameters

in	<i>handler</i>	handler of type <code>v2x_wsmp_handler_t</code>
in	<i>wsmp_cbs</i>	pointer to <code>v2x_wsmp_callbacks</code> , which contains the set of callbacks needed by the API
in	<i>ctx</i>	context passed from the application

## Returns

`V2X_SUCCESS` on success and error number on failure

## Return values

<i>V2X_SUCCESS</i>	The receive is successful
<i>V2X_PEER_CLOSE</i>	The peer connection closed by 1609.3 daemon
<i>V2X_EINVAL</i>	Invalid data received or handler is not setup
<i>V2X_EFAULT</i>	NULL <code>wme_callback</code> and (or) <code>ctx</code>

**5.5.4.7** `v2x_status_t v2x_wsmp_send ( v2x_wsmp_handler_t handler, v2x_wsmp_send_req_t * msg_param, uint8_t * msgbuf )`

Send WSMP with specified message and parameters

## Description

This function is used by a higher layer entity to request sending a WAVE short message.

## Parameters

in	<i>handler</i>	handler of type <code>v2x_wsmp_handler_t</code>
in	<i>msg_param</i>	The request structure, contains transmission parameters such as channel, datarate, txpower for the message to be transmitted.
in	<i>msgbuf</i>	Message to be transmitted

## Returns

`V2X_SUCCESS` on success or error number on failure

## Return values

<i>V2X_SUCCESS</i>	API operation was successful
<i>V2X_EINVAL</i>	Invalid argument passed
<i>V2X_EFAULT</i>	NULL pointer passed as argument
<i>V2X_ELENGTH</i>	Invalid transmit length passed
<i>V2X_ECHANNEL</i>	Invalid channel number passed

### 5.5.5 Variable Documentation

#### 5.5.5.1 enum { ... } LIBWME\_USER\_ACTION

enums for the registration confirmation

These are used to whether join/delete/change a service

#### 5.5.5.2 enum { ... } P1609\_CHANNEL\_ACCESS

channel access types

#### 5.5.5.3 enum { ... } P1609\_WSATYPE

WSA types

## 5.6 The V2X Security module API

### Data Structures

- struct [ae\\_ui\\_rx\\_callbacks](#)

### Typedefs

- typedef struct [aerolink\\_sign\\_secprofile](#) [v2x\\_sign\\_sec\\_profile\\_t](#)
- typedef struct [aerolink\\_verify\\_secprofile](#) [v2x\\_verify\\_sec\\_profile\\_t](#)
- typedef struct [security\\_cmd\\_profile\\_msg](#) [v2x\\_security\\_profile\\_t](#)
- typedef struct [ae\\_ui\\_rx\\_callbacks](#) [v2x\\_sec\\_rx\\_cblist\\_t](#)

### Enumerations

- enum [v2x\\_security\\_profile\\_type\\_t](#) { [SECURITY\\_SIGNATURE\\_PROFILE](#) = 0x01, [SECURITY\\_VERIFICATION\\_PROFILE](#) = 0x02 }

### Functions

- [v2x\\_status\\_t](#) [v2x\\_sec\\_init](#) ([v2x\\_security\\_profile\\_t](#) \*profile, int \*sock)
- void [v2x\\_sec\\_deinit](#) (int sock)
- [v2x\\_status\\_t](#) [v2x\\_sec\\_sign\\_request](#) (int sock, uint32\_t [psid](#), uint8\_t \*ssp, uint32\_t sspLength, int signer\_id, uint8\_t \*payload, int payload\_len, void \*app\_ref)
- [v2x\\_status\\_t](#) [v2x\\_sec\\_verify\\_request](#) (int sock, uint32\_t [psid](#), uint8\_t \*mac, int verify\_disable, uint8\_t \*payload, int payload\_len, void \*app\_ref)
- [v2x\\_status\\_t](#) [v2x\\_sec\\_rx](#) (int sock, [v2x\\_sec\\_rx\\_cblist\\_t](#) \*cblist, void \*ctx)
- [v2x\\_status\\_t](#) [v2x\\_sec\\_cert\\_change\\_register](#) (int sock, uint32\_t [psid](#), void \*app\_ref)
- [v2x\\_status\\_t](#) [v2x\\_sec\\_cert\\_change\\_unregister](#) (int sock)
- [v2x\\_status\\_t](#) [v2x\\_sec\\_idchange\\_lock](#) (int sock)
- [v2x\\_status\\_t](#) [v2x\\_sec\\_idchange\\_unlock](#) (int sock)

#### 5.6.1 Detailed Description

The V2X Security module defines the data structures and API's for WAVE security services (1609.2)

#### 5.6.2 Typedef Documentation

##### 5.6.2.1 typedef struct [ae\\_ui\\_rx\\_callbacks](#) [v2x\\_sec\\_rx\\_cblist\\_t](#)

[v2x\\_sec\\_rx\\_cblist\\_t](#) - callback list from the server socket

##### 5.6.2.2 typedef struct [security\\_cmd\\_profile\\_msg](#) [v2x\\_security\\_profile\\_t](#)

provide a security profile for the aerolink security engine

##### 5.6.2.3 typedef struct [aerolink\\_sign\\_secprofile](#) [v2x\\_sign\\_sec\\_profile\\_t](#)

signing security profile passed from the application

#### 5.6.2.4 typedef struct aerolink\_verify\_secprofile v2x\_verify\_sec\_profile\_t

•

verificaiton profile passed from the application

verification security profile passed from the application

### 5.6.3 Enumeration Type Documentation

#### 5.6.3.1 enum v2x\_security\_profile\_type\_t

v2x\_security\_profile\_type\_t - security profile type

this is set of bits to describe if the profile contains just the signing or verification or both

##### Enumerator

**SECURITY\_SIGNATURE\_PROFILE** SECURITY\_SIGNATURE\_PROFILE - signing profile

**SECURITY\_VERIFICATION\_PROFILE** SECURITY\_VERIFICATION\_PROFILE - verification profile

### 5.6.4 Function Documentation

#### 5.6.4.1 v2x\_status\_t v2x\_sec\_cert\_change\_register ( int sock, uint32\_t psid, void \* app\_ref )

register for the cert change

##### Description

This API registers for the certificate change. The API callback gets called and notifies the application when the certificate change has happened. Refer to the [v2x\\_security\\_base.h](#) for the notifications.

##### Parameters

in	<i>sock</i>	socket handler
in	<i>psid</i>	PSID of the message
in	<i>app_ref</i>	application reference

returns V2X\_SUCCESS on success.

##### Return values

<i>V2X_EIO</i>	on failed to send on the security socket.
----------------	---

#### 5.6.4.2 v2x\_status\_t v2x\_sec\_cert\_change\_unregister ( int sock )

unregister for the cert change

##### Description

This API unregisters from the certificate change.

##### Parameters



<i>in</i>	<i>sock</i>	socket handler
-----------	-------------	----------------

returns V2X\_SUCCESS on success.

**Return values**

<i>V2X_EIO</i>	on failed to send on the security socket.
----------------	---

#### 5.6.4.3 void v2x\_sec\_deinit ( int *sock* )

Deinitialize the connection to the security engine

##### Description

Deinitialize or close the connection with the security engine.

##### Parameters

<i>in</i>	<i>sock</i>	socket handler
-----------	-------------	----------------

#### 5.6.4.4 v2x\_status\_t v2x\_sec\_idchange\_lock ( int *sock* )

lock the cert change requests

##### Description

This API locks the cert change methods at the security engine. The use is that when there are event based BSM transmissions a certificate shall not be changed. Locking the certificate change would let the event based BSM transmissions unaffected without causing the delay.

##### Parameters

<i>in</i>	<i>sock</i>	socket handler
-----------	-------------	----------------

returns V2X\_SUCCESS on success.

**Return values**

<i>V2X_EIO</i>	on failed to send on the security socket.
----------------	---

#### 5.6.4.5 v2x\_status\_t v2x\_sec\_idchange\_unlock ( int *sock* )

unlock the cert change requests

##### Description

This API unlocks the cert change methods at the security engine. Thus re-enabling the certificate change.

##### Parameters

<i>in</i>	<i>sock</i>	socket handler
-----------	-------------	----------------

returns V2X\_SUCCESS on success.

**Return values**

<i>V2X_EIO</i>	on failed to send on the security socket.
----------------	---

#### 5.6.4.6 `v2x_status_t v2x_sec_init ( v2x_security_profile_t * profile, int * sock )`

Initialize the connection to security engine

##### Description

Create a connection with the security engine with the profile. Return the socket address int the sock argument.

##### Parameters

out	<i>sock</i>	socket handler
in	<i>profile</i>	profile for the application given to the security engine returns V2X_SUCCESS on success

##### Return values

<i>V2X_SUCCESS</i>	upon connection success and socket address is returned in the sock argument.
<i>V2X_EIO</i>	when socket failed to open or failed to connect with the security engine

#### 5.6.4.7 `v2x_status_t v2x_sec_rx ( int sock, v2x_sec_rx_cblist_t * cblist, void * ctx )`

perform a reception on the sign or verify responses

##### Description

This API allows to perform a reception of sign and verify requests performed from the caller. The caller registers a `v2x_sec_rx` on the socket for the reception.

##### Parameters

in	<i>sock</i>	socket handler
in	<i>cblist</i>	list of aerolink callbacks that gets called upon the response from the security engine
in	<i>ctx</i>	application callback data

returns V2X\_SUCCESS on success.

##### Return values

<i>V2X_EIO</i>	on failed to receive on the socket.
<i>V2X_EINVAL</i>	invalid argument.

#### 5.6.4.8 `v2x_status_t v2x_sec_sign_request ( int sock, uint32_t psid, uint8_t * ssp, uint32_t sspLength, int signer_id, uint8_t * payload, int payload_len, void * app_ref )`

send a sign message request to the security engine

##### Description

Send a sign message (WSMP/WSA/Any) request to the security engine.

**Parameters**

in	<i>sock</i>	socket handler
in	<i>psid</i>	PSID of the message
in	<i>ssp</i>	SSP of the message
in	<i>sspLength</i>	length of SSP
in	<i>signer_id</i>	one of the types of <i>v2x_sec_signer_id_t</i>
in	<i>payload</i>	payload of the message
in	<i>payload_len</i>	length of the payload
in	<i>app_ref</i>	application reference pointer

returns V2X\_SUCCESS on success.

**Return values**

<i>V2X_EIO</i>	on failed to send the sign request.
----------------	-------------------------------------

**5.6.4.9** *v2x\_status\_t* *v2x\_sec\_verify\_request* ( *int sock*, *uint32\_t psid*, *uint8\_t \* mac*, *int verify\_disable*, *uint8\_t \* payload*, *int payload\_len*, *void \* app\_ref* )

send a verify message (WSMP/WSA/Any) request to the security engine

**Description**

Send a verify message request to the security engine.

**Parameters**

in	<i>sock</i>	socket handler
in	<i>psid</i>	PSID of the message
in	<i>mac</i>	mac address of the receiver
in	<i>verify_disable</i>	flag to disable verification
in	<i>payload</i>	payload of the message
in	<i>payload_len</i>	length of the payload
in	<i>app_ref</i>	application reference pointer

returns V2X\_SUCCESS on success.

**Return values**

<i>V2X_EIO</i>	on failed to send the verify request.
----------------	---------------------------------------

## 5.7 The V2X Sensors API

### Files

- file [v2x\\_canctl.h](#)

### Data Structures

- struct [pf\\_gnss\\_data](#)

### Typedefs

- typedef struct [pf\\_gnss\\_data](#) [pf\\_gnss\\_data\\_t](#)
- typedef struct [gps\\_data\\_t](#) [v2x\\_gnss\\_handle\\_t](#)
- typedef [pf\\_gnss\\_data\\_t](#) [v2x\\_gnss\\_data\\_t](#)
- typedef void \* [v2x\\_pf\\_handle](#)

### Functions

- [v2x\\_status\\_t](#) [v2x\\_libpcan\\_net\\_rcv](#) (struct [v2x\\_canctl](#) \*ctl, int \*can\_id, int \*can\_dlc, uint8\_t \*data, int size)
- void [v2x\\_libpcan\\_net\\_deinit](#) (struct [v2x\\_canctl](#) \*ctl)
- [v2x\\_status\\_t](#) [v2x\\_can\\_open](#) (int \*can\_hdl, int flag)
- [v2x\\_status\\_t](#) [v2x\\_can\\_rcv](#) (int can\_hdl, struct [cand\\_rcv\\_buf](#) \*buf)
- [v2x\\_status\\_t](#) [v2x\\_can\\_close](#) (int can\_hdl)
- [v2x\\_status\\_t](#) [v2x\\_gnss\\_open](#) (int \*gps\_fd, void \*\*handle)
- [v2x\\_status\\_t](#) [v2x\\_gnss\\_read](#) ([pf\\_gnss\\_data\\_t](#) \*gnss\_data, void \*handle)
- [v2x\\_status\\_t](#) [v2x\\_gnss\\_close](#) (void \*handle)
- [v2x\\_status\\_t](#) [v2x\\_gnss\\_filters\\_init](#) ([pf\\_gnss\\_data\\_t](#) \*gnss\_data)
- [v2x\\_status\\_t](#) [v2x\\_gnss\\_filters\\_deinit](#) ([pf\\_gnss\\_data\\_t](#) \*gnss\_data)

#### 5.7.1 Detailed Description

The V2X sensor module defines data structures and API functions for applications to access sensor data from navigation and CAN services.

#### 5.7.2 Typedef Documentation

##### 5.7.2.1 typedef struct [pf\\_gnss\\_data](#) [pf\\_gnss\\_data\\_t](#)

This structure is used to get GNSS data from GNSS service. This structure contains all the navigational information including latitude, longitude, elevation, time and mode of fix, DOP etc.,.

##### 5.7.2.2 typedef [pf\\_gnss\\_data\\_t](#) [v2x\\_gnss\\_data\\_t](#)

This structure is used to get GNSS data from GNSS service. This structure contain all the navigational information including latitude, longitude, elevation, time and mode of fix etc.,.

##### 5.7.2.3 typedef struct [gps\\_data\\_t](#) [v2x\\_gnss\\_handle\\_t](#)

This structure is used to get GNSS data from GNSS Chip

#### 5.7.2.4 typedef void\* v2x\_pf\_handle

This pointer is used to get GNSS pointer used in GNSS service.

### 5.7.3 Function Documentation

#### 5.7.3.1 v2x\_status\_t v2x\_can\_close ( int *can\_hdl* )

Close connection to CAN service

##### Description

This API is used to close the connection to a CAN service

##### Parameters

in	<i>can_hdl</i>	socket descriptor which identifies an open connection with CAN service
----	----------------	--

##### Returns

V2X\_SUCCESS on success and error number on failure

##### Return values

V2X_SUCCESS	API operation was successful
V2X_ACCES	Error accessing CAN service

#### 5.7.3.2 v2x\_status\_t v2x\_can\_open ( int \* *can\_hdl*, int *flag* )

Opens a connection to can service

##### Description

This API opens a connection to CAN service and returns a handle which can be used further to communicate with the CAN service to access vehicle information.

##### Parameters

out	<i>can_hdl</i>	Points to descriptor value that identifies opened connection with the CAN service
out	<i>flag</i>	For future use. Set to 0.

##### Returns

V2X\_SUCCESS on success and error number on failure

##### Return values

V2X_SUCCESS	API operation was successful
V2X_EACCES	Error in opening CAN service
V2X_EIO	CAN service read/write error
V2x_EFAULT	NULL pointer passed as argument

#### 5.7.3.3 v2x\_status\_t v2x\_can\_recv ( int *can\_hdl*, struct cand\_recv\_buf \* *buf* )

Receives data from CAN service

**Description**

This API is used to recieve data from CAN service

**Parameters**

in	<i>can_hndl</i>	socket descriptor which identifies an open connection with CAN service
out	<i>buf</i>	buffer which is filled with CAN data by this API

**Returns**

V2X\_SUCCESS on success and error number on failure

**Return values**

V2X_SUCCESS	API operation was successful
V2X_EINVAL	Invalid argument passed
V2X_EIO	CAN service read/write error

**5.7.3.4 v2x\_status\_t v2x\_gnss\_close ( void \* handle )**

Close an earlier opened connection to GNSS service.

**Parameters**

in	<i>handle</i>	Handle to GNSS service as returned by v2x_gnss_open
----	---------------	---

**Returns**

V2X\_SUCCESS on success or error number on failure.

**Return values**

V2X_SUCCESS	API function was successful
V2X_EIO	Error in read or write operation.

**5.7.3.5 v2x\_status\_t v2x\_gnss\_filters\_deinit ( pf\_gnss\_data\_t \* gnss\_data )**

De-initialize various GNSS filters.

**Description**

Initialize GNSS filters as specified in v2x\_gnss\_data\_t.

**Parameters**

in	<i>gnss_data</i>	Specifies navigational filters to be applied.
----	------------------	---

**Returns**

V2X\_SUCCESS on success or an error number on failure.

**Return values**

V2X_SUCCESS	API function was successful
V2X_EINVAL	Invalid argument passed.
V2X_EFAULT	An invalid user space address was specified for an argument.

### 5.7.3.6 `v2x_status_t v2x_gnss_filters_init ( pf_gnss_data_t * gnss_data )`

Initialize various GNSS filters.

#### Description

Initialize GNSS filters as specified in `v2x_gnss_data_t`.

#### Parameters

in	<i>gnss_data</i>	Specifies navigational filters to be applied.
----	------------------	---

#### Returns

V2X\_SUCCESS on success or an error number on failure.

#### Return values

V2X_SUCCESS	API function was successful
V2X_EINVAL	Invalid argument passed.
V2X_EFAULT	An invalid user space address was specified for an argument.

### 5.7.3.7 `v2x_status_t v2x_gnss_open ( int * gps_fd, void ** handle )`

Open a connection to GPS service

#### Description

This function opens a connection to GNSS service and returns a connection handle. The handle can be used for communication with the GNSS service to get navigational information and other operations.

#### Parameters

out	<i>gps_fd</i>	Points to descriptor value that identifies opned connection to GNSS service.
out	<i>handle</i>	Allocated and initialized pointer to handle is returned in this parameter.

#### Returns

V2X\_SUCCESS on success or error number on failure.

#### Return values

V2X_SUCCESS	API function was successful
V2X_EACCES	error in opening gps device
V2X_EIO	gps device read/write error
V2X_ENOMEM	not enough memory

### 5.7.3.8 `v2x_status_t v2x_gnss_read ( pf_gnss_data_t * gnss_data, void * handle )`

Get current navigational data from GNSS service

#### Description

This function gets current navigational data from GNSS service and fills application provided structure of type `v2x_gnss_data_t`.



**Parameters**

out	<i>gnss_data</i>	Contains current navigational data on success.
in	<i>handle</i>	Handle to GNSS service as returned by <code>v2x_gnss_open</code>

**Returns**

V2X\_SUCCESS on success or error number on failure.

**Return values**

V2X_SUCCESS	API function was successful.
V2X_EIO	Error in read or write operation.
V2X_EFAULT	An invalid user space address was specified for an argument.

**5.7.3.9 void v2x\_libpcan\_net\_deinit ( struct v2x\_canctl \* ctl )**

close the CAN interface

**Parameters**

in	<i>ctl</i>	The ctl control structure
----	------------	---------------------------

**Description**

This API closes the connection with the CAN interface

**Returns**

void

**5.7.3.10 v2x\_status\_t v2x\_libpcan\_net\_rcv ( struct v2x\_canctl \* ctl, int \* can\_id, int \* can\_dlc, uint8\_t \* data, int size )**

receive from the CAN interface

**Parameters**

in	<i>ctl</i>	The ctl control structure
in	<i>size</i>	The length of the data variable
out	<i>can_id</i>	The ID of the received CAN frame
out	<i>can_dlc</i>	The data length of the frame
out	<i>data</i>	The data of the CAN frame

**Description**

This API is used to receive the CAN frames from the CAN hardware

**Returns**

V2X\_SUCCESS on success and error on failure

**Return values**

<i>V2X_SUCCESS</i>	API is successful
<i>V2X_EFAULT</i>	NULL ctl / can_id / can_dlc / data passed
<i>V2X_EIO</i>	read / write error
<i>V2X_EINVAL</i>	length of the data received is less than the size of data variable

## Chapter 6

# Data Structure Documentation

### 6.1 `_J2735_2016_EnableLaneList_t` Struct Reference

```
#include <v2x_msg_spat.h>
```

#### Data Fields

- long `lane_id`

#### 6.1.1 Detailed Description

`J2735_2016_EnableLaneList_t` - Enabled Lanes that are advertised in SPaT

#### 6.1.2 Field Documentation

##### 6.1.2.1 long `lane_id`

`lane_id` - lane ID

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

- `v2x_msg_spat.h`

### 6.2 `_J2735_2016_ManeuverAssistList_t` Struct Reference

```
#include <v2x_msg_spat.h>
```

#### Data Fields

- long `connection_id`
- long `ped_bicycle_detect`

#### 6.2.1 Detailed Description

`J2735_2016_ManeuverAssistList_t` - Contains information about the the dynamic flow of traffic for the lane(s) and maneuvers in question (as determined by the LaneConnectionID). Note that this information can be sent regarding

any lane-to-lane movement; it need not be limited to the lanes with active (non-red) phases when sent. XXX: NOTE FOR NYC DEMO WE ARE USING SIGNAL GROUP IDs FOR LaneConnectionID

## 6.2.2 Field Documentation

### 6.2.2.1 long connection\_id

connection\_id - connection identifier indexed from the MAP

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

- [v2x\\_msg\\_spat.h](#)

## 6.3 \_J2735\_2016\_MovementEventList\_t Struct Reference

```
#include <v2x_msg_spat.h>
```

### Data Fields

- [J2735\\_2016\\_MovementPhaseState\\_t phase\\_state](#)
- [J2735\\_2016\\_TimeChange\\_t time\\_change\\_info](#)

### 6.3.1 Detailed Description

J2735\_2016\_MovementEventList\_t - a movement event describing the phase and the time information of the phase or the future phase

### 6.3.2 Field Documentation

#### 6.3.2.1 J2735\_2016\_MovementPhaseState\_t phase\_state

phase\_state - phase state of the movement

#### 6.3.2.2 J2735\_2016\_TimeChange\_t time\_change\_info

time\_change\_info - timing information of the phase

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

- [v2x\\_msg\\_spat.h](#)

## 6.4 \_J2735\_2016\_MovementState\_t Struct Reference

```
#include <v2x_msg_spat.h>
```

### Data Fields

- long [signal\\_group\\_id](#)
- int [num\\_movement\\_event](#)
- [J2735\\_2016\\_MovementEventList\\_t](#) \* [movement\\_event](#)

### 6.4.1 Detailed Description

J2735\_2016\_MovementState\_t - movement state enclosure describing the signal group id and a set of movement events.

### 6.4.2 Field Documentation

#### 6.4.2.1 J2735\_2016\_MovementEventList\_t\* movement\_event

movement\_event - movement event list

#### 6.4.2.2 int num\_movement\_event

num\_movement\_event - number of movement events (1 to 16)

#### 6.4.2.3 long signal\_group\_id

signal\_group\_id - signal group id of the movement. Obtained from MAP

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

- [v2x\\_msg\\_spat.h](#)

## 6.5 \_J2735\_2016\_PathPrediction Struct Reference

```
#include <v2x_msg_common.h>
```

### Data Fields

- double [radius\\_of\\_curvature](#)
- int [confidence](#)

### 6.5.1 Detailed Description

path prediction information

### 6.5.2 Field Documentation

#### 6.5.2.1 int confidence

confidence - confidence value of the radius

#### 6.5.2.2 double radius\_of\_curvature

radius\_of\_curvature - radius of curvature

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

- [v2x\\_msg\\_common.h](#)

## 6.6 \_J2735\_2016\_PHObject Struct Reference

```
#include <v2x_msg_pvd.h>
```

### Data Fields

- int [valid\\_fps](#)
- [J2735\\_2016\\_FullPosVec\\_t](#) [fps](#)
- [J2735\\_2016\\_GPSSStatus\\_t](#) [gps\\_status](#)
- int [ph\\_points\\_len](#)
- [J2735\\_2016\\_PathHistory\\_t](#) [ph\\_points](#) [J2735\_2016\_PH\_PT\_MAX]

### 6.6.1 Detailed Description

J2735 PH object

### 6.6.2 Field Documentation

#### 6.6.2.1 J2735\_2016\_FullPosVec\_t [fps](#)

full position vector ...

#### 6.6.2.2 J2735\_2016\_GPSSStatus\_t [gps\\_status](#)

GPS status

#### 6.6.2.3 J2735\_2016\_PathHistory\_t [ph\\_points](#)[J2735\_2016\_PH\_PT\_MAX]

Ph point sets

#### 6.6.2.4 int [ph\\_points\\_len](#)

Ph points length - min 1 and max 23

#### 6.6.2.5 int [valid\\_fps](#)

set this to 1 if fps is being filled with valid data

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

- [v2x\\_msg\\_pvd.h](#)

## 6.7 \_J2735\_2016\_ProbeSnapshot Struct Reference

```
#include <v2x_msg_pvd.h>
```

## Data Fields

- [J2735\\_2016\\_FullPosVec\\_t](#) fps
- int [valid\\_veh\\_safety\\_ext](#)
- [J2735\\_2016\\_VehSafetyExtensions\\_t](#) veh\_safety\_ext
- int [valid\\_veh\\_status](#)
- [J2735\\_2016\\_VehicleStatus\\_t](#) vehicle\_status

### 6.7.1 Detailed Description

PVD snapshot object ..

### 6.7.2 Field Documentation

#### 6.7.2.1 J2735\_2016\_FullPosVec\_t fps

Full position vector of the snapshot when it is generated

#### 6.7.2.2 int valid\_veh\_safety\_ext

set this to 1 if the vehicle safety extensions are being filled

#### 6.7.2.3 int valid\_veh\_status

set this to 1 if the vehicle status is being filled

#### 6.7.2.4 J2735\_2016\_VehSafetyExtensions\_t veh\_safety\_ext

vehicle safety extensions

#### 6.7.2.5 J2735\_2016\_VehicleStatus\_t vehicle\_status

vehicle status

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

- [v2x\\_msg\\_pvd.h](#)

## 6.8 \_J2735\_2016\_PVD Struct Reference

```
#include <v2x_msg_pvd.h>
```

## Data Fields

- int [min\\_of\\_year](#)
- int [probe\\_seg\\_num](#)
- int [vid\\_info\\_present](#)
- [J2735\\_2016\\_VehicleIdent\\_t](#) vid\_info
- int [valid\\_fullpos\\_vec](#)
- [J2735\\_2016\\_FullPosVec\\_t](#) full\_posvec

- [J2735\\_2016\\_VehicleClassification\\_t veh\\_class](#)
- int [snapshot\\_len](#)
- [J2735\\_2016\\_ProbeSnapshot\\_t snapshot](#) [J2735\_2016\_PVD\_SNAPSHOT\_MAX]

### 6.8.1 Detailed Description

PVD message

Contains a set of snapshots and a PSN related to the snapshots.

### 6.8.2 Field Documentation

#### 6.8.2.1 J2735\_2016\_FullPosVec\_t full\_posvec

full position vector

#### 6.8.2.2 int min\_of\_year

minute of the year

#### 6.8.2.3 int probe\_seg\_num

probe segment number .. PSN

#### 6.8.2.4 J2735\_2016\_ProbeSnapshot\_t snapshot[J2735\_2016\_PVD\_SNAPSHOT\_MAX]

probe snapshots

#### 6.8.2.5 int snapshot\_len

snapshot length (1 - 32)

#### 6.8.2.6 int valid\_fullpos\_vec

set this to 1, if full position vector is being filled

#### 6.8.2.7 J2735\_2016\_VehicleClassification\_t veh\_class

vehicle class

#### 6.8.2.8 J2735\_2016\_VehicleIdent\_t vid\_info

vehicle identification information

#### 6.8.2.9 int vid\_info\_present

set this to 1, if Vehicle identification data is available

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

- [v2x\\_msg\\_pvd.h](#)



## 6.9 \_J2735\_2016\_TimeChange\_t Struct Reference

```
#include <v2x_msg_spat.h>
```

### Data Fields

- long [start\\_time](#)
- long [min\\_end\\_time](#)
- long [max\\_end\\_time](#)
- long [likely\\_time](#)
- long [confidence](#)
- long [next\\_time](#)

#### 6.9.1 Detailed Description

J2735\_2016\_TimeChange\_t - Timechange describing each phase time active and remaining values. Likely change is reported in the likely\_time. The confidence value indicates the quality of the likely\_time.

#### 6.9.2 Field Documentation

##### 6.9.2.1 long confidence

confidence - confidence value of the likely\_time

##### 6.9.2.2 long likely\_time

likely\_time - likely time of the phase

##### 6.9.2.3 long max\_end\_time

max\_end\_time - maximum guaranteed phase timing

##### 6.9.2.4 long min\_end\_time

min\_end\_time - minimum guaranteed phase timing

##### 6.9.2.5 long next\_time

next\_time - the future timing information

##### 6.9.2.6 long start\_time

start\_time - start of the phase

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

- [v2x\\_msg\\_spat.h](#)

## 6.10 \_J2735\_2016\_VehicleMeasurements Struct Reference

### Data Fields

- double **vehicle\_height**
- double **bumper\_height\_front**
- double **bumper\_height\_rear**
- double **vehicle\_mass**
- double **trailer\_weight**
- J2735\_2016\_VehicleType\_t **vehicle\_type**

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

- [v2x\\_msg\\_common.h](#)

## 6.11 \_J2735\_2016\_VehicleStatus Struct Reference

```
#include <v2x_msg_pvd.h>
```

### Data Fields

- [J2735\\_2016\\_ExteriorLights\\_t](#) **exterior\_lights**
- [J2735\\_2016\\_LightBar\\_t](#) **lightbar\_in\_use**
- int **valid\_wiper\_set**
- [J2735\\_2016\\_WiperSet\\_t](#) **wiper\_set**
- int **valid\_vehicle\_data**
- [J2735\\_2016\\_VehicleMeasurements\\_t](#) **vehicle\_data**

### 6.11.1 Detailed Description

Vehicle status

### 6.11.2 Field Documentation

#### 6.11.2.1 J2735\_2016\_ExteriorLights\_t exterior\_lights

Exterior lights

#### 6.11.2.2 J2735\_2016\_LightBar\_t lightbar\_in\_use

Light bar in use

#### 6.11.2.3 int valid\_vehicle\_data

set this to 1, if the vehicle data is filled

#### 6.11.2.4 int valid\_wiper\_set

set this to 1 if wipers are being filled

## 6.11.2.5 J2735\_2016\_VehicleMeasurements\_t vehicle\_data

vehicle data

## 6.11.2.6 J2735\_2016\_WiperSet\_t wiper\_set

wiper set status

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

- [v2x\\_msg\\_pvd.h](#)

## 6.12 \_J2735\_2016\_VehSafetyExtensions Struct Reference

```
#include <v2x_msg_pvd.h>
```

### Data Fields

- [J2735\\_2016\\_VehicleEventFlags\\_t event\\_flags](#)
- int [valid\\_ph](#)
- [J2735\\_2016\\_PHObject\\_t ph](#)
- int [valid\\_pp](#)
- [J2735\\_2016\\_PathPrediction\\_t pp](#)
- [J2735\\_2016\\_ExteriorLights\\_t exterior\\_lights](#)

### 6.12.1 Detailed Description

Vehicle safety extensions

### 6.12.2 Field Documentation

## 6.12.2.1 J2735\_2016\_VehicleEventFlags\_t event\_flags

event flags

## 6.12.2.2 J2735\_2016\_ExteriorLights\_t exterior\_lights

exterior lights

## 6.12.2.3 J2735\_2016\_PHObject\_t ph

PH data (calculated using methods 1, 2 or 3)

## 6.12.2.4 J2735\_2016\_PathPrediction\_t pp

PP data (calculated using PP methods)

## 6.12.2.5 int valid\_ph

set this to 1 if the PH data is being filled

#### 6.12.2.6 int valid\_pp

set this to 1 if the PP data is being filled

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

- [v2x\\_msg\\_pvd.h](#)

### 6.13 \_J2735\_2016\_Wiperset Struct Reference

```
#include <v2x_msg_common.h>
```

#### Data Fields

- [J2735\\_2016\\_WiperStatus\\_t wiper\\_front](#)
- int [wiper\\_front\\_rate](#)
- [J2735\\_2016\\_WiperStatus\\_t wiper\\_rear](#)
- int [wiper\\_rear\\_rate](#)

#### 6.13.1 Detailed Description

wiper set - describing the fron, rear wipers and the rates

#### 6.13.2 Field Documentation

##### 6.13.2.1 J2735\_2016\_WiperStatus\_t wiper\_front

wiper\_front - front wiper status

##### 6.13.2.2 int wiper\_front\_rate

wiper\_front\_rate - front wiper rate

##### 6.13.2.3 J2735\_2016\_WiperStatus\_t wiper\_rear

wiper\_rear - rear wiper status

##### 6.13.2.4 int wiper\_rear\_rate

wiper\_rear\_rate - rear wiper rate

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

- [v2x\\_msg\\_common.h](#)

### 6.14 ae\_ui\_rx\_callbacks Struct Reference

```
#include <v2x_security_base.h>
```

## Data Fields

- void(\* [ae\\_sign\\_callback](#))(int result, uint8\_t \*buf, int buf\_len, long app\_ref, void \*usr\_ctx)
- void(\* [ae\\_verify\\_callback](#))(int result, uint8\_t \*buf, int buf\_len, long app\_ref, void \*usr\_ctx)

### 6.14.1 Detailed Description

v2x\_sec\_rx\_cblist\_t - callback list from the server socket

### 6.14.2 Field Documentation

#### 6.14.2.1 void(\* ae\_sign\_callback)(int result, uint8\_t \*buf, int buf\_len, long app\_ref, void \*usr\_ctx)

ae\_sign\_callback - sign callback gets called from the security engine

#### Description

Sign callback from the security engine. The result of the message is stored in the result.

The signed packet is copied into the buf and the length of the message in the buf\_len. The usr\_ctx and app\_ref pointers are returned back to the user

#### Parameters

out	<i>result</i>	result of the security signing
out	<i>buf</i>	signed buffer from the security signing
out	<i>buf_len</i>	length of the signed buffer
out	<i>app_ref</i>	application reference data
out	<i>usr_ctx</i>	user context

#### 6.14.2.2 void(\* ae\_verify\_callback)(int result, uint8\_t \*buf, int buf\_len, long app\_ref, void \*usr\_ctx)

ae\_verify\_callback - verify callback gets called from the security engine

#### Description

Verify callback from the security engine. The result of the message is stored in the result.

The verify packet is copied into the buf and the length of the message in the buf\_len. The verified packet becomes unsecured and provided back to the caller in the buf. The usr\_ctx and app\_ref pointers are returned back to the user

#### Parameters

out	<i>result</i>	result of the security verification
out	<i>buf</i>	verified unsecured buffer from the security verification
out	<i>buf_len</i>	length of the verified unsecured buffer
out	<i>app_ref</i>	application reference data
out	<i>usr_ctx</i>	user context

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

- [v2x\\_security\\_base.h](#)

## 6.15 aerolink\_sign\_secprofile Struct Reference

### Data Fields

- int [cert\\_attach\\_rate](#)
- int [use\\_generation\\_time](#)
- int [use\\_expiry\\_time](#)
- int [use\\_generation\\_loc](#)

### 6.15.1 Field Documentation

#### 6.15.1.1 int cert\_attach\_rate

cert\_attach\_rate - certificate attach rate for the message

1000 msec for BSM for example

#### 6.15.1.2 int use\_expiry\_time

use\_expiry\_time - set expiry time in security headers

#### 6.15.1.3 int use\_generation\_loc

use\_generation\_loc - set generation location in security headers

#### 6.15.1.4 int use\_generation\_time

use\_generation\_time - set generation time in security headers

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

- [v2x\\_security\\_base.h](#)

## 6.16 aerolink\_verify\_secprofile Struct Reference

### Data Fields

- int [verification\\_interval](#)
- int [check\\_replay](#)
- int [check\\_generation\\_time](#)
- int [check\\_expiry\\_time](#)
- int [check\\_generation\\_location](#)

### 6.16.1 Field Documentation

#### 6.16.1.1 int check\_expiry\_time

check\_expiry\_time - check for the expiry time in security headers 1 to enable, 0 to disable

#### 6.16.1.2 int check\_generation\_location

check\_generation\_location - check for the generation location in security headers 1 to enable, 0 to disable

#### 6.16.1.3 int check\_generation\_time

check\_generation\_time - check for the generation time in security headers 1 to enable, 0 to disable

#### 6.16.1.4 int check\_replay

check\_replay - check for the replay 1 to enable, 0 to disable

#### 6.16.1.5 int verification\_interval

verification\_interval - verification interval

for ex: 1000 msec for BSM

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

- [v2x\\_security\\_base.h](#)

## 6.17 connects\_to Struct Reference

```
#include <v2x_msg_common.h>
```

### Data Fields

- int [id](#)
- int [signal\\_group\\_id](#)

### 6.17.1 Detailed Description

connects\_to list describing the lane connection with lane id and the signal group ID

### 6.17.2 Field Documentation

#### 6.17.2.1 int id

id - lane id

#### 6.17.2.2 int signal\_group\_id

signal\_group\_id - signal group ID

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

- [v2x\\_msg\\_common.h](#)

## 6.18 datetime Struct Reference

### Data Fields

- int **year**
- int **month**

- int **date**
- int **hour**
- int **min**
- int **sec**

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

- [v2x\\_msg\\_common.h](#)

## 6.19 encbrakestatus Struct Reference

### Data Fields

- uint16\_t **auxbrakes**:2
- uint16\_t **brakeboost**:2
- uint16\_t **scs**:2
- uint16\_t **abs**:2
- uint16\_t **traction**:2
- uint16\_t **sparebit**:1
- uint16\_t **wheelbrakeunavailable**:1
- uint16\_t **wheelbrakes**:4

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

- [v2x\\_msg\\_common.h](#)

## 6.20 encpadbytes Struct Reference

```
#include <v2x_msg_bsm.h>
```

### Data Fields

- uint16\_t **reserved**:6
- uint16\_t **pad**:10

### 6.20.1 Detailed Description

unused and internals to the library.. please do not use

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

- [v2x\\_msg\\_bsm.h](#)

## 6.21 endpointsetstype04 Struct Reference

### Data Fields

- long long **time\_offset**:16
- long long **elev\_offset**:12
- long long **long\_offset**:18



- long long **lat\_offset**:18

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

- [v2x\\_msg\\_common.h](#)

## 6.22 encprndlspeed Struct Reference

### Data Fields

- uint16\_t **speed**:13
- uint16\_t **prndl**:3

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

- [v2x\\_msg\\_common.h](#)

## 6.23 enctimestamp Struct Reference

```
#include <v2x_msg_bsm.h>
```

### Data Fields

- unsigned int **reserved**:8
- unsigned int **unused**:3
- unsigned int **microsec**:20
- unsigned int **second**:1

### 6.23.1 Detailed Description

unused and internals to the library.. please do not use

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

- [v2x\\_msg\\_bsm.h](#)

## 6.24 encvehiclesize Struct Reference

### Data Fields

- unsigned int **reserved**:8
- unsigned int **length**:14
- unsigned int **width**:10

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

- [v2x\\_msg\\_common.h](#)

## 6.25 etsi\_cam\_dgc Struct Reference

### Data Fields

- ETSIDangerousGoodsType\_t **dangerous\_goods\_type**

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

- [v2x\\_msg\\_cam.h](#)

## 6.26 etsi\_cam\_ec Struct Reference

### Data Fields

- ETSILBSirenUse\_t **lb\_siren\_in\_use**
- ETSICauseCode\_t **causecode**
- ETSIEmergencyPriority\_t **emergency\_priority**

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

- [v2x\\_msg\\_cam.h](#)

## 6.27 etsi\_cam\_ptc Struct Reference

### Data Fields

- int **embarkation\_status**
- ETSIPtActivationType\_t **pt\_activation\_type**
- int **pt\_activation\_data\_len**
- uint8\_t **pt\_activation\_data** [ETSI\_CAM\_PT\_ACTIVATION\_DATA\_LEN]

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

- [v2x\\_msg\\_cam.h](#)

## 6.28 etsi\_cam\_rc Struct Reference

### Data Fields

- ETSILBSirenUse\_t **lb\_siren\_in\_use**

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

- [v2x\\_msg\\_cam.h](#)

## 6.29 etsi\_cam\_rwc\_basic Struct Reference

### Data Fields

- int **sub\_cause\_code**
- ETSLBSirenUse\_t **lb\_siren\_in\_use**
- int **closed\_lanes\_available**
- ETSClosedLanes\_t **closed\_lanes**

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

- [v2x\\_msg\\_cam.h](#)

## 6.30 etsi\_cam\_scc Struct Reference

### Data Fields

- ETSLBSirenUse\_t **lb\_siren\_in\_use**
- ETSCauseCode\_t **causecode**
- ETSTrafficRule\_t **trafficrule**
- double **speedlimit**

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

- [v2x\\_msg\\_cam.h](#)

## 6.31 etsi\_cam\_stc Struct Reference

### Data Fields

- ETSTcType\_t **stc\_type**
- ETSLBSirenUse\_t **lb\_siren\_in\_use**

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

- [v2x\\_msg\\_cam.h](#)

## 6.32 etsi\_connects Struct Reference

### Data Fields

- int [lane\\_id](#)
- ETSAIAllowedManeuvers\_t [maneuvers](#)
- int [signal\\_group\\_id](#)

### 6.32.1 Field Documentation

#### 6.32.1.1 int lane\_id

Lane Id to which the current lane connects to

#### 6.32.1.2 ETSIAllowedManeuvers\_t maneuvers

Allowed maneuvers of the connecting lane

#### 6.32.1.3 int signal\_group\_id

Signal Group of the connecting lane

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

- [v2x\\_msg\\_etsi\\_map.h](#)

### 6.33 etsi\_node Struct Reference

#### Data Fields

- [ETSIPos3d\\_t node\\_point](#)
- [ETSINodeAttr\\_t node\\_attr](#)

#### 6.33.1 Field Documentation

##### 6.33.1.1 ETSINodeAttr\_t node\_attr

Node attributes at the waypoint

##### 6.33.1.2 ETSIPos3d\_t node\_point

Position 3D of the waypoint

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

- [v2x\\_msg\\_etsi\\_map.h](#)

### 6.34 etsi\_node\_attr Struct Reference

#### Data Fields

- double **d\_width**
- double **d\_elev**

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

- [v2x\\_msg\\_etsi\\_map.h](#)

### 6.35 etsi\_pos3d Struct Reference

#### Data Fields

- double **latitude**
- double **longitude**

- double **elevation**

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

- [v2x\\_msg\\_etsi\\_map.h](#)

## 6.36 ETSIItineraryPath\_t Struct Reference

### Data Fields

- int **len**
- ETSIPosition\_t **pos** [ETSI\_ITINERARY\_PATH\_MAX]

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

- [v2x\\_msg\\_denm.h](#)

## 6.37 ETSIRefDENMs\_t Struct Reference

### Data Fields

- int **ref\_denms\_len**
- ETSIActionID\_t **ref\_denm** [ETSI\_REF\_DENMS\_MAX]

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

- [v2x\\_msg\\_denm.h](#)

## 6.38 ETSIRestrictedTypes\_t Struct Reference

### Data Fields

- int **station\_type\_len**
- int **station\_type** [ETSI\_DENM\_REST\_TYPE\_MAX]

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

- [v2x\\_msg\\_denm.h](#)

## 6.39 intersec\_access\_pt Struct Reference

### Data Fields

- int **lane\_id**
- int **approach\_id**
- int **lane\_conn\_id**

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

- [v2x\\_msg\\_common.h](#)

## 6.40 intersection\_data Struct Reference

```
#include <v2x_msg_spat.h>
```

### Data Fields

- long [intersection\\_id](#)
- long [msgCount](#)
- [J2735\\_2016\\_IntersectionStatus\\_t](#) [intersection\\_status](#)
- int [min\\_of\\_the\\_year](#)
- [uint64\\_t](#) [timestamp\\_ms](#)
- int [n\\_lane\\_list](#)
- int [num\\_movement\\_states](#)
- int [num\\_maneuver\\_assist](#)
- [J2735\\_2016\\_EnableLaneList\\_t](#) \* [enabled\\_lanes](#)
- [J2735\\_2016\\_MovementState\\_t](#) \* [movement\\_states](#)
- [J2735\\_2016\\_ManeuverAssistList\\_t](#) \* [maneuver\\_assist](#)

### 6.40.1 Detailed Description

[J2735\\_2016\\_Intersection\\_t](#) - Intersection information

### 6.40.2 Field Documentation

#### 6.40.2.1 [J2735\\_2016\\_EnableLaneList\\_t](#)\* [enabled\\_lanes](#)

[enabled\\_lanes](#) - enabled lanes. If set they will be transmitted over the air. (0 - 16)

#### 6.40.2.2 long [intersection\\_id](#)

[intersection\\_id](#) - intersection id

#### 6.40.2.3 [J2735\\_2016\\_IntersectionStatus\\_t](#) [intersection\\_status](#)

[intersection\\_status](#) - status of the intersection

#### 6.40.2.4 [J2735\\_2016\\_ManeuverAssistList\\_t](#)\* [maneuver\\_assist](#)

[maneuver\\_assist](#) - maneuver assist list

#### 6.40.2.5 int [min\\_of\\_the\\_year](#)

[min\\_of\\_the\\_year](#) - minute of the year

#### 6.40.2.6 [J2735\\_2016\\_MovementState\\_t](#)\* [movement\\_states](#)

[movement\\_states](#) - set of movements describing the intersection

## 6.40.2.7 long msgCount

msgCount - message count of the SPaT. Increment only when there is an information change in the message (0 - 127)

## 6.40.2.8 int n\_lane\_list

n\_lane\_list - number of lanes if needed to send enabled\_lanes

## 6.40.2.9 int num\_maneuver\_assist

num\_maneuver\_assist - number of maneuver assist frames

## 6.40.2.10 int num\_movement\_states

num\_movement\_states - number of movement states

## 6.40.2.11 uint64\_t timestamp\_ms

timestamp\_ms - msec timestamp of the message

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

- [v2x\\_msg\\_spat.h](#)

## 6.41 intersection\_geodescr Struct Reference

```
#include <v2x_msg_map.h>
```

### Data Fields

- uint32\_t [intersection\\_id](#)
- int [msg\\_revision](#)
- [J2735\\_2016\\_Pos3d\\_2\\_t](#) ref\_pt
- double [lanewidth](#)
- int [n\\_lanelist](#)
- [J2735\\_2016\\_LaneList\\_t](#) \* [lanelist](#)

### 6.41.1 Detailed Description

[J2735\\_2016\\_IsecGeoDescr\\_t](#) - intersection geometric description the intersection description consists of a reference point, common lanewidth and a set of lanes describing the intersection.

### 6.41.2 Field Documentation

## 6.41.2.1 uint32\_t intersection\_id

intersection\_id - intersection ID of this intersection

#### 6.41.2.2 J2735\_2016\_LaneList\_t\* lanelist

lanelist - lane list

#### 6.41.2.3 double lanewidth

lanewidth - lane width of all the lanes. The different lanewidths are represented using the dElev and dWidth of the nodeList

#### 6.41.2.4 int msg\_revision

msg\_revision - message revision

#### 6.41.2.5 int n\_lanelist

n\_lanelist - number of lanes (1 - 64)

#### 6.41.2.6 J2735\_2016\_Pos3d\_2\_t ref\_pt

reference\_point - intersection reference point.

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

- [v2x\\_msg\\_map.h](#)

## 6.42 J2735\_2016\_Computed\_Lanes Struct Reference

```
#include <v2x_msg_tim.h>
```

### 6.42.1 Detailed Description

J2735\_2016\_Computed\_Lanes\_t - not supported

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

- [v2x\\_msg\\_tim.h](#)

## 6.43 j2735\_2016\_fullposvec Struct Reference

### Data Fields

- [J2735Position3D\\_t](#) pos3d
- double heading
- double prndl
- double speed
- struct [utcTimeFrame](#) \* utctime
- [J2735\\_2016\\_PosConf\\_t](#) posconf

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

- [v2x\\_msg\\_common.h](#)



## 6.44 J2735\_2016\_GeometricProjection Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- [J2735\\_2016\\_HeadingSlice\\_t heading\\_slice](#)
- [J2735Extent\\_t extent](#)
- double [lane\\_width](#)
- [J2735CircularRegion\\_t circle](#)

### 6.44.1 Detailed Description

J2735\_2016\_GeometricProjection\_t - geometric projection structure The geometric projection is used to define simple geometric shapes.

### 6.44.2 Field Documentation

#### 6.44.2.1 J2735CircularRegion\_t circle

circle - circular geometric shape

#### 6.44.2.2 J2735Extent\_t extent

extent - the range to which the message is valid

#### 6.44.2.3 J2735\_2016\_HeadingSlice\_t heading\_slice

heading\_slice - direction that is valid to this message

#### 6.44.2.4 double lane\_width

lane\_width - width of the lane if any that is useful for the geometry shape

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

- [v2x\\_msg\\_tim.h](#)

## 6.45 J2735\_2016\_LatLong Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- [J2735Position3D\\_t latlong\\_offsets](#) [J2735\_2016\_NODE\_OFF\_MAX]

### 6.45.1 Detailed Description

J2735\_2016\_LatLong\_t - set of lat and long offsets describing the LL offsets

## 6.45.2 Field Documentation

### 6.45.2.1 J2735Position3D\_t latlong\_offsets[J2735\_2016\_NODE\_OFF\_MAX]

J2735Position3D\_t - set of position3d points representing the offsets in lat and long

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

- [v2x\\_msg\\_tim.h](#)

## 6.46 J2735\_2016\_NodeSet\_XY Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- [int n\\_node\\_offsets](#)
- [J2735Position3D\\_t node\\_offsets](#) [J2735\_2016\_NODE\_OFF\_MAX]

### 6.46.1 Detailed Description

J2735\_2016\_NodeSet\_XY\_t - NodeSet list used to describe a set of nodes

## 6.46.2 Field Documentation

### 6.46.2.1 int n\_node\_offsets

n\_node\_offsets - number of node offsets (2 to 64)

### 6.46.2.2 J2735Position3D\_t node\_offsets[J2735\_2016\_NODE\_OFF\_MAX]

node\_offsets - number of node offsets described using lat and long

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

- [v2x\\_msg\\_tim.h](#)

## 6.47 J2735\_2016\_NodeXY Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- [J2735\\_2016\\_NodeXY\\_type\\_t node\\_xy\\_type](#)
- [J2735\\_2016\\_NodeSet\\_XY\\_t nodeset](#)
- [J2735\\_2016\\_Computed\\_Lanes\\_t computed\\_lanes](#)

### 6.47.1 Detailed Description

J2735\_2016\_NodeXY\_t - Node XY data structure enclosing the nodes

## 6.47.2 Field Documentation

### 6.47.2.1 J2735\_2016\_Computed\_Lanes\_t computed\_lanes

computed\_lanes - set of computed lanes

### 6.47.2.2 J2735\_2016\_NodeXY\_type\_t node\_xy\_type

node\_xy\_type - one of XY or LL

### 6.47.2.3 J2735\_2016\_NodeSet\_XY\_t nodeset

nodeset - set of nodes describing the region

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

- [v2x\\_msg\\_tim.h](#)

## 6.48 J2735\_2016\_OffsetSystem Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- int [zoom\\_level](#)
- [J2735\\_2016\\_OffsetSystemType\\_t](#) off\_sys\_type
- [J2735\\_2016\\_NodeXY\\_t](#) node\_set\_xy
- [J2735\\_2016\\_LatLong\\_t](#) node\_set\_ll

### 6.48.1 Detailed Description

J2735\_2016\_OffsetSystem\_t - offset system describing the nodes in detail either in xy or ll

## 6.48.2 Field Documentation

### 6.48.2.1 J2735\_2016\_LatLong\_t node\_set\_ll

- node\_set\_ll - enclosed structure for the set of ll node offset points

### 6.48.2.2 J2735\_2016\_NodeXY\_t node\_set\_xy

- node\_set\_xy - enclosed structure for the set of xy node offset points

### 6.48.2.3 J2735\_2016\_OffsetSystemType\_t off\_sys\_type

off\_sys\_type - offset system type in use - only xy are supported

#### 6.48.2.4 int zoom\_level

zoom\_level - zoom level to scale the offsets. Zoom level reduces the precision of the offsets while increasing the range

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

- [v2x\\_msg\\_tim.h](#)

## 6.49 J2735\_2016\_RegionOffsets Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- [J2735Position3D\\_t](#) **pos3d**

#### 6.49.1 Detailed Description

J2735RegionPointSet\_t - Region pointset

This is used to represent or describe an enclosed region. It is typically employed to define a region where signs or advisories would be valid.

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

- [v2x\\_msg\\_tim.h](#)

## 6.50 J2735\_2016\_RequestorType\_t Struct Reference

### Data Fields

- J2735\_2016\_BasicVehicleRole\_t **role**
- int **req\_subrole**
- int **req\_importance\_level**
- int **iso\_3833\_vehicle\_type**
- J2735\_2016\_VehicleType\_t **hpms\_vehicle\_type**

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

- [v2x\\_msg\\_common.h](#)

## 6.51 J2735\_2016\_SignalRequest Struct Reference

```
#include <v2x_msg_ssm.h>
```

### Data Fields

- [J2735VehicleID\\_t](#) **vehicle\_id**
- int **request\_id**
- int **msgcount**

- J2735\_2016\_BasicVehicleRole\_t [basic\\_vehicle\\_role](#)
- int [requestor\\_type\\_present](#)
- J2735\_2016\_RequestorType\_t [requestor\\_type](#)

### 6.51.1 Detailed Description

Signal request info

### 6.51.2 Field Documentation

#### 6.51.2.1 J2735\_2016\_BasicVehicleRole\_t [basic\\_vehicle\\_role](#)

basic vehicl role

#### 6.51.2.2 int [msgcount](#)

msg count

#### 6.51.2.3 int [request\\_id](#)

request ID - unique number

#### 6.51.2.4 J2735\_2016\_RequestorType\_t [requestor\\_type](#)

requestor type

#### 6.51.2.5 int [requestor\\_type\\_present](#)

set to 1, if [requestor\\_type](#) is being filled

#### 6.51.2.6 J2735VehicleID\_t [vehicle\\_id](#)

Vehicle ID - unique number

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

- [v2x\\_msg\\_ssm.h](#)

## 6.52 J2735\_2016\_SignalRequest\_t Struct Reference

```
#include <v2x_msg_srm.h>
```

### Data Fields

- int [iid](#)
- int [request\\_id](#)
- J2735\_2016\_PriorityReqType\_t [req\\_type](#)
- J2735\_2016\_IsecAccessPoint\_t [in\\_bound\\_access\\_point](#)
- J2735\_2016\_IsecAccessPoint\_t [out\\_bound\\_access\\_point](#)

### 6.52.1 Detailed Description

Signal request description for each lane / particular lane

### 6.52.2 Field Documentation

#### 6.52.2.1 int iid

intersection id

#### 6.52.2.2 J2735\_2016\_IsecAccessPoint\_t in\_bound\_access\_point

inbound accesspoint (lane, approach)

#### 6.52.2.3 J2735\_2016\_IsecAccessPoint\_t out\_bound\_access\_point

outbound accesspoint (lane, approach)

#### 6.52.2.4 J2735\_2016\_PriorityReqType\_t req\_type

request type

#### 6.52.2.5 int request\_id

request id = tempid

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

- [v2x\\_msg\\_srm.h](#)

## 6.53 J2735\_2016\_SignalRequestorInfo\_t Struct Reference

### Data Fields

- [J2735VehicleID\\_t](#) **vehicle\_id**

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

- [v2x\\_msg\\_common.h](#)

## 6.54 J2735\_2016\_SignalStatusList Struct Reference

```
#include <v2x_msg_ssm.h>
```

### Data Fields

- int [msgcount](#)
- int [iid](#)
- int [ss\\_pkg\\_list\\_len](#)
- [J2735\\_2016\\_SignalStatusPackageList\\_t](#) [ss\\_pkg\\_list](#) [SSM\_SS\_PKG\_LIST\_MAX]

### 6.54.1 Detailed Description

Signal status list

### 6.54.2 Field Documentation

6.54.2.1 int iid

intersection id

6.54.2.2 int msgcount

msgcount

6.54.2.3 J2735\_2016\_SignalStatusPackageList\_t ss\_pkg\_list[SSM\_SS\_PKG\_LIST\_MAX]

signal status package list

6.54.2.4 int ss\_pkg\_list\_len

signal status package list count

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

- [v2x\\_msg\\_ssm.h](#)

## 6.55 J2735\_2016\_SignalStatusPackageList Struct Reference

```
#include <v2x_msg_ssm.h>
```

### Data Fields

- [J2735\\_2016\\_SignalRequesterInfo\\_t](#) signal\_request
- [J2735\\_2016\\_IsecAccessPoint\\_t](#) in\_bound\_access\_point
- int valid\_out\_bound\_access\_point
- [J2735\\_2016\\_IsecAccessPoint\\_t](#) out\_bound\_access\_point
- int min\_of\_year
- int second
- int duration
- [J2735\\_2016\\_PrioritizationResponseStatus\\_t](#) prio\_resp

### 6.55.1 Detailed Description

Signal status package list

### 6.55.2 Field Documentation

6.55.2.1 int duration

duration

#### 6.55.2.2 J2735\_2016\_IsecAccessPoint\_t in\_bound\_access\_point

inbound access point (lane, approach)

#### 6.55.2.3 int min\_of\_year

minute of the year

#### 6.55.2.4 J2735\_2016\_IsecAccessPoint\_t out\_bound\_access\_point

outbound access point (lane, approach)

#### 6.55.2.5 J2735\_2016\_PrioritizationResponseStatus\_t prio\_resp

priority response status

#### 6.55.2.6 int second

second

#### 6.55.2.7 J2735\_2016\_SignalRequesterInfo\_t signal\_request

Requestor info

#### 6.55.2.8 int valid\_out\_bound\_access\_point

set to 1, if outbound access point is available

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

- [v2x\\_msg\\_ssm.h](#)

## 6.56 j2735\_2016\_srm Struct Reference

```
#include <v2x_msg_srm.h>
```

### Data Fields

- int [dsecond](#)
- int [msg\\_count](#)
- int [srm\\_list\\_count](#)
- [J2735SRMList\\_t srmlist](#) [SRM\_MSG\_LIST\_MAX]
- [J2735SRMRequestor\\_t requestor](#)

### 6.56.1 Detailed Description

SRM message



## 6.56.2 Field Documentation

### 6.56.2.1 int dsecond

dsecond

### 6.56.2.2 int msg\_count

msg count

### 6.56.2.3 J2735SRMRequestor\_t requestor

requestor

### 6.56.2.4 int srm\_list\_count

SRM list count (1, 32)

### 6.56.2.5 J2735SRMList\_t srmlist[SRM\_MSG\_LIST\_MAX]

SRM list

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

- [v2x\\_msg\\_srm.h](#)

## 6.57 j2735\_2016\_ssm Struct Reference

```
#include <v2x_msg_ssm.h>
```

### Data Fields

- int [dsecond](#)
- int [msgcount](#)
- int [sslist\\_len](#)
- [J2735\\_2016\\_SignalStatusList\\_t sslist](#) [SSM\_SIGNAL\_STATUS\_LIST\_MAX]

### 6.57.1 Detailed Description

SSM message

### 6.57.2 Field Documentation

#### 6.57.2.1 int dsecond

dsecond

#### 6.57.2.2 int msgcount

msgcount

### 6.57.2.3 J2735\_2016\_SignalStatusList\_t slist[SSM\_SIGNAL\_STATUS\_LIST\_MAX]

signal status list

### 6.57.2.4 int slist\_len

signal status list count (1 .. 32)

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

- [v2x\\_msg\\_ssm.h](#)

## 6.58 J2735\_2016\_VehicleClassification\_t Struct Reference

### Data Fields

- J2735\_2016\_BasicVehicleClass\_t **basic\_veh\_class**
- J2735\_2016\_BasicVehicleRole\_t **basic\_veh\_role**
- int **iso\_3833\_vehicle\_type**
- J2735\_2016\_VehicleType\_t **vehicle\_type**
- J2735\_2016\_VgroupAffected\_t **vgroup\_affected**
- J2735\_2016\_RgroupAffected\_t **rgroup\_affected**
- J2735\_2016\_IREquipment\_t **irequipment**

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

- [v2x\\_msg\\_common.h](#)

## 6.59 J2735\_2016\_VehicleIdent\_t Struct Reference

### Data Fields

- int **vin\_string\_len**
- char **vin\_string** [J2735\_2016\_VIN\_STRING\_MAX]
- int64\_t **vehicle\_id**
- J2735\_2016\_VehicleType\_t **vehicle\_type**
- J2735\_2016\_VgroupAffected\_t **vgroup\_affected**
- J2735\_2016\_RgroupAffected\_t **rgroup\_affected**
- J2735\_2016\_IREquipment\_t **ir\_equipment**

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

- [v2x\\_msg\\_common.h](#)

## 6.60 J2735CircularRegion Struct Reference

```
#include <v2x_msg_tim.h>
```

## Data Fields

- [J2735Position3D\\_t position](#)
- [J2735RadiusType\\_t radius\\_type](#)
- double [value](#)

### 6.60.1 Detailed Description

J2735CircularRegion\_t - Circular region

denotes the valid region in the form of a circle.

### 6.60.2 Field Documentation

#### 6.60.2.1 J2735Position3D\_t position

position - position 3d point with lat, long and elev

#### 6.60.2.2 J2735RadiusType\_t radius\_type

radius\_type - one of J2735RadiusType\_t types

#### 6.60.2.3 double value

value - value of the radius, the value should be filled based on type

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

- [v2x\\_msg\\_tim.h](#)

## 6.61 J2735Position3D Struct Reference

```
#include <v2x_msg_common.h>
```

## Data Fields

- double [latitude](#)
- double [longitude](#)
- double [elevation](#)

### 6.61.1 Detailed Description

J2735Position3D\_t - position 3d object

### 6.61.2 Field Documentation

#### 6.61.2.1 double elevation

elevation in cms

### 6.61.2.2 double latitude

latitude from -90.000000 to 90.000000

### 6.61.2.3 double longitude

longitude from -180.000000 to 180.000000

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

- [v2x\\_msg\\_common.h](#)

## 6.62 J2735RegionPointSet Struct Reference

### Data Fields

- [J2735Position3D\\_t](#) \* [position](#)
- int [zoom\\_level](#)
- int [region\\_offset\\_len](#)
- [J2735\\_2016\\_RegionOffsets](#) [reg\\_offsets](#) [[J2735\\_2016\\_REG\\_OFFSET\\_MAX](#)]
- int [num\\_wp](#)
- [J2735Position3D\\_t](#) \* [wplist](#)

### 6.62.1 Field Documentation

#### 6.62.1.1 int num\_wp

num\_reg - number of region point offsets that are in the regions pointer below

#### 6.62.1.2 J2735Position3D\_t\* position

position - position 3d point with lat, long and elev if non null this will be filled into the region points of TIM

#### 6.62.1.3 J2735\_2016\_RegionOffsets reg\_offsets[J2735\_2016\_REG\_OFFSET\_MAX]

reg\_offsets - region offsets of type position3d which are then converted back into offsets per 2016

#### 6.62.1.4 int region\_offset\_len

region\_offset\_len - length of the region offsets (1 to 64)

#### 6.62.1.5 J2735Position3D\_t\* wplist

regions - region offsets from the position point if present otherwise the offsets are from the common anchor point

#### 6.62.1.6 int zoom\_level

zoom\_level - zoom level to scale the offsets. Zoom level reduces the precision of the offsets while increasing the range

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

- [v2x\\_msg\\_tim.h](#)

## 6.63 J2735RoadSignID Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- [J2735Position3D\\_t](#) position
- [J2735\\_2016\\_HeadingSlice\\_t](#) heading\_slice
- [J2735MUTCDCode\\_t](#) mutcd\_code

### 6.63.1 Detailed Description

RoadSign ID

### 6.63.2 Field Documentation

#### 6.63.2.1 J2735\_2016\_HeadingSlice\_t heading\_slice

Heading slice. Value should be one of J2735HeadingSlice\_t enum values.

#### 6.63.2.2 J2735MUTCDCode\_t mutcd\_code

MUTCD Code

#### 6.63.2.3 J2735Position3D\_t position

Position 3d object

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

- [v2x\\_msg\\_tim.h](#)

## 6.64 J2735ShapePointSet Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- [J2735Position3D\\_t](#) position
- double lanewidth
- [J2735DirectionOfUse\\_t](#) directionofuse
- int num\_wp
- [J2735Position3D\\_t](#) \* wplist

### 6.64.1 Detailed Description

J2735ShapePoints\_t - shapepoint set region the shapepoint region is a region formed by connecting the lat, long and elevation points. these are found by adding offsets to the anchor point.

## 6.64.2 Field Documentation

### 6.64.2.1 double lanewidth

lanewidth - common lanewidth for the described shape points at the wplist

### 6.64.2.2 int num\_wp

num\_nodelist - no of nodes

### 6.64.2.3 J2735Position3D\_t position

position - anchor point to tell the base position

### 6.64.2.4 J2735Position3D\_t\* wplist

nodelist -set of nodes conforming to this shapepoint region

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

- [v2x\\_msg\\_tim.h](#)

## 6.65 J2735TIM Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- int [ssp\\_tim\\_rights](#)
- [J2735TravelerInfoType\\_t](#) type
- uint32\_t [further\\_info\\_id](#)
- [J2735RoadSignID\\_t](#) road\_sign\_id
- int32\_t [start\\_year](#)
- uint32\_t [start\\_minute\\_of\\_year](#)
- int32\_t [duration](#)
- int32\_t [priority](#)
- int [ssp\\_loc\\_rights](#)
- int [n\\_geo\\_paths](#)
- [J2735TIMGeoPath\\_t](#) geographical\_path [TIM\_GEOG\_PATH\_MAX]
- int [ssp\\_msg\\_rights1](#)
- int [ssp\\_msg\\_rights2](#)
- uint32\_t [content\\_type](#)
- uint32\_t [num\\_advisories](#)
- uint32\_t \* [advisory\\_codes](#)
- uint32\_t [num\\_workzones](#)
- uint32\_t \* [workzone\\_codes](#)
- uint32\_t [num\\_genericsigns](#)
- uint32\_t \* [genericsign\\_codes](#)
- uint32\_t [num\\_speedlimits](#)
- uint32\_t \* [speedlimit\\_codes](#)
- uint32\_t [num\\_exitsservices](#)
- uint32\_t \* [exitsservice\\_codes](#)

### 6.65.1 Detailed Description

J2735DATAFRAME\_t - dataframe of TIM message

### 6.65.2 Field Documentation

#### 6.65.2.1 uint32\_t\* advisory\_codes

advisory\_codes - the advisory codes set it is allocated to a num\_advisories size

#### 6.65.2.2 uint32\_t content\_type

content\_type - one of J2735ContentType\_t. Only one of the all codes should be specified. at a time ORing or ANDing of two content types is invalid or wrong assignment and the remaining groups should have 0 codes.

#### 6.65.2.3 int32\_t duration

duration - the duration of each TIM message 0 - min 32000 - max the duration is in minutes

#### 6.65.2.4 uint32\_t\* exitservice\_codes

exitservice\_codes - the exit service codes set this pointer is allocated to num\_exitsservices size.

#### 6.65.2.5 uint32\_t further\_info\_id

further\_info\_id - further info ID of the TIM message

#### 6.65.2.6 uint32\_t\* genericsign\_codes

genericsign\_codes - the genericsigns codes set this pointer is allocated to num\_genericsigns size.

#### 6.65.2.7 J2735TIMGeoPath\_t geographical\_path[TIM\_GEOG\_PATH\_MAX]

geographical\_path - geographical path representing the TIM region

#### 6.65.2.8 int n\_geo\_paths

n\_geo\_path - number of geographic path points (1 to 16)

#### 6.65.2.9 uint32\_t num\_advisories

num\_advisories - number of advisory codes to encode

#### 6.65.2.10 uint32\_t num\_exitsservices

num\_exitsservices - number of exit service codes to encode

#### 6.65.2.11 uint32\_t num\_genericsigns

num\_genericsigns - number of generic sign codes to encode

**6.65.2.12 uint32\_t num\_speedlimits**

num\_speedlimits - number of speed limit codes to encode

**6.65.2.13 uint32\_t num\_workzones**

num\_workzones - number of workzone codes to encode

**6.65.2.14 int32\_t priority**

priority - TIM message priority

**6.65.2.15 J2735RoadSignID\_t road\_sign\_id**

road\_sign\_id - roadsign ID of the TIM message

**6.65.2.16 uint32\_t\* speedlimit\_codes**

speedlimit\_codes - the speedlimits codes set this pointer is allocated to num\_speedlimits size.

**6.65.2.17 int ssp\_loc\_rights**

ssp\_loc\_rights - ssp bits that are set in 1609.2 of this message and for this content

**6.65.2.18 int ssp\_msg\_rights1**

ssp\_msg\_rights1 - ssp bits that are set in 1609.2 of this message and for this content

**6.65.2.19 int ssp\_msg\_rights2**

ssp\_msg\_rights2 - ssp bits that are set in 1609.2 of this message and for this content

**6.65.2.20 int ssp\_tim\_rights**

ssp\_tim\_rights - ssp bits that are set in 1609.2 of this message and for this header

**6.65.2.21 uint32\_t start\_minute\_of\_year**

start\_minute\_of\_year - start minute of the year, it is from JAN1 at 00:00 hrs and mins, and at current year to the present date, hrs:mins.

**6.65.2.22 int32\_t start\_year**

start\_year - the year at which the message started transmitting



### 6.65.2.23 J2735TravelerInfoType\_t type

type - type is one of J2735TravelerInfoType\_t type.

if its further info id , the further\_info\_id field should be populated

if its road sign id, then the road\_sign\_id field should be populated

### 6.65.2.24 uint32\_t\* workzone\_codes

workzone\_codes - the workzone codes set this pointer is allocated to num\_workzones size

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

- [v2x\\_msg\\_tim.h](#)

## 6.66 J2735TIMGeoPath Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- [RoadSegmentRefId\\_t id](#)
- int [valid\\_pos3d](#)
- [J2735Position3D\\_t pos3d](#)
- double [lane\\_width](#)
- [J2735DirectionOfUse\\_t directionality](#)
- int [closedpath](#)
- [J2735\\_2016\\_HeadingSlice\\_t heading\\_slice](#)
- [J2735\\_2016\\_ProjectionType\\_t projection\\_type](#)
- [J2735\\_2016\\_OffsetSystem\\_t offset\\_system](#)
- [J2735\\_2016\\_GeometricProjection\\_t geometric\\_projection](#)
- [J2735ValidRegion\\_t valid\\_region](#)

### 6.66.1 Detailed Description

J2735TIMGeoPath\_t - geographic path representation of the TIM message this allows TIM to specifically inform the valid regions of the message. The geographic path consists of different set of regions to describe the validity.

### 6.66.2 Field Documentation

#### 6.66.2.1 int closedpath

closedpath - when set the last point closes to first.

#### 6.66.2.2 J2735DirectionOfUse\_t directionality

directionality - direction of this message. A car going in opposite direction will ignore this message

#### 6.66.2.3 J2735\_2016\_GeometricProjection\_t geometric\_projection

geometric\_projection - geometric projection describing different set of geometric areas

#### 6.66.2.4 J2735\_2016\_HeadingSlice\_t heading\_slice

heading\_slice - field of view of this message

#### 6.66.2.5 RoadSegmentRefId\_t id

id - unique roadsegment that this message refers to

#### 6.66.2.6 double lane\_width

lane\_width - common lanewidth of the node points

#### 6.66.2.7 J2735\_2016\_OffsetSystem\_t offset\_system

offset\_system - offset system describing the precise node points

#### 6.66.2.8 J2735Position3D\_t pos3d

pos3d - anchor point to describe the regions with node offsets from here.

#### 6.66.2.9 J2735\_2016\_ProjectionType\_t projection\_type

projection\_type - type of the projection in use. one of the offset\_system, geometric\_projection or valid\_region.

#### 6.66.2.10 int valid\_pos3d

valid\_pos3d - a bit to inform if the pos3d is valid. set to 1 if pos3d is valid, 0 otherwise

#### 6.66.2.11 J2735ValidRegion\_t valid\_region

valid\_region - older valid region defined in 2009. Not to be used in 2016.

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

- [v2x\\_msg\\_tim.h](#)

## 6.67 J2735ValidRegion Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- [J2735\\_2016\\_HeadingSlice\\_t heading\\_slice](#)
- double extent
- [J2735RegionsList\\_t region\\_flag](#)
- [J2735ShapePoints\\_t shape\\_points](#)
- [J2735CircularRegion\\_t circle](#)
- [J2735RegionPointSet\\_t regionpoint](#)

### 6.67.1 Detailed Description

J2735ValidRegion\_t - valid region, as of now only J2735\_REGION\_SHAPEPT is supported for a valid\_region.

As of 2016 J2735 standard, the Valid regions are legacy and shall not be used

### 6.67.2 Field Documentation

#### 6.67.2.1 J2735CircularRegion\_t circle

circle - circular region data

#### 6.67.2.2 double extent

extent - extent a value in meters a value of 0 implies the use of extent instantaneously

#### 6.67.2.3 J2735\_2016\_HeadingSlice\_t heading\_slice

heading\_slice. Value should be one of J2735HeadingSlice\_t enum values

#### 6.67.2.4 J2735RegionsList\_t region\_flag

region\_flag - region type

#### 6.67.2.5 J2735RegionPointSet\_t regionpoint

regionpoint - region point set data

#### 6.67.2.6 J2735ShapePoints\_t shape\_points

shape\_points - shape point data

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

- [v2x\\_msg\\_tim.h](#)

## 6.68 J2735VehicleID\_t Struct Reference

### Data Fields

- int64\_t **temp\_id**
- int64\_t **station\_id**

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

- [v2x\\_msg\\_common.h](#)

## 6.69 lane\_validity\_params Struct Reference

### Data Fields

- LaneValidity\_Days\_t **lv\_days**
- int **lane\_discarded**
- int **is\_next\_day\_active**
- int **n\_lv\_time**
- [Savari\\_LVTime\\_t](#) \* **lv\_time**

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

- [v2x\\_msg\\_common.h](#)

## 6.70 lane\_validity\_time Struct Reference

### Data Fields

- int **start\_time**
- int **end\_time**

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

- [v2x\\_msg\\_common.h](#)

## 6.71 laneattr Struct Reference

```
#include <v2x_msg_common.h>
```

### Data Fields

- [J2735\\_2016\\_LaneDir\\_t](#) **direction**
- [J2735\\_2016\\_SharedLaneType\\_t](#) **sharedlane**
- [J2735\\_2016\\_LaneTypeAttrVeh\\_t](#) \* **vehattr**
- [J2735\\_2016\\_LaneTypeAttrCrossWk\\_t](#) \* **crwattr**

### 6.71.1 Detailed Description

Lane attributes describing the lane such as the direction, shared lanes, vehicle or cross walk attributes

### 6.71.2 Field Documentation

#### 6.71.2.1 J2735\_2016\_LaneTypeAttrCrossWk\_t\* crwattr

crwattr - crosswalk attributes

#### 6.71.2.2 J2735\_2016\_LaneDir\_t direction

direction - lane direction

## 6.71.2.3 J2735\_2016\_SharedLaneType\_t sharedlane

sharedlane - shared with any other traffic such as motor, car, cycle etc

## 6.71.2.4 J2735\_2016\_LaneTypeAttrVeh\_t\* vehattr

vehattr - vehicle attributes

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

- [v2x\\_msg\\_common.h](#)

## 6.72 lanedataattr\_list Struct Reference

### Data Fields

- double **path\_end\_point\_angle**
- double **lane\_crown\_point\_center**
- double **lane\_crown\_point\_left**
- double **lane\_crown\_point\_right**
- double **lane\_angle**
- int **n\_splim**
- [J2735\\_2016\\_SpeedLimit\\_t](#) \* **splim**

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

- [v2x\\_msg\\_common.h](#)

## 6.73 laneset Struct Reference

```
#include <v2x_msg_common.h>
```

### Data Fields

- uint32\_t **lane\_id**
- int **phase**
- [J2735\\_2016\\_AllowedManeuvers\\_t](#) **man**
- int **signal\_group\_id**
- [J2735\\_2016\\_LaneAttribute\\_t](#) **lane\_attr**
- int **n\_nodepoint**
- [J2735\\_2016\\_NodePoint\\_t](#) \* **node\_point**
- int **n\_connect**
- [J2735\\_2016\\_Connectsto\\_Lanes\\_t](#) \* **connects\_to**
- [Savari\\_LVParams\\_t](#) \* **lv\_params**

### 6.73.1 Detailed Description

lane describing the information pertaining to the lane such as its phase, allowed maneuvers, signal group ID, lane attributes and a set of nodes

## 6.73.2 Field Documentation

### 6.73.2.1 J2735\_2016\_Connectsto\_Lanes\_t\* connects\_to

[connects\\_to](#) - connection list from this lane

### 6.73.2.2 J2735\_2016\_LaneAttribute\_t lane\_attr

lane\_attr - lane attributes

### 6.73.2.3 uint32\_t lane\_id

lane\_id - lane id

### 6.73.2.4 J2735\_2016\_AllowedManeuvers\_t man

man - allowed maneuvers

### 6.73.2.5 int n\_connect

n\_connect - number of connects to list

### 6.73.2.6 int n\_nodepoint

n\_nodepoint - number of nodes 2 - 63

### 6.73.2.7 J2735\_2016\_NodePoint\_t\* node\_point

node\_point - node point list describing the lane

### 6.73.2.8 int phase

phase - phase that is valid for this lane

### 6.73.2.9 int signal\_group\_id

signal\_group\_id - signal group ID mapping with the corresponding SPAT

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

- [v2x\\_msg\\_common.h](#)

## 6.74 libcan\_filters Struct Reference

### Data Fields

- int **can\_id**
- canmsg\_type **can\_type**

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

- [v2x\\_canctl.h](#)

## 6.75 libwme\_gpsinfo Struct Reference

### Data Fields

- int **latitude**
- int **longitude**
- int **elevation**
- double **pos\_confidence**
- double **elev\_confidence**
- int **positional\_accuracy**

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

- [v2x\\_net\\_wme.h](#)

## 6.76 libwme\_radio\_settings::libwme\_radio Struct Reference

### Data Fields

- int **channel\_service**
- int **channel\_mode**

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

- [v2x\\_net\\_wme.h](#)

## 6.77 libwme\_radio\_settings Struct Reference

### Data Structures

- struct [libwme\\_radio](#)

### Data Fields

- int **num\_radios**
- struct [libwme\\_radio\\_settings::libwme\\_radio](#) **radio** [LIBWME\_RADIO\_MAX]

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

- [v2x\\_net\\_wme.h](#)

## 6.78 nodeattrs Struct Reference

```
#include <v2x_msg_common.h>
```

## Data Fields

- [J2735\\_2016\\_NodeAttrType\\_t](#) type
- int [n\\_lane\\_data\\_attr](#)
- [J2735\\_2016\\_LaneDataAttributes\\_t](#) \* [lane\\_data\\_attr](#)
- double [dWidth](#)
- double [dElev](#)

### 6.78.1 Detailed Description

nodeattributes of a particular node point

### 6.78.2 Field Documentation

#### 6.78.2.1 double [dElev](#)

[dElev](#) - delta elevation

#### 6.78.2.2 double [dWidth](#)

[dWidth](#) - delta lanewidth

#### 6.78.2.3 [J2735\\_2016\\_LaneDataAttributes\\_t](#)\* [lane\\_data\\_attr](#)

[lane\\_data\\_attr](#) - lane data attribute list

#### 6.78.2.4 int [n\\_lane\\_data\\_attr](#)

[n\\_lane\\_data\\_attr](#) - number of lane data attributes

#### 6.78.2.5 [J2735\\_2016\\_NodeAttrType\\_t](#) type

[type](#) - node attribute types

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

- [v2x\\_msg\\_common.h](#)

## 6.79 nodelist Struct Reference

```
#include <v2x_msg_common.h>
```

## Data Fields

- [J2735\\_2016\\_Pos3d\\_2\\_t](#) node
- double [lanewidth](#)

### 6.79.1 Detailed Description

list of nodes



## 6.79.2 Field Documentation

### 6.79.2.1 double lanewidth

lanewidth - unused, place holder

### 6.79.2.2 J2735\_2016\_Pos3d\_2\_t node

node - position3d of a node point

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

- [v2x\\_msg\\_common.h](#)

## 6.80 nodepoint\_list Struct Reference

```
#include <v2x_msg_common.h>
```

### Data Fields

- [J2735\\_2016\\_NodeList\\_t node\\_id](#)
- [J2735\\_2016\\_NodeAttr\\_t nodeattr](#)

### 6.80.1 Detailed Description

- 

node point describing a node and its attributes

### 6.80.2 Field Documentation

#### 6.80.2.1 J2735\_2016\_NodeList\_t node\_id

node\_id - node expressed in pos3d form

#### 6.80.2.2 J2735\_2016\_NodeAttr\_t nodeattr

nodeattr - attributes of a node

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

- [v2x\\_msg\\_common.h](#)

## 6.81 path\_history Struct Reference

```
#include <v2x_msg_common.h>
```

## Data Fields

- double [lat\\_offset](#)
- double [long\\_offset](#)
- double [elev\\_offset](#)
- uint32\_t [time\\_offset](#)
- double [positional\\_accuracy](#)
- double [heading](#)

### 6.81.1 Detailed Description

path history information represented in the positional offsets

### 6.81.2 Field Documentation

#### 6.81.2.1 double elev\_offset

elev\_offset - elevation offset

#### 6.81.2.2 double heading

heading - heading

#### 6.81.2.3 double lat\_offset

lat\_offset - latitude offset

#### 6.81.2.4 double long\_offset

long\_offset - longitude offset

#### 6.81.2.5 double positional\_accuracy

positional\_accuracy - positional accuracy

#### 6.81.2.6 uint32\_t time\_offset

time\_offset - time offset

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

- [v2x\\_msg\\_common.h](#)

## 6.82 pathhistoryframe Struct Reference

```
#include <v2x_msg_bsm.h>
```

## Data Fields

- double [lat\\_offset](#)
- double [long\\_offset](#)
- double [elev\\_offset](#)
- int [time\\_offset](#)
- double [positional\\_accuracy](#)
- double [heading](#)

### 6.82.1 Detailed Description

Path history information contains the data needed to encode a pathhistory element in the BSM. The offsets are the difference of the sequence of lat, long and elev points multiplied by the 1/10th of a microdegree value from the current position.

### 6.82.2 Field Documentation

#### 6.82.2.1 double elev\_offset

Elevation offset

#### 6.82.2.2 double heading

Heading of the vehicle

#### 6.82.2.3 double lat\_offset

Latitude offset

#### 6.82.2.4 double long\_offset

Longitude offset

#### 6.82.2.5 double positional\_accuracy

Positional accuracy

#### 6.82.2.6 int time\_offset

Time offset

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

- [v2x\\_msg\\_bsm.h](#)

## 6.83 pf\_gnss\_data Struct Reference

```
#include <v2x_sensor_gnss.h>
```

## Data Fields

- FLOAT64\_t [time](#)
- INT32\_t [mode](#)
- CHAR\_t [utc\\_time](#) [PF\_GNSS\_MAX\_TIME\_STR]
- FLOAT64\_t [latitude](#)
- FLOAT64\_t [longitude](#)
- FLOAT64\_t [altitude](#)
- FLOAT64\_t [dsecond](#)
- FLOAT64\_t [speed](#)
- FLOAT64\_t [heading](#)
- FLOAT64\_t [semi\\_major\\_deviation](#)
- FLOAT64\_t [semi\\_minor\\_deviation](#)
- FLOAT64\_t [semi\\_major\\_orientation](#)
- FLOAT64\_t [lonaccel](#)
- FLOAT64\_t [lataccel](#)
- FLOAT64\_t [vertaccel](#)
- FLOAT64\_t [yawrate](#)
- INT32\_t [gps\\_update\\_hz](#)
- FLOAT64\_t [linear\\_accel\\_filter\\_cutoff\\_hz](#)
- FLOAT64\_t [linear\\_accel\\_filter\\_damp\\_factor](#)
- FLOAT64\_t [angular\\_accel\\_filter\\_cutoff\\_hz](#)
- FLOAT64\_t [angular\\_accel\\_filter\\_damp\\_factor](#)
- void \* [yawrate\\_filter](#)
- void \* [lacceleration\\_filter](#)
- void \* [vspeed\\_filter](#)
- void \* [vacceleration\\_filter](#)
- FLOAT64\_t [hdop](#)
- FLOAT64\_t [vdop](#)
- FLOAT64\_t [tdop](#)
- FLOAT64\_t [pdop](#)
- FLOAT64\_t [hAcc](#)
- FLOAT64\_t [vAcc](#)
- INT32\_t [numSvs](#)

### 6.83.1 Detailed Description

This structure is used to get GNSS data from GNSS service. This structure contains all the navigational information including latitude, longitude, elevation, time and mode of fix, DOP etc,.

### 6.83.2 Field Documentation

#### 6.83.2.1 FLOAT64\_t altitude

Altitude in meters

#### 6.83.2.2 FLOAT64\_t dsecond

Milliseconds in the current minute when the last fix was obtained

#### 6.83.2.3 FLOAT64\_t heading

Heading in degrees

#### 6.83.2.4 FLOAT64\_t latitude

Latitude in degrees

#### 6.83.2.5 FLOAT64\_t longitude

Longitude in degrees

#### 6.83.2.6 INT32\_t mode

mode of fix

- 1 -> no fix
- 2 -> 2D fix
- 3 -> 3d fix

#### 6.83.2.7 FLOAT64\_t speed

KPH speed

#### 6.83.2.8 FLOAT64\_t time

UTC time in double format

#### 6.83.2.9 CHAR\_t utc\_time[PF\_GNSS\_MAX\_TIME\_STR]

UTC time in string format

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

- [v2x\\_sensor\\_gnss.h](#)

## 6.84 position\_confidence Struct Reference

### Data Fields

- int **pos\_conf**
- int **elv\_conf**

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

- [v2x\\_msg\\_common.h](#)

## 6.85 positional\_accuracy\_data Struct Reference

### Data Fields

- unsigned char **a**
- unsigned char **b**

- unsigned short int **c**

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

- [v2x\\_msg\\_common.h](#)

## 6.86 positional\_accuracy\_data\_2016 Struct Reference

### Data Fields

- unsigned short **a**
- unsigned short **b**
- unsigned short **c**

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

- [v2x\\_msg\\_common.h](#)

## 6.87 road\_segment Struct Reference

### Data Fields

- int **id**
- int **msg\_count**
- [J2735\\_2016\\_Pos3d\\_2\\_t](#) **ref\_pos**
- double **lane\_width**
- int **n\_spdlim**
- [J2735\\_2016\\_SpeedLimit\\_t](#) \* **spdlim**
- int **n\_lanes**
- [J2735\\_2016\\_LaneList\\_t](#) \* **lanelist**

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

- [v2x\\_msg\\_common.h](#)

## 6.88 RoadSegmentRefId Struct Reference

```
#include <v2x_msg_tim.h>
```

### Data Fields

- int [region\\_id](#)
- int [roadsegment\\_id](#)

### 6.88.1 Detailed Description

RoadSegmentRefId\_t - RoadSegment ID is unique to a particular region of road segment

## 6.88.2 Field Documentation

### 6.88.2.1 int region\_id

region\_id - unique regional ID

### 6.88.2.2 int roadsegment\_id

roadsegment\_id - ID of the particular road segment

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

- [v2x\\_msg\\_tim.h](#)

## 6.89 savari1609Wra Struct Reference

### Data Fields

- uint16\_t [lifetime](#)
- struct in6\_addr [ipv6addr](#)
- uint8\_t [prefixlen](#)
- struct in6\_addr [default\\_gw](#)
- uint8\_t [gw\\_macaddr](#) [SAVARI\_MAC\_LEN]
- struct in6\_addr [primarydns](#)
- struct in6\_addr [secondarydns](#)

### 6.89.1 Field Documentation

#### 6.89.1.1 struct in6\_addr default\_gw

Default gateway is 128 bit IPv6 address of a router that provides internet connectivity to subnet

#### 6.89.1.2 uint8\_t gw\_macaddr[SAVARI\_MAC\_LEN]

Macaddress of the default gateway.

#### 6.89.1.3 struct in6\_addr ipv6addr

ipv6addr indicates IPv6 subnet prefix of the link

#### 6.89.1.4 uint16\_t lifetime

router lifetime

#### 6.89.1.5 uint8\_t prefixlen

prefixlen indicates the IPv6 subnet prefix of the link. (RFC 3513)

#### 6.89.1.6 struct in6\_addr primarydns

Primary DNS is the 128 bit IPv6 address that can provide DNS lookup for the subnet devices.

### 6.89.1.7 struct in6\_addr secondarydns

Secondary DNS is the 128 bit IPv6 address of an alternate device that can provide DNS lookup for the subnet devices.

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

- [v2x\\_net\\_wme.h](#)

## 6.90 savari1609WsaService Struct Reference

```
#include <v2x_net_wme.h>
```

### Data Fields

- uint8\_t **bit\_mask**
- uint8\_t **priority**
- uint8\_t **radio**
- uint16\_t **port**
- uint32\_t **psid**
- struct in6\_addr **ipv6addr**
- char **contents** [[LIBWME\\_PSC\\_STRING\\_LENGTH](#)]
- uint8\_t **provider\_mac** [SAVARI1609\_IEEE80211\_ADDR\_LEN]
- uint8\_t **rcpi\_threshold**
- uint8\_t **wsa\_count\_threshold**
- uint8\_t **wsa\_count\_threshold\_interval**

### 6.90.1 Detailed Description

struct [savari1609WsaService](#) - WSA Service. No need to use this when running as OBU / ASD.

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

- [v2x\\_net\\_wme.h](#)

## 6.91 savari\_antennaoffset Struct Reference

### Data Fields

- int **antennaoffsetX**
- int **antennaoffsetY**
- int **antennaoffsetZ**

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

- [v2x\\_msg\\_rtcn.h](#)



## 6.92 savari\_rtcn\_payload Struct Reference

### Data Fields

- uint8\_t \* **buf**
- int **payload\_len**

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

- [v2x\\_msg\\_rtcn.h](#)

## 6.93 security\_cmd\_profile\_msg Struct Reference

### Data Fields

- [securityProfileType\\_t](#) **prof\_type**
- struct [aerolink\\_sign\\_secprofile](#) **sign\_profile**
- struct [aerolink\\_verify\\_secprofile](#) **verify\_profile**

### 6.93.1 Field Documentation

#### 6.93.1.1 securityProfileType\_t prof\_type

prof\_type - prof\_type is one of the OR combinations of SECURITY\_SIGNATURE\_PROFILE and / or SECURITY\_VERIFICATION\_PROFILE

#### 6.93.1.2 struct aerolink\_sign\_secprofile sign\_profile

sign\_profile - signature profile

#### 6.93.1.3 struct aerolink\_verify\_secprofile verify\_profile

verify\_profile - verification profile

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

- [v2x\\_security\\_base.h](#)

## 6.94 spedelimits Struct Reference

### Data Fields

- J2735\_2016\_SpeedlimitType\_t **type**
- double **speed**

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

- [v2x\\_msg\\_common.h](#)

## 6.95 srm\_list Struct Reference

```
#include <v2x_msg_srm.h>
```

### Data Fields

- [J2735\\_2016\\_SignalRequest\\_t](#) `signal_request`
- `uint32_t` `min_of_year`
- `int` `second`
- `int` `duration`

### 6.95.1 Detailed Description

SRM set

### 6.95.2 Field Documentation

#### 6.95.2.1 `int` `duration`

duration

#### 6.95.2.2 `uint32_t` `min_of_year`

minute of the year

#### 6.95.2.3 `int` `second`

seconds

#### 6.95.2.4 `J2735_2016_SignalRequest_t` `signal_request`

signal request description

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

- [v2x\\_msg\\_srm.h](#)

## 6.96 srm\_requestor Struct Reference

```
#include <v2x_msg_srm.h>
```

### Data Fields

- `J2735VehicleID_t` `id`
- `int` `full_pos_valid`
- `J2735_2016_FullPosVec_t` `req_pos_vector`
- `int` `requestor_type_valid`
- `J2735_2016_RequestorType_t` `requestor_type`

### 6.96.1 Detailed Description

SRM requestor

### 6.96.2 Field Documentation

#### 6.96.2.1 int full\_pos\_valid

set to 1, if the req\_pos\_vector is being filled

#### 6.96.2.2 J2735VehicleID\_t id

Vehicle ID

#### 6.96.2.3 J2735\_2016\_FullPosVec\_t req\_pos\_vector

full position vector

#### 6.96.2.4 J2735\_2016\_RequestorType\_t requestor\_type

requestor type

#### 6.96.2.5 int requestor\_type\_valid

set to 1 if requestor\_type is being filled

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

- [v2x\\_msg\\_srm.h](#)

## 6.97 testdataframe Struct Reference

### Data Fields

- uint32\_t [formatversion](#)
- uint32\_t [testcheck](#)
- uint8\_t [obeid](#) [6]
- uint32\_t [msgseqnum](#)
- uint32\_t [testconfiguration](#)
- uint32\_t [channelmode](#)
- uint32\_t [datarate](#)
- uint32\_t [msgrate](#)
- uint32\_t [numobe](#)
- uint32\_t [testrun](#)
- uint8\_t [isinvalidtestnum](#)
- struct [timestamp](#) [timestamp](#)
- uint64\_t [time\\_ms](#)
- uint32\_t [congctrl](#)
- uint32\_t [padbytes](#)

### 6.97.1 Field Documentation

#### 6.97.1.1 uint32\_t channelmode

channelmode - the channel mode the radio is configured by the app

#### 6.97.1.2 uint32\_t datarate

datarate

#### 6.97.1.3 uint32\_t formatversion

formatversion - TDF format version

#### 6.97.1.4 uint32\_t msgrate

msgrate - msg transmit rate in msecs

#### 6.97.1.5 uint32\_t msgseqnum

msgseqnum - message sequence number

#### 6.97.1.6 uint32\_t numobe

numobe - number of obes

#### 6.97.1.7 uint8\_t obeid[6]

obeid - obe mac address

#### 6.97.1.8 uint32\_t testcheck

testcheck - TDF testcheck

#### 6.97.1.9 uint32\_t testconfiguration

testconfiguration - test configuration while running the application

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

- [v2x\\_msg\\_bsm.h](#)

## 6.98 timestamp Struct Reference

### Data Fields

- uint32\_t **sec**
- uint32\_t **usec**

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

- [v2x\\_msg\\_bsm.h](#)

## 6.99 utcTimeFrame Struct Reference

```
#include <v2x_msg_common.h>
```

### Data Fields

- long [year](#)
- long [month](#)
- long [day](#)
- long [hour](#)
- long [minute](#)
- long [second](#)

### 6.99.1 Detailed Description

struct [utcTimeFrame](#) - utc timestamp data fits into part2 of BSM

### 6.99.2 Field Documentation

#### 6.99.2.1 long day

day - Day

#### 6.99.2.2 long hour

hour - hour

#### 6.99.2.3 long minute

minute - minute

#### 6.99.2.4 long month

month - Month

#### 6.99.2.5 long second

second - second

#### 6.99.2.6 long year

year - Year

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

- [v2x\\_msg\\_common.h](#)

## 6.100 v2x\_canctl Struct Reference

```
#include <v2x_canctl.h>
```

### Data Fields

- int **mode**
- char \* **devname**
- int [can\\_fd](#)
- int **type**
- int **baud\_rate**
- int **frame\_type**
- void \* **handle**
- int **debugflag**
- FILE \* **hyundai\_fp**
- struct can\_handler \* **handler**
- int **filter\_length**
- struct [libcan\\_filters](#) **filters** [LIBCAN\_FILTERS\_LEN]

### 6.100.1 Detailed Description

struct canctl - canctl structure passed from the app

### 6.100.2 Field Documentation

#### 6.100.2.1 int can\_fd

can file descriptor, filled by the lib

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

- [v2x\\_canctl.h](#)

## 6.101 v2x\_emergency\_details Struct Reference

```
#include <v2x_msg_eva.h>
```

### Data Fields

- int [ssp\\_rights](#)
- J2735\_2016\_SirenInUse\_t [siren\\_in\\_use](#)
- J2735\_2016\_LightBar\_t [lightbar\\_in\\_use](#)
- J2735\_2016\_MultiVehicleResponse\_t [multi\\_vehicle\\_response](#)

### 6.101.1 Detailed Description

Emergency details

### 6.101.2 Field Documentation

#### 6.101.2.1 J2735\_2016\_LightBar\_t lightbar\_in\_use

Light bar in use

#### 6.101.2.2 J2735\_2016\_MultiVehicleResponse\_t multi\_vehicle\_response

Multi vehicle response

#### 6.101.2.3 J2735\_2016\_SirenInUse\_t siren\_in\_use

Siren in use

#### 6.101.2.4 int ssp\_rights

SSP permissions

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

- [v2x\\_msg\\_eva.h](#)

## 6.102 v2x\_etsi\_cam Struct Reference

```
#include <v2x_msg_cam.h>
```

### Data Fields

- [ETSIItsPdu\\_t pdu](#)
- [uint64\\_t generation\\_delta\\_time](#)
- [int type](#)
- [struct v2x\\_etsi\\_cam\\_bc bc](#)
- [struct v2x\\_etsi\\_cam\\_hf hf](#)
- [struct v2x\\_etsi\\_cam\\_lf lf](#)
- [ETSSpecialContainerType\\_t sc\\_type](#)
- [struct etsi\\_cam\\_ptc ptc](#)
- [struct etsi\\_cam\\_stc stc](#)
- [struct etsi\\_cam\\_dgc dgc](#)
- [struct etsi\\_cam\\_rwc\\_basic rwc\\_basic](#)
- [struct etsi\\_cam\\_rc rc](#)
- [struct etsi\\_cam\\_ec ec](#)
- [struct etsi\\_cam\\_scc scc](#)

### 6.102.1 Detailed Description

The CAM Data Structure

### 6.102.2 Field Documentation

#### 6.102.2.1 struct v2x\_etsi\_cam\_bc bc

Basic Container

#### 6.102.2.2 uint64\_t generation\_delta\_time

Generation Delta Time Time of the current reference position

#### 6.102.2.3 struct v2x\_etsi\_cam\_hf hf

High Frequency Container

#### 6.102.2.4 struct v2x\_etsi\_cam\_lf lf

Low Frequency Container

#### 6.102.2.5 ETSIItsPdu\_t pdu

ETSI ITS PDU Header

#### 6.102.2.6 int type

CAM Type Indicates the presence of optional containers

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

- [v2x\\_msg\\_cam.h](#)

### 6.103 v2x\_etsi\_cam\_bc Struct Reference

#### Data Fields

- ETSIStationType\_t [station\\_type](#)
- ETSIPosition\_t [station\\_position](#)

#### 6.103.1 Field Documentation

##### 6.103.1.1 ETSIPosition\_t station\_position

Position and Positional accuracy measured from the reference position of originating ITS-Station

##### 6.103.1.2 ETSIStationType\_t station\_type

Station Type of the originating ITS-Station

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

- [v2x\\_msg\\_cam.h](#)

### 6.104 v2x\_etsi\_cam\_hf Struct Reference

#### Data Fields

- int [type](#)



- struct {  
    struct [v2x\\_etsi\\_cam\\_hf\\_rsu](#) rsu  
    struct [v2x\\_etsi\\_cam\\_hf\\_vehicle](#) vehicle  
} **choice**

### 6.104.1 Field Documentation

#### 6.104.1.1 struct [v2x\\_etsi\\_cam\\_hf\\_rsu](#) rsu

RSU High Frequency Container

#### 6.104.1.2 int type

Type of High Frequency Container

#### 6.104.1.3 struct [v2x\\_etsi\\_cam\\_hf\\_vehicle](#) vehicle

Vehicle High Frequency Container

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

- [v2x\\_msg\\_cam.h](#)

## 6.105 v2x\_etsi\_cam\_hf\_rsu Struct Reference

### Data Fields

- int [n\\_zones](#)
- struct [v2x\\_etsi\\_protected\\_zone](#) zonelist [ETSI\_PROTECTED\_RSU\_ZONE\_LEN]

### 6.105.1 Field Documentation

#### 6.105.1.1 int n\_zones

Number of protected zones

#### 6.105.1.2 struct [v2x\\_etsi\\_protected\\_zone](#) zonelist[ETSI\_PROTECTED\_RSU\_ZONE\_LEN]

Protected zone list

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

- [v2x\\_msg\\_cam.h](#)

## 6.106 v2x\_etsi\_cam\_hf\_vehicle Struct Reference

### Data Fields

- double [heading](#)
- ETSIHeadingAcc\_t [heading\\_acc](#)

- double [speed](#)
- int [speed\\_conf](#)
- ETSDriveDirection\_t [drive\\_direction](#)
- double [vehicle\\_length](#)
- ETSIVehLengthConfidence\_t [vehicle\\_confidence](#)
- double [vehicle\\_width](#)
- double [longitude\\_accel](#)
- int [longitude\\_accel\\_conf](#)
- double [curvature](#)
- int [curvature\\_conf](#)
- ETSICurvatureCalcMode\_t [curvature\\_calc](#)
- double [yawrate](#)
- int [yawrate\\_conf](#)
- ETSIAccelCtrl\_t [accel\\_ctrl](#)
- ETSILanePosition\_t [lane\\_position](#)
- double [steeringwhl\\_angle](#)
- int [steeringwhl\\_angle\\_conf](#)
- double [lateral\\_accel](#)
- int [lateral\\_accel\\_conf](#)
- double [vertical\\_accel](#)
- int [vertical\\_accel\\_conf](#)
- ETSIPerformanceClass\_t [performance\\_class](#)
- double [tollzone\\_lat](#)
- double [tollzone\\_long](#)
- int [tollzone\\_id](#)

### 6.106.1 Field Documentation

#### 6.106.1.1 ETSDriveDirection\_t drive\_direction

Acceleration control in the longitudinal direction.

#### 6.106.1.2 double curvature

Curvature of the vehicle trajectory

#### 6.106.1.3 ETSICurvatureCalcMode\_t curvature\_calc

Curvature calculation mode indicating whether yawrate is used in calculating

#### 6.106.1.4 int curvature\_conf

Curvature confidence

#### 6.106.1.5 ETSDriveDirection\_t drive\_direction

Direction of the vehicle

#### 6.106.1.6 double heading

Heading of the vehicle

**6.106.1.7 ETSIHeadingAcc\_t heading\_acc**

Heading acceleration of the vehicle

**6.106.1.8 ETSILanePosition\_t lane\_position**

Position of the lane

**6.106.1.9 double lateral\_accel**

Lateral acceleration of the vehicle

**6.106.1.10 int lateral\_accel\_conf**

Lateral acceleration confidence

**6.106.1.11 double longitude\_accel**

Longitudinal acceleration of the vehicle

**6.106.1.12 int longitude\_accel\_conf**

Longitudinal acceleration confidence

**6.106.1.13 ETSIPerformanceClass\_t performance\_class**

Performance class of the vehicle indicating age of the CAM data w.r.t generation delta time

**6.106.1.14 double speed**

Speed of the vehicle

**6.106.1.15 int speed\_conf**

Speed Confidence

**6.106.1.16 double steeringwhl\_angle**

Steering wheel angle

**6.106.1.17 int steeringwhl\_angle\_conf**

Steering wheel angle confidence

**6.106.1.18 int tollzone\_id**

Toll zone id

6.106.1.19 double tollzone\_lat

Latitude of the Toll Zone

6.106.1.20 double tollzone\_long

Longitude of the Toll Zone

6.106.1.21 ETSIVehLengthConfidence\_t vehicle\_confidence

Vehicle length confidence

6.106.1.22 double vehicle\_length

Length of the vehicle

6.106.1.23 double vehicle\_width

Vehicle width

6.106.1.24 double vertical\_accel

Vertical acceleration of the vehicle

6.106.1.25 int vertical\_accel\_conf

Vertical acceleration confidence

6.106.1.26 double yawrate

Yaw rate of the vehicle

6.106.1.27 int yawrate\_conf

Yaw rate confidence

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

- [v2x\\_msg\\_cam.h](#)

## 6.107 v2x\_etsi\_cam\_If Struct Reference

### Data Fields

- ETSIVehicleRole\_t [role](#)
- ETSIExtLights\_t [ext\\_lights](#)
- struct etsi\_pathhistory [ph](#)

### 6.107.1 Field Documentation

#### 6.107.1.1 ETSIExtLights\_t ext\_lights

Exterior lights status

#### 6.107.1.2 struct etsi\_pathhistory ph

Path history of the vehicle

#### 6.107.1.3 ETSIVehicleRole\_t role

Vehicle Role of originating ITS-Station

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

- [v2x\\_msg\\_cam.h](#)

## 6.108 v2x\_etsi\_denm Struct Reference

```
#include <v2x_msg_denm.h>
```

### Data Fields

- ETSIIItsPdu\_t pdu
- ETSIDenmContainerTypes\_t type
- struct v2x\_etsi\_denm\_mc mc
- struct v2x\_etsi\_denm\_sc sc
- struct v2x\_etsi\_denm\_lc lc
- struct v2x\_etsi\_denm\_ac ac

### 6.108.1 Detailed Description

DEN Message structure

### 6.108.2 Field Documentation

#### 6.108.2.1 struct v2x\_etsi\_denm\_ac ac

Alacarte Container

#### 6.108.2.2 struct v2x\_etsi\_denm\_lc lc

Location Container

#### 6.108.2.3 struct v2x\_etsi\_denm\_mc mc

Management Container

#### 6.108.2.4 ETSIItsPdu\_t pdu

ETSI ITS PDU Header

#### 6.108.2.5 struct v2x\_etsi\_denm\_sc sc

Situation Container

#### 6.108.2.6 ETSIDenmContainerTypes\_t type

ETSI Containers bit mask

Set the corresponding bit for a container

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

- [v2x\\_msg\\_denm.h](#)

### 6.109 v2x\_etsi\_denm\_ac Struct Reference

```
#include <v2x_msg_denm.h>
```

#### Data Fields

- ETSIACSubContainer\_t [bit\\_mask](#)
- struct etsi\_denm\_rwc [rwc](#)

#### 6.109.1 Detailed Description

Alacarte Container Structure

#### 6.109.2 Field Documentation

##### 6.109.2.1 ETSIACSubContainer\_t bit\_mask

Bit Mask to indicate presence of a sub-container

##### 6.109.2.2 struct etsi\_denm\_rwc rwc

Road Works Container

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

- [v2x\\_msg\\_denm.h](#)

### 6.110 v2x\_etsi\_denm\_lc Struct Reference

```
#include <v2x_msg_denm.h>
```

## Data Fields

- double [event\\_speed](#)
- int [event\\_speed\\_conf](#)
- double [event\\_position\\_heading](#)
- int [event\\_position\\_heading\\_conf](#)
- ETSIEventTraces\_t [event\\_traces](#)
- ETSIRoadType\_t [road\\_type](#)

### 6.110.1 Detailed Description

Location Container Structure

### 6.110.2 Field Documentation

#### 6.110.2.1 double event\_position\_heading

The heading of the event

#### 6.110.2.2 int event\_position\_heading\_conf

The heading confidence of the event

#### 6.110.2.3 double event\_speed

Moving Speed of the event

#### 6.110.2.4 int event\_speed\_conf

Moving Speed Confidence of the event

#### 6.110.2.5 ETSIEventTraces\_t event\_traces

Event traces Path Points leading to the event position

#### 6.110.2.6 ETSIRoadType\_t road\_type

Road Type at the event position

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

- [v2x\\_msg\\_denm.h](#)

## 6.111 v2x\_etsi\_denm\_mc Struct Reference

```
#include <v2x_msg_denm.h>
```

## Data Fields

- ETSIActionID\_t [action\\_id](#)
- uint64\_t [detection\\_time](#)
- uint64\_t [reference\\_time](#)
- ETSITerminationType\_t [termination](#)
- ETSIPosition\_t [event\\_position](#)
- ETSIRelevanceDistance\_t [relevance\\_distance](#)
- ETSIRelevanceTrafDirection\_t [relevance\\_traf\\_dir](#)
- uint32\_t [validity\\_duration](#)
- uint32\_t [transmit\\_interval](#)
- ETSIStationType\_t [station\\_type](#)

### 6.111.1 Detailed Description

Management Container Structure

### 6.111.2 Field Documentation

#### 6.111.2.1 ETSIActionID\_t action\_id

ETSI Action ID which contains station\_id -> The station Id of originating ITS-Station and sequence\_no

#### 6.111.2.2 uint64\_t detection\_time

Event Detection time: The time at which event is detected

#### 6.111.2.3 ETSIPosition\_t event\_position

ETSI Event Position: The point/position at which the event is detected

#### 6.111.2.4 uint64\_t reference\_time

Reference time: To be filled by the DEN basic service

#### 6.111.2.5 ETSIRelevanceDistance\_t relevance\_distance

Relevance distance: The distance starting from event position upto which DEN message is applicable

#### 6.111.2.6 ETSIRelevanceTrafDirection\_t relevance\_traf\_dir

Relevance Traffic Direction: The direction in which DEN message is applicable

#### 6.111.2.7 ETSIStationType\_t station\_type

Station Type of the originating ITS-Station

#### 6.111.2.8 ETSITerminationType\_t termination

DEN Termination type



#### 6.111.2.9 uint32\_t transmit\_interval

Transmit Interval: The interval at which DEN message is transmitted

#### 6.111.2.10 uint32\_t validity\_duration

Validity Duration: The duration upto which DEN message is transmitted

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

- [v2x\\_msg\\_denm.h](#)

## 6.112 v2x\_etsi\_denm\_rwc Struct Reference

```
#include <v2x_msg_denm.h>
```

### Data Fields

- ETSIRWSubContainer\_t [bit\\_mask](#)
- ETSILBSirenUse\_t [lb\\_siren](#)
- ETSIClosedLanes\_t [closed\\_lanes](#)
- ETSIRestrictedTypes\_t [restricted\\_types](#)
- int [speed\\_limit](#)
- ETSICauseCode\_t [incident\\_indication](#)
- ETSIItineraryPath\_t [itinerary\\_path](#)
- ETSIDeltaPosition\_t [starting\\_point\\_splimit](#)
- ETSITrafficRule\_t [traffic\\_rule](#)
- ETSIRefDENMs\_t [ref\\_denms](#)

### 6.112.1 Detailed Description

RoadWorks Structure

### 6.112.2 Field Documentation

#### 6.112.2.1 ETSIRWSubContainer\_t bit\_mask

Bit Mask to indicate the presence of optional fields in this container

#### 6.112.2.2 ETSIClosedLanes\_t closed\_lanes

Closed Lanes

#### 6.112.2.3 ETSICauseCode\_t incident\_indication

Incident related to the roadworks to provide additional information

#### 6.112.2.4 ESIItineraryPath\_t itinerary\_path

PathPoints from start point closest to road works zone to the end

#### 6.112.2.5 ETSILBSirenUse\_t lb\_siren

Light Bar and Siren Stauts

#### 6.112.2.6 ETSIRefDENMs\_t ref\_denms

Reference DEN message actionIDs which represent the same event

#### 6.112.2.7 ETSIRestrictedTypes\_t restricted\_types

Restricted types Type of vehicles restricted in the Road works zone

#### 6.112.2.8 int speed\_limit

Speed limit in the Road works zone

#### 6.112.2.9 ETSIDeltaPosition\_t starting\_point\_splimit

Delta position from the event position from where speedlimit is applicable

#### 6.112.2.10 ETSITrafficRule\_t traffic\_rule

Traffic rule Indicates traffic flow at the road work zone

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

- [v2x\\_msg\\_denm.h](#)

### 6.113 v2x\_etsi\_denm\_sc Struct Reference

```
#include <v2x_msg_denm.h>
```

#### Data Fields

- ETSIInformationQuality\_t [information\\_quality](#)
- ETSICauseCode\_t [event\\_type](#)
- ETSICauseCode\_t [linked\\_cause](#)
- ETSIEventHistoryList\_t [event\\_history\\_list](#)

#### 6.113.1 Detailed Description

Situation Container Structure

#### 6.113.2 Field Documentation

##### 6.113.2.1 ETSIEventHistoryList\_t event\_history\_list

Event History The path in which the same event is detected

## 6.113.2.2 ETSICauseCode\_t event\_type

Event Type which specifies the direct cause and sub cause

## 6.113.2.3 ETSIInformationQuality\_t information\_quality

Information quality level provided by the originating ITS-Station

## 6.113.2.4 ETSICauseCode\_t linked\_cause

Event type with which current DEN is linked with

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

- [v2x\\_msg\\_denm.h](#)

## 6.114 v2x\_etsi\_lane Struct Reference

### Data Fields

- int [lane\\_id](#)
- int [phase\\_no](#)
- int [signal\\_group\\_id](#)
- ETSIApproachType\_t [approach\\_type](#)
- int [ingress\\_approach](#)
- int [egress\\_approach](#)
- struct [v2x\\_etsi\\_lane\\_attr](#) laneattr
- ETSIAllowedManeuvers\_t [maneuvers](#)
- int [nodes\\_len](#)
- [ETSI NodeList2\\_t](#) nodes [ETSI\_MAP\_NODE\_LEN]
- int [connects\\_len](#)
- [ETSIConnectsTo\\_t](#) connects [ETSI\_MAP\_CONNECTS\_LEN]

### 6.114.1 Field Documentation

## 6.114.1.1 ETSIApproachType\_t approach\_type

Type of approach ingress/egress

## 6.114.1.2 ETSIConnectsTo\_t connects[ETSI\_MAP\_CONNECTS\_LEN]

Connecting lanes list

## 6.114.1.3 int connects\_len

Number of connecting lanes to this lane

## 6.114.1.4 int egress\_approach

This value is read is egress approach type is set in the 'approach\_type'.

#### 6.114.1.5 int ingress\_approach

This value is read if ingress approach type is set in the 'approach\_type'.

#### 6.114.1.6 int lane\_id

Lane ID Must be unique w.r.t an Intersection

#### 6.114.1.7 struct v2x\_etsi\_lane\_attr laneattr

Lane attributes

#### 6.114.1.8 ETSIAllowedManeuvers\_t maneuvers

Allowed maneuvers from this lane

#### 6.114.1.9 ETSINodeList2\_t nodes[ETSI\_MAP\_NODE\_LEN]

Nodes List

#### 6.114.1.10 int nodes\_len

Number of node/way points

#### 6.114.1.11 int phase\_no

Phase in which lane is operated

#### 6.114.1.12 int signal\_group\_id

Signal Group of the lane

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

- [v2x\\_msg\\_etsi\\_map.h](#)

### 6.115 v2x\_etsi\_lane\_attr Struct Reference

#### Data Fields

- ETSILaneDirection\_t **lanedir**
- ETSILaneSharing\_t **lanesharing**
- ETSILaneAttrType\_t **laneattr**
- ETSILaneTypeAttrVehicle\_t **vehattr**

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

- [v2x\\_msg\\_etsi\\_map.h](#)

## 6.116 v2x\_etsi\_map Struct Reference

```
#include <v2x_msg_etsi_map.h>
```

### Data Fields

- [ETSIItsPdu\\_t pdu](#)
- [struct v2x\\_etsi\\_map\\_msg map\\_msg](#)

### 6.116.1 Detailed Description

MAP Data Structure

### 6.116.2 Field Documentation

#### 6.116.2.1 [struct v2x\\_etsi\\_map\\_msg map\\_msg](#)

MAP Message structure

#### 6.116.2.2 [ETSIItsPdu\\_t pdu](#)

ETSI ITS PDU Contains PDU version and station\_id

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

- [v2x\\_msg\\_etsi\\_map.h](#)

## 6.117 v2x\_etsi\_map\_isec Struct Reference

### Data Fields

- [int iid](#)
- [int region](#)
- [int msg\\_count](#)
- [ETSIPos3d\\_t ref\\_pt](#)
- [double lanewidth](#)
- [int laneset\\_len](#)
- [struct v2x\\_etsi\\_lane laneset](#) [ETSI\_MAP\_LANE\_LIST\_LEN]

### 6.117.1 Field Documentation

#### 6.117.1.1 [int iid](#)

Intersection ID Must be matched with the corresponding SPAT message

#### 6.117.1.2 [struct v2x\\_etsi\\_lane laneset](#)[ETSI\_MAP\_LANE\_LIST\_LEN]

List of lanes and their info

#### 6.117.1.3 int laneset\_len

Number of lanes in the intersection

#### 6.117.1.4 double lanewidth

Width of the lane

#### 6.117.1.5 int msg\_count

Message count indicating change in the current intersection

#### 6.117.1.6 ETSIPos3d\_t ref\_pt

Reference point position in 3D

#### 6.117.1.7 int region

Intersection region, globally unique regional assignment value

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

- [v2x\\_msg\\_etsi\\_map.h](#)

## 6.118 v2x\_etsi\_map\_msg Struct Reference

### Data Fields

- int [msg\\_subid](#)
- int [msg\\_count](#)
- int [ilist\\_len](#)
- struct [v2x\\_etsi\\_map\\_isec](#) \* [ilist](#)

### 6.118.1 Field Documentation

#### 6.118.1.1 struct v2x\_etsi\_map\_isec\* ilist

Intersection List

#### 6.118.1.2 int ilist\_len

Number of Intersections

#### 6.118.1.3 int msg\_count

Message count indicating the update in the message.

6.118.1.4 int msg\_subid

Message SubID

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

- [v2x\\_msg\\_etsi\\_map.h](#)

## 6.119 v2x\_etsi\_protected\_zone Struct Reference

### Data Fields

- ETSProtectedZoneType\_t **type**
- uint64\_t **expiry\_time\_ms**
- double **lat**
- double **longt**
- double **radius**
- int **zone\_id**

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

- [v2x\\_msg\\_cam.h](#)

## 6.120 v2x\_etsi\_spat Struct Reference

```
#include <v2x_msg_etsi_spat.h>
```

### Data Fields

- ETSltsPdu\_t [pdu](#)
- struct [v2x\\_etsi\\_spat\\_msg](#) [spat\\_msg](#)

### 6.120.1 Detailed Description

SPAT Data Structure

### 6.120.2 Field Documentation

#### 6.120.2.1 ETSltsPdu\_t pdu

ETSI ITS PDU containing PDU version and station\_id

#### 6.120.2.2 struct [v2x\\_etsi\\_spat\\_msg](#) spat\_msg

SPAT message structre

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

- [v2x\\_msg\\_etsi\\_spat.h](#)

## 6.121 v2x\_etsi\_spat\_isec Struct Reference

### Data Fields

- int [isec\\_id](#)
- int [msg\\_count](#)
- ETSIIsecObjStatus\_t [isec\\_status](#)
- int [min\\_of\\_yr](#)
- int [dsecond](#)
- int [enabled\\_lanes\\_len](#)
- int [enabled\\_lanes](#) [ETSI\_SPAT\_ENABLED\_LANES\_LEN]
- int [mmlist\\_len](#)
- struct [v2x\\_etsi\\_spat\\_mm\\_state](#) [mmlist](#) [ETSI\_SPAT\_MMLIST\_LEN]

### 6.121.1 Field Documentation

#### 6.121.1.1 int dsecond

Current Dsecond

#### 6.121.1.2 int enabled\_lanes[ETSI\_SPAT\_ENABLED\_LANES\_LEN]

List of enabled lanes

#### 6.121.1.3 int enabled\_lanes\_len

Number of current enabled lanes

#### 6.121.1.4 int isec\_id

Id of the Intersection Must be same as the corresponding MAP message

#### 6.121.1.5 ETSIIsecObjStatus\_t isec\_status

Intersection Status

#### 6.121.1.6 int min\_of\_yr

Current Minute of the Year

#### 6.121.1.7 struct v2x\_etsi\_spat\_mm\_state mmlist[ETSI\_SPAT\_MMLIST\_LEN]

List of movements and their info

#### 6.121.1.8 int mmlist\_len

Number of Movemnts



6.121.1.9 int msg\_count

Message count indicating update in the message

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

- [v2x\\_msg\\_etsi\\_spat.h](#)

## 6.122 v2x\_etsi\_spat\_mm\_evt Struct Reference

### Data Fields

- ETSMmPhaseState\_t [phase\\_state](#)
- int [start\\_time](#)
- int [min\\_end\\_time](#)
- int [max\\_end\\_time](#)
- int [likely\\_time](#)
- int [confidence](#)
- int [next\\_time](#)

### 6.122.1 Field Documentation

6.122.1.1 int confidence

Likely time confidence

6.122.1.2 int likely\_time

Likely time of the current phase state

6.122.1.3 int max\_end\_time

Max end time of the current phase state

6.122.1.4 int min\_end\_time

Min end time of the current phase state

6.122.1.5 int next\_time

Time at which the current phase state occurs again

6.122.1.6 ETSMmPhaseState\_t [phase\\_state](#)

Current phase state of the Movemnt Event

6.122.1.7 int start\_time

Start time of the current phase state

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

- [v2x\\_msg\\_etsi\\_spat.h](#)

## 6.123 v2x\_etsi\_spat\_mm\_state Struct Reference

### Data Fields

- int [signal\\_group\\_id](#)
- int [mm\\_evt\\_len](#)
- struct [v2x\\_etsi\\_spat\\_mm\\_evt](#) [mm\\_evt](#) [ETSI\_SPAT\_MMEVT\_LEN]

### 6.123.1 Field Documentation

#### 6.123.1.1 struct [v2x\\_etsi\\_spat\\_mm\\_evt](#) [mm\\_evt](#)[ETSI\_SPAT\_MMEVT\_LEN]

List of Movement Events and their info

#### 6.123.1.2 int [mm\\_evt\\_len](#)

Number of Movement Events

#### 6.123.1.3 int [signal\\_group\\_id](#)

Signal Group of the movement

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

- [v2x\\_msg\\_etsi\\_spat.h](#)

## 6.124 v2x\_etsi\_spat\_msg Struct Reference

### Data Fields

- int [spat\\_msg\\_subid](#)
- int [isec\\_list\\_len](#)
- struct [v2x\\_etsi\\_spat\\_isec](#) [isec\\_list](#) [ETSI\_SPAT\_ISEC\_LIST\_LEN]

### 6.124.1 Field Documentation

#### 6.124.1.1 struct [v2x\\_etsi\\_spat\\_isec](#) [isec\\_list](#)[ETSI\_SPAT\_ISEC\_LIST\_LEN]

List of Intersections and their info

#### 6.124.1.2 int [isec\\_list\\_len](#)

Number of Intersections

## 6.124.1.3 int spat\_msg\_subid

SPAT message sub id

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

- [v2x\\_msg\\_etsi\\_spat.h](#)

## 6.125 v2x\_msg\_bsm Struct Reference

```
#include <v2x_msg_bsm.h>
```

### Data Fields

- uint32\_t [msgid](#)
- uint32\_t [msgcount](#)
- int64\_t [temp\\_id](#)
- int32\_t [secmark](#)
- double [latitude](#)
- double [longitude](#)
- double [elevation](#)
- double [positionalaccuracy](#) [3]
- uint32\_t [transmissionstate](#)
- double [speed](#)
- double [heading](#)
- double [angle](#)
- double [longaccel](#)
- double [lataccel](#)
- double [vertaccel](#)
- double [yawrate](#)
- uint32\_t [wheelbrake](#)
- uint32\_t [wheelbrakeavailable](#)
- uint32\_t [sparebit](#)
- int [traction](#)
- int [abs](#)
- int [stabilitycontrol](#)
- uint32\_t [brakeboost](#)
- uint32\_t [auxbrakes](#)
- double [width](#)
- double [length](#)
- uint32\_t [gpsstatus](#)
- uint32\_t [event\\_hazardlights](#)
- uint32\_t [event\\_absactivate](#)
- uint32\_t [event\\_tractionctrlloss](#)
- uint32\_t [event\\_stabilityctrlactivated](#)
- uint32\_t [event\\_hazardbraking](#)
- uint32\_t [event\\_airbag](#)
- int32\_t [count](#)
- struct [pathhistoryframe](#) [phf](#) [23]
- double [radius](#)
- double [confidence](#)
- uint32\_t [lowbeam](#)
- uint32\_t [highbeam](#)

- uint32\_t [leftturnsignal](#)
- uint32\_t [rightturnsignal](#)
- uint32\_t [hazardlights](#)
- uint32\_t [autolightcontrol](#)
- uint32\_t [dtimerunlights](#)
- uint32\_t [foglights](#)
- uint32\_t [parkinglights](#)
- uint32\_t [lightbarinuse](#)
- int [wipers\\_swfnt](#)
- int [wipers\\_rtfnt](#)
- int [wipers\\_swrear](#)
- int [wipers\\_rtrear](#)
- double [throttle\\_pos](#)
- double [vehicleheight](#)
- double [bumperheight\\_front](#)
- double [bumperheight\\_rear](#)
- int [vehiclemass](#)
- uint32\_t **[basicvehicleclass](#)**
- uint32\_t **[siren\\_in\\_use](#)**
- J2735\_2016\_LightBar\_t [lightbar\\_in\\_use](#)
- J2540\_2\_ITISCodes\_t [emergency\\_type\\_event](#)
- int [event\\_descr\\_len](#)
- J2540\_2\_ITISCodes\_t [event\\_descr](#) [8]
- uint32\_t [vehicletype](#)
- struct [testdataframe](#) tdf
- struct [utcTimeFrame](#) utime
- uint32\_t [time\\_confidence](#)
- uint32\_t [pos\\_confidence](#)
- uint32\_t [elev\\_confidence](#)
- uint32\_t [head\\_confidence](#)
- uint32\_t [speed\\_confidence](#)
- uint32\_t [throtrl\\_confidence](#)
- double **[avg\\_sendrate](#)**
- uint64\_t **[timestamp\\_msecs](#)**
- uint32\_t **[seconds](#)**
- uint32\_t **[microsec](#)**
- struct [vendor\\_frame](#) **[vendorframe](#)**

### 6.125.1 Detailed Description

Basic Safety Message structure is used for communicating BSM data with an application program. For more information about the BSM message elements, see SAE J2735 document. The BSM part2 starts from the events. The events are dictated based on vehicle CAN messages.

### 6.125.2 Field Documentation

#### 6.125.2.1 int abs

Antilock brake status. It is encode as per J2735. Look at BSMAbsActive\_t

#### 6.125.2.2 double angle

Steering wheel angle in degrees.

**6.125.2.3 uint32\_t autolightcontrol**

Auto light control on/off.

**6.125.2.4 uint32\_t auxbrakes**

Auxiliary brake status. It is encoded as per J2735.

**6.125.2.5 uint32\_t brakeboost**

Brake boost applied status. It is encoded as per J2735.

**6.125.2.6 double bumperheight\_front**

Front bumper height in meters.

**6.125.2.7 double bumperheight\_rear**

Rear bumper height in meters.

**6.125.2.8 double confidence**

Confidence value as percentage (between 0-100). This is part of path prediction.

**6.125.2.9 int32\_t count**

Path history element count 1 - 23

**6.125.2.10 uint32\_t dtimerunlights**

Day time running lights on/off.

**6.125.2.11 uint32\_t elev\_confidence**

Path history initial position elevation confidence - value as per J2735

**6.125.2.12 double elevation**

Elevation in meters.

**6.125.2.13 J2540\_2\_ITISCodes\_t emergency\_type\_event**

ITIS code describing the event

**6.125.2.14 uint32\_t event\_absactivate**

ABS activated event. this element is boolean

**6.125.2.15 uint32\_t event\_airbag**

Airbag activated event.

**6.125.2.16 J2540\_2\_ITISCodes\_t event\_descr[8]**

set of events that are related to the type event

**6.125.2.17 int event\_descr\_len**

Event description length

**6.125.2.18 uint32\_t event\_hazardbraking**

Hazard braking activated event. this element is boolean

**6.125.2.19 uint32\_t event\_hazardlights**

Hazard lights active event.

**6.125.2.20 uint32\_t event\_stabilityctrlactivated**

Stability control activated event. this element is boolean

**6.125.2.21 uint32\_t event\_tractionctrlloss**

Traction control loss event. this element is boolean

**6.125.2.22 uint32\_t foglights**

Fog lights on/off.

**6.125.2.23 uint32\_t gpsstatus**

Mode of fix available for the GPS. It is encode as per J2735.

**6.125.2.24 uint32\_t hazardlights**

Hazard lights on/off.

**6.125.2.25 uint32\_t head\_confidence**

Path history initial position heading confidence - value as per J2735

**6.125.2.26 double heading**

Heading of the vehicle in degrees.

**6.125.2.27 uint32\_t highbeam**

High beam light status. 1=on, 0=off.

**6.125.2.28 double lataccel**

Lateral acceleration in meters/sec<sup>2</sup>.

**6.125.2.29 double latitude**

Latitude in degrees.

**6.125.2.30 uint32\_t leftturnsignal**

Left turn signal light on/off.

**6.125.2.31 double length**

Vehicle length in cm.

**6.125.2.32 J2735\_2016\_LightBar\_t lightbar\_in\_use**

Lightbar in use flag for emergency vehicle

**6.125.2.33 uint32\_t lightbarinuse**

Light bar on/off.

**6.125.2.34 double longaccel**

Longitudinal acceleration in meters/sec<sup>2</sup>.

**6.125.2.35 double longitude**

Longitude in degrees.

**6.125.2.36 uint32\_t lowbeam**

Low beam light status. 1=on, 0=off.

**6.125.2.37 uint32\_t msgcount**

Message count. Value ranges from 0 to 127.

**6.125.2.38 uint32\_t msgid**

Message ID value. Must be set to 0x02 for BSM.

**6.125.2.39 uint32\_t parkinglights**

Parking lights on/off.

**6.125.2.40 struct pathhistoryframe phf[23]**

Set of upto 23 path history points.

**6.125.2.41 uint32\_t pos\_confidence**

Path history initial position position confidence - value as per J2735

**6.125.2.42 double positionalaccuracy[3]**

Positional accuracy. It is encoded as per J2735. 0 -> SemiMajorAxisAccuracy 1 -> SemiMinorAxisAccuracy 2 -> SemiMajorAxisOrientation

**6.125.2.43 double radius**

Radius of curvature in CM. this is part of path prediction

**6.125.2.44 uint32\_t rightturnsignal**

Right turn signal light on/off.

**6.125.2.45 int32\_t secmark**

Second mark (DSecond). Milliseconds in the current minute.

**6.125.2.46 double speed**

Speed of the vehicle in KPH.

**6.125.2.47 uint32\_t speed\_confidence**

Path history initial position speed confidence - value as per J2735

**6.125.2.48 int stabilitycontrol**

Stability control status. It is encode as per J2735. Look at BSMStabilityControlStatus\_t

**6.125.2.49 struct testdataframe tdf**

tdf - TestDataFrame info TDF is a CAMP proprietary blob at the end of BSM Part I Do not use it unless you are using CAMP specific devices

**6.125.2.50 int64\_t temp\_id**

Temporary ID.



**6.125.2.51 uint32\_t throtl\_confidence**

Path history initial position throttle position confidence. Not available in 2016 J2735.

**6.125.2.52 double throttle\_pos**

Throttle position as percentage (0-100). Not available in 2016 standard.

**6.125.2.53 uint32\_t time\_confidence**

Path history initial position time confidence - value as per J2735

**6.125.2.54 int traction**

Traction control state. It is encoded as per J2735. Look at BSMTractionControl\_t

**6.125.2.55 uint32\_t transmissionstate**

Transmission state (PRNDL) of the vehicle. It is encoded as per J2735. 0 - parking gear 1 - neutral gear 2 - reverse gear 3 - 9 - drive gear(s)

**6.125.2.56 struct utcTimeFrame utime**

Path history initial position time

**6.125.2.57 double vehicleheight**

Vehicle height in CM.

**6.125.2.58 int vehiclemass**

Vehicle mass in KG.

**6.125.2.59 uint32\_t vehicletype**

Vehicle type as per J2735.

**6.125.2.60 double vertaccel**

Vertical acceleration in meters/sec<sup>2</sup>.

**6.125.2.61 uint32\_t wheelbrake**

Wheel brake applied status. It is encoded as per J2735.

**6.125.2.62 uint32\_t wheelbrakeavailable**

Wheel brake status availability. Set to 1 if wheel brakes are available.

#### 6.125.2.63 double width

Vehicle width in cm.

#### 6.125.2.64 int wipers\_rtftnt

Wiper rate front value as per J2735.

#### 6.125.2.65 int wipers\_rtrear

Wiper rate rear value as per J2735.

#### 6.125.2.66 int wipers\_swfnt

Wiper status front: 0=off, 1-5=intermittent, 6=low, 7=high.

#### 6.125.2.67 int wipers\_swrear

Wiper status rear - 0:off, 1-5:intermittent, 6:low, 7:high

#### 6.125.2.68 double yawrate

Yaw rate in degrees per second.

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

- [v2x\\_msg\\_bsm.h](#)

## 6.126 v2x\_msg\_eva Struct Reference

```
#include <v2x_msg_eva.h>
```

### Data Fields

- [uint32\\_t tempid](#)
- [v2x\\_msg\\_rsa\\_t rsa\\_msg](#)
- [v2x\\_eva\\_response\\_type\\_t resp\\_type](#)
- [int emergency\\_details\\_avail](#)
- [V2XEvaEmergencyDetails\\_t emergency\\_details](#)

### 6.126.1 Detailed Description

EVA message

### 6.126.2 Field Documentation

#### 6.126.2.1 V2XEvaEmergencyDetails\_t emergency\_details

emergency details

#### 6.126.2.2 int emergency\_details\_avail

set this to 1, if the emergency\_details are being set

#### 6.126.2.3 v2x\_eva\_response\_type\_t resp\_type

response type

#### 6.126.2.4 v2x\_msg\_rsa\_t rsa\_msg

RSA message embedded into the EVA.. for the typeEvent

#### 6.126.2.5 uint32\_t tempid

temporary id

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

- [v2x\\_msg\\_eva.h](#)

## 6.127 v2x\_msg\_map Struct Reference

```
#include <v2x_msg_map.h>
```

### Data Fields

- int [msg\\_revision](#)
- int [n\\_geo\\_descr](#)
- [J2735\\_2016\\_IsecGeoDescr\\_t](#) \* [isec\\_geo\\_descr](#)

### 6.127.1 Detailed Description

MAP message structure

### 6.127.2 Field Documentation

#### 6.127.2.1 J2735\_2016\_IsecGeoDescr\_t\* isec\_geo\_descr

isec\_geo\_descr - intersection geometric description list

#### 6.127.2.2 int msg\_revision

msg\_revision - message revision

#### 6.127.2.3 int n\_geo\_descr

n\_geo\_descr - number of geographic descriptions

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

- [v2x\\_msg\\_map.h](#)

## 6.128 v2x\_msg\_psm Struct Reference

```
#include <v2x_msg_psm.h>
```

### Data Fields

- [personal\\_device\\_user\\_type\\_t](#) [personal\\_device\\_use\\_type](#)
- [uint32\\_t](#) [secmark](#)
- [uint32\\_t](#) [msgcount](#)
- [int64\\_t](#) [temp\\_id](#)
- [double](#) [latitude](#)
- [double](#) [longitude](#)
- [double](#) [elevation](#)
- [double](#) [pos\\_accuracy](#) [3]
- [double](#) [speed](#)
- [double](#) [heading](#)
- [double](#) [radius](#)
- [double](#) [confidence](#)
- [personal\\_device\\_usage\\_state\\_t](#) [device\\_usage\\_state](#)
- [uint32\\_t](#) [cross\\_request](#)
- [uint32\\_t](#) [cross\\_state](#)
- [uint32\\_t](#) [personal\\_cluster\\_radius](#)

### 6.128.1 Detailed Description

PSM data structure

### 6.128.2 Field Documentation

#### 6.128.2.1 double confidence

pp confidence

#### 6.128.2.2 uint32\_t cross\_request

cross request

set to 1 if cross is requested set to 0 if not

#### 6.128.2.3 personal\_device\_usage\_state\_t device\_usage\_state

device usage state

#### 6.128.2.4 double elevation

current device elevation

#### 6.128.2.5 double heading

heading of the device

## 6.128.2.6 double latitude

current device latitude

## 6.128.2.7 double longitude

current device longitude

## 6.128.2.8 uint32\_t msgcount

msg count

## 6.128.2.9 personal\_device\_user\_type\_t personal\_device\_use\_type

personal device user type

## 6.128.2.10 double pos\_accuracy[3]

posacc[0] - semi major axis posacc[1] - semi minor axis posacc[2] - heading angle

## 6.128.2.11 double radius

pp radius

## 6.128.2.12 uint32\_t secmark

dsecond

## 6.128.2.13 double speed

speed of the user

## 6.128.2.14 int64\_t temp\_id

tempid of the message

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

- [v2x\\_msg\\_psm.h](#)

## 6.129 v2x\_msg\_rtcn Struct Reference

```
#include <v2x_msg_rtcn.h>
```

### Data Fields

- V2X\_RTCNRevision\_t **rtcn\_rev\_no**
- int **no\_of\_msgs**
- uint8\_t **msgcnt**
- [J2735FullPosVector](#) **full\_pos**

- `uint8_t gpsstatus`
- `savari_antennaoffset_t antenna_offset`
- `savari_rtcml_payload_t * rtcml_payload`

### 6.129.1 Detailed Description

RTCM corrections structure

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

- `v2x_msg_rtcml.h`

## 6.130 v2x\_msg\_spat Struct Reference

```
#include <v2x_msg_spat.h>
```

### Data Fields

- `int minute_of_the_year`
- `int n_intersections`
- `J2735_2016_Intersection_t * Intersection_list`

### 6.130.1 Detailed Description

SPAT message structure

### 6.130.2 Field Documentation

#### 6.130.2.1 J2735\_2016\_Intersection\_t\* Intersection\_list

intersection\_list - intersection list

#### 6.130.2.2 int minute\_of\_the\_year

minute\_of\_the\_year - minute of the year

#### 6.130.2.3 int n\_intersections

n\_intersections - number of intersections (1 - 16)

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

- `v2x_msg_spat.h`

## 6.131 v2x\_msg\_tim Struct Reference

```
#include <v2x_msg_tim.h>
```

## Data Fields

- uint32\_t [message\\_id](#)
- int [msg\\_count](#)
- int [unique\\_id\\_len](#)
- char [unique\\_id](#) [18]
- uint32\_t [num\\_dataframes](#)
- [J2735DATAFRAME\\_t](#) \* [dataframes](#)
- struct wz\_hop\_info \* [hopinfo](#)
- uint16\_t [msg\\_crc](#)

### 6.131.1 Detailed Description

The TIM Message structure

The format of a tim message looks like this

```
+-----+
| TIM_MSG_ID | TIM_UNIQUE_ID | TIM_DATAFRAME_1 | TIM_DATAFRAME_2 | - - - | TIM_CRC |
+-----+
```

Each dataframe have its header, region and contents like the below

```
+-----+
| TIM_DF_HEAD | TIM_DF_REGIONS | TIM_DF_CONTENTS |
+-----+
```

Each TIM\_DF\_HEAD consists of one of two types, further info id or roadsign id. FURTHER\_INFO\_ID is two byte number or ROADSIGN\_ID is a dataframe as shown below

```
+-----+
| LAT | LONG | ELEV | HEADING_SLICE | MUTCDCODE |
+-----+
```

Each TIM\_DF\_REGION consists of a anchor point and heading slice and a extent field, and lastly a region field.

```
+-----+
| LAT | LONG | ELEV | HEADING_SLICE | EXTENT | REGION |
+-----+
```

Each REGION again will be one of three circle, polygon, and shapepoints. as of now this library supports only shapepoints we specify only shapepoints now.

Each shapepoint consists of a set of offsets with heading slice and a anchor point. The offsets added to the anchor point to get a shapepoint region

```
+-----+
| ANCHOR | HEADING_SLICE | LANEWIDTH | XOFF_1|YOFF_1|ZOFF_1| X_OFF2|Y_OFF2|Z_OFF2| --- | XOFF_N|YOFF_N|ZOFF_N
+-----+
```

Each and every content filed (For Ex. advisory\_codes, workzone\_codes, genericsign\_code and speedlimit\_codes) should follow a correct assignment of values.

```
Example:
for (i = 0; i < num_advisories; i++) {
    advisory_codes[i] = *user_configured_adv_code;
    user_configured_adv_code++;
}
```

where the user\_configured\_adv\_code is a uint32\_t pointer.

### 6.131.2 Field Documentation

#### 6.131.2.1 J2735DATAFRAME\_t\* dataframes

Dataframes in the TIM message

#### 6.131.2.2 struct wz\_hop\_info\* hopinfo

Optional workzone addition : for hopping

#### 6.131.2.3 uint32\_t message\_id

DSRC Message ID -> 0x10 for TIM

#### 6.131.2.4 uint16\_t msg\_crc

Message CRC. - obsolete, do not use this CRC field in 2016

#### 6.131.2.5 uint32\_t num\_dataframes

Number of dataframes that the TIM has. Min is 1 and max is 8.

#### 6.131.2.6 char unique\_id[18]

Packet ID must be a unique value

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

- [v2x\\_msg\\_tim.h](#)

## 6.132 v2x\_wme\_reg\_req Struct Reference

```
#include <v2x_net_wme.h>
```

### Data Fields

- int [channel](#)
- uint8\_t [srcmacaddr](#) [SAVARI\_MAC\_LEN]
- int [destmacaddr](#) [SAVARI\_MAC\_LEN]
- uint32\_t [psid](#)
- int [repeatrate](#)
- int [priority](#)
- int [request\\_type](#)
- int [extended\\_access](#)
- int [channel\\_access](#)
- int [immediate\\_access](#)
- int [wsatype](#)
- char [psc](#) [32]
- int [psc\\_length](#)
- int [local\\_service\\_index](#)
- int [ipservice](#)



- struct in6\_addr [service\\_ipv6addr](#)
- int [service\\_port](#)
- int [secondradio](#)
- char **advertiser\_id** [[ADVERTISER\\_LEN](#)]
- int **linkquality**

### 6.132.1 Detailed Description

This structure is used to register an application with WME service.

### 6.132.2 Field Documentation

#### 6.132.2.1 int channel

should be either service channel(SC) or continuous channel(CC)

The WAVE stack support channels ranging from 172 to 184

#### 6.132.2.2 int channel\_access

Provider channel switching mode

One of LIBWME\_CHANNEL\_ACCESS\_CONTINUOUS(non channel switching, stay on channel),

LIBWME\_CHANNEL\_ACCESS\_ALTERNATING(forced/conditional switching between 178 and channel)

#### 6.132.2.3 int destmacaddr[SAVARI\_MAC\_LEN]

destination macaddress to which WSAs should be sent

#### 6.132.2.4 int extended\_access

set to 0xffff for continuous access otherwise 0.

#### 6.132.2.5 int immediate\_access

this indicates the device should immediately switch to SCH, rather than waiting for the next SCH interval (0/1)

#### 6.132.2.6 int local\_service\_index

index to the associated MIB table or internal datastructure. Must be unique for a give PSID and psc combination - not used as of now

#### 6.132.2.7 int priority

priority of Provider/User

#### 6.132.2.8 char psc[32]

provider service context - ignore for WSMP

**6.132.2.9 int psc\_length**

provider service context length - ignore for WSMP

**6.132.2.10 uint32\_t psid**

psid - provider service identifier

PSID is used to differentiate between different safety/roadside application and messages

**6.132.2.11 int repeatrate**

the number of WSAs transmitted for 5sec. Ignore for WSMP traffic.

**6.132.2.12 int request\_type**

type of user application request.

LIBWME\_USER\_AUTOACCESS\_ONMATCH (Switch Between 178 and SCH after receiving Matching WSA from RSE)

LIBWME\_USER\_AUTOACCESS\_UNCOND (Start Switching between 178 and SCH Without Waiting for a Matching WSA from RSEs)

In case of LIBWME\_USER\_AUTOACCESS\_UNCOND set extended\_access to 0xffff for a prolonged continuous mode of operation

LIBWME\_USER\_AUTOACCESS\_NOSCHACCESS(CCH Only Mode. No Switching) Only applicable if channel\_access is ALTERNATING

**6.132.2.13 int secondradio**

for doing registration of first radio set it to 0 and for the second radio set it to 1.

If second radio is set to 1. The service is going to get the channel access on the radio 1. The radio 0 is a free running radio and runs in 178 continuous hunting for the WSA.

When second radio is set to 1, the application can access the channel using the ONMATCH configuration. If the second radio is set to 0, the application will wait for the WSAs and looks for the services of interest.

**6.132.2.14 struct in6\_addr service\_ipv6addr**

service IPv6 address; memset to 0 if not used - ignore for WSMP

**6.132.2.15 int service\_port**

port on which service is provided; memset to 0 if not used - ignore for WSMP

**6.132.2.16 uint8\_t srcmacaddr[SAVARI\_MAC\_LEN]**

source mac address

**6.132.2.17 int wsatype**

secured (SAVARI1609\_WSA\_SECURED) or unsecured (SAVARI1609\_WSA\_UNSECURED) WSA - ignore for WSMP

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

- [v2x\\_net\\_wme.h](#)

## 6.133 v2x\_wsmp\_callbacks Struct Reference

```
#include <v2x_net_wme.h>
```

### Data Fields

- void(\* [wme\\_provider\\_confirm](#) )(void \*ctx, int conf\_result\_indication)  
*indicates to the application about the provider registration Confirmation*
- void(\* [wme\\_user\\_confirm](#) )(void \*ctx, int conf\_result\_indication)  
*indicates to the application about the user registration \ - confirmation*
- void(\* [wme\\_cch\\_confirm](#) )(void \*ctx, int conf\_result\_indication)  
*confirm message about the cch registration*
- void(\* [wme\\_wsm\\_indication](#) )(void \*ctx, struct [savariwme\\_rx\\_indication](#) \*ind)  
*indicates to the application about the WSM reception.*
- void(\* [wme\\_get\\_wme\\_err](#) )(void \*ctx, int length, void \*data, int cmd\_indication)
- void(\* [wme\\_clear\\_wme\\_err](#) )(void \*ctx, int length, void \*data, int cmd\_indication)
- void(\* [wme\\_get\\_rse\\_info](#) )(void \*ctx, int length, void \*data, int cmd\_indication)  
*get a list of RSU information based on the WSAs*
- void(\* [wme\\_get\\_avail\\_services](#) )(void \*ctx, int length, void \*data, int cmd\_indication)
- void(\* [wme\\_get\\_connected\\_rse\\_cnt](#) )(void \*ctx, int length, void \*data, int cmd\_indication)
- void(\* [wme\\_scanning](#) )(void \*ctx, int length, void \*data)
- void(\* [wme\\_connected](#) )(void \*ctx, int length, void \*data)  
*connection success notification*
- void(\* [wme\\_disconnected](#) )(void \*ctx, int length, void \*data)  
*disconnection success notification*
- void(\* [wme\\_service\\_preempted](#) )(void \*ctx, int length, void \*data)
- void(\* [get\\_wsa\\_cnt](#) )(void \*ctx, uint32\_t wsa\_cnt)
- void(\* [get\\_wsa\\_status](#) )(void \*ctx, int status)
- void(\* [get\\_radio\\_config](#) )(void \*ctx, int [channel\\_access](#), int [channel](#), int wsa\_iface)
- void(\* [get\\_radio\\_settings](#) )(void \*ctx, struct [libwme\\_radio\\_settings](#) \*radio\_settings)

### 6.133.1 Detailed Description

This structure contains a set of callbacks associated with the application about the indication of WSMs/commands etc.

This is set to a list of function pointers, and they will be called from the library based on the communication protocol type between the caller and the 1609.3. The protocol sends a confirmation upon a call to `wme_register_user(provider)` request. the confirmation callback `wme_user(provider)_confirm` is called (if its a valid pointer) and the application can decide to transmit/receive WSMP.

The protocol sends a WSMP decoded packet and fills into the `savariwme_rx_indication` and callback `wme_wsm_indication` is called.

### 6.133.2 Field Documentation

#### 6.133.2.1 void(\* [wme\\_cch\\_confirm](#) )(void \*ctx, int conf\_result\_indication)

confirm message about the cch registration

**Parameters**

<i>ctx</i>	- context of the application
<i>conf_result_indication</i>	- result of the cch registration. 0 - success non-zero : failure

**6.133.2.2 void(\* wme\_connected)(void \*ctx, int length, void \*data)**

connection success notification

**Parameters**

<i>ctx</i>	- application ctx
<i>length</i>	- length of the data
<i>data</i>	- data is of type struct savari1609NotificationIndication

**Description -**

This function is an indication callback called from the 1609.3 daemon upon a connection success for a requested RSU based on the WSA.

**6.133.2.3 void(\* wme\_disconnected)(void \*ctx, int length, void \*data)**

disconnection success notification

**Parameters**

<i>ctx</i>	- application ctx
<i>length</i>	- length of the data
<i>data</i>	- data is of type struct savari1609NotificationIndication

**Description -**

This function is an indication callback called from the 1609.3 daemon upon a disconnection success for a requested RSU w.r.t WSA timeout / user request.

**6.133.2.4 void(\* wme\_get\_rse\_info)(void \*ctx, int length, void \*data, int cmd\_indication)**

get a list of RSU information based on the WSAs

**Parameters**

<i>ctx</i>	- application ctx
<i>length</i>	- length of the data
<i>data</i>	- data is of type struct savari1609RSE_Node_Info
<i>cmd_indication</i>	- command name

**Description -**

This function indicates the list of RSU information based on the WSAs.

**6.133.2.5 void(\* wme\_provider\_confirm)(void \*ctx, int conf\_result\_indication)**

indicates to the application about the provider registration Confirmation

## Parameters

<i>ctx</i>	- context of the application
<i>conf_result_ - indication</i>	- result of the wme_register_provider. 0 - success non-zero : failure

## 6.133.2.6 void(\* wme\_user\_confirm)(void \*ctx, int conf\_result\_indication)

indicates to the application about the user registration \ - confirmation

## Parameters

<i>ctx</i>	- context of the application
<i>conf_result_ - indication</i>	- result of the wme_register_user. 0 - success non-zero : failure

## 6.133.2.7 void(\* wme\_wsm\_indication)(void \*ctx, struct savariwme\_rx\_indication \*ind)

indicates to the application about the WSM reception.

## Parameters

<i>ctx</i>	- application ctx.
<i>ind</i>	- rx indication buffer about the rxdatalen, psid, datarate etc.

## Description -

This function indicates to the application about the WSM packet matching packet based on PSID. and returns in the buffer ,ind->rx\_buf.

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

- [v2x\\_net\\_wme.h](#)

## 6.134 v2x\_wsmp\_rcv\_indication Struct Reference

```
#include <v2x_net_wme.h>
```

## Data Fields

- int [version](#)
- uint64\_t [tstamp](#)
- int [plcp\\_length](#)
- int [txpower](#)
- int [datarate](#)
- uint8\_t [rx\\_mac](#) [SAVARI\_MAC\_LEN]
- uint32\_t [rx\\_buf\\_length](#)
- uint32\_t [rx\\_supp](#)
- int [channel](#)
- uint8\_t [psid](#) [PSID\_LEN]
- int [priority](#)
- int [rssi](#)
- int [num\\_rx](#)
- uint8\_t [rx\\_buf](#) [MAX\_DATA]

### 6.134.1 Detailed Description

This structure used to indicate received WSMP packets to a higher layer entity.

### 6.134.2 Field Documentation

#### 6.134.2.1 int channel

channel on which the message was received

#### 6.134.2.2 int datarate

datarate

#### 6.134.2.3 int num\_rx

number of received bytes

#### 6.134.2.4 int plcp\_length

plcp length

#### 6.134.2.5 int priority

priority at which the packet received

#### 6.134.2.6 uint8\_t psid[PSID\_LEN]

Provider Service Identifier

#### 6.134.2.7 int rssi

received signalstrength indication

#### 6.134.2.8 uint8\_t rx\_buf[MAX\_DATA]

received buffer

#### 6.134.2.9 uint32\_t rx\_buf\_length

received bufferlen

#### 6.134.2.10 uint8\_t rx\_mac[SAVARI\_MAC\_LEN]

received mac

#### 6.134.2.11 uint32\_t rx\_supp

received WSMP supplement

## 6.134.2.12 uint64\_t tstamp

timestamp at which packet got received

## 6.134.2.13 int txpower

transmitted power

## 6.134.2.14 int version

WAVE version number

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

- [v2x\\_net\\_wme.h](#)

## 6.135 v2x\_wsmp\_send\_req Struct Reference

```
#include <v2x_net_wme.h>
```

### Data Fields

- int [channel](#)
- uint32\_t [psid](#)
- int [priority](#)
- int [datarate](#)
- int [txpower](#)
- uint8\_t [mac](#) [SAVARI1609\_IEEE80211\_ADDR\_LEN]
- uint8\_t [srcmac](#) [SAVARI1609\_IEEE80211\_ADDR\_LEN]
- int [expiry\\_time](#)
- int [element\\_id](#)
- int [tx\\_length](#)
- int [supp\\_enable](#)
- uint32\_t [safetysupp](#)
- uint32\_t [header\\_ext](#)

### 6.135.1 Detailed Description

This structure contains the configuration parameters for a message to be transmitted through 1609.3 stack.

### 6.135.2 Field Documentation

## 6.135.2.1 int channel

channel of transmission of WSMs/WSMPs.

## 6.135.2.2 int datarate

datarate

#### 6.135.2.3 int element\_id

WAVE element id set to WAVE\_ELEMID\_WSMP for WSMP set to WAVE\_ELEMID\_WSMPS for WSMPS

#### 6.135.2.4 int expiry\_time

indicates the time at which the message is no longer valid

#### 6.135.2.5 uint32\_t header\_ext

WSM header extention

#### 6.135.2.6 uint8\_t mac[SAVARI1609\_IEEE80211\_ADDR\_LEN]

destination mac address

#### 6.135.2.7 int priority

Qos for Packet

#### 6.135.2.8 uint32\_t psid

Proivder Service Identifier

#### 6.135.2.9 uint32\_t safetysupp

WSM safety supplement

#### 6.135.2.10 uint8\_t srcmac[SAVARI1609\_IEEE80211\_ADDR\_LEN]

source mac address .. place holder and does not perform any task

#### 6.135.2.11 int supp\_enable

sup\_enable when 1 reads and transmits safetysupp when 0 it doesn't

#### 6.135.2.12 int tx\_length

tx buffer length

#### 6.135.2.13 int txpower

transmit power

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

- [v2x\\_net\\_wme.h](#)



## 6.136 vendor\_frame Struct Reference

```
#include <v2x_msg_bsm.h>
```

### Data Fields

- int **vendorframelength**
- uint8\_t **vendorframe** [512]

### 6.136.1 Detailed Description

unused and internals to the library.. please do not use

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

- [v2x\\_msg\\_bsm.h](#)



# Chapter 7

## File Documentation

### 7.1 v2x\_canctl.h File Reference

```
#include <stdint.h>
#include <stdio.h>
#include "v2x_error.h"
```

#### Data Structures

- struct [libcan\\_filters](#)
- struct [v2x\\_canctl](#)

#### Macros

- #define **LIBCAN\_FILTERS\_LEN** 64
- #define **canctl** [v2x\\_canctl](#)

#### Enumerations

- enum **canmsg\_type** { **CANMSG\_STANDARD**, **CANMSG\_EXTENDED** }

#### Functions

- [v2x\\_status\\_t](#) **v2x\_libpcan\_net\_init** (struct [v2x\\_canctl](#) \*ctl)
- [v2x\\_status\\_t](#) **v2x\_libpcan\_net\_recv** (struct [v2x\\_canctl](#) \*ctl, int \*can\_id, int \*can\_dlc, uint8\_t \*data, int size)
- void **v2x\_libpcan\_net\_deinit** (struct [v2x\\_canctl](#) \*ctl)

#### 7.1.1 Detailed Description

create a connection to the CAN interface

##### Parameters

<i>in</i>	<i>ctl</i>	The ctl control structure
-----------	------------	---------------------------

##### Description

This API is used to create connection with the CAN interface The file descriptor is set into the can\_fd variable of the ctl.

**Returns**

V2X\_SUCCESS on success and error on failure

**Return values**

V2X_SUCCESS	API is successful
V2X_EINVAL	invalid argument passed
V2X_EFAULT	NULL ctl pointer passed
V2X_EIO	communication with the CAN driver failed

**7.2 v2x\_debug.h File Reference**

V2X debug levels and related functions.

```
#include "v2x_error.h"
```

**Enumerations**

- enum **v2x\_debug\_t** {  
**V2X\_EMERG** = 0, **V2X\_ALERT** = 1,  
**V2X\_CRIT** = 2, **V2X\_ERR** = 3,  
**V2X\_WARNING** = 4, **V2X\_NOTICE** = 5,  
**V2X\_INFO** = 6, **V2X\_DEBUG** = 7 }

**Functions**

- [v2x\\_status\\_t v2x\\_debug\\_level\\_set](#) (int dbg\_level)
- void [v2x\\_log](#) (int log\_level, const char \*fmt,...)

**7.2.1 Detailed Description**

V2X debug levels and related functions. Include libj2735.h. Link with -lj2735 -lsae -lm.

**7.3 v2x\_error.h File Reference**

V2X Status values, error codes and related functions.

```
#include "stddef.h"
```

**Enumerations**

- enum [v2x\\_status\\_t](#) {  
[V2X\\_SUCCESS](#) = 0, [V2X\\_EINVAL](#) = -1,  
[V2X\\_ENOMEM](#) = -2, [V2X\\_EFAULT](#) = -3,  
[V2X\\_EENCODE](#) = -4, [V2X\\_EDECODE](#) = -5,  
[V2X\\_EACCES](#) = -6, [V2X\\_EIO](#) = -7,  
[V2X\\_PEER\\_CLOSE](#) = -8, [V2X\\_ECONN\\_REFUSED](#) = -9,  
[V2X\\_ELENGTH](#) = -10, [V2X\\_ECHANNEL](#) = -11 }

## Functions

- [v2x\\_status\\_t v2x\\_strerror](#) ([v2x\\_status\\_t](#) errnum, char \*buf, size\_t buflen)

### 7.3.1 Detailed Description

V2X Status values, error codes and related functions. This header file contains the APIs and data structures that are common to rest of V2X library modules.

Include libj2735.h. Link with -lj2735 -lsae -lm.

## 7.4 v2x\_msg\_bsm.h File Reference

BSM message API definitions and functions.

```
#include <stdio.h>
#include <string.h>
#include <stdint.h>
#include <stdlib.h>
#include <unistd.h>
#include <time.h>
#include <sys/time.h>
#include "asndefines.h"
#include "v2x_msg_common.h"
#include "v2x_error.h"
```

## Data Structures

- struct [timestamp](#)
- struct [testdataframe](#)
- struct [pathhistoryframe](#)
- struct [enctimestamp](#)
- struct [encpadbytes](#)
- struct [vendor\\_frame](#)
- struct [v2x\\_msg\\_bsm](#)

## Macros

- #define [ASN1\\_BSM\\_MSG\\_ID](#) 20
- #define [BSM\\_BLOBSIZE](#) 38
- #define [TDF\\_FORMATVERSION](#) 1
- #define [TDF\\_TESTCHECK](#) 1
- #define [CAMP\\_TSTCONF\\_UNAVAIL](#) 0x80000000
- #define [CAMP\\_TSTCHMODE\\_INVALID](#) 0x80000000
- #define [CAMP\\_TSTDATARATE\\_INVALID](#) 0x80000000
- #define [CAMP\\_TSTMSGRATE\\_INVALID](#) 0x80000000
- #define [CAMP\\_TSTNUMOBE\\_INVALID](#) 0x80000000
- #define [CAMP\\_TSTTESTRUN\\_INVALID](#) 0x80000000
- #define [CAMP\\_TDF\\_TIMESTAMP\\_M](#) 0x0000FFFFFFFFFFFFFFFULL
- #define [TDF\\_FORMATVERSION\\_M](#) 0x7
- #define [TDF\\_FORMATVERSION\\_S](#) 5
- #define [TDF\\_TESTCHECK\\_M](#) 0x1
- #define [TDF\\_TESTCHECK\\_S](#) 0x4

- `#define TDF_TESTCONFIG_M 0x7F`
- `#define TDF_TESTCONFIG_S 1`
- `#define TDF_DATARATE_M 0x3`
- `#define TDF_DATARATE_S 5`
- `#define TDF_MSGRATE_M 0x7`
- `#define TDF_MSGRATE_S 0x2`
- `#define BSM_LATITUDE_MAX 90.0`
- `#define BSM_LATITUDE_MIN -90.0`
- `#define BSM_LONGITUDE_MAX 180.0`
- `#define BSM_LONGITUDE_MIN -180.0`
- `#define bsm v2x_msg_bsm`
- `#define bsm_t v2x_msg_bsm_t`

## Typedefs

- `typedef struct v2x_msg_bsm v2x_msg_bsm_t`

## Enumerations

- `enum BSM_TYPE {`  
`BSM_PART1 = 0x1, BSM_PATHPREDICTION = 0x2,`  
`BSM_PATHHISTORY = 0x4, BSM_VEHICLESTATUS = 0x8,`  
`BSM_TESTDATAFRAME = 0x10, BSM_TDF_UNENCODED = 0x20,`  
`BSM_VENDORFRAME = 0x40 }`
- `enum BSMBrakeAppliedStatus_t {`  
`BSM_WHEELBRK_UNAVAIL = 0x01, BSM_WHEELBRK_LEFTFRNT = 0x2,`  
`BSM_WHEELBRK_LEFTREAR = 0x4, BSM_WHEELBRK_RIGHTFRNT = 0x8,`  
`BSM_WHEELBRK_RIGHTREAR = 0x10, BSM_WHEELBRK_ALL_ON = 0x1E }`
- `enum BSMAbsActive_t {`  
`BSM_ABS_ACTIVE_UNAVAIL = 0, BSM_ABS_ACTIVE_OFF,`  
`BSM_ABS_ACTIVE_ON, BSM_ABS_ACTIVE_ENGAGED }`
- `enum BSMStabilityControlStatus_t {`  
`BSM_SCS_UNAVAIL = 0, BSM_SCS_OFF,`  
`BSM_SCS_ON, BSM_SCS_ENGAGED }`
- `enum BSMTractionControl_t {`  
`BSM_TRC_UNAVAIL = 0, BSM_TRC_OFF,`  
`BSM_TRC_ON, BSM_TRC_ENGAGED }`
- `enum BSMWiperStatus_t {`  
`BSM_WIPERS_UNAVAIL = 0, BSM_WIPERS_OFF,`  
`BSM_WIPERS_INTERMITTENT, BSM_WIPERS_LOW,`  
`BSM_WIPERS_HIGH, BSM_WIPERS_WASHER_IN_USE,`  
`BSM_WIPERS_AUTOMATIC_PRESENT }`
- `enum BSMSirenInUse_t {`  
`Siren_unavailable = 0, Siren_notInUse,`  
`Siren_inUse, Siren_reserved }`

## Functions

- `struct enctimestamp __attribute__((packed))`
- `int J2735_Encode_full_pos_vector (J2735FullPosVector *pos, int valid_fix, char *buff, int buflen)`
- `int J2735_Decode_full_pos_vector (J2735FullPosVector *pos, char *buff, int buflen)`
- `void v2x_msg_bsm_init (v2x_msg_bsm_t *bsm)`
- `void j2735_reset_bsm (bsm_t *bsm)`
- `v2x_status_t v2x_msg_bsm_encode (v2x_msg_bsm_t *bsm, BSM_TYPE type, unsigned char *msgbuf, int msgbuf_size, int *enclen)`

- int **j2735\_encode\_bsm** (bsm\_t \*bsm, [BSM\\_TYPE](#) type, char \*encoded\_buffer, int msgbuf\_size)
- [v2x\\_status\\_t](#) **v2x\_msg\_bsm\_decode** ([v2x\\_msg\\_bsm\\_t](#) \*bsm, unsigned char \*msgbuf, int msgbuf\_len, [BSM\\_TYPE](#) \*type, uint32\_t \*oob, int debug)
- int **j2735\_decode\_bsm** (bsm\_t \*bsm, char \*decode\_buffer, int msgbuf\_len, [BSM\\_TYPE](#) \*type, uint32\_t \*oob)
- void **j2735\_2016\_print\_bsm** ([v2x\\_msg\\_bsm\\_t](#) \*bsm, FILE \*fp)

## Variables

- unsigned int **reserved**
- unsigned int **unused**
- unsigned int **microsec**
- unsigned int **second**
- uint16\_t **pad**
- struct [vendor\\_frame](#) **\_\_attribute\_\_**

### 7.4.1 Detailed Description

BSM message API definitions and functions. This header file contains the APIs and data structures used to encode or decode a BSM.

Include libj2735.h. Link with -lj2735\_2016 -lsae\_2016 -losstoed -lm.

### 7.4.2 Macro Definition Documentation

#### 7.4.2.1 #define ASN1\_BSM\_MSG\_ID 20

BSM message-ID as defined in ASN1 j2735-2016 document

### 7.4.3 Enumeration Type Documentation

#### 7.4.3.1 enum BSM\_TYPE

The elements type in the BSM

There are two parts in a BSM. Part 1 & Part 2. BSM\_TYPE flag is used for specifying BSM elements present in the BSM structure.

#### Enumerator

- BSM\_PART1** BSM part-1 elements are present
- BSM\_PATHPREDICTION** BSM path prediction elements are present
- BSM\_PATHHISTORY** BSM path history elements are present
- BSM\_VEHICLESTATUS** BSM vehicle status elements are present
- BSM\_TESTDATAFRAME** TDF frame, attached before to the BSM - CAMP proprietary
- BSM\_TDF\_UNENCODED** Unencoded TDF frame - CAMP proprietary

#### 7.4.3.2 enum BSMAbsActive\_t

BSMAbsActive - ABS active flag

#### Enumerator

- BSM\_ABS\_ACTIVE\_UNAVAIL** unavailable and not present

***BSM\_ABS\_ACTIVE\_OFF*** off, not active  
***BSM\_ABS\_ACTIVE\_ON*** on, active  
***BSM\_ABS\_ACTIVE\_ENGAGED*** ABS engaged

#### 7.4.3.3 enum BSMBrakeAppliedStatus\_t

BrakeAppliedStatus - Wheel brakeAct

#### 7.4.3.4 enum BSMStabilityControlStatus\_t

BSMStabilityControlStatus\_t - stability control status flag

Enumerator

***BSM\_SCS\_UNAVAIL*** unavailable and not present  
***BSM\_SCS\_OFF*** off, not active  
***BSM\_SCS\_ON*** on, active  
***BSM\_SCS\_ENGAGED*** SCS engaged

#### 7.4.3.5 enum BSMTractionControl\_t

BSMTractionControl\_t - Traction control status flag

Enumerator

***BSM\_TRC\_UNAVAIL*** unavailable and not present  
***BSM\_TRC\_OFF*** off, not active  
***BSM\_TRC\_ON*** on, active  
***BSM\_TRC\_ENGAGED*** TRC engaged

#### 7.4.3.6 enum BSMWiperStatus\_t

BSMWiperStatus\_t - Wiper status flag

Enumerator

***BSM\_WIPERS\_UNAVAIL*** unavailable and not present  
***BSM\_WIPERS\_OFF*** off, not active  
***BSM\_WIPERS\_INTERMITTENT*** intermittent  
***BSM\_WIPERS\_LOW*** low wipers  
***BSM\_WIPERS\_HIGH*** high wipers  
***BSM\_WIPERS\_WASHER\_IN\_USE*** washer in use  
***BSM\_WIPERS\_AUTOMATIC\_PRESENT*** Automatically active

## 7.5 v2x\_msg\_cam.h File Reference

CAM data APIs and data structures.

```
#include "etsi_pathhistory.h"
#include "etsi_its_decl.h"
#include <v2x_error.h>
```



## Data Structures

- struct [v2x\\_etsi\\_cam\\_hf\\_vehicle](#)
- struct [v2x\\_etsi\\_protected\\_zone](#)
- struct [v2x\\_etsi\\_cam\\_hf\\_rsu](#)
- struct [v2x\\_etsi\\_cam\\_lf](#)
- struct [v2x\\_etsi\\_cam\\_hf](#)
- struct [v2x\\_etsi\\_cam\\_bc](#)
- struct [etsi\\_cam\\_ptc](#)
- struct [etsi\\_cam\\_stc](#)
- struct [etsi\\_cam\\_dgc](#)
- struct [etsi\\_cam\\_rwc\\_basic](#)
- struct [etsi\\_cam\\_rc](#)
- struct [etsi\\_cam\\_ec](#)
- struct [etsi\\_cam\\_scc](#)
- struct [v2x\\_etsi\\_cam](#)

## Macros

- `#define etsi\_cam\_hf\_vehicle v2x\_etsi\_cam\_hf\_vehicle`
- `#define ETSI\_PROTECTED\_RSU\_ZONES\_MIN 1`
- `#define ETSI\_PROTECTED\_RSU\_ZONE\_LEN 16`
- `#define ETSI\_PROTECTED\_ZONE\_TYPE\_NAV ETSI\_NAV`
- `#define ETSI\_PROTECTED\_ZONE\_RADIUS\_MIN 1`
- `#define ETSI\_PROTECTED\_ZONE\_RADIUS\_MAX UINT\_MAX`
- `#define ETSI\_PROTECTED\_ZONE\_RADIUS\_NAV ETSI\_NAV`
- `#define ETSI\_PROTECTED\_ZONE\_ID\_MIN 0`
- `#define ETSI\_PROTECTED\_ZONE\_ID\_MAX 134217727`
- `#define ETSI\_PROTECTED\_ZONE\_ID\_NAV ETSI\_NAV`
- `#define etsi\_protected\_zone v2x\_etsi\_protected\_zone`
- `#define etsi\_cam\_hf\_rsu v2x\_etsi\_cam\_hf\_rsu`
- `#define etsi\_cam\_lf v2x\_etsi\_cam\_lf`
- `#define ETSI\_CAM\_HF\_VEHICLE 1`
- `#define ETSI\_CAM\_HF\_RSU 2`
- `#define etsi\_cam\_hf v2x\_etsi\_cam\_hf`
- `#define etsi\_cam\_bc v2x\_etsi\_cam\_bc`
- `#define ETSI\_CAM\_PT\_ACTIVATION\_UNAVAIL ETSI\_NAV`
- `#define ETSI\_CAM\_PT\_ACTIVATION\_TYPE\_MIN 0`
- `#define ETSI\_CAM\_PT\_ACTIVATION\_TYPE\_MAX 255`
- `#define ETSI\_CAM\_PT\_ACTIVATION\_DATA\_LEN 20`
- `#define ETSI\_CAM\_EMBARKATION\_STATUS\_UNAVAIL ETSI\_NAV`
- `#define etsi\_cam v2x\_etsi\_cam`
- `#define V2X\_ETSI\_SET\_BC(__cam) (__cam->type |= ETSI\_BC\_AVAILABLE)`
- `#define V2X\_ETSI\_SET\_HF(__cam) (__cam->type |= ETSI\_HF\_AVAILABLE)`
- `#define V2X\_ETSI\_SET\_LF(__cam) (__cam->type |= ETSI\_LF\_AVAILABLE)`
- `#define V2X\_ETSI\_SET\_SV(__cam) (__cam->type |= ETSI\_SV\_AVAILABLE)`
- `#define ETSI\_SET\_BC V2X\_ETSI\_SET\_BC`
- `#define ETSI\_SET\_HF V2X\_ETSI\_SET\_HF`
- `#define ETSI\_SET\_LF V2X\_ETSI\_SET\_LF`
- `#define ETSI\_SET\_SV V2X\_ETSI\_SET\_SV`
- `#define V2X\_ETSI\_BC\_PRESENT(__cam) (!!(__cam->type & ETSI\_BC\_AVAILABLE))`
- `#define V2X\_ETSI\_HF\_PRESENT(__cam) (!!(__cam->type & ETSI\_HF\_AVAILABLE))`
- `#define V2X\_ETSI\_LF\_PRESENT(__cam) (!!(__cam->type & ETSI\_LF\_AVAILABLE))`
- `#define V2X\_ETSI\_SV\_PRESENT(__cam) (!!(__cam->type & ETSI\_SV\_AVAILABLE))`
- `#define ETSI\_BC\_PRESENT V2X\_ETSI\_BC\_PRESENT`
- `#define ETSI\_HF\_PRESENT V2X\_ETSI\_HF\_PRESENT`
- `#define ETSI\_LF\_PRESENT V2X\_ETSI\_LF\_PRESENT`
- `#define ETSI\_SV\_PRESENT V2X\_ETSI\_SV\_PRESENT`

## Typedefs

- typedef struct [v2x\\_etsi\\_cam](#) [v2x\\_msg\\_cam\\_t](#)

## Enumerations

- enum {  
**ETSI\_BC\_AVAILABLE** = 0x01, **ETSI\_HF\_AVAILABLE** = 0x02,  
**ETSI\_LF\_AVAILABLE** = 0x04, **ETSI\_SV\_AVAILABLE** = 0x08 }
- enum **ETSIProtectedZoneType\_t** { **PROTECTED\_ZONE\_PERMANENT**, **PROTECTED\_ZONE\_TEMPORARY** }
- enum **ETSIspecialContainerType\_t** {  
**ETSI\_CAM\_SV\_PTC** = 1, **ETSI\_CAM\_SV\_STC**,  
**ETSI\_CAM\_SV\_DGC**, **ETSI\_CAM\_SV\_RWC\_BASIC**,  
**ETSI\_CAM\_SV\_RC**, **ETSI\_CAM\_SV\_EC**,  
**ETSI\_CAM\_SV\_SCC**, **ETSI\_CAM\_SV\_UNAVAIL** }
- enum **ETSIptActivationType\_t** {  
**ETSI\_PT\_ACTIVATION\_TYPE\_UNDEFINED\_CODING** = 0, **ETSI\_PT\_ACTIVATION\_TYPE\_R09\_16\_CODING**,  
**ETSI\_PT\_ACTIVATION\_TYPE\_VDV\_50149\_CODING** }
- enum **ETSIStcType\_t** {  
**ETSI\_STC\_TYPE\_HEAVY\_LOAD** = 0x01, **ETSI\_STC\_TYPE\_EXCESS\_WIDTH** = 0x02,  
**ETSI\_STC\_TYPE\_EXCESS\_LENGTH** = 0x04, **ETSI\_STC\_TYPE\_EXCESS\_HEIGHT** = 0x08 }

## Functions

- [v2x\\_status\\_t](#) [v2x\\_etsi\\_cam\\_encode](#) (struct [v2x\\_etsi\\_cam](#) \*cam, uint8\_t \*encbuf, int buf\_len, int \*encbuf\_len)
- int [etsi\\_cam\\_encode](#) (struct [etsi\\_cam](#) \*cam, uint8\_t \*tbebuf, int len)
- [v2x\\_status\\_t](#) [v2x\\_etsi\\_cam\\_decode](#) (struct [v2x\\_etsi\\_cam](#) \*cam, uint8\_t \*encbuf, int encbuf\_len)
- int [etsi\\_cam\\_decode](#) (struct [etsi\\_cam](#) \*cam, uint8\_t \*tbebuf, int len)
- void [etsi\\_cam\\_reset\\_lf](#) (struct [etsi\\_cam\\_lf](#) \*lf)
- void [etsi\\_cam\\_reset\\_bc](#) (struct [etsi\\_cam\\_bc](#) \*bc)
- void [etsi\\_cam\\_reset\\_hf\\_vehicle](#) (struct [etsi\\_cam\\_hf\\_vehicle](#) \*veh)
- void [v2x\\_etsi\\_cam\\_reset](#) (struct [v2x\\_etsi\\_cam](#) \*cam)
- void [etsi\\_cam\\_reset](#) (struct [etsi\\_cam](#) \*cam)
- void [etsi\\_cam\\_reset\\_sv](#) (struct [etsi\\_cam](#) \*cam)
- void [v2x\\_etsi\\_cam\\_dump\\_hex](#) (uint8\_t \*encbuf, int encbuf\_len)
- void [v2x\\_etsi\\_cam\\_print](#) (FILE \*fp, struct [v2x\\_etsi\\_cam](#) \*cam)

### 7.5.1 Detailed Description

CAM data APIs and data structures. This header file contains the APIs and data structures to encode/decode CAM  
 Include libetsi.h and link with -letsi

### 7.5.2 Macro Definition Documentation

7.5.2.1 **#define V2X\_ETSI\_BC\_PRESENT( \_\_cam ) (!(\_\_cam->type & ETSI\_BC\_AVAILABLE))**

Macros to check if a container is present

7.5.2.2 **#define V2X\_ETSI\_SET\_BC( \_\_cam ) (\_\_cam->type |= ETSI\_BC\_AVAILABLE)**

Macros to set the bit to indicate presence of a container

## 7.6 v2x\_msg\_common.h File Reference

```
#include <stdint.h>
#include <syslog.h>
#include "J2735_201603DA.h"
#include "ossasn1.h"
#include "j2735_defines.h"
#include "v2x_error.h"
```

### Data Structures

- struct [spedelimits](#)
- struct [intersec\\_access\\_pt](#)
- struct [J2735Position3D](#)
- struct [utcTimeFrame](#)
- struct [position\\_confidence](#)
- struct [j2735\\_2016\\_fullposvec](#)
- struct [positional\\_accuracy\\_data](#)
- struct [positional\\_accuracy\\_data\\_2016](#)
- struct [encbrakestatus](#)
- struct [encprndlspeed](#)
- struct [encvehiclesize](#)
- struct [encpointsetstype04](#)
- struct [lanedataattr\\_list](#)
- struct [nodeattrs](#)
- struct [odelist](#)
- struct [nodepoint\\_list](#)
- struct [\\_J2735\\_2016\\_PathPrediction](#)
- struct [path\\_history](#)
- struct [\\_J2735\\_2016\\_Wiperset](#)
- struct [\\_J2735\\_2016\\_VehicleMeasurements](#)
- struct [laneattr](#)
- struct [connects\\_to](#)
- struct [lane\\_validity\\_time](#)
- struct [lane\\_validity\\_params](#)
- struct [laneset](#)
- struct [road\\_segment](#)
- struct [datetime](#)
- struct [J2735VehicleID\\_t](#)
- struct [J2735\\_2016\\_RequestorType\\_t](#)
- struct [J2735\\_2016\\_SignalRequestorInfo\\_t](#)
- struct [J2735\\_2016\\_VehicleIdent\\_t](#)
- struct [J2735\\_2016\\_VehicleClassification\\_t](#)

### Macros

- `#define J2735_NAV 2125315823`
- `#define J2735_2016_BSM_TSTATE_MIN 0`
- `#define J2735_2016_BSM_TSTATE_MAX 7`
- `#define J2735_2016_BSM_TSTATE_NAV J2735_NAV`
- `#define J2735_2016_NUM_DATAFRAMES_NAV J2735_NAV`
- `#define J2735_2016_TYPE_NAV J2735_NAV`
- `#define J2735_2016_YEAR_NAV J2735_NAV`

- `#define J2735_2016_MONTH_NAV J2735_NAV`
- `#define J2735_2016_DAY_NAV J2735_NAV`
- `#define J2735_2016_HOUR_NAV J2735_NAV`
- `#define J2735_2016_MINUTE_NAV J2735_NAV`
- `#define J2735_2016_SECOND_NAV J2735_NAV`
- `#define J2735_2016_DURATION_NAV J2735_NAV`
- `#define J2735_2016_MIN_OF_YEAR_MIN 0`
- `#define J2735_2016_MIN_OF_YEAR_MAX 527040`
- `#define J2735_2016_MIN_OF_YEAR_NAV J2735_NAV`
- `#define J2735_2016_MIN_DUR_MIN 0`
- `#define J2735_2016_MIN_DUR_MAX 32000`
- `#define J2735_2016_MIN_DUR_NAV J2735_NAV`
- `#define J2735_2016_SECMARK_MIN 0`
- `#define J2735_2016_SECMARK_MAX 65535`
- `#define J2735_2016_MSG_CNT_MIN 0`
- `#define J2735_2016_MSG_CNT_MAX 127`
- `#define J2735_2016_MSG_CNT_MOD 128`
- `#define J2735_2016_MSG_CNT_NAV J2735_NAV`
- `#define J2735_2016_ITIS_CODE_NAV J2735_NAV`
- `#define J2735_2016_ITIS_CODE_MIN 0`
- `#define J2735_2016_ITIS_CODE_MAX 65535`
- `#define J2735_2016_PRIORITY_NAV J2735_NAV`
- `#define J2735_2016_PRIORITY_MIN 0`
- `#define J2735_2016_PRIORITY_MAX 7`
- `#define J2735_2016_PRIORITY_NAV J2735_NAV`
- `#define J2735_2016_RADIUS_MIN 0`
- `#define J2735_2016_RADIUS_MAX 4095`
- `#define J2735_2016_RADIUS_NAV J2735_NAV`
- `#define J2735_2016_LAT_NAV J2735_NAV`
- `#define J2735_2016_LONG_NAV J2735_NAV`
- `#define J2735_2016_ELEV_NAV J2735_NAV`
- `#define J2735_2016_EXTENT_MIN_NUM 0`
- `#define J2735_2016_EXTENT_MAX_NUM 1000000`
- `#define J2735_2016_EXTENT_NAV_NUM J2735_NAV`
- `#define J2735_2016_HS_MIN 0`
- `#define J2735_2016_HS_MAX 0xffff`
- `#define J2735_2016_HS_NAV J2735_NAV`
- `#define J2735_2016_INTERSECTION_LIST_MIN 1`
- `#define J2735_2016_INTERSECTION_LIST_MAX 32`
- `#define J2735_2016_INTERSECTION_LIST_NAV J2735_NAV`
- `#define J2735_2016_INTERSECTION_ID_NAV J2735_NAV`
- `#define J2735_2016_INTERSECTION_ID_MIN 0`
- `#define J2735_2016_INTERSECTION_ID_MAX 65535`
- `#define J2735_2016_INTERSECTION_ID_UNAVAIL J2735_2016_INTERSECTION_ID_MAX`
- `#define J2735_2016_START_TIME_NAV J2735_NAV`
- `#define J2735_2016_MIN_END_TIME_NAV J2735_NAV`
- `#define J2735_2016_MAX_END_TIME_NAV J2735_NAV`
- `#define J2735_2016_LIKELY_TIME_NAV J2735_NAV`
- `#define J2735_2016_CONFIDENCE_NAV J2735_NAV`
- `#define J2735_2016_NEXT_TIME_NAV J2735_NAV`
- `#define J2735_2016_START_TIME_MIN 0`
- `#define J2735_2016_START_TIME_MAX 36000`
- `#define J2735_2016_START_TIME_UNAVAIL 36002`
- `#define J2735_2016_START_TIME_UNIT 1`
- `#define J2735_2016_MIN_END_TIME_MIN 0`

- #define J2735\_2016\_MIN\_END\_TIME\_MAX 36000
- #define J2735\_2016\_MIN\_END\_TIME\_UNAVAIL 36001
- #define J2735\_2016\_MIN\_END\_TIME\_UNIT 1
- #define J2735\_2016\_MAX\_END\_TIME\_MIN 0
- #define J2735\_2016\_MAX\_END\_TIME\_MAX 36000
- #define J2735\_2016\_MAX\_END\_TIME\_UNAVAIL 36001
- #define J2735\_2016\_MAX\_END\_TIME\_UNIT 1
- #define J2735\_2016\_LANEWIDTH\_NAV J2735\_NAV
- #define J2735\_2016\_LANEWIDTH\_MIN 0.0
- #define J2735\_2016\_LANEWIDTH\_MAX 327.67
- #define J2735\_2016\_D\_WIDTH\_NAV J2735\_NAV
- #define J2735\_2016\_D\_WIDTH\_MIN -512
- #define J2735\_2016\_D\_WIDTH\_MAX 511
- #define J2735\_2016\_D\_ELEVATION\_NAV J2735\_NAV
- #define J2735\_2016\_D\_ELEVATION\_MIN -512
- #define J2735\_2016\_D\_ELEVATION\_MAX 511
- #define J2735\_2016\_LANEWIDTH\_UNITS 100.0
- #define J2735\_2016\_NODESET\_NAV J2735\_NAV
- #define J2735\_2016\_NODESET\_MIN 2
- #define J2735\_2016\_NODESET\_MAX 63
- #define J2735\_2016\_NODELIST\_MIN 1
- #define J2735\_2016\_NODELIST\_MAX 64
- #define J2735\_2016\_LANELIST\_NAV J2735\_NAV
- #define J2735\_2016\_LANELIST\_MIN 0
- #define J2735\_2016\_LANELIST\_MAX 255
- #define J2735\_2016\_ENABLED\_LANELIST\_NAV J2735\_NAV
- #define J2735\_2016\_ENABLED\_LANELIST\_MIN 1
- #define J2735\_2016\_ENABLED\_LANELIST\_MAX 16
- #define J2735\_2016\_MMLIST\_MIN 1
- #define J2735\_2016\_MMLIST\_MAX 255
- #define J2735\_2016\_MMLIST\_NAV J2735\_NAV
- #define J2735\_2016\_MANLIST\_MIN 1
- #define J2735\_2016\_MANLIST\_MAX 16
- #define J2735\_2016\_MANLIST\_NAV J2735\_NAV
- #define J2735\_2016\_MM\_EVT\_LIST\_MIN 1
- #define J2735\_2016\_MM\_EVT\_LIST\_MAX 16
- #define J2735\_2016\_MM\_EVT\_LIST\_NAV J2735\_NAV
- #define J2735\_2016\_PHASE\_NAV J2735\_NAV
- #define J2735\_2016\_PHASE\_MIN 1
- #define J2735\_2016\_PHASE\_MAX 16
- #define J2735\_2016\_DIRECTION\_OF\_USE\_MIN 0
- #define J2735\_2016\_DIRECTION\_OF\_USE\_MAX 3
- #define J2735\_2016\_DIRECTION\_OF\_USE\_NAV J2735\_NAV
- #define J2735\_2016\_SPEED\_MIN 0
- #define J2735\_2016\_SPEED\_MAX 8191
- #define J2735\_2016\_SPEED\_NAV J2735\_NAV
- #define J2735\_2016\_SPEEDCONF\_UNAVAIL 0
- #define J2735\_2016\_SPEEDCONF\_PREC\_100MS 1
- #define J2735\_2016\_SPEEDCONF\_PREC\_10MS 2
- #define J2735\_2016\_SPEEDCONF\_PREC\_5MS 3
- #define J2735\_2016\_SPEEDCONF\_PREC\_1MS 4
- #define J2735\_2016\_SPEEDCONF\_PREC\_0\_1MS 5
- #define J2735\_2016\_SPEEDCONF\_PREC\_0\_05MS 6
- #define J2735\_2016\_SPEEDCONF\_PREC\_0\_01MS 7
- #define J2735\_2016\_SPEEDCONF\_NAV 8

- #define J2735\_2016\_HEADING\_CONF\_UNAVAIL 0
- #define J2735\_2016\_HEADING\_CONF\_PREC\_45DEG 1
- #define J2735\_2016\_HEADING\_CONF\_PREC\_10DEG 2
- #define J2735\_2016\_HEADING\_CONF\_PREC\_5DEG 3
- #define J2735\_2016\_HEADING\_CONF\_PREC\_1DEG 4
- #define J2735\_2016\_HEADING\_CONF\_PREC\_0\_1DEG 5
- #define J2735\_2016\_HEADING\_CONF\_PREC\_0\_05DEG 6
- #define J2735\_2016\_HEADING\_CONF\_PREC\_0\_01DEG 7
- #define J2735\_2016\_HEADING\_NAV J2735\_NAV
- #define J2735\_2016\_HEADING\_CONF\_NAV J2735\_NAV
- #define J2735\_2016\_SHAREDLANE\_NAV J2735\_NAV
- #define J2735\_2016\_GENERIC\_LANE\_MIN 1
- #define J2735\_2016\_GENERIC\_LANE\_MAX 255
- #define J2735\_2016\_ROADSEGMENT\_LIST\_MIN 1
- #define J2735\_2016\_ROADSEGMENT\_LIST\_MAX 32
- #define J2735\_2016\_ROADSEG\_ID\_MIN 0
- #define J2735\_2016\_ROADSEG\_ID\_MAX 65535
- #define J2735\_2016\_ROADSEG\_ID\_NAV J2735\_NAV
- #define J2735\_2016\_LANEATTR\_MIN 1
- #define J2735\_2016\_LANEATTR\_MAX 9
- #define J2735\_2016\_VALIDITY\_TIME\_LIST\_MIN 1
- #define J2735\_2016\_VALIDITY\_TIME\_LIST\_MAX 10
- #define J2735\_2016\_SPEEDLIM\_LIST\_MIN 1
- #define J2735\_2016\_SPEEDLIM\_LIST\_MAX 9
- #define J2735\_2016\_VELOCITY\_UNITS 0.02
- #define J2735\_2016\_ZOOM\_MIN 0
- #define J2735\_2016\_ZOOM\_MAX 15
- #define J2735\_2016\_ZOOM\_NAV J2735\_NAV
- #define J2735\_2016\_MOVEMENT\_PHASE\_STATE\_MIN 0
- #define J2735\_2016\_MOVEMENT\_PHASE\_STATE\_MAX 9
- #define J2735\_2016\_MOVEMENT\_PHASE\_STATE\_NAV J2735\_NAV
- #define J2735\_2016\_TIME\_INT\_CONF\_MIN 0
- #define J2735\_2016\_TIME\_INT\_CONF\_MAX 15
- #define J2735\_2016\_TIME\_INT\_CONF\_NAV J2735\_NAV
- #define J2735\_2016\_LIKELY\_TIME\_MIN 0
- #define J2735\_2016\_LIKELY\_TIME\_MAX 36000
- #define J2735\_2016\_LIKELY\_TIME\_NAV J2735\_NAV
- #define J2735\_2016\_NEXT\_TIME\_MIN 0
- #define J2735\_2016\_NEXT\_TIME\_MAX 36000
- #define J2735\_2016\_STRWHLANGLE\_NAV STRWHLANGLE\_NAV\_NUM
- #define J2735\_EARTH\_RAD 6367449.0
- #define J2735\_DEG\_TO\_RAD(\_\_x) ((\_\_x)\*0.0174532925)
- #define J2735\_RAD\_TO\_DEG(\_\_x) ((\_\_x)\*57.2957795)
- #define J2735\_PI 3.14159
- #define J2735\_2016\_GPSSTATUS\_MIN 0
- #define J2735\_2016\_GPSSTATUS\_MAX 3
- #define J2735\_2016\_GPSSTATUS\_NAV J2735\_NAV
- #define J2735\_2016\_PSM\_DEVICE\_USER\_TYPE\_MIN 0
- #define J2735\_2016\_PSM\_DEVICE\_USER\_TYPE\_MAX 4
- #define J2735\_2016\_PSM\_DEVICE\_USER\_TYPE\_NAV J2735\_NAV
- #define J2735\_2016\_PSM\_DEVICE\_USAGE\_STATE\_MIN 0
- #define J2735\_2016\_PSM\_DEVICE\_USAGE\_STATE\_MAX 0x1FF
- #define J2735\_2016\_PSM\_DEVICE\_USAGE\_STATE\_NAV J2735\_NAV
- #define J2735\_2016\_PSM\_CROSSING\_REQ\_MIN 0
- #define J2735\_2016\_PSM\_CROSSING\_REQ\_MAX 1

- **#define J2735\_2016\_PSM\_CROSSING\_REQ\_NAV** J2735\_NAV
- **#define J2735\_2016\_PSM\_CROSS\_STATE\_MIN** 0
- **#define J2735\_2016\_PSM\_CROSS\_STATE\_MAX** 1
- **#define J2735\_2016\_PSM\_CROSS\_STATE\_NAV** J2735\_NAV
- **#define DERROR**(\_format,...) { (void)\_format; }
- **#define DTERSE**(\_format,...) { (void)\_format; }
- **#define DVERBOSE**(\_format,...) { (void)\_format; }
- **#define DPRINTF**(\_format,...) { (void)\_format; }
- **#define DDEBUG**(\_format,...) { (void)\_format; }
- **#define DWARN**(\_format,...) { (void)\_format; }
- **#define J2735\_ERR** DERROR
- **#define J2735\_WARN** DWARN
- **#define J2735\_DEBUG** DDEBUG
- **#define J2735\_MSG\_TYPE\_TIM** [J2735\\_2016\\_TIM\\_MSG\\_D](#)
- **#define J2735\_MSG\_TYPE\_BSM** [J2735\\_2016\\_BSM\\_MSG\\_D](#)
- **#define J2735\_2016\_LANE\_ID\_MIN** 0
- **#define J2735\_2016\_LANE\_ID\_MAX** 255
- **#define J2735\_2016\_LANE\_ID\_NAV** J2735\_NAV
- **#define J2735\_2016\_APPROACH\_ID\_MIN** 0
- **#define J2735\_2016\_APPROACH\_ID\_MAX** 15
- **#define J2735\_2016\_APPROACH\_ID\_NAV** J2735\_NAV
- **#define J2735\_2016\_LANE\_CONN\_ID\_MIN** 0
- **#define J2735\_2016\_LANE\_CONN\_ID\_MAX** 255
- **#define J2735\_2016\_LANE\_CONN\_ID\_NAV** J2735\_NAV
- **#define J2735\_2016\_PRNDL\_MIN\_NUM** 0
- **#define J2735\_2016\_PRNDL\_MAX\_NUM** 7
- **#define J2735\_2016\_PRNDL\_NAV\_NUM** J2735\_NAV
- **#define J2735\_2016\_PATH\_END\_PT\_MIN** -150
- **#define J2735\_2016\_PATH\_END\_PT\_MAX** 150
- **#define J2735\_LANECROWN\_PT\_MIN** -38.1
- **#define J2735\_LANECROWN\_PT\_MAX** 38.1
- **#define J2735\_2016\_LANECROWN\_PT\_MIN** -128
- **#define J2735\_2016\_LANECROWN\_PT\_MAX** 127
- **#define J2735\_2016\_LANECROWN\_PT\_UNAVAIL** -128
- **#define J2735\_2016\_LANECROWN\_PT\_UNITS** 0.3
- **#define J2735\_2016\_LANEANGLE\_MIN** -180
- **#define J2735\_2016\_LANEANGLE\_MAX** 180
- **#define J2735\_2016\_LANEANGLE\_UNAVAIL** -180
- **#define J2735\_2016\_LANEANGLE\_UNITS** 1.5
- **#define J2735\_2016\_PATH\_END\_PT\_NAV** J2735\_NAV
- **#define J2735\_2016\_CROWN\_POINT\_CENTER\_NAV** J2735\_NAV
- **#define J2735\_2016\_CROWN\_POINT\_LEFT\_NAV** J2735\_NAV
- **#define J2735\_2016\_CROWN\_POINT\_RIGHT\_NAV** J2735\_NAV
- **#define J2735\_2016\_LANE\_ANGLE\_NAV** J2735\_NAV
- **#define J2735\_2016\_LONGITUDE\_MIN** -179.9999999
- **#define J2735\_2016\_LONGITUDE\_MAX** 180.0000001
- **#define J2735\_2016\_LONGITUDE\_MAX1** 180.0000000
- **#define J2735\_2016\_LONGITUDE\_UNAVAIL** J2735\_2016\_LONGITUDE\_MAX
- **#define J2735\_2016\_LATITUDE\_MIN** -90.0
- **#define J2735\_2016\_LATITUDE\_MAX** 90.0
- **#define J2735\_2016\_NODE\_OFF\_MIN** 2
- **#define J2735\_2016\_NODE\_OFF\_MAX** 63
- **#define J2735\_2016\_LANE\_DATA\_ATTR\_MIN** 1
- **#define J2735\_2016\_LANE\_DATA\_ATTR\_MAX** 8
- **#define J2735\_2016\_XOFFSET\_MIN** -32767

- #define J2735\_2016\_XOFFSET\_MAX 32767
- #define J2735\_2016\_XOFFSET\_NAV J2735\_NAV
- #define J2735\_2016\_YOFFSET\_MIN -32767
- #define J2735\_2016\_YOFFSET\_MAX 32767
- #define J2735\_2016\_YOFFSET\_NAV J2735\_NAV
- #define J2735\_2016\_ZOFFSET\_MIN -32767
- #define J2735\_2016\_ZOFFSET\_MAX 32767
- #define J2735\_2016\_ZOFFSET\_NAV J2735\_NAV
- #define J2735\_2016\_EVT\_FLAGS\_MIN 0x0000
- #define J2735\_2016\_EVT\_FLAGS\_MAX 0x1fff
- #define J2735\_2016\_EVT\_FLAGS\_NAV J2735\_NAV
- #define J2735\_2016\_EXT\_LIGHTS\_MIN 0x0001
- #define J2735\_2016\_EXT\_LIGHTS\_MAX 0x01ff
- #define J2735\_2016\_EXT\_LIGHTS\_NAV J2735\_NAV
- #define J2735\_2016\_LIGHTBAR\_MIN 0x0001
- #define J2735\_2016\_LIGHTBAR\_MAX 0x00ff
- #define J2735\_2016\_LIGHTBAR\_NAV J2735\_NAV
- #define J2735\_2016\_WIPERS\_REAR\_NAV J2735\_NAV
- #define J2735\_2016\_WIPERS\_REAR\_RATE\_NAV J2735\_NAV
- #define J2735\_2016\_VEHMASS\_MIN 0
- #define J2735\_2016\_VEHMASS\_MAX 170000
- #define J2735\_2016\_VEHMASS\_NAV J2735\_NAV
- #define J2735\_2016\_SHARED\_LANE\_MIN 0x0001
- #define J2735\_2016\_SHARED\_LANE\_MAX 0x0200
- #define J2735\_2016\_SHARED\_LANE\_NAV J2735\_NAV
- #define J2735\_2016\_SIGNAL\_GROUP\_ID\_MIN 0
- #define J2735\_2016\_SIGNAL\_GROUP\_ID\_MAX 255
- #define J2735\_2016\_SIGNAL\_GROUP\_ID\_NAV J2735\_NAV
- #define J2735\_2016\_SIGNAL\_GROUP\_ID\_UNAVAIL J2735\_NAV
- #define J2735\_2016\_PHASE\_NUMBER\_UNAVAIL -1
- #define J2735\_2016\_ALLOWED\_MANEUVERS\_UNAVAL J2735\_NAV
- #define J2735\_2016\_CONNECTS\_TO\_MIN 0
- #define J2735\_2016\_CONNECTS\_TO\_MAX 16
- #define J2735\_2016\_CONNECTS\_TO\_NAV J2735\_NAV
- #define J2735\_2016\_LANE\_ATTR\_LIST\_MIN 1
- #define J2735\_2016\_LANE\_ATTR\_LIST\_MAX 8
- #define J2735\_2016\_SSP\_INDEX\_MIN 0
- #define J2735\_2016\_SSP\_INDEX\_MAX 31
- #define J2735\_2016\_SSP\_INDEX\_NAV J2735\_NAV
- #define J2735\_2016\_BASIC\_VEH\_ROLE\_MIN 0
- #define J2735\_2016\_BASIC\_VEH\_ROLE\_MAX 22
- #define J2735\_2016\_BASIC\_VEH\_ROLE\_NAV J2735\_NAV
- #define J2735\_2016\_REQ\_SUB\_ROLE\_MIN 0
- #define J2735\_2016\_REQ\_SUB\_ROLE\_MAX 15
- #define J2735\_2016\_REQ\_SUB\_ROLE\_NAV J2735\_NAV
- #define J2735\_2016\_REQ\_IMPORTANCE\_LEVEL\_MIN 0
- #define J2735\_2016\_REQ\_IMPORTANCE\_LEVEL\_MAX 15
- #define J2735\_2016\_REQ\_IMPORTANCE\_LEVEL\_NAV J2735\_NAV
- #define J2735\_2016\_REQID\_MIN 0
- #define J2735\_2016\_REQID\_MAX 255
- #define J2735\_2016\_REQID\_NAV J2735\_NAV
- #define J2735\_2016\_ISO\_3833\_VEH\_TYPE\_MIN 0
- #define J2735\_2016\_ISO\_3833\_VEH\_TYPE\_MAX 100
- #define J2735\_2016\_ISO\_3833\_VEH\_TYPE\_NAV J2735\_NAV
- #define J2735\_2016\_VEHICLE\_ID\_MIN 0



- `#define J2735_2016_VEHICLE_ID_MAX` 4294967295ULL
- `#define J2735_2016_VEHICLE_ID_NAV` J2735\_NAV
- `#define J2735_2016_VEH_TYPE_MIN` 0
- `#define J2735_2016_VEH_TYPE_MAX` 15
- `#define J2735_2016_VEH_TYPE_NAV` J2735\_NAV
- `#define J2735_2016_VIN_STRING_MIN` 0
- `#define J2735_2016_VIN_STRING_MAX` 17
- `#define J2735_2016_VIN_STRING_NAV` J2735\_NAV
- `#define J2735_2016_VGROUP_MIN` 9217
- `#define J2735_2016_VGROUP_MAX` 9251
- `#define J2735_2016_VGROUP_NAV` J2735\_NAV
- `#define J2735_2016_RGROUP_MIN` 9729
- `#define J2735_2016_RGROUP_MAX` 9742
- `#define J2735_2016_RGROUP_NAV` J2735\_NAV
- `#define J2735_2016_IREQUIP_MIN` 9985
- `#define J2735_2016_IREQUIP_MAX` 10114
- `#define J2735_2016_IREQUIP_NAV` J2735\_NAV
- `#define J2735_2016_BASIC_VEHICLE_CLASS_MIN` 0
- `#define J2735_2016_BASIC_VEHICLE_CLASS_MAX` 93
- `#define J2735_2016_BASIC_VEHICLE_CLASS_NAV` J2735\_NAV
- `#define J2735_2016_SIREN_IN_USE_MIN` 0
- `#define J2735_2016_SIREN_IN_USE_MAX` 3
- `#define J2735_2016_MVR_MIN` 0
- `#define J2735_2016_MVR_MAX` 3

## Typedefs

- typedef enum [j2735\\_dsrc\\_msgids](#) J2735\_2016\_DSRC\_MsgID\_t
- typedef struct [spedelimits](#) J2735\_2016\_SpeedLimit\_t
- typedef struct [intersec\\_access\\_pt](#) J2735\_2016\_IsecAccessPoint\_t
- typedef struct [J2735Position3D](#) J2735Position3D\_t
- typedef [J2735Position3D\\_t](#) J2735\_2016\_Pos3d\_2\_t
- typedef struct [position\\_confidence](#) J2735\_2016\_PosConf\_t
- typedef struct [j2735\\_2016\\_fullposvec](#) J2735\_2016\_FullPosVec\_t
- typedef [J2735\\_2016\\_FullPosVec\\_t](#) J2735FullPosVector
- typedef struct [lanedataattr\\_list](#) J2735\_2016\_LaneDataAttributes\_t
- typedef struct [nodeattrs](#) J2735\_2016\_NodeAttr\_t
- typedef struct [nodelist](#) J2735\_2016\_NodeList\_t
- typedef struct [nodepoint\\_list](#) J2735\_2016\_NodePoint\_t
- typedef enum [lanedirection](#) J2735\_2016\_LaneDir\_t
- *MAP & LANE specific objects / definitions.*
- typedef struct [\\_J2735\\_2016\\_PathPrediction](#) J2735\_2016\_PathPrediction\_t
- typedef struct [path\\_history](#) J2735\_2016\_PathHistory\_t
- typedef struct [\\_J2735\\_2016\\_Wiperset](#) J2735\_2016\_WiperSet\_t
- typedef struct [\\_J2735\\_2016\\_VehicleMeasurements](#) J2735\_2016\_VehicleMeasurements\_t
- typedef enum [laneshare](#) J2735\_2016\_SharedLaneType\_t
- typedef enum [lanetypevehicle](#) J2735\_2016\_LaneTypeAttrVeh\_t
- typedef enum [lanetypecrosswalk](#) J2735\_2016\_LaneTypeAttrCrossWk\_t
- typedef struct [laneattr](#) J2735\_2016\_LaneAttribute\_t
- typedef struct [connects\\_to](#) J2735\_2016\_Connectsto\_Lanes\_t
- typedef struct [lane\\_validity\\_time](#) Savari\_LVTime\_t

- typedef struct [lane\\_validity\\_params](#) **Savari\_LVParams\_t**
- typedef struct [laneset J2735\\_2016\\_LaneList\\_t](#)
- typedef struct [road\\_segment J2735\\_2016\\_RoadSegment\\_t](#)
- typedef struct [datetime J2735\\_2016\\_DateTime\\_t](#)
- typedef enum  
[J2735\\_2016\\_HeadingSlice J2735\\_2016\\_HeadingSlice\\_t](#)
- typedef enum [J2735Extent J2735Extent\\_t](#)

## Enumerations

- enum [BSMPartII\\_ID\\_Types\\_t](#) {  
[BSM\\_PART\\_II\\_ID\\_TYPE\\_VEH\\_SAFETY\\_EXT](#), [BSM\\_PART\\_II\\_ID\\_TYPE\\_SPECIAL\\_VEH\\_EXT](#),  
[BSM\\_PART\\_II\\_ID\\_TYPE\\_SUPPLEMENTARY\\_VEH\\_EXT](#) }
  - enum [j2735\\_dsrc\\_msgids](#) {  
[J2735\\_2016\\_BSM\\_MSG\\_D](#) = 0x02, [J2735\\_2016\\_TIM\\_MSG\\_D](#) = 0x10,  
[J2735\\_2016\\_RTCM\\_CORRECTIONS\\_MSG\\_D](#) = 0x0C, [J2735\\_2016\\_UPERFRAME\\_D](#) = 0x11,  
[J2735\\_2016\\_MAP\\_MSG\\_P](#) = 0x12, [J2735\\_2016\\_SPaT\\_MSG\\_P](#) = 0x13,  
[J2735\\_2016\\_TIM\\_MSG\\_P](#) = 0x1F, [J2735\\_2016\\_BSM\\_MSG\\_U](#) = 0x14,  
[J2735\\_2016\\_CSR\\_MSG\\_U](#) = 0x15, [J2735\\_2016\\_EVA\\_MSG\\_U](#) = 0x16,  
[J2735\\_2016\\_IC\\_MSG\\_U](#) = 0x17, [J2735\\_2016\\_NMEA\\_CORRECTION\\_MSG\\_U](#) = 0x18,  
[J2735\\_2016\\_PDM\\_MSG\\_U](#) = 0x19, [J2735\\_2016\\_PVD\\_MSG\\_U](#) = 0x1A,  
[J2735\\_2016\\_RSA\\_MSG\\_U](#) = 0x1B, [J2735\\_2016\\_RTCM\\_CORRECTIONS\\_MSG\\_U](#) = 0x1C,  
[J2735\\_2016\\_SRM\\_MSG\\_U](#) = 0x1D, [J2735\\_2016\\_SSM\\_MSG\\_U](#) = 0x1E,  
[J2735\\_2016\\_TIM\\_MSG\\_U](#) = 0x1F, [J2735\\_2016\\_PSM\\_MSG\\_U](#) = 0x20 }
  - enum [J2735\\_2016\\_SpeedlimitType\\_t](#) {  
[J2735\\_2016\\_SPDLIM\\_TYPE\\_UNKNOWN](#) = 0, [J2735\\_2016\\_SPDLIM\\_TYPE\\_MAXSPEED\\_IN\\_SCHOOLZONE](#) = 1,  
[J2735\\_2016\\_SPDLIM\\_TYPE\\_MAXSPEED\\_IN\\_SCHOOL\\_WITH\\_CHILDREN](#) = 2, [J2735\\_2016\\_SPDLIM\\_TYPE\\_CONSTRUCTION](#) = 3,  
[J2735\\_2016\\_SPDLIM\\_TYPE\\_VEH\\_MIN\\_SPD](#) = 4, [J2735\\_2016\\_SPDLIM\\_TYPE\\_VEH\\_MAX\\_SPD](#) = 5,  
[J2735\\_2016\\_SPDLIM\\_TYPE\\_VEH\\_NIGHT\\_MAX\\_SPD](#) = 6, [J2735\\_2016\\_SPDLIM\\_TYPE\\_TRUCK\\_MIN\\_SPD](#) = 7,  
[J2735\\_2016\\_SPDLIM\\_TYPE\\_TRUCK\\_MAX\\_SPD](#) = 8, [J2735\\_2016\\_SPDLIM\\_TYPE\\_TRUCK\\_NIGHT\\_MAX\\_SPD](#) = 9,  
[J2735\\_2016\\_SPDLIM\\_TYPE\\_VEH\\_WITH\\_TRAILER\\_MIN\\_SPEED](#) = 10, [J2735\\_2016\\_SPDLIM\\_TYPE\\_VEH\\_WITH\\_TRAILER\\_MAX\\_SPEED](#) = 11,  
[J2735\\_2016\\_SPDLIM\\_TYPE\\_VEH\\_WITH\\_TRAILER\\_NIGHT\\_MAX\\_SPEED](#) = 12 }
  - enum [J2735\\_2016\\_NodeType\\_t](#) {  
[J2735\\_2016\\_NODE\\_TYPE\\_NOT\\_PRESENT](#) = 0, [J2735\\_2016\\_NODE\\_TYPE\\_NODE\\_OBJS](#),  
[J2735\\_2016\\_NODE\\_TYPE\\_COMP](#) }
  - enum [J2735\\_2016\\_NodeAttrType\\_t](#) {  
[J2735\\_2016\\_NODEATTR\\_NODEATTR](#) = 0x01, [J2735\\_2016\\_NODEATTR\\_DISABLED\\_SEGMENTS](#) = 0x02,  
[J2735\\_2016\\_NODEATTR\\_ENABLED\\_SEGMENTS](#) = 0x04, [J2735\\_2016\\_NODEATTR\\_LANEDATA](#) = 0x08,  
[J2735\\_2016\\_NODEATTR\\_UNAVAIL](#) = 0x00 }
  - enum [lanedirection](#) {  
[J2735\\_2016\\_LANE\\_DIRECTION\\_UNAVAIL](#) = 0x00, [J2735\\_2016\\_LANE\\_DIRECTION\\_INGRESS](#) = 0x01,  
[J2735\\_2016\\_LANE\\_DIRECTION\\_EGRESS](#) = 0x02 }
- MAP & LANE specific objects / definitions.*
- enum [J2735\\_2016\\_DirofUse\\_t](#) {  
[J2735\\_2016\\_DIROFUSE\\_FORWARD](#) = 0, [J2735\\_2016\\_DIROFUSE\\_REVERSE](#),  
[J2735\\_2016\\_DIROFUSE\\_BOTH](#) }
  - enum [J2735\\_2016\\_VehicleEventFlags\\_t](#) {  
[J2735\\_2016\\_EVT\\_HAZARD\\_LIGHTS](#) = 0x0001, [J2735\\_2016\\_EVT\\_STOPLINE\\_VIOLATION](#) = 0x0002,  
[J2735\\_2016\\_EVT\\_ABS\\_ACTIVATED](#) = 0x0004, [J2735\\_2016\\_EVT\\_TRACTION\\_CONTROL](#) = 0x0008,  
[J2735\\_2016\\_EVT\\_STABILITY\\_CONTROL](#) = 0x0010, [J2735\\_2016\\_EVT\\_HAZARDOUS\\_MATERIAL](#) =

```

0x0020,
J2735_2016_EVT_RESERVED1 = 0x0040, J2735_2016_EVT_HARDBRAKING = 0x0080,
J2735_2016_EVT_LIGHTS_CHANGED = 0x0100, J2735_2016_EVT_WIPERS_CHANGED = 0x0200,
J2735_2016_EVT_FLATTIRE = 0x0400, J2735_2016_EVT_DISABLED_VEHICLE = 0x0800,
J2735_2016_EVT_AIRBAG_DEPLOYMENT = 0x1000 }
• enum J2735_2016_ExteriorLights_t {
J2735_2016_EXT_LOWBEAM_ON = 0x0001, J2735_2016_EXT_HIGHBEAM_ON = 0x0002,
J2735_2016_EXT_LEFTTURN_ON = 0x0004, J2735_2016_EXT_RIGHTTURN_ON = 0x0008,
J2735_2016_EXT_HAZARDSIGNAL_ON = 0x0010, J2735_2016_EXT_AUTOMATIC_LIGHT_CONTROL_
ON = 0x0020,
J2735_2016_EXT_DAYTIMERUNLIGHTS_ON = 0x0040, J2735_2016_EXT_FOGLIGHTS_ON = 0x0080,
J2735_2016_EXT_PARKINGLIGHTS_ON = 0x0100 }
• enum J2735_2016_LightBar_t {
J2735_2016_LIGHTBAR_UNAVAIL = 0x0001, J2735_2016_LIGHTBAR_NOT_IN_USE = 0x0002,
J2735_2016_LIGHTBAR_IN_USE = 0x0004, J2735_2016_LIGHTBAR_YELLOW_CAUTION_LIGHTS =
0x0008,
J2735_2016_LIGHTBAR_SCHOOLBUS_LIGHTS = 0x0010, J2735_2016_LIGHTBAR_ARROW_SIGNS_
ACTIVE = 0x0020,
J2735_2016_LIGHTBAR_SLOW_MOVING_VEHICLE = 0x0040, J2735_2016_LIGHTBAR_FREQUENT_
STOPS = 0x0080 }
• enum J2735_2016_WiperStatus_t {
J2735_2016_WIPER_STATUS_UNAVAIL = 0, J2735_2016_WIPER_STATUS_OFF,
J2735_2016_WIPER_INTERMITTENT, J2735_2016_WIPER_LOW,
J2735_2016_WIPER_HIGH, J2735_2016_WIPER_WASHER_IN_USE,
J2735_2016_WIPER_AUTOMATIC_PRESENT }
• enum J2735_2016_VehicleType_t {
VEHICLE_TYPE_NONE, VEHICLE_TYPE_UNKNOWN,
VEHICLE_TYPE_SPECIAL, VEHICLE_TYPE_MOTO,
VEHICLE_TYPE_CAR, VEHICLE_TYPE_CAR_OTHER,
VEHICLE_TYPE_BUS, VEHICLE_TYPE_AXEL_CNT_2,
VEHICLE_TYPE_AXEL_CNT_3, VEHICLE_TYPE_AXEL_CNT_4,
VEHICLE_TYPE_AXEL_CNT_4_TRAILER, VEHICLE_TYPE_AXEL_CNT_5_TRAILER,
VEHICLE_TYPE_AXEL_CNT_6_TRAILER, VEHICLE_TYPE_AXEL_CNT_5_MULTI_TRAILER,
VEHICLE_TYPE_AXEL_CNT_6_MULTI_TRAILER, VEHICLE_TYPE_AXEL_CNT_7_MULTI_TRAILER }
• enum laneshare {
J2735_2016_OVERLAP_LANE_DESCR = 0x0001, J2735_2016_MULTI_LANE_TREATED_AS_ONE =
0x0002,
J2735_2016_OTHER_NON_MOTORIZED = 0x0004, J2735_2016_INDIVIDUAL_MOTOR_TRAFFIC =
0x0008,
J2735_2016_BUS_VEHICLE_TRAFFIC = 0x0010, J2735_2016_TAXI_VEHICLE_TRAFFIC = 0x0020,
J2735_2016_PED_TRAFFIC = 0x0040, J2735_2016_CYCLIST_TRAFFIC = 0x0080,
J2735_2016_TRACKED_VEHICLE_TRAFFIC = 0x0100, J2735_2016_PED_TRAFFIC_2 = 0x0200 }
• enum lanetypevehicle {
J2735_2016_VEH_IS_VEHICLE_REVOCABLE_LANE = 0x01, J2735_2016_VEH_IS_VEHICLE_FLYOVE-
R_LANE = 0x02,
J2735_2016_VEH_HOV_USE_ONLY = 0x04, J2735_2016_VEH_RESTRICTED_TO_BUS_USE = 0x08,
J2735_2016_VEH_RESTRICTED_TO_TAXI_USE = 0x10, J2735_2016_VEH_RESTRICTED_FROM_PUB-
LIC_USE = 0x20,
J2735_2016_VEH_HAS_IR_BEACON_COVERAGE = 0x40, J2735_2016_VEH_PERMISSION_ON_REQ-
UEST = 0x80 }
• enum lanetypecrosswalk {
J2735_2016_CRW_CRW_REVOCABLE_LANE = 0x0001, J2735_2016_CRW_BICYCLE_USE_ONLY =
0x0002,
J2735_2016_CRW_IS_FLYOVER_CRW_LANE = 0x0004, J2735_2016_CRW_FIXED_CYCLE_TIME =
0x0008,
J2735_2016_CRW_BIDIRECTIONAL_CYCLE_TIME = 0x0010, J2735_2016_CRW_HAS_PUSH_TO_WA-
LK_BUTTON = 0x0020,
J2735_2016_CRW_AUDIO_SUPPORT = 0x0040, J2735_2016_CRW_RFSIGNAL_REQ_PRESENT =

```

```

0x0080,
J2735_2016_CRW_UNSIGNALIZED_SEGMENTS_PRESENT = 0x0100 }
• enum J2735_2016_AllowedManeuvers_t {
    J2735_2016_MAN_STRAIGHT_ALLOWED = 0x0001, J2735_2016_MAN_LEFT_ALLOWED = 0x0002,
    J2735_2016_MAN_RIGHT_ALLOWED = 0x0004, J2735_2016_MAN_UTURN_ALLOWED = 0x0008,
    J2735_2016_MAN_LEFT_TURN_ON_RED_ALLOWED = 0x0010, J2735_2016_MAN_RIGHT_TURN_ON-
    _RED_ALLOWED = 0x0020,
    J2735_2016_MAN_LANE_CHANGE_ALLOWED = 0x0040, J2735_2016_MAN_NO_STOPPING_ALLOWE-
    D = 0x0080,
    J2735_2016_MAN_YIELD_ALLWAYS_REQUIRED = 0x0100, J2735_2016_MAN_GOWITH_HALT =
    0x0200,
    J2735_2016_MAN_CAUTION = 0x0400, J2735_2016_MAN_RESERVED1 = 0x0800 }
• enum LaneValidity_Days_t {
    LANE_VALID_DAY_SUNDAY = 0x01, LANE_VALID_DAY_MONDAY = 0x02,
    LANE_VALID_DAY_TUESDAY = 0x04, LANE_VALID_DAY_WEDNESDAY = 0x08,
    LANE_VALID_DAY_THURSDAY = 0x10, LANE_VALID_DAY_FRIDAY = 0x20,
    LANE_VALID_DAY_SATURDAY = 0x40 }
• enum J2735_2016_BasicVehicleRole_t {
    BASIC_VEHICLE_TYPE_BASIC_VEHICLE, BASIC_VEHICLE_TYPE_PUBLIC_TRANSPORT,
    BASIC_VEHICLE_TYPE_SPECIAL_TRANSPORT, BASIC_VEHICLE_TYPE_DANGEROUS_GOODS,
    BASIC_VEHICLE_TYPE_ROAD_WORKS, BASIC_VEHICLE_TYPE_ROAD_RESCUE,
    BASIC_VEHICLE_TYPE_EMERGENCY, BASIC_VEHICLE_TYPE_SAFETYCAR,
    BASIC_VEHICLE_TYPE_NONE_OR_UNKNOWN, BASIC_VEHICLE_TYPE_TRUCK,
    BASIC_VEHICLE_TYPE_MOTORCYCLE, BASIC_VEHICLE_TYPE_ROADSIDE_SOURCE,
    BASIC_VEHICLE_TYPE_POLICE, BASIC_VEHICLE_TYPE_FIRE,
    BASIC_VEHICLE_TYPE_AMBULANCE, BASIC_VEHICLE_TYPE_DOT,
    BASIC_VEHICLE_TYPE_TRANSIT, BASIC_VEHICLE_TYPE_SLOW_MOVING,
    BASIC_VEHICLE_TYPE_STOP_AND_GO, BASIC_VEHICLE_TYPE_CYCLIST,
    BASIC_VEHICLE_TYPE_PEDESTRIAN, BASIC_VEHICLE_TYPE_NON_MOTORIZED,
    BASIC_VEHICLE_TYPE_MILLITARY }
• enum J2735_2016_HeadingSlice {
    J2735_2016_HS_FROM00_0_TO_22_5_DEGREES = 0x0001, J2735_2016_HS_FROM22_5_TO_45_0_D-
    EGREES = 0x0002,
    J2735_2016_HS_FROM45_0_TO_67_5_DEGREES = 0x0004, J2735_2016_HS_FROM67_5_TO_90_0_D-
    EGREES = 0x0008,
    J2735_2016_HS_FROM90_0_TO_112_5_DEGREES = 0x0010, J2735_2016_HS_FROM112_5_TO_135_-
    0_DEGREES = 0x0020,
    J2735_2016_HS_FROM135_0_TO_157_5_DEGREES = 0x0040, J2735_2016_HS_FROM157_5_TO_180-
    _0_DEGREES = 0x0080,
    J2735_2016_HS_FROM180_0_TO_202_5_DEGREES = 0x0100, J2735_2016_HS_FROM202_5_TO_225-
    _0_DEGREES = 0x0200,
    J2735_2016_HS_FROM225_0_TO_247_5_DEGREES = 0x0400, J2735_2016_HS_FROM247_5_TO_270-
    _0_DEGREES = 0x0800,
    J2735_2016_HS_FROM270_0_TO_292_5_DEGREES = 0x1000, J2735_2016_HS_FROM292_5_TO_315-
    _0_DEGREES = 0x2000,
    J2735_2016_HS_FROM315_0_TO_337_5_DEGREES = 0x4000, J2735_2016_HS_FROM337_5_TO_360-
    _0_DEGREES = 0x8000 }
• enum J2735Extent {
    J2735_EXTENT_USEINSTANTLYONLY = 0, J2735_EXTENT_USEFOR3METERS = 1,
    J2735_EXTENT_USEFOR10METERS = 2, J2735_EXTENT_USEFOR50METERS = 3,
    J2735_EXTENT_USEFOR100METERS = 4, J2735_EXTENT_USEFOR500METERS = 5,
    J2735_EXTENT_USEFOR1000METERS = 6, J2735_EXTENT_USEFOR5000METERS = 7,
    J2735_EXTENT_USEFOR10000METERS = 8, J2735_EXTENT_USEFOR50000METERS = 9,
    J2735_EXTENT_USEFOR100000METERS = 10, J2735_EXTENT_USEFOR500000METERS = 11,
    J2735_EXTENT_USEFOR1000000METERS = 12, J2735_EXTENT_USEFOR5000000METERS = 13,
    J2735_EXTENT_USEFOR10000000METERS = 14, J2735_EXTENT_FOREVER = 15 }
• enum J2735_2016_PrioritizationResponseStatus_t {
    J2735_PRIORITIZATION_RESPONSE_UNKNOWN = 0, J2735_PRIORITIZATION_RESPONSE_REQUE-

```

```

STED,
J2735_PRIORITIZATION_RESPONSE_PROCESSING, J2735_PRIORITIZATION_RESPONSE_WATCH-
_OTHER_TRAFFIC,
J2735_PRIORITIZATION_RESPONSE_GRANTED, J2735_PRIORITIZATION_RESPONSE_REJECTED,
J2735_PRIORITIZATION_RESPONSE_MAX_PRESENCE, J2735_PRIORITIZATION_RESPONSE_RES-
SERVICE_LOCKED }

```

- enum **J2735\_2016\_VgroupAffected\_t** {
 

```

VGROUP_ALL_VEHICLES = 9217, VGROUP_BICYCLES = 9218,
VGROUP_MOTOR_CYCLES = 9219, VGROUP_CARS = 9220,
VGROUP_LIGHT_VEHICLES = 9221, VGROUP_CARS_AND_LIGHT_VEHICLES = 9222,
VGROUP_CARS_WITH_TRAILERS = 9223, VGROUP_CARS_WITH_RECREATIONAL_TRAILERS =
9224,
VGROUP_VEHICLES_WITH_TRAILERS = 9225, VGROUP_HEAVY_VEHICLES = 9226,
VGROUP_TRUCKS = 9227, VGROUP_BUSES = 9228,
VGROUP ARTICULATED_BUSES = 9229, VGROUP_SCHOOL_BUSES = 9230,
VGROUP_VEHICLES_WITH_SEMI_TRAILERS = 9231, VGROUP_VEHICLES_WITH_DOBULE_TRAILE-
S = 9232,
VGROUP_HIGH_PROFILE_VEHICLES = 9233, VGROUP_WIDE_VEHICLES = 9234,
VGROUP_LONG_VEHICLES = 9235, VGROUP_HAZARDOUS_LOADS = 9236,
VGROUP_EXCEPTIONAL_LOADS = 9237, VGROUP_ABNORMAL_LOADS = 9238,
VGROUP_CONVOYS = 9239, VGROUP_MAINTENANCE_VEHICLES = 9240,
VGROUP_DELIVERY_VEHICLES = 9241, VGROUP_VEHICLES_WITH_EVEN_NUMBERED_LICENSE -
PLATES = 9242,
VGROUP_VEHICLES_WITH_ODD_NUMBERED_LICENSE_PLATES = 9243, VGROUP_VEHICLES_WI-
TH_PARKING_PERMITS = 9244,
VGROUP_VEHICLES_WITH_CATALYTIC_CONVERTERS = 9245, VGROUP_VEHICLES_WITH_OUT -
CATALYTIC_CONVERTERS = 9246,
VGROUP_GAS_POWERED_VEHICLES = 9247, VGROUP_DIESEL_POWERED_VEHICLES = 9248,
VGROUP_LPG_VEHICLES = 9249, VGROUP_MILLITARY_CONVOYS = 9250,
VGROUP_MILLITARY_VEHICLES = 9251 }

```
- enum **J2735\_2016\_RgroupAffected\_t** {
 

```

RGROUP_EMERGENCY_VEHICLE_UNITS = 9729, RGROUP_FEDERAL_LAW_ENFORCEMENT_UNIT-
S = 9730,
RGROUP_STATE_POLICE_UNITS = 9731, RGROUP_COUNTRY_POLICE_UNITS = 9732,
RGROUP_LOCAL_POLICE_UNITS = 9733, RGROUP_AMBULANCE_UNITS = 9734,
RGROUP_RESCUE_UNITS = 9735, RGROUP_FIRE_UNITS = 9736,
RGROUP_HAZMAT_UNITS = 9737, RGROUP_LIGHT_TOW_UNITS = 9738,
RGROUP_HEAVY_TOW_UNITS = 9739, RGROUP_FREEWAY_SERVICE_PATROLS = 9740,
RGROUP_TRANSPORTATION_RESPONSE_UNITS = 9741, RGROUP_PRIVATE_CONTRACTOR_RES-
PONSE_UNITS = 9742 }

```
- enum **J2735\_2016\_IREquipment\_t** {

```

IR_GROUND_FIRE_SUPPRESSION = 9985, IR_HEAVY_GROUND_EQUIPMENT = 9986,
IR_AIRCRAFT = 9988, IR_MARINE_EQUIPMENT = 9989,
IR_SUPPORT_EQUIPMENT = 9990, IR_MEDICAL_RESCUE_UNIT = 9991,
IR_OTHER = 9993, IR_GROUND_FIRE_SUPPRESSION_OTHER = 9994,
IR_ENGINE = 9995, IR_TRUCK_OR_AERIAL = 9996,
IR_QUINT = 9997, IR_TANKER_PUMPER_COMBINATION = 9998,
IR_BRUSH_TRUCK = 10000, IR_AIRCRAFT_RESCUE_FIREFIGHTING = 10001,
IR_HEAVY_GROUND_EQUIPMENT_OTHER = 10004, IR_DOZER_OR_PLOW = 10005,
IR_TRACTOR = 10006, IR_TANKER_OR_TENDER = 10008,
IR_AIRCRAFT_OTHER = 10024, IR_AIRCRAFT_FIXED_WING_TANKER = 10025,
IR_HELITANKER = 10026, IR_HELICOPTER = 10027,
IR_MARINE_EQUIPMENT_OTHER = 10034, IR_FIRE_BOAT_WITH_PUMP = 10035,
IR_BOAT_NO_PUMP = 10036, IR_SUPPORT_APPARATUS_OTHER = 10044,
IR_BREATHING_APPARATUS_SUPPORT = 10045, IR_LIGHT_AND_AIR_UNIT = 10046,
IR_MEDICAL_RESCUE_UNIT_OTHER = 10054, IR_RESCUE_UNIT = 10055,
IR_URBAN_SEARCH_AND_RESCUE_UNIT = 10056, IR_HIGH_ANGLE_RESCUE = 10057,
IR_CRASH_FIRE_RESCUE = 10058, IR_BLS_UNIT = 10059,
IR_ALS_UNIT = 10060, IR_MOBILE_COMMAND_POST = 10075,
IR_CHIEF_OFFICER_CAR = 10076, IR_HAZMAT_UNIT = 10077,
IR_TYPE_I_HAND_CREW = 10078, IR_TYPE_II_HAND_CREW = 10079,
IR_PRIVATELY_OWNED_VEHICLE = 10083, IR_OTHER_APPARATUS_RESOURCE = 10084,
IR_AMBULANCE = 10085, IR_BOMB_SQUAD_VAN = 10086,
IR_COMBINE_HARVESTER = 10087, IR_CONSTRUCTION_VEHICLE = 10088,
IR_FARM_TRACTOR = 10089, IR_GRASS_CUTTING_MACHINES = 10090,
IR_HAZMANT_CONTAINMENT_TOW = 10091, IR_HEAVY_TOW = 10092,
IR_HEDGE_CUTTING_MACHIENS = 10093, IR_LIGHT_TOW = 10094,
IR_MOBILE_CRANE = 10095, IR_REFUSE_COLLECTION_VEHICLE = 10096,
IR_RESURFACING_VEHICLE = 10097, IR_ROAD_SWEEPER = 10098,
IR_ROAD_SIDE_LITTER_COLLECTION_CREWS = 10099, IR_SALVAGE_VEHICLE = 10100,
IR_SAND_TRUCK = 10101, IR_SNOW_PLOW = 10102,
IR_STEAM_ROLLER = 10103, IR_SWAT_TEAM_VAN = 10104,
IR_TRACK_LAYING_VEHICLE = 10105, IR_UNKNOWN_VEHICLE = 10106,
IR_WHITE_LINING_VEHICLE = 10107, IR_DUMP_TRUCK = 10108,
IR_SUPERVISOR_VEHICLE = 10109, IR_SNOW_BLOWER = 10110,
IR_ROTARY_SNOW_BLOWER = 10111, IR_ROAD_GRADER = 10112,
IR_STEAM_TRUCK = 10113, IR_FLATBED_TOW = 10114 }

```

- enum J2735\_2016\_BasicVehicleClass\_t {
 BASIC\_VEH\_CLASS\_UNKNOWN = 0, BASIC\_VEH\_CLASS\_SPECIAL\_VEHICLE = 1,
 BASIC\_VEH\_CLASS\_PASSENGER\_VEHICLE\_UNKNOWN = 10, BASIC\_VEH\_CLASS\_PASSENGER\_VEHICLE\_OTHER = 11,
 BASIC\_VEH\_CLASS\_LIGHT\_TRUCK\_TYPE\_UNKNOWN = 20, BASIC\_VEH\_CLASS\_LIGHT\_TRUCK\_TYPE\_OTHER = 21,
 BASIC\_VEH\_CLASS\_TRUCK\_UNKNOWN = 25, BASIC\_VEH\_CLASS\_TRUCK\_OTHER = 26,
 BASIC\_VEH\_CLASS\_TRUCK\_AXLECNT2 = 27, BASIC\_VEH\_CLASS\_TRUCK\_AXLECNT3 = 28,
 BASIC\_VEH\_CLASS\_TRUCK\_AXLECNT4 = 29, BASIC\_VEH\_CLASS\_TRUCK\_AXLECNT4\_TRAILER = 30,
 BASIC\_VEH\_CLASS\_TRUCK\_AXLECNT5\_TRAILER = 31, BASIC\_VEH\_CLASS\_TRUCK\_AXLECNT6\_TRAILER = 32,
 BASIC\_VEH\_CLASS\_TRUCK\_AXLECNT5\_MULTI\_TRAILER = 33, BASIC\_VEH\_CLASS\_TRUCK\_AXLECNT6\_MULTI\_TRAILER = 34,
 BASIC\_VEH\_CLASS\_MOTOR\_CYCLE\_UNKNOWN = 40, BASIC\_VEH\_CLASS\_MOTOR\_CYCLE\_OTHER = 41,
 BASIC\_VEH\_CLASS\_MOTOR\_CYCLE\_CRUISER\_STANDARD = 42, BASIC\_VEH\_CLASS\_MOTOR\_CYCLE\_SPORT\_UNCLAD = 43,
 BASIC\_VEH\_CLASS\_MOTOR\_SPORT\_TOURING = 44, BASIC\_VEH\_CLASS\_MOTOR\_SUPER\_SPORT = 45,
 BASIC\_VEH\_CLASS\_MOTOR\_TOURING = 46, BASIC\_VEH\_CLASS\_MOTOR\_TRIKE = 47,
 BASIC\_VEH\_CLASS\_MOTOR\_WPASSENGERS = 48, BASIC\_VEH\_CLASS\_TRANSIT\_UNKNOWN =

```

50,
BASIC_VEH_CLASS_TRANSIT_OTHER = 51, BASIC_VEH_CLASS_TRANSIT_BRT = 52,
BASIC_VEH_CLASS_TRANSIT_EXPRESS_BUS = 53, BASIC_VEH_CLASS_TRANSIT_LOCAL_BUS =
54,
BASIC_VEH_CLASS_TRANSIT_SCHOOL_BUS = 55, BASIC_VEH_CLASS_TRANSIT_FIXED_GUIDE_
WAY = 56,
BASIC_VEH_CLASS_TRANSIT_PARA_TRANSIT = 57, BASIC_VEH_CLASS_TRANSIT_PARA_TRANS-
IT_AMBULANCE = 58,
BASIC_VEH_CLASS_EMERGENCY_UNKNOWN = 60, BASIC_VEH_CLASS_EMERGENCY_OTHER =
61,
BASIC_VEH_CLASS_EMERGENCY_FIRE_LIGHT_VEHICLE = 62, BASIC_VEH_CLASS_EMERGENCY-
_FIRE_HEAVY_VEHICLE = 63,
BASIC_VEH_CLASS_EMERGENCY_FIRE_PARAMEDIC_VEHICLE = 64, BASIC_VEH_CLASS_EMER-
GENCY_FIRE_AMBULANCE_VEHICLE = 65,
BASIC_VEH_CLASS_EMERGENCY_POLICE_LIGHT_VEHICLE = 66, BASIC_VEH_CLASS_EMERGEN-
CY_POLICE_HEAVY_VEHICLE = 67,
BASIC_VEH_CLASS_EMERGENCY_OTHER_RESPONDER = 68, BASIC_VEH_CLASS_EMERGENCY-
_OTHER_AMBULANCE = 69,
BASIC_VEH_CLASS_OTHER_TRAVELER_UNKNOWN = 80, BASIC_VEH_CLASS_OTHER_TRAVELE-
R_OTHER = 81,
BASIC_VEH_CLASS_OTHER_TRAVELER_PEDESTRIAN = 82, BASIC_VEH_CLASS_OTHER_TRAVE-
LER_VISUALLY_DISABLED = 83,
BASIC_VEH_CLASS_OTHER_TRAVELER_PHYSICALLY_DISABLED = 84, BASIC_VEH_CLASS_OTH-
ER_TRAVELER_BICYCLE = 85,
BASIC_VEH_CLASS_OTHER_TRAVELER_VULNERABLE_ROAD_WORKER = 86, BASIC_VEH_CLAS-
S_INFRASTRUCTURE_UNKNOWN = 90,
BASIC_VEH_CLASS_INFRASTRUCTURE_FIXED = 91, BASIC_VEH_CLASS_INFRASTRUCTURE_MO-
VABLE = 92,
BASIC_VEH_CLASS_EQUIPPED_CARGO_TRAILER = 93 }
• enum J2735_2016_SirenInUse_t {
J2735_2016_SIREN_NOT_AVAIL = 0, J2735_2016_SIREN_NOT_IN_USE,
J2735_2016_SIREN_IN_USE, J2735_2016_SIREN_RESERVED }
• enum J2735_2016_MultiVehicleResponse_t {
J2735_2016_MVR_UNAVAIL = 0, J2735_2016_MVR_SINGLE_VEHICLE,
J2735_2016_MVR_MULTI_VEHICLE, J2735_2016_MVR_RESERVED }
• enum J2540_2_ITISCodes_t {
J2540_2_ITIS_EMERGENCY_VEH_ON_ROADWAY = 1796, J2540_2_ITIS_HIGHSPEED_EMERGENCY-
_VEHICLES = 1797,
J2540_2_ITIS_TRAFFIC_BEING_DIR_AROUND_ACCIDENT_AREA = 6926, J2540_2_ITIS_POLICE_DI-
RECTING_TRAFFIC = 6927,
J2540_2_ITIS_RESCUE_WORKERS_DIRECTING_TRAFFIC = 6928, J2540_2_ITIS_REPAIRS_IN_PRO-
GRESS = 6929 }

```

## Functions

- struct [encbrakestatus](#) \_\_attribute\_\_((packed))
- int [j2735\\_common\\_get\\_iid](#) (long iid)
- int [j2735\\_common\\_get\\_msgcount](#) (long msgcount)
- int [j2735\\_common\\_get\\_lanewidth](#) (int lanewidth)
- int [check\\_range](#) (int data, int min, int max)
- int [j2735\\_common\\_get\\_latitude](#) (double \*lat\_double)
- double [j2735\\_common\\_convert\\_to\\_latitude](#) (long data, uint32\_t \*oob)
- int [j2735\\_common\\_get\\_longitude](#) (double \*long\_double)
- double [j2735\\_common\\_convert\\_to\\_longitude](#) (long data, uint32\_t \*oob)
- int [j2735\\_common\\_get\\_elevation2](#) (double \*elevation)
- double [j2735\\_common\\_convert\\_to\\_elevation2](#) (long data, uint32\_t \*oob)
- int [j2735\\_common\\_get\\_elevation](#) (double \*elevation)



- double **j2735\_common\_convert\_to\_elevation** (int data, uint32\_t \*oob)
- int **check\_data\_int** (int \*var, int min, int max)
- double **j2735\_common\_velocity\_mps\_to\_mph** (int speed\_mps)
- int **j2735\_common\_velocity\_mph\_to\_mps** (double speed\_mph)
- uint16\_t **j2735\_common\_get\_heading** (double \*heading)
- double **j2735\_common\_convert\_to\_heading** (int data, uint32\_t \*oob)
- double **j2735\_common\_convert\_to\_speed** (int speed)
- int **j2735\_common\_circular\_shift** (int data)
- void **j2735\_common\_encode\_bit\_string** (uint8\_t \*buf, uint16\_t set\_bits, int size)
- int **j2735\_common\_decode\_bit\_string** (uint8\_t \*buf, int size)
- void **j2735\_common\_reset\_fullpos** ([J2735\\_2016\\_FullPosVec\\_t](#) \*pos)
- void **j2735\_reset\_utctime** (struct [utcTimeFrame](#) \*utc)
- double **j2735\_common\_convert\_to\_yawrate** (int data, uint32\_t \*oob)
- double **j2735\_common\_convert\_to\_accel** (int16\_t data, uint32\_t \*oob)
- int64\_t **j2735\_common\_get\_tempid** (int64\_t \*temp\_id)
- int **j2735\_common\_get\_positionalaccuracy** (double \*pos1, double \*pos2, double \*pos3, struct [positional-accuracy\\_data](#) \*pad)
- double **j2735\_common\_convert\_to\_positionalaccuracy\_1** (uint32\_t data, uint32\_t \*oob)
- int **j2735\_common\_get\_transmission\_and\_speed** (uint32\_t \*transmission, double \*speed, struct [encprndlspeed](#) \*prnspeed)
- int **j2735\_common\_convert\_to\_transmission\_and\_speed** (uint16\_t data, uint32\_t \*transmission, double \*speed, uint32\_t \*oob)
- double **j2735\_common\_convert\_to\_positionalaccuracy\_2** (uint32\_t data, uint32\_t \*oob)
- double **j2735\_common\_convert\_to\_positionalaccuracy\_3** (uint32\_t data, uint32\_t \*oob)
- int16\_t **j2735\_common\_get\_angle** (double \*angle)
- double **j2735\_common\_convert\_to\_angle** (int8\_t data, uint32\_t \*oob)
- int **j2735\_common\_get\_accel** (double \*accel)
- int **j2735\_common\_get\_vertaccel** (double \*accel)
- double **j2735\_common\_convert\_to\_vertaccel** (int16\_t data, uint32\_t \*oob)
- int **j2735\_common\_get\_yawrate** (double \*yawrate)
- int16\_t **j2735\_common\_get\_radiusofcurve** (double \*roc)
- double **j2735\_common\_convert\_to\_radius** (int data, uint32\_t \*oob)
- unsigned short **j2735\_common\_get\_confidence** (double \*conf)
- double **j2735\_common\_convert\_to\_confidence** (long data, uint32\_t \*oob)
- double **j2735\_common\_convert\_to\_ph\_longitude** (long data, uint32\_t \*oob)
- double **j2735\_common\_convert\_to\_ph\_latitude** (long data, uint32\_t \*oob)
- int **j2735\_common\_convert\_to\_pos\_and\_elev\_confidence** (int data, int \*pos, int \*elev)
- int **j2735\_common\_convert\_to\_speed\_head\_throttle\_confidence** (int data, int \*head, int \*speed, int \*throttle)
- int **j2735\_common\_get\_ph4\_all** (double \*lat, double \*lon, double \*eleva, uint32\_t \*tim, struct [encpointset-stype04](#) \*temp)
- int **j2735\_common\_get\_extent** (double \*extent)
- double **j2735\_common\_convert\_to\_extent** (uint8\_t extent)
- void **j2735\_dump\_hex** (char \*msg, unsigned char \*buf, int len)
- void **j2735\_dump\_hex\_file** (FILE \*fp, char \*msg, unsigned char \*buf, int len)
- void **j2735\_common\_gettimestamp** (uint64\_t \*secs, uint64\_t \*usecs)
- OssGlobal \* **v2x\_msg\_init** (void)
- void **v2x\_msg\_deinit** (void)
- void **j2735\_common\_reset\_full\_pos\_vector** ([J2735\\_2016\\_FullPosVec\\_t](#) \*fps)
- [v2x\\_status\\_t](#) **j2735\_common\_encode\_heading\_slice** (OssGlobal \*world, [J2735\\_2016\\_HeadingSlice\\_t](#) heading\_slice, [HeadingSlice](#) \*e\_heading\_slice)
- [v2x\\_status\\_t](#) **j2735\_common\_fill\_full\_pos\_vector** ([J2735\\_2016\\_FullPosVec\\_t](#) \*fps, [FullPositionVector](#) \*e\_fps)
- [v2x\\_status\\_t](#) **j2735\_common\_copy\_full\_pos\_vector** ([J2735\\_2016\\_FullPosVec\\_t](#) \*fps, [FullPositionVector](#) \*e\_fps)



- [v2x\\_status\\_t j2735\\_common\\_decode\\_heading\\_slice](#) (OssGlobal \*world, [J2735\\_2016\\_HeadingSlice\\_t](#) \*heading\_slice, HeadingSlice \*e\_heading\_slice)
- [v2x\\_status\\_t j2735\\_common\\_enc\\_pos\\_3d2](#) ([J2735\\_2016\\_Pos3d\\_2\\_t](#) \*pos, Position3D \*epos)
- [v2x\\_status\\_t j2735\\_common\\_dec\\_pos\\_3d2](#) ([J2735\\_2016\\_Pos3d\\_2\\_t](#) \*pos, Position3D \*epos)
- void [j2735\\_common\\_reset\\_refpt2](#) ([J2735\\_2016\\_Pos3d\\_2\\_t](#) \*ref\_pt)
- [v2x\\_status\\_t j2735\\_encode\\_speedlimit\\_list](#) (OssGlobal \*world, int n\_speedlimits, [J2735\\_2016\\_SpeedLimit\\_t](#) \*splims, struct SpeedLimitList\_ \*\*splist)
- [v2x\\_status\\_t j2735\\_decode\\_speedlimit\\_list](#) (int n\_splims, [J2735\\_2016\\_SpeedLimit\\_t](#) \*splims, struct SpeedLimitList\_ \*splim\_list)
- void [j2735\\_memalloc\\_err](#) (const char \*file, const char \*func, int line)
- void [j2735\\_common\\_copy\\_iap](#) ([J2735\\_2016\\_IsecAccessPoint\\_t](#) \*iap, IntersectionAccessPoint \*e\_iap)
- [v2x\\_status\\_t j2735\\_common\\_fill\\_iap](#) ([J2735\\_2016\\_IsecAccessPoint\\_t](#) \*iap, IntersectionAccessPoint \*e\_iap)
- uint32\_t [convert\\_prndl\\_to\\_asn](#) (int32\_t data)
- int [j2735\\_common\\_velocity\\_kmph\\_to\\_mps](#) (double speed\_kmph)
- uint32\_t [convert\\_wheelbrake\\_to\\_asn](#) (uint32\_t data)
- uint32\_t [convert\\_traction\\_to\\_asn](#) (int data)
- uint32\_t [convert\\_absact\\_to\\_asn](#) (int data)
- void [j2735\\_common\\_reset\\_pos3d](#) ([J2735Position3D\\_t](#) \*threed\_pos)
- int [j2540\\_2\\_itis\\_code\\_to\\_str](#) (int itis\_code, char \*itis\_str)

## Variables

- uint16\_t **auxbrakes**
- uint16\_t **brakeboost**
- uint16\_t **scs**
- uint16\_t **abs**
- uint16\_t **traction**
- uint16\_t **sparebit**
- uint16\_t **wheelbrakeunavailable**
- uint16\_t **wheelbrakes**
- uint16\_t **speed**
- uint16\_t **prndl**
- unsigned int **reserved**
- unsigned int **length**
- unsigned int **width**
- long long **time\_offset**
- long long **elev\_offset**
- long long **long\_offset**
- long long **lat\_offset**

## 7.6.1 Detailed Description

partII BSM types

## 7.6.2 Enumeration Type Documentation

### 7.6.2.1 enum BSMPartII\_ID\_Types\_t

Enumerator

**BSM\_PART\_II\_ID\_TYPE\_VEH\_SAFETY\_EXT** Vehicle safety extensions

**BSM\_PART\_II\_ID\_TYPE\_SPECIAL\_VEH\_EXT** Special vehicle extensions

**BSM\_PART\_II\_ID\_TYPE\_SUPPLEMENTARY\_VEH\_EXT** Supplementary vehicle extensions

### 7.6.2.2 enum laneshare

#### Enumerator

**J2735\_2016\_OVERLAP\_LANE\_DESCR** J2735\_2016\_OVERLAP\_LANE\_DESCR - over lapping lane description provided

**J2735\_2016\_MULTI\_LANE\_TREATED\_AS\_ONE** J2735\_2016\_MULTI\_LANE\_TREATED\_AS\_ONE - multiple lanes treated as one

**J2735\_2016\_OTHER\_NON\_MOTORIZED** J2735\_2016\_OTHER\_NON\_MOTORIZED - other non motorized traffic

**J2735\_2016\_INDIVIDUAL\_MOTOR\_TRAFFIC** J2735\_2016\_INDIVIDUAL\_MOTOR\_TRAFFIC - individual motor traffic

**J2735\_2016\_BUS\_VEHICLE\_TRAFFIC** J2735\_2016\_BUS\_VEHICLE\_TRAFFIC - bus vehicle traffic

**J2735\_2016\_TAXI\_VEHICLE\_TRAFFIC** J2735\_2016\_TAXI\_VEHICLE\_TRAFFIC - taxi vehicle traffic

**J2735\_2016\_PED\_TRAFFIC** J2735\_2016\_PED\_TRAFFIC - pedestrian traffic

**J2735\_2016\_CYCLIST\_TRAFFIC** J2735\_2016\_CYCLIST\_TRAFFIC - cyclist traffic

**J2735\_2016\_TRACKED\_VEHICLE\_TRAFFIC** J2735\_2016\_TRACKED\_VEHICLE\_TRAFFIC - tracked vehicle traffic

## 7.7 v2x\_msg\_denm.h File Reference

DEN Message APIs and data structures.

```
#include "etsi_pathhistory.h"
#include "etsi_proto_common.h"
#include "etsi_its_decl.h"
#include "etsi_its_gen_nav.h"
#include <v2x_error.h>
```

### Data Structures

- struct [ETSIItineraryPath\\_t](#)
- struct [ETSIRefDENMs\\_t](#)
- struct [ETSIRestrictedTypes\\_t](#)
- struct [v2x\\_etsi\\_denm\\_rwc](#)
- struct [v2x\\_etsi\\_denm\\_ac](#)
- struct [v2x\\_etsi\\_denm\\_lc](#)
- struct [v2x\\_etsi\\_denm\\_sc](#)
- struct [v2x\\_etsi\\_denm\\_mc](#)
- struct [v2x\\_etsi\\_denm](#)

### Macros

- #define **ETSI\_DENM\_VALIDITY\_DURATION\_DEFAULT** 600
- #define **ETSI\_LBSIREN\_NAV** ETSI\_NAV
- #define **ETSI\_HARD\_SHOULDER\_STATUS\_NAV** ETSI\_NAV
- #define **ETSI\_DRIVING\_LANE\_STATUS\_NAV** ETSI\_NAV
- #define **ETSI\_REF\_DENMS\_MIN** 1
- #define **ETSI\_REF\_DENMS\_MAX** 8
- #define **ETSI\_REF\_DENMS\_NAV** ETSI\_NAV
- #define **ETSI\_ITINERARY\_PATH\_MIN** 1
- #define **ETSI\_ITINERARY\_PATH\_MAX** 40

- #define **ETSI\_ITINERARY\_PATH\_NAV** ETSI\_NAV
- #define **ETSI\_DENM\_REST\_TYPE\_MAX** 3
- #define **etsi\_denm\_rwc** v2x\_etsi\_denm\_rwc
- #define **etsi\_denm\_ac** v2x\_etsi\_denm\_ac
- #define **etsi\_denm\_lc** v2x\_etsi\_denm\_lc
- #define **etsi\_denm\_sc** v2x\_etsi\_denm\_sc
- #define **etsi\_denm\_mc** v2x\_etsi\_denm\_mc
- #define **V2X\_ETSI\_DENM\_HAS\_MC**(\_\_denm) ((\_\_denm->type) |= ETSI\_DENM\_MC\_PRESENT)
- #define **V2X\_ETSI\_DENM\_HAS\_SC**(\_\_denm) ((\_\_denm->type) |= ETSI\_DENM\_SC\_PRESENT)
- #define **V2X\_ETSI\_DENM\_HAS\_LC**(\_\_denm) ((\_\_denm->type) |= ETSI\_DENM\_LC\_PRESENT)
- #define **V2X\_ETSI\_DENM\_HAS\_AC**(\_\_denm) ((\_\_denm->type) |= ETSI\_DENM\_AC\_PRESENT)
- #define **ETSI\_DENM\_HAS\_MC** V2X\_ETSI\_DENM\_HAS\_MC
- #define **ETSI\_DENM\_HAS\_SC** V2X\_ETSI\_DENM\_HAS\_SC
- #define **ETSI\_DENM\_HAS\_LC** V2X\_ETSI\_DENM\_HAS\_LC
- #define **ETSI\_DENM\_HAS\_AC** V2X\_ETSI\_DENM\_HAS\_AC
- #define **V2X\_ETSI\_IS\_DENM\_HAS\_MC**(\_\_denm) (!((\_\_denm->type) & ETSI\_DENM\_MC\_PRESENT))
- #define **V2X\_ETSI\_IS\_DENM\_HAS\_SC**(\_\_denm) (!((\_\_denm->type) & ETSI\_DENM\_SC\_PRESENT))
- #define **V2X\_ETSI\_IS\_DENM\_HAS\_LC**(\_\_denm) (!((\_\_denm->type) & ETSI\_DENM\_LC\_PRESENT))
- #define **V2X\_ETSI\_IS\_DENM\_HAS\_AC**(\_\_denm) (!((\_\_denm->type) & ETSI\_DENM\_AC\_PRESENT))
- #define **ETSI\_IS\_DENM\_HAS\_MC** V2X\_ETSI\_IS\_DENM\_HAS\_MC
- #define **ETSI\_IS\_DENM\_HAS\_SC** V2X\_ETSI\_IS\_DENM\_HAS\_SC
- #define **ETSI\_IS\_DENM\_HAS\_LC** V2X\_ETSI\_IS\_DENM\_HAS\_LC
- #define **ETSI\_IS\_DENM\_HAS\_AC** V2X\_ETSI\_IS\_DENM\_HAS\_AC
- #define **V2X\_ETSI\_RESET\_DENM\_SC**(\_\_denm) ((\_\_denm->type) ^= ETSI\_DENM\_SC\_PRESENT)
- #define **V2X\_ETSI\_RESET\_DENM\_LC**(\_\_denm) ((\_\_denm->type) ^= ETSI\_DENM\_LC\_PRESENT)
- #define **V2X\_ETSI\_RESET\_DENM\_AC**(\_\_denm) ((\_\_denm->type) ^= ETSI\_DENM\_AC\_PRESENT)
- #define **ETSI\_RESET\_DENM\_SC** V2X\_ETSI\_RESET\_DENM\_SC
- #define **ETSI\_RESET\_DENM\_LC** V2X\_ETSI\_RESET\_DENM\_LC
- #define **ETSI\_RESET\_DENM\_AC** V2X\_ETSI\_RESET\_DENM\_AC
- #define **etsi\_denm** v2x\_etsi\_denm

## Typedefs

- typedef struct v2x\_etsi\_denm v2x\_msg\_denm\_t

## Enumerations

- enum **ETSI\_RWSContainer\_t** {  
**ETSI\_LIGHT\_BAR\_SIREN\_PRESENT** = 0x0001, **ETSI\_RWC\_EXT\_CLOSED\_LANE\_PRESENT** = 0x0002,  
**ETSI\_RESTRICTION\_PRESENT** = 0x0004, **ETSI\_RWC\_EXT\_SPEEDLIM\_PRESENT** = 0x0008,  
**ETSI\_RWC\_EXT\_IIC\_PRESENT** = 0x0010, **ETSI\_REC\_PATH\_PRESENT** = 0x0020,  
**ETSI\_START\_POINT\_SPEEDLIM\_PRESENT** = 0x0040, **ETSI\_TRAF\_FLOW\_RULE\_PRESENT** = 0x0080,  
**ETSI\_REF\_DENM\_PRESENT** = 0x0100 }
- enum **ETSI\_ACSubContainer\_t** {  
**ETSI\_AC\_LANE\_POS\_PRESENT** = 0x01, **ETSI\_AC\_IR\_PRESENT** = 0x02,  
**ETSI\_AC\_EXT\_TEMP\_PRESENT** = 0x04, **ETSI\_AC\_RW\_PRESENT** = 0x08,  
**ETSI\_AC\_POS\_SOL\_PRESENT** = 0x10, **ETSI\_AC\_SV\_PRESENT** = 0x20 }
- enum **ETSIDenmContainerTypes\_t** {  
**ETSI\_DENM\_MC\_PRESENT** = 0x01, **ETSI\_DENM\_SC\_PRESENT** = 0x02,  
**ETSI\_DENM\_LC\_PRESENT** = 0x04, **ETSI\_DENM\_AC\_PRESENT** = 0x08 }

## Functions

- `v2x_status_t v2x_etsi_denm_encode` (struct `v2x_etsi_denm` \*denm, uint8\_t \*encbuf, int buf\_len, int \*encbuf\_len)
- int `etsi_denm_encode` (struct `etsi_denm` \*denm, uint8\_t \*encbuf, int encbuf\_len)
- `v2x_status_t v2x_etsi_denm_decode` (struct `v2x_etsi_denm` \*denm, uint8\_t \*encbuf, int encbuf\_len)
- int `etsi_denm_decode` (struct `etsi_denm` \*denm, uint8\_t \*encbuf, int encbuf\_len)
- void `etsi_reset_denm_ac` (struct `etsi_denm_ac` \*ac)
- void `etsi_reset_denm_lc` (struct `etsi_denm_lc` \*lc)
- void `etsi_reset_denm_sc` (struct `etsi_denm_sc` \*sc)
- void `etsi_reset_event_pos` (ETSIPosition\_t \*pos)
- void `etsi_reset_denm_mc` (struct `etsi_denm_mc` \*mc)
- void `v2x_etsi_reset_denm` (struct `v2x_etsi_denm` \*denm)
- void `etsi_reset_denm` (struct `etsi_denm` \*denm)
- void `v2x_etsi_print_denm` (FILE \*fp, struct `v2x_etsi_denm` \*denm)
- void `etsi_print_denm` (FILE \*fp, struct `etsi_denm` \*denm)
- void `v2x_etsi_denm_dumphex` (uint8\_t \*encbuf, int encbuf\_len)
- double `v2x_etsi_denm_get_speed` (struct `etsi_denm` \*den)

### 7.7.1 Detailed Description

DEN Message APIs and data structures. This header file contains the APIs and data structures to encode/decode DEN Message.

Include libetsi and link with -letsi

### 7.7.2 Macro Definition Documentation

7.7.2.1 `#define V2X_ETSI_DENM_HAS_MC( __denm ) ((__denm->type) |= ETSI_DENM_MC_PRESENT)`

Macros to set the bit corresponding to a container

7.7.2.2 `#define V2X_ETSI_IS_DENM_HAS_MC( __denm ) (!((__denm->type) & ETSI_DENM_MC_PRESENT))`

Macros to check if a container is present

7.7.2.3 `#define V2X_ETSI_RESET_DENM_SC( __denm ) ((__denm->type) ^= ETSI_DENM_SC_PRESENT)`

Macros to reset the bit corresponding to a container

## 7.8 v2x\_msg\_etsi\_map.h File Reference

MAP message APIs and data structures.

```
#include "etsi_proto_common.h"
#include "etsi_its_decl.h"
#include <v2x_error.h>
```

## Data Structures

- struct [etsi\\_pos3d](#)
- struct [v2x\\_etsi\\_lane\\_attr](#)
- struct [etsi\\_node\\_attr](#)
- struct [etsi\\_node](#)
- struct [etsi\\_connects](#)
- struct [v2x\\_etsi\\_lane](#)
- struct [v2x\\_etsi\\_map\\_isec](#)
- struct [v2x\\_etsi\\_map\\_msg](#)
- struct [v2x\\_etsi\\_map](#)

## Macros

- **#define ETSI\_MAP\_ISEC\_LIST\_LEN 32**
- **#define ETSI\_MAP\_LANE\_LIST\_LEN 255**
- **#define ETSI\_MAP\_NODE\_LEN 64**
- **#define ETSI\_MAP\_CONNECTS\_LEN 16**
- **#define ETSI\_MAP\_SUBID 1**
- **#define ETSI\_MAP\_ISEC\_REGION\_MIN 0**
- **#define ETSI\_MAP\_ISEC\_REGION\_MAX 127**
- **#define ETSI\_MAP\_ISEC\_IID\_MIN 0**
- **#define ETSI\_MAP\_ISEC\_IID\_MAX 65535**
- **#define ETSI\_MAP\_ISEC\_REV\_MIN 0**
- **#define ETSI\_MAP\_ISEC\_REV\_MAX 127**
- **#define ETSI\_MAP\_LW\_MIN 0.0**
- **#define ETSI\_MAP\_LW\_MAX 327.67**
- **#define ETSI\_MAP\_LW\_UNIT 100.0**
- **#define ETSI\_MAP\_LANESET\_MIN 1**
- **#define ETSI\_MAP\_LANESET\_MAX 255**
- **#define ETSI\_MAP\_LANE\_ID\_MIN 0**
- **#define ETSI\_MAP\_LANE\_ID\_MAX 255**
- **#define ETSI\_MAP\_XOFFSET\_MIN -32767**
- **#define ETSI\_MAP\_XOFFSET\_MAX 32767**
- **#define ETSI\_MAP\_YOFFSET\_MIN -32767**
- **#define ETSI\_MAP\_YOFFSET\_MAX 32767**
- **#define ETSI\_MAP\_ZOFFSET\_MIN -32767**
- **#define ETSI\_MAP\_ZOFFSET\_MAX 32767**
- **#define ETSI\_MAP\_MANEUVERS\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_CONNECTS\_TO\_LEN\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_NODE\_LEN\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_MSG\_SUBID\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_MSGID\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_LANEWIDTH\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_ZOFFSET\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_XOFFSET\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_YOFFSET\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_ISEC\_LEN\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_ISEC\_REGION\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_ISEC\_IID\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_ISEC\_REV\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_LW\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_LANESET\_NAV ETSI\_NAV**
- **#define ETSI\_MAP\_LANEID\_NAV ETSI\_NAV**

- `#define ETSI_MAP_APPROACH_TYPE_NAV ETSI_NAV`
- `#define ETSI_MAP_MSGCNT_NAV ETSI_NAV`
- `#define ETSI_MAP_PHASE_NO_NAV ETSI_NAV`
- `#define ETSI_MAP_SIG_GROUP_ID_NAV ETSI_NAV`
- `#define ETSI_MAP_INGRESS_APPROACH_NAV ETSI_NAV`
- `#define ETSI_MAP_EGRESS_APPROACH_NAV ETSI_NAV`
- `#define ETSI_MAP_LANEDIR_NAV ETSI_NAV`
- `#define ETSI_MAP_LANESHARING_NAV ETSI_NAV`
- `#define ETSI_MAP_LANEATTR_TYPE_NAV ETSI_NAV`
- `#define ETSI_MAP_VEHATTR_NAV ETSI_NAV`
- `#define ETSI_MAP_ALLOWED_MANEUVERS_NAV ETSI_NAV`
- `#define etsi_lane_attr v2x_etsi_lane_attr`
- `#define etsi_lane v2x_etsi_lane`
- `#define etsi_map_isec v2x_etsi_map_isec`
- `#define etsi_map_msg v2x_etsi_map_msg`
- `#define etsi_map v2x_etsi_map`

## Typedefs

- `typedef struct etsi_pos3d ETSIPos3d_t`
- `typedef struct etsi_node_attr ETSINodeAttr_t`
- `typedef struct etsi_node ETSINodeList2_t`
- `typedef struct etsi_connects ETSIConnectsTo_t`

## Enumerations

- `enum ETSIApproachType_t { ETSI_APPROACH_INGRESS = 0x1, ETSI_APPROACH_EGRESS = 0x2 }`
- `enum ETSILaneDirection_t { ETSI_LANE_DIR_INGRESS = 0x01, ETSI_LANE_DIR_EGRESS = 0x02 }`
- `enum ETSILaneSharing_t {  
ETSI_LANE_SHARE_OVERLAP_LANE_DESC = 0x0001, ETSI_LANE_SHARE_MULTIPLE_LANES_AS_ONE = 0x0002,  
ETSI_LANE_SHARE_OTHER_NON_MOTOR_TRAFFIC = 0x0004, ETSI_LANE_SHARE_INDIVIDUAL_MOTORIZED_TRAFFIC = 0x0008,  
ETSI_LANE_SHARE_BUS_VEHICLE_TRAFFIC = 0x0010, ETSI_LANE_SHARE_TAXI_VEHICLE_TRAFFIC = 0x0020,  
ETSI_LANE_SHARE_PEDESTRIANS_TRAFFIC = 0x0040, ETSI_LANE_SHARE_CYCLIST_VEHICLE_TRAFFIC = 0x0080,  
ETSI_LANE_SHARE_TRACKED_VEHICLE_TRAFFIC = 0x0100, ETSI_LANE_SHARE_PEDESTRIAN_TRAFFIC = 0x0200 }`
- `enum ETSILaneAttrType_t {  
ETSI_LANE_TYPE_ATTR_VEHICLE = 0x0001, ETSI_LANE_TYPE_ATTR_CROSSWALK = 0x0002,  
ETSI_LANE_TYPE_ATTR_BIKE_LANE = 0x0004, ETSI_LANE_TYPE_ATTR_SIDEWALK = 0x0008,  
ETSI_LANE_TYPE_ATTR_MEDIAN = 0x0010, ETSI_LANE_TYPE_ATTR_STRIPPING = 0x0020,  
ETSI_LANE_TYPE_ATTR_TRACKED_VEHICLE = 0x0040, ETSI_LANE_TYPE_ATTR_PARKING = 0x0080 }`
- `enum ETSILaneTypeAttrVehicle_t {  
ETSI_LANE_TYPE_ATTR_VEH_REVOKABLE_LANE = 0x0001, ETSI_LANE_TYPE_ATTR_VEH_FLYOVER_LANE = 0x0002,  
ETSI_LANE_TYPE_ATTR_VEH_HOV_LANE = 0x0004, ETSI_LANE_TYPE_ATTR_VEH_RESTRICTED_TO_BUS = 0x0008,  
ETSI_LANE_TYPE_ATTR_VEH_RESTRICTED_TO_TAXI = 0x0010, ETSI_LANE_TYPE_ATTR_VEH_RESTRICTED_FROM_PUBUSE = 0x0020,  
ETSI_LANE_TYPE_ATTR_VEH_HAS_IR_BEACON_COVERAGE = 0x0040 }`

- enum **ETSIAllowedManeuvers\_t** {  
**ETSI\_ALLOWED\_MANEUVER\_STRAIGHT\_ALLOWED** = 0x0001, **ETSI\_ALLOWED\_MANEUVER\_LEFT\_ALLOWED** = 0x0002,  
**ETSI\_ALLOWED\_MANEUVER\_RIGHT\_ALLOWED** = 0x0004, **ETSI\_ALLOWED\_MANEUVER\_UTURN\_ALLOWED** = 0x0008,  
**ETSI\_ALLOWED\_MANEUVER\_LEFT\_TURN\_ON\_RED** = 0x0010, **ETSI\_ALLOWED\_MANEUVER\_RIGHT\_TURN\_ON\_RED** = 0x0020,  
**ETSI\_ALLOWED\_MANEUVER\_LANE\_CHANGE\_ALLOWED** = 0x0040, **ETSI\_ALLOWED\_MANEUVER\_NO\_STOPPING\_ALLOWED** = 0x0080,  
**ETSI\_ALLOWED\_MANEUVER\_YIELD\_ALWAYS\_REQUIRED** = 0x0100, **ETSI\_ALLOWED\_MANEUVER\_GO\_WITH\_HALT** = 0x0200,  
**ETSI\_ALLOWED\_MANEUVER\_CAUTION** = 0x0400, **ETSI\_ALLOWED\_MANEUVER\_RESERVED** = 0x0800 }

## Functions

- [v2x\\_status\\_t v2x\\_etsi\\_map\\_encode](#) (struct [v2x\\_etsi\\_map](#) \*map, uint8\_t \*encbuf, int buf\_len, int \*encbuf\_len)
- int **etsi\_map\_encode** (struct etsi\_map \*map, uint8\_t \*encbuf, int encbuf\_len)
- [v2x\\_status\\_t v2x\\_etsi\\_map\\_decode](#) (struct [v2x\\_etsi\\_map](#) \*map, uint8\_t \*encbuf, int encbuf\_len)
- int **etsi\_map\_decode** (struct etsi\_map \*map, uint8\_t \*encbuf, int encbuf\_len)
- void [v2x\\_etsi\\_map\\_reset](#) (struct [v2x\\_etsi\\_map](#) \*map)
- void **etsi\_map\_reset** (struct etsi\_map \*map)
- void [v2x\\_etsi\\_map\\_free](#) (struct [v2x\\_etsi\\_map](#) \*map)
- void **etsi\_map\_free** (struct etsi\_map \*map)
- void [v2x\\_etsi\\_map\\_print](#) (struct [v2x\\_etsi\\_map](#) \*map, FILE \*fp)
- void **etsi\_map\_print** (struct etsi\_map \*map, FILE \*fp)
- void [v2x\\_etsi\\_map\\_dump](#) (uint8\_t \*encbuf, int encbuf\_len)

### 7.8.1 Detailed Description

MAP message APIs and data structures. This header file contains the APIs and data structures to encode/decode MAP message

Include libetsi.h and link with -letsi

## 7.9 v2x\_msg\_etsi\_spat.h File Reference

SPAT Message APIs and data structures.

```
#include "etsi_proto_common.h"
#include "etsi_its_decl.h"
#include <v2x_error.h>
```

## Data Structures

- struct [v2x\\_etsi\\_spat\\_mm\\_evt](#)
- struct [v2x\\_etsi\\_spat\\_mm\\_state](#)
- struct [v2x\\_etsi\\_spat\\_isec](#)
- struct [v2x\\_etsi\\_spat\\_msg](#)
- struct [v2x\\_etsi\\_spat](#)

## Macros

- `#define ETSI_SPAT_SUBID 1`
- `#define ETSI_SPAT_ISEC_LIST_LEN 1`
- `#define ETSI_SPAT_ENABLED_LANES_LEN 16`
- `#define ETSI_SPAT_MMLIST_LEN 256`
- `#define ETSI_SPAT_MMEVT_LEN 16`
- `#define ETSI_SPAT_ASN_IID_MIN 0`
- `#define ETSI_SPAT_ASN_IID_MAX 65535`
- `#define ETSI_SPAT_MSGCNT_MIN 0`
- `#define ETSI_SPAT_MSGCNT_MAX 127`
- `#define ETSI_SPAT_MSGCNT_MOD 128`
- `#define ETSI_SPAT_ISEC_STATUS_MIN 0x0000`
- `#define ETSI_SPAT_ISEC_STATUS_MAX 0x3fff`
- `#define ETSI_SPAT_ENABLED_LANES_MIN 1`
- `#define ETSI_SPAT_ENABLED_LANES_MAX 16`
- `#define ETSI_SPAT_ISEC_LIST_MIN 1`
- `#define ETSI_SPAT_ISEC_LIST_MAX 32`
- `#define ETSI_SPAT_LANEID_MIN 0`
- `#define ETSI_SPAT_LANEID_MAX 255`
- `#define ETSI_SPAT_MMLIST_LEN_MIN 1`
- `#define ETSI_SPAT_MMLIST_LEN_MAX 255`
- `#define ETSI_SPAT_MMEVT_LEN_MIN 1`
- `#define ETSI_SPAT_MMEVT_LEN_MAX 16`
- `#define ETSI_SPAT_SIGGROUPID_MIN 0`
- `#define ETSI_SPAT_SIGGROUPID_MAX 255`
- `#define ETSI_SPAT_MM_PH_STATE_MIN 0`
- `#define ETSI_SPAT_MM_PH_STATE_MAX 9`
- `#define ETSI_SPAT_START_TIME_MIN 0`
- `#define ETSI_SPAT_START_TIME_MAX 36001`
- `#define ETSI_SPAT_START_TIME_UNAVAIL 36002`
- `#define ETSI_SPAT_MIN_END_TIME_MIN 0`
- `#define ETSI_SPAT_MIN_END_TIME_MAX 36001`
- `#define ETSI_SPAT_MIN_END_TIME_UNAVAIL 36002`
- `#define ETSI_SPAT_MAX_END_TIME_MIN 0`
- `#define ETSI_SPAT_MAX_END_TIME_MAX 36001`
- `#define ETSI_SPAT_MAX_END_TIME_UNAVAIL 36002`
- `#define ETSI_SPAT_LIKELY_TIME_MIN 0`
- `#define ETSI_SPAT_LIKELY_TIME_MAX 36001`
- `#define ETSI_SPAT_LIKELY_TIME_UNAVAIL 36002`
- `#define ETSI_SPAT_TIME_CONF_MIN 0`
- `#define ETSI_SPAT_TIME_CONF_MAX 15`
- `#define ETSI_SPAT_NEXT_TIME_MIN 0`
- `#define ETSI_SPAT_NEXT_TIME_MAX 36001`
- `#define ETSI_SPAT_NEXT_TIME_UNAVAIL 36002`
- `#define ETSI_SPAT_PDU_VER_NAV ETSI_NAV`
- `#define ETSI_SPAT_IID_NAV ETSI_NAV`
- `#define ETSI_SPAT_REVISION_NAV ETSI_NAV`
- `#define ETSI_SPAT_ISEC_STATUS_NAV ETSI_NAV`
- `#define ETSI_SPAT_MIN_OF_YR_NAV ETSI_NAV`
- `#define ETSI_SPAT_DSEC_NAV ETSI_NAV`
- `#define ETSI_SPAT_ENABLED_LANES_NAV ETSI_NAV`
- `#define ETSI_SPAT_LANEID_NAV ETSI_NAV`
- `#define ETSI_SPAT_MMLIST_LEN_NAV ETSI_NAV`
- `#define ETSI_SPAT_MMEVT_LEN_NAV ETSI_NAV`



- `#define ETSI_SPAT_SIGGROUPIX_NAV` ETSI\_NAV
- `#define ETSI_SPAT_MM_PH_STATE_NAV` ETSI\_NAV
- `#define ETSI_SPAT_START_TIME_NAV` ETSI\_NAV
- `#define ETSI_SPAT_MIN_END_TIME_NAV` ETSI\_NAV
- `#define ETSI_SPAT_MAX_END_TIME_NAV` ETSI\_NAV
- `#define ETSI_SPAT_LIKELY_TIME_NAV` ETSI\_NAV
- `#define ETSI_SPAT_TIME_CONF_NAV` ETSI\_NAV
- `#define ETSI_SPAT_NEXT_TIME_NAV` ETSI\_NAV
- `#define etsi_spat_mm_evt` v2x\_etsi\_spat\_mm\_evt
- `#define etsi_spat_mm_state` v2x\_etsi\_spat\_mm\_state
- `#define etsi_spat_isec` v2x\_etsi\_spat\_isec
- `#define etsi_spat_msg` v2x\_etsi\_spat\_msg
- `#define etsi_spat` v2x\_etsi\_spat

## Enumerations

- `enum ETSIIsSecObjStatus_t` {  
`ETSI_ISEC_STATUS_MANUAL_CTRL_ON` = 0x0001, `ETSI_ISEC_STATUS_STOP_TIME_ON` = 0x0002,  
`ETSI_ISEC_STATUS_FAILURE_FLASH` = 0x0004, `ETSI_ISEC_STATUS_PREEMPT_ACTIVE` = 0x0008,  
`ETSI_ISEC_STATUS_TSP_ACTIVE` = 0x0010, `ETSI_ISEC_STATUS_FIXED_TIME_OP` = 0x0020,  
`ETSI_ISEC_STATUS_TRAFFIC_DEPENDENT_OP` = 0x0040, `ETSI_ISEC_STATUS_STANDBY_OP` =  
0x0080,  
`ETSI_ISEC_STATUS_FAILURE_MODE` = 0x0100, `ETSI_ISEC_STATUS_OFF` = 0x0200,  
`ETSI_ISEC_STATUS_RECENT_MAP_UPDATE` = 0x0400, `ETSI_ISEC_STATUS_RECENT_CHANGE_I-`  
`N_ASSIGN_LANES` = 0x0800,  
`ETSI_ISEC_STATUS_NO_VALID_MAP_AVAILABLE` = 0x1000, `ETSI_ISEC_STATUS_NO_VALID SPA-`  
`T_AVAILABLE` = 0x2000 }
- `enum ETSIMmPhaseState_t` {  
`ETSI_MM_PH_STATE_UNAVAIL` = 0, `ETSI_MM_PH_STATE_DARK`,  
`ETSI_MM_PH_STATE_STOP_AND_PROCEED`, `ETSI_MM_PH_STATE_STOP_AND_REMAIN`,  
`ETSI_MM_PH_STATE_PRE_MM`, `ETSI_MM_PH_STATE_PERMISSIVE_MM_ALLOWED`,  
`ETSI_MM_PH_STATE_PROTECTED_MM_ALLOWED`, `ETSI_MM_PH_STATE_PERMISSIVE CLEARA-`  
`NCE`,  
`ETSI_MM_PH_STATE_PROTECTED_CLEARANCE`, `ETSI_MM_PH_STATE_CAUTION CONFLICTING-`  
`TRAFFIC` }

## Functions

- `v2x_status_t v2x_etsi_spat_encode` (struct etsi\_spat \*spat, uint8\_t \*encbuf, int buf\_len, int \*encbuf\_len)
- `int etsi_spat_encode` (struct etsi\_spat \*spat, uint8\_t \*encbuf, int encbuf\_len)
- `v2x_status_t v2x_etsi_spat_decode` (struct etsi\_spat \*spat, uint8\_t \*encbuf, int encbuf\_len)
- `int etsi_spat_decode` (struct etsi\_spat \*spat, uint8\_t \*encbuf, int encbuf\_len)
- `void v2x_etsi_reset_spat` (struct v2x\_etsi\_spat \*spat)
- `void etsi_reset_spat` (struct etsi\_spat \*spat)
- `void v2x_etsi_spat_print` (struct v2x\_etsi\_spat \*spat, FILE \*fp)
- `void etsi_spat_print` (struct etsi\_spat \*spat, FILE \*fp)
- `void v2x_etsi_spat_dumphex` (uint8\_t \*encbuf, int encbuf\_len)

### 7.9.1 Detailed Description

SPAT Message APIs and data structures. This header file contains the APIs and data structures to encode/decode SPAT Message

Include libetsi.h and link with -lets

## 7.10 v2x\_msg\_eva.h File Reference

EVA message API definitions and functions.

```
#include <stdint.h>
#include <v2x_msg_rsa.h>
#include <v2x_msg_common.h>
```

### Data Structures

- struct [v2x\\_emergency\\_details](#)
- struct [v2x\\_msg\\_eva](#)

### Macros

- #define [J2735\\_2016\\_EVA\\_RESP\\_TYPE\\_MIN](#) 0
- #define [J2735\\_2016\\_EVA\\_RESP\\_TYPE\\_MAX](#) 6

### Typedefs

- typedef struct [v2x\\_emergency\\_details](#) [V2xEvaEmergencyDetails\\_t](#)
- typedef struct [v2x\\_msg\\_eva](#) [v2x\\_msg\\_eva\\_t](#)

### Enumerations

- enum [v2x\\_eva\\_response\\_type\\_t](#) {  
[V2X\\_EVA\\_RESPONSE\\_NOT\\_EQUIPPED](#), [V2X\\_EVA\\_RESPONSE\\_EMERGENCY](#),  
[V2X\\_EVA\\_RESPONSE\\_NON\\_EMERGENCY](#), [V2X\\_EVA\\_RESPONSE\\_PURSUIT](#),  
[V2X\\_EVA\\_RESPONSE\\_STATIONARY](#), [V2X\\_EVA\\_RESPONSE\\_SLOW\\_MOVING](#),  
[V2X\\_EVA\\_RESPONSE\\_STOP\\_AND\\_GO](#) }

### Functions

- [v2x\\_status\\_t v2x\\_msg\\_eva\\_decode](#) ([v2x\\_msg\\_eva\\_t](#) \*eva, [uint8\\_t](#) \*encbytes, int encLen)
- [v2x\\_status\\_t v2x\\_msg\\_eva\\_encode](#) ([v2x\\_msg\\_eva\\_t](#) \*eva, [uint8\\_t](#) \*encbytes, int encbytes\_size, int \*enc\_len)
- void [v2x\\_msg\\_eva\\_reset](#) ([v2x\\_msg\\_eva\\_t](#) \*eva)
- void [v2x\\_msg\\_eva\\_print](#) (FILE \*fp, [v2x\\_msg\\_eva\\_t](#) \*eva)

#### 7.10.1 Detailed Description

EVA message API definitions and functions. This header file contains the APIs and data structures used to encode or decode EVA.

Include libj2735.h. Link with -lj2735\_2016, -lsae\_2016 -losstoed -lm.

#### 7.10.2 Typedef Documentation

##### 7.10.2.1 typedef struct [v2x\\_emergency\\_details](#) [V2xEvaEmergencyDetails\\_t](#)

Emergency details

### 7.10.3 Enumeration Type Documentation

#### 7.10.3.1 enum v2x\_eva\_response\_type\_t

EVA response type

Enumerator

**V2X\_EVA\_RESPONSE\_NOT\_EQUIPPED** unequipped  
**V2X\_EVA\_RESPONSE\_EMERGENCY** Emergency vehicle  
**V2X\_EVA\_RESPONSE\_NON\_EMERGENCY** Non Emergency vehicle  
**V2X\_EVA\_RESPONSE\_PURSUIT** Pursuit vehicle  
**V2X\_EVA\_RESPONSE\_STATIONARY** stationary vehicle  
**V2X\_EVA\_RESPONSE\_SLOW\_MOVING** slow moving vehicle  
**V2X\_EVA\_RESPONSE\_STOP\_AND\_GO** stop and go vehicle

## 7.11 v2x\_msg\_frame.h File Reference

### Functions

- [v2x\\_status\\_t](#) **v2x\_msg\_decode\_msgframe\_pdu** (uint8\_t \*buf, void \*spat\_msg, void \*map\_msg, int buflen, int \*msg\_id, int debug)
- [v2x\\_status\\_t](#) **decode\_msgframe\_pdu** (uint8\_t \*buf, void \*spat\_msg, void \*map\_msg, int buflen, int \*msg\_id, int debug)

#### 7.11.1 Detailed Description

Decode msgframe PDU based on the message ID (SPAT / MAP).

Parameters

in	<i>buf</i>	encoded buffer
out	<i>spat_msg</i>	spat message
out	<i>map_msg</i>	map message
in	<i>buflen</i>	encoded buffer length
out	<i>msg_id</i>	message ID (SPAT / MAP)

#### Description

This function decodes the encoded payload passed in the buf and length buflen into either spat\_msg or into map\_msg and places the message ID into the msg\_id field. The caller should dereference the spat\_msg or map\_msg based on the contents of the msg\_id (SPAT or MAP)

#### Returns

V2X\_SUCCESS on success  
V2X\_EINVAL invalid argument passed / invalid or unsupported msgID decoded  
V2X\_EFAULT invalid pointer addresses  
V2X\_EDECODE decoding failure

## 7.12 v2x\_msg\_map.h File Reference

MAP message API definitions and functions.

```
#include <stdio.h>
#include "j2735_2016_common.h"
#include "v2x_error.h"
```

## Data Structures

- struct [intersection\\_geodescr](#)
- struct [v2x\\_msg\\_map](#)

## Macros

- #define [mapmsg v2x\\_msg\\_map](#)
- #define [J2735\\_2016\\_MapMsg\\_t v2x\\_msg\\_map\\_t](#)
- #define [j2735\\_encode\\_mapdata\\_upper](#)(map, buf, buflen) [j2735\\_encode\\_mapdata](#)(map, buf, buflen)
- #define [j2735\\_decode\\_mapdata\\_upper](#)(map, buf, buflen) [j2735\\_decode\\_mapdata](#)(map, buf, buflen)

## Typedefs

- typedef struct [intersection\\_geodescr](#) [J2735\\_2016\\_IsecGeoDescr\\_t](#)
- typedef struct [v2x\\_msg\\_map](#) [v2x\\_msg\\_map\\_t](#)

## Functions

- void [v2x\\_msg\\_map\\_init](#) ([v2x\\_msg\\_map\\_t](#) \*map\_msg)
- void [j2735\\_mapdata\\_reset](#) ([J2735\\_2016\\_MapMsg\\_t](#) \*map\_msg)
- void [j2735\\_mapdata\\_reset\\_isec\\_geodescr](#) ([J2735\\_2016\\_IsecGeoDescr\\_t](#) \*isec)
- void [j2735\\_mapdata\\_reset\\_lane](#) ([J2735\\_2016\\_LaneList\\_t](#) \*lane)
- void [j2735\\_2016\\_print\\_map](#) ([J2735\\_2016\\_MapMsg\\_t](#) \*map, FILE \*fp)
- [v2x\\_status\\_t](#) [v2x\\_msg\\_map\\_encode](#) ([v2x\\_msg\\_map\\_t](#) \*map, [uint8\\_t](#) \*msgbuf, int msgbuf\_size, int \*enclen)
- int [j2735\\_encode\\_mapdata](#) ([J2735\\_2016\\_MapMsg\\_t](#) \*map, [uint8\\_t](#) \*buf, int buflen)
- [v2x\\_status\\_t](#) [v2x\\_msg\\_map\\_decode](#) ([v2x\\_msg\\_map\\_t](#) \*map, [uint8\\_t](#) \*buf, int bufsize, int debug)
- int [j2735\\_2016\\_decode\\_mapdata](#) ([J2735\\_2016\\_MapMsg\\_t](#) \*map, [uint8\\_t](#) \*buf, int buflen, int debug)
- void [j2735\\_free\\_mapdata](#) ([J2735\\_2016\\_MapMsg\\_t](#) \*mapmsg)
- void [v2x\\_map\\_free](#) ([v2x\\_msg\\_map\\_t](#) \*map)

### 7.12.1 Detailed Description

MAP message API definitions and functions. This header file contains the APIs and data structures used to encode or decode a MAP message.

Include libj2735.h. Link with -lj2735 -lsae -lm.

### 7.12.2 Typedef Documentation

#### 7.12.2.1 typedef struct [intersection\\_geodescr](#) [J2735\\_2016\\_IsecGeoDescr\\_t](#)

[J2735\\_2016\\_IsecGeoDescr\\_t](#) - intersection geometric description the intersection description consists of a reference point, common lanewidth and a set of lanes describing the intersection.

## 7.13 v2x\_msg\_psm.h File Reference

PSM message API definitions and functions.

```
#include "asndefines.h"
#include "v2x_msg_common.h"
#include "v2x_error.h"
```

### Data Structures

- struct [v2x\\_msg\\_psm](#)

### Macros

- #define **PSM\_DEVICE\_USAGE\_STATE\_SIZE** 2
- #define **psm\_t** [v2x\\_msg\\_psm\\_t](#)

### Typedefs

- typedef enum  
[personal\\_device\\_user\\_type](#) [personal\\_device\\_user\\_type\\_t](#)
- typedef enum  
[personal\\_device\\_usage\\_state](#) [personal\\_device\\_usage\\_state\\_t](#)
- typedef struct [v2x\\_msg\\_psm](#) [v2x\\_msg\\_psm\\_t](#)

### Enumerations

- enum [personal\\_device\\_user\\_type](#) {  
[PSM\\_DEVICE\\_USER\\_TYPE\\_UNAVAILABLE](#) = 0, [PSM\\_DEVICE\\_USER\\_TYPE\\_PEDESTRIAN](#),  
[PSM\\_DEVICE\\_USER\\_TYPE\\_PEDALCYCLIST](#), [PSM\\_DEVICE\\_USER\\_TYPE\\_PUBLICSAFETYWORKER](#),  
[PSM\\_DEVICE\\_USER\\_TYPE\\_ANIMAL](#) }
- enum [personal\\_device\\_usage\\_state](#) {  
[PSM\\_DEVICE\\_USAGE\\_STATE\\_UNAVAIL](#) = 0x0001, [PSM\\_DEVICE\\_USAGE\\_STATE\\_OTHER](#) = 0x0002,  
[PSM\\_DEVICE\\_USAGE\\_STATE\\_IDLE](#) = 0x0004, [PSM\\_DEVICE\\_USAGE\\_STATE\\_LISTENING\\_TO\\_AUDIO](#)  
= 0x0008,  
[PSM\\_DEVICE\\_USAGE\\_STATE\\_TYPING](#) = 0x0010, [PSM\\_DEVICE\\_USAGE\\_STATE\\_CALLING](#) = 0x0020,  
[PSM\\_DEVICE\\_USAGE\\_STATE\\_PLAYING\\_GAMES](#) = 0x0040, [PSM\\_DEVICE\\_USAGE\\_STATE\\_READING](#)  
= 0x0080,  
[PSM\\_DEVICE\\_USAGE\\_STATE\\_VIEWING](#) = 0x0100 }

### Functions

- [v2x\\_status\\_t](#) [v2x\\_msg\\_psm\\_encode](#) ([v2x\\_msg\\_psm\\_t](#) \*psm, unsigned char \*encoded\_buffer, int bufsize, int \*encoded\_len)
- [v2x\\_status\\_t](#) [v2x\\_msg\\_psm\\_decode](#) ([v2x\\_msg\\_psm\\_t](#) \*psm, uint8\_t \*enc\_buf, int buflen, uint32\_t \*oob, int debug)
- void [v2x\\_msg\\_psm\\_reset](#) ([v2x\\_msg\\_psm\\_t](#) \*psm)

#### 7.13.1 Detailed Description

PSM message API definitions and functions. This header file contains the APIs and data structures used to encode or decode a PSM.

Include libj2735.h. Link with -lj2735\_2016 -lsae\_2016 -losstoed -lm

### 7.13.2 Typedef Documentation

#### 7.13.2.1 typedef enum `personal_device_usage_state` `personal_device_usage_state_t`

personal device user state

#### 7.13.2.2 typedef enum `personal_device_user_type` `personal_device_user_type_t`

The PSM device user type

#### 7.13.2.3 typedef struct `v2x_msg_psm` `v2x_msg_psm_t`

PSM data structure

### 7.13.3 Enumeration Type Documentation

#### 7.13.3.1 enum `personal_device_usage_state`

personal device user state

Enumerator

**`PSM_DEVICE_USAGE_STATE_UNAVAIL`** unavailable  
**`PSM_DEVICE_USAGE_STATE_OTHER`** other usage state  
**`PSM_DEVICE_USAGE_STATE_IDLE`** idle state  
**`PSM_DEVICE_USAGE_STATE_LISTENING_TO_AUDIO`** listening to audio  
**`PSM_DEVICE_USAGE_STATE_TYPING`** typing on the keyboard  
**`PSM_DEVICE_USAGE_STATE_CALLING`** in call  
**`PSM_DEVICE_USAGE_STATE_PLAYING_GAMES`** playing games  
**`PSM_DEVICE_USAGE_STATE_READING`** reading on the screen  
**`PSM_DEVICE_USAGE_STATE_VIEWING`** viewing on the screen

#### 7.13.3.2 enum `personal_device_user_type`

The PSM device user type

Enumerator

**`PSM_DEVICE_USER_TYPE_UNAVAILABLE`** unavailable user type  
**`PSM_DEVICE_USER_TYPE_PEDESTRIAN`** pedestrian  
**`PSM_DEVICE_USER_TYPE_PEDALCYCLIST`** ped cyclist  
**`PSM_DEVICE_USER_TYPE_PUBLICSAFETYWORKER`** public safety worker  
**`PSM_DEVICE_USER_TYPE_ANIMAL`** animal

## 7.14 `v2x_msg_pvd.h` File Reference

PVD message API definitions and functions.

```
#include "v2x_msg_common.h"
```

## Data Structures

- struct [\\_J2735\\_2016\\_PHObject](#)
- struct [\\_J2735\\_2016\\_VehSafetyExtensions](#)
- struct [\\_J2735\\_2016\\_VehicleStatus](#)
- struct [\\_J2735\\_2016\\_ProbeSnapshot](#)
- struct [\\_J2735\\_2016\\_PVD](#)

## Macros

- `#define J2735_2016_PSN_MIN 0`
- `#define J2735_2016_PSN_MAX 32767`
- `#define J2735_2016_PSN_NAV J2735_NAV`
- `#define J2735_2016_VEHICLE_ID_MIN 0`
- `#define J2735_2016_VEHICLE_ID_MAX 4294967295ULL`
- `#define J2735_2016_PVD_SNAPSHOT_MIN 1`
- `#define J2735_2016_PVD_SNAPSHOT_MAX 32`
- `#define J2735_2016_PVD_SNAPSHOT_NAV J2735_NAV`
- `#define J2735_2016_PH_PT_MIN 0`
- `#define J2735_2016_PH_PT_MAX 23`
- `#define J2735_2016_PH_PT_NAV J2735_NAV`

## Typedefs

- typedef struct [\\_J2735\\_2016\\_PHObject](#) [J2735\\_2016\\_PHObject\\_t](#)
- typedef struct [\\_J2735\\_2016\\_VehSafetyExtensions](#) [J2735\\_2016\\_VehSafetyExtensions\\_t](#)
- typedef struct [\\_J2735\\_2016\\_VehicleStatus](#) [J2735\\_2016\\_VehicleStatus\\_t](#)
- typedef struct [\\_J2735\\_2016\\_ProbeSnapshot](#) [J2735\\_2016\\_ProbeSnapshot\\_t](#)
- typedef struct [\\_J2735\\_2016\\_PVD](#) [v2x\\_msg\\_pvd\\_t](#)

## Enumerations

- enum [J2735\\_2016\\_GPSStatus\\_t](#) {  
[J2735\\_GPS\\_STATUS\\_UNAVAIL](#) = 0x0001, [J2735\\_GPS\\_STATUS\\_IS\\_HEALTHY](#) = 0x0002,  
[J2735\\_GPS\\_STATUS\\_IS\\_MONITORED](#) = 0x0004, [J2735\\_GPS\\_STATUS\\_BASE\\_STATION\\_TYPE](#) =  
0x0008,  
[J2735\\_GPS\\_STATUS\\_PDOP\\_UNDER\\_5](#) = 0x0010, [J2735\\_GPS\\_STATUS\\_INVIEW\\_OF\\_UNDER\\_5](#) =  
0x0020,  
[J2735\\_GPS\\_STATUS\\_LOCAL\\_CORRECTIONS\\_PRESENT](#) = 0x0040, [J2735\\_GPS\\_STATUS\\_NETWORK-](#)  
[\\_CORRECTIONS\\_PRESENT](#) = 0x0080 }

## Functions

- [v2x\\_status\\_t v2x\\_pvd\\_encode](#) ([v2x\\_msg\\_pvd\\_t](#) \*pvd, [uint8\\_t](#) \*encbuf, int size, int \*encbuf\_len)
- [v2x\\_status\\_t v2x\\_pvd\\_decode](#) ([v2x\\_msg\\_pvd\\_t](#) \*pvd, int type, [uint8\\_t](#) \*encbuf, int buflen, [uint32\\_t](#) \*oob, int \*dec\_len)
- void [v2x\\_pvd\\_reset](#) ([v2x\\_msg\\_pvd\\_t](#) \*pvd)
- void [v2x\\_pvd\\_snapshot\\_reset](#) ([J2735\\_2016\\_ProbeSnapshot\\_t](#) \*snapshot)

### 7.14.1 Detailed Description

PVD message API definitions and functions. This header file contains the APIs and data structures used to encode and decode a PVD

include libj2735.h. Link with -lj2735\_2016 -lsae\_2016 -losstoed -lm.

### 7.14.2 Typedef Documentation

#### 7.14.2.1 typedef struct \_J2735\_2016\_PHObject J2735\_2016\_PHObject\_t

J2735 PH object

#### 7.14.2.2 typedef struct \_J2735\_2016\_ProbeSnapshot J2735\_2016\_ProbeSnapshot\_t

PVD snapshot object ..

#### 7.14.2.3 typedef struct \_J2735\_2016\_VehicleStatus J2735\_2016\_VehicleStatus\_t

Vehicle status

#### 7.14.2.4 typedef struct \_J2735\_2016\_VehSafetyExtensions J2735\_2016\_VehSafetyExtensions\_t

Vehicle safety extensions

#### 7.14.2.5 typedef struct \_J2735\_2016\_PVD v2x\_msg\_pvd\_t

PVD message

Contains a set of snapshots and a PSN related to the snapshots.

### 7.14.3 Enumeration Type Documentation

#### 7.14.3.1 enum J2735\_2016\_GPSStatus\_t

GPS status

Enumerator

**J2735\_GPS\_STATUS\_UNAVAIL** unavailable

**J2735\_GPS\_STATUS\_IS\_HEALTHY** healthy

**J2735\_GPS\_STATUS\_IS\_MONITORED** GPS is monitored

**J2735\_GPS\_STATUS\_BASE\_STATION\_TYPE** base station type

**J2735\_GPS\_STATUS\_PDOP\_UNDER\_5** pdop is under 5

**J2735\_GPS\_STATUS\_INVIEW\_OF\_UNDER\_5** in view of under 5 satellites

**J2735\_GPS\_STATUS\_LOCAL\_CORRECTIONS\_PRESENT** local corrections are present

**J2735\_GPS\_STATUS\_NETWORK\_CORRECTIONS\_PRESENT** network corrections are present



## 7.15 v2x\_msg\_rtcn.h File Reference

RTCM message API definitions and functions.

```
#include "v2x_msg_common.h"
#include "v2x_error.h"
```

### Data Structures

- struct [savari\\_rtcn\\_payload](#)
- struct [savari\\_antennaoffset](#)
- struct [v2x\\_msg\\_rtcn](#)

### Macros

- #define **ERR\_BUF\_LEN** 128 /\* Length of error buffer \*/
- #define **GPSSTATUS\_LEN** 1 /\* gpsstatus 1 byte \*/
- #define **ANTENNA\_OFF\_LEN** 4 /\* antenna offset 4 bytes \*/
- #define **RTCM\_HEADER\_LEN** (GPSSTATUS\_LEN + ANTENNA\_OFF\_LEN)
- #define **RTCM\_REV\_NAV** J2735\_NAV
- #define **RTCM\_REV\_MIN** 0
- #define **RTCM\_REV\_MAX** 3
- #define **RTCM\_PAYLOAD\_LEN\_MIN** 1
- #define **RTCM\_PAYLOAD\_LEN\_MAX** 1023
- #define **RTCM\_PAYLOAD\_LEN\_NAV** J2735\_NAV
- #define **RTCM\_MSG\_LIST\_MIN** 1
- #define **RTCM\_MSG\_LIST\_MAX** 5
- #define **RTCM\_MSG\_LIST\_NAV** J2735\_NAV
- #define **RTCM\_ANTENNA\_OFFSET\_X\_MAX** 2047
- #define **RTCM\_ANTENNA\_OFFSET\_X\_MIN** -2048
- #define **RTCM\_ANTENNA\_OFFSET\_X\_NAV** J2735\_NAV
- #define **RTCM\_ANTENNA\_OFFSET\_Y\_MAX** 255
- #define **RTCM\_ANTENNA\_OFFSET\_Y\_MIN** -256
- #define **RTCM\_ANTENNA\_OFFSET\_Y\_NAV** J2735\_NAV
- #define **RTCM\_ANTENNA\_OFFSET\_Z\_MAX** 511
- #define **RTCM\_ANTENNA\_OFFSET\_Z\_MIN** -512
- #define **RTCM\_ANTENNA\_OFFSET\_Z\_NAV** J2735\_NAV
- #define **RTCM\_ANTENNA\_OFFSET\_MASK\_X** 0x3fff /\* 14 bit mask \*/
- #define **RTCM\_ANTENNA\_OFFSET\_MASK\_YZ** 0x1ff /\* 9 bit mask \*/
- #define **RTCM\_ANTENNA\_OFFSET\_X\_SIGN\_CHECK** 0x00002000
- #define **RTCM\_ANTENNA\_OFFSET\_Y\_SIGN\_CHECK** 0x00000100
- #define **RTCM\_BIT\_MASK\_18\_BIT** 0xffffc000
- #define **RTCM\_BIT\_MASK\_23\_BIT** 0xfffffe00
- #define **savari\_rtcn\_corrections** [v2x\\_msg\\_rtcn](#)
- #define **savari\_rtcn\_corrections\_t** [v2x\\_msg\\_rtcn\\_t](#)

### Typedefs

- typedef struct [savari\\_rtcn\\_payload](#) **savari\_rtcn\_payload\_t**
- typedef struct [savari\\_antennaoffset](#) **savari\_antennaoffset\_t**
- typedef struct [v2x\\_msg\\_rtcn](#) **v2x\_msg\_rtcn\_t**

## Enumerations

- enum **V2X\_RTCMRevision\_t** {  
**V2X\_RTCM\_REV\_UNKNOWN** = 0, **V2X\_RTCM\_REV\_2**,  
**V2X\_RTCM\_REV\_3**, **V2X\_RTCM\_REV\_RESERVED** }

## Functions

- [v2x\\_status\\_t v2x\\_msg\\_rtcmm\\_encode](#) ([v2x\\_msg\\_rtcmm\\_t](#) \*rtcm, unsigned char \*msgbuf, int msgbuf\_size, int \*enclen)
- int **j2735\_encode\_rtcmmdata** (savari\_rtcmm\_corrections\_t \*RtcmmCorrections, unsigned char \*encoded\_buffer, int bufsize)
- [v2x\\_status\\_t v2x\\_msg\\_rtcmm\\_decode](#) ([v2x\\_msg\\_rtcmm\\_t](#) \*rtcm, uint8\_t \*msgbuf, int msgbuf\_len, int \*declen)
- void [v2x\\_msg\\_rtcmm\\_print](#) (FILE \*fp, [v2x\\_msg\\_rtcmm\\_t](#) \*rtcm)
- int **j2735\_decode\_rtcmmdata** (savari\_rtcmm\_corrections\_t \*RtcmmCorrections, uint8\_t \*encoded\_buffer, int bufsize)
- void **j2735\_free\_rtcmmdata** (savari\_rtcmm\_corrections\_t \*RtcmmCorrections)

### 7.15.1 Detailed Description

RTCM message API definitions and functions. This header file contains the APIs and data structures used to encode or decode a RTCM message.

Include `v2x_msg_rtcmm.h` and link with `-lj2735 -lsae -lm`.

## 7.16 v2x\_msg\_spat.h File Reference

SPAT message API definitions and functions.

```
#include <stdio.h>
#include "j2735_2016_common.h"
#include "v2x_error.h"
```

## Data Structures

- struct [\\_J2735\\_2016\\_ManeuverAssistList\\_t](#)
- struct [\\_J2735\\_2016\\_TimeChange\\_t](#)
- struct [\\_J2735\\_2016\\_MovementEventList\\_t](#)
- struct [\\_J2735\\_2016\\_MovementState\\_t](#)
- struct [\\_J2735\\_2016\\_EnableLaneList\\_t](#)
- struct [intersection\\_data](#)
- struct [v2x\\_msg\\_spat](#)

## Macros

- #define **ASN1\_MSG\_ID\_SPAT** 19
- #define **J2735\_2016\_SPAT\_N\_INTERSECTIONS** 255
- #define **J2735\_2016\_SPAT\_ROAD\_REG\_NAV**
- #define **J2735\_2016\_SPAT\_TIME\_STAMP\_NAV** 65535
- #define **J2735\_2016\_SPAT\_TIME\_STAMP\_MAX** 60000
- #define **J2735\_2016\_SPAT\_TIME\_STAMP\_NAV** 65535
- #define **J2735\_2016\_SPAT\_TIME\_CHANGE\_NAV** -1

- `#define J2735_2016_SPAT_TIME_NOT_PRESENT` 255
- `#define J2735_2016_SPAT_LANEID_NOT_AVAIL` 0
- `#define MAX_LANE_NUMBER` 255
- `#define MAX_SIGNAL_GROUP_ID` 255
- `#define SPAT_MAX_MMT_STATES` 255
- `#define J2735_PED_DETECT_TRUE` 1
- `#define J2735_PED_DETECT_FALSE` 0
- `#define J2735_PED_DETECT_NAV` 2
- `#define J2735_2016_TIMESTAMP_MS_MIN` 0
- `#define J2735_2016_TIMESTAMP_MS_MAX` 65535
- `#define J2735_2016_TIMESTAMP_MS_UNAVAIL` 65535
- `#define J2735_2016_TIMESTAMP_MS_NAV` J2735\_NAV
- `#define J2735_2016_spatmsg` v2x\_msg\_spat
- `#define J2735_2016_SPaTMsg_t` v2x\_msg\_spat\_t
- `#define j2735_spat_encode_upper`(spatmsg, buf, len) libj2735\_2016\_spat\_encode(J2735\_2016\_ENCODE-R\_UPER, spatmsg, buf, len)
- `#define j2735_spat_decode_upper`(spatmsg, buf, len) libj2735\_2016\_spat\_decode(spatmsg, buf, len)
- `#define j2735_free_spat` j2735\_2016\_free\_spatmsg

## Typedefs

- typedef enum [intersection\\_status](#) J2735\_2016\_IntersectionStatus\_t
- typedef struct [\\_J2735\\_2016\\_ManeuverAssistList\\_t](#) J2735\_2016\_ManeuverAssistList\_t
- typedef struct [\\_J2735\\_2016\\_TimeChange\\_t](#) J2735\_2016\_TimeChange\_t
- typedef struct [\\_J2735\\_2016\\_MovementEventList\\_t](#) J2735\_2016\_MovementEventList\_t
- typedef struct [\\_J2735\\_2016\\_MovementState\\_t](#) J2735\_2016\_MovementState\_t
- typedef struct [\\_J2735\\_2016\\_EnableLaneList\\_t](#) J2735\_2016\_EnableLaneList\_t
- typedef struct [intersection\\_data](#) J2735\_2016\_Intersection\_t
- typedef struct [v2x\\_msg\\_spat](#) v2x\_msg\_spat\_t

## Enumerations

- enum [intersection\\_status](#) {  
[J2735\\_2016\\_ISEC\\_STATUS\\_MANUAL\\_CONTROL\\_IS\\_ON](#) = 0x0001, [J2735\\_2016\\_ISEC\\_STATUS\\_STOP\\_TIME\\_IS\\_ACTIVE](#) = 0x0002,  
[J2735\\_2016\\_ISEC\\_STATUS\\_FAILURE\\_FLASH](#) = 0x0004, [J2735\\_2016\\_ISEC\\_STATUS\\_PREEMPT\\_IS\\_ACTIVE](#) = 0x0008,  
[J2735\\_2016\\_ISEC\\_STATUS\\_TSP\\_IS\\_ACTIVE](#) = 0x0010, [J2735\\_2016\\_ISEC\\_STATUS\\_FIXED\\_TIME\\_OPERATION](#) = 0x0020,  
[J2735\\_2016\\_ISEC\\_STATUS\\_TRAFFIC\\_DEPENDENT\\_OPERATION](#) = 0x0040, [J2735\\_2016\\_ISEC\\_STATUS\\_STAND\\_BY\\_OPERATION](#) = 0x0080,  
[J2735\\_2016\\_ISEC\\_STATUS\\_FAILURE\\_MODE](#) = 0x0100, [J2735\\_2016\\_ISEC\\_STATUS\\_OFF](#) = 0x0200,  
[J2735\\_2016\\_ISEC\\_STATUS\\_RECENT\\_MAP\\_MESSAGE\\_UPDATE](#) = 0x0400, [J2735\\_2016\\_ISEC\\_STATUS\\_RECENT\\_CHANGE\\_IN\\_MAP\\_ASSIGNED\\_LANEIDS\\_USED](#) = 0x0800,  
[J2735\\_2016\\_ISEC\\_STATUS\\_NOVALID\\_MAP\\_IS\\_AVAILABLE](#) = 0x1000, [J2735\\_2016\\_ISEC\\_STATUS\\_NOVALID\\_SPAT\\_IS\\_AVAILABLE](#) = 0x2000 }
- enum [\\_J2735\\_2016\\_MovementPhaseState\\_t](#)

## Functions

- enum [\\_J2735\\_2016\\_MovementPhaseState\\_t \\_\\_attribute\\_\\_\(\(packed\)\)](#) J2735\_2016\_MovementPhaseState\_t
- void [j2735\\_2016\\_print\\_spat](#) (J2735\_2016\_SPaTMsg\_t \*spat, FILE \*fp)
- [v2x\\_status\\_t v2x\\_msg\\_spat\\_encode](#) (v2x\_msg\_spat\_t \*spatmsg, uint8\_t \*buf, int len, int \*enc\_len)
- int [libj2735\\_2016\\_spat\\_encode](#) (J2735\_2016\_SPaTMsg\_t \*spatmsg, uint8\_t \*buf, int len)
- int [v2x\\_msg\\_spat\\_decode](#) (v2x\_msg\_spat\_t \*spatmsg, uint8\_t \*buf, int len, int \*dec\_len, int debug)
- int [libj2735\\_2016\\_spat\\_decode](#) (J2735\_2016\_SPaTMsg\_t \*spatmsg, uint8\_t \*buf, int len, int debug)
- void [j2735\\_2016\\_free\\_spatmsg](#) (J2735\_2016\_SPaTMsg\_t \*spat)
- void [j2735\\_reset\\_spat\\_timechange](#) (J2735\_2016\_TimeChange\_t \*time\_chg)
- void [v2x\\_spat\\_free](#) (v2x\_msg\_spat\_t \*spat)

## Variables

- [J2735\\_2016\\_PHASE\\_STATE\\_UNKNOWN](#) = 0
- [J2735\\_2016\\_PHASE\\_STATE\\_DARK](#)
- [J2735\\_2016\\_PHASE\\_STATE\\_STOP\\_THEN\\_PROCEED](#)
- [J2735\\_2016\\_PHASE\\_STATE\\_STOP\\_AND\\_REMAIN](#)
- [J2735\\_2016\\_PHASE\\_STATE\\_PRE\\_MOVEMENT](#)
- [J2735\\_2016\\_PHASE\\_STATE\\_PERMISSIVE\\_MOVEMENT\\_ALLOWED](#)
- [J2735\\_2016\\_PHASE\\_STATE\\_PROTECTED\\_MOVEMENT\\_ALLOWED](#)
- [J2735\\_2016\\_PHASE\\_STATE\\_PERMISSIVE\\_CLEARANCE](#)
- [J2735\\_2016\\_PHASE\\_STATE\\_PROTECTED\\_CLEARANCE](#)
- [J2735\\_2016\\_PHASE\\_STATE\\_CAUTION\\_CONFLICTING\\_TRAFFIC](#)

### 7.16.1 Detailed Description

SPAT message API definitions and functions. This header file contains the APIs and data structures used to encode or decode a SPAT message.

Include `libj2735.h`. Link with `-lj2735 -lsae -lm`.

### 7.16.2 Macro Definition Documentation

#### 7.16.2.1 `#define J2735_PED_DETECT_TRUE 1`

`ped_bicycle_detect` - 1 if a pedestrian is in the intersection, 0 if its fairly certain a ped is not. NAV if uncertain.

#### 7.16.2.2 `#define SPAT_MAX_MMT_STATES 255`

MAX movement states in an intersection object

### 7.16.3 Typedef Documentation

#### 7.16.3.1 `typedef struct _J2735_2016_EnableLaneList_t J2735_2016_EnableLaneList_t`

`J2735_2016_EnableLaneList_t` - Enabled Lanes that are advertised in SPaT

#### 7.16.3.2 `typedef struct intersection_data J2735_2016_Intersection_t`

`J2735_2016_Intersection_t` - Intersection information

### 7.16.3.3 typedef enum intersection\_status J2735\_2016\_IntersectionStatus\_t

J2735\_2016\_IntersectionStatus\_t - Intersection status object

### 7.16.3.4 typedef struct \_J2735\_2016\_ManeuverAssistList\_t J2735\_2016\_ManeuverAssistList\_t

J2735\_2016\_ManeuverAssistList\_t - Contains information about the the dynamic flow of traffic for the lane(s) and maneuvers in question (as determined by the LaneConnectionID). Note that this information can be sent regarding any lane-to-lane movement; it need not be limited to the lanes with active (non-red) phases when sent. XXX: NOTE FOR NYC DEMO WE ARE USING SIGNAL GROUP IDs FOR LaneConnectionID

### 7.16.3.5 typedef struct \_J2735\_2016\_MovementEventList\_t J2735\_2016\_MovementEventList\_t

J2735\_2016\_MovementEventList\_t - a movement event describing the phase and the time information of the phase or the future phase

### 7.16.3.6 typedef struct \_J2735\_2016\_MovementState\_t J2735\_2016\_MovementState\_t

J2735\_2016\_MovementState\_t - movement state enclosure describing the signal group id and a set of movement events.

### 7.16.3.7 typedef struct \_J2735\_2016\_TimeChange\_t J2735\_2016\_TimeChange\_t

J2735\_2016\_TimeChange\_t - Timechange describing each phase time active and remaining values. Likely change is reported in the likely\_time. The confidence value indicates the quality of the likely\_time.

## 7.16.4 Enumeration Type Documentation

### 7.16.4.1 enum \_J2735\_2016\_MovementPhaseState\_t

J2735\_2016\_MovementPhaseState\_t - Movement phase state values. They are used to describe the phase states that are coming out of the traffic controller

### 7.16.4.2 enum intersection\_status

J2735\_2016\_IntersectionStatus\_t - Intersection status object

#### Enumerator

**J2735\_2016\_ISEC\_STATUS\_MANUAL\_CONTROL\_IS\_ON** Manual control is activated  
**J2735\_2016\_ISEC\_STATUS\_STOP\_TIME\_IS\_ACTIVE** Stop time is activated  
**J2735\_2016\_ISEC\_STATUS\_FAILURE\_FLASH** Failure flash is active  
**J2735\_2016\_ISEC\_STATUS\_PREEMPT\_IS\_ACTIVE** Preempt is active  
**J2735\_2016\_ISEC\_STATUS\_TSP\_IS\_ACTIVE** TSP is active  
**J2735\_2016\_ISEC\_STATUS\_FIXED\_TIME\_OPERATION** Fixed time operation  
**J2735\_2016\_ISEC\_STATUS\_TRAFFIC\_DEPENDENT\_OPERATION** Traffic dependent operation  
**J2735\_2016\_ISEC\_STATUS\_STAND\_BY\_OPERATION** Standby operation  
**J2735\_2016\_ISEC\_STATUS\_FAILURE\_MODE** Failure mode  
**J2735\_2016\_ISEC\_STATUS\_OFF** Intersection status off  
**J2735\_2016\_ISEC\_STATUS\_RECENT\_MAP\_MESSAGE\_UPDATE** Recent MAP message in update

**J2735\_2016\_ISEC\_STATUS\_RECENT\_CHANGE\_IN\_MAP\_ASSIGNED\_LANEIDS\_USED** Recent change in MAP assigned lanes is in use

**J2735\_2016\_ISEC\_STATUS\_NOVALID\_MAP\_IS\_AVAILABLE** No valid MAP is available

**J2735\_2016\_ISEC\_STATUS\_NOVALID\_SPAT\_IS\_AVAILABLE** No valid SPaT is available

## 7.16.5 Function Documentation

7.16.5.1 `enum _J2735_2016_MovementPhaseState_t __attribute__((packed))`

J2735\_2016\_MovementPhaseState\_t - Movement phase state values. They are used to describe the phase states that are coming out of the traffic controller

## 7.16.6 Variable Documentation

7.16.6.1 **J2735\_2016\_PHASE\_STATE\_CAUTION\_CONFLICTING\_TRAFFIC**

J2735\_2016\_PHASE\_STATE\_CAUTION\_CONFLICTING\_TRAFFIC - 'flashing yellow'

- Proceed with caution

7.16.6.2 **J2735\_2016\_PHASE\_STATE\_DARK**

J2735\_2016\_PHASE\_STATE\_DARK - The signal head is dark.

7.16.6.3 **J2735\_2016\_PHASE\_STATE\_PERMISSIVE\_CLEARANCE**

J2735\_2016\_PHASE\_STATE\_PERMISSIVE\_CLEARANCE - 'permissive yellow'

- Prepare to stop.
- Proceed if unable to stop,
- Clear Intersection.

7.16.6.4 **J2735\_2016\_PHASE\_STATE\_PERMISSIVE\_MOVEMENT\_ALLOWED**

J2735\_2016\_PHASE\_STATE\_PERMISSIVE\_MOVEMENT\_ALLOWED - 'permissive green'

- Proceed with caution,
- must yield to all conflicting traffic

7.16.6.5 **J2735\_2016\_PHASE\_STATE\_PRE\_MOVEMENT**

J2735\_2016\_PHASE\_STATE\_PRE\_MOVEMENT - red + yellow (Not used in the US)

## 7.16.6.6 J2735\_2016\_PHASE\_STATE\_PROTECTED\_CLEARANCE

J2735\_2016\_PHASE\_STATE\_PROTECTED\_CLEARANCE - 'protected yellow'

- Prepare to stop.
- Proceed if unable to stop,
- in indicated direction (to connected lane)
- Clear Intersection.

## 7.16.6.7 J2735\_2016\_PHASE\_STATE\_PROTECTED\_MOVEMENT\_ALLOWED

J2735\_2016\_PHASE\_STATE\_PROTECTED\_MOVEMENT\_ALLOWED - 'protected green'

- Proceed, tossing caution to the wind,
- in indicated (allowed) direction.

## 7.16.6.8 J2735\_2016\_PHASE\_STATE\_STOP\_AND\_REMAIN

J2735\_2016\_PHASE\_STATE\_STOP\_AND\_REMAIN - 'red light'

- Stop vehicle at stop line.
- Do not proceed.

## 7.16.6.9 J2735\_2016\_PHASE\_STATE\_STOP\_THEN\_PROCEED

J2735\_2016\_PHASE\_STATE\_STOP\_THEN\_PROCEED - flashing red

- Stop vehicle at stop line.
- Do not proceed unless it is safe.

## 7.16.6.10 J2735\_2016\_PHASE\_STATE\_UNKNOWN = 0

J2735\_2016\_PHASE\_STATE\_UNKNOWN - state is unknown or error

## 7.17 v2x\_msg\_srm.h File Reference

SRM message API definitions and functions.

#include "v2x\_msg\_common.h"

## Data Structures

- struct [J2735\\_2016\\_SignalRequest\\_t](#)
- struct [srm\\_list](#)
- struct [srm\\_requestor](#)
- struct [j2735\\_2016\\_srm](#)

## Macros

- `#define SRM_DSECOND_MIN 0`
- `#define SRM_DSECOND_MAX 65535`
- `#define SRM_DSECOND_NAV J2735_NAV`
- `#define SRM_MSG_COUNT_MIN 0`
- `#define SRM_MSG_COUNT_MAX 127`
- `#define SRM_MSG_COUNT_NAV J2735_NAV`
- `#define SRM_MSG_LIST_MIN 1`
- `#define SRM_MSG_LIST_MAX 32`
- `#define SRM_MSG_LIST_NAV J2735_NAV`
- `#define SRM_MSG_IID_MIN 0`
- `#define SRM_MSG_IID_MAX 65535`
- `#define SRM_MSG_IID_NAV J2735_NAV`
- `#define SRM_MSG_REQID_MIN 0`
- `#define SRM_MSG_REQID_MAX 255`
- `#define SRM_MSG_REQID_NAV J2735_NAV`
- `#define SRM_MSG_PRIO_REQ_TYPE_MIN 0`
- `#define SRM_MSG_PRIO_REQ_TYPE_MAX 3`
- `#define SRM_MSG_PRIO_REQ_TYPE_NAV J2735_NAV`

## Typedefs

- `typedef struct srm_list J2735SRMList_t`
- `typedef struct srm_requestor J2735SRMRequestor_t`
- `typedef struct j2735_2016_srm v2x_msg_srm_t`

## Enumerations

- `enum J2735_2016_PriorityReqType_t {  
SRM_PRIO_REQ_TYPE_RESERVED, SRM_PRIO_REQ_TYPE_REQ,  
SRM_PRIO_REQ_TYPE_UPDATE, SRM_PRIO_REQ_TYPE_CANCEL }`

## Functions

- `v2x_status_t v2x_srm_encode(v2x_msg_srm_t *srm, uint8_t *encbuf, int *encbuf_len, int size)`
- `v2x_status_t v2x_srm_decode(v2x_msg_srm_t *srm, uint8_t *encbuf, int *encbuf_len, int size)`
- `void v2x_srm_reset(v2x_msg_srm_t *srm)`

### 7.17.1 Detailed Description

SRM message API definitions and functions. This header file contains the APIs and data structures used to encode or decode a SRM.

Include `libj2735.h`. Link with `-lj2735_2016 -lsae_2016 -losstoed -lm`.

### 7.17.2 Typedef Documentation

#### 7.17.2.1 `typedef struct srm_list J2735SRMList_t`

SRM set



## 7.17.2.2 typedef struct srm\_requestor J2735SRMRequestor\_t

SRM requestor

## 7.17.3 Enumeration Type Documentation

## 7.17.3.1 enum J2735\_2016\_PriorityReqType\_t

Priority request type

Enumerator

**SRM\_PRIO\_REQ\_TYPE\_RESERVED** Reserved

**SRM\_PRIO\_REQ\_TYPE\_REQ** Request

**SRM\_PRIO\_REQ\_TYPE\_UPDATE** Update

**SRM\_PRIO\_REQ\_TYPE\_CANCEL** Cancel

## 7.18 v2x\_msg\_ssm.h File Reference

SSM message API definitions and functions.

```
#include "v2x_msg_common.h"
```

```
#include "v2x_error.h"
```

## Data Structures

- struct [J2735\\_2016\\_SignalRequest](#)
- struct [J2735\\_2016\\_SignalStatusPackageList](#)
- struct [J2735\\_2016\\_SignalStatusList](#)
- struct [j2735\\_2016\\_ssm](#)

## Macros

- #define **SSM\_SIGNAL\_STATUS\_LIST\_MIN** 1
- #define **SSM\_SIGNAL\_STATUS\_LIST\_MAX** 32
- #define **SSM\_SIGNAL\_STATUS\_LIST\_NAV** J2735\_NAV
- #define **SSM\_SS\_PKG\_LIST\_MIN** 1
- #define **SSM\_SS\_PKG\_LIST\_MAX** 32
- #define **SSM\_SS\_PKG\_LIST\_NAV** J2735\_NAV

## Typedefs

- typedef struct [J2735\\_2016\\_SignalRequest J2735\\_2016\\_SignalRequesterInfo\\_t](#)
- typedef struct [J2735\\_2016\\_SignalStatusPackageList J2735\\_2016\\_SignalStatusPackageList\\_t](#)
- typedef struct [J2735\\_2016\\_SignalStatusList J2735\\_2016\\_SignalStatusList\\_t](#)
- typedef struct [j2735\\_2016\\_ssm v2x\\_msg\\_ssm\\_t](#)

## Functions

- [v2x\\_status\\_t v2x\\_ssm\\_encode](#) ([v2x\\_msg\\_ssm\\_t](#) \*ssm, [uint8\\_t](#) \*encbuf, [int](#) \*encbuf\_len, [int](#) size)
- [v2x\\_status\\_t v2x\\_ssm\\_decode](#) ([v2x\\_msg\\_ssm\\_t](#) \*ssm, [int](#) type, [uint8\\_t](#) \*encbuf, [int](#) buflen, [uint32\\_t](#) \*oob, [int](#) \*dec\_len)
- [void v2x\\_ssm\\_reset](#) ([v2x\\_msg\\_ssm\\_t](#) \*ssm)

### 7.18.1 Detailed Description

SSM message API definitions and functions. This header file contains the necessary APIs and data structures used to encode and decode a SSM.

Include `libj2735.h`. Link with `-lj2735_2016 -lsae_2016 -losstoed -lm`.

### 7.18.2 Typedef Documentation

#### 7.18.2.1 `typedef struct J2735_2016_SignalRequest J2735_2016_SignalRequesterInfo_t`

Signal request info

#### 7.18.2.2 `typedef struct J2735_2016_SignalStatusList J2735_2016_SignalStatusList_t`

Signal status list

#### 7.18.2.3 `typedef struct J2735_2016_SignalStatusPackageList J2735_2016_SignalStatusPackageList_t`

Signal status package list

## 7.19 `v2x_msg_tim.h` File Reference

TIM message API definitions and functions.

```
#include <stdint.h>
#include <stdlib.h>
#include <string.h>
#include <strings.h>
#include "j2735_2016_common.h"
#include "j2735_2016_node.h"
#include "asndefines.h"
#include "wzhop.h"
#include "v2x_error.h"
```

## Data Structures

- [struct J2735RoadSignID](#)
- [struct J2735ShapePointSet](#)
- [struct J2735CircularRegion](#)
- [struct J2735\\_2016\\_RegionOffsets](#)
- [struct J2735RegionPointSet](#)
- [struct J2735ValidRegion](#)
- [struct RoadSegmentRefId](#)
- [struct J2735\\_2016\\_NodeSet\\_XY](#)

- struct [J2735\\_2016\\_Computed\\_Lanes](#)
- struct [J2735\\_2016\\_NodeXY](#)
- struct [J2735\\_2016\\_LatLong](#)
- struct [J2735\\_2016\\_OffsetSystem](#)
- struct [J2735\\_2016\\_GeometricProjection](#)
- struct [J2735TIMGeoPath](#)
- struct [J2735TIM](#)
- struct [v2x\\_msg\\_tim](#)

## Macros

- `#define TIM_GEOG_PATH_MIN 1`
- `#define TIM_GEOG_PATH_MAX 16`
- `#define TIM_GEOG_PATH_NAV J2735_NAV`
- `#define TIM_UNIQID_LEN 9`
- `#define HEADING_SLICE_BIT_STR_SIZE 16`
- `#define J2735_2016_TIM_TYPE_NAV J2735_NAV`
- `#define ASN1_MSG_ID_TIM 31`
- `#define TIM_MUTCD_CODE_MIN 0`
- `#define TIM_MUTCD_CODE_MAX 6`
- `#define TIM_MUTCD_CODE_NAV J2735_NAV`
- `#define J2735_2016_REG_OFFSET_MIN 1`
- `#define J2735_2016_REG_OFFSET_MAX 64`
- `#define J2735_2016_REG_OFFSET_NAV J2735_NAV`
- `#define J2735_2016_CLOSED_PATH_MIN 0`
- `#define J2735_2016_CLOSED_PATH_MAX 1`
- `#define J2735_2016_CLOSED_PATH_NAV J2735_NAV`
- `#define J2735TIMMsg v2x_msg_tim`
- `#define J2735TIMMsg_t v2x_msg_tim_t`
- `#define TIM_LAT_MIN "-90.000000"`
- `#define TIM_LAT_MAX "90.000000"`
- `#define TIM_LAT_NAV "2125315840"`
- `#define TIM_LAT_MIN_NUM -90.000000`
- `#define TIM_LAT_MAX_NUM 90.000000`
- `#define TIM_LAT_NAV_NUM 2125315840`
- `#define TIM_LONG_MIN "-180.000000"`
- `#define TIM_LONG_MAX "180.000000"`
- `#define TIM_LONG_NAV "2125315840"`
- `#define TIM_LONG_MIN_NUM -180.000000`
- `#define TIM_LONG_MAX_NUM 180.000000`
- `#define TIM_LONG_NAV_NUM 2125315840`
- `#define TIM_ELEV_MIN "-410.0"`
- `#define TIM_ELEV_MAX "6144.0"`
- `#define TIM_ELEV_NAV "2125315840"`
- `#define TIM_ELEV_MIN_NUM -410.0`
- `#define TIM_ELEV_MAX_NUM 6144.0`
- `#define TIM_ELEV_NAV_NUM 2125315840`
- `#define TIM_LANEWIDTH_MIN "0"`
- `#define TIM_LANEWIDTH_MAX "32767"`
- `#define TIM_LANEWIDTH_NAV "2125315840"`
- `#define TIM_LANEWIDTH_MIN_NUM 0`
- `#define TIM_LANEWIDTH_MAX_NUM 32767`
- `#define TIM_LANEWIDTH_NAV_NUM 2125315840`
- `#define TIM_DIRECTION_NAV_NUM 2125315840`

- #define TIM\_FRAME\_COUNT\_MAX 8
- #define TIM\_MAX\_VALID\_REGIONS 64
- #define TIM\_MAX\_ADVISORIES 100
- #define TIM\_MAX\_WORKZONES 16
- #define TIM\_MAX\_GENERICSIGNS 16
- #define TIM\_MAX\_SPEEDLIMITS 16
- #define TIM\_MAX\_EXITSERVICES 16
- #define SHORT\_INT 2
- #define j2735\_free\_tim\_contents\_only(a) free\_tim(a, 1)
- #define j2735\_free\_tim\_struct(a) free\_tim(a, 0)
- #define TIM\_UNIQID\_MIN "0x00000000"
- #define TIM\_UNIQID\_MAX "0xF3B0448"
- #define TIM\_UNIQID\_NAV "0xF3B0449"
- #define TIM\_UNIQID\_MIN\_NUM 0x00000000
- #define TIM\_UNIQID\_MAX\_NUM 0xF3B0448
- #define TIM\_UNIQID\_NAV\_NUM 0xF3B0449
- #define TIM\_UNIQID\_LEN\_MIN 0
- #define TIM\_UNIQID\_LEN\_MAX 9
- #define TIM\_UNIQID\_LEN\_NAV J2735\_NAV
- #define TIM\_NUMDF\_MIN "1"
- #define TIM\_NUMDF\_MAX "8"
- #define TIM\_NUMDF\_NAV "10"
- #define TIM\_NUMDF\_NAV\_NUM 10
- #define TIM\_DFTYPE\_MIN "1"
- #define TIM\_DFTYPE\_MAX "2"
- #define TIM\_DFTYPE\_NAV "-1"
- #define TIM\_DFTYPE\_NAV\_NUM -1
- #define TIM\_ADV\_ID\_MIN "1"
- #define TIM\_ADV\_ID\_MAX "65535"
- #define TIM\_ADV\_ID\_NAV "0"
- #define TIM\_ADV\_ID\_MIN\_NUM 1
- #define TIM\_ADV\_ID\_MAX\_NUM 65535
- #define TIM\_ADV\_ID\_NAV\_NUM 0
- #define TIM\_TXINTVL\_MIN "100"
- #define TIM\_TXINTVL\_MAX "10000"
- #define TIM\_TXINTVL\_NAV "2125315823"
- #define TIM\_TXINTVL\_MIN\_NUM 100
- #define TIM\_TXINTVL\_MAX\_NUM 10000
- #define TIM\_TXINTVL\_NAV\_NUM 2125315823
- #define TIM\_START\_YEAR\_MIN "0000"
- #define TIM\_START\_YEAR\_MAX "4095"
- #define TIM\_START\_YEAR\_NAV "-1"
- #define TIM\_START\_YEAR\_MIN\_NUM 0
- #define TIM\_START\_YEAR\_MAX\_NUM 4095
- #define TIM\_START\_YEAR\_NAV\_NUM J2735\_NAV
- #define TIM\_STARTMIN\_YEAR\_MIN "0"
- #define TIM\_STARTMIN\_YEAR\_MAX "525960"
- #define TIM\_STARTMIN\_YEAR\_NAV "2125315823"
- #define TIM\_STARTMIN\_YEAR\_MIN\_NUM 0
- #define TIM\_STARTMIN\_YEAR\_MAX\_NUM 525960
- #define TIM\_STARTMIN\_YEAR\_NAV\_NUM 2125315823
- #define TIM\_STARTDATE\_MIN "01/2011/01"
- #define TIM\_STARTDATE\_MAX "12/9999/31"
- #define TIM\_STARTDATE\_NAV "14/10000/32"
- #define TIM\_ENDDATE\_MIN TIM\_STARTDATE\_MIN

- `#define TIM_ENDDATE_MAX TIM_STARTDATE_MAX`
- `#define TIM_ENDDATE_NAV TIM_STARTDATE_NAV`
- `#define TIM_STARTTIME_MIN "00:00"`
- `#define TIM_STARTTIME_MAX "23:59"`
- `#define TIM_STARTTIME_NAV "24:00"`
- `#define TIM_ENDTIME_MIN TIM_STARTTIME_MIN`
- `#define TIM_ENDTIME_MAX TIM_STARTTIME_MAX`
- `#define TIM_ENDTIME_NAV TIM_STARTTIME_NAV`
- `#define TIM_STARTDUR_MIN "1"`
- `#define TIM_STARTDUR_MAX "65000"`
- `#define TIM_STARTDUR_NAV "2125315823"`
- `#define TIM_STARTDUR_MIN_NUM 1`
- `#define TIM_STARTDUR_MAX_NUM 65000`
- `#define TIM_STARTDUR_NAV_NUM 2125315823`
- `#define TIM_REPDAYS_MIN "1"`
- `#define TIM_REPDAYS_MAX "127"`
- `#define TIM_REPDAYS_NAV "-1"`
- `#define TIM_REPDAYS_MIN_NUM 1`
- `#define TIM_REPDAYS_MAX_NUM 127`
- `#define TIM_REPDAYS_NAV_NUM -1`
- `#define TIM_PRIORITY_MIN "0"`
- `#define TIM_PRIORITY_MAX "7"`
- `#define TIM_PRIORITY_NAV "2125315823"`
- `#define TIM_PRIORITY_MIN_NUM 0`
- `#define TIM_PRIORITY_MAX_NUM 7`
- `#define TIM_PRIORITY_NAV_NUM 2125315823`
- `#define TIM_ITIS_CODES_TEXTS_MIN "0"`
- `#define TIM_ITIS_CODES_TEXTS_MAX "65565"`
- `#define TIM_ITIS_CODES_TEXTS_NAV "2125315823"`
- `#define TIM_ITIS_CODES_TEXTS_MIN_NUM 0`
- `#define TIM_ITIS_CODES_TEXTS_MAX_NUM 65565`
- `#define TIM_ITIS_CODES_TEXTS_NAV_NUM 2125315823`
- `#define TIM_NUMVALID_REGS_MIN "1"`
- `#define TIM_NUMVALID_REGS_MAX "16"`
- `#define TIM_NUMVALID_REGS_NAV "0"`
- `#define TIM_NUMVALID_REGS_MIN_NUM 1`
- `#define TIM_NUMVALID_REGS_MAX_NUM 16`
- `#define TIM_NUMVALID_REGS_NAV_NUM 0`
- `#define TIM_REGTYPE_MIN "0"`
- `#define TIM_REGTYPE_MAX "3"`
- `#define TIM_REGTYPE_NAV "4"`
- `#define TIM_REGTYPE_MIN_NUM 0`
- `#define TIM_REGTYPE_MAX_NUM 3`
- `#define TIM_REGTYPE_NAV_NUM 4`
- `#define TIM_NUMNODES_MIN "1"`
- `#define TIM_NUMNODES_MAX "64"`
- `#define TIM_NUMNODES_NAV "65"`
- `#define TIM_NUMNODES_MIN_NUM 1`
- `#define TIM_NUMNODES_MAX_NUM 64`
- `#define TIM_NUMNODES_NAV_NUM 65`
- `#define TIM_HSL_MIN "0"`
- `#define TIM_HSL_MAX "360"`
- `#define TIM_HSL_NAV "2125315823"`
- `#define TIM_HSL_MIN_NUM 0`
- `#define TIM_HSL_MAX_NUM 360`

- #define TIM\_HSL\_NAV\_NUM 2125315823
- #define TIM\_DFCONTENT\_MIN "1"
- #define TIM\_DFCONTENT\_MAX "5"
- #define TIM\_DFCONTENT\_NAV "0"
- #define TIM\_DFCONTENT\_MIN\_NUM 1
- #define TIM\_DFCONTENT\_MAX\_NUM 5
- #define TIM\_DFCONTENT\_NAV\_NUM 0
- #define TIM\_NUMADVISORY\_MIN "1"
- #define TIM\_NUMADVISORY\_MAX "100"
- #define TIM\_NUMADVISORY\_NAV "101"
- #define TIM\_NUMADVISORY\_MIN\_NUM 1
- #define TIM\_NUMADVISORY\_MAX\_NUM 100
- #define TIM\_NUMADVISORY\_NAV\_NUM 101
- #define TIM\_NUMROADSIGN\_MIN "1"
- #define TIM\_NUMROADSIGN\_MAX "10"
- #define TIM\_NUMROADSIGN\_NAV "11"
- #define TIM\_NUMROADSIGN\_MIN\_NUM 1
- #define TIM\_NUMROADSIGN\_MAX\_NUM 10
- #define TIM\_NUMROADSIGN\_NAV\_NUM 11
- #define TIM\_MUTCDCODE\_MIN "0"
- #define TIM\_MUTCDCODE\_MAX "6"
- #define TIM\_MUTCDCODE\_NAV "2125315823"
- #define TIM\_MUTCDCODE\_MIN\_NUM 0
- #define TIM\_MUTCDCODE\_MAX\_NUM 6
- #define TIM\_MUTCDCODE\_NAV\_NUM 2125315823
- #define TIM\_DIRO\_USE\_MIN "0"
- #define TIM\_DIRO\_USE\_MAX "2"
- #define TIM\_DIRO\_USE\_NAV "2125315840"
- #define TIM\_DIRO\_USE\_ASN\_NAV 0
- #define TIM\_DIRO\_USE\_MIN\_NUM 1
- #define TIM\_DIRO\_USE\_MAX\_NUM 3
- #define TIM\_DIRO\_USE\_NAV\_NUM 2125315840
- #define TIM\_EXTENT\_MIN "0"
- #define TIM\_EXTENT\_MAX "100001"
- #define TIM\_EXTENT\_NAV "2125315823"
- #define TIM\_EXTENT\_MIN\_NUM 0
- #define TIM\_EXTENT\_MAX\_NUM 100001
- #define TIM\_EXTENT\_NAV\_NUM 2125315823
- #define TIM\_XOFF\_MIN "-32767"
- #define TIM\_XOFF\_MAX "32767"
- #define TIM\_XOFF\_NAV "2125315840"
- #define TIM\_XOFF\_MIN\_NUM -32767
- #define TIM\_XOFF\_MAX\_NUM 32767
- #define TIM\_XOFF\_NAV\_NUM 2125315840
- #define TIM\_YOFF\_MIN "-32767"
- #define TIM\_YOFF\_MAX "32767"
- #define TIM\_YOFF\_NAV "2125315840"
- #define TIM\_YOFF\_MIN\_NUM -32767
- #define TIM\_YOFF\_MAX\_NUM 32767
- #define TIM\_YOFF\_NAV\_NUM 2125315840
- #define TIM\_ZOFF\_MIN "-32767"
- #define TIM\_ZOFF\_MAX "32767"
- #define TIM\_ZOFF\_NAV "2125315840"
- #define TIM\_ZOFF\_MIN\_NUM -32767
- #define TIM\_ZOFF\_MAX\_NUM 32767

- `#define TIM_ZOFF_NAV_NUM 2125315840`
- `#define TIM_CIRCLE_RADIUS_STEPS 2.5`
- `#define TIM_CIRCLE_RADIUS_STEPS_MIN 0`
- `#define TIM_CIRCLE_RADIUS_STEPS_MAX 32767`
- `#define TIM_CIRCLE_RADIUSMILES_MIN 1`
- `#define TIM_CIRCLE_RADIUSMILES_MAX 2000`
- `#define TIM_CIRCLE_RADIUSKMS_MIN 1`
- `#define TIM_CIRCLE_RADIUSKMS_MAX 5000`
- `#define TIM_MAX_REGION_OFFSETS 64`
- `#define TIM_MSG_CNT_MIN 0`
- `#define TIM_MSG_CNT_MAX 127`
- `#define TIM_MSG_CNT_NAV J2735_NAV`
- `#define TIM_DATA_FRAME_COUNT_MIN 1`
- `#define TIM_DATA_FRAME_COUNT_MAX 8`
- `#define TIM_DATA_FRAME_COUNT_NAV J2735_NAV`

## Typedefs

- typedef enum [J2735ContentType](#) [J2735ContentType\\_t](#)
- typedef enum [J2735TravelerInfoType](#) [J2735TravelerInfoType\\_t](#)
- typedef enum [J2735MUTCDCode](#) [J2735MUTCDCode\\_t](#)
- typedef struct [J2735RoadSignID](#) [J2735RoadSignID\\_t](#)
- typedef enum [J2735DirectionOfUse](#) [J2735DirectionOfUse\\_t](#)
- typedef enum [J2735Regions](#) [J2735RegionsList\\_t](#)
- typedef struct [J2735ShapePointSet](#) [J2735ShapePoints\\_t](#)
- typedef enum [J2735RadiusType](#) [J2735RadiusType\\_t](#)
- typedef struct [J2735CircularRegion](#) [J2735CircularRegion\\_t](#)
- typedef struct [J2735RegionPointSet](#) [J2735RegionPointSet\\_t](#)
- typedef struct [J2735ValidRegion](#) [J2735ValidRegion\\_t](#)
- typedef struct [RoadSegmentRefId](#) [RoadSegmentRefId\\_t](#)
- typedef struct [J2735\\_2016\\_NodeSet\\_XY](#) [J2735\\_2016\\_NodeSet\\_XY\\_t](#)
- typedef struct [J2735\\_2016\\_Computed\\_Lanes](#) [J2735\\_2016\\_Computed\\_Lanes\\_t](#)
- typedef struct [J2735\\_2016\\_NodeXY](#) [J2735\\_2016\\_NodeXY\\_t](#)
- typedef struct [J2735\\_2016\\_LatLong](#) [J2735\\_2016\\_LatLong\\_t](#)
- typedef struct [J2735\\_2016\\_OffsetSystem](#) [J2735\\_2016\\_OffsetSystem\\_t](#)
- typedef struct [J2735\\_2016\\_GeometricProjection](#) [J2735\\_2016\\_GeometricProjection\\_t](#)
- typedef struct [J2735TIMGeoPath](#) [J2735TIMGeoPath\\_t](#)
- typedef struct [J2735TIM](#) [J2735DATAFRAME\\_t](#)
- typedef struct [v2x\\_msg\\_tim](#) [v2x\\_msg\\_tim\\_t](#)

## Enumerations

- enum [J2735ContentType](#) {  
[J2735\\_CONTENT\\_ADVISORY](#) = 1, [J2735\\_CONTENT\\_WORKZONES](#),  
[J2735\\_CONTENT\\_GENERIC\\_SIGNS](#), [J2735\\_CONTENT\\_SPEED\\_LIMITS](#),  
[J2735\\_CONTENT\\_EXIT\\_SERVICES](#) }
- enum [J2735TravelerInfoType](#) {  
[J2735\\_TIM\\_TYPE\\_UNKNOWN](#), [J2735\\_TIM\\_TYPE\\_ADVISORY](#),  
[J2735\\_TIM\\_TYPE\\_ROAD\\_SIGNAGE](#), [J2735\\_TIM\\_TYPE\\_COMMERCIAL\\_SIGNAGE](#) }
- enum [J2735HeadingSlice](#)

- enum J2735MUTCDCode {  
J2735\_MUTCD\_CODE\_NONE, J2735\_MUTCD\_CODE\_REGULATORY,  
J2735\_MUTCD\_CODE\_WARNING, J2735\_MUTCD\_CODE\_MAINTENANCE,  
J2735\_MUTCD\_CODE\_MOTORIST\_SERVICE, J2735\_MUTCD\_CODE\_GUIDE,  
J2735\_MUTCD\_CODE\_REC }
- enum J2735DirectionOfUse {  
J2735\_DIRECTION\_UNAVAIL = 0, J2735\_DIRECTION\_FORWARD,  
J2735\_DIRECTION\_REVERSE, J2735\_DIRECTION\_BOTH }
- enum J2735Regions {  
J2735\_REGION\_NOTHING, J2735\_REGION\_SHAPEPT,  
J2735\_REGION\_CIRCLE, J2735\_REGION\_POLYGON }
- enum J2735RadiusType {  
RADIUS\_TYPE\_CENTIMETER, RADIUS\_TYPE\_CM\_2\_5,  
RADIUS\_TYPE\_DECIMETER, RADIUS\_TYPE\_METER,  
RADIUS\_TYPE\_KILOMETER, RADIUS\_TYPE\_FOOT,  
RADIUS\_TYPE\_YARD, RADIUS\_TYPE\_MILE }
- enum J2735\_2016\_ProjectionType\_t {  
J2735\_2016\_PROJECTION\_OFFSET\_SYSTEM, J2735\_2016\_PROJECTION\_GEO\_PROJECTION,  
J2735\_2016\_PROJECTION\_VALID\_REGION }
- enum J2735\_2016\_OffsetSystemType\_t { J2735\_2016\_OFFSET\_TYPE\_XY, J2735\_2016\_OFFSET\_TYPE\_LL }
- enum J2735\_2016\_NodeXY\_type\_t { J2735\_2016\_NODESET\_XY, J2735\_2016\_COMPUTED\_LANES }

## Functions

- enum J2735HeadingSlice \_\_attribute\_\_((packed)) J2735HeadingSlice\_t
- void free\_tim (J2735TIMMsg\_t \*tim, int flag)
- v2x\_status\_t v2x\_msg\_tim\_encode (v2x\_msg\_tim\_t \*tim, int type, uint8\_t \*msgbuf, int msgbuf\_size, int \*enclen)
- int j2735\_encode\_tim (J2735TIMMsg\_t \*tim, int type, uint8\_t \*msgbuf, int msgbuf\_size)
- v2x\_status\_t v2x\_msg\_tim\_decode (v2x\_msg\_tim\_t \*tim, int type, uint8\_t \*msgbuf, int msgbuf\_len, uint32\_t \*oob, int \*declen, int debug)
- void v2x\_msg\_tim\_free (v2x\_msg\_tim\_t \*tim)
- void j2735\_2016\_free\_tim (J2735TIMMsg\_t \*tim, int flag)
- int j2735\_decode\_tim (J2735TIMMsg\_t \*tim, int type, uint8\_t \*msgbuf, int msgbuf\_len, uint32\_t \*oob, int debug)
- void j2735\_2016\_print\_tim (v2x\_msg\_tim\_t \*tim, FILE \*fp)

## Variables

- J2735\_HS\_NO\_HEADING = 0x0
- J2735\_HS\_ALL\_HEADINGS = 0xffff
- J2735\_HS\_FROM000\_0TO022\_5DEGREES = 0x0001
- J2735\_HS\_FROM022\_5TO045\_0DEGREES = 0x0002
- J2735\_HS\_FROM045\_0TO067\_5DEGREES = 0x0004
- J2735\_HS\_FROM067\_5TO090\_0DEGREES = 0x0008
- J2735\_HS\_FROM090\_0TO112\_5DEGREES = 0x0010
- J2735\_HS\_FROM112\_5TO135\_0DEGREES = 0x0020
- J2735\_HS\_FROM135\_0TO157\_5DEGREES = 0x0040
- J2735\_HS\_FROM157\_5TO180\_0DEGREES = 0x0080
- J2735\_HS\_FROM180\_0TO202\_5DEGREES = 0x0100
- J2735\_HS\_FROM202\_5TO225\_0DEGREES = 0x0200
- J2735\_HS\_FROM225\_0TO247\_5DEGREES = 0x0400
- J2735\_HS\_FROM247\_5TO270\_0DEGREES = 0x0800
- J2735\_HS\_FROM270\_0TO292\_5DEGREES = 0x1000



- [J2735\\_HS\\_FROM292\\_5TO315\\_0DEGREES](#) = 0x2000
- [J2735\\_HS\\_FROM315\\_0TO337\\_5DEGREES](#) = 0x4000
- [J2735\\_HS\\_FROM337\\_5TO360\\_0DEGREES](#) = 0x8000

### 7.19.1 Detailed Description

TIM message API definitions and functions. This header file contains the APIs and data structures used to encode or decode a Traveller Information Message (TIM).

Include [v2x\\_msg\\_tim.h](#) and link with -lj2735 -lsae -lm.

### 7.19.2 Macro Definition Documentation

7.19.2.1 `#define j2735_free_tim_contents_only( a ) free_tim(a, 1)`

`j2735_free_tim_contents_only` - frees the contents of the tim structure not the tim itself this assumes that the user passed structure is not an allocated one thats why this macro never frees the tim structure, but frees its contents.

7.19.2.2 `#define j2735_free_tim_struct( a ) free_tim(a, 0)`

`j2735_free_tim_struct` - frees the entire tim structure this assumes that the user passed structure is allocated one thats why this macro frees out the whole datastructure. One of the free macros must be called after `j2735_decode_tim()`

### 7.19.3 Typedef Documentation

7.19.3.1 `typedef struct J2735_2016_Computed_Lanes J2735_2016_Computed_Lanes_t`

`J2735_2016_Computed_Lanes_t` - not supported

7.19.3.2 `typedef struct J2735_2016_GeometricProjection J2735_2016_GeometricProjection_t`

`J2735_2016_GeometricProjection_t` - geometric projection structure The geometric projection is used to define simple geometric shapes.

7.19.3.3 `typedef struct J2735_2016_LatLong J2735_2016_LatLong_t`

`J2735_2016_LatLong_t` - set of lat and long offsets describing the LL offsets

7.19.3.4 `typedef struct J2735_2016_NodeSet_XY J2735_2016_NodeSet_XY_t`

`J2735_2016_NodeSet_XY_t` - NodeSet list used to describe a set of nodes

7.19.3.5 `typedef struct J2735_2016_NodeXY J2735_2016_NodeXY_t`

`J2735_2016_NodeXY_t` - Node XY data structure enclosing the nodes

7.19.3.6 `typedef struct J2735_2016_OffsetSystem J2735_2016_OffsetSystem_t`

`J2735_2016_OffsetSystem_t` - offset system describing the nodes in detail either in xy or ll

**7.19.3.7 typedef struct J2735CircularRegion J2735CircularRegion\_t**

J2735CircularRegion\_t - Circular region

denotes the valid region in the form of a circle.

**7.19.3.8 typedef enum J2735ContentType J2735ContentType\_t**

TIM content types

**7.19.3.9 typedef struct J2735TIM J2735DATAFRAME\_t**

J2735DATAFRAME\_t - dataframe of TIM message

**7.19.3.10 typedef enum J2735DirectionOfUse J2735DirectionOfUse\_t**

J2735DirectionOfUse\_t - direction of use

**7.19.3.11 typedef enum J2735MUTCDCode J2735MUTCDCode\_t**

MUTCD Code information.

**7.19.3.12 typedef enum J2735RadiusType J2735RadiusType\_t**

J2735RadiusType\_t - Radius type enums

**7.19.3.13 typedef enum J2735Regions J2735RegionsList\_t**

J2735RegionsList\_t - region of use

**7.19.3.14 typedef struct J2735RoadSignID J2735RoadSignID\_t**

RoadSign ID

**7.19.3.15 typedef struct J2735ShapePointSet J2735ShapePoints\_t**

J2735ShapePoints\_t - shapepoint set region the shapepoint region is a region formed by connecting the lat, long and elevation points. these are found by adding offsets to the anchor point.

**7.19.3.16 typedef struct J2735TIMGeoPath J2735TIMGeoPath\_t**

J2735TIMGeoPath\_t - geographic path representation of the TIM message this allows TIM to specifically inform the valid regions of the message. The geographic path consists of different set of regions to describe the validity.

**7.19.3.17 typedef enum J2735TravelerInfoType J2735TravelerInfoType\_t**

Traveller information types.

## 7.19.3.18 typedef struct J2735ValidRegion J2735ValidRegion\_t

J2735ValidRegion\_t - valid region, as of now only J2735\_REGION\_SHAPEPT is supported for a valid\_region.

As of 2016 J2735 standard, the Valid regions are legacy and shall not be used

## 7.19.3.19 typedef struct RoadSegmentRefId RoadSegmentRefId\_t

RoadSegmentRefId\_t - RoadSegment ID is unique to a particular region of road segment

## 7.19.4 Enumeration Type Documentation

## 7.19.4.1 enum J2735\_2016\_NodeXY\_type\_t

J2735\_2016\_NodeXY\_type\_t - Node XY type

Enumerator

**J2735\_2016\_NODESET\_XY** Node XY

**J2735\_2016\_COMPUTED\_LANES** Computed Lanes - not supported

## 7.19.4.2 enum J2735\_2016\_OffsetSystemType\_t

J2735\_2016\_OffsetSystemType\_t - Offset system type

Enumerator

**J2735\_2016\_OFFSET\_TYPE\_XY** XY offsets

**J2735\_2016\_OFFSET\_TYPE\_LL** LL offsets - not supported

## 7.19.4.3 enum J2735\_2016\_ProjectionType\_t

J2735\_2016\_ProjectionType\_t - Projection type

Enumerator

**J2735\_2016\_PROJECTION\_OFFSET\_SYSTEM** Offset system

**J2735\_2016\_PROJECTION\_GEO\_PROJECTION** Geometric projection

**J2735\_2016\_PROJECTION\_VALID\_REGION** Valid region - Legacy valid region

## 7.19.4.4 enum J2735ContentType

TIM content types

Enumerator

**J2735\_CONTENT\_ADVISORY** Advisory Content type

**J2735\_CONTENT\_WORKZONES** WorkZone Content type

**J2735\_CONTENT\_GENERIC\_SIGNS** Generic Sign Content type

**J2735\_CONTENT\_SPEED\_LIMITS** SpeedLimit Content type

**J2735\_CONTENT\_EXIT\_SERVICES** Exit service Content type

#### 7.19.4.5 enum J2735DirectionOfUse

J2735DirectionOfUse\_t - direction of use

##### Enumerator

**J2735\_DIRECTION\_UNAVAIL** Direction unavailable  
**J2735\_DIRECTION\_FORWARD** Forward direction case  
**J2735\_DIRECTION\_REVERSE** Reverse direction case  
**J2735\_DIRECTION\_BOTH** Both the cases

#### 7.19.4.6 enum J2735HeadingSlice

Heading slice information. Refer to DE\_HeadingSlice object in SAE J735 for details.

#### 7.19.4.7 enum J2735MUTCDCode

MUTCD Code information.

##### Enumerator

**J2735\_MUTCD\_CODE\_NONE** No MUTCD information.  
**J2735\_MUTCD\_CODE\_REGULATORY** Regulatory signs.  
**J2735\_MUTCD\_CODE\_WARNING** Warning signs.  
**J2735\_MUTCD\_CODE\_MAINTENANCE** Maintenance and construction.  
**J2735\_MUTCD\_CODE\_MOTORIST\_SERVICE** Motorist Services.  
**J2735\_MUTCD\_CODE\_GUIDE** Guide signs.  
**J2735\_MUTCD\_CODE\_REC** Recreation and Cultural Interest.

#### 7.19.4.8 enum J2735RadiusType

J2735RadiusType\_t - Radius type enums

##### Enumerator

**RADIUS\_TYPE\_CENTIMETER** RadiusSteps range 0 - 32767, in steps of 2.5 cm. the library takes care of converting the value in to steps of 2.5 cm and if the value is less than 0 it sets it back to 0 and if its more than 32767 it sets it to 32767 Centimeters unit  
**RADIUS\_TYPE\_CM\_2\_5** In steps of 2.5 cms  
**RADIUS\_TYPE\_DECIMETER** In steps of decimeters  
**RADIUS\_TYPE\_METER** In steps of meters  
**RADIUS\_TYPE\_KILOMETER** In steps of kilometers  
**RADIUS\_TYPE\_FOOT** In steps of foot  
**RADIUS\_TYPE\_YARD** In steps of yard  
**RADIUS\_TYPE\_MILE** In steps of mile

## 7.19.4.9 enum J2735Regions

J2735RegionsList\_t - region of use

## Enumerator

**J2735\_REGION\_NOTHING** No Region  
**J2735\_REGION\_SHAPEPT** Shapepoint region  
**J2735\_REGION\_CIRCLE** Circular region  
**J2735\_REGION\_POLYGON** Polygon region

## 7.19.4.10 enum J2735TravelerInfoType

Traveller information types.

## Enumerator

**J2735\_TIM\_TYPE\_UNKNOWN** TIM type is unknown  
**J2735\_TIM\_TYPE\_ADVISORY** TIM type is advisory  
**J2735\_TIM\_TYPE\_ROAD\_SIGNAGE** TIM type is road signage  
**J2735\_TIM\_TYPE\_COMMERCIAL\_SIGNAGE** TIM type is commercial signage

## 7.19.5 Function Documentation

## 7.19.5.1 enum J2735HeadingSlice \_\_attribute\_\_((packed))

Heading slice information. Refer to DE\_HeadingSlice object in SAE J735 for details.

## 7.19.5.2 void free\_tim ( J2735TIMMsg\_t \* tim, int flag )

free\_tim - This routine frees TIM structure

## Note

- It is advised not to use this function directly. use the j2735\_free\_tim\_contents\_only and j2735\_free\_tim\_struct macros instead

## 7.19.6 Variable Documentation

## 7.19.6.1 J2735\_HS\_ALL\_HEADINGS = 0xffff

All Heading values

## 7.19.6.2 J2735\_HS\_FROM000\_0TO022\_5DEGREES = 0x0001

Heading range 0 - 22.5 deg

## 7.19.6.3 J2735\_HS\_FROM022\_5TO045\_0DEGREES = 0x0002

Heading range 22.5 - 45 deg

7.19.6.4 J2735\_HS\_FROM045\_0TO067\_5DEGREES = 0x0004

Heading range 45 - 67.5 deg

7.19.6.5 J2735\_HS\_FROM067\_5TO090\_0DEGREES = 0x0008

Heading range 67.5 - 90 deg

7.19.6.6 J2735\_HS\_FROM090\_0TO112\_5DEGREES = 0x0010

Heading range 90 - 112.5 deg

7.19.6.7 J2735\_HS\_FROM112\_5TO135\_0DEGREES = 0x0020

Heading range 112.5 - 135 deg

7.19.6.8 J2735\_HS\_FROM135\_0TO157\_5DEGREES = 0x0040

Heading range 135 - 157.5 deg

7.19.6.9 J2735\_HS\_FROM157\_5TO180\_0DEGREES = 0x0080

Heading range 157.5 - 180 deg

7.19.6.10 J2735\_HS\_FROM180\_0TO202\_5DEGREES = 0x0100

Heading range 180 - 202.5 deg

7.19.6.11 J2735\_HS\_FROM202\_5TO225\_0DEGREES = 0x0200

Heading range 202.5 - 225 deg

7.19.6.12 J2735\_HS\_FROM225\_0TO247\_5DEGREES = 0x0400

Heading range 225 - 247.5 deg

7.19.6.13 J2735\_HS\_FROM247\_5TO270\_0DEGREES = 0x0800

Heading range 247.5 - 270 deg

7.19.6.14 J2735\_HS\_FROM270\_0TO292\_5DEGREES = 0x1000

Heading range 270 - 292.5 deg

7.19.6.15 J2735\_HS\_FROM292\_5TO315\_0DEGREES = 0x2000

Heading range 292.5 - 315 deg

7.19.6.16 J2735\_HS\_FROM315\_0TO337\_5DEGREES = 0x4000

Heading range 315 - 337.5 deg

7.19.6.17 J2735\_HS\_FROM337\_5TO360\_0DEGREES = 0x8000

Heading range 337.5 - 360 deg

7.19.6.18 J2735\_HS\_NO\_HEADING = 0x0

No Heading

## 7.20 v2x\_net\_btp.h File Reference

BTP API definitions and functions.

```
#include "v2x_btp_intf.h"
#include "libbtp.h"
#include "v2x_error.h"
```

### Macros

- `#define v2x_btp_reserve BTP_RESERVE`
- `#define v2x_btp_reserve_data BTP_RESERVE_DATA`
- `#define v2x_btp_get_txreq BTP_GET_TXREQ`
- `#define v2x_btp_get_data BTP_GET_DATA_PTR`
- `#define v2x_btp_packet_len BTP_PACKET_LEN`

### Functions

- `v2x_status_t v2x_btp_init` (int port, void \*\*btp\_handle)
- `v2x_status_t v2x_btp_get_sock_id` (void \*btp\_handler, int \*sock)
- `v2x_status_t v2x_btp_transmit` (void \*btp\_handler, BTPReq\_t \*req)
- `v2x_status_t v2x_btp_receive` (void \*btp\_handler, void \*app\_ctx, void(\*recv\_callback)(BTPInd\_t \*ind, void \*app\_ctx))
- `v2x_status_t v2x_btp_deinit` (void \*btp\_handle)

### 7.20.1 Detailed Description

BTP API definitions and functions. This header file contains APIs and macros for BTP services

Include v2x\_btp.h . Link with -lbtp -lgeonet

### 7.20.2 Macro Definition Documentation

7.20.2.1 `#define v2x_btp_get_data BTP_GET_DATA_PTR`

Gives a pointer to the application payload in the buffer of type BTPReq\_t\*

#### 7.20.2.2 `#define v2x_btp_get_txreq BTP_GET_TXREQ`

Typecasts the buffer to `BTPReq_t *`

#### 7.20.2.3 `#define v2x_btp_packet_len BTP_PACKET_LEN`

Gives the length of the application payload including BTP header length

#### 7.20.2.4 `#define v2x_btp_reserve BTP_RESERVE`

Gives a pointer to application payload in the buffer of type `BTPReq_t*`

#### 7.20.2.5 `#define v2x_btp_reserve_data BTP_RESERVE_DATA`

Gives a pointer to application payload in the buffer of type `BTPReq_t*` and typecasts to `uint8_t*`

## 7.21 `v2x_net_wme.h` File Reference

WME API definitions and functions.

```
#include <stdint.h>
#include "v2x_error.h"
```

### Data Structures

- struct [v2x\\_wme\\_reg\\_req](#)
- struct [libwme\\_gpsinfo](#)
- struct [v2x\\_wsmp\\_rcv\\_indication](#)
- struct [savari1609Wra](#)
- struct [savari1609WsaService](#)
- struct [v2x\\_wsmp\\_send\\_req](#)
- struct [libwme\\_radio\\_settings](#)
- struct [libwme\\_radio\\_settings::libwme\\_radio](#)
- struct [v2x\\_wsmp\\_callbacks](#)

### Macros

- `#define SUCCESS 0`
- `#define FAIL -1`
- `#define PSID_LEN 4`
- `#define SSI_LEN 16`
- `#define ADVERTISER_LEN 127`
- `#define P1609_ADVID_LEN_MAX 31`
- `#define SAVARI_MAC_LEN 6`
- `#define MAX_DATA 2000`
- `#define WAVE_ELEMID_WSMP 128`
- `#define WAVE_ELEMID_WSMPS 129`
- `#define LIBWME_PSC_STRING_LENGTH 31`
- `#define SAVARI1609_IEEE80211_ADDR_LEN 6`
- `#define SAVARI1609_MAXLINE 2000`
- `#define P16093_SRV_IP ":::1"`



- `#define P16093_RADIO_INTF1 "ath0"`
- `#define P16093_RADIO_INTF2 "ath1"`
- `#define P16093_RADIO_INTFX "athx"`
- `#define LIBWME_CHAN_MODE_CONT 1`
- `#define LIBWME_CHAN_MODE_ALT 2`
- `#define savariwme_reg_req v2x_wme_reg_req`
- `#define savariwme_rx_indication v2x_wsmp_rcv_indication`
- `#define RCPI_THRESHOLD_PRESENT 0x80`
- `#define WSA_COUNT_THRESHOLD_PRESENT 0x40`
- `#define WSA_COUNT_THRESHOLD_INTVL_PRESENT 0x20`
- `#define savariwme_tx_req v2x_wsmp_send_req`
- `#define LIBWME_RADIO_MAX 2`
- `#define savariwme_cbs v2x_wsmp_callbacks`

## Typedefs

- `typedef int savari_socket_desc_t`
- `typedef savari_socket_desc_t savari_wme_handler_t`
- `typedef savari_socket_desc_t v2x_wsmp_handler_t`
- `typedef struct v2x_wme_reg_req v2x_wme_reg_req_t`
- `typedef struct`  
`v2x_wsmp_rcv_indication v2x_wsmp_rcv_indication_t`
- `typedef struct v2x_wsmp_send_req v2x_wsmp_send_req_t`
- `typedef struct v2x_wsmp_callbacks v2x_wsmp_callbacks_t`

## Enumerations

- `enum LIBWME_RC_RESULT_CODE {`  
`LIBWME_RC_ACCEPTED, LIBWME_RC_INVALID_PARAMETERS,`  
`LIBWME_RC_UNSPECIFIED }`
- `enum {`  
`LIBWME_USER_AUTOACCESS_ONMATCH, LIBWME_USER_AUTOACCESS_UNCOND,`  
`LIBWME_USER_NOSCHACCESS }`  
*Requested channel access type.*
- `enum {`  
`LIBWME_ACTION_ADD, LIBWME_ACTION_DELETE,`  
`LIBWME_ACTION_CHANGE }`  
*enums for the registration confirmation*
- `enum {`  
`P1609_CHANNEL_ACCESS_CONTINUOUS = 0, P1609_CHANNEL_ACCESS_ALTERNATING,`  
`P1609_CHANNEL_ACCESS_ONMATCH, P1609_CHANNEL_ACCESS_NOSCHACCESS,`  
`P1609_CHANNEL_ACCESS_UNKNOWN = 0xFF }`
- `enum {`  
`LIBWME_CCH_INTERVAL = 0x1, LIBWME_SCH_INTERVAL = 0x10,`  
`LIBWME_UNKNOWN_INTERVAL = 0xFF }`
- `enum { P1609_WSA_UNSECURED = 0, P1609_WSA_SECURED = 1 }`

## Functions

- `struct v2x_wme_reg_req __attribute__((packed))`
- `struct savari1609WsaService __attribute__((packed))`
- `v2x_status_t v2x_wme_open (char *wme_ip, char *iface, v2x_wsmp_handler_t *wme_handler)`
- `int wme_init (char *serverip, char *iface)`

- [v2x\\_status\\_t v2x\\_wme\\_register\\_user](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, [v2x\\_wme\\_reg\\_req\\_t](#) \*wme\_req)
- int [wme\\_register\\_user](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [savariwme\\_reg\\_req](#) \*wme\_req)
- int [wme\\_set\\_chnl\\_prm\\_sch](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [dot11p\\_cfg\\_n](#) \*dot11pcfg)
- [v2x\\_status\\_t v2x\\_wme\\_unregister\\_user](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, [v2x\\_wme\\_reg\\_req\\_t](#) \*wme\_req)
- void [wme\\_unregister\\_user](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [savariwme\\_reg\\_req](#) \*wme\_req)
- int [wme\\_register\\_cch\\_request](#) ([savari\\_wme\\_handler\\_t](#) handler, int intvl, int priority)  
*registers a CCH access request*
- int [wme\\_unregister\\_cch\\_request](#) ([savari\\_wme\\_handler\\_t](#) handler, int intvl, int priority)  
*unregisters CCH access request*
- [v2x\\_status\\_t v2x\\_wme\\_user\\_service\\_confirm](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, int action, [v2x\\_wme\\_reg\\_req\\_t](#) \*wme\_req)
- void [wme\\_user\\_service\\_confirm](#) ([savari\\_wme\\_handler\\_t](#) handler, int action, struct [savariwme\\_reg\\_req](#) \*wme\_req)
- int [wme\\_register\\_provider](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [savariwme\\_reg\\_req](#) \*wme\_req)  
*registers the provider application*
- void [wme\\_unregister\\_provider](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [savariwme\\_reg\\_req](#) \*wme\_req)  
*unregisters the provider*
- void [wme\\_provider\\_service\\_confirm](#) ([savari\\_wme\\_handler\\_t](#) handler, int action, struct [savariwme\\_reg\\_req](#) \*wme\_req)  
*confirm provider application to the 1609.3 stack*
- [v2x\\_status\\_t v2x\\_wsmp\\_send](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, [v2x\\_wsmp\\_send\\_req\\_t](#) \*msg\_param, uint8\_t \*msgbuf)
- int [wme\\_wsm\\_tx](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [savariwme\\_tx\\_req](#) \*wme\_wsm\_tx, uint8\_t \*tx\_buffer)
- [v2x\\_status\\_t v2x\\_wsmp\\_rcv](#) ([v2x\\_wsmp\\_handler\\_t](#) handler, [v2x\\_wsmp\\_callbacks\\_t](#) \*wme\_cbs, void \*ctx)
- int [wme\\_rx](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [savariwme\\_cbs](#) \*handle, void \*ctx)
- [v2x\\_status\\_t v2x\\_wsmp\\_convert\\_psid\\_be](#) (uint32\_t psid, uint32\_t \*psid\_be)
- uint32\_t [wme\\_convert\\_psid\\_be](#) (uint32\_t psid)
- int [wme\\_getpsidlen](#) (uint8\_t \*psid)  
*gets the psid length*
- int [wme\\_process\\_wme\\_cmd](#) ([savari\\_wme\\_handler\\_t](#) handler, int cmd, int subcmd)  
*Sends stack specific commands.*
- int [wme\\_set\\_gpslocation](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [libwme\\_gpsinfo](#) \*gpsinfo)  
*sets the gps location information to the 1609.3 stack*
- int [wme\\_set\\_inactivity\\_timer](#) ([savari\\_wme\\_handler\\_t](#) handler, int timer\_value)  
*sets the WRA information to the 1609.3stack*
- int [wme\\_set\\_wrainfo](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [savari1609Wra](#) \*wra)
- int [wme\\_set\\_wsatype](#) ([savari\\_wme\\_handler\\_t](#) handler, int)
- int [wme\\_set\\_repeatrate](#) ([savari\\_wme\\_handler\\_t](#) handler, int repeatrate)
- int [wme\\_get\\_wsa\\_cnt](#) ([savari\\_wme\\_handler\\_t](#) handler)
- int [wme\\_get\\_radio\\_cfg](#) ([savari\\_wme\\_handler\\_t](#) handler)
- int [wme\\_get\\_wsa\\_status](#) ([savari\\_wme\\_handler\\_t](#) handler)
- int [wme\\_set\\_wsaintvl](#) ([savari\\_wme\\_handler\\_t](#) handler, int wsa\_intvl)  
*sets the wsa interval*
- int [wme\\_set\\_wsatxpower](#) ([savari\\_wme\\_handler\\_t](#) handler, int txpower)  
*sets the txpower of a WSA*
- int [wme\\_set\\_advertiserid](#) ([savari\\_wme\\_handler\\_t](#) handler, char \*adv\_id)  
*sets the advertiser id*
- int [wme\\_send\\_wmectl](#) ([savari\\_wme\\_handler\\_t](#) handler, int cmd, int subcmd, void \*data, int len)
- void [v2x\\_wsmp\\_close](#) ([v2x\\_wsmp\\_handler\\_t](#) handler)
- void [wme\\_deinit](#) ([savari\\_wme\\_handler\\_t](#) handler)
- int [wme\\_set\\_filter\\_mode](#) ([savari\\_wme\\_handler\\_t](#) handler, uint8\_t \*psid, int psid\_len)
- int [wme\\_psid\\_match](#) (uint8\_t \*psid1, uint8\_t \*psid2)
- int [wme\\_get\\_radio\\_settings](#) ([savari\\_wme\\_handler\\_t](#) handler)
- int [wme\\_get\\_matched\\_rse\\_info](#) ([savari\\_wme\\_handler\\_t](#) handler, uint8\_t \*psid)
- int [wme\\_change\\_wsaservice](#) ([savari\\_wme\\_handler\\_t](#) handler, struct [savari1609WsaService](#) \*wsaservice)

## Variables

- enum { ... } [LIBWME\\_USERREQUEST](#)  
*Requested channel access type.*
- enum { ... } [LIBWME\\_USER\\_ACTION](#)  
*enums for the registration confirmation*
- int [channel](#)
- uint8\_t [srcmacaddr](#) [SAVARI\_MAC\_LEN]
- int [destmacaddr](#) [SAVARI\_MAC\_LEN]
- uint32\_t [psid](#)
- int [repeatrate](#)
- int [priority](#)
- int [request\\_type](#)
- int [extended\\_access](#)
- int [channel\\_access](#)
- int [immediate\\_access](#)
- int [wsatype](#)
- char [psc](#) [32]
- int [psc\\_length](#)
- int [local\\_service\\_index](#)
- int [ipservice](#)
- struct in6\_addr [service\\_ipv6addr](#)
- int [service\\_port](#)
- int [secondradio](#)
- char [advertiser\\_id](#) [ADVERTISER\_LEN]
- int [linkquality](#)
- int [latitude](#)
- int [longitude](#)
- int [elevation](#)
- double [pos\\_confidence](#)
- double [elev\\_confidence](#)
- int [positional\\_accuracy](#)
- int [version](#)
- uint64\_t [tstamp](#)
- int [plcp\\_length](#)
- int [txpower](#)
- int [datarate](#)
- uint8\_t [rx\\_mac](#) [SAVARI\_MAC\_LEN]
- uint32\_t [rx\\_buf\\_length](#)
- uint32\_t [rx\\_supp](#)
- int [rssi](#)
- int [num\\_rx](#)
- uint8\_t [rx\\_buf](#) [MAX\_DATA]
- uint16\_t [lifetime](#)
- struct in6\_addr [ipv6addr](#)
- uint8\_t [prefixlen](#)
- struct in6\_addr [default\\_gw](#)
- uint8\_t [gw\\_macaddr](#) [SAVARI\_MAC\_LEN]
- struct in6\_addr [primarydns](#)
- struct in6\_addr [secondarydns](#)
- uint8\_t [bit\\_mask](#)
- uint8\_t [radio](#)
- uint16\_t [port](#)
- char [contents](#) [LIBWME\_PSC\_STRING\_LENGTH]

- uint8\_t **provider\_mac** [SAVARI1609\_IEEE80211\_ADDR\_LEN]
- uint8\_t **rcpi\_threshold**
- uint8\_t **wsa\_count\_threshold**
- uint8\_t **wsa\_count\_threshold\_interval**
- uint8\_t **mac** [SAVARI1609\_IEEE80211\_ADDR\_LEN]
- uint8\_t **srcmac** [SAVARI1609\_IEEE80211\_ADDR\_LEN]
- int **expiry\_time**
- int **element\_id**
- int **tx\_length**
- int **supp\_enable**
- uint32\_t **safetysupp**
- uint32\_t **header\_ext**
- enum { ... } **P1609\_CHANNEL\_ACCESS**
- enum { ... } **P1609\_WSATYPE**

### 7.21.1 Detailed Description

WME API definitions and functions. This header file contains the API's and data structures for WAVE WSA, WSMP and WME services

Include v2x\_network\_wme.h. Link with -lwme

### 7.21.2 Macro Definition Documentation

#### 7.21.2.1 #define ADVERTISER\_LEN 127

Advertiser id length

#### 7.21.2.2 #define FAIL -1

FAIL Status of the API in this library

#### 7.21.2.3 #define LIBWME\_CHAN\_MODE\_ALT 2

Alternating channel mode

#### 7.21.2.4 #define LIBWME\_CHAN\_MODE\_CONT 1

Continuous channel mode

#### 7.21.2.5 #define LIBWME\_PSC\_STRING\_LENGTH 31

PSC length

#### 7.21.2.6 #define P16093\_RADIO\_INTF1 "ath0"

Radio 0 interface

#### 7.21.2.7 #define P16093\_RADIO\_INTF2 "ath1"

Radio 1 interface

**7.21.2.8 #define P16093\_RADIO\_INTFX "athx"**

Athx interface - use this, when running as an OBU / ASD. More explanation is in the SDK documentation.

**7.21.2.9 #define P16093\_SRV\_IP "::1"**

server ip

**7.21.2.10 #define P1609\_ADVID\_LEN\_MAX 31**

Advertiser ID length

**7.21.2.11 #define PSID\_LEN 4**

PSID length

**7.21.2.12 #define savariwme\_cbs v2x\_wsmp\_callbacks**

For backward compatibility with "struct savariwme\_cbs"

**7.21.2.13 #define savariwme\_reg\_req v2x\_wme\_reg\_req**

For backward compatibility of "struct savariwme\_reg\_req"

**7.21.2.14 #define savariwme\_rx\_indication v2x\_wsmp\_rcv\_indication**

For backward compatibility of "struct savariwme\_rx\_indication"

**7.21.2.15 #define savariwme\_tx\_req v2x\_wsmp\_send\_req**

For backward compatibility of "struct savariwme\_tx\_req"

**7.21.2.16 #define SSI\_LEN 16**

SSI length

**7.21.2.17 #define SUCCESS 0**

SUCCESS status of the API in this library

**7.21.3 Function Documentation****7.21.3.1 void v2x\_wsmp\_close ( v2x\_wsmp\_handler\_t handler )**

close the connection with WME engine

**Description**

This API closes the connection with WNE engine

**Parameters**

<i>in</i>	<i>handler</i>	handler of type <code>v2x_wsmp_handler_t</code>
-----------	----------------	---

**Returns**

None

**7.21.3.2 int wme\_getpsidlen ( uint8\_t \* *psid* )**

gets the psid length

**Parameters**

<i>psid</i>	- an <code>uint8_t</code> pointer.
-------------	------------------------------------

**Returns**

- returns the length of psid on success, and 0 on failure.

**7.21.3.3 int wme\_process\_wme\_cmd ( savari\_wme\_handler\_t *handler*, int *cmd*, int *subcmd* )**

Sends stack specific commands.

**Parameters**

<i>handler</i>	- a handler returned from the <code>wme_init</code>
<i>cmd</i>	- command sent from the application (GET_WME_ERROR/CLEAR_WME_ERROR)

**7.21.3.4 void wme\_provider\_service\_confirm ( savari\_wme\_handler\_t *handler*, int *action*, struct savariwme\_reg\_req \* *wme\_req* )**

confirm provider application to the 1609.3 stack

**Parameters**

<i>handler</i>	- a handler returned from the <code>wme_init</code>
<i>action</i>	- a action to specify whether wanted to do tx/rx or not.
<i>wme_req</i>	- a pointer to the <code>savariwme_reg_req</code> structure, passed from the application at the time of <code>wme_register_user</code>

**Returns**

- returns SUCCESS on success and FAIL on failure.

**7.21.3.5 int wme\_register\_cch\_request ( savari\_wme\_handler\_t *handler*, int *intvl*, int *priority* )**

registers a CCH access request

**Parameters**

<i>handler</i>	- a handler returned from the <code>wme_init</code>
----------------	---

<i>intvl</i>	- CCH interval
<i>priority</i>	- priority

**Returns**

- returns SUCCESS on success and FAIL on failure.

### 7.21.3.6 int wme\_register\_provider ( savari\_wme\_handler\_t *handler*, struct savariwme\_reg\_req \* *wme\_req* )

registers the provider application

**Parameters**

<i>handler</i>	- a handler returned from the wme_init
<i>wme_req</i>	- a pointer to the savariwme_reg_req structure passed from the application

**Returns**

- returns SUCCESS on success and FAIL on failure.

### 7.21.3.7 int wme\_send\_wmectl ( savari\_wme\_handler\_t *handler*, int *cmd*, int *subcmd*, void \* *data*, int *len* )

wme\_deinit - deinitialise and disassociate the application from 1609.3 stack

**Parameters**

<i>handler</i>	- a handler returned from the wme_init
----------------	--

### 7.21.3.8 int wme\_set\_advertiserid ( savari\_wme\_handler\_t *handler*, char \* *adv\_id* )

sets the advertiser id

**Parameters**

<i>handler</i>	- a handler returned from the wme_init
<i>adv_id</i>	- advertiser id from the application to set

**Returns**

- returns SUCCESS on success and FAIL on failure.

### 7.21.3.9 int wme\_set\_gpslocation ( savari\_wme\_handler\_t *handler*, struct libwme\_gpsinfo \* *gpsinfo* )

sets the gps location information to the 1609.3 stack

**Parameters**

<i>handler</i>	- a handler returned from the wme_init
<i>gpsinfo</i>	- The gpsdata passed by the user application. Application may get this by querying the gps stack, refer to libgpsapi on how to query the gps stack , by using those apis.

**Returns**

- returns SUCCESS on success and FAIL on failure.

7.21.3.10 `int wme_set_inactivity_timer ( savari_wme_handler_t handler, int timer_value )`

sets the WRA information to the 1609.3stack



## Parameters

<i>handler</i>	- a handler returned from the wme_init
<i>wra</i>	- the wra information passed by the application

## Returns

- returns SUCCESS on success and FAIL on failure.

7.21.3.11 `int wme_set_wsaintvl ( savari_wme_handler_t handler, int wsa_intvl )`

sets the wsa interval

## Parameters

<i>handler</i>	- a handler returned from the wme_init
<i>wsa_intvl</i>	- WSA interval

## Returns

- returns SUCCESS on success and FAIL on failure.

7.21.3.12 `int wme_set_wsatxpower ( savari_wme_handler_t handler, int txpower )`

sets the txpower of a WSA

## Parameters

<i>handler</i>	- a handler returned from the wme_init
<i>txpower</i>	- transmit power of a WSA

## Returns

- returns SUCCESS on success and FAIL on failure.

7.21.3.13 `int wme_unregister_cch_request ( savari_wme_handler_t handler, int intvl, int priority )`

unregisters CCH access request

## Parameters

<i>handler</i>	- a handler returned from the wme_init
<i>intvl</i>	- CCH interval
<i>priority</i>	- priority

## Returns

- returns SUCCESS on success and FAIL on failure.

7.21.3.14 `void wme_unregister_provider ( savari_wme_handler_t handler, struct savariwme_reg_req * wme_req )`

unregisters the provider

## Parameters

<i>handler</i>	- handler returned from the wme_init
<i>wme_req</i>	- a savariwme_reg_req structure passed from the application at the time of registering using the API wme_register_provider

## 7.21.4 Variable Documentation

### 7.21.4.1 int channel

should be either service channel(SC) or continuous channel(CC)

The WAVE stack support channels ranging from 172 to 184

channel on which the message was received

channel of transmission of WSMs/WSMPs.

### 7.21.4.2 int channel\_access

Provider channel switching mode

One of LIBWME\_CHANNEL\_ACCESS\_CONTINUOUS(non channel switching, stay on channel),

LIBWME\_CHANNEL\_ACCESS\_ALTERNATING(forced/conditional switching between 178 and channel)

### 7.21.4.3 int datarate

datarate

### 7.21.4.4 struct in6\_addr default\_gw

Default gateway is 128 bit IPv6 address of a router that provides internet connectivity to subnet

### 7.21.4.5 int destmacaddr[SAVARI\_MAC\_LEN]

destination macaddress to which WSAs should be sent

### 7.21.4.6 int element\_id

WAVE element id set to WAVE\_ELEMID\_WSMP for WSMP set to WAVE\_ELEMID\_WSMPS for WSMPS

### 7.21.4.7 int expiry\_time

indicates the time at which the message is no longer valid

### 7.21.4.8 int extended\_access

set to 0xffff for continuous access otherwise 0.

### 7.21.4.9 uint8\_t gw\_macaddr[SAVARI\_MAC\_LEN]

Macaddress of the default gateway.

#### 7.21.4.10 uint32\_t header\_ext

WSM header extention

#### 7.21.4.11 int immediate\_access

this indicates the device should immediately switch to SCH, rather than waiting for the next SCH interval (0/1)

#### 7.21.4.12 struct in6\_addr ipv6addr

ipv6addr indicates IPv6 subnet prefix of the link

#### 7.21.4.13 uint16\_t lifetime

router lifetime

#### 7.21.4.14 int local\_service\_index

index to the associated MIB table or internal datastructure. Must be unique for a give PSID and psc combination - not used as of now

#### 7.21.4.15 uint8\_t mac[SAVARI1609\_IEEE80211\_ADDR\_LEN]

destination mac address

#### 7.21.4.16 int num\_rx

number of received bytes

#### 7.21.4.17 int plcp\_length

plcp length

#### 7.21.4.18 uint8\_t prefixlen

prefixlen indicates the IPv6 subnet prefix of the link. (RFC 3513)

#### 7.21.4.19 struct in6\_addr primarydns

Primary DNS is the 128 bit IPv6 address that can provide DNS lookup for the subnet devices.

#### 7.21.4.20 int priority

priority of Provider/User

priority at which the packet received

Qos for Packet

#### 7.21.4.21 char psc[32]

provider service context - ignore for WSMP

#### 7.21.4.22 int psc\_length

provider service context length - ignore for WSMP

#### 7.21.4.23 uint32\_t psid

psid - provider service identifier

PSID is used to differentiate between different safety/roadside application and messages

Provider Service Identifier

Provider Service Identifier

#### 7.21.4.24 int repeatrate

the number of WSAs transmitted for 5sec. Ignore for WSMP traffic.

#### 7.21.4.25 int request\_type

type of user application request.

LIBWME\_USER\_AUTOACCESS\_ONMATCH (Switch Between 178 and SCH after receiving Matching WSA from RSE)

LIBWME\_USER\_AUTOACCESS\_UNCOND (Start Switching between 178 and SCH Without Waiting for a Matching WSA from RSEs)

In case of LIBWME\_USER\_AUTOACCESS\_UNCOND set extended\_access to 0xffff for a prolonged continuous mode of operation

LIBWME\_USER\_AUTOACCESS\_NOSCHACCESS(CCH Only Mode. No Switching) Only applicable if channel\_access is ALTERNATING

#### 7.21.4.26 int rssi

received signalstrength indication

#### 7.21.4.27 uint8\_t rx\_buf[MAX\_DATA]

received buffer

#### 7.21.4.28 uint32\_t rx\_buf\_length

received bufferlen

#### 7.21.4.29 uint8\_t rx\_mac[SAVARI\_MAC\_LEN]

received mac

#### 7.21.4.30 uint32\_t rx\_supp

received WSMP supplement

#### 7.21.4.31 uint32\_t safetysupp

WSM safety supplement

#### 7.21.4.32 struct in6\_addr secondarydns

Secondary DNS is the 128 bit IPv6 address of an alternate device that can provide DNS lookup for the subnet devices.

#### 7.21.4.33 int secondradio

for doing registration of first radio set it to 0 and for the second radio set it to 1.

If second radio is set to 1. The service is going to get the channel access on the radio 1. The radio 0 is a free running radio and runs in 178 continuous hunting for the WSA.

When second radio is set to 1, the application can access the channel using the ONMATCH configuration. If the second radio is set to 0, the application will wait for the WSAs and looks for the services of interest.

#### 7.21.4.34 struct in6\_addr service\_ipv6addr

service IPv6 address; memset to 0 if not used - ignore for WSMP

#### 7.21.4.35 int service\_port

port on which service is provided; memset to 0 if not used - ignore for WSMP

#### 7.21.4.36 uint8\_t srcmac[SAVARI1609\_IEEE80211\_ADDR\_LEN]

source mac address .. place holder and does not perform any task

#### 7.21.4.37 uint8\_t srcmacaddr[SAVARI\_MAC\_LEN]

source mac address

#### 7.21.4.38 int supp\_enable

sup\_enable when 1 reads and transmits safetysupp when 0 it doesn't

#### 7.21.4.39 uint64\_t tstamp

timestamp at which packet got received

#### 7.21.4.40 int tx\_length

tx buffer length

#### 7.21.4.41 int txpower

transmitted power

transmit power

#### 7.21.4.42 int version

WAVE version number

#### 7.21.4.43 int wsatype

secured (SAVARI1609\_WSA\_SECURED) or unsecured (SAVARI1609\_WSA\_UNSECURED) WSA - ignore for W-SMP

## 7.22 v2x\_security.h File Reference

1609.2 API definitions and functions

```
#include "v2x_error.h"
#include "v2x_security_base.h"
```

### Functions

- [v2x\\_status\\_t v2x\\_sec\\_init](#) ([v2x\\_security\\_profile\\_t](#) \*profile, int \*sock)
- void [v2x\\_sec\\_deinit](#) (int sock)
- [v2x\\_status\\_t v2x\\_sec\\_sign\\_request](#) (int sock, uint32\_t [psid](#), uint8\_t \*ssp, uint32\_t sspLength, int signer\_id, uint8\_t \*payload, int payload\_len, void \*app\_ref)
- [v2x\\_status\\_t v2x\\_sec\\_verify\\_request](#) (int sock, uint32\_t [psid](#), uint8\_t \*[mac](#), int verify\_disable, uint8\_t \*payload, int payload\_len, void \*app\_ref)
- [v2x\\_status\\_t v2x\\_sec\\_rx](#) (int sock, [v2x\\_sec\\_rx\\_cblist\\_t](#) \*cblist, void \*ctx)
- [v2x\\_status\\_t v2x\\_sec\\_cert\\_change\\_register](#) (int sock, uint32\_t [psid](#), void \*app\_ref)
- [v2x\\_status\\_t v2x\\_sec\\_cert\\_change\\_unregister](#) (int sock)
- [v2x\\_status\\_t v2x\\_sec\\_idchange\\_lock](#) (int sock)
- [v2x\\_status\\_t v2x\\_sec\\_idchange\\_unlock](#) (int sock)

### 7.22.1 Detailed Description

1609.2 API definitions and functions This header contains the structures and APIs that are needed to sign/verify and encrypt/decrypt the messages. The messages could be encoded saej2735 data from the v2x\_message libraries or WSAs.

Include [v2x\\_security.h](#) and Link with -laero16092.

## 7.23 v2x\_security\_base.h File Reference

1609.2 API definitions and functions

```
#include <stdio.h>
#include <errno.h>
#include <viicsec.h>
```

## Data Structures

- struct [aerolink\\_sign\\_secprofile](#)
- struct [aerolink\\_verify\\_secprofile](#)
- struct [security\\_cmd\\_profile\\_msg](#)
- struct [ae\\_ui\\_rx\\_callbacks](#)

## Macros

- #define **AE\_MAX\_RCVBUF\_LEN** 4096
- #define **V2X\_SEC\_SERVER\_UNIX\_ADDR** "/tmp/aerolinkV2Xd.sock"
- #define **AE\_SET\_PRIV**(\_\_cli\_data, \_\_priv) \_\_cli\_data->priv = \_\_priv
- #define **ae\_set\_priv** AE\_SET\_PRIV

## Typedefs

- typedef [v2x\\_security\\_profile\\_type\\_t](#) **securityProfileType\_t**
- typedef struct [aerolink\\_sign\\_secprofile](#) [v2x\\_sign\\_sec\\_profile\\_t](#)
- typedef struct [aerolink\\_verify\\_secprofile](#) [v2x\\_verify\\_sec\\_profile\\_t](#)
- typedef struct [security\\_cmd\\_profile\\_msg](#) [v2x\\_security\\_profile\\_t](#)
- typedef struct [ae\\_ui\\_rx\\_callbacks](#) [v2x\\_sec\\_rx\\_cblist\\_t](#)
- typedef [v2x\\_sec\\_rx\\_cblist\\_t](#) **ae\_ui\_rx\_cblist**

## Enumerations

- enum [v2x\\_sec\\_signer\\_id\\_t](#) {  
[AE\\_SIGNER\\_ID\\_TYPE\\_CERT](#), [AE\\_SIGNER\\_ID\\_TYPE\\_DIGEST](#),  
[AE\\_SIGNER\\_ID\\_TYPE\\_AUTO](#) }
- enum [v2x\\_security\\_profile\\_type\\_t](#) { [SECURITY\\_SIGNATURE\\_PROFILE](#) = 0x01, [SECURITY\\_VERIFICATION\\_PROFILE](#) = 0x02 }

## Functions

- void **ae\_debug** (void \*, int, char \*,...)
- struct [aerolink\\_sign\\_secprofile](#) **\_\_attribute\_\_((\_\_packed\_\_))**

## Variables

- int [cert\\_attach\\_rate](#)
- int [use\\_generation\\_time](#)
- int [use\\_expiry\\_time](#)
- int [use\\_generation\\_loc](#)
- int [verification\\_interval](#)
- int [check\\_replay](#)
- int [check\\_generation\\_time](#)
- int [check\\_expiry\\_time](#)
- int [check\\_generation\\_location](#)
- [securityProfileType\\_t](#) [prof\\_type](#)
- struct [aerolink\\_sign\\_secprofile](#) [sign\\_profile](#)
- struct [aerolink\\_verify\\_secprofile](#) [verify\\_profile](#)

### 7.23.1 Detailed Description

1609.2 API definitions and functions This header contains the structures and APIs related to security service.

### 7.23.2 Enumeration Type Documentation

#### 7.23.2.1 enum v2x\_sec\_signer\_id\_t

v2x\_sec\_signer\_id\_t - signer ID of the message

Enumerator

***AE\_SIGNER\_ID\_TYPE\_CERT*** AE\_SIGNER\_ID\_TYPE\_CERT - certificate

***AE\_SIGNER\_ID\_TYPE\_DIGEST*** AE\_SIGNER\_ID\_TYPE\_DIGEST - digest

***AE\_SIGNER\_ID\_TYPE\_AUTO*** AE\_SIGNER\_ID\_TYPE\_AUTO - auto

### 7.23.3 Variable Documentation

#### 7.23.3.1 int cert\_attach\_rate

cert\_attach\_rate - certificate attach rate for the message

1000 msec for BSM for example

#### 7.23.3.2 int check\_expiry\_time

check\_expiry\_time - check for the expiry time in security headers 1 to enable, 0 to disable

#### 7.23.3.3 int check\_generation\_location

check\_generation\_location - check for the generation location in security headers 1 to enable, 0 to disable

#### 7.23.3.4 int check\_generation\_time

check\_generation\_time - check for the generation time in security headers 1 to enable, 0 to disable

#### 7.23.3.5 int check\_replay

check\_replay - check for the replay 1 to enable, 0 to disable

#### 7.23.3.6 securityProfileType\_t prof\_type

prof\_type - prof\_type is one of the OR combinations of SECURITY\_SIGNATURE\_PROFILE and / or SECURITY\_VERIFICATION\_PROFILE

#### 7.23.3.7 struct aerolink\_sign\_secprofile sign\_profile

sign\_profile - signature profile

#### 7.23.3.8 int use\_expiry\_time

use\_expiry\_time - set expiry time in security headers



#### 7.23.3.9 int use\_generation\_loc

use\_generation\_loc - set generation location in security headers

#### 7.23.3.10 int use\_generation\_time

use\_generation\_time - set generation time in security headers

#### 7.23.3.11 int verification\_interval

verification\_interval - verification interval

for ex: 1000 msec for BSM

#### 7.23.3.12 struct aerolink\_verify\_secprofile verify\_profile

verify\_profile - verification profile

## 7.24 v2x\_sensor\_can.h File Reference

CAN sensor API definitions and function.

```
#include "libpcan.h"
#include "canheader.h"
#include "canreader.h"
#include "v2x_error.h"
```

### Functions

- [v2x\\_status\\_t v2x\\_can\\_open](#) (int \*can\_hdl, int flag)
- [v2x\\_status\\_t v2x\\_can\\_rcv](#) (int can\_hdl, struct cand\_rcv\_buf \*buf)
- [v2x\\_status\\_t v2x\\_can\\_close](#) (int can\_hdl)

#### 7.24.1 Detailed Description

CAN sensor API definitions and function. This header file contains the APIs and data structures used to access CAN sensor.

Include [v2x\\_sensor\\_can.h](#). Link with -lcan.

## 7.25 v2x\_sensor\_gnss.h File Reference

Navigation sensor API definitions and functions.

```
#include "pf_common.h"
#include "v2x_error.h"
```

### Data Structures

- struct [pf\\_gnss\\_data](#)

## Macros

- `#define PF_GNSS_NAV 0x7eadbeef`
- `#define PF_GNSS_YAWRATE_NAV PF_GNSS_NAV`
- `#define PF_GNSS_LATACCEL_NAV PF_GNSS_NAV`
- `#define PF_GNSS_LONGACCEL_NAV PF_GNSS_NAV`
- `#define PF_GNSS_VERTACCEL_NAV PF_GNSS_NAV`
- `#define PF_GNSS_MAX_TIME_STR 64`
- `#define SUCCESS 0`
- `#define FAIL -1`

## Typedefs

- `typedef struct pf_gnss_data pf_gnss_data_t`
- `typedef struct gps_data_t v2x_gnss_handle_t`
- `typedef pf_gnss_data_t v2x_gnss_data_t`
- `typedef void * v2x_pf_handle`

## Functions

- `pf_status_t pf_init (void **pf_handle)`
- `pf_status_t pf_deinit (void *pf_handle)`
- `v2x_status_t v2x_gnss_open (int *gps_fd, void **handle)`
- `pf_status_t pf_gnss_open (void **handle)`
- `pf_status_t pf_gnss_get_gpsfd (INT32_t *fd, void *handle)`
- `v2x_status_t v2x_gnss_read (pf_gnss_data_t *gnss_data, void *handle)`
- `pf_status_t pf_gnss_rcv_async (pf_gnss_data_t *gnss_data, void *gnss_handle)`
- `pf_status_t pf_gnss_read (pf_gnss_data_t *gnss_data, void *pf_gnss_handle)`
- `v2x_status_t v2x_gnss_close (void *handle)`
- `pf_status_t pf_gnss_close (void *gnss_handle)`
- `v2x_status_t v2x_gnss_filters_init (pf_gnss_data_t *gnss_data)`
- `pf_status_t pf_gnss_filters_init (void *handle, double lcf, double ldf, double acf, double adf, double update_hz)`
- `v2x_status_t v2x_gnss_filters_deinit (pf_gnss_data_t *gnss_data)`
- `pf_status_t pf_gnss_filters_deinit (void *handle)`
- `pf_status_t pf_gnss_get_pfname (CHAR_t *name, void *gnss_handle)`
- `pf_status_t pf_gnss_get_nmea (void **handle, char **data)`
- `void * pf_gnss_open_nw (int *fd, int is_async, void **pf_handle, char *device_ip, int device_port, int, char *)`
- `int pf_gnss_read_nw (void *ctx, pf_gnss_data_t *gps_data, void *pf_handle, double speed)`
- `int pf_gnss_close_nw (void *ctx, struct gps_data_t *gpsd_handle)`

### 7.25.1 Detailed Description

Navigation sensor API definitions and functions. This header file contains the APIs and data structures used to access navigation sensor data.

Include [v2x\\_sensor\\_gnss.h](#). Link with `-lpf`, `-lgps`.

### 7.25.2 Macro Definition Documentation

#### 7.25.2.1 `#define FAIL -1`

status returned if an API in this library fails

### 7.25.2.2 #define SUCCESS 0

#### PF GNSS NW APIs

Navigation API definitions and functions to receive GNSS data over network. status returned if an API in this library succeeds

### 7.25.3 Function Documentation

#### 7.25.3.1 pf\_status\_t pf\_gnss\_get\_nmea ( void \*\* *handle*, char \*\* *data* )

function: pf\_gnss\_filters\_deinit

Get the gps NMEA strings

parameter [in] handle The pf handle parameter [out] data The NMEA string

return: SUCCESS on success or error number on failure

#### 7.25.3.2 pf\_status\_t pf\_gnss\_get\_pfname ( CHAR\_t \* *name*, void \* *gnss\_handle* )

function: pf\_gnss\_get\_pfname

Get the platform name

parameter [in] gnss\_handle The GNSS handler

returns: SUCCESS on success or error number on failure.

# Index

- [\\_J2735\\_2016\\_EnableLaneList\\_t, 71](#)
  - [lane\\_id, 71](#)
- [\\_J2735\\_2016\\_ManeuverAssistList\\_t, 71](#)
  - [connection\\_id, 72](#)
- [\\_J2735\\_2016\\_MovementEventList\\_t, 72](#)
  - [phase\\_state, 72](#)
  - [time\\_change\\_info, 72](#)
- [\\_J2735\\_2016\\_MovementPhaseState\\_t](#)
  - [v2x\\_msg\\_spat.h, 217](#)
- [\\_J2735\\_2016\\_MovementState\\_t, 72](#)
  - [movement\\_event, 73](#)
  - [num\\_movement\\_event, 73](#)
  - [signal\\_group\\_id, 73](#)
- [\\_J2735\\_2016\\_PHObject, 74](#)
  - [fps, 74](#)
  - [gps\\_status, 74](#)
  - [ph\\_points, 74](#)
  - [ph\\_points\\_len, 74](#)
  - [valid\\_fps, 74](#)
- [\\_J2735\\_2016\\_PVD, 75](#)
  - [full\\_posvec, 76](#)
  - [min\\_of\\_year, 76](#)
  - [probe\\_seg\\_num, 76](#)
  - [snapshot, 76](#)
  - [snapshot\\_len, 76](#)
  - [valid\\_fullpos\\_vec, 76](#)
  - [veh\\_class, 76](#)
  - [vid\\_info, 76](#)
  - [vid\\_info\\_present, 76](#)
- [\\_J2735\\_2016\\_PathPrediction, 73](#)
  - [confidence, 73](#)
  - [radius\\_of\\_curvature, 73](#)
- [\\_J2735\\_2016\\_ProbeSnapshot, 74](#)
  - [fps, 75](#)
  - [valid\\_veh\\_safety\\_ext, 75](#)
  - [valid\\_veh\\_status, 75](#)
  - [veh\\_safety\\_ext, 75](#)
  - [vehicle\\_status, 75](#)
- [\\_J2735\\_2016\\_TimeChange\\_t, 77](#)
  - [confidence, 77](#)
  - [likely\\_time, 77](#)
  - [max\\_end\\_time, 77](#)
  - [min\\_end\\_time, 77](#)
  - [next\\_time, 77](#)
  - [start\\_time, 77](#)
- [\\_J2735\\_2016\\_VehSafetyExtensions, 79](#)
  - [event\\_flags, 79](#)
  - [exterior\\_lights, 79](#)
  - [ph, 79](#)
  - [pp, 79](#)
  - [valid\\_ph, 79](#)
  - [valid\\_pp, 79](#)
- [\\_J2735\\_2016\\_VehicleMeasurements, 78](#)
- [\\_J2735\\_2016\\_VehicleStatus, 78](#)
  - [exterior\\_lights, 78](#)
  - [lightbar\\_in\\_use, 78](#)
  - [valid\\_vehicle\\_data, 78](#)
  - [valid\\_wiper\\_set, 78](#)
  - [vehicle\\_data, 78](#)
  - [wiper\\_set, 79](#)
- [\\_J2735\\_2016\\_Wiperset, 80](#)
  - [wiper\\_front, 80](#)
  - [wiper\\_front\\_rate, 80](#)
  - [wiper\\_rear, 80](#)
  - [wiper\\_rear\\_rate, 80](#)
- [\\_\\_attribute\\_\\_](#)
  - [v2x\\_msg\\_spat.h, 218](#)
  - [v2x\\_msg\\_tim.h, 233](#)
- [AE\\_SIGNER\\_ID\\_TYPE\\_AUTO](#)
  - [v2x\\_security\\_base.h, 251](#)
- [AE\\_SIGNER\\_ID\\_TYPE\\_CERT](#)
  - [v2x\\_security\\_base.h, 251](#)
- [AE\\_SIGNER\\_ID\\_TYPE\\_DIGEST](#)
  - [v2x\\_security\\_base.h, 251](#)
- [ADVERTISER\\_LEN](#)
  - [v2x\\_net\\_wme.h, 240](#)
- [ASN1\\_BSM\\_MSG\\_ID](#)
  - [v2x\\_msg\\_bsm.h, 179](#)
- [abs](#)
  - [v2x\\_msg\\_bsm, 152](#)
- [ac](#)
  - [v2x\\_etsi\\_denm, 137](#)
- [accel\\_ctrl](#)
  - [v2x\\_etsi\\_cam\\_hf\\_vehicle, 134](#)
- [action\\_id](#)
  - [v2x\\_etsi\\_denm\\_mc, 140](#)
- [advisory\\_codes](#)
  - [J2735TIM, 107](#)
- [ae\\_sign\\_callback](#)
  - [ae\\_ui\\_rx\\_callbacks, 81](#)
- [ae\\_ui\\_rx\\_callbacks, 80](#)
  - [ae\\_sign\\_callback, 81](#)
  - [ae\\_verify\\_callback, 81](#)
- [ae\\_verify\\_callback](#)
  - [ae\\_ui\\_rx\\_callbacks, 81](#)
- [aerolink\\_sign\\_secprofile, 82](#)
  - [cert\\_attach\\_rate, 82](#)
  - [use\\_expiry\\_time, 82](#)

- use\_generation\_loc, 82
  - use\_generation\_time, 82
- aerolink\_verify\_secprofile, 82
  - check\_expiry\_time, 82
  - check\_generation\_location, 82
  - check\_generation\_time, 82
  - check\_replay, 83
  - verification\_interval, 83
- altitude
  - pf\_gnss\_data, 120
- angle
  - v2x\_msg\_bsm, 152
- approach\_type
  - v2x\_etsi\_lane, 143
- autolightcontrol
  - v2x\_msg\_bsm, 152
- auxbrakes
  - v2x\_msg\_bsm, 153
- BSM\_ABS\_ACTIVE\_ENGAGED
  - v2x\_msg\_bsm.h, 180
- BSM\_ABS\_ACTIVE\_OFF
  - v2x\_msg\_bsm.h, 179
- BSM\_ABS\_ACTIVE\_ON
  - v2x\_msg\_bsm.h, 180
- BSM\_ABS\_ACTIVE\_UNAVAIL
  - v2x\_msg\_bsm.h, 179
- BSM\_PART1
  - v2x\_msg\_bsm.h, 179
- BSM\_PART\_II\_ID\_TYPE\_SPECIAL\_VEH\_EXT
  - v2x\_msg\_common.h, 197
- BSM\_PART\_II\_ID\_TYPE\_SUPPLEMENTARY\_VEH\_EXT
  - v2x\_msg\_common.h, 197
- BSM\_PART\_II\_ID\_TYPE\_VEH\_SAFETY\_EXT
  - v2x\_msg\_common.h, 197
- BSM\_PATHHISTORY
  - v2x\_msg\_bsm.h, 179
- BSM\_PATHPREDICTION
  - v2x\_msg\_bsm.h, 179
- BSM\_SCS\_ENGAGED
  - v2x\_msg\_bsm.h, 180
- BSM\_SCS\_OFF
  - v2x\_msg\_bsm.h, 180
- BSM\_SCS\_ON
  - v2x\_msg\_bsm.h, 180
- BSM\_SCS\_UNAVAIL
  - v2x\_msg\_bsm.h, 180
- BSM\_TDF\_UNENCODED
  - v2x\_msg\_bsm.h, 179
- BSM\_TESTDATAFRAME
  - v2x\_msg\_bsm.h, 179
- BSM\_TRC\_ENGAGED
  - v2x\_msg\_bsm.h, 180
- BSM\_TRC\_OFF
  - v2x\_msg\_bsm.h, 180
- BSM\_TRC\_ON
  - v2x\_msg\_bsm.h, 180
- BSM\_TRC\_UNAVAIL
  - v2x\_msg\_bsm.h, 180
- BSM\_VEICLESTATUS
  - v2x\_msg\_bsm.h, 179
- BSM\_WIPERS\_AUTOMATIC\_PRESENT
  - v2x\_msg\_bsm.h, 180
- BSM\_WIPERS\_HIGH
  - v2x\_msg\_bsm.h, 180
- BSM\_WIPERS\_INTERMITTENT
  - v2x\_msg\_bsm.h, 180
- BSM\_WIPERS\_LOW
  - v2x\_msg\_bsm.h, 180
- BSM\_WIPERS\_OFF
  - v2x\_msg\_bsm.h, 180
- BSM\_WIPERS\_UNAVAIL
  - v2x\_msg\_bsm.h, 180
- BSM\_WIPERS\_WASHER\_IN\_USE
  - v2x\_msg\_bsm.h, 180
- BSM\_TYPE
  - v2x\_msg\_bsm.h, 179
- BSMAbsActive\_t
  - v2x\_msg\_bsm.h, 179
- BSMBrakeAppliedStatus\_t
  - v2x\_msg\_bsm.h, 180
- BSMPartII\_ID\_Types\_t
  - v2x\_msg\_common.h, 197
- BSMStabilityControlStatus\_t
  - v2x\_msg\_bsm.h, 180
- BSMTractionControl\_t
  - v2x\_msg\_bsm.h, 180
- BSMWiperStatus\_t
  - v2x\_msg\_bsm.h, 180
- basic\_vehicle\_role
  - J2735\_2016\_SignalRequest, 97
- bc
  - v2x\_etsi\_cam, 131
- bit\_mask
  - v2x\_etsi\_denm\_ac, 138
  - v2x\_etsi\_denm\_rwc, 141
- brakeboost
  - v2x\_msg\_bsm, 153
- bumperheight\_front
  - v2x\_msg\_bsm, 153
- bumperheight\_rear
  - v2x\_msg\_bsm, 153
- can\_fd
  - v2x\_canctl, 130
- cert\_attach\_rate
  - aerolink\_sign\_secprofile, 82
  - v2x\_security\_base.h, 251
- channel
  - v2x\_net\_wme.h, 245
  - v2x\_wme\_reg\_req, 165
  - v2x\_wsmp\_rcv\_indication, 170
  - v2x\_wsmp\_send\_req, 171
- channel\_access
  - v2x\_net\_wme.h, 245
  - v2x\_wme\_reg\_req, 165
- channelmode

- testdataframe, 128
- check\_expiry\_time
  - aerolink\_verify\_secprofile, 82
  - v2x\_security\_base.h, 251
- check\_generation\_location
  - aerolink\_verify\_secprofile, 82
  - v2x\_security\_base.h, 251
- check\_generation\_time
  - aerolink\_verify\_secprofile, 82
  - v2x\_security\_base.h, 251
- check\_replay
  - aerolink\_verify\_secprofile, 83
  - v2x\_security\_base.h, 251
- circle
  - J2735\_2016\_GeometricProjection, 93
  - J2735ValidRegion, 111
- closed\_lanes
  - v2x\_etsi\_denm\_rwc, 141
- closedpath
  - J2735TIMGeoPath, 109
- computed\_lanes
  - J2735\_2016\_NodeXY, 95
- confidence
  - \_J2735\_2016\_PathPrediction, 73
  - \_J2735\_2016\_TimeChange\_t, 77
  - v2x\_etsi\_spat\_mm\_evt, 149
  - v2x\_msg\_bsm, 153
  - v2x\_msg\_psm, 160
- connection\_id
  - \_J2735\_2016\_ManeuverAssistList\_t, 72
- connects
  - v2x\_etsi\_lane, 143
- connects\_len
  - v2x\_etsi\_lane, 143
- connects\_to, 83
  - id, 83
  - laneset, 114
  - signal\_group\_id, 83
- content\_type
  - J2735TIM, 107
- count
  - v2x\_msg\_bsm, 153
- cross\_request
  - v2x\_msg\_psm, 160
- crwattr
  - laneattr, 112
- curvature
  - v2x\_etsi\_cam\_hf\_vehicle, 134
- curvature\_calc
  - v2x\_etsi\_cam\_hf\_vehicle, 134
- curvature\_conf
  - v2x\_etsi\_cam\_hf\_vehicle, 134
- dElev
  - nodeattrs, 116
- dWidth
  - nodeattrs, 116
- dataframes
  - v2x\_msg\_tim, 164
- datarate
  - testdataframe, 128
  - v2x\_net\_wme.h, 245
  - v2x\_wsmp\_rcv\_indication, 170
  - v2x\_wsmp\_send\_req, 171
- datetime, 83
- day
  - utcTimeFrame, 129
- default\_gw
  - savari1609Wra, 123
  - v2x\_net\_wme.h, 245
- destmacaddr
  - v2x\_net\_wme.h, 245
  - v2x\_wme\_reg\_req, 165
- detection\_time
  - v2x\_etsi\_denm\_mc, 140
- device\_usage\_state
  - v2x\_msg\_psm, 160
- direction
  - laneattr, 112
- directionality
  - J2735TIMGeoPath, 109
- drive\_direction
  - v2x\_etsi\_cam\_hf\_vehicle, 134
- dsecond
  - j2735\_2016\_srm, 101
  - j2735\_2016\_ssm, 101
  - pf\_gnss\_data, 120
  - v2x\_etsi\_spat\_isec, 148
- dtimerunlights
  - v2x\_msg\_bsm, 153
- duration
  - J2735\_2016\_SignalStatusPackageList, 99
  - J2735TIM, 107
  - srm\_list, 126
- ETSIItineraryPath\_t, 89
- ETSIRefDENMs\_t, 89
- ETSIRestrictedTypes\_t, 89
- egress\_approach
  - v2x\_etsi\_lane, 143
- element\_id
  - v2x\_net\_wme.h, 245
  - v2x\_wsmp\_send\_req, 171
- elev\_confidence
  - v2x\_msg\_bsm, 153
- elev\_offset
  - path\_history, 118
  - pathhistoryframe, 119
- elevation
  - J2735Position3D, 103
  - v2x\_msg\_bsm, 153
  - v2x\_msg\_psm, 160
- emergency\_details
  - v2x\_msg\_eva, 158
- emergency\_details\_avail
  - v2x\_msg\_eva, 158
- emergency\_type\_event
  - v2x\_msg\_bsm, 153

- enabled\_lanes
  - intersection\_data, [90](#)
  - v2x\_etsi\_spat\_isec, [148](#)
- enabled\_lanes\_len
  - v2x\_etsi\_spat\_isec, [148](#)
- encbrakestatus, [84](#)
- encpadbytes, [84](#)
- encpointsetstype04, [84](#)
- encprndlspeed, [85](#)
- enctimestamp, [85](#)
- encvehiclesize, [85](#)
- etsi\_cam\_dgc, [86](#)
- etsi\_cam\_ec, [86](#)
- etsi\_cam\_ptc, [86](#)
- etsi\_cam\_rc, [86](#)
- etsi\_cam\_rwc\_basic, [87](#)
- etsi\_cam\_scc, [87](#)
- etsi\_cam\_stc, [87](#)
- etsi\_connects, [87](#)
  - lane\_id, [87](#)
  - maneuvers, [87](#)
  - signal\_group\_id, [88](#)
- etsi\_node, [88](#)
  - node\_attr, [88](#)
  - node\_point, [88](#)
- etsi\_node\_attr, [88](#)
- etsi\_pos3d, [88](#)
- event\_absactivate
  - v2x\_msg\_bsm, [153](#)
- event\_airbag
  - v2x\_msg\_bsm, [153](#)
- event\_descr
  - v2x\_msg\_bsm, [154](#)
- event\_descr\_len
  - v2x\_msg\_bsm, [154](#)
- event\_flags
  - \_J2735\_2016\_VehSafetyExtensions, [79](#)
- event\_hazardbraking
  - v2x\_msg\_bsm, [154](#)
- event\_hazardlights
  - v2x\_msg\_bsm, [154](#)
- event\_history\_list
  - v2x\_etsi\_denm\_sc, [142](#)
- event\_position
  - v2x\_etsi\_denm\_mc, [140](#)
- event\_position\_heading
  - v2x\_etsi\_denm\_lc, [139](#)
- event\_position\_heading\_conf
  - v2x\_etsi\_denm\_lc, [139](#)
- event\_speed
  - v2x\_etsi\_denm\_lc, [139](#)
- event\_speed\_conf
  - v2x\_etsi\_denm\_lc, [139](#)
- event\_stabilityctrlactivated
  - v2x\_msg\_bsm, [154](#)
- event\_traces
  - v2x\_etsi\_denm\_lc, [139](#)
- event\_tractionctrlloss
  - v2x\_msg\_bsm, [154](#)
- event\_type
  - v2x\_etsi\_denm\_sc, [142](#)
- exitservice\_codes
  - J2735TIM, [107](#)
- expiry\_time
  - v2x\_net\_wme.h, [245](#)
  - v2x\_wsmp\_send\_req, [172](#)
- ext\_lights
  - v2x\_etsi\_cam\_lf, [137](#)
- extended\_access
  - v2x\_net\_wme.h, [245](#)
  - v2x\_wme\_reg\_req, [165](#)
- extent
  - J2735\_2016\_GeometricProjection, [93](#)
  - J2735ValidRegion, [111](#)
- exterior\_lights
  - \_J2735\_2016\_VehSafetyExtensions, [79](#)
  - \_J2735\_2016\_VehicleStatus, [78](#)
- FAIL
  - v2x\_net\_wme.h, [240](#)
  - v2x\_sensor\_gnss.h, [253](#)
- foglights
  - v2x\_msg\_bsm, [154](#)
- formatversion
  - testdataframe, [128](#)
- fps
  - \_J2735\_2016\_PHObject, [74](#)
  - \_J2735\_2016\_ProbeSnapshot, [75](#)
- free\_tim
  - v2x\_msg\_tim.h, [233](#)
- full\_pos\_valid
  - srm\_requestor, [127](#)
- full\_posvec
  - \_J2735\_2016\_PVD, [76](#)
- further\_info\_id
  - J2735TIM, [107](#)
- generation\_delta\_time
  - v2x\_etsi\_cam, [131](#)
- genericsign\_codes
  - J2735TIM, [107](#)
- geographical\_path
  - J2735TIM, [107](#)
- geometric\_projection
  - J2735TIMGeoPath, [109](#)
- gps\_status
  - \_J2735\_2016\_PHObject, [74](#)
- gpsstatus
  - v2x\_msg\_bsm, [154](#)
- gw\_macaddr
  - savari1609Wra, [123](#)
  - v2x\_net\_wme.h, [245](#)
- hazardlights
  - v2x\_msg\_bsm, [154](#)
- head\_confidence
  - v2x\_msg\_bsm, [154](#)

- header\_ext
  - v2x\_net\_wme.h, 245
  - v2x\_wsmp\_send\_req, 172
- heading
  - path\_history, 118
  - pathhistoryframe, 119
  - pf\_gnss\_data, 120
  - v2x\_etsi\_cam\_hf\_vehicle, 134
  - v2x\_msg\_bsm, 154
  - v2x\_msg\_psm, 160
- heading\_acc
  - v2x\_etsi\_cam\_hf\_vehicle, 134
- heading\_slice
  - J2735\_2016\_GeometricProjection, 93
  - J2735RoadSignID, 105
  - J2735TIMGeoPath, 109
  - J2735ValidRegion, 111
- hf
  - v2x\_etsi\_cam, 132
- highbeam
  - v2x\_msg\_bsm, 154
- hopinfo
  - v2x\_msg\_tim, 164
- hour
  - utcTimeFrame, 129
- id
  - connects\_to, 83
  - J2735TIMGeoPath, 110
  - srm\_requestor, 127
- iid
  - J2735\_2016\_SignalRequest\_t, 98
  - J2735\_2016\_SignalStatusList, 99
  - v2x\_etsi\_map\_isec, 145
- ilist
  - v2x\_etsi\_map\_msg, 146
- ilist\_len
  - v2x\_etsi\_map\_msg, 146
- immediate\_access
  - v2x\_net\_wme.h, 246
  - v2x\_wme\_reg\_req, 165
- in\_bound\_access\_point
  - J2735\_2016\_SignalRequest\_t, 98
  - J2735\_2016\_SignalStatusPackageList, 99
- incident\_indication
  - v2x\_etsi\_denm\_rwc, 141
- information\_quality
  - v2x\_etsi\_denm\_sc, 143
- ingress\_approach
  - v2x\_etsi\_lane, 143
- intersec\_access\_pt, 89
- intersection\_data, 90
  - enabled\_lanes, 90
  - intersection\_id, 90
  - intersection\_status, 90
  - maneuver\_assist, 90
  - min\_of\_the\_year, 90
  - movement\_states, 90
  - msgCount, 90
  - n\_lane\_list, 91
  - num\_maneuver\_assist, 91
  - num\_movement\_states, 91
  - timestamp\_ms, 91
- intersection\_geodescr, 91
  - intersection\_id, 91
  - lanelist, 91
  - lanewidth, 92
  - msg\_revision, 92
  - n\_lanelist, 92
  - ref\_pt, 92
- intersection\_id
  - intersection\_data, 90
  - intersection\_geodescr, 91
- Intersection\_list
  - v2x\_msg\_spat, 162
- intersection\_status
  - intersection\_data, 90
  - v2x\_msg\_spat.h, 217
- ipv6addr
  - savari1609Wra, 123
  - v2x\_net\_wme.h, 246
- isec\_geo\_descr
  - v2x\_msg\_map, 159
- isec\_id
  - v2x\_etsi\_spat\_isec, 148
- isec\_list
  - v2x\_etsi\_spat\_msg, 150
- isec\_list\_len
  - v2x\_etsi\_spat\_msg, 150
- isec\_status
  - v2x\_etsi\_spat\_isec, 148
- itinerary\_path
  - v2x\_etsi\_denm\_rwc, 141
- J2735\_2016\_BSM\_MSG\_D
  - The V2X Message API (US), 26
- J2735\_2016\_BSM\_MSG\_U
  - The V2X Message API (US), 26
- J2735\_2016\_BUS\_VEHICLE\_TRAFFIC
  - v2x\_msg\_common.h, 198
- J2735\_2016\_COMPUTED\_LANES
  - v2x\_msg\_tim.h, 231
- J2735\_2016\_CRW\_AUDIO\_SUPPORT
  - The V2X Message API (US), 27
- J2735\_2016\_CRW\_BICYCLE\_USE\_ONLY
  - The V2X Message API (US), 26
- J2735\_2016\_CRW\_BIDIRECTIONAL\_CYCLE\_TIME
  - The V2X Message API (US), 26
- J2735\_2016\_CRW\_CRW\_REVOCABLE\_LANE
  - The V2X Message API (US), 26
- J2735\_2016\_CRW\_FIXED\_CYCLE\_TIME
  - The V2X Message API (US), 26
- J2735\_2016\_CRW\_HAS\_PUSH\_TO\_WALK\_BUTTON
  - The V2X Message API (US), 26
- J2735\_2016\_CRW\_IS\_FLYOVER\_CRW\_LANE
  - The V2X Message API (US), 26
- J2735\_2016\_CRW\_RFSIGNAL\_REQ\_PRESENT
  - The V2X Message API (US), 27



- J2735\_2016\_CRW\_UNSIGNALIZED\_SEGMENTS\_P-RESENT  
The V2X Message API (US), [27](#)
- J2735\_2016\_CYCLIST\_TRAFFIC  
v2x\_msg\_common.h, [198](#)
- J2735\_2016\_DIROFUSE\_BOTH  
The V2X Message API (US), [23](#)
- J2735\_2016\_DIROFUSE\_FORWARD  
The V2X Message API (US), [23](#)
- J2735\_2016\_DIROFUSE\_REVERSE  
The V2X Message API (US), [23](#)
- J2735\_2016\_EVT\_ABS\_ACTIVATED  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_AIRBAG\_DEPLOYMENT  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_DISABLED\_VEHICLE  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_FLATTIRE  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_HARDBRAKING  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_HAZARD\_LIGHTS  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_HAZARDOUS\_MATERIAL  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_LIGHTS\_CHANGED  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_RESERVED1  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_STABILITY\_CONTROL  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_STOPLINE\_VIOLATION  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_TRACTION\_CONTROL  
The V2X Message API (US), [25](#)
- J2735\_2016\_EVT\_WIPERS\_CHANGED  
The V2X Message API (US), [25](#)
- J2735\_2016\_EXT\_AUTOMATIC\_LIGHT\_CONTROL\_ON  
The V2X Message API (US), [23](#)
- J2735\_2016\_EXT\_DAYTIMERUNLIGHTS\_ON  
The V2X Message API (US), [23](#)
- J2735\_2016\_EXT\_FOGLIGHTS\_ON  
The V2X Message API (US), [23](#)
- J2735\_2016\_EXT\_HAZARDSIGNAL\_ON  
The V2X Message API (US), [23](#)
- J2735\_2016\_EXT\_HIGHBEAM\_ON  
The V2X Message API (US), [23](#)
- J2735\_2016\_EXT\_LEFTTURN\_ON  
The V2X Message API (US), [23](#)
- J2735\_2016\_EXT\_LOWBEAM\_ON  
The V2X Message API (US), [23](#)
- J2735\_2016\_EXT\_PARKINGLIGHTS\_ON  
The V2X Message API (US), [23](#)
- J2735\_2016\_EXT\_RIGHTTURN\_ON  
The V2X Message API (US), [23](#)
- J2735\_2016\_HS\_FROM00\_0\_TO\_22\_5\_DEGREES  
The V2X Message API (US), [23](#)
- J2735\_2016\_HS\_FROM112\_5\_TO\_135\_0\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM135\_0\_TO\_157\_5\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM157\_5\_TO\_180\_0\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM180\_0\_TO\_202\_5\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM202\_5\_TO\_225\_0\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM225\_0\_TO\_247\_5\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM22\_5\_TO\_45\_0\_DEGREES  
The V2X Message API (US), [23](#)
- J2735\_2016\_HS\_FROM247\_5\_TO\_270\_0\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM270\_0\_TO\_292\_5\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM292\_5\_TO\_315\_0\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM315\_0\_TO\_337\_5\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM337\_5\_TO\_360\_0\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM45\_0\_TO\_67\_5\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM67\_5\_TO\_90\_0\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_HS\_FROM90\_0\_TO\_112\_5\_DEGREES  
The V2X Message API (US), [24](#)
- J2735\_2016\_INDIVIDUAL\_MOTOR\_TRAFFIC  
v2x\_msg\_common.h, [198](#)
- J2735\_2016\_ISEC\_STATUS\_FAILURE\_FLASH  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_FAILURE\_MODE  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_FIXED\_TIME\_OPERATION  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_MANUAL\_CONTROL\_IS\_ON  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_NOVALID\_MAP\_IS\_AVAILABLE  
v2x\_msg\_spat.h, [218](#)
- J2735\_2016\_ISEC\_STATUS\_NOVALID\_SPAT\_IS\_AVAILABLE  
v2x\_msg\_spat.h, [218](#)
- J2735\_2016\_ISEC\_STATUS\_OFF  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_PREEMPT\_IS\_ACTIVE  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_RECENT\_CHANGE\_IN\_MAP\_ASSIGNED\_LANEIDS\_USED  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_RECENT\_MAP\_MESSAGE\_UPDATE  
v2x\_msg\_spat.h, [217](#)

- J2735\_2016\_ISEC\_STATUS\_STAND\_BY\_OPERATI-  
ON  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_STOP\_TIME\_IS\_ACTIV-  
E  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_TRAFFIC\_DEPENDENT-  
\_OPERATION  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_ISEC\_STATUS\_TSP\_IS\_ACTIVE  
v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_LANE\_DIRECTION\_EGRESS  
The V2X Message API (US), [26](#)
- J2735\_2016\_LANE\_DIRECTION\_INGRESS  
The V2X Message API (US), [26](#)
- J2735\_2016\_LANE\_DIRECTION\_UNAVAIL  
The V2X Message API (US), [26](#)
- J2735\_2016\_MAN\_CAUTION  
The V2X Message API (US), [23](#)
- J2735\_2016\_MAN\_GOWITH\_HALT  
The V2X Message API (US), [23](#)
- J2735\_2016\_MAN\_LANE\_CHANGE\_ALLOWED  
The V2X Message API (US), [23](#)
- J2735\_2016\_MAN\_LEFT\_ALLOWED  
The V2X Message API (US), [22](#)
- J2735\_2016\_MAN\_LEFT\_TURN\_ON\_RED\_ALLOWE-  
D  
The V2X Message API (US), [23](#)
- J2735\_2016\_MAN\_NO\_STOPPING\_ALLOWED  
The V2X Message API (US), [23](#)
- J2735\_2016\_MAN\_RIGHT\_ALLOWED  
The V2X Message API (US), [22](#)
- J2735\_2016\_MAN\_RIGHT\_TURN\_ON\_RED\_ALLOW-  
ED  
The V2X Message API (US), [23](#)
- J2735\_2016\_MAN\_STRAIGHT\_ALLOWED  
The V2X Message API (US), [22](#)
- J2735\_2016\_MAN\_UTURN\_ALLOWED  
The V2X Message API (US), [23](#)
- J2735\_2016\_MAN\_YIELD\_ALLWAYS\_REQUIRED  
The V2X Message API (US), [23](#)
- J2735\_2016\_MAP\_MSG\_P  
The V2X Message API (US), [26](#)
- J2735\_2016\_MULTI\_LANE\_TREATED\_AS\_ONE  
v2x\_msg\_common.h, [198](#)
- J2735\_2016\_NODE\_TYPE\_COMP  
The V2X Message API (US), [25](#)
- J2735\_2016\_NODE\_TYPE\_NODE\_OBJS  
The V2X Message API (US), [25](#)
- J2735\_2016\_NODE\_TYPE\_NOT\_PRESENT  
The V2X Message API (US), [25](#)
- J2735\_2016\_NODEATTR\_DISABLED\_SEGMENTS  
The V2X Message API (US), [24](#)
- J2735\_2016\_NODEATTR\_ENABLED\_SEGMENTS  
The V2X Message API (US), [24](#)
- J2735\_2016\_NODEATTR\_LANEDATA  
The V2X Message API (US), [24](#)
- J2735\_2016\_NODEATTR\_NODEATTR  
The V2X Message API (US), [24](#)
- J2735\_2016\_NODEATTR\_UNAVAIL  
The V2X Message API (US), [24](#)
- J2735\_2016\_NODESET\_XY  
v2x\_msg\_tim.h, [231](#)
- J2735\_2016\_OFFSET\_TYPE\_LL  
v2x\_msg\_tim.h, [231](#)
- J2735\_2016\_OFFSET\_TYPE\_XY  
v2x\_msg\_tim.h, [231](#)
- J2735\_2016\_OTHER\_NON\_MOTORIZED  
v2x\_msg\_common.h, [198](#)
- J2735\_2016\_OVERLAP\_LANE\_DESCR  
v2x\_msg\_common.h, [198](#)
- J2735\_2016\_PED\_TRAFFIC  
v2x\_msg\_common.h, [198](#)
- J2735\_2016\_PROJECTION\_GEO\_PROJECTION  
v2x\_msg\_tim.h, [231](#)
- J2735\_2016\_PROJECTION\_OFFSET\_SYSTEM  
v2x\_msg\_tim.h, [231](#)
- J2735\_2016\_PROJECTION\_VALID\_REGION  
v2x\_msg\_tim.h, [231](#)
- J2735\_2016\_RTCM\_CORRECTIONS\_MSG\_D  
The V2X Message API (US), [26](#)
- J2735\_2016\_RTCM\_CORRECTIONS\_MSG\_U  
The V2X Message API (US), [26](#)
- J2735\_2016\_SPaT\_MSG\_P  
The V2X Message API (US), [26](#)
- J2735\_2016\_SRM\_MSG\_U  
The V2X Message API (US), [26](#)
- J2735\_2016\_SSM\_MSG\_U  
The V2X Message API (US), [26](#)
- J2735\_2016\_TAXI\_VEHICLE\_TRAFFIC  
v2x\_msg\_common.h, [198](#)
- J2735\_2016\_TIM\_MSG\_D  
The V2X Message API (US), [26](#)
- J2735\_2016\_TIM\_MSG\_P  
The V2X Message API (US), [26](#)
- J2735\_2016\_TIM\_MSG\_U  
The V2X Message API (US), [26](#)
- J2735\_2016\_TRACKED\_VEHICLE\_TRAFFIC  
v2x\_msg\_common.h, [198](#)
- J2735\_2016\_VEH\_HOV\_USE\_ONLY  
The V2X Message API (US), [27](#)
- J2735\_2016\_VEH\_IS\_VEHICLE\_FLYOVER\_LANE  
The V2X Message API (US), [27](#)
- J2735\_2016\_VEH\_IS\_VEHICLE\_REVOCABLE\_LANE  
The V2X Message API (US), [27](#)
- J2735\_2016\_VEH\_RESTRICTED\_FROM\_PUBLIC\_U-  
SE  
The V2X Message API (US), [27](#)
- J2735\_2016\_VEH\_RESTRICTED\_TO\_BUS\_USE  
The V2X Message API (US), [27](#)
- J2735\_2016\_VEH\_RESTRICTED\_TO\_TAXI\_USE  
The V2X Message API (US), [27](#)
- J2735\_2016\_WIPER\_AUTOMATIC\_PRESENT  
The V2X Message API (US), [25](#)
- J2735\_2016\_WIPER\_HIGH  
The V2X Message API (US), [25](#)

- J2735\_2016\_WIPER\_INTERMITTENT  
The V2X Message API (US), [25](#)
- J2735\_2016\_WIPER\_LOW  
The V2X Message API (US), [25](#)
- J2735\_2016\_WIPER\_STATUS\_OFF  
The V2X Message API (US), [25](#)
- J2735\_2016\_WIPER\_STATUS\_UNAVAIL  
The V2X Message API (US), [25](#)
- J2735\_2016\_WIPER\_WASHER\_IN\_USE  
The V2X Message API (US), [25](#)
- J2735\_CONTENT\_ADVISORY  
v2x\_msg\_tim.h, [231](#)
- J2735\_CONTENT\_EXIT\_SERVICES  
v2x\_msg\_tim.h, [231](#)
- J2735\_CONTENT\_GENERIC\_SIGNS  
v2x\_msg\_tim.h, [231](#)
- J2735\_CONTENT\_SPEED\_LIMITS  
v2x\_msg\_tim.h, [231](#)
- J2735\_CONTENT\_WORKZONES  
v2x\_msg\_tim.h, [231](#)
- J2735\_DIRECTION\_BOTH  
v2x\_msg\_tim.h, [232](#)
- J2735\_DIRECTION\_FORWARD  
v2x\_msg\_tim.h, [232](#)
- J2735\_DIRECTION\_REVERSE  
v2x\_msg\_tim.h, [232](#)
- J2735\_DIRECTION\_UNAVAIL  
v2x\_msg\_tim.h, [232](#)
- J2735\_GPS\_STATUS\_BASE\_STATION\_TYPE  
v2x\_msg\_pvd.h, [212](#)
- J2735\_GPS\_STATUS\_INVIEW\_OF\_UNDER\_5  
v2x\_msg\_pvd.h, [212](#)
- J2735\_GPS\_STATUS\_IS\_HEALTHY  
v2x\_msg\_pvd.h, [212](#)
- J2735\_GPS\_STATUS\_IS\_MONITORED  
v2x\_msg\_pvd.h, [212](#)
- J2735\_GPS\_STATUS\_LOCAL\_CORRECTIONS\_PRESENT  
v2x\_msg\_pvd.h, [212](#)
- J2735\_GPS\_STATUS\_NETWORK\_CORRECTIONS\_PRESENT  
v2x\_msg\_pvd.h, [212](#)
- J2735\_GPS\_STATUS\_PDOP\_UNDER\_5  
v2x\_msg\_pvd.h, [212](#)
- J2735\_GPS\_STATUS\_UNAVAIL  
v2x\_msg\_pvd.h, [212](#)
- J2735\_MUTCD\_CODE\_GUIDE  
v2x\_msg\_tim.h, [232](#)
- J2735\_MUTCD\_CODE\_MAINTENANCE  
v2x\_msg\_tim.h, [232](#)
- J2735\_MUTCD\_CODE\_MOTORIST\_SERVICE  
v2x\_msg\_tim.h, [232](#)
- J2735\_MUTCD\_CODE\_NONE  
v2x\_msg\_tim.h, [232](#)
- J2735\_MUTCD\_CODE\_REC  
v2x\_msg\_tim.h, [232](#)
- J2735\_MUTCD\_CODE\_REGULATORY  
v2x\_msg\_tim.h, [232](#)
- J2735\_MUTCD\_CODE\_WARNING  
v2x\_msg\_tim.h, [232](#)
- J2735\_REGION\_CIRCLE  
v2x\_msg\_tim.h, [233](#)
- J2735\_REGION\_NOTHING  
v2x\_msg\_tim.h, [233](#)
- J2735\_REGION\_POLYGON  
v2x\_msg\_tim.h, [233](#)
- J2735\_REGION\_SHAPEPT  
v2x\_msg\_tim.h, [233](#)
- J2735\_TIM\_TYPE\_ADVISORY  
v2x\_msg\_tim.h, [233](#)
- J2735\_TIM\_TYPE\_COMMERCIAL\_SIGNAGE  
v2x\_msg\_tim.h, [233](#)
- J2735\_TIM\_TYPE\_ROAD\_SIGNAGE  
v2x\_msg\_tim.h, [233](#)
- J2735\_TIM\_TYPE\_UNKNOWN  
v2x\_msg\_tim.h, [233](#)
- J2735\_2016\_AllowedManeuvers\_t  
The V2X Message API (US), [22](#)
- J2735\_2016\_Computed\_Lanes, [92](#)
- J2735\_2016\_Computed\_Lanes\_t  
v2x\_msg\_tim.h, [229](#)
- J2735\_2016\_Connectsto\_Lanes\_t  
The V2X Message API (US), [20](#)
- J2735\_2016\_DSRC\_MsgID\_t  
The V2X Message API (US), [20](#)
- J2735\_2016\_DirofUse\_t  
The V2X Message API (US), [23](#)
- J2735\_2016\_EnableLaneList\_t  
v2x\_msg\_spat.h, [216](#)
- J2735\_2016\_ExteriorLights\_t  
The V2X Message API (US), [23](#)
- J2735\_2016\_GPSStatus\_t  
v2x\_msg\_pvd.h, [212](#)
- J2735\_2016\_GeometricProjection, [93](#)  
circle, [93](#)  
extent, [93](#)  
heading\_slice, [93](#)  
lane\_width, [93](#)
- J2735\_2016\_GeometricProjection\_t  
v2x\_msg\_tim.h, [229](#)
- J2735\_2016\_HeadingSlice  
The V2X Message API (US), [23](#)
- J2735\_2016\_HeadingSlice\_t  
The V2X Message API (US), [20](#)
- J2735\_2016\_Intersection\_t  
v2x\_msg\_spat.h, [216](#)
- J2735\_2016\_IntersectionStatus\_t  
v2x\_msg\_spat.h, [216](#)
- J2735\_2016\_IsecGeoDescr\_t  
v2x\_msg\_map.h, [208](#)
- J2735\_2016\_LaneAttribute\_t  
The V2X Message API (US), [20](#)
- J2735\_2016\_LaneDir\_t  
The V2X Message API (US), [20](#)
- J2735\_2016\_LaneList\_t  
The V2X Message API (US), [20](#)

- J2735\_2016\_LaneTypeAttrCrossWk\_t
  - The V2X Message API (US), [20](#)
- J2735\_2016\_LaneTypeAttrVeh\_t
  - The V2X Message API (US), [20](#)
- J2735\_2016\_LatLong, [93](#)
  - latlong\_offsets, [94](#)
- J2735\_2016\_LatLong\_t
  - v2x\_msg\_tim.h, [229](#)
- J2735\_2016\_ManeuverAssistList\_t
  - v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_MovementEventList\_t
  - v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_MovementState\_t
  - v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_NodeAttr\_t
  - The V2X Message API (US), [20](#)
- J2735\_2016\_NodeAttrType\_t
  - The V2X Message API (US), [24](#)
- J2735\_2016\_NodeList\_t
  - The V2X Message API (US), [20](#)
- J2735\_2016\_NodePoint\_t
  - The V2X Message API (US), [20](#)
- J2735\_2016\_NodeSet\_XY, [94](#)
  - n\_node\_offsets, [94](#)
  - node\_offsets, [94](#)
- J2735\_2016\_NodeSet\_XY\_t
  - v2x\_msg\_tim.h, [229](#)
- J2735\_2016\_NodeType\_t
  - The V2X Message API (US), [24](#)
- J2735\_2016\_NodeXY, [94](#)
  - computed\_lanes, [95](#)
  - node\_xy\_type, [95](#)
  - nodeset, [95](#)
- J2735\_2016\_NodeXY\_t
  - v2x\_msg\_tim.h, [229](#)
- J2735\_2016\_NodeXY\_type\_t
  - v2x\_msg\_tim.h, [231](#)
- J2735\_2016\_OffsetSystem, [95](#)
  - node\_set\_ll, [95](#)
  - node\_set\_xy, [95](#)
  - off\_sys\_type, [95](#)
  - zoom\_level, [95](#)
- J2735\_2016\_OffsetSystem\_t
  - v2x\_msg\_tim.h, [229](#)
- J2735\_2016\_OffsetSystemType\_t
  - v2x\_msg\_tim.h, [231](#)
- J2735\_2016\_PHObject\_t
  - v2x\_msg\_pvd.h, [212](#)
- J2735\_2016\_PathHistory\_t
  - The V2X Message API (US), [20](#)
- J2735\_2016\_PathPrediction\_t
  - The V2X Message API (US), [21](#)
- J2735\_2016\_PriorityReqType\_t
  - v2x\_msg\_srm.h, [221](#)
- J2735\_2016\_ProbeSnapshot\_t
  - v2x\_msg\_pvd.h, [212](#)
- J2735\_2016\_ProjectionType\_t
  - v2x\_msg\_tim.h, [231](#)
- J2735\_2016\_RegionOffsets, [96](#)
- J2735\_2016\_RequestorType\_t, [96](#)
- J2735\_2016\_SignalRequest, [96](#)
  - basic\_vehicle\_role, [97](#)
  - msgcount, [97](#)
  - request\_id, [97](#)
  - requestor\_type, [97](#)
  - requestor\_type\_present, [97](#)
  - vehicle\_id, [97](#)
- J2735\_2016\_SignalRequest\_t, [97](#)
  - iid, [98](#)
  - in\_bound\_access\_point, [98](#)
  - out\_bound\_access\_point, [98](#)
  - req\_type, [98](#)
  - request\_id, [98](#)
- J2735\_2016\_SignalRequesterInfo\_t
  - v2x\_msg\_ssm.h, [222](#)
- J2735\_2016\_SignalRequestorInfo\_t, [98](#)
- J2735\_2016\_SignalStatusList, [98](#)
  - iid, [99](#)
  - msgcount, [99](#)
  - ss\_pkg\_list, [99](#)
  - ss\_pkg\_list\_len, [99](#)
- J2735\_2016\_SignalStatusList\_t
  - v2x\_msg\_ssm.h, [222](#)
- J2735\_2016\_SignalStatusPackageList, [99](#)
  - duration, [99](#)
  - in\_bound\_access\_point, [99](#)
  - min\_of\_year, [100](#)
  - out\_bound\_access\_point, [100](#)
  - prio\_resp, [100](#)
  - second, [100](#)
  - signal\_request, [100](#)
  - valid\_out\_bound\_access\_point, [100](#)
- J2735\_2016\_SignalStatusPackageList\_t
  - v2x\_msg\_ssm.h, [222](#)
- J2735\_2016\_TimeChange\_t
  - v2x\_msg\_spat.h, [217](#)
- J2735\_2016\_VehSafetyExtensions\_t
  - v2x\_msg\_pvd.h, [212](#)
- J2735\_2016\_VehicleClassification\_t, [102](#)
- J2735\_2016\_VehicleEventFlags\_t
  - The V2X Message API (US), [25](#)
- J2735\_2016\_VehicleIdent\_t, [102](#)
- J2735\_2016\_VehicleStatus\_t
  - v2x\_msg\_pvd.h, [212](#)
- J2735\_2016\_WiperSet\_t
  - The V2X Message API (US), [21](#)
- J2735\_2016\_WiperStatus\_t
  - The V2X Message API (US), [25](#)
- j2735\_2016\_fullposvec, [92](#)
- j2735\_2016\_srm, [100](#)
  - dsecond, [101](#)
  - msg\_count, [101](#)
  - requestor, [101](#)
  - srm\_list\_count, [101](#)
  - srmlist, [101](#)
- j2735\_2016\_ssm, [101](#)

- dsecond, [101](#)
- msgcount, [101](#)
- sslist, [101](#)
- sslist\_len, [102](#)
- J2735\_HS\_NO\_HEADING
  - v2x\_msg\_tim.h, [235](#)
- j2735\_dsrc\_msgids
  - The V2X Message API (US), [25](#)
- j2735\_free\_tim\_contents\_only
  - v2x\_msg\_tim.h, [229](#)
- j2735\_free\_tim\_struct
  - v2x\_msg\_tim.h, [229](#)
- J2735CircularRegion, [102](#)
  - position, [103](#)
  - radius\_type, [103](#)
  - value, [103](#)
- J2735CircularRegion\_t
  - v2x\_msg\_tim.h, [229](#)
- J2735ContentType
  - v2x\_msg\_tim.h, [231](#)
- J2735ContentType\_t
  - v2x\_msg\_tim.h, [230](#)
- J2735DATAFRAME\_t
  - v2x\_msg\_tim.h, [230](#)
- J2735DirectionOfUse
  - v2x\_msg\_tim.h, [231](#)
- J2735DirectionOfUse\_t
  - v2x\_msg\_tim.h, [230](#)
- J2735Extent
  - The V2X Message API (US), [26](#)
- J2735Extent\_t
  - The V2X Message API (US), [21](#)
- J2735HeadingSlice
  - v2x\_msg\_tim.h, [232](#)
- J2735MUTCDCode
  - v2x\_msg\_tim.h, [232](#)
- J2735MUTCDCode\_t
  - v2x\_msg\_tim.h, [230](#)
- J2735Position3D, [103](#)
  - elevation, [103](#)
  - latitude, [103](#)
  - longitude, [104](#)
- J2735Position3D\_t
  - The V2X Message API (US), [21](#)
- J2735RadiusType
  - v2x\_msg\_tim.h, [232](#)
- J2735RadiusType\_t
  - v2x\_msg\_tim.h, [230](#)
- J2735RegionPointSet, [104](#)
  - num\_wp, [104](#)
  - position, [104](#)
  - reg\_offsets, [104](#)
  - region\_offset\_len, [104](#)
  - wplist, [104](#)
  - zoom\_level, [104](#)
- J2735Regions
  - v2x\_msg\_tim.h, [232](#)
- J2735RegionsList\_t
  - v2x\_msg\_tim.h, [230](#)
- J2735RoadSignID, [105](#)
  - heading\_slice, [105](#)
  - mutcd\_code, [105](#)
  - position, [105](#)
- J2735RoadSignID\_t
  - v2x\_msg\_tim.h, [230](#)
- J2735SRMList\_t
  - v2x\_msg\_srm.h, [220](#)
- J2735SRMRequestor\_t
  - v2x\_msg\_srm.h, [220](#)
- J2735ShapePointSet, [105](#)
  - lanewidth, [106](#)
  - num\_wp, [106](#)
  - position, [106](#)
  - wplist, [106](#)
- J2735ShapePoints\_t
  - v2x\_msg\_tim.h, [230](#)
- J2735TIM, [106](#)
  - advisory\_codes, [107](#)
  - content\_type, [107](#)
  - duration, [107](#)
  - exitservice\_codes, [107](#)
  - further\_info\_id, [107](#)
  - genericsign\_codes, [107](#)
  - geographical\_path, [107](#)
  - n\_geo\_paths, [107](#)
  - num\_advisories, [107](#)
  - num\_exitservices, [107](#)
  - num\_genericsigns, [107](#)
  - num\_speedlimits, [107](#)
  - num\_workzones, [108](#)
  - priority, [108](#)
  - road\_sign\_id, [108](#)
  - speedlimit\_codes, [108](#)
  - ssp\_loc\_rights, [108](#)
  - ssp\_msg\_rights1, [108](#)
  - ssp\_msg\_rights2, [108](#)
  - ssp\_tim\_rights, [108](#)
  - start\_minute\_of\_year, [108](#)
  - start\_year, [108](#)
  - type, [108](#)
  - workzone\_codes, [109](#)
- J2735TIMGeoPath, [109](#)
  - closedpath, [109](#)
  - directionality, [109](#)
  - geometric\_projection, [109](#)
  - heading\_slice, [109](#)
  - id, [110](#)
  - lane\_width, [110](#)
  - offset\_system, [110](#)
  - pos3d, [110](#)
  - projection\_type, [110](#)
  - valid\_pos3d, [110](#)
  - valid\_region, [110](#)
- J2735TIMGeoPath\_t
  - v2x\_msg\_tim.h, [230](#)
- J2735TravelerInfoType



- v2x\_msg\_tim.h, 233
- J2735TravelerInfoType\_t
  - v2x\_msg\_tim.h, 230
- J2735ValidRegion, 110
  - circle, 111
  - extent, 111
  - heading\_slice, 111
  - region\_flag, 111
  - regionpoint, 111
  - shape\_points, 111
- J2735ValidRegion\_t
  - v2x\_msg\_tim.h, 230
- J2735VehicleID\_t, 111
- LIBWME\_ACTION\_ADD
  - The V2X Network and Transport module API (US), 54
- LIBWME\_ACTION\_CHANGE
  - The V2X Network and Transport module API (US), 54
- LIBWME\_ACTION\_DELETE
  - The V2X Network and Transport module API (US), 54
- LIBWME\_RC\_ACCEPTED
  - The V2X Network and Transport module API (US), 54
- LIBWME\_RC\_INVALID\_PARAMETERS
  - The V2X Network and Transport module API (US), 54
- LIBWME\_RC\_UNSPECIFIED
  - The V2X Network and Transport module API (US), 54
- LIBWME\_USER\_AUTOACCESS\_ONMATCH
  - The V2X Network and Transport module API (US), 53
- LIBWME\_USER\_AUTOACCESS\_UNCOND
  - The V2X Network and Transport module API (US), 53
- LIBWME\_USER\_NOSCHACCESS
  - The V2X Network and Transport module API (US), 53
- lane\_attr
  - laneset, 114
- lane\_data\_attr
  - nodeattrs, 116
- lane\_id
  - \_J2735\_2016\_EnableLaneList\_t, 71
  - etsi\_connects, 87
  - laneset, 114
  - v2x\_etsi\_lane, 144
- lane\_position
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- lane\_validity\_params, 112
- lane\_validity\_time, 112
- lane\_width
  - J2735\_2016\_GeometricProjection, 93
  - J2735TIMGeoPath, 110
- laneattr, 112
  - crwattr, 112
  - direction, 112
  - sharedlane, 112
  - v2x\_etsi\_lane, 144
  - vehattr, 113
- lanedataattr\_list, 113
- lanedirection
  - The V2X Message API (US), 26
- lanelist
  - intersection\_geodescr, 91
- laneset, 113
  - connects\_to, 114
  - lane\_attr, 114
  - lane\_id, 114
  - man, 114
  - n\_connect, 114
  - n\_nodepoint, 114
  - node\_point, 114
  - phase, 114
  - signal\_group\_id, 114
  - v2x\_etsi\_map\_isec, 145
- laneset\_len
  - v2x\_etsi\_map\_isec, 145
- laneshare
  - v2x\_msg\_common.h, 197
- lanetypecrosswalk
  - The V2X Message API (US), 26
- lanetypevehicle
  - The V2X Message API (US), 27
- lanewidth
  - intersection\_geodescr, 92
  - J2735ShapePointSet, 106
  - odelist, 117
  - v2x\_etsi\_map\_isec, 146
- lat\_offset
  - path\_history, 118
  - pathhistoryframe, 119
- lataccel
  - v2x\_msg\_bsm, 155
- lateral\_accel
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- lateral\_accel\_conf
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- latitude
  - J2735Position3D, 103
  - pf\_gnss\_data, 120
  - v2x\_msg\_bsm, 155
  - v2x\_msg\_psm, 160
- latlong\_offsets
  - J2735\_2016\_LatLong, 94
- lb\_siren
  - v2x\_etsi\_denm\_rwc, 141
- lc
  - v2x\_etsi\_denm, 137
- leftturnsignal
  - v2x\_msg\_bsm, 155
- length
  - v2x\_msg\_bsm, 155
- lf

- v2x\_etsi\_cam, 132
- libcan\_filters, 114
- libwme\_gpsinfo, 115
- libwme\_radio\_settings, 115
- libwme\_radio\_settings::libwme\_radio, 115
- lifetime
  - savari1609Wra, 123
  - v2x\_net\_wme.h, 246
- lightbar\_in\_use
  - \_J2735\_2016\_VehicleStatus, 78
  - v2x\_emergency\_details, 131
  - v2x\_msg\_bsm, 155
- lightbarinuse
  - v2x\_msg\_bsm, 155
- likely\_time
  - \_J2735\_2016\_TimeChange\_t, 77
  - v2x\_etsi\_spat\_mm\_evt, 149
- linked\_cause
  - v2x\_etsi\_denm\_sc, 143
- local\_service\_index
  - v2x\_net\_wme.h, 246
  - v2x\_wme\_reg\_req, 165
- long\_offset
  - path\_history, 118
  - pathhistoryframe, 119
- longaccel
  - v2x\_msg\_bsm, 155
- longitude
  - J2735Position3D, 104
  - pf\_gnss\_data, 121
  - v2x\_msg\_bsm, 155
  - v2x\_msg\_psm, 161
- longitude\_accel
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- longitude\_accel\_conf
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- lowbeam
  - v2x\_msg\_bsm, 155
- mac
  - v2x\_net\_wme.h, 246
  - v2x\_wsmp\_send\_req, 172
- man
  - laneset, 114
- maneuver\_assist
  - intersection\_data, 90
- maneuvers
  - etsi\_connects, 87
  - v2x\_etsi\_lane, 144
- map\_msg
  - v2x\_etsi\_map, 145
- max\_end\_time
  - \_J2735\_2016\_TimeChange\_t, 77
  - v2x\_etsi\_spat\_mm\_evt, 149
- mc
  - v2x\_etsi\_denm, 137
- message\_id
  - v2x\_msg\_tim, 164
- min\_end\_time
  - \_J2735\_2016\_TimeChange\_t, 77
  - v2x\_etsi\_spat\_mm\_evt, 149
- min\_of\_the\_year
  - intersection\_data, 90
- min\_of\_year
  - \_J2735\_2016\_PVD, 76
  - J2735\_2016\_SignalStatusPackageList, 100
  - srm\_list, 126
- min\_of\_yr
  - v2x\_etsi\_spat\_isec, 148
- minute
  - utcTimeFrame, 129
- minute\_of\_the\_year
  - v2x\_msg\_spat, 162
- mm\_evt
  - v2x\_etsi\_spat\_mm\_state, 150
- mm\_evt\_len
  - v2x\_etsi\_spat\_mm\_state, 150
- mmlist
  - v2x\_etsi\_spat\_isec, 148
- mmlist\_len
  - v2x\_etsi\_spat\_isec, 148
- mode
  - pf\_gnss\_data, 121
- month
  - utcTimeFrame, 129
- movement\_event
  - \_J2735\_2016\_MovementState\_t, 73
- movement\_states
  - intersection\_data, 90
- msg\_count
  - j2735\_2016\_srm, 101
  - v2x\_etsi\_map\_isec, 146
  - v2x\_etsi\_map\_msg, 146
  - v2x\_etsi\_spat\_isec, 148
- msg\_crc
  - v2x\_msg\_tim, 164
- msg\_revision
  - intersection\_geodescr, 92
  - v2x\_msg\_map, 159
- msg\_subid
  - v2x\_etsi\_map\_msg, 146
- msgCount
  - intersection\_data, 90
- msgcount
  - J2735\_2016\_SignalRequest, 97
  - J2735\_2016\_SignalStatusList, 99
  - j2735\_2016\_ssm, 101
  - v2x\_msg\_bsm, 155
  - v2x\_msg\_psm, 161
- msgid
  - v2x\_msg\_bsm, 155
- msgrate
  - testdataframe, 128
- msgseqnum
  - testdataframe, 128
- multi\_vehicle\_response
  - v2x\_emergency\_details, 131

- mutcd\_code
  - J2735RoadSignID, 105
- n\_connect
  - laneset, 114
- n\_geo\_descr
  - v2x\_msg\_map, 159
- n\_geo\_paths
  - J2735TIM, 107
- n\_intersections
  - v2x\_msg\_spat, 162
- n\_lane\_data\_attr
  - nodeattrs, 116
- n\_lane\_list
  - intersection\_data, 91
- n\_lanelist
  - intersection\_geodescr, 92
- n\_node\_offsets
  - J2735\_2016\_NodeSet\_XY, 94
- n\_nodepoint
  - laneset, 114
- n\_zones
  - v2x\_etsi\_cam\_hf\_rsu, 133
- next\_time
  - \_J2735\_2016\_TimeChange\_t, 77
  - v2x\_etsi\_spat\_mm\_evt, 149
- node
  - odelist, 117
- node\_attr
  - etsi\_node, 88
- node\_id
  - nodepoint\_list, 117
- node\_offsets
  - J2735\_2016\_NodeSet\_XY, 94
- node\_point
  - etsi\_node, 88
  - laneset, 114
- node\_set\_ll
  - J2735\_2016\_OffsetSystem, 95
- node\_set\_xy
  - J2735\_2016\_OffsetSystem, 95
- node\_xy\_type
  - J2735\_2016\_NodeXY, 95
- nodeattr
  - nodepoint\_list, 117
- nodeattrs, 115
  - dElev, 116
  - dWidth, 116
  - lane\_data\_attr, 116
  - n\_lane\_data\_attr, 116
  - type, 116
- odelist, 116
  - lanewidth, 117
  - node, 117
- nodepoint\_list, 117
  - node\_id, 117
  - nodeattr, 117
- nodes
  - v2x\_etsi\_lane, 144
- nodes\_len
  - v2x\_etsi\_lane, 144
- nodeset
  - J2735\_2016\_NodeXY, 95
- num\_advisories
  - J2735TIM, 107
- num\_dataframes
  - v2x\_msg\_tim, 164
- num\_exitservices
  - J2735TIM, 107
- num\_genericsigns
  - J2735TIM, 107
- num\_maneuver\_assist
  - intersection\_data, 91
- num\_movement\_event
  - \_J2735\_2016\_MovementState\_t, 73
- num\_movement\_states
  - intersection\_data, 91
- num\_rx
  - v2x\_net\_wme.h, 246
  - v2x\_wsmp\_rcv\_indication, 170
- num\_speedlimits
  - J2735TIM, 107
- num\_workzones
  - J2735TIM, 108
- num\_wp
  - J2735RegionPointSet, 104
  - J2735ShapePointSet, 106
- numobe
  - testdataframe, 128
- obeid
  - testdataframe, 128
- off\_sys\_type
  - J2735\_2016\_OffsetSystem, 95
- offset\_system
  - J2735TIMGeoPath, 110
- out\_bound\_access\_point
  - J2735\_2016\_SignalRequest\_t, 98
  - J2735\_2016\_SignalStatusPackageList, 100
- P16093\_RADIO\_INTF1
  - v2x\_net\_wme.h, 240
- P16093\_RADIO\_INTF2
  - v2x\_net\_wme.h, 240
- P16093\_RADIO\_INTFX
  - v2x\_net\_wme.h, 240
- P16093\_SRV\_IP
  - v2x\_net\_wme.h, 241
- P1609\_CHANNEL\_ACCESS\_ALTERNATING
  - The V2X Network and Transport module API (US), 54
- P1609\_CHANNEL\_ACCESS\_CONTINUOUS
  - The V2X Network and Transport module API (US), 54
- P1609\_CHANNEL\_ACCESS\_NOSCHACCESS
  - The V2X Network and Transport module API (US), 54
- P1609\_CHANNEL\_ACCESS\_ONMATCH



- The V2X Network and Transport module API (US), [54](#)
- P1609\_CHANNEL\_ACCESS\_UNKNOWN
  - The V2X Network and Transport module API (US), [54](#)
- P1609\_WSA\_SECURED
  - The V2X Network and Transport module API (US), [54](#)
- P1609\_WSA\_UNSECURED
  - The V2X Network and Transport module API (US), [54](#)
- P1609\_ADVID\_LEN\_MAX
  - [v2x\\_net\\_wme.h](#), [241](#)
- P1609\_CHANNEL\_ACCESS
  - The V2X Network and Transport module API (US), [58](#)
- P1609\_WSATYPE
  - The V2X Network and Transport module API (US), [58](#)
- PSM\_DEVICE\_USAGE\_STATE\_CALLING
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USAGE\_STATE\_IDLE
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USAGE\_STATE\_LISTENING\_TO\_AUDIO
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USAGE\_STATE\_OTHER
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USAGE\_STATE\_PLAYING\_GAMES
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USAGE\_STATE\_READING
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USAGE\_STATE\_TYPING
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USAGE\_STATE\_UNAVAIL
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USAGE\_STATE\_VIEWING
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USER\_TYPE\_ANIMAL
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USER\_TYPE\_PEDALCYCLIST
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USER\_TYPE\_PEDESTRIAN
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USER\_TYPE\_PUBLICSAFETYWORKER
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSM\_DEVICE\_USER\_TYPE\_UNAVAILABLE
  - [v2x\\_msg\\_psm.h](#), [210](#)
- PSID\_LEN
  - [v2x\\_net\\_wme.h](#), [241](#)
- parkinglights
  - [v2x\\_msg\\_bsm](#), [155](#)
- path\_history, [117](#)
  - [elev\\_offset](#), [118](#)
  - [heading](#), [118](#)
  - [lat\\_offset](#), [118](#)
  - [long\\_offset](#), [118](#)
  - [positional\\_accuracy](#), [118](#)
  - [time\\_offset](#), [118](#)
- pathhistoryframe, [118](#)
  - [elev\\_offset](#), [119](#)
  - [heading](#), [119](#)
  - [lat\\_offset](#), [119](#)
  - [long\\_offset](#), [119](#)
  - [positional\\_accuracy](#), [119](#)
  - [time\\_offset](#), [119](#)
- pdu
  - [v2x\\_etsi\\_cam](#), [132](#)
  - [v2x\\_etsi\\_denm](#), [137](#)
  - [v2x\\_etsi\\_map](#), [145](#)
  - [v2x\\_etsi\\_spat](#), [147](#)
- performance\_class
  - [v2x\\_etsi\\_cam\\_hf\\_vehicle](#), [135](#)
- personal\_device\_usage\_state
  - [v2x\\_msg\\_psm.h](#), [210](#)
- personal\_device\_usage\_state\_t
  - [v2x\\_msg\\_psm.h](#), [210](#)
- personal\_device\_use\_type
  - [v2x\\_msg\\_psm](#), [161](#)
- personal\_device\_user\_type
  - [v2x\\_msg\\_psm.h](#), [210](#)
- personal\_device\_user\_type\_t
  - [v2x\\_msg\\_psm.h](#), [210](#)
- pf\_gnss\_data, [119](#)
  - [altitude](#), [120](#)
  - [dsecond](#), [120](#)
  - [heading](#), [120](#)
  - [latitude](#), [120](#)
  - [longitude](#), [121](#)
  - [mode](#), [121](#)
  - [speed](#), [121](#)
  - [time](#), [121](#)
  - [utc\\_time](#), [121](#)
- pf\_gnss\_data\_t
  - The V2X Sensors API, [64](#)
- pf\_gnss\_get\_nmea
  - [v2x\\_sensor\\_gnss.h](#), [254](#)
- pf\_gnss\_get\_pfname
  - [v2x\\_sensor\\_gnss.h](#), [254](#)
- ph
  - [\\_J2735\\_2016\\_VehSafetyExtensions](#), [79](#)
  - [v2x\\_etsi\\_cam\\_if](#), [137](#)
- ph\_points
  - [\\_J2735\\_2016\\_PHObject](#), [74](#)
- ph\_points\_len
  - [\\_J2735\\_2016\\_PHObject](#), [74](#)
- phase
  - [laneset](#), [114](#)
- phase\_no
  - [v2x\\_etsi\\_lane](#), [144](#)
- phase\_state
  - [\\_J2735\\_2016\\_MovementEventList\\_t](#), [72](#)
  - [v2x\\_etsi\\_spat\\_mm\\_evt](#), [149](#)
- phf
  - [v2x\\_msg\\_bsm](#), [156](#)

- plcp\_length
  - v2x\_net\_wme.h, [246](#)
  - v2x\_wsmp\_rcv\_indication, [170](#)
- pos3d
  - J2735TIMGeoPath, [110](#)
- pos\_accuracy
  - v2x\_msg\_psm, [161](#)
- pos\_confidence
  - v2x\_msg\_bsm, [156](#)
- position
  - J2735CircularRegion, [103](#)
  - J2735RegionPointSet, [104](#)
  - J2735RoadSignID, [105](#)
  - J2735ShapePointSet, [106](#)
- position\_confidence, [121](#)
- positional\_accuracy
  - path\_history, [118](#)
  - pathhistoryframe, [119](#)
- positional\_accuracy\_data, [121](#)
- positional\_accuracy\_data\_2016, [122](#)
- positionalaccuracy
  - v2x\_msg\_bsm, [156](#)
- pp
  - \_J2735\_2016\_VehSafetyExtensions, [79](#)
- prefixlen
  - savari1609Wra, [123](#)
  - v2x\_net\_wme.h, [246](#)
- primarydns
  - savari1609Wra, [123](#)
  - v2x\_net\_wme.h, [246](#)
- prio\_resp
  - J2735\_2016\_SignalStatusPackageList, [100](#)
- priority
  - J2735TIM, [108](#)
  - v2x\_net\_wme.h, [246](#)
  - v2x\_wme\_reg\_req, [165](#)
  - v2x\_wsmp\_rcv\_indication, [170](#)
  - v2x\_wsmp\_send\_req, [172](#)
- probe\_seg\_num
  - \_J2735\_2016\_PVD, [76](#)
- prof\_type
  - security\_cmd\_profile\_msg, [125](#)
  - v2x\_security\_base.h, [251](#)
- projection\_type
  - J2735TIMGeoPath, [110](#)
- psc
  - v2x\_net\_wme.h, [246](#)
  - v2x\_wme\_reg\_req, [165](#)
- psc\_length
  - v2x\_net\_wme.h, [247](#)
  - v2x\_wme\_reg\_req, [165](#)
- psid
  - v2x\_net\_wme.h, [247](#)
  - v2x\_wme\_reg\_req, [166](#)
  - v2x\_wsmp\_rcv\_indication, [170](#)
  - v2x\_wsmp\_send\_req, [172](#)
- RADIUS\_TYPE\_CENTIMETER
  - v2x\_msg\_tim.h, [232](#)
- RADIUS\_TYPE\_CM\_2\_5
  - v2x\_msg\_tim.h, [232](#)
- RADIUS\_TYPE\_DECIMETER
  - v2x\_msg\_tim.h, [232](#)
- RADIUS\_TYPE\_FOOT
  - v2x\_msg\_tim.h, [232](#)
- RADIUS\_TYPE\_KILOMETER
  - v2x\_msg\_tim.h, [232](#)
- RADIUS\_TYPE\_METER
  - v2x\_msg\_tim.h, [232](#)
- RADIUS\_TYPE\_MILE
  - v2x\_msg\_tim.h, [232](#)
- RADIUS\_TYPE\_YARD
  - v2x\_msg\_tim.h, [232](#)
- radius
  - v2x\_msg\_bsm, [156](#)
  - v2x\_msg\_psm, [161](#)
- radius\_of\_curvature
  - \_J2735\_2016\_PathPrediction, [73](#)
- radius\_type
  - J2735CircularRegion, [103](#)
- ref\_denms
  - v2x\_etsi\_denm\_rwc, [142](#)
- ref\_pt
  - intersection\_geodescr, [92](#)
  - v2x\_etsi\_map\_isec, [146](#)
- reference\_time
  - v2x\_etsi\_denm\_mc, [140](#)
- reg\_offsets
  - J2735RegionPointSet, [104](#)
- region
  - v2x\_etsi\_map\_isec, [146](#)
- region\_flag
  - J2735ValidRegion, [111](#)
- region\_id
  - RoadSegmentRefId, [123](#)
- region\_offset\_len
  - J2735RegionPointSet, [104](#)
- regionpoint
  - J2735ValidRegion, [111](#)
- relevance\_distance
  - v2x\_etsi\_denm\_mc, [140](#)
- relevance\_traf\_dir
  - v2x\_etsi\_denm\_mc, [140](#)
- repeatrate
  - v2x\_net\_wme.h, [247](#)
  - v2x\_wme\_reg\_req, [166](#)
- req\_pos\_vector
  - srm\_requestor, [127](#)
- req\_type
  - J2735\_2016\_SignalRequest\_t, [98](#)
- request\_id
  - J2735\_2016\_SignalRequest, [97](#)
  - J2735\_2016\_SignalRequest\_t, [98](#)
- request\_type
  - v2x\_net\_wme.h, [247](#)
  - v2x\_wme\_reg\_req, [166](#)
- requestor

- j2735\_2016\_srm, 101
- requestor\_type
  - J2735\_2016\_SignalRequest, 97
  - srm\_requestor, 127
- requestor\_type\_present
  - J2735\_2016\_SignalRequest, 97
- requestor\_type\_valid
  - srm\_requestor, 127
- resp\_type
  - v2x\_msg\_eva, 159
- restricted\_types
  - v2x\_etsi\_denm\_rwc, 142
- rightturnsignal
  - v2x\_msg\_bsm, 156
- road\_segment, 122
- road\_sign\_id
  - J2735TIM, 108
- road\_type
  - v2x\_etsi\_denm\_lc, 139
- RoadSegmentRefId, 122
  - region\_id, 123
  - roadsegment\_id, 123
- RoadSegmentRefId\_t
  - v2x\_msg\_tim.h, 231
- roadsegment\_id
  - RoadSegmentRefId, 123
- role
  - v2x\_etsi\_cam\_lf, 137
- rsa\_msg
  - v2x\_msg\_eva, 159
- rsi
  - v2x\_net\_wme.h, 247
  - v2x\_wsmp\_rcv\_indication, 170
- rsu
  - v2x\_etsi\_cam\_hf, 133
- rwc
  - v2x\_etsi\_denm\_ac, 138
- rx\_buf
  - v2x\_net\_wme.h, 247
  - v2x\_wsmp\_rcv\_indication, 170
- rx\_buf\_length
  - v2x\_net\_wme.h, 247
  - v2x\_wsmp\_rcv\_indication, 170
- rx\_mac
  - v2x\_net\_wme.h, 247
  - v2x\_wsmp\_rcv\_indication, 170
- rx\_supp
  - v2x\_net\_wme.h, 247
  - v2x\_wsmp\_rcv\_indication, 170
- SECURITY\_SIGNATURE\_PROFILE
  - The V2X Security module API, 60
- SECURITY\_VERIFICATION\_PROFILE
  - The V2X Security module API, 60
- SRM\_PRIO\_REQ\_TYPE\_CANCEL
  - v2x\_msg\_srm.h, 221
- SRM\_PRIO\_REQ\_TYPE\_REQ
  - v2x\_msg\_srm.h, 221
- SRM\_PRIO\_REQ\_TYPE\_RESERVED
  - v2x\_msg\_srm.h, 221
- SRM\_PRIO\_REQ\_TYPE\_UPDATE
  - v2x\_msg\_srm.h, 221
- SSI\_LEN
  - v2x\_net\_wme.h, 241
- SUCCESS
  - v2x\_net\_wme.h, 241
  - v2x\_sensor\_gnss.h, 253
- safetysupp
  - v2x\_net\_wme.h, 248
  - v2x\_wsmp\_send\_req, 172
- savari1609Wra, 123
  - default\_gw, 123
  - gw\_macaddr, 123
  - ipv6addr, 123
  - lifetime, 123
  - prefixlen, 123
  - primarydns, 123
  - secondarydns, 123
- savari1609WsaService, 124
- savari\_antennaoffset, 124
- savari\_rtcn\_payload, 125
- savariwme\_cbs
  - v2x\_net\_wme.h, 241
- savariwme\_reg\_req
  - v2x\_net\_wme.h, 241
- savariwme\_rx\_indication
  - v2x\_net\_wme.h, 241
- savariwme\_tx\_req
  - v2x\_net\_wme.h, 241
- sc
  - v2x\_etsi\_denm, 138
- secmark
  - v2x\_msg\_bsm, 156
  - v2x\_msg\_psm, 161
- second
  - J2735\_2016\_SignalStatusPackageList, 100
  - srm\_list, 126
  - utcTimeFrame, 129
- secondarydns
  - savari1609Wra, 123
  - v2x\_net\_wme.h, 248
- secondradio
  - v2x\_net\_wme.h, 248
  - v2x\_wme\_reg\_req, 166
- security\_cmd\_profile\_msg, 125
  - prof\_type, 125
  - sign\_profile, 125
  - verify\_profile, 125
- service\_ipv6addr
  - v2x\_net\_wme.h, 248
  - v2x\_wme\_reg\_req, 166
- service\_port
  - v2x\_net\_wme.h, 248
  - v2x\_wme\_reg\_req, 166
- shape\_points
  - J2735ValidRegion, 111
- sharedlane

- laneattr, 112
- sign\_profile
  - security\_cmd\_profile\_msg, 125
  - v2x\_security\_base.h, 251
- signal\_group\_id
  - \_J2735\_2016\_MovementState\_t, 73
  - connects\_to, 83
  - etsi\_connects, 88
  - laneset, 114
  - v2x\_etsi\_lane, 144
  - v2x\_etsi\_spat\_mm\_state, 150
- signal\_request
  - J2735\_2016\_SignalStatusPackageList, 100
  - srmlist, 126
- siren\_in\_use
  - v2x\_emergency\_details, 131
- snapshot
  - \_J2735\_2016\_PVD, 76
- snapshot\_len
  - \_J2735\_2016\_PVD, 76
- spat\_msg
  - v2x\_etsi\_spat, 147
- spat\_msg\_subid
  - v2x\_etsi\_spat\_msg, 150
- spedelimits, 125
- speed
  - pf\_gnss\_data, 121
  - v2x\_etsi\_cam\_hf\_vehicle, 135
  - v2x\_msg\_bsm, 156
  - v2x\_msg\_psm, 161
- speed\_conf
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- speed\_confidence
  - v2x\_msg\_bsm, 156
- speed\_limit
  - v2x\_etsi\_denm\_rwc, 142
- speedlimit\_codes
  - J2735TIM, 108
- srcmac
  - v2x\_net\_wme.h, 248
  - v2x\_wsmp\_send\_req, 172
- srcmacaddr
  - v2x\_net\_wme.h, 248
  - v2x\_wme\_reg\_req, 166
- srmlist, 126
  - duration, 126
  - min\_of\_year, 126
  - second, 126
  - signal\_request, 126
- srmlist\_count
  - j2735\_2016\_srm, 101
- srm\_requestor, 126
  - full\_pos\_valid, 127
  - id, 127
  - req\_pos\_vector, 127
  - requestor\_type, 127
  - requestor\_type\_valid, 127
- srmlist
  - j2735\_2016\_srm, 101
- ss\_pkg\_list
  - J2735\_2016\_SignalStatusList, 99
- ss\_pkg\_list\_len
  - J2735\_2016\_SignalStatusList, 99
- sslist
  - j2735\_2016\_ssm, 101
- sslist\_len
  - j2735\_2016\_ssm, 102
- ssp\_loc\_rights
  - J2735TIM, 108
- ssp\_msg\_rights1
  - J2735TIM, 108
- ssp\_msg\_rights2
  - J2735TIM, 108
- ssp\_rights
  - v2x\_emergency\_details, 131
- ssp\_tim\_rights
  - J2735TIM, 108
- stabilitycontrol
  - v2x\_msg\_bsm, 156
- start\_minute\_of\_year
  - J2735TIM, 108
- start\_time
  - \_J2735\_2016\_TimeChange\_t, 77
  - v2x\_etsi\_spat\_mm\_evt, 149
- start\_year
  - J2735TIM, 108
- starting\_point\_splimit
  - v2x\_etsi\_denm\_rwc, 142
- station\_position
  - v2x\_etsi\_cam\_bc, 132
- station\_type
  - v2x\_etsi\_cam\_bc, 132
  - v2x\_etsi\_denm\_mc, 140
- steeringwhl\_angle
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- steeringwhl\_angle\_conf
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- supp\_enable
  - v2x\_net\_wme.h, 248
  - v2x\_wsmp\_send\_req, 172
- tdf
  - v2x\_msg\_bsm, 156
- temp\_id
  - v2x\_msg\_bsm, 156
  - v2x\_msg\_psm, 161
- tempid
  - v2x\_msg\_eva, 159
- termination
  - v2x\_etsi\_denm\_mc, 140
- testcheck
  - testdataframe, 128
- testconfiguration
  - testdataframe, 128
- testdataframe, 127
  - channelmode, 128
  - datarate, 128

- formatversion, [128](#)
- msgrate, [128](#)
- msgseqnum, [128](#)
- numobe, [128](#)
- obeid, [128](#)
- testcheck, [128](#)
- testconfiguration, [128](#)
- The V2X Common API
  - V2X\_EACCES, [14](#)
  - V2X\_ECHANNEL, [14](#)
  - V2X\_ECONN\_REFUSED, [14](#)
  - V2X\_EDECODE, [13](#)
  - V2X\_EENCODE, [13](#)
  - V2X\_EFAULT, [13](#)
  - V2X\_EINVAL, [13](#)
  - V2X\_EIO, [14](#)
  - V2X\_ELENGTH, [14](#)
  - V2X\_ENOMEM, [13](#)
  - V2X\_PEER\_CLOSE, [14](#)
  - V2X\_SUCCESS, [13](#)
- The V2X Message API (US)
  - J2735\_2016\_BSM\_MSG\_D, [26](#)
  - J2735\_2016\_BSM\_MSG\_U, [26](#)
  - J2735\_2016\_CRW\_AUDIO\_SUPPORT, [27](#)
  - J2735\_2016\_CRW\_BICYCLE\_USE\_ONLY, [26](#)
  - J2735\_2016\_CRW\_BIDIRECTIONAL\_CYCLE\_TIME, [26](#)
  - J2735\_2016\_CRW\_CRW\_REVOCABLE\_LANE, [26](#)
  - J2735\_2016\_CRW\_FIXED\_CYCLE\_TIME, [26](#)
  - J2735\_2016\_CRW\_HAS\_PUSH\_TO\_WALK\_BUTTON, [26](#)
  - J2735\_2016\_CRW\_IS\_FLYOVER\_CRW\_LANE, [26](#)
  - J2735\_2016\_CRW\_RFSIGNAL\_REQ\_PRESENT, [27](#)
  - J2735\_2016\_CRW\_UNSIGNALIZED\_SEGMENT\_S\_PRESENT, [27](#)
  - J2735\_2016\_DIROFUSE\_BOTH, [23](#)
  - J2735\_2016\_DIROFUSE\_FORWARD, [23](#)
  - J2735\_2016\_DIROFUSE\_REVERSE, [23](#)
  - J2735\_2016\_EVT\_ABS\_ACTIVATED, [25](#)
  - J2735\_2016\_EVT\_AIRBAG\_DEPLOYMENT, [25](#)
  - J2735\_2016\_EVT\_DISABLED\_VEHICLE, [25](#)
  - J2735\_2016\_EVT\_FLATTIRE, [25](#)
  - J2735\_2016\_EVT\_HARDBRAKING, [25](#)
  - J2735\_2016\_EVT\_HAZARD\_LIGHTS, [25](#)
  - J2735\_2016\_EVT\_HAZARDOUS\_MATERIAL, [25](#)
  - J2735\_2016\_EVT\_LIGHTS\_CHANGED, [25](#)
  - J2735\_2016\_EVT\_RESERVED1, [25](#)
  - J2735\_2016\_EVT\_STABILITY\_CONTROL, [25](#)
  - J2735\_2016\_EVT\_STOPLINE\_VIOLATION, [25](#)
  - J2735\_2016\_EVT\_TRACTION\_CONTROL, [25](#)
  - J2735\_2016\_EVT\_WIPERS\_CHANGED, [25](#)
  - J2735\_2016\_EXT\_AUTOMATIC\_LIGHT\_CONTROL\_ON, [23](#)
  - J2735\_2016\_EXT\_DAYTIMERUNLIGHTS\_ON, [23](#)
  - J2735\_2016\_EXT\_FOGLIGHTS\_ON, [23](#)
  - J2735\_2016\_EXT\_HAZARDSIGNAL\_ON, [23](#)
  - J2735\_2016\_EXT\_HIGHBEAM\_ON, [23](#)
  - J2735\_2016\_EXT\_LEFTTURN\_ON, [23](#)
  - J2735\_2016\_EXT\_LOWBEAM\_ON, [23](#)
  - J2735\_2016\_EXT\_PARKINGLIGHTS\_ON, [23](#)
  - J2735\_2016\_EXT\_RIGHTTURN\_ON, [23](#)
  - J2735\_2016\_HS\_FROM00\_0\_TO\_22\_5\_DEGREES, [23](#)
  - J2735\_2016\_HS\_FROM112\_5\_TO\_135\_0\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM135\_0\_TO\_157\_5\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM157\_5\_TO\_180\_0\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM180\_0\_TO\_202\_5\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM202\_5\_TO\_225\_0\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM225\_0\_TO\_247\_5\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM22\_5\_TO\_45\_0\_DEGREES, [23](#)
  - J2735\_2016\_HS\_FROM247\_5\_TO\_270\_0\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM270\_0\_TO\_292\_5\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM292\_5\_TO\_315\_0\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM315\_0\_TO\_337\_5\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM337\_5\_TO\_360\_0\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM45\_0\_TO\_67\_5\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM67\_5\_TO\_90\_0\_DEGREES, [24](#)
  - J2735\_2016\_HS\_FROM90\_0\_TO\_112\_5\_DEGREES, [24](#)
  - J2735\_2016\_LANE\_DIRECTION\_EGRESS, [26](#)
  - J2735\_2016\_LANE\_DIRECTION\_INGRESS, [26](#)
  - J2735\_2016\_LANE\_DIRECTION\_UNAVAIL, [26](#)
  - J2735\_2016\_MAN\_CAUTION, [23](#)
  - J2735\_2016\_MAN\_GOWITH\_HALT, [23](#)
  - J2735\_2016\_MAN\_LANE\_CHANGE\_ALLOWED, [23](#)
  - J2735\_2016\_MAN\_LEFT\_ALLOWED, [22](#)
  - J2735\_2016\_MAN\_LEFT\_TURN\_ON\_RED\_ALLOWED, [23](#)
  - J2735\_2016\_MAN\_NO\_STOPPING\_ALLOWED, [23](#)
  - J2735\_2016\_MAN\_RIGHT\_ALLOWED, [22](#)
  - J2735\_2016\_MAN\_RIGHT\_TURN\_ON\_RED\_ALLOWED, [23](#)
  - J2735\_2016\_MAN\_STRAIGHT\_ALLOWED, [22](#)
  - J2735\_2016\_MAN\_UTURN\_ALLOWED, [23](#)
  - J2735\_2016\_MAN\_YIELD\_ALLWAYS\_REQUIRED, [23](#)
  - J2735\_2016\_MAP\_MSG\_P, [26](#)

- J2735\_2016\_NODE\_TYPE\_COMP, [25](#)
- J2735\_2016\_NODE\_TYPE\_NODE\_OBJS, [25](#)
- J2735\_2016\_NODE\_TYPE\_NOT\_PRESENT, [25](#)
- J2735\_2016\_NODEATTR\_DISABLED\_SEGMENTS, [24](#)
- J2735\_2016\_NODEATTR\_ENABLED\_SEGMENTS, [24](#)
- J2735\_2016\_NODEATTR\_LANEDATA, [24](#)
- J2735\_2016\_NODEATTR\_NODEATTR, [24](#)
- J2735\_2016\_NODEATTR\_UNAVAIL, [24](#)
- J2735\_2016\_RTCM\_CORRECTIONS\_MSG\_D, [26](#)
- J2735\_2016\_RTCM\_CORRECTIONS\_MSG\_U, [26](#)
- J2735\_2016\_SPaT\_MSG\_P, [26](#)
- J2735\_2016\_SRM\_MSG\_U, [26](#)
- J2735\_2016\_SSM\_MSG\_U, [26](#)
- J2735\_2016\_TIM\_MSG\_D, [26](#)
- J2735\_2016\_TIM\_MSG\_P, [26](#)
- J2735\_2016\_TIM\_MSG\_U, [26](#)
- J2735\_2016\_VEH\_HOV\_USE\_ONLY, [27](#)
- J2735\_2016\_VEH\_IS\_VEHICLE\_FLYOVER\_LANE, [27](#)
- J2735\_2016\_VEH\_IS\_VEHICLE\_REVOCABLE\_LANE, [27](#)
- J2735\_2016\_VEH\_RESTRICTED\_FROM\_PUBLIC\_USE, [27](#)
- J2735\_2016\_VEH\_RESTRICTED\_TO\_BUS\_USE, [27](#)
- J2735\_2016\_VEH\_RESTRICTED\_TO\_TAXI\_USE, [27](#)
- J2735\_2016\_WIPER\_AUTOMATIC\_PRESENT, [25](#)
- J2735\_2016\_WIPER\_HIGH, [25](#)
- J2735\_2016\_WIPER\_INTERMITTENT, [25](#)
- J2735\_2016\_WIPER\_LOW, [25](#)
- J2735\_2016\_WIPER\_STATUS\_OFF, [25](#)
- J2735\_2016\_WIPER\_STATUS\_UNAVAIL, [25](#)
- J2735\_2016\_WIPER\_WASHER\_IN\_USE, [25](#)
- The V2X Network and Transport module API (US)
  - LIBWME\_ACTION\_ADD, [54](#)
  - LIBWME\_ACTION\_CHANGE, [54](#)
  - LIBWME\_ACTION\_DELETE, [54](#)
  - LIBWME\_RC\_ACCEPTED, [54](#)
  - LIBWME\_RC\_INVALID\_PARAMETERS, [54](#)
  - LIBWME\_RC\_UNSPECIFIED, [54](#)
  - LIBWME\_USER\_AUTOACCESS\_ONMATCH, [53](#)
  - LIBWME\_USER\_AUTOACCESS\_UNCOND, [53](#)
  - LIBWME\_USER\_NOSCHACCESS, [53](#)
  - P1609\_CHANNEL\_ACCESS\_ALTERNATING, [54](#)
  - P1609\_CHANNEL\_ACCESS\_CONTINUOUS, [54](#)
  - P1609\_CHANNEL\_ACCESS\_NOSCHACCESS, [54](#)
  - P1609\_CHANNEL\_ACCESS\_ONMATCH, [54](#)
  - P1609\_CHANNEL\_ACCESS\_UNKNOWN, [54](#)
  - P1609\_WSA\_SECURED, [54](#)
  - P1609\_WSA\_UNSECURED, [54](#)
- The V2X Security module API
  - SECURITY\_SIGNATURE\_PROFILE, [60](#)
  - SECURITY\_VERIFICATION\_PROFILE, [60](#)
- The V2X Common API, [13](#)
  - v2x\_debug\_level\_set, [14](#)
  - v2x\_log, [14](#)
  - v2x\_status\_t, [13](#)
  - v2x\_strerror, [14](#)
- The V2X Message API (EU), [40](#)
  - v2x\_etsi\_cam\_decode, [40](#)
  - v2x\_etsi\_cam\_encode, [41](#)
  - v2x\_etsi\_cam\_print, [41](#)
  - v2x\_etsi\_cam\_reset, [41](#)
  - v2x\_etsi\_denm\_decode, [42](#)
  - v2x\_etsi\_denm\_dumphex, [42](#)
  - v2x\_etsi\_denm\_encode, [42](#)
  - v2x\_etsi\_map\_decode, [43](#)
  - v2x\_etsi\_map\_encode, [43](#)
  - v2x\_etsi\_map\_free, [44](#)
  - v2x\_etsi\_map\_print, [44](#)
  - v2x\_etsi\_map\_reset, [44](#)
  - v2x\_etsi\_print\_denm, [46](#)
  - v2x\_etsi\_reset\_denm, [46](#)
  - v2x\_etsi\_reset\_spat, [46](#)
  - v2x\_etsi\_spat\_decode, [47](#)
  - v2x\_etsi\_spat\_encode, [47](#)
  - v2x\_etsi\_spat\_print, [48](#)
- The V2X Message API (US), [16](#)
  - J2735\_2016\_AllowedManeuvers\_t, [22](#)
  - J2735\_2016\_Connectsto\_Lanes\_t, [20](#)
  - J2735\_2016\_DSRC\_MsgID\_t, [20](#)
  - J2735\_2016\_DirofUse\_t, [23](#)
  - J2735\_2016\_ExteriorLights\_t, [23](#)
  - J2735\_2016\_HeadingSlice, [23](#)
  - J2735\_2016\_HeadingSlice\_t, [20](#)
  - J2735\_2016\_LaneAttribute\_t, [20](#)
  - J2735\_2016\_LaneDir\_t, [20](#)
  - J2735\_2016\_LaneList\_t, [20](#)
  - J2735\_2016\_LaneTypeAttrCrossWk\_t, [20](#)
  - J2735\_2016\_LaneTypeAttrVeh\_t, [20](#)
  - J2735\_2016\_NodeAttr\_t, [20](#)
  - J2735\_2016\_NodeAttrType\_t, [24](#)
  - J2735\_2016\_NodeList\_t, [20](#)
  - J2735\_2016\_NodePoint\_t, [20](#)
  - J2735\_2016\_NodeType\_t, [24](#)
  - J2735\_2016\_PathHistory\_t, [20](#)
  - J2735\_2016\_PathPrediction\_t, [21](#)
  - J2735\_2016\_VehicleEventFlags\_t, [25](#)
  - J2735\_2016\_WiperSet\_t, [21](#)
  - J2735\_2016\_WiperStatus\_t, [25](#)
  - j2735\_dsrc\_msgids, [25](#)
  - J2735Extent, [26](#)
  - J2735Extent\_t, [21](#)
  - J2735Position3D\_t, [21](#)
  - lanedirection, [26](#)
  - lanetypecrosswalk, [26](#)
  - lanetypevehicle, [27](#)
  - v2x\_map\_free, [27](#)
  - v2x\_msg\_bsm\_decode, [27](#)



- v2x\_msg\_bsm\_encode, 28
- v2x\_msg\_bsm\_init, 28
- v2x\_msg\_bsm\_t, 21
- v2x\_msg\_eva\_decode, 29
- v2x\_msg\_eva\_encode, 29
- v2x\_msg\_eva\_print, 29
- v2x\_msg\_eva\_reset, 30
- v2x\_msg\_eva\_t, 21
- v2x\_msg\_map\_decode, 30
- v2x\_msg\_map\_encode, 30
- v2x\_msg\_map\_init, 31
- v2x\_msg\_map\_t, 21
- v2x\_msg\_psm\_decode, 31
- v2x\_msg\_psm\_encode, 31
- v2x\_msg\_psm\_reset, 32
- v2x\_msg\_rtc\_m\_decode, 32
- v2x\_msg\_rtc\_m\_encode, 33
- v2x\_msg\_rtc\_m\_print, 33
- v2x\_msg\_rtc\_m\_t, 21
- v2x\_msg\_spat\_decode, 33
- v2x\_msg\_spat\_encode, 34
- v2x\_msg\_spat\_t, 21
- v2x\_msg\_srm\_t, 21
- v2x\_msg\_ssm\_t, 21
- v2x\_msg\_tim\_decode, 34
- v2x\_msg\_tim\_encode, 35
- v2x\_msg\_tim\_free, 36
- v2x\_msg\_tim\_t, 21
- v2x\_pvd\_decode, 36
- v2x\_pvd\_encode, 36
- v2x\_pvd\_reset, 37
- v2x\_pvd\_snapshot\_reset, 37
- v2x\_spat\_free, 37
- v2x\_srm\_decode, 37
- v2x\_srm\_encode, 38
- v2x\_srm\_reset, 38
- v2x\_ssm\_decode, 38
- v2x\_ssm\_encode, 39
- v2x\_ssm\_reset, 39
- The V2X Network and Transport module API (EU), 49
  - v2x\_btp\_deinit, 49
  - v2x\_btp\_get\_sock\_id, 49
  - v2x\_btp\_init, 50
  - v2x\_btp\_receive, 50
  - v2x\_btp\_transmit, 51
- The V2X Network and Transport module API (US), 52
  - P1609\_WSATYPE, 58
  - v2x\_wme\_open, 54
  - v2x\_wme\_register\_user, 55
  - v2x\_wme\_unregister\_user, 55
  - v2x\_wme\_user\_service\_confirm, 56
  - v2x\_wsmp\_callbacks\_t, 53
  - v2x\_wsmp\_convert\_psid\_be, 56
  - v2x\_wsmp\_handler\_t, 53
  - v2x\_wsmp\_rcv, 57
  - v2x\_wsmp\_rcv\_indication\_t, 53
  - v2x\_wsmp\_send, 57
- The V2X Security module API, 59
  - v2x\_sec\_cert\_change\_register, 60
  - v2x\_sec\_cert\_change\_unregister, 60
  - v2x\_sec\_deinit, 61
  - v2x\_sec\_idchange\_lock, 61
  - v2x\_sec\_idchange\_unlock, 61
  - v2x\_sec\_init, 62
  - v2x\_sec\_rx, 62
  - v2x\_sec\_rx\_cblist\_t, 59
  - v2x\_sec\_sign\_request, 62
  - v2x\_sec\_verify\_request, 63
  - v2x\_security\_profile\_t, 59
  - v2x\_security\_profile\_type\_t, 60
  - v2x\_sign\_sec\_profile\_t, 59
  - v2x\_verify\_sec\_profile\_t, 59
- The V2X Sensors API, 64
  - pf\_gnss\_data\_t, 64
  - v2x\_can\_close, 65
  - v2x\_can\_open, 65
  - v2x\_can\_rcv, 65
  - v2x\_gnss\_close, 67
  - v2x\_gnss\_data\_t, 64
  - v2x\_gnss\_filters\_deinit, 67
  - v2x\_gnss\_filters\_init, 67
  - v2x\_gnss\_handle\_t, 64
  - v2x\_gnss\_open, 68
  - v2x\_gnss\_read, 68
  - v2x\_libpcan\_net\_deinit, 69
  - v2x\_libpcan\_net\_rcv, 69
  - v2x\_pf\_handle, 64
- throtl\_confidence
  - v2x\_msg\_bsm, 156
- throttle\_pos
  - v2x\_msg\_bsm, 157
- time
  - pf\_gnss\_data, 121
- time\_change\_info
  - \_J2735\_2016\_MovementEventList\_t, 72
- time\_confidence
  - v2x\_msg\_bsm, 157
- time\_offset
  - path\_history, 118
  - pathhistoryframe, 119
- timestamp, 128
- timestamp\_ms
  - intersection\_data, 91
- tollzone\_id
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- tollzone\_lat
  - v2x\_etsi\_cam\_hf\_vehicle, 135
- tollzone\_long
  - v2x\_etsi\_cam\_hf\_vehicle, 136
- traction
  - v2x\_msg\_bsm, 157
- traffic\_rule
  - v2x\_etsi\_denm\_rwc, 142
- transmissionstate
  - v2x\_msg\_bsm, 157
- transmit\_interval

- v2x\_etsi\_denm\_mc, 140
- tstamp
  - v2x\_net\_wme.h, 248
  - v2x\_wsmp\_rcv\_indication, 170
- tx\_length
  - v2x\_net\_wme.h, 248
  - v2x\_wsmp\_send\_req, 172
- txpower
  - v2x\_net\_wme.h, 248
  - v2x\_wsmp\_rcv\_indication, 171
  - v2x\_wsmp\_send\_req, 172
- type
  - J2735TIM, 108
  - nodeattrs, 116
  - v2x\_etsi\_cam, 132
  - v2x\_etsi\_cam\_hf, 133
  - v2x\_etsi\_denm, 138
- unique\_id
  - v2x\_msg\_tim, 164
- use\_expiry\_time
  - aerolink\_sign\_secprofile, 82
  - v2x\_security\_base.h, 251
- use\_generation\_loc
  - aerolink\_sign\_secprofile, 82
  - v2x\_security\_base.h, 251
- use\_generation\_time
  - aerolink\_sign\_secprofile, 82
  - v2x\_security\_base.h, 252
- utc\_time
  - pf\_gnss\_data, 121
- utcTimeFrame, 129
  - day, 129
  - hour, 129
  - minute, 129
  - month, 129
  - second, 129
  - year, 129
- utime
  - v2x\_msg\_bsm, 157
- V2X\_EACCES
  - The V2X Common API, 14
- V2X\_ECHANNEL
  - The V2X Common API, 14
- V2X\_ECONN\_REFUSED
  - The V2X Common API, 14
- V2X\_EDECODE
  - The V2X Common API, 13
- V2X\_EENCODE
  - The V2X Common API, 13
- V2X\_EFAULT
  - The V2X Common API, 13
- V2X\_EINVAL
  - The V2X Common API, 13
- V2X\_EIO
  - The V2X Common API, 14
- V2X\_ELENGTH
  - The V2X Common API, 14
- V2X\_ENOMEM
  - The V2X Common API, 13
- V2X\_EVA\_RESPONSE\_EMERGENCY
  - v2x\_msg\_eva.h, 207
- V2X\_EVA\_RESPONSE\_NON\_EMERGENCY
  - v2x\_msg\_eva.h, 207
- V2X\_EVA\_RESPONSE\_NOT\_EQUIPPED
  - v2x\_msg\_eva.h, 207
- V2X\_EVA\_RESPONSE\_PURSUIT
  - v2x\_msg\_eva.h, 207
- V2X\_EVA\_RESPONSE\_SLOW\_MOVING
  - v2x\_msg\_eva.h, 207
- V2X\_EVA\_RESPONSE\_STATIONARY
  - v2x\_msg\_eva.h, 207
- V2X\_EVA\_RESPONSE\_STOP\_AND\_GO
  - v2x\_msg\_eva.h, 207
- V2X\_PEER\_CLOSE
  - The V2X Common API, 14
- V2X\_SUCCESS
  - The V2X Common API, 13
- V2X\_ETSI\_SET\_BC
  - v2x\_msg\_cam.h, 182
- V2XEvaEmergencyDetails\_t
  - v2x\_msg\_eva.h, 206
- v2x\_msg\_bsm.h
  - BSM\_ABS\_ACTIVE\_ENGAGED, 180
  - BSM\_ABS\_ACTIVE\_OFF, 179
  - BSM\_ABS\_ACTIVE\_ON, 180
  - BSM\_ABS\_ACTIVE\_UNAVAIL, 179
  - BSM\_PART1, 179
  - BSM\_PATHHISTORY, 179
  - BSM\_PATHPREDICTION, 179
  - BSM\_SCS\_ENGAGED, 180
  - BSM\_SCS\_OFF, 180
  - BSM\_SCS\_ON, 180
  - BSM\_SCS\_UNAVAIL, 180
  - BSM\_TDF\_UNENCODED, 179
  - BSM\_TESTDATAFRAME, 179
  - BSM\_TRC\_ENGAGED, 180
  - BSM\_TRC\_OFF, 180
  - BSM\_TRC\_ON, 180
  - BSM\_TRC\_UNAVAIL, 180
  - BSM\_VEHICLESTATUS, 179
  - BSM\_WIPERS\_AUTOMATIC\_PRESENT, 180
  - BSM\_WIPERS\_HIGH, 180
  - BSM\_WIPERS\_INTERMITTENT, 180
  - BSM\_WIPERS\_LOW, 180
  - BSM\_WIPERS\_OFF, 180
  - BSM\_WIPERS\_UNAVAIL, 180
  - BSM\_WIPERS\_WASHER\_IN\_USE, 180
- v2x\_msg\_common.h
  - BSM\_PART\_II\_ID\_TYPE\_SPECIAL\_VEH\_EXT, 197
  - BSM\_PART\_II\_ID\_TYPE\_SUPPLEMENTARY\_VEH\_EXT, 197
  - BSM\_PART\_II\_ID\_TYPE\_VEH\_SAFETY\_EXT, 197
  - J2735\_2016\_BUS\_VEHICLE\_TRAFFIC, 198



- J2735\_2016\_CYCLIST\_TRAFFIC, [198](#)
- J2735\_2016\_INDIVIDUAL\_MOTOR\_TRAFFIC, [198](#)
- J2735\_2016\_MULTI\_LANE\_TREATED\_AS\_ONE, [198](#)
- J2735\_2016\_OTHER\_NON\_MOTORIZED, [198](#)
- J2735\_2016\_OVERLAP\_LANE\_DESCR, [198](#)
- J2735\_2016\_PED\_TRAFFIC, [198](#)
- J2735\_2016\_TAXI\_VEHICLE\_TRAFFIC, [198](#)
- J2735\_2016\_TRACKED\_VEHICLE\_TRAFFIC, [198](#)
- v2x\_msg\_eva.h
  - V2X\_EVA\_RESPONSE\_EMERGENCY, [207](#)
  - V2X\_EVA\_RESPONSE\_NON\_EMERGENCY, [207](#)
  - V2X\_EVA\_RESPONSE\_NOT\_EQUIPPED, [207](#)
  - V2X\_EVA\_RESPONSE\_PURSUIT, [207](#)
  - V2X\_EVA\_RESPONSE\_SLOW\_MOVING, [207](#)
  - V2X\_EVA\_RESPONSE\_STATIONARY, [207](#)
  - V2X\_EVA\_RESPONSE\_STOP\_AND\_GO, [207](#)
- v2x\_msg\_psm.h
  - PSM\_DEVICE\_USAGE\_STATE\_CALLING, [210](#)
  - PSM\_DEVICE\_USAGE\_STATE\_IDLE, [210](#)
  - PSM\_DEVICE\_USAGE\_STATE\_LISTENING\_TO\_AUDIO, [210](#)
  - PSM\_DEVICE\_USAGE\_STATE\_OTHER, [210](#)
  - PSM\_DEVICE\_USAGE\_STATE\_PLAYING\_GAMES, [210](#)
  - PSM\_DEVICE\_USAGE\_STATE\_READING, [210](#)
  - PSM\_DEVICE\_USAGE\_STATE\_TYPING, [210](#)
  - PSM\_DEVICE\_USAGE\_STATE\_UNAVAIL, [210](#)
  - PSM\_DEVICE\_USAGE\_STATE\_VIEWING, [210](#)
  - PSM\_DEVICE\_USER\_TYPE\_ANIMAL, [210](#)
  - PSM\_DEVICE\_USER\_TYPE\_PEDALCYCLIST, [210](#)
  - PSM\_DEVICE\_USER\_TYPE\_PEDESTRIAN, [210](#)
  - PSM\_DEVICE\_USER\_TYPE\_PUBLICSAFETYWORKER, [210](#)
  - PSM\_DEVICE\_USER\_TYPE\_UNAVAILABLE, [210](#)
- v2x\_msg\_pvd.h
  - J2735\_GPS\_STATUS\_BASE\_STATION\_TYPE, [212](#)
  - J2735\_GPS\_STATUS\_INVIEW\_OF\_UNDER\_5, [212](#)
  - J2735\_GPS\_STATUS\_IS\_HEALTHY, [212](#)
  - J2735\_GPS\_STATUS\_IS\_MONITORED, [212](#)
  - J2735\_GPS\_STATUS\_LOCAL\_CORRECTIONS\_PRESENT, [212](#)
  - J2735\_GPS\_STATUS\_NETWORK\_CORRECTIONS\_PRESENT, [212](#)
  - J2735\_GPS\_STATUS\_PDOP\_UNDER\_5, [212](#)
  - J2735\_GPS\_STATUS\_UNAVAIL, [212](#)
- v2x\_msg\_spat.h
  - J2735\_2016\_ISEC\_STATUS\_FAILURE\_FLASH, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_FAILURE\_MODE, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_FIXED\_TIME\_OPERATION, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_MANUAL\_CONTROL\_IS\_ON, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_NOVALID\_MAP\_IS\_AVAILABLE, [218](#)
  - J2735\_2016\_ISEC\_STATUS\_NOVALID\_SPAT\_IS\_AVAILABLE, [218](#)
  - J2735\_2016\_ISEC\_STATUS\_OFF, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_PREEMPT\_IS\_ACTIVE, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_RECENT\_CHANGE\_IN\_MAP\_ASSIGNED\_LANEIDS\_USED, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_RECENT\_MAP\_MESSAGE\_UPDATE, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_STAND\_BY\_OPERATION, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_STOP\_TIME\_IS\_ACTIVE, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_TRAFFIC\_DEPENDENT\_OPERATION, [217](#)
  - J2735\_2016\_ISEC\_STATUS\_TSP\_IS\_ACTIVE, [217](#)
- v2x\_msg\_srm.h
  - SRM\_PRIO\_REQ\_TYPE\_CANCEL, [221](#)
  - SRM\_PRIO\_REQ\_TYPE\_REQ, [221](#)
  - SRM\_PRIO\_REQ\_TYPE\_RESERVED, [221](#)
  - SRM\_PRIO\_REQ\_TYPE\_UPDATE, [221](#)
- v2x\_msg\_tim.h
  - J2735\_2016\_COMPUTED\_LANES, [231](#)
  - J2735\_2016\_NODESET\_XY, [231](#)
  - J2735\_2016\_OFFSET\_TYPE\_LL, [231](#)
  - J2735\_2016\_OFFSET\_TYPE\_XY, [231](#)
  - J2735\_2016\_PROJECTION\_GEO\_PROJECTION, [231](#)
  - J2735\_2016\_PROJECTION\_OFFSET\_SYSTEM, [231](#)
  - J2735\_2016\_PROJECTION\_VALID\_REGION, [231](#)
  - J2735\_CONTENT\_ADVISORY, [231](#)
  - J2735\_CONTENT\_EXIT\_SERVICES, [231](#)
  - J2735\_CONTENT\_GENERIC\_SIGNS, [231](#)
  - J2735\_CONTENT\_SPEED\_LIMITS, [231](#)
  - J2735\_CONTENT\_WORKZONES, [231](#)
  - J2735\_DIRECTION\_BOTH, [232](#)
  - J2735\_DIRECTION\_FORWARD, [232](#)
  - J2735\_DIRECTION\_REVERSE, [232](#)
  - J2735\_DIRECTION\_UNAVAIL, [232](#)
  - J2735\_MUTCD\_CODE\_GUIDE, [232](#)
  - J2735\_MUTCD\_CODE\_MAINTENANCE, [232](#)
  - J2735\_MUTCD\_CODE\_MOTORIST\_SERVICE, [232](#)
  - J2735\_MUTCD\_CODE\_NONE, [232](#)
  - J2735\_MUTCD\_CODE\_REC, [232](#)
  - J2735\_MUTCD\_CODE\_REGULATORY, [232](#)
  - J2735\_MUTCD\_CODE\_WARNING, [232](#)
  - J2735\_REGION\_CIRCLE, [233](#)
  - J2735\_REGION\_NOTHING, [233](#)
  - J2735\_REGION\_POLYGON, [233](#)

- J2735\_REGION\_SHAPEPT, [233](#)
- J2735\_TIM\_TYPE\_ADVISORY, [233](#)
- J2735\_TIM\_TYPE\_COMMERCIAL\_SIGNAGE, [233](#)
- J2735\_TIM\_TYPE\_ROAD\_SIGNAGE, [233](#)
- J2735\_TIM\_TYPE\_UNKNOWN, [233](#)
- RADIUS\_TYPE\_CENTIMETER, [232](#)
- RADIUS\_TYPE\_CM\_2\_5, [232](#)
- RADIUS\_TYPE\_DECIMETER, [232](#)
- RADIUS\_TYPE\_FOOT, [232](#)
- RADIUS\_TYPE\_KILOMETER, [232](#)
- RADIUS\_TYPE\_METER, [232](#)
- RADIUS\_TYPE\_MILE, [232](#)
- RADIUS\_TYPE\_YARD, [232](#)
- v2x\_security\_base.h
  - AE\_SIGNER\_ID\_TYPE\_AUTO, [251](#)
  - AE\_SIGNER\_ID\_TYPE\_CERT, [251](#)
  - AE\_SIGNER\_ID\_TYPE\_DIGEST, [251](#)
- v2x\_btp\_deinit
  - The V2X Network and Transport module API (EU), [49](#)
- v2x\_btp\_get\_data
  - v2x\_net\_btp.h, [235](#)
- v2x\_btp\_get\_sock\_id
  - The V2X Network and Transport module API (EU), [49](#)
- v2x\_btp\_get\_txreq
  - v2x\_net\_btp.h, [235](#)
- v2x\_btp\_init
  - The V2X Network and Transport module API (EU), [50](#)
- v2x\_btp\_packet\_len
  - v2x\_net\_btp.h, [236](#)
- v2x\_btp\_receive
  - The V2X Network and Transport module API (EU), [50](#)
- v2x\_btp\_reserve
  - v2x\_net\_btp.h, [236](#)
- v2x\_btp\_reserve\_data
  - v2x\_net\_btp.h, [236](#)
- v2x\_btp\_transmit
  - The V2X Network and Transport module API (EU), [51](#)
- v2x\_can\_close
  - The V2X Sensors API, [65](#)
- v2x\_can\_open
  - The V2X Sensors API, [65](#)
- v2x\_can\_rcv
  - The V2X Sensors API, [65](#)
- v2x\_cancel, [130](#)
  - can\_fd, [130](#)
- v2x\_cancel.h, [175](#)
- v2x\_debug.h, [176](#)
- v2x\_debug\_level\_set
  - The V2X Common API, [14](#)
- v2x\_emergency\_details, [130](#)
  - lightbar\_in\_use, [131](#)
  - multi\_vehicle\_response, [131](#)
  - siren\_in\_use, [131](#)
  - ssp\_rights, [131](#)
- v2x\_error.h, [176](#)
- v2x\_etsi\_cam, [131](#)
  - bc, [131](#)
  - generation\_delta\_time, [131](#)
  - hf, [132](#)
  - lf, [132](#)
  - pdu, [132](#)
  - type, [132](#)
- v2x\_etsi\_cam\_bc, [132](#)
  - station\_position, [132](#)
  - station\_type, [132](#)
- v2x\_etsi\_cam\_decode
  - The V2X Message API (EU), [40](#)
- v2x\_etsi\_cam\_encode
  - The V2X Message API (EU), [41](#)
- v2x\_etsi\_cam\_hf, [132](#)
  - rsu, [133](#)
  - type, [133](#)
  - vehicle, [133](#)
- v2x\_etsi\_cam\_hf\_rsu, [133](#)
  - n\_zones, [133](#)
  - zonelist, [133](#)
- v2x\_etsi\_cam\_hf\_vehicle, [133](#)
  - accel\_ctrl, [134](#)
  - curvature, [134](#)
  - curvature\_calc, [134](#)
  - curvature\_conf, [134](#)
  - drive\_direction, [134](#)
  - heading, [134](#)
  - heading\_acc, [134](#)
  - lane\_position, [135](#)
  - lateral\_accel, [135](#)
  - lateral\_accel\_conf, [135](#)
  - longitude\_accel, [135](#)
  - longitude\_accel\_conf, [135](#)
  - performance\_class, [135](#)
  - speed, [135](#)
  - speed\_conf, [135](#)
  - steeringwhl\_angle, [135](#)
  - steeringwhl\_angle\_conf, [135](#)
  - tollzone\_id, [135](#)
  - tollzone\_lat, [135](#)
  - tollzone\_long, [136](#)
  - vehicle\_confidence, [136](#)
  - vehicle\_length, [136](#)
  - vehicle\_width, [136](#)
  - vertical\_accel, [136](#)
  - vertical\_accel\_conf, [136](#)
  - yawrate, [136](#)
  - yawrate\_conf, [136](#)
- v2x\_etsi\_cam\_lf, [136](#)
  - extLights, [137](#)
  - ph, [137](#)
  - role, [137](#)
- v2x\_etsi\_cam\_print
  - The V2X Message API (EU), [41](#)

- v2x\_etsi\_cam\_reset
  - The V2X Message API (EU), [41](#)
- v2x\_etsi\_denm, [137](#)
  - ac, [137](#)
  - lc, [137](#)
  - mc, [137](#)
  - pdu, [137](#)
  - sc, [138](#)
  - type, [138](#)
- v2x\_etsi\_denm\_ac, [138](#)
  - bit\_mask, [138](#)
  - rcw, [138](#)
- v2x\_etsi\_denm\_decode
  - The V2X Message API (EU), [42](#)
- v2x\_etsi\_denm\_dumphex
  - The V2X Message API (EU), [42](#)
- v2x\_etsi\_denm\_encode
  - The V2X Message API (EU), [42](#)
- v2x\_etsi\_denm\_lc, [138](#)
  - event\_position\_heading, [139](#)
  - event\_position\_heading\_conf, [139](#)
  - event\_speed, [139](#)
  - event\_speed\_conf, [139](#)
  - event\_traces, [139](#)
  - road\_type, [139](#)
- v2x\_etsi\_denm\_mc, [139](#)
  - action\_id, [140](#)
  - detection\_time, [140](#)
  - event\_position, [140](#)
  - reference\_time, [140](#)
  - relevance\_distance, [140](#)
  - relevance\_traf\_dir, [140](#)
  - station\_type, [140](#)
  - termination, [140](#)
  - transmit\_interval, [140](#)
  - validity\_duration, [141](#)
- v2x\_etsi\_denm\_rwc, [141](#)
  - bit\_mask, [141](#)
  - closed\_lanes, [141](#)
  - incident\_indication, [141](#)
  - itinerary\_path, [141](#)
  - lb\_siren, [141](#)
  - ref\_denms, [142](#)
  - restricted\_types, [142](#)
  - speed\_limit, [142](#)
  - starting\_point\_splimit, [142](#)
  - traffic\_rule, [142](#)
- v2x\_etsi\_denm\_sc, [142](#)
  - event\_history\_list, [142](#)
  - event\_type, [142](#)
  - information\_quality, [143](#)
  - linked\_cause, [143](#)
- v2x\_etsi\_lane, [143](#)
  - approach\_type, [143](#)
  - connects, [143](#)
  - connects\_len, [143](#)
  - egress\_approach, [143](#)
  - ingress\_approach, [143](#)
  - lane\_id, [144](#)
  - laneattr, [144](#)
  - maneuvers, [144](#)
  - nodes, [144](#)
  - nodes\_len, [144](#)
  - phase\_no, [144](#)
  - signal\_group\_id, [144](#)
- v2x\_etsi\_lane\_attr, [144](#)
- v2x\_etsi\_map, [145](#)
  - map\_msg, [145](#)
  - pdu, [145](#)
- v2x\_etsi\_map\_decode
  - The V2X Message API (EU), [43](#)
- v2x\_etsi\_map\_encode
  - The V2X Message API (EU), [43](#)
- v2x\_etsi\_map\_free
  - The V2X Message API (EU), [44](#)
- v2x\_etsi\_map\_isec, [145](#)
  - iid, [145](#)
  - laneset, [145](#)
  - laneset\_len, [145](#)
  - lanewidth, [146](#)
  - msg\_count, [146](#)
  - ref\_pt, [146](#)
  - region, [146](#)
- v2x\_etsi\_map\_msg, [146](#)
  - ilist, [146](#)
  - ilist\_len, [146](#)
  - msg\_count, [146](#)
  - msg\_subid, [146](#)
- v2x\_etsi\_map\_print
  - The V2X Message API (EU), [44](#)
- v2x\_etsi\_map\_reset
  - The V2X Message API (EU), [44](#)
- v2x\_etsi\_print\_denm
  - The V2X Message API (EU), [46](#)
- v2x\_etsi\_protected\_zone, [147](#)
- v2x\_etsi\_reset\_denm
  - The V2X Message API (EU), [46](#)
- v2x\_etsi\_reset\_spat
  - The V2X Message API (EU), [46](#)
- v2x\_etsi\_spat, [147](#)
  - pdu, [147](#)
  - spat\_msg, [147](#)
- v2x\_etsi\_spat\_decode
  - The V2X Message API (EU), [47](#)
- v2x\_etsi\_spat\_encode
  - The V2X Message API (EU), [47](#)
- v2x\_etsi\_spat\_isec, [148](#)
  - dsecond, [148](#)
  - enabled\_lanes, [148](#)
  - enabled\_lanes\_len, [148](#)
  - isec\_id, [148](#)
  - isec\_status, [148](#)
  - min\_of\_yr, [148](#)
  - mmllist, [148](#)
  - mmllist\_len, [148](#)
  - msg\_count, [148](#)

- v2x\_etsi\_spat\_mm\_evt, 149
  - confidence, 149
  - likely\_time, 149
  - max\_end\_time, 149
  - min\_end\_time, 149
  - next\_time, 149
  - phase\_state, 149
  - start\_time, 149
- v2x\_etsi\_spat\_mm\_state, 150
  - mm\_evt, 150
  - mm\_evt\_len, 150
  - signal\_group\_id, 150
- v2x\_etsi\_spat\_msg, 150
  - isec\_list, 150
  - isec\_list\_len, 150
  - spat\_msg\_subid, 150
- v2x\_etsi\_spat\_print
  - The V2X Message API (EU), 48
- v2x\_eva\_response\_type\_t
  - v2x\_msg\_eva.h, 207
- v2x\_gnss\_close
  - The V2X Sensors API, 67
- v2x\_gnss\_data\_t
  - The V2X Sensors API, 64
- v2x\_gnss\_filters\_deinit
  - The V2X Sensors API, 67
- v2x\_gnss\_filters\_init
  - The V2X Sensors API, 67
- v2x\_gnss\_handle\_t
  - The V2X Sensors API, 64
- v2x\_gnss\_open
  - The V2X Sensors API, 68
- v2x\_gnss\_read
  - The V2X Sensors API, 68
- v2x\_libpcan\_net\_deinit
  - The V2X Sensors API, 69
- v2x\_libpcan\_net\_recv
  - The V2X Sensors API, 69
- v2x\_log
  - The V2X Common API, 14
- v2x\_map\_free
  - The V2X Message API (US), 27
- v2x\_msg\_bsm, 151
  - abs, 152
  - angle, 152
  - autolightcontrol, 152
  - auxbrakes, 153
  - brakeboost, 153
  - bumperheight\_front, 153
  - bumperheight\_rear, 153
  - confidence, 153
  - count, 153
  - dtimerunlights, 153
  - elev\_confidence, 153
  - elevation, 153
  - emergency\_type\_event, 153
  - event\_absactivate, 153
  - event\_airbag, 153
  - event\_descr, 154
  - event\_descr\_len, 154
  - event\_hazardbraking, 154
  - event\_hazardlights, 154
  - event\_stabilityctrlactivated, 154
  - event\_tractionctrlloss, 154
  - foglights, 154
  - gpsstatus, 154
  - hazardlights, 154
  - head\_confidence, 154
  - heading, 154
  - highbeam, 154
  - lataccel, 155
  - latitude, 155
  - leftturnsignal, 155
  - length, 155
  - lightbar\_in\_use, 155
  - lightbarinuse, 155
  - longaccel, 155
  - longitude, 155
  - lowbeam, 155
  - msgcount, 155
  - msgid, 155
  - parkinglights, 155
  - phf, 156
  - pos\_confidence, 156
  - positionalaccuracy, 156
  - radius, 156
  - rightturnsignal, 156
  - secmark, 156
  - speed, 156
  - speed\_confidence, 156
  - stabilitycontrol, 156
  - tdf, 156
  - temp\_id, 156
  - throtl\_confidence, 156
  - throttle\_pos, 157
  - time\_confidence, 157
  - traction, 157
  - transmissionstate, 157
  - utime, 157
  - vehicleheight, 157
  - vehiclemass, 157
  - vehicletype, 157
  - vertaccel, 157
  - wheelbrake, 157
  - wheelbrakeavailable, 157
  - width, 157
  - wipers\_rtftnt, 158
  - wipers\_rtrear, 158
  - wipers\_swfnt, 158
  - wipers\_swrear, 158
  - yawrate, 158
- v2x\_msg\_bsm.h, 177
  - ASN1\_BSM\_MSG\_ID, 179
  - BSM\_TYPE, 179
  - BSMAbsActive\_t, 179
  - BSMBrakeAppliedStatus\_t, 180

- BSMStabilityControlStatus\_t, 180
- BSMTractionControl\_t, 180
- BSMWiperStatus\_t, 180
- v2x\_msg\_bsm\_decode
  - The V2X Message API (US), 27
- v2x\_msg\_bsm\_encode
  - The V2X Message API (US), 28
- v2x\_msg\_bsm\_init
  - The V2X Message API (US), 28
- v2x\_msg\_bsm\_t
  - The V2X Message API (US), 21
- v2x\_msg\_cam.h, 180
  - V2X\_ETSI\_SET\_BC, 182
- v2x\_msg\_common.h, 183
  - BSMPartII\_ID\_Types\_t, 197
  - laneshare, 197
- v2x\_msg\_denm.h, 198
- v2x\_msg\_etsi\_map.h, 200
- v2x\_msg\_etsi\_spat.h, 203
- v2x\_msg\_eva, 158
  - emergency\_details, 158
  - emergency\_details\_avail, 158
  - resp\_type, 159
  - rsa\_msg, 159
  - tempid, 159
- v2x\_msg\_eva.h, 206
  - V2XEvaEmergencyDetails\_t, 206
  - v2x\_eva\_response\_type\_t, 207
- v2x\_msg\_eva\_decode
  - The V2X Message API (US), 29
- v2x\_msg\_eva\_encode
  - The V2X Message API (US), 29
- v2x\_msg\_eva\_print
  - The V2X Message API (US), 29
- v2x\_msg\_eva\_reset
  - The V2X Message API (US), 30
- v2x\_msg\_eva\_t
  - The V2X Message API (US), 21
- v2x\_msg\_frame.h, 207
- v2x\_msg\_map, 159
  - isec\_geo\_descr, 159
  - msg\_revision, 159
  - n\_geo\_descr, 159
- v2x\_msg\_map.h, 207
  - J2735\_2016\_IsecGeoDescr\_t, 208
- v2x\_msg\_map\_decode
  - The V2X Message API (US), 30
- v2x\_msg\_map\_encode
  - The V2X Message API (US), 30
- v2x\_msg\_map\_init
  - The V2X Message API (US), 31
- v2x\_msg\_map\_t
  - The V2X Message API (US), 21
- v2x\_msg\_psm, 160
  - confidence, 160
  - cross\_request, 160
  - device\_usage\_state, 160
  - elevation, 160
  - heading, 160
  - latitude, 160
  - longitude, 161
  - msgcount, 161
  - personal\_device\_use\_type, 161
  - pos\_accuracy, 161
  - radius, 161
  - secmark, 161
  - speed, 161
  - temp\_id, 161
- v2x\_msg\_psm.h, 209
  - personal\_device\_usage\_state, 210
  - personal\_device\_usage\_state\_t, 210
  - personal\_device\_user\_type, 210
  - personal\_device\_user\_type\_t, 210
  - v2x\_msg\_psm\_t, 210
- v2x\_msg\_psm\_decode
  - The V2X Message API (US), 31
- v2x\_msg\_psm\_encode
  - The V2X Message API (US), 31
- v2x\_msg\_psm\_reset
  - The V2X Message API (US), 32
- v2x\_msg\_psm\_t
  - v2x\_msg\_psm.h, 210
- v2x\_msg\_pvd.h, 210
  - J2735\_2016\_GPSSStatus\_t, 212
  - J2735\_2016\_PHObject\_t, 212
  - J2735\_2016\_ProbeSnapshot\_t, 212
  - J2735\_2016\_VehSafetyExtensions\_t, 212
  - J2735\_2016\_VehicleStatus\_t, 212
  - v2x\_msg\_pvd\_t, 212
- v2x\_msg\_pvd\_t
  - v2x\_msg\_pvd.h, 212
- v2x\_msg\_rtcn, 161
- v2x\_msg\_rtcn.h, 213
- v2x\_msg\_rtcn\_decode
  - The V2X Message API (US), 32
- v2x\_msg\_rtcn\_encode
  - The V2X Message API (US), 33
- v2x\_msg\_rtcn\_print
  - The V2X Message API (US), 33
- v2x\_msg\_rtcn\_t
  - The V2X Message API (US), 21
- v2x\_msg\_spat, 162
  - Intersection\_list, 162
  - minute\_of\_the\_year, 162
  - n\_intersections, 162
- v2x\_msg\_spat.h, 214
  - J2735\_2016\_MovementPhaseState\_t, 217
  - \_\_attribute\_\_, 218
  - intersection\_status, 217
  - J2735\_2016\_EnableLaneList\_t, 216
  - J2735\_2016\_Intersection\_t, 216
  - J2735\_2016\_IntersectionStatus\_t, 216
  - J2735\_2016\_ManeuverAssistList\_t, 217
  - J2735\_2016\_MovementEventList\_t, 217
  - J2735\_2016\_MovementState\_t, 217
  - J2735\_2016\_TimeChange\_t, 217

- v2x\_msg\_spat\_decode
  - The V2X Message API (US), [33](#)
- v2x\_msg\_spat\_encode
  - The V2X Message API (US), [34](#)
- v2x\_msg\_spat\_t
  - The V2X Message API (US), [21](#)
- v2x\_msg\_srm.h, [219](#)
  - J2735\_2016\_PriorityReqType\_t, [221](#)
  - J2735SRMList\_t, [220](#)
  - J2735SRMRequestor\_t, [220](#)
- v2x\_msg\_srm\_t
  - The V2X Message API (US), [21](#)
- v2x\_msg\_ssm.h, [221](#)
  - J2735\_2016\_SignalRequesterInfo\_t, [222](#)
  - J2735\_2016\_SignalStatusList\_t, [222](#)
  - J2735\_2016\_SignalStatusPackageList\_t, [222](#)
- v2x\_msg\_ssm\_t
  - The V2X Message API (US), [21](#)
- v2x\_msg\_tim, [162](#)
  - dataframes, [164](#)
  - hopinfo, [164](#)
  - message\_id, [164](#)
  - msg\_crc, [164](#)
  - num\_dataframes, [164](#)
  - unique\_id, [164](#)
- v2x\_msg\_tim.h, [222](#)
  - \_\_attribute\_\_, [233](#)
  - free\_tim, [233](#)
  - J2735\_2016\_Computed\_Lanes\_t, [229](#)
  - J2735\_2016\_GeometricProjection\_t, [229](#)
  - J2735\_2016\_LatLong\_t, [229](#)
  - J2735\_2016\_NodeSet\_XY\_t, [229](#)
  - J2735\_2016\_NodeXY\_t, [229](#)
  - J2735\_2016\_NodeXY\_type\_t, [231](#)
  - J2735\_2016\_OffsetSystem\_t, [229](#)
  - J2735\_2016\_OffsetSystemType\_t, [231](#)
  - J2735\_2016\_ProjectionType\_t, [231](#)
  - j2735\_free\_tim\_contents\_only, [229](#)
  - j2735\_free\_tim\_struct, [229](#)
  - J2735CircularRegion\_t, [229](#)
  - J2735ContentType, [231](#)
  - J2735ContentType\_t, [230](#)
  - J2735DATAFRAME\_t, [230](#)
  - J2735DirectionOfUse, [231](#)
  - J2735DirectionOfUse\_t, [230](#)
  - J2735HeadingSlice, [232](#)
  - J2735MUTCDCCode, [232](#)
  - J2735MUTCDCCode\_t, [230](#)
  - J2735RadiusType, [232](#)
  - J2735RadiusType\_t, [230](#)
  - J2735Regions, [232](#)
  - J2735RegionsList\_t, [230](#)
  - J2735RoadSignID\_t, [230](#)
  - J2735ShapePoints\_t, [230](#)
  - J2735TIMGeoPath\_t, [230](#)
  - J2735TravelerInfoType, [233](#)
  - J2735TravelerInfoType\_t, [230](#)
  - J2735ValidRegion\_t, [230](#)
  - RoadSegmentRefId\_t, [231](#)
- v2x\_msg\_tim\_decode
  - The V2X Message API (US), [34](#)
- v2x\_msg\_tim\_encode
  - The V2X Message API (US), [35](#)
- v2x\_msg\_tim\_free
  - The V2X Message API (US), [36](#)
- v2x\_msg\_tim\_t
  - The V2X Message API (US), [21](#)
- v2x\_net\_btp.h, [235](#)
  - v2x\_btp\_get\_data, [235](#)
  - v2x\_btp\_get\_txreq, [235](#)
  - v2x\_btp\_packet\_len, [236](#)
  - v2x\_btp\_reserve, [236](#)
  - v2x\_btp\_reserve\_data, [236](#)
- v2x\_net\_wme.h, [236](#)
  - ADVERTISER\_LEN, [240](#)
  - channel, [245](#)
  - channel\_access, [245](#)
  - datarate, [245](#)
  - default\_gw, [245](#)
  - destmacaddr, [245](#)
  - element\_id, [245](#)
  - expiry\_time, [245](#)
  - extended\_access, [245](#)
  - FAIL, [240](#)
  - gw\_macaddr, [245](#)
  - header\_ext, [245](#)
  - immediate\_access, [246](#)
  - ipv6addr, [246](#)
  - lifetime, [246](#)
  - local\_service\_index, [246](#)
  - mac, [246](#)
  - num\_rx, [246](#)
  - P16093\_RADIO\_INTF1, [240](#)
  - P16093\_RADIO\_INTF2, [240](#)
  - P16093\_RADIO\_INTFX, [240](#)
  - P16093\_SRV\_IP, [241](#)
  - PSID\_LEN, [241](#)
  - plcp\_length, [246](#)
  - prefixlen, [246](#)
  - primarydns, [246](#)
  - priority, [246](#)
  - psc, [246](#)
  - psc\_length, [247](#)
  - psid, [247](#)
  - repeatrate, [247](#)
  - request\_type, [247](#)
  - rss, [247](#)
  - rx\_buf, [247](#)
  - rx\_buf\_length, [247](#)
  - rx\_mac, [247](#)
  - rx\_supp, [247](#)
  - SSI\_LEN, [241](#)
  - SUCCESS, [241](#)
  - safetysupp, [248](#)
  - savariwme\_cbs, [241](#)
  - savariwme\_reg\_req, [241](#)



- savariwme\_rx\_indication, [241](#)
- savariwme\_tx\_req, [241](#)
- secondarydns, [248](#)
- secondradio, [248](#)
- service\_ipv6addr, [248](#)
- service\_port, [248](#)
- srcmac, [248](#)
- srcmacaddr, [248](#)
- supp\_enable, [248](#)
- tstamp, [248](#)
- tx\_length, [248](#)
- txpower, [248](#)
- v2x\_wsmp\_close, [241](#)
- version, [249](#)
- wme\_getpsidlen, [242](#)
- wme\_process\_wme\_cmd, [242](#)
- wme\_provider\_service\_confirm, [242](#)
- wme\_register\_cch\_request, [242](#)
- wme\_register\_provider, [242](#)
- wme\_send\_wmectl, [243](#)
- wme\_set\_advertiserid, [243](#)
- wme\_set\_gpslocation, [243](#)
- wme\_set\_inactivity\_timer, [243](#)
- wme\_set\_wsaintvl, [244](#)
- wme\_set\_wsatxpower, [244](#)
- wme\_unregister\_cch\_request, [244](#)
- wme\_unregister\_provider, [244](#)
- wsatype, [249](#)
- v2x\_pf\_handle
  - The V2X Sensors API, [64](#)
- v2x\_pvd\_decode
  - The V2X Message API (US), [36](#)
- v2x\_pvd\_encode
  - The V2X Message API (US), [36](#)
- v2x\_pvd\_reset
  - The V2X Message API (US), [37](#)
- v2x\_pvd\_snapshot\_reset
  - The V2X Message API (US), [37](#)
- v2x\_sec\_cert\_change\_register
  - The V2X Security module API, [60](#)
- v2x\_sec\_cert\_change\_unregister
  - The V2X Security module API, [60](#)
- v2x\_sec\_deinit
  - The V2X Security module API, [61](#)
- v2x\_sec\_idchange\_lock
  - The V2X Security module API, [61](#)
- v2x\_sec\_idchange\_unlock
  - The V2X Security module API, [61](#)
- v2x\_sec\_init
  - The V2X Security module API, [62](#)
- v2x\_sec\_rx
  - The V2X Security module API, [62](#)
- v2x\_sec\_rx\_cblst\_t
  - The V2X Security module API, [59](#)
- v2x\_sec\_sign\_request
  - The V2X Security module API, [62](#)
- v2x\_sec\_signer\_id\_t
  - v2x\_security\_base.h, [251](#)
- v2x\_sec\_verify\_request
  - The V2X Security module API, [63](#)
- v2x\_security.h, [249](#)
- v2x\_security\_base.h, [249](#)
  - cert\_attach\_rate, [251](#)
  - check\_expiry\_time, [251](#)
  - check\_generation\_location, [251](#)
  - check\_generation\_time, [251](#)
  - check\_replay, [251](#)
  - prof\_type, [251](#)
  - sign\_profile, [251](#)
  - use\_expiry\_time, [251](#)
  - use\_generation\_loc, [251](#)
  - use\_generation\_time, [252](#)
  - v2x\_sec\_signer\_id\_t, [251](#)
  - verification\_interval, [252](#)
  - verify\_profile, [252](#)
- v2x\_security\_profile\_t
  - The V2X Security module API, [59](#)
- v2x\_security\_profile\_type\_t
  - The V2X Security module API, [60](#)
- v2x\_sensor\_can.h, [252](#)
- v2x\_sensor\_gnss.h, [252](#)
  - FAIL, [253](#)
  - pf\_gnss\_get\_nmea, [254](#)
  - pf\_gnss\_get\_pfname, [254](#)
  - SUCCESS, [253](#)
- v2x\_sign\_sec\_profile\_t
  - The V2X Security module API, [59](#)
- v2x\_spat\_free
  - The V2X Message API (US), [37](#)
- v2x\_srm\_decode
  - The V2X Message API (US), [37](#)
- v2x\_srm\_encode
  - The V2X Message API (US), [38](#)
- v2x\_srm\_reset
  - The V2X Message API (US), [38](#)
- v2x\_ssm\_decode
  - The V2X Message API (US), [38](#)
- v2x\_ssm\_encode
  - The V2X Message API (US), [39](#)
- v2x\_ssm\_reset
  - The V2X Message API (US), [39](#)
- v2x\_status\_t
  - The V2X Common API, [13](#)
- v2x\_strerror
  - The V2X Common API, [14](#)
- v2x\_verify\_sec\_profile\_t
  - The V2X Security module API, [59](#)
- v2x\_wme\_open
  - The V2X Network and Transport module API (US), [54](#)
- v2x\_wme\_reg\_req, [164](#)
  - channel, [165](#)
  - channel\_access, [165](#)
  - destmacaddr, [165](#)
  - extended\_access, [165](#)
  - immediate\_access, [165](#)

- local\_service\_index, [165](#)
- priority, [165](#)
- psc, [165](#)
- psc\_length, [165](#)
- psid, [166](#)
- repeatrate, [166](#)
- request\_type, [166](#)
- secondradio, [166](#)
- service\_ipv6addr, [166](#)
- service\_port, [166](#)
- srcmacaddr, [166](#)
- wsatype, [166](#)
- v2x\_wme\_register\_user
  - The V2X Network and Transport module API (US), [55](#)
- v2x\_wme\_unregister\_user
  - The V2X Network and Transport module API (US), [55](#)
- v2x\_wme\_user\_service\_confirm
  - The V2X Network and Transport module API (US), [56](#)
- v2x\_wsmp\_callbacks, [167](#)
  - wme\_cch\_confirm, [167](#)
  - wme\_connected, [168](#)
  - wme\_disconnected, [168](#)
  - wme\_get\_rse\_info, [168](#)
  - wme\_provider\_confirm, [168](#)
  - wme\_user\_confirm, [169](#)
  - wme\_wsm\_indication, [169](#)
- v2x\_wsmp\_callbacks\_t
  - The V2X Network and Transport module API (US), [53](#)
- v2x\_wsmp\_close
  - v2x\_net\_wme.h, [241](#)
- v2x\_wsmp\_convert\_psid\_be
  - The V2X Network and Transport module API (US), [56](#)
- v2x\_wsmp\_handler\_t
  - The V2X Network and Transport module API (US), [53](#)
- v2x\_wsmp\_recv
  - The V2X Network and Transport module API (US), [57](#)
- v2x\_wsmp\_recv\_indication, [169](#)
  - channel, [170](#)
  - datarate, [170](#)
  - num\_rx, [170](#)
  - plcp\_length, [170](#)
  - priority, [170](#)
  - psid, [170](#)
  - rsi, [170](#)
  - rx\_buf, [170](#)
  - rx\_buf\_length, [170](#)
  - rx\_mac, [170](#)
  - rx\_supp, [170](#)
  - tstamp, [170](#)
  - txpower, [171](#)
  - version, [171](#)
- v2x\_wsmp\_recv\_indication\_t
  - The V2X Network and Transport module API (US), [53](#)
- v2x\_wsmp\_send
  - The V2X Network and Transport module API (US), [57](#)
- v2x\_wsmp\_send\_req, [171](#)
  - channel, [171](#)
  - datarate, [171](#)
  - element\_id, [171](#)
  - expiry\_time, [172](#)
  - header\_ext, [172](#)
  - mac, [172](#)
  - priority, [172](#)
  - psid, [172](#)
  - safetysupp, [172](#)
  - srcmac, [172](#)
  - supp\_enable, [172](#)
  - tx\_length, [172](#)
  - txpower, [172](#)
- valid\_fps
  - \_J2735\_2016\_PHObject, [74](#)
- valid\_fullpos\_vec
  - \_J2735\_2016\_PVD, [76](#)
- valid\_out\_bound\_access\_point
  - J2735\_2016\_SignalStatusPackageList, [100](#)
- valid\_ph
  - \_J2735\_2016\_VehSafetyExtensions, [79](#)
- valid\_pos3d
  - J2735TIMGeoPath, [110](#)
- valid\_pp
  - \_J2735\_2016\_VehSafetyExtensions, [79](#)
- valid\_region
  - J2735TIMGeoPath, [110](#)
- valid\_veh\_safety\_ext
  - \_J2735\_2016\_ProbeSnapshot, [75](#)
- valid\_veh\_status
  - \_J2735\_2016\_ProbeSnapshot, [75](#)
- valid\_vehicle\_data
  - \_J2735\_2016\_VehicleStatus, [78](#)
- valid\_wiper\_set
  - \_J2735\_2016\_VehicleStatus, [78](#)
- validity\_duration
  - v2x\_etsi\_denm\_mc, [141](#)
- value
  - J2735CircularRegion, [103](#)
- veh\_class
  - \_J2735\_2016\_PVD, [76](#)
- veh\_safety\_ext
  - \_J2735\_2016\_ProbeSnapshot, [75](#)
- vehattr
  - laneattr, [113](#)
- vehicle
  - v2x\_etsi\_cam\_hf, [133](#)
- vehicle\_confidence
  - v2x\_etsi\_cam\_hf\_vehicle, [136](#)
- vehicle\_data
  - \_J2735\_2016\_VehicleStatus, [78](#)



- vehicle\_id
  - J2735\_2016\_SignalRequest, 97
- vehicle\_length
  - v2x\_etsi\_cam\_hf\_vehicle, 136
- vehicle\_status
  - \_J2735\_2016\_ProbeSnapshot, 75
- vehicle\_width
  - v2x\_etsi\_cam\_hf\_vehicle, 136
- vehicleheight
  - v2x\_msg\_bsm, 157
- vehiclemass
  - v2x\_msg\_bsm, 157
- vehicletype
  - v2x\_msg\_bsm, 157
- vendor\_frame, 173
- verification\_interval
  - aerolink\_verify\_secprofile, 83
  - v2x\_security\_base.h, 252
- verify\_profile
  - security\_cmd\_profile\_msg, 125
  - v2x\_security\_base.h, 252
- version
  - v2x\_net\_wme.h, 249
  - v2x\_wsmp\_rcv\_indication, 171
- vertaccel
  - v2x\_msg\_bsm, 157
- vertical\_accel
  - v2x\_etsi\_cam\_hf\_vehicle, 136
- vertical\_accel\_conf
  - v2x\_etsi\_cam\_hf\_vehicle, 136
- vid\_info
  - \_J2735\_2016\_PVD, 76
- vid\_info\_present
  - \_J2735\_2016\_PVD, 76
- wheelbrake
  - v2x\_msg\_bsm, 157
- wheelbrakeavailable
  - v2x\_msg\_bsm, 157
- width
  - v2x\_msg\_bsm, 157
- wiper\_front
  - \_J2735\_2016\_Wiperset, 80
- wiper\_front\_rate
  - \_J2735\_2016\_Wiperset, 80
- wiper\_rear
  - \_J2735\_2016\_Wiperset, 80
- wiper\_rear\_rate
  - \_J2735\_2016\_Wiperset, 80
- wiper\_set
  - \_J2735\_2016\_VehicleStatus, 79
- wipers\_rtfnr
  - v2x\_msg\_bsm, 158
- wipers\_rtrear
  - v2x\_msg\_bsm, 158
- wipers\_swfnt
  - v2x\_msg\_bsm, 158
- wipers\_swrear
  - v2x\_msg\_bsm, 158
- wme\_cch\_confirm
  - v2x\_wsmp\_callbacks, 167
- wme\_connected
  - v2x\_wsmp\_callbacks, 168
- wme\_disconnected
  - v2x\_wsmp\_callbacks, 168
- wme\_get\_rse\_info
  - v2x\_wsmp\_callbacks, 168
- wme\_getpsidlen
  - v2x\_net\_wme.h, 242
- wme\_process\_wme\_cmd
  - v2x\_net\_wme.h, 242
- wme\_provider\_confirm
  - v2x\_wsmp\_callbacks, 168
- wme\_provider\_service\_confirm
  - v2x\_net\_wme.h, 242
- wme\_register\_cch\_request
  - v2x\_net\_wme.h, 242
- wme\_register\_provider
  - v2x\_net\_wme.h, 242
- wme\_send\_wmectl
  - v2x\_net\_wme.h, 243
- wme\_set\_advertiserid
  - v2x\_net\_wme.h, 243
- wme\_set\_gpslocation
  - v2x\_net\_wme.h, 243
- wme\_set\_inactivity\_timer
  - v2x\_net\_wme.h, 243
- wme\_set\_wsaintvl
  - v2x\_net\_wme.h, 244
- wme\_set\_wsatxpower
  - v2x\_net\_wme.h, 244
- wme\_unregister\_cch\_request
  - v2x\_net\_wme.h, 244
- wme\_unregister\_provider
  - v2x\_net\_wme.h, 244
- wme\_user\_confirm
  - v2x\_wsmp\_callbacks, 169
- wme\_wsm\_indication
  - v2x\_wsmp\_callbacks, 169
- workzone\_codes
  - J2735TIM, 109
- wplist
  - J2735RegionPointSet, 104
  - J2735ShapePointSet, 106
- wsatype
  - v2x\_net\_wme.h, 249
  - v2x\_wme\_reg\_req, 166
- yawrate
  - v2x\_etsi\_cam\_hf\_vehicle, 136
  - v2x\_msg\_bsm, 158
- yawrate\_conf
  - v2x\_etsi\_cam\_hf\_vehicle, 136
- year
  - utcTimeFrame, 129
- zonelist
  - v2x\_etsi\_cam\_hf\_rsu, 133

zoom\_level  
    J2735\_2016\_OffsetSystem, [95](#)  
    J2735RegionPointSet, [104](#)